Available online at www.sciencedirect.com

SCIENCE DIRECT®

ELSEVIER

International Journal of Thermal Sciences 43 (2004) 925–938

International Journal of Thermal Sciences

www.elsevier.com/locate/ijts

On the judicious use of the k–ε model, wall functions and adaptivity

David Lacasse, Éric Turgeon [1], Dominique Pelletier *

Department of Mechanical Engineering, École Polytechnique de Montréal, Montréal, Québec, Canada

Received 8 January 2001; received in revised form 2 March 2004; accepted 4 March 2004

Available online 17 June 2004

Abstract

This paper shows that accurate predictions of skin friction and Stanton number for a flat plate boundary layer can be achieved with the k–ε model of turbulence with wall functions, provided the computational model (turbulence model, geometry, boundary conditions, etc.) is properly defined and that the simulation is grid converged. Emphasis is put on good CFD practices such as verification and validation. Verification allows to obtain numerical predictions with controlled accuracy. Validation of the computational model is thus performed on solid grounds. Predictions of the local skin friction coefficient on the plate were obtained for Reynolds numbers of 2×10^5 and 2×10^6. In the case $Re = 2 \times 10^5$, the effects of inlet boundary turbulence Reynolds number, geometry of the plate and type of wall functions are assessed. It is found that an inlet turbulence Reynolds number equal to 10% of the Reynolds number yields realistic results. Significant improvements are achieved if the plate thickness is included in the computational model. Two-velocity scale wall functions prove to be superior to the more popular one-velocity scale wall functions. For the case $Re = 2 \times 10^6$, predictions of the local skin friction coefficient ($C_f(x)$) and the local heat transfer coefficient ($St(x)$) are in good agreement with correlations.

Keywords: Turbulent flow; Flat plate; Computational model; Wall functions; k–ε turbulence model; Adaptive finite elements; Verification; Validation

1. Introduction

The subject of turbulent flow over slender bluff bodies, such as a flat plate, is a topic of interest to many engineers. They usually tackle this problem with commercial software that often implement the k–ε model with wall functions or a variant. Modeling and computational decisions that they make, as well as the careful meshing of the computational domain, have a direct impact on the accuracy of the numerical predictions.

The objective of the present paper is to control the numerical precision of the solution so as to eliminate the mesh as a source of error. Once the accuracy is verified, validation and comparison of numerical models may be performed on a more solid basis. Under such circumstances, the discrepancies between predictions and measurements are due to modeling errors (turbulence model, geometry, boundary conditions, etc.). Such issues are of concern to a broad spectrum of fluid dynamics problems and have been observed in many instances. In fact, accurate and reliable prediction of turbulent flows has been the subject of much research by the CFD community over the past few years. A review of the literature reveals that in many cases, for a given flow, predictions by different authors show an unacceptable amount of scatter. At times this gets even more disconcerting given that people using similar models and numerical algorithms produce vastly differing predictions. Table 1 presents the predicted length of the recirculation zone for turbulent flow over a backward facing step. All authors use a variant of the k–ε model with wall functions and a TEACH type solution algorithm [1]. The only exceptions are the predictions of Donaldson, who used a Reynolds Stress Model, and that of Ilinca et al. obtained with an adaptive finite element method. The scatter between predictions is even worse for turbulent heat transfer predictions. Table 2 presents the maximum Nusselt number downstream of a sudden pipe expansion. These results were taken from Launder [2]. Hutton et al. collected the original data [3]. The largest predicted value is five times bigger than the lowest value. The lowest predicted value is in error by

---

* Corresponding author.
E-mail address: dominique.pelletier@polymtl.ca (D. Pelletier).

[1] Current address: Pratt & Whitney Canada.

1290-0729/$ – see front matter © 2004 Elsevier SAS. All rights reserved.
doi:10.1016/j.ijthermalsci.2004.03.004

Nomenclature

$C_f(x)$	local skin friction coefficient
$C_1, C_2, C_\mu, \sigma_k, \sigma_\varepsilon$	constants for the k–ε model
c_p	specific heat at constant pressure
d	distance between solid wall and computational boundary
E	roughness parameter
$\mathcal{E}$	natural logarithm of ε
$\mathcal{K}$	natural logarithm of k
L	plate length
Nu	Nusselt number
p	pressure
Pr	Prandtl number
q	heat flux
Re	Reynolds number
$St(x)$	Stanton number
T	temperature
TKE	turbulence kinetic energy
U	free-stream velocity
$\mathbf{u}$	velocity vector
u, v	horizontal and vertical velocity components
u_*	friction velocity
u_{**}	velocity scale from the two-velocity scale wall function
u_k	velocity scale based on TKE
x, y	Cartesian axes
k	turbulence kinetic energy (TKE)
ε	dissipation rate of TKE
κ	Kármán Constant
λ	thermal Conductivity
μ	viscosity
ρ	density
τ	shear stress

Subscripts

o	inlet boundary
t	turbulent
trans	transition laminar-turbulent
w	wall value
x	based on plate length
∞	free-stream value

Superscript

$+$	dimensionless (wall functions)

Table 1
Flow over a backward facing step

Reference	Turbulence model	L/H
Kim et al.	experiment	7 ± 0.5
Mansour and Morel	k–ε	5.2
Pollard	k–ε	5.88
Rodi et al.	k–ε	5.8
Launder et al.	ASM	6.9
Abdelmeguid et al.	k–ε	6
Demirdzic et al.	modified k–ε	6.2
Donaldson et al.	RSM	6.1
Ilegbusi and Spalding	modified k–ε	7.2
Nallasamy and Chen	k–ε	5.8
Syed et al.	k–ε	5.8
Ilinca et al.	k–ε	6.2

Table 2
Maximum Nusselt number downstream of a sudden pipe expansion

Author	$Nu_{\max}$
Numerical #1	1660
Numerical #2	375
Numerical #3	1205
Numerical #4	1330
Numerical #5	915
Numerical #6	2036
Numerical #7	574
Numerical #8	1440
Numerical #9	921
Numerical #10	943
Numerical #11	975
Experimental	932

50% while the highest prediction is in error by more than 100%. According to Launder the main source of error is the near wall model. However, mesh size and artificial diffusion due to upwind discretization of convective terms provide an uncontrolled and often non-negligible source of error.

While turbulence modeling issues are still a topic of hot debate, numerical and discretization issues can now be addressed in a rigorous and systematic manner so as to minimize their impact on the uncertainty of predictions. Adaptive methods are a powerful tool to control numerical errors. It is now possible to obtain “numerically exact” solutions to the differential equations so that computational modeling issues can be studied and evaluated with confidence.

Roache proposes a two-step approach in order to distinguish mathematical modeling errors and numerical errors [4]. The first step is called *verification*. Simply stated, it provides an answer to the question: *Are we solving the equations right?* Questions of numerical accuracy are at the heart of the verification process. The second step is called *validation* and provides answers to the question: *Are we solving the right equations for this problem?* The key question in validation is one of *the suitability of the computational model to accurately represent the physical process of interest*.

This paper proposes an adaptive finite element method as an efficient tool for performing both verification and validation. This methodology has already proved its ability to produce high quality and very accurate solutions to a wide variety of problems. Initial efforts were focused on laminar isothermal flows [5,6], turbulent incompressible flows [7–9], and compressible flows [10]. Applications to laminar heat transfer have also been presented [11,12], including conjugate and compressibility effects [13,14]. Applications to turbulent heat transfer may be found in [15–17]. Using the adaptive methodology, the paper presents an example of code verification by the method of manufactured solution. Examples of successful and unsuccessful validation show that agreement with experiments or correlations is achieved only with a good computational model of flow physics combined with accurate numerical solution of the differential equations. The paper emphasizes good CFD practice to systematically achieve verification so that validation studies are always performed on solid grounds.

Adaptive methods provide a powerful approach to handle flows with such a spectrum of behaviors within the flow field. Automatic grid point clustering in regions of rapid variation of the solution ensures accurate resolution in all parts of the flow. Mesh adaptation can yield "numerically exact" (grid independent) solutions. The error estimator allows for quality control of the solution. The adaptive strategy provides a simple means of quantifying the convergence of an adaptive grid refinement study. Our previous work on adaptive finite element methods has described both verification and validation computations for a variety of problems of practical interest [7–9,15,16,18].

The paper is organized as follows. Definitions of Verification and Validation are given in the first section, which is followed by a presentation of the equations modeling the problem. Then, we simulate isothermal flow on a flat plate and assess the accuracy of results by looking at the skin friction coefficient along the plate. White's correlation [19] is used as a reference because of its accuracy and reliability. The effects of the inflow's turbulence Reynolds number, the type of wall functions and the geometry of the plate are studied in turn for a Reynolds number of 2×10^5 to show their impact on the accuracy of predictions.

A heat transfer variant of the flow is then presented, for a Reynolds number of 2×10^6. Predictions of the local skin friction coefficient and Stanton number along the plate are presented and compared to White's correlation [19] and the Kays–Crawford correlation [20].

