A high-resolution code for turbulent boundary layers

Mark P. Simens [a,*], Javier Jiménez [a,b], Sergio Hoyas [a,c], Yoshinori Mizuno [a]

[a] *School of Aeronautics, Universidad Politécnica de Madrid, 28040 Madrid, Spain*
[b] *Center for Turbulence Research, Stanford University, Stanford, CA 94305, USA*
[c] *Univ. Politécnica de Valencia, CMT Motores Térmicos, E-46022 Valencia, Spain*

ARTICLE INFO

ABSTRACT

A new high-resolution code for the direct simulation of incompressible boundary layers over a flat plate is described. It can accommodate a wide range of pressure gradients, and general time-dependent boundary conditions such as incoming wakes or wall forcing. The consistency orders of the advective and pressure-correction steps are different, but it is shown that the overall resolution is controlled by the higher-order advection step. The formulation of boundary conditions to ensure global mass conservation in the presence of arbitrary forcing is carefully analyzed. Two validation boundary layers with and without a strong adverse pressure gradient are presented, with maximum Reynolds numbers $Re_\theta \approx 2000$. They agree well with the available experiments. Turbulent inflow conditions for the zero-pressure case are implemented by a recycling method, and it is shown that at least the initial 300 momentum thicknesses have to be discarded before the effect of the artificial inflow is forgotten. It is argued that this is not a defect of the method used to generate the inflow, but a property of the boundary layer.

1. Introduction

This paper describes the design and performance of a new direct simulation code to compute turbulent and transitional boundary layers. It is primarily intended to explore flow physics in relatively large simulations, and part of the design requirements were that it should be useful in control applications including flows both with arbitrary free-stream perturbations, and with time-dependent injection and suction at the wall. In contrast, the flow geometry is kept simple; a parallel-epiped over a flat wall, and an upper boundary that is assumed to be far enough into the free stream to model flow at infinity. Arbitrary pressure gradients are created by manipulating the transpiration velocities over that boundary. Both laminar and turbulent inflow conditions are of interest, and the problems associated with the latter are discussed in some detail.

Although the fractional-step finite differences numerical scheme [1,2] with inflow and outflow boundary conditions is relatively classical, the requirements just mentioned necessitate implementation adaptations which are not. For example, efficiency suggests that the order of the advective and of the pressure-correction substeps should be chosen different, and the requirements of arbitrary mass injection forces a careful rethink of how to implement global mass conservation. Those adaptations, and their performance in test and application problems, are the main subject of the paper.

Turbulent boundary layers have been subjects of interest from the first days of fluid mechanics, both the canonical case with zero pressure gradient (ZPG), and the more applied ones subject to adverse pressure gradients (APG). As a consequence, they were some of the first flows to be simulated [3,4], but the Reynolds numbers of those simulations have increased more slowly than for streamwise-homogeneous flows, such as pipes or channels [5,6]. Part of the reason is that they are harder to

---

* Corresponding author.
  E-mail addresses: mark@torroja.dmt.upm.es (M.P. Simens), jimenez@torroja.dmt.upm.es (J. Jiménez).

compute because they are inhomogeneous in at least two directions, but perhaps equally important is that they require inflow boundary conditions. In transitional cases this is relatively straightforward, since the inflow is laminar, but even there the question of how to seed the perturbations has to be considered. Transition in boundary layers is typically subcritical, and its details depend on the level and character of the upstream perturbations. That is why experimental layers are usually tripped, and why some equivalent mechanism is needed in simulations. If the problem of transition is to be completely avoided, some way has to be devised to generate turbulent inflow conditions. This nontrivial problem has been the subject of a lot of work, and influences the choices made for the rest of the code.

Although the boundary layer is a spatially developing flow, attempts have often been made to reintroduce periodicity in the numerical problem, in some cases to retain spectral methods, and in others to use the outflow as a convenient source of realistic fluctuations to synthesize a turbulent inflow. The strategies that use spectral methods further subdivide into those in which the flow is expanded in terms of a slow growth plus periodic small-scale fluctuations [3,7], and those in which the outflow is damped to minimize the spectral errors due to edge discontinuities, and reintroduced at the inflow after some manipulation [4]. Although the latter 'fringe' methods are accurate [8], and have been used to simulate boundary layers at relatively high-Reynolds numbers [9–11], one of the effects of the damping is that the incoming flow is typically almost laminar, and that increasing the Reynolds numbers becomes increasingly expensive. A further problem is that periodicity is difficult to reconcile with strong spatial inhomogeneity, such as strong pressure gradients, or with control strategies involving net mass injection.

Once spectral accuracy is abandoned, the choice of inflow strategies becomes wider, and reduces to determining how a synthetic turbulent inflow can be made to converge fastest to a more realistic structure. A recent survey is [12], which, although primarily concerned with large-eddy simulations and therefore with low-order statistics, shows that it is difficult to create inflows with accommodation lengths shorter than several boundary layer thicknesses. In this paper we use the version of the fringe method introduced in [13], in which an intermediate flow plane is rescaled and copied to the inflow without damping. Our primary interest is to determine the length that has to be discarded before the flow can be assumed to have forgotten the artificial inflow, and our assumption will be that the goal is to clarify the flow physics, about which nothing is known a-priori.

If the simulation is non-periodic, spectral methods lose much of their attraction, and primitive-variable finite difference fractional-step methods suggest themselves [14–16]. The fractional-step method was introduced in [17], and was first used for turbulent flow simulations in [1]. Although, it is still an area of investigation and of some controversy [18], most of the important problems surrounding it [1], can now been considered solved [19,2,20,21], and obtaining at least second-order accuracy for the velocity components is straightforward.

Second-order central finite difference schemes have been applied [14–16]. They conserve momentum, energy and mass, which makes them highly suitable for large-eddy simulations, but their relative poor resolution properties makes them less attractive for high-Reynolds numbers direct simulations. Rather than obtaining very high accuracy, the problem is how to improve the resolution to reduce the size of the necessary grids, and optimize memory use. This usually also reduces computational time.

An obvious choice to improve resolution is to use compact finite difference schemes, especially fourth-order ones that retain narrow stencils in collocated [22] and staggered grids [23]. Those schemes do not conserve energy in the inviscid limit, but this is of minor importance in direct simulations in which the resolution should in any case be chosen so that viscosity is the primary dissipation mechanism. Examples of the use of compact finite differences in DNS are [24,25], who do not use a fractional-step formulation. When using fractional steps, it turns out that the pressure projection scheme cannot be efficiently implemented with compact differences, and that the best choice is to mix a lower-order formulation for the pressure with a higher-order one for the velocities. Whether this maintains the required resolution will be one of the subjects of our analysis.

The structure of this paper is as follows. The basic code is described in Section 2, followed in Section 2.1 by the analysis of the effect of the low-order pressure scheme. The question of global mass conservation is addressed in Section 3 for the general case, and for our particular formulation. Two validation boundary layers with ZPG and APG, at relatively high-Reynolds numbers, are described in Section 4. The ZPG case has a turbulent inflow, and especial attention is given to the effect of the inflow procedure. A preliminary report of some of the material presented here, as well as further implementation details and examples of code use, can be found in [26].

2. Problem formulation

The problem to be solved in a parallelepiped is the primitive-variable formulation of the incompressible Navier–Stokes equations,

$$\nabla \cdot \vec{u} = 0, \tag{1}$$

$$\partial_t \vec{u} + \nabla \cdot \vec{u}\vec{u} = -\nabla p + \frac{1}{Re}\nabla^2 \vec{u}. \tag{2}$$

The velocity components of $\vec{u}$ in the streamwise, x, wall-normal, y, and spanwise, z, directions are u, v and w. The pressure p includes the constant fluid density. The boundary conditions at the inlet and at the top and bottom x–z planes depend on the

flow being simulated, but the domain is always periodic in the z direction. The velocities at the outflow are estimated by a convective boundary condition,

$$\partial_t \vec{u} + U_c \partial_x \vec{u} = 0, \tag{3}$$

where U_c is usually the free-stream velocity at the exit, with a small correction to enforce global mass conservation that is discussed in Section 3.