2. Definitions

First and foremost we must make the essential distinction between Verification and Validation. In a common English thesaurus, verification and validation are synonymous. However, in CFD these two words have acquired a generally accepted *technical meaning* which is provided in the *specific technical context* of CFD. We follow accepted definitions [4,21–23] and adopt the succinct description of *Verification* as *solving the equations right*, and of *Validation* as *solving the right equations*.

For Verification, the code author defines precisely what partial differential equations and boundary conditions are being solved and convincingly demonstrates that they are solved correctly (i.e., with some order of accuracy) and always consistently so that, as the mesh size tends to zero, the code produces a solution to the continuum equations. Whether or not those equations and that solution bear any relation to a physical problem of interest to the code users is the subject of Validation. Thus in a meaningful but scrupulous sense, one cannot validate a "code". The best one can do is validate a simulation or perhaps a range of calculations for a well defined class of problems.

Another way to make the distinction between Verification and Validation is to speak of numerical errors versus conceptual modeling errors. An example is the assumption of incompressibility. For instance, dynamic stall of helicopter rotor blades entails compressibility effects at surprisingly low free-stream Mach number. Results from an incompressible flow code will likely not agree with experimental data. However, one cannot claim that code failed verification because it was applied to a compressible flow. In this case, the lack of agreement with data is not a code problem. It is a modeling problem: the user chose the wrong model for his flow.

Another way of distinguishing Verification from Validation is to follow the classical distinction between mathematics and engineering science. Verification is strictly an activity in the mathematics of numerical analysis. It answers the question *Are we doing good numerical analysis to solve the differential equations at hand?* Validation is essentially and strictly an activity in engineering science. It answers the question *Are we doing good engineering modeling for the problem of interest?* This distinction is further enhanced by looking at the IEEE definition of code Verification [24]: *"Formal proof of program correctness"*. We agree with Oberkampf's evaluation of this terse definition [25]: *"Although brief, this definition brings unprecedented clarity to the meaning of the term, and it adds a new perspective to the issue. Specifically, this definition bluntly requires correctness or veracity of prediction, without bringing in supportive topics as to what is being predicted or how it is done"*. While more general, the IEEE definition is compatible with the one used here, and is also compatible with the distinction between Verification and Validation. That is, *program correctness* for a PDE code would naturally include *solving the equations right*, and of course a definition of what those continuum equations are, without getting into the question of whether certain problems are appropriate for those equations and that code, i.e., Validation. In other words Verification and Validation are separate steps.

Questions of numerical accuracy are at the heart of the Verification process. Thus, both the code and individual simulations must be verified. Verification of a code involves error estimation from a known solution, whereas verification of a specific calculation involves error estimation or banding to ensure that problem specification does not prevent the code from delivering the expected accuracy. See Roache [4] for cases where a verified code may deliver non-verified simulations.

The key question in Validation is centered on *the suitability of the computational model to accurately represent the physical process of interest*. The computational model includes the PDEs solved, boundary conditions, the geometry, the turbulence model, physical properties, etc. Predictions are compared to experiments to determine the degree of accuracy to which the model represents reality. Again, this is a matter of physics and engineering, not of mathematics. Any Validation exercise looses its significance and credibility if prior Verifications (of the code and the calculations) are not performed. The rule is: verify first, validate next. Roache recommends systematic grid refinement studies for structured non-adaptive meshes combined with Richardson extrapolation as a means of performing verification studies. In this paper, adaptive remeshing is used as a cost-effective alternative that automates the tedious process of manual generation of finer meshes.

To many people, Validation simply consists in comparing predictions to experimental measurements. In practice, Validation is a more difficult exercise than one would expect. One must start with *good* CFD (verified) predictions and compare them to *good* experimental data. The first difficulty to overcome is best described by the following saying in the aerodynamics community:*“No one believes the CFD prediction except the one who performed the calculation, and every one believes the experimental data except the one who performed the experiment”*. Second, good data is difficult to obtain due to experimental errors. For instance, wind tunnels suffer from flow angularity and blockage effects which are further complicated because they vary with angle of attack. Furthermore, experimental data must be interpreted with care because of the possible sources of errors: calibration errors, data acquisition errors, data reduction errors, test technique errors etc. [26].

In general, most experiments were never designed for CFD Validation. As a result some data critical to CFD is often missing: geometry, boundary conditions, initial conditions etc. [4]. Aeschliman et al. [27] also report that *“as one progresses down the list to more difficult quantities for CFD to predict, the experimental uncertainty generally increases also”*. This can result in false invalidation (failed Validation) or false Validation. Wilcox [28] describes a case where the data were incomplete and lead to a false invalidation. Aeschliman et al. [27] describe a case for which both the CFD and experiments contained serious errors, yet agreed extremely well with each other!

The flat plate flow variants presented in this paper were selected to illustrate a range of situations. All cases involve verified computations. Examples show that a verified code may lead to successful validation in one situation and failed validation in another. This highlights the delicate and difficult nature of verification and validation activities.

3. Modeling of the problem

3.1. Reynolds-averaged Navier–Stokes equations

The problem of interest in the present study is modeled by the time-averaged continuity, momentum and energy equations:

$$\nabla \cdot \mathbf{u} = 0$$

$$\rho \mathbf{u} \cdot \nabla \mathbf{u} = -\nabla p + \nabla \cdot \left[(\mu + \mu_t)\left(\nabla \mathbf{u} + \nabla \mathbf{u}^{\mathrm{T}}\right)\right]$$

$$\rho c_p \mathbf{u} \cdot \nabla T = \nabla \cdot \left[(\lambda + \lambda_t)\nabla T\right]$$

where ρ is the density, $\mathbf{u}$ the velocity, p the pressure, μ the viscosity, μ_t the eddy viscosity, c_p the specific heat, T the temperature, λ the conductivity and λ_t the eddy conductivity. This system of equations is closed with the k–ε turbulence model [29].

3.2. The standard k–ε turbulence model

The eddy viscosity and conductivity are expressed in terms of two turbulence variables, the turbulence kinetic energy k and its rate of dissipation ε:

$$\mu_t = \rho C_\mu \frac{k^2}{\varepsilon}$$

$$\lambda_t = \frac{c_p \mu_t}{Pr_t}$$

The turbulent Prandtl number Pr_t is equal to unity.

The turbulence quantities are governed with the following transport equations:

$$\rho \mathbf{u} \cdot \nabla k = \nabla \cdot \left[\left(\mu + \frac{\mu_t}{\sigma_k}\right)\nabla k\right] + \mu_t P(\mathbf{u}) - \rho\varepsilon \tag{1}$$

$$\rho \mathbf{u} \cdot \nabla \varepsilon = \nabla \cdot \left[\left(\mu + \frac{\mu_t}{\sigma_\varepsilon}\right)\nabla \varepsilon\right] + C_1 \frac{\varepsilon}{k}\mu_t P(\mathbf{u}) - C_2 \rho \frac{\varepsilon^2}{k} \tag{2}$$

The production of turbulence $P(\mathbf{u})$ is defined as:

$$P(\mathbf{u}) = \nabla \mathbf{u} : \left(\nabla \mathbf{u} + \nabla \mathbf{u}^{\mathrm{T}}\right)$$

The constants C_1, C_2, C_μ, σ_k, and σ_ε are set to the values recommended by Launder and Spalding [29] and listed in Table 3.

3.3. Wall boundary conditions

The standard k–ε turbulence model is not valid for low values of the turbulence Reynolds number (e.g., the near-wall region). The strategy adopted here uses wall functions

Table 3
Constants for the k–ε model

C_μ	C_1	C_2	σ_k	σ_ε
0.09	1.44	1.92	1.0	1.3

which describe the solution in the near wall region. This section describes two such wall functions which are used as boundary conditions.

3.3.1. *One-velocity scale wall function*

A wall function expresses the velocity tangent to the solid wall u as a function of the distance from the wall d in terms of the dimensionless variables u^+ and y^+:

$$u^+ = y^+ \qquad \text{for } y^+ < y_c^+$$

$$u^+ = \frac{1}{\kappa}\ln(Ey^+) \qquad \text{for } y^+ \geqslant y_c^+$$

where κ is the Karman constant and E is a roughness parameter, $\kappa = 0.42$ and for smooth walls, $E = 9.0$. The critical value y_c^+ separating the linear and logarithmic velocity profiles is approximately 10.8 [20]. For the one-velocity scale wall law, u^+ and y^+ are defined as follows:

$$u^+ = \frac{u}{u_*}$$

$$y^+ = \frac{d\rho u_*}{\mu}$$

The velocity scale u_*, called the friction velocity, is obtained from the value of the wall shear stress τ_w:

$$u_* \equiv \sqrt{\frac{\tau_w}{\rho}}$$

The combination of the logarithmic wall function and the definition of u_* results in a non-linear mixed (Robin) boundary condition for u:

$$u = \sqrt{\frac{\tau_w}{\rho}}\frac{1}{\kappa}\ln\left(Ed\rho\frac{\sqrt{\tau_w/\rho}}{\mu}\right)$$

that involves both the tangent velocity u and its derivative since

$$\tau_w = (\mu + \mu_t)\,\frac{\partial u}{\partial y}$$

Of course, the specific value of d where the boundary condition is applied is chosen so that y^+ lies within a safe range of validity of the logarithmic wall function. *A posteriori* computations of y^+ are always performed to make sure that $30 < y^+ < 300$.