The fractional-step method described in [2,19] is used to enforce continuity and to efficiently solve for the pressure. The basic idea is to write Eqs. (1) and (2) in semi-discrete form, and to factorize the resulting matrix system using an LU-decomposition [2]. For a three-step Runge–Kutta time integration this results in

$$(\Delta t^{-1} I - \beta_l L)\vec{u}_*^l = \Delta t^{-1} \vec{u}^{l-1} - \delta_l G p^{l-1} - \gamma_l N \vec{u}^{l-1} - \zeta_l N \vec{u}^{l-2} + \alpha_l L \vec{u}^{l-1} + b.c.'s, \tag{4}$$

$$\Delta t \delta_l DG \Delta p^l = D \vec{u}_*^l + b.c.'s, \tag{5}$$

$$\vec{u}^l = \vec{u}_*^l - \Delta t \delta_l G \Delta p^l, \tag{6}$$

$$p^l = p^{l-1} + \Delta p^l, \tag{7}$$

where $l = 1 \cdots 3$ is the sub-step number, $l - 2$ is ignored for $l = 1$ and $\vec{u}^0$ and $\vec{u}^3$ are the velocities at time steps n and $n + 1$. The coefficients for the Runge–Kutta are those in [21,27]. The nonlinear terms, and in general all the terms that are treated explicitly in time, are included in the operator N. The linear implicit operator L includes only the viscous terms in the y-direction. It was shown in [27] that this treatment of the viscous terms does not lead to excessively small time steps in pipes, and the same has been found by us in boundary layers. The divergence and the gradient are D and G. It was shown in [21,27] that this Runge–Kutta is second-order accurate for the velocity, with the third-order errors being mostly viscous terms. This was checked independently for our code in the model problems described below. The main advantage of the scheme (4)–(6) is to allow the longer time steps associated with its good representation of the nonlinear advective terms.

The original paper [19] used the pressure-increment decomposition applied here, but they did not analyze the accuracy of Eq. (7). It can be shown that the error of the pressure updated in this way is $O(\Delta t)$, but does not influence the velocities. Other choices of the coefficients of Δp in Eq. (7) results in pressure errors that vary in magnitude, but which do not change the order. The coefficients used here are those for which the pressure error does not propagate from one time step to the next. In any case, accurate pressures for the purpose of compiling statistics can be recovered from the velocity fields by solving the full Poisson's equation.

The convective and viscous terms in the x and y-directions are calculated using staggered three-point compact finite differences. The coefficients for uniform grids are given in [23]. Those for nonuniform grids are obtained numerically using Taylor expansions to enforce the highest possible consistency order. Fig. 1 defines how the variables are arranged in the x–y planes. The velocity components and the pressure are expanded in Fourier series along z. No staggering is used in that direction, and the computation of the nonlinear terms is pseudo-spectral, using the 2/3 rule to prevent aliasing [28].

The time-step Δt is determined, using a constant CFL, from

$$\Delta t = \text{CFL} \min \left(\frac{Re(\Delta x)^2}{6}, \frac{Re(\Delta z)^2}{\pi^2}, \frac{\Delta x}{\sqrt{3}|u|}, \frac{\Delta y}{\sqrt{3}|v|}, \frac{\Delta z}{\pi|w|} \right). \tag{8}$$

Although the time stepper is stable up to CFL ≈ 1.7, most of the simulations described below are run with CFL $\approx 0.5 - 1$, to preserve time accuracy.

2.1. Lower-order schemes for the Poisson matrix

The Poisson operator in (5) has the form DG. We considered various discretizations for the divergence and gradient operators in Eqs. (4)–(7). The obvious one is to use the same compact difference schemes as for the convective and viscous terms.
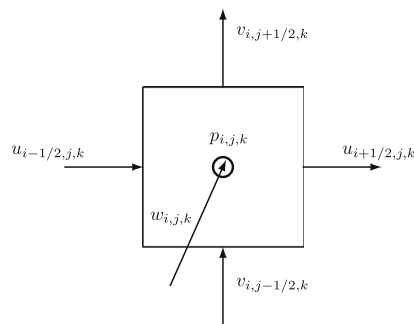

Fig. 1. The staggered grid configuration used in the code. Only the u, v velocities are staggered in the x–y plane. The velocities and the pressure are not staggered in z, so that w and p are collocated. Ghost points are used to impose boundary conditions.

The divergence and the gradient would then be calculated from $A_D(D\vec{u}) = B_D\vec{u}$ and $A_G(Gp) = B_G p$, where A_G, A_D, B_D, B_G contain the finite difference operators, and the discretized Poisson equation would be

$$A_D^{-1} B_D A_G^{-1} B_G p = A_D^{-1} B_D \vec{u} \rightarrow B_D A_G^{-1} B_G p = B_D \vec{u}. \tag{9}$$

Unfortunately, the matrix $B_D A_G^{-1} B_G$ is full, and discretizing DG in this manner is impractical.

From the various ways of achieving a sparser A_G^{-1}, it was eventually decided to use a standard finite difference centred second-order scheme [1]. Although this makes the code formally second-order accurate, it will be shown below that the effect on the resolution is not severe. Reducing the formal order of the code has the added advantage of allowing the use of a fast Fourier scheme to solve the Poisson problem. The two sides of Eq. (5) are expanded in cosines along x, and the problem reduces to a series of one-dimensional Helmholtz problems along y, with a modified wavenumber for each cosine representing the finite differences in x [29]. The Gibb's errors due to the implicit derivative discontinuities at the non-periodic end points are of order Δx^2, except at the end points, and do not degrade the consistency. Higher-order schemes in non-periodic domains cannot be treated in this way, because they require boundary schemes with different modified wavenumbers than the interior operator.

The use of cosines also requires uniform grids along x, which are used in all the tests described below. A two-dimensional multigrid solver was implemented for use with nonuniform grids in both x and y, but it turned out to be several times slower than the direct method, and it was always found that the extra cost overwhelmed any savings obtained from nonuniform x grids.

The influence of these low-order schemes on the resolution properties of the overall solution can be analyzed for triply-periodic boundary conditions, assuming in addition that $D = G^T$ [30]. The velocities can then be expressed as Fourier expansions,

$$\vec{u}(\vec{x}) = \sum \hat{u}(\vec{k}) \exp(i\vec{k} \cdot \vec{x}), \quad \text{where } \vec{k} = (k_x, k_y, k_z), \tag{10}$$

and both the divergence and the gradient reduce to $\widehat{D\vec{u}} = i\vec{k} \cdot \hat{u}$, and $\widehat{Gp} = i\vec{k}p$, where $\vec{k}$ is either the exact or the modified wavevector.

The pressure correction is a projection on the solenoidal plane $\vec{k} \cdot \hat{u} = 0$, given by

$$\hat{u} = \hat{u}_* - \vec{a}(\vec{a} \cdot \hat{u}_*), \tag{11}$$

where $\vec{a} = \vec{k}/|k|$ is the unit vector in the direction of $\vec{k}$, where $|k|$ is the Cartesian norm. Assume that $\vec{a}$ and $\vec{b}$ are unit vectors along the true and modified numerical wavevectors, respectively. The velocity error due the pressure step is

$$\hat{u}_\epsilon = \vec{b}(\vec{b} \cdot \hat{u}_*) - \vec{a}(\vec{a} \cdot \hat{u}_*), \tag{12}$$

which can be written in matrix form as

$$\hat{u}_\epsilon = E\hat{u}_*, \quad \text{where } E_{ij} = b_i b_j - a_i a_j, \tag{13}$$

and i and j vary over (x, y, z). Since E is symmetric, the magnitude of the error is bounded by its spectral radius,

$$|\hat{u}_\epsilon| \leqslant \|E\||\hat{u}_*|, \tag{14}$$

which is easily found to be $\|E\| = |\sin\theta_{ab}|$, where θ_{ab} is the angle formed by the true and modified wavevectors. Note in particular that the error vanishes whenever both wavevectors are parallel, even if they have different magnitudes. For example, that is the case for all the Fourier modes aligned with the coordinate axes, or for any of the diagonals in Fourier space. It is because of all these null directions that the error of the projection step tends to be moderate. Moreover, this error does not accumulate over time, but is a bound for the error in the velocity field due to the approximate representation of continuity.

Two examples are given in Fig. 2(a) for the two-dimensional case in (k_x, k_y). Because of the symmetries in the problem, the error norm $\|E\|$ can be computed for a single octant. The lower octant in the figure displays error isolines for a pressure step based on centred second-order finite differences, given along each direction by

$$u_i' = \frac{u_{i+1/2} - u_{i-1/2}}{\Delta x}, \tag{15}$$

whose modified wavenumber is

$$k_{mod2}^* = 2\sin(k^*/2), \tag{16}$$

where $k^* = k\Delta x$. The upper octant displays error isolines for a projection step based on the fourth-order compact derivative

$$u_{i-1}' + 22u_i' + u_{i+1}' = \frac{24(u_{i+i/2} - u_{i-1/2})}{\Delta x}, \quad k_{mod4}^* = \frac{24\sin(k^*/2)}{11 + \cos(k^*)}. \tag{17}$$