Fig. 1 shows the different discretizations of the computational domain close to the wall, for finite elements and finite volumes. In finite elements, the mesh does not reach the wall. In finite volumes, a wall function cell is used instead. Its height d must also satisfy $30 < y^+ < 300$. However, because the mesh reaches the wall, both u and v are set to zero at the wall. Wall functions determine the values of the velocity at the first nodes off the wall. While simplifying the CFD, meshing to the wall imposes non-trivial constraints on mesh generation to ensure that $30 < y^+ < 300$.

Once the velocity distribution is known throughout the domain, the distribution of the turbulence kinetic energy and its dissipation rate are computed by solving equations (1) and (2) with the following Dirichlet boundary conditions [29]:

$$k = \frac{\tau_w}{\rho\sqrt{C_\mu}}$$

$$\varepsilon = \frac{u_*^3}{\kappa d}$$

3.3.2. *Two-velocity scale wall function*

The one-velocity scale wall function predicts zero heat flux near a reattachement point because the friction velocity u_* is not the appropriate scale as it vanishes at a stagnation point [30–32]. This contradicts observations. Near a stagnation point, turbulence fluctuations are responsible for heat transfer. The following definition provides a velocity scale based on the turbulence kinetic energy [33]:

$$u_k \equiv C_\mu^{1/4}k^{1/2}$$

The second velocity scale u_{**} is defined as follows:

$$u_{**} \equiv \frac{\tau_w}{\rho u_k}$$

The definitions of u^+ and y^+ are then:

$$u^+ = \frac{u}{u_{**}}$$

$$y^+ = \frac{d\rho u_k}{\mu}$$

Using the logarithmic wall function and the definitions of u_k and u_{**}, one obtains a mixed boundary condition for u which is linear:

$$u = \frac{\tau_w}{\rho u_k}\frac{1}{\kappa}\ln\left(E\frac{d\rho u_k}{\mu}\right) \tag{3}$$

The k-equation is solved with a zero gradient ($\partial k/\partial \mathrm{n} = 0$) wall boundary condition. This condition arises from the

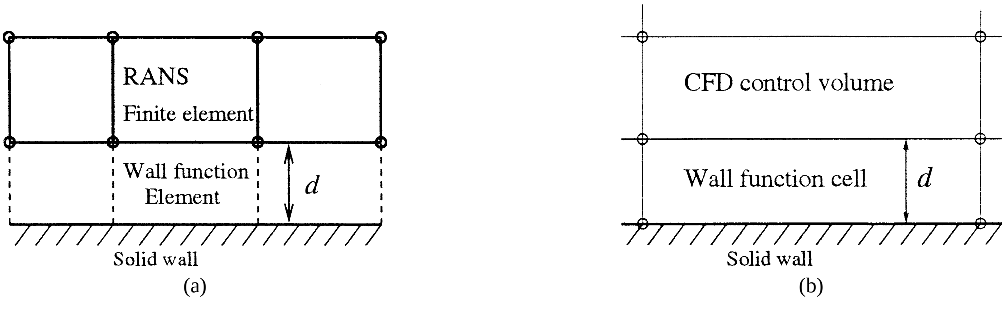


Fig. 1. Discretization close to the wall: (a) Finite elements; (b) Finite volumes.

fact that the wall shear stress is considered constant in the region $0 < y^+ < \pm 300$ [34] ($\tau_w = \rho\sqrt{C_\mu}k$). Therefore, $\partial\tau_w/\partial\mathrm{n} = \partial k/\partial\mathrm{n} = 0$. This Neumann condition is required to compute the distribution of u_k along the boundary and to determine the velocity boundary condition. Finally, the boundary condition for ε is given by:

$$\varepsilon = \frac{u_k^3}{\kappa d}$$

The use of the velocity scale u_k enables physically realistic predictions of the heat flux near stagnation points where the shear stress is zero but the heat flux is not.

3.3.3. *Thermal wall function*

Boundary conditions at a wall for the energy equation are enforced through a temperature wall function similar to that used for the momentum equations. For constant wall temperature, the effective heat flux in the wall function is computed as

$$q_w = \frac{\rho c_p C_\mu^{1/4} k_w^{1/2}(T_w - T)}{T^+}$$

where T and k are the temperature and the TKE of points located on the boundary of the computational domain; T_w is the wall temperature and T^+ satisfies the following relations [31]:

$$T^+ = \begin{cases} Pr\, y^+ & \text{for } y^+ < y_1^+ \\ a_2 - \frac{Pr_t}{2a_1(y^+)^2} & \text{for } y_1^+ \leqslant y^+ < y_2^+ \\ \frac{Pr_t}{\kappa}\ln(y^+) + \beta & \text{for } y_2^+ \leqslant y^+ \end{cases}$$

with the following definitions:

$$y_1{}^+ = \frac{10}{Pr^{1/3}}, \qquad y_2{}^+ = \left(\frac{\kappa}{a_1}\right)^{1/2}$$
$$a_1 = 10^{-3} Pr_t, \qquad a_2 = 15 Pr^{2/3}$$
$$\beta = a_2 - \frac{Pr_t}{2\kappa}\left[1 + \ln\left(\frac{\kappa}{a_1}\right)\right]$$

The Prandtl and turbulent Prandtl numbers are set respectively to 0.7 and 1.

3.4. *Turbulence equations in logarithmic form*

While mathematically correct, the turbulence equations (1) and (2) may cause numerical difficulties. For example, the eddy viscosity may become negative if ε becomes negative, causing the solver to breakdown. Also, several source terms contain division by a turbulence variable. Negative or small values of the denominator can cause improper sign or overly large values of the source terms. Enhanced robustness of the algorithm is achieved if one can ensure that turbulence variables remain positive throughout the domain and the course of iterations.

One way to preserve positivity of the dependent variables consists in solving for their logarithms [35]. This can be viewed as using the following change of dependent variables:

$$\mathcal{K} = \ln(k), \qquad \mathcal{E} = \ln(\varepsilon)$$

Solving for $\mathcal{K}$ and $\mathcal{E}$ guarantees that k and ε will remain positive throughout the computations. Hence, the eddy viscosity μ_t will always remain positive. This approach is referred to as solving for *logarithmic variables*. Furthermore, turbulence quantities often present very steep fronts which are difficult to resolve accurately. Logarithmic variables $\mathcal{K}$ and $\mathcal{E}$ present smoother variations than those of k and ε because the logarithm varies more slowly than its argument. Thus, more accurate solutions are obtained when logarithmic variables are used.

The transport equations for the logarithmic variables are:

$$\rho\mathbf{u}\cdot\nabla\mathcal{K} = \nabla\cdot\left[\left(\mu + \frac{\mu_t}{\sigma_k}\right)\nabla\mathcal{K}\right] + \left(\mu + \frac{\mu_t}{\sigma_k}\right)(\nabla\mathcal{K})^2 + \mu_t e^{-\mathcal{K}} P(\mathbf{u}) - \rho e^{\mathcal{E}-\mathcal{K}}$$

$$\rho\mathbf{u}\cdot\nabla\mathcal{E} = \nabla\cdot\left[\left(\mu + \frac{\mu_t}{\sigma_\varepsilon}\right)\nabla\mathcal{E}\right] + \left(\mu + \frac{\mu_t}{\sigma_\varepsilon}\right)(\nabla\mathcal{E})^2 + C_1\mu_t e^{-\mathcal{K}} P(\mathbf{u}) - C_2\rho e^{\mathcal{E}-\mathcal{K}}$$

Note that the equations for logarithmic variables are equivalent to the original equations of the turbulence model. Hence, the turbulence model is unchanged. Only the computational variables are different.

3.5. *Finite element formulation and adaptive remeshing*

The Navier–Stokes and energy equations, along with the logarithmic form of the turbulence equations are solved by a finite element method. The equations are multiplied by a test function and integrated over the domain. Integration by parts of diffusion terms and pressure gradients leads to the Galerkin variational formulation. Because the equations are dominated by convection, the standard Galerkin discretization of such equations may lead to non-physical oscillations and convergence difficulties. These situations are mitigated by using one of the following stabilized formulations: SU [36], SUPG [37], or GLS [38]. The equations are discretized using the seven-node triangular element [39], which uses an enriched quadratic velocity field, a linear discontinuous pressure and a quadratic interpolant for the temperature and the logarithms of turbulence variables.

The equations are solved in the following partly segregated manner:

1. For a given field of μ_t, solve momentum, continuity and energy equations;
2. Solve TKE equation;
3. Solve the dissidation equation;
4. Repeat from step 1.

Faster convergence is achieved by repeating steps 2 and 3 within a global iteration. Wall functions are solved implicitly at all steps.