Although the fourth-order errors are smaller than those of (16), specially in the consistency limit, $|k^*/\pi| \ll 1$, both errors remain small throughout the range of numerical wavenumbers. The largest error norm is $\|E\| \approx 0.15$ for Eq. (16), and $\|E\| \approx 0.11$ for Eq. (17).
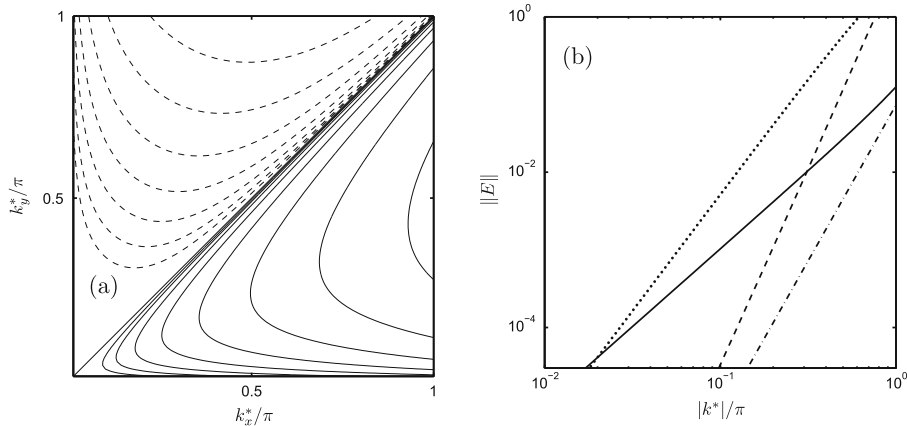
Fig. 2. (a) Resolution characteristics of the Poisson projection step based on the second-order derivative in Eq. (16) (lower octant), and on the fourth-order derivatives compact finite differences in Eq. (17) (upper octant). The isolines are $\|E\| = 0.001\,(\times 2)\,0.128$. (b) Maximum error bound for the same two Poissons, and for the fourth-order compact finite difference advective derivative, as a function of the magnitude of the wavenumber. —, second-order Poisson in Eq. (16); - - - -, fourth-order advection in Eq. (20); – ·–, fourth-order Poisson in Eq. (17); ······, advective error of centred second-order finite differences.

These small error magnitudes suggest that a lower-order Poisson discretization is acceptable even when combined with higher-order advective derivatives. Consider the schemes used for the latter in the present code, which uses compact fourth-order formulas both for differentiation

$$\frac{1}{22}\left(f'_{i+3/2} + f'_{i-1/2}\right) + f'_{i+1/2} = \frac{12}{11\Delta x}\left(f^I_{i+1} - f^I_i\right), \tag{18}$$

and interpolation [23],

$$\frac{1}{6}\left(f^I_{i+1} + f^I_{i-1}\right) + f^I_i = \frac{2}{3}\left(f_{i+1/2} + f_{i-1/2}\right). \tag{19}$$

The resulting combined dispersion relation is

$$k^*_{adv} = \frac{96\sin(k^*)}{67 + 28\cos(k^*) + \cos(2k^*)} = k^* + O(k^{*5}). \tag{20}$$

Restricting ourselves to the one-dimensional case of an operator of the form $U\partial_x$, the error norm after an integration time T becomes $\|E_T\| = (UkT)|k_{adv}/k - 1|$, where the coefficient is the number of eddy turnovers. Fig. 2(b) compares the error magnitudes for the two Poisson steps based on the derivatives in Eqs. (16) and (17), with the advective error during one turnover due to Eq. (20). Even if the second-order error of the Poisson step dominates for the longest wavelengths, the resolution limit, $|k^*/\pi| = O(1)$, is controlled by the fourth-order advection, whose error grows faster with the wavenumber. The fourth-order Poisson approximation is more accurate, but its extra accuracy does not improve the error of the overall simulation, which is dominated by the advection at all wavenumbers. We have included in the figure the advective errors that would result from using standard second-order centred finite differences for the advective term. As expected, they are high everywhere, and degrade both the resolution and the consistency limits. For example, the resolution limit for the present scheme, which combines fourth-order advection and second-order Poisson projection, is $|k^*/\pi| \approx 0.3$ for $\|E\| = 10^{-2}$. It would fall to $|k^*/\pi| \approx 0.07$ with second-order advection, more than quadrupling the required number of points. Note that this analysis is conservative towards the side of the advection, because we have neglected the number of turnovers, which is typically large. An example is given next in which the error is dominated by the fourth-order advection scheme over most of the resolution range.