Mesh independent solutions of the equations are obtained with an adaptive finite element algorithm [40]. Error estimates are obtained by a local least squares reconstruction of the solution derivatives for the velocity field, the temperature and the logarithms of turbulence variables. An error estimate for the eddy viscosity is also constructed since slowly varying fields of $\mathcal{K}$ and $\mathcal{E}$ may result in rapid variation of μ_t [35]. The adaptive methodology is set to reduce the error by a factor of two at each cycle of adaptation.

4. Verification

This section presents one example of verification of the adaptive finite-element method. We use the Method of Manufactured Solution proposed by Roache [4]. The availability of an analytical solution provides a rigorous framework for assessment of solution accuracy. The following expressions are used as a solution:

$$
\begin{aligned}
U &= \frac{U_1+U_2}{2} - \frac{U_1-U_2}{2}\operatorname{erf}(\eta)\\
V &= \frac{U_1-U_2}{2}\frac{1}{\sigma\sqrt{\pi}}B\\
T &= \frac{T_1+T_2}{2} - \frac{T_1-T_2}{2}\operatorname{erf}(\eta)\\
p &= 0\\
k &= k_0(c_k+B)\\
\varepsilon &= \frac{\varepsilon_0}{x}(c_k+B)\\
\mu_t &= \mu_{t_0}x(c_k+B)
\end{aligned}
$$

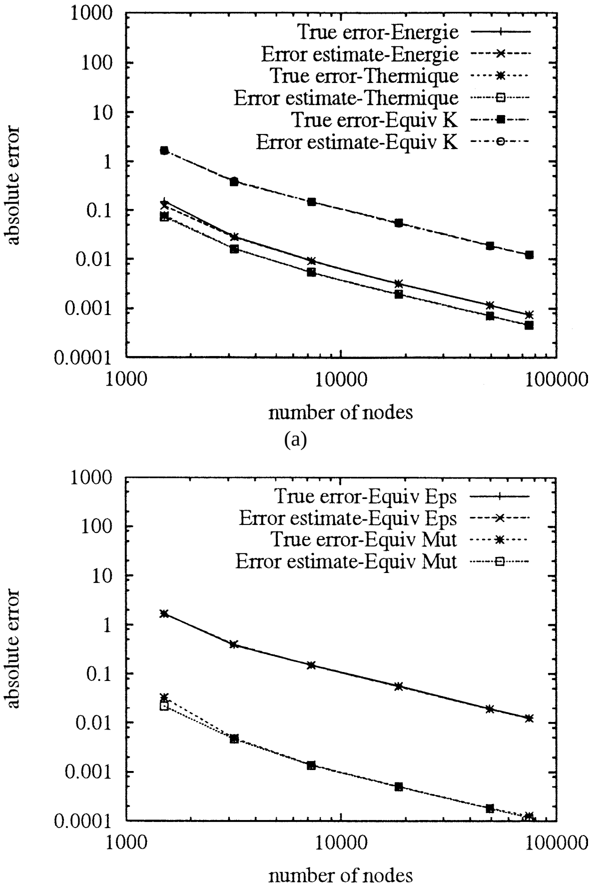
where

$$
\begin{aligned}
&\eta = \frac{\sigma y}{x}, \qquad B = e^{-\eta^2}\\
&U_1 = 1, \qquad U_2 = 0.3\\
&T_1 = 0.3, \qquad T_2 = 1\\
&\sigma = 13.5, \qquad \rho = 1.0\\
&C_\mu = 0.09, \qquad \mu = 10^{-4}\\
&k_0 = \frac{343}{75000}U_1(U_1-U_2)\frac{\sigma}{\sqrt{\pi}}\\
&\varepsilon_0 = \frac{343}{22500}C_\mu U_1(U_1-U_2)^2\frac{\sigma^2}{\pi}\\
&c_k = \frac{10^{-4}}{k_0}, \qquad \mu_{t_0} = \frac{343}{250000}\rho U_1
\end{aligned}
$$

Appropriate source terms are obtained by substituting the above expressions for u, v, p, k, ε, and μ_t into the Navier–Stokes, energy and turbulence equations. While any expression can be used, the above choice mimics a thermal free shear layer. Equations are solved over the domain $(100,-75)\times(300,75)$. Dirichlet boundary conditions are applied everywhere, except on a portion of the bottom boundary, where a zero normal traction is specified. This ensures that mass is conserved and improves convergence over our previous attempt with an all-Dirichlet approach [32]. Following Turgeon et al. [41], computations are performed using a GLS stabilized finite element scheme for all equations (momentum, continuity, energy, turbulence). The adaptive remeshing strategy is set to reduce the global error by a factor of 2 at each adaptive cycle. All dependent variables contribute to mesh refinement.

Fig. 2 shows the grid convergence of the flow solution. The error decreases at each cycle of mesh adaptation indicating that grid convergence is occurring. Notice also that agreement between the error estimates and the true errors improves with mesh adaptation. The estimators exhibit asymptotic exactness with grid refinement, as confirmed by Table 4, which shows efficiency indices of the error estimators close to 1 (ratio of the estimator to the true error). Asymptotic exactness guarantees sharpness (reliability) of the estimator with mesh refinement.

Error trajectories confirm that the finite element scheme exhibits second order accuracy for the derivatives of all



(a)

(b)

Fig. 2. Grid convergence for the flow: (a) $u, T, \mathcal{K}$; (b) $\mathcal{E}, \mu_t$.

Table 4
Shear layer: efficiencies on final mesh

u	T	$\mathcal{K}$	$\mathcal{E}$
0.9988	1.018	0.9792	0.9769

dependent variables. This verifies the performance, in the sense of Roache [4], and accuracy of the solver and error estimators.

5. Isothermal flat plate

The verified code is now applied to turbulent flow on an infinitely thin plate. The domain and boundary conditions are defined as shown in Fig. 3. Computations are performed in dimensionless form using the plate length and inlet velocity as reference quantities. The Reynolds number is set to 200 000 (selected as viable while at the same time minimizing meshing requirements).

5.1. *Effect of the inflow turbulence Reynolds number*

The realism of the flow solution may depend on a number of user specified input parameters. Among those are the distance to the wall in the wall functions and the inflow values of the turbulence kinetic energy and its dissipation rate. This section shows that proper selection of inflow values of turbulence variables is critical to achieving physically realistic predictions. In this section, a two-velocity scale wall function is used along the plate. Similar trends were observed with a one-velocity scale wall function.

The inlet turbulence Reynolds number characterises the flow and is defined as:

$$Re_t = \frac{\varepsilon_o}{\rho C_\mu k_o^2}$$

where k_o and ε_o are the inlet boundary conditions.

Following previous success in the prediction of internal flows [16], the turbulence Reynolds number at the inflow was set to 1000 with the values of k_o and ε_o given in Table 5. This low value of the inlet Re_t results in a rather large eddy viscosity which dissipates the TKE. Contours of TKE for $Re_t = 1000$ are shown in Fig. 4(a). The decay of grid turbulence outside of the boundary layer is clearly seen. The high values of inflow boundary conditions of k and ε also have a significant non-physical side effect: the value of μ_t is lower in the boundary layer than in the free stream flow, as illustrated in Fig. 5(a). Fig. 5(a) shows that grid independent results are easily achieved on the first adapted mesh. This simulation is thus verified in the sense of Roache. However, it is not validated as it presents an unphysical behavior of μ_t.

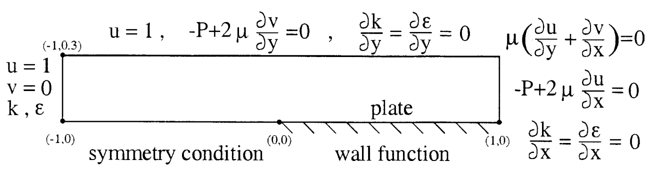
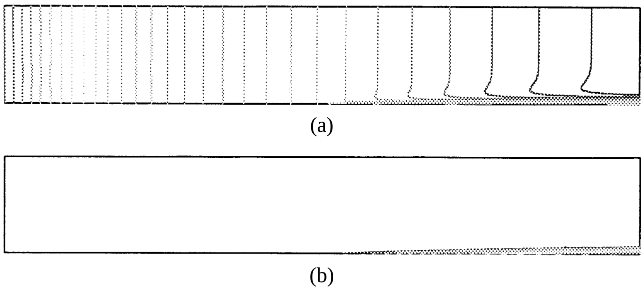


Fig. 3. Domain and boundary conditions.

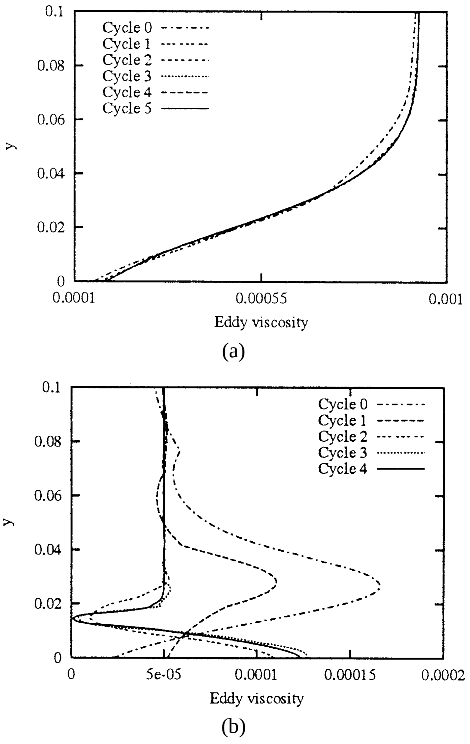

(a)

(b)

Fig. 4. Contours of TKE: (a) $Re_t = 1000$; (b) $Re_t = 20\,000$.

A higher turbulence Reynolds number must be specified in order to make the computational model physically more realistic. Re_t is now set to 20 000 (10% of Re) with the values of k_o and ε_o given in Table 5. The freestream turbulence is now very low compared to the levels of TKE in the boundary layer, so that the decay of grid turbulence does not show on Fig. 4(a).

Fig. 5(b) is a very good example of the importance of achieving grid independent solutions. The behavior of the eddy viscosity (μ_t) changes dramatically between the initial and the final mesh, where it is more along expected trends. Note that four cycles of adaptive remeshing are needed to achieve a grid independent verified solution. The final meshes for the cases $Re_t = 1000$ and $Re_t = 20\,000$ are



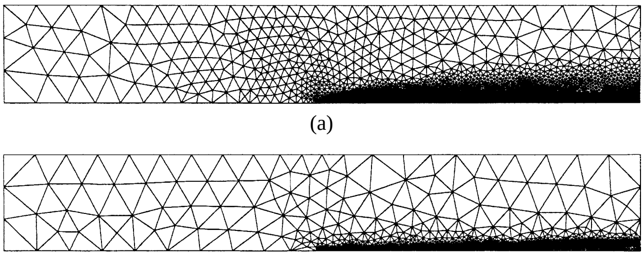
(a)

(b)

Fig. 5. Variation of μ_t in the boundary layer at $x/L = 0.5$: (a) $Re_t = 1000$; (b) $Re_t = 20\,000$.

Table 5
Inlet values of k and ε

Re_t	k_o	ε_o
1000	10^{-2}	9×10^{-3}
20 000	10^{-6}	1.8×10^{-9}


(a)

(b)

Fig. 6. Final meshes of the adaptive remeshing procedure: (a) $Re_t = 1000$; (b) $Re_t = 20\,000$.

shown in Fig. 6. Increasing Re_t yields a thinner boundary layer. Fig. 6 shows that the meshes are adapted accordingly.

We now make a more quantitative assessment of predictions by looking at skin friction predictions.

5.2. Effect of the type of wall functions

One and two velocity wall functions are used in turn to predict skin friction over an infinitely thin plate (zero thickness). The local skin friction coefficient, defined as:

$$C_f(x) = \frac{\tau_w}{\frac{1}{2}\rho U^2} \quad \text{where } \tau_w = \mu\left(\frac{\partial u}{\partial y}\right)\bigg|_{y=0}$$

is used to assess the predictions.

Following results from the previous sections, the turbulence Reynolds number is set to 20 000. The predicted values of $C_f(x)$ are compared to White's correlation:

$$C_f(x) = \frac{0.455}{\ln^2(0.06Re_x)}$$

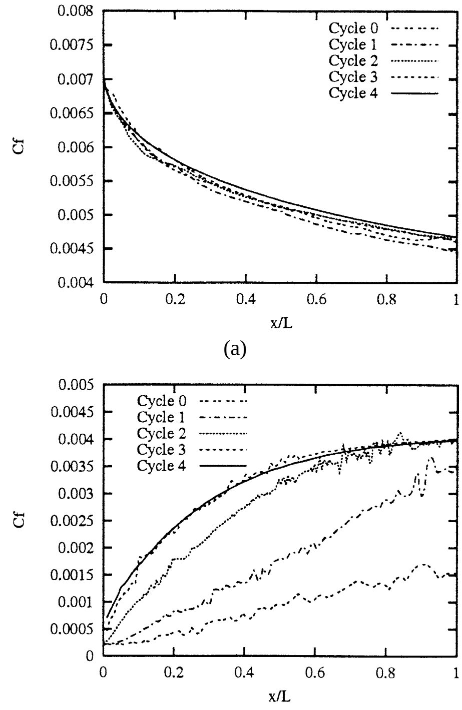
which agrees with measurements to ±2% over the entire turbulent range [19]. This comparison amounts to validation of FEM predictions.

5.2.1. One velocity scale

The distribution of $C_f(x)$ along the plate is computed from the one-velocity scale wall function boundary condition for k:

$$C_f(x) = \frac{k\sqrt{C_\mu}}{\frac{1}{2}U^2} \tag{4}$$

Fig. 7(a) shows distributions of $C_f(x)$ obtained for each of the adaptive cycles. Numerical instabilities have necessitated the use of different stabilization formulations (SU, SUPG, GLS) for the turbulence equations. Converged solutions could not be achieved with the same formulation for all adaptive cycles. This may explain why we do not see a clear evolution of the distribution of $C_f(x)$ towards a grid independent solution, even though the differences between cycles are small. However, predictions are so smooth that we believe they are useful in assessing the results. The 'Cycle 4' solution is plotted in Fig. 8. As can be seen, the agreement with the correlation is not good. This simulation fails validation in the sense of Roache.



(a)

(b)

Fig. 7. Evolution of the distributions of $C_f(x)$ with adaptive cycles: (a) One velocity scale, (b) Two velocity scales.

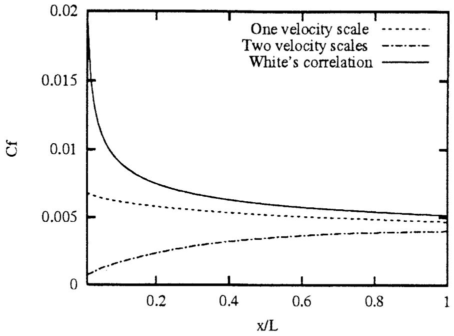


Fig. 8. Distributions of $C_f(x)$ on an infinitely thin plate.

5.2.2. Two velocity scales

With two velocity scales, the local skin friction coefficient is computed in the following manner:

$$C_f(x) = \frac{u_k u}{\frac{1}{2}U^2 u^+} \quad \text{since } \tau_w = \rho u_k u_{**}$$

$$\Longrightarrow C_f(x) = \frac{C_\mu^{1/4} k^{1/2} u}{\frac{1}{2}U^2 \frac{1}{\kappa}\ln\left(E\frac{d\rho C_\mu^{1/4}k^{1/2}}{\mu}\right)} \tag{5}$$

The evolution of the distribution of $C_f(x)$ with adaptive cycles is illustrated in Fig. 7(b) where grid convergence is clearly achieved. Thus this prediction is verified in the sense of Roache. The distribution obtained on the final mesh is plotted in Fig. 8. Results are poor when compared to White's correlation, especially near the leading edge of the plate where the two-velocity scale wall functions fail to reproduce an observed peak of $C_f(x)$ and TKE. In fact, the behavior

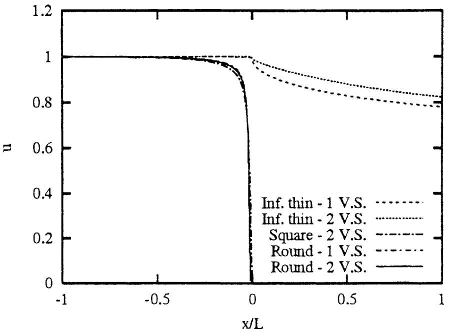


Fig. 9. Horizontal velocity at $y = 0$.

is opposite to that of measurements. The predicted level of k is very low at the inflow boundary of the domain and remains low because the computational velocity gradients near the leading edge of the plate remain small. Hence, the value of u_k is very low at the leading edge of the plate so that equation (5) predicts low values of τ_w and $C_f(x)$. This simulation fails validation in the sense of Roache.

Realistic distributions of $C_f(x)$ can only be achieved if the expected higher levels of TKE can be generated by the computational model. It is the velocity gradients in the production term of the k-equation that are responsible for generating this turbulence and they are most important where the flow meets the plate. This explains the measured large peak in the value of $C_f(x)$ at $x/L = 0$. Such a peak is not observed in the present computations because there is no stagnation point in the computational model of the leading edge of the plate as illustrated in Fig. 9. The two curves of interest are labeled as "Inf. thin" (infinitely thin plate). Therefore, the geometry of the computational model is not appropriate to reproduce the true physics of the flow. Fig. 10(a) illustrates what happens when wall functions are applied to a zero thickness plate. The computational domain is slightly offset from the plate. The fluid flows from a zero traction boundary condition along the symmetry line to a non-zero traction specified by the wall function. In the absence of a stagnation point the flow slows down gradually and does not generate high levels of TKE required to cause the observed peak of C_f near the leading edge. Fig. 10(b) shows an other computational domain that contains a stagnation point that will cause sudden deceleration of the fluid. This causes large velocity gradients which in turn lead to high production of TKE. The leading edge peak of C_f then naturally occurs.