2.2. Taylor–Green vortex

To test if the resolution predicted in Fig. 2(b) is obtained by our code, the decay of two-dimensional Taylor–Green viscous vortices is simulated and compared with the analytical solution

$$\begin{aligned} u(x,y,t) &= -\cos(kx)\sin(ky)\exp(-2k^2t/Re), \\ v(x,y,t) &= \sin(kx)\cos(ky)\exp(-2k^2t/Re), \\ p(x,y,t) &= -\frac{1}{4k}(\cos(2kx) + \cos(2ky))\exp(-4k^2t/Re), \end{aligned} \tag{21}$$

for wavenumbers in the range $k = 3 - 19$. The Reynolds number is $Re = 1000$, and all simulations are run for $T = 10\pi/k^2$, corresponding to about $10\pi/k$ turnovers. The decay of the velocities during this time is about 0.94, independently of the
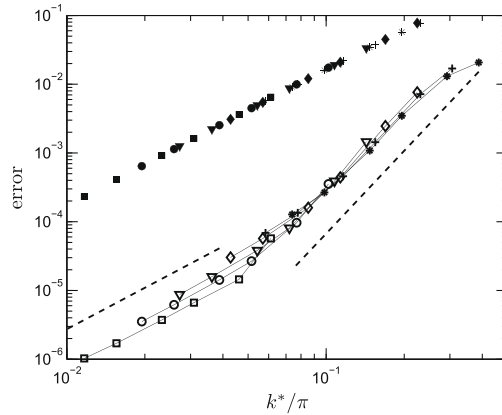
Fig. 3. Solid symbols are the relative L_2 errors for the pressure, $\|p_{num} - p\|/\|p_{num}\|$, and the open ones are the errors per turnover for the velocity $\|u_{num} - u\|/\|u_{num}\|kT$. $\square$: $k = 3$; $\circ$: $k = 5$; $\triangledown$: $k = 7$; $\diamond$: $k = 11$; $+$: $k = 15$; $*$: $k = 19$. The two dashed lines are second- and fourth-order convergence respectively.

wavenumber. Tests are run with a small CFL $= 0.1$, to isolate the errors due to the spatial discretization on a square domain of size $L_x \times L_y = \pi \times \pi$. The grid has N uniformly distributed points along x and y. Initial and Dirichlet boundary conditions are imposed on u and v, but no conditions are required for the pressure.

The relative L_2 errors are computed at the end of each run, and are given in Fig. 3. Following the argument in Section 2.1, the velocity errors are scaled with the number kT of turnovers, while those of the pressure are not. Both errors collapse well in this normalization, and the velocity errors display the predicted change from second- to fourth-order accuracy around $k = 0.05\pi$. Even if the advective terms cancel identically in the analytic solution to this particular problem, the numerical error for the velocities is dominated by them over most of the resolution range.

3. Pressure, mass conservation and boundary conditions

As explained in [2,19], the solution of Poisson's equation (5) for the pressure does not require boundary conditions if velocity conditions have been previously imposed. Conceptually, this is equivalent to using homogeneous Neumann conditions for the pressure, since otherwise the correction step in Eq. (6) would change the imposed normal velocities at the boundary. The problems created by the tangential pressure gradients, which do change the tangential boundary velocities, were discussed in [1], and are minimized here by using pressure increments.

However, there are occasions in which the velocity boundary conditions are poorly known, and they typically require adjustment before the equations can be solved [9,15]. The most common case is the outflow boundary, where velocities are typically 'guessed' from a heuristic advection formula such as Eq. (3), and which can generate substantial pressure pulses whenever large eddies exit the domain.

The root of the problem is that Poisson's equation with Neumann boundary conditions is singular, being invariant to an additive constant pressure. This also implies that its right-hand side has to satisfy a compatibility condition which is equivalent to global mass conservation.

The Poisson problem is

$$\nabla^2(\Delta p) = \nabla \cdot \vec{u}_* \tag{22}$$

in the continuous case, and Gauss' integration over the domain Ω leads to

$$\int_\Omega \nabla^2(\Delta p)\,d\Omega = \int_{\partial\Omega} \partial_n(\Delta p)\,d\partial\Omega = 0 = \int_\Omega \nabla \cdot \vec{u}_*\,d\Omega = \int_{\partial