In the next section, the geometry of the model is modified so that the physically observed stagnation point is reproduced in the simulation.

5.3. Effect of the plate geometry

When the thickness of the plate is included in the geometry of the model, the physically observed stagnation point at the leading edge of the plate is reproduced in the simulation (Fig. 9). In this figure the labels "Square" and "Round" correspond to two geometries of the leading edge shown in Fig. 11. With these geometries, the production of k, which depends on the velocity gradients, is triggered in the leading edge region. As will be shown, the behavior of the shear stress is improved. The half-thickness of the plate is set to 1% of its length.

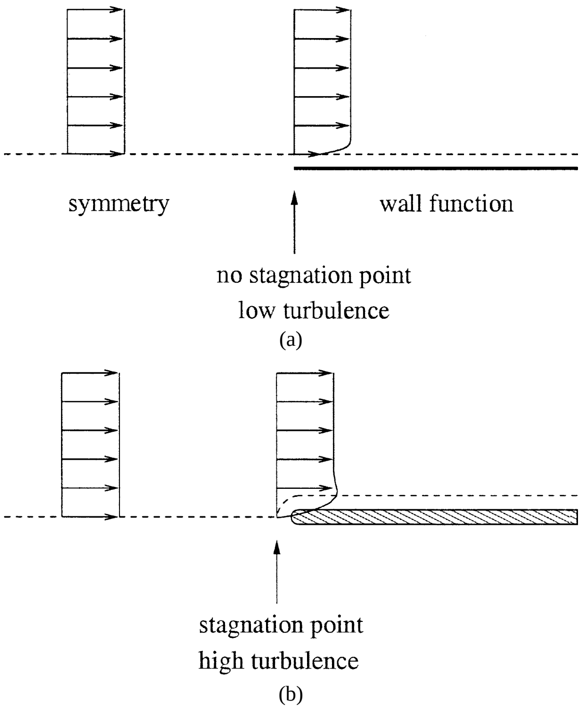


Fig. 10. Profile of u near the leading edge of the plate: (a) Zero thickness plate; (b) Finite thickness plate.

5.3.1. Square edge—1 velocity scale

In this attempt, the plate presents a square leading edge, as shown in Fig. 11(a). A one-velocity scale wall function is used on the horizontal and vertical portions of the plate. Unfortunately, a solution could not be achieved due to convergence difficulties. A distribution of $C_f(x)$ for this particular case is therefore not presented. It is believed that the high non-linearity of the wall functions, combined with the singularity located at the plate leading edge corner, where the wall functions meet, are responsible for the difficulties encountered.

5.3.2. Square edge—2 velocity scales

Wall functions with two velocity scales are used on both portions of the plate surface. This time, it was possible to obtain a mesh independent solution. As expected, a peak of turbulence kinetic energy is generated at the leading edge of the plate, as shown in Fig. 12. Plots of $C_f(x)$ for each cycle of mesh adaptation are presented in Fig. 13(a) along with White's correlation. This figure indicates that several cycles of adaptation are required to achieve a grid converged distribution of skin friction. The square edge produces the highest level of turbulence. However, this does not automatically translate into higher values of skin friction. After the flow reaches the leading edge of the plate, it is

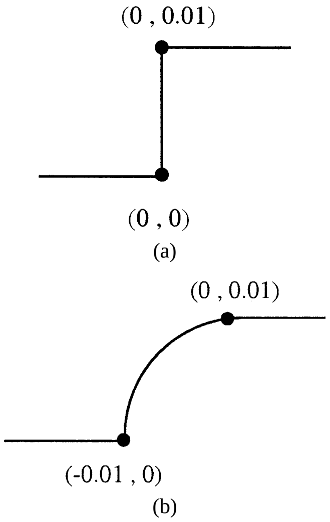


Fig. 11. Enlargement of the plate leading edge: (a) Square edge; (b) Round edge.

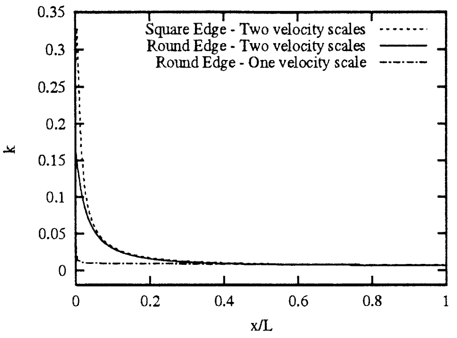


Fig. 12. Variation of TKE along the plate.

deflected upwards, away from the horizontal surface. As a consequence, the horizontal component of the velocity on the plate surface will be low near the leading edge. Fig. 14 shows plots of the slip velocity along the plate. The slip velocity is the tangential component of the velocity at the computational boundary. Low values of the slip velocity yield low wall shear so that the effect of the turbulence peak is diminished. As a result, the distribution of $C_f(x)$ obtained on the final mesh does not compare well with White's correlation (Fig. 13(a)). Hence this simulation fails validation.

By rounding off the plate leading edge, we will see that:

(1) the problem is easier to solve due to the elimination of the singularity, and
(2) the component of the velocity tangent to the plate is increased, resulting in higher values of $C_f(x)$.

5.3.3. Round edge—1 velocity scale

An enlarged view of the round edge (circular arc) is shown in Fig. 11(b). The local skin friction coefficient is calculated on the horizontal surface, between the points $(0, 0.01)$ and $(1, 0.01)$. The discrepancy with White's correlation is large, as seen in Fig. 13(b), because the peak of

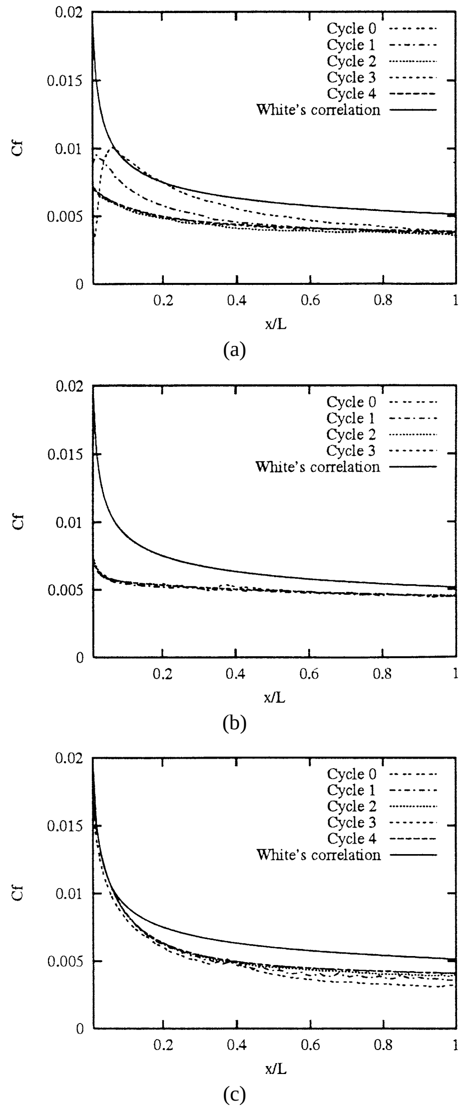


Fig. 13. Distributions of $C_f(x)$ along a plate with a finite thickness for $Re = 2 \times 10^5$: (a) Square leading edge, 2 velocity scales; (b) Round leading edge, 1 velocity scale; (c) Round leading edge, 2 velocity scales.

turbulence kinetic energy is too low (Fig. 12 and Eq. (4)). The one-velocity scale wall function thus appears to be inappropriate for this problem. The model using one-velocity scale wall function fails validation.

5.3.4. Round edge—2 velocity scales

Turbulent fluctuations are at the origin of the velocity scale u_k. The latter, along with the velocity scale u_{**}, determines the boundary condition for the streamwise momentum equation. As seen in Fig. 13(c), the use of two velocity scales yields better results. At last, the peak at the leading edge is reproduced and the values of $C_f(x)$ come within 13% of the values predicted with White's correlation. Elsewhere on the plate, the maximum discrepancy with White's correlation is approximately 25%. This level of discrepancy is unacceptable in practice and may be due to the fact that the Reynolds number of 200 000 is too low for comparing with the cor-

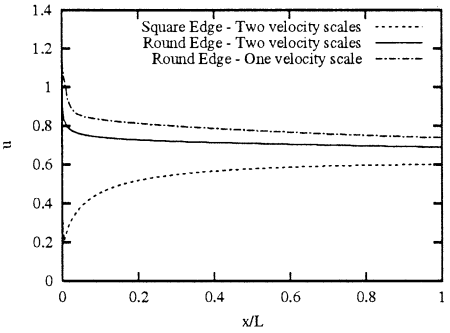


Fig. 14. Slip velocity along the plate.

relations. Indeed, authors such as Schultz-Grunow [42] and Smith and Walker [43] have reported experimental values of $C_f(x)$ over a flat plate in a turbulent incompressible flow for a range of higher Reynolds numbers. For instance, Smith and Walker, who have conducted their experiments on a plate with an elliptic leading edge, present results for the range $1.5 \times 10^6 < Re_x < 5 \times 10^8$. Thus, the present Reynolds number of 2×10^5 is at the low end of the turbulent range, which may partly explain the previously observed discrepancies. Again, the simulation fails validation.

In the next section we increase the Reynolds number to 2×10^6 and achieve better agreement between numerical predictions and correlations.

6. Heated flat plate

Turbulent heat transfer on a heated flat plate with a round leading edge is now solved using the k–ε model and a two-velocity scale wall function. The flow conditions are as follows: $Re = 2 \times 10^6$, $Re_t = 2 \times 10^5$, $T_\infty = 0$, $T_w = 1$.

6.1. Prediction of $C_f(x)$

The evolution of the distributions of $C_f(x)$ with adaptive cycles is shown in Fig. 15(a). Once again, the importance of adaptive remeshing is clearly illustrated. Grid convergence is achieved confirming verification of the simulation in the sense of Roache. The grid independent distribution (Cycle 6) is plotted in Fig. 15(b) along with White's correlation and another obtained from Schultz-Grunow's experimental results [42]. The maximum discrepancy between the numerical results and White's correlation is 13%. At the leading edge, for example, the discrepancy is 11%. The agreement is even better when comparing with Schultz-Grunow's results. A difference of approximately 17% is observed at the leading edge, but on 98% of the plate length ($0.02 < x/L < 1$), the discrepancy never exceeds 10%. This is a noticeable improvement compared to the case $Re = 2 \times 10^5$ of Fig. 13(c). If such levels of agreement are acceptable then the computational model is validated in the sense of Roache. Otherwise, it fails validation.

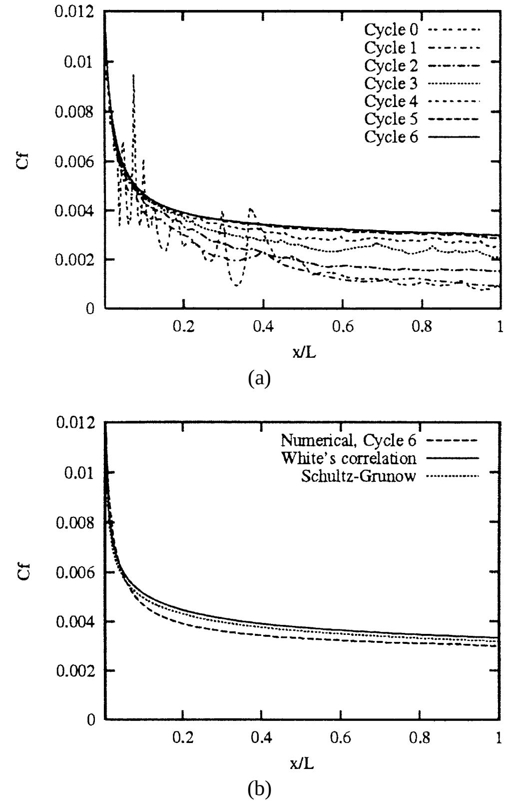


Fig. 15. Distributions of $C_f(x)$ for $Re = 2 \times 10^6$: (a) Evolution of $C_f(x)$ with adaptive cycles; (b) Comparison of $C_f(x)$ with correlations.

6.2. Prediction of $St(x)$

The local heat transfer coefficient is reported in dimensionless form using the Stanton number:

$$St(x) = \frac{Nu_x}{Re_x Pr} = \frac{q_w x}{\lambda(T_w - T_\infty) Re_x Pr}$$

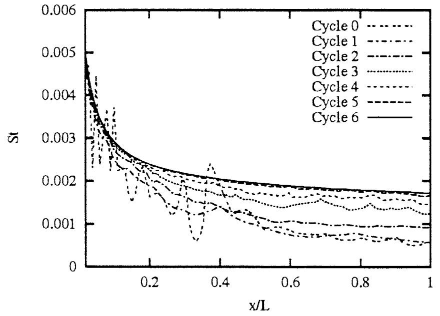
Predictions of the $St(x)$ distribution are computed with the effective value of the heat flux (q_w) used in the temperature wall function:

$$St(x) = \frac{\rho c_p C_\mu^{1/4} k^{1/2}}{T^+} \frac{(T_w - T)}{(T_w - T_\infty)}$$

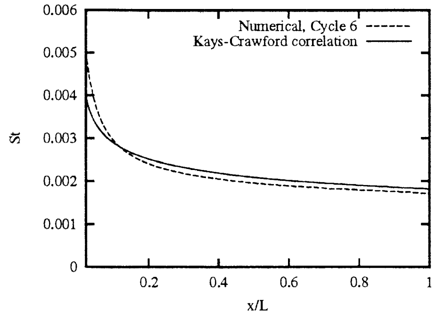
Fig. 16(a) shows the evolution towards a grid independent, thus verified, prediction (Cycle 6). The latter is plotted in Fig. 16(b) and compared to the Kays–Crawford correlation [20]:

$$St(x) = \frac{0.0287}{Pr^{0.4} Re_x^{0.2}} \quad \text{with } Pr = 0.7$$

which is valid within the ranges $0.5 < Pr < 1.0$ and $5 \times 10^5 < Re_x < 5 \times 10^6$. The comparison between the correlation and the grid independent prediction of $St(x)$ is excellent within the range of validity of the correlation ($0.25 < x/L < 1$) where the discrepancy never exceeds 7%. This confirms that $Re = 2 \times 10^6$ is in the range of validity of both the computational model and the correlations.



(a)



(b)

Fig. 16. Distributions of $St(x)$ for $Re = 2 \times 10^6$: (a) Evolution of $St(x)$ with adaptive cycles; (b) Comparison of $St(x)$ with correlation.

Fig. 17 shows three meshes resulting from the adaptive remeshing procedure: the initial coarse mesh (818 nodes), an intermediate mesh (6542 nodes) and the final mesh (37 375 nodes). As can be seen, refinement is required near and upstream of the leading edge, near the wall and also near the edge of the boundary layer in order to capture all features of this flow.

6.3. Evolution of y^+ with the mesh

An other important feature of the adaptive remeshing algorithm is that it allows achieving grid independence of boundary conditions for wall functions. Fig. 18 shows that the value of y^+ depends not only on the distance between the wall and the computational boundary, but also on the mesh. Both the velocity and temperature wall functions depend on y^+ and hence on u_* or u_k, which in turn depend on the flow solution. Thus, achieving grid independent solutions is critical to ensure accurate wall function boundary conditions.

7. Conclusion

This paper has shown that surprisingly good results may be achieved with the k–ε model and wall functions. Success hinges on ensuring that the computational model is specified so as to ensure that key features of the flow are reproduced and that grid refinement studies are performed to verify grid independence of the prediction. The adaptive procedure always leads to verified simulations. However, validation can only be achieved for computational models incorporating all the flow physics. Specification of the computational model includes geometry, boundary conditions, the type of wall functions, and inflow values of k and ε. The latter should be such that the inflow eddy viscosity is small enough to reflect the true nature of the flow. Two-velocity scale wall functions usually produce better results because one of its velocity scales is directly determined by the level of TKE. This has a direct impact on the accuracy of the skin friction and Stanton number predictions. For the flow over a flat plate, TKE production is triggered by strong velocity gradients that can only be generated by including the thickness of the plate in the computational model. Results show that predictions of $C_f(x)$ and $St(x)$ are critically dependent on the use of sufficiently fine meshes.

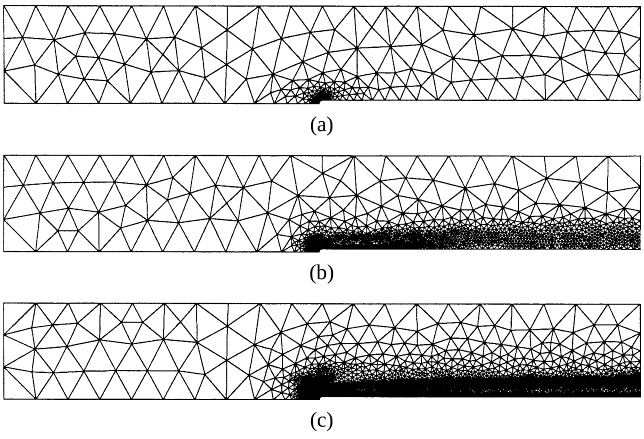


Fig. 17. Meshes obtained with the adaptive remeshing procedure: (a) Initial mesh (Cycle 0: 818 nodes); (b) Intermediate mesh (Cycle 3: 6 542 nodes); (c) Final mesh (Cycle 6: 37 375 nodes).

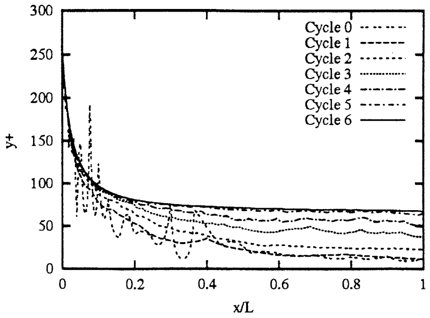


Fig. 18. Evolution of y^+ with the mesh.

The adaptive methodology is a powerful tool to perform the systematic grid refinement studies required for rigorous verification of codes and simulations. Verification and validation become fairly simple and straightforward processes. Asymptotic exactness of the estimator provides for quantitative estimates of the error that are reliable enough ot assess the accuracy of solutions. It becomes possible to ensure that numerical errors are small enough that verification can be performed with confidence. Examples presented show that verification is no substitute for validation; adaptivity simply makes the whole process simpler.

Acknowledgements

This work was supported in part by the Canada Research Chair program (government of Canada), the Natural Science and Engineering Research Counsil (NSERC, government of Canada), le Fonds Québécois de la Recherche sur la Nature et les Technologies (FQRNT, government of Québec) and AFOSR grant F49620-96-1-0329.

References

[1] M. Nallasamy, Turbulence models and their applications to the predictions of internal flows, Comput. Fluids 15 (1987) 151–194.

[2] B.E. Launder, Current capabilities for modelling turbulence in industrial flows, Appl. Sci. Res. 48 (1991) 247–269.

[3] A. Hutton, R. Szczepura, Turbulent flow and heat transfer in a sudden pipe expansion: A comparison of current models of turbulence, CEGB Report TPRD/B/0926/R87 (1987).

[4] P.J. Roache, Verification and Validation in Computational Science and Engineering, Hermosa Publishers, 1998.

[5] J.-F. Hétu, D.H. Pelletier, Adaptive remeshing for viscous incompressible flows, AIAA J. 30 (8) (1992) 1986–1992.

[6] J.-F. Hétu, D.H. Pelletier, Fast, adaptive finite element scheme for viscous incompressible flows, AIAA J. 30 (11) (1992) 2677–2682.

[7] F. Ilinca, D. Pelletier, A unified approach for adaptive solutions of compressible and incompressible flows, in: AIAA 35th Aerospace Sciences Meeting and Exhibit, January 6–10, 1997 / Reno, NV, AIAA Paper 97-0330.

[8] F. Ilinca, D. Pelletier, Posivity preservation and adaptive solution for the k–epsilon model of turbulence, in: AIAA 35th Aerospace Sciences Meeting and Exhibit, January 6–10, 1997 / Reno, NV, AIAA Paper 97-0205.

[9] F. Ilinca, D. Pelletier, A. Garon, An adpative finite element method for a two-equation turbulence model in wall-bounded flows, Internat. J. Numer. Methods Fluids 24 (1997) 101–120.

[10] F. Ilinca, D. Pelletier, L. Ignat, Adaptive finite element solution of compressible turbulent flows, AIAA J. 36 (12) (1998) 2187–2194.

[11] D. Pelletier, J.-F. Hétu, F. Ilinca, Adaptive finite element method for thermal flows, AIAA J. 32 (4) (1994) 741–747.

[12] D. Pelletier, F. Ilinca, J.-F. Hétu, Adaptive finite element method for convective heat transfer with variable fluid properties, AIAA J. Thermophys. Heat Transfer 8 (4) (1994) 687–694.

[13] D. Pelletier, L. Ignat, F. Ilinca, Adaptive finite element method for conjugate heat transfer, Numer. Heat Transfer A 32 (3) (1997) 267–287.

[14] É. Turgeon, D. Pelletier, F. Ilinca, Compressible heat transfer computations by and adaptive finite element method, in: 37th AIAA Aerospace Sciences Meeting and Exhibit, Reno, NV, 1999, AIAA Paper 99-0875.

[15] L. Ignat, D. Pelletier, F. Ilinca, An adaptive finite element method for turbulent heat transfer, in: AIAA 34th Aerospace Sciences Meeting and Exhibit, January 15–18, 1996 / Reno, NV, AIAA Paper 96-0607.

[16] É. Turgeon, D. Pelletier, L. Ignat, Effects of adaptivity on various finite element schemes for turbulent heat transfer and flow predictions, Numer. Heat Transfer A 38 (2000) 847–868.

[17] É. Turgeon, D. Pelletier, Computation of jet impingement heat transfer by an adaptive finite element algorithm, in: 7th AIAA/ASME Joint Thermophysics and Heat Transfer Conference, Albuquerque, NM, 1998, AIAA Paper 98-2585.

[18] D. Lacasse, É. Turgeon, D. Pelletier, Prediction of turbulent separated flow in a turnaround duct using wall functions and adaptivity, Internat. J. CFD 15 (2001) 209–225.

[19] F.M. White, Viscous Flow, McGraw-Hill, New York, 1974.

[20] W. Kays, M. Crawford, Convective Heat and Mass Transfer, second ed, McGraw-Hill Book Company, 1980.

[21] AIAA, Guide for the verification and validation of computational fluid dynamics simulations, Technical Report AIAA-G-077-1998, America Institute of Aeronautics and Astronautics (June 1998).

[22] B.W. Boehm, Software Engineering Economics, Prentice-Hall, Englewood Cliffs, NJ, 1981.

[23] F.G. Blottner, Accurate Navier–Stokes results for the hypersonic flow over a spherical nosetip, AIAA J. Spacecroft Rockets 27 (2) (1990) 113–122.

[24] F. Jay, IEEE Standard Dictionary of Electrical and Electronic Terms, ANSI/IEEE Std 100, Institute of Electrical and Electronic Engineers, 1984.

[25] W.L. Oberkampf, A proposed framework for computational fluid dynamics code calibration/validation, in: 18th AIAA Aerospace Ground Testing Conference, Colorado Springs, Colorado, 1994, AIAA Paper 94-2540.

[26] H.W. Coleman, Uncertainty considerations in validating cfd codes with experimental data transfer, in: AIAA 27th Fluid DYnamics Conference, New Orleans, Louisiana, 1996, AIAA Paper 96-2027.

[27] D.P. Aeschliman, W.L. Oberkampf, F.G. Blottner, A proposed methodology for computational fluid dynamics code verification, calibration and validation, in: Proceedings 16th International Congress on Instrumentation in Aerospace Simulation Facilities, July, 1995, pp. 27.1–27.13, 18–21, Ohio, 95-CH3482-7.

[28] D. Wilcox, Personnal communication, 1994.

[29] B.E. Launder, D.B. Spalding, The numerical computation of turbulent flows, Comput. Methods Appl. Mech. Engrg. 3 (1974) 269–289.

[30] B.E. Launder, Current capabilities for modeling turbulence in industrial flows, Appl. Sci. Res. 48 (1991) 247–269.

[31] V. Arpaci, P. Larsen, Convection Heat Transfer, Prentice-Hall, Englewood Cliffs, NJ, 1984.

[32] L. Ignat, D. Pelletier, F. Ilinca, Adaptive computations of turbulent forced convection, Numer. Heat Transfer A 34 (1998) 847–871.

[33] J.P. Chabard, Projet n3s de mécanique des fluides—manuel théorique de la version 3, Technical Report EDF HE-41/91.30B, Électricité de France (1991).

[34] J.A. Schetz, Boundary Layer Analysis, Prentice-Hall, Englewood Cliffs, NJ, 1993.

[35] F. Ilinca, D. Pelletier, Positivity preservation and adaptive solution for the k–ε model of turbulence, AIAA J. 36 (1) (1998) 44–51.

[36] T.J.R. Hughes, A.N. Brooks, A multidimensional upwind scheme with no crosswind diffusion, in: T.J.R. Hughes (Ed.), Finite Element Methods for Convection Dominated Flows, vol. 34, ASME, 1979, pp. 19–35.

[37] A.N. Brooks, T.J.R. Hughes, Streamline upwind / Petrov–Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier–Stokes equations, Comput. Methods Appl. Mech. Engrg. 32 (1982) 199–259.

[38] T.J.R. Hughes, L.P. Franca, G.M. Hulbert, A new finite element formulation for computational fluid dynamics: viii. the Galerkin/least-squares method for advective–diffusive equations, Comput. Methods Appl. Mech. Engrg. 73 (1989) 173–189.

[39] M. Crouzeix, P. Raviart, Conforming and non-conforming finite element methods for solving the stationary stokes equations, RAIRO 3 (1973) 77–104.

[40] D. Pelletier, Adaptive finite element computations of complex flows, Internat. J. Numer. Methods Fluids 31 (1999) 189–202.

[41] É. Turgeon, D. Pelletier, J. Borggaard, A general continuous sensitivity equation formulation for the k–ε model of turbulence, in: 31st AIAA Fluid Dynamics Conference and Exhibit, Anaheim, CA, 2001, AIAA Paper 2001–3000.

[42] F. Schultz-Grunow, New frictional resistance law for smooth plates, NACA TM 986 (1941).

[43] D.W. Smith, J.H. Walker, Skin-friction measurements in incompressible flow, Technical Report R-26, NASA (1959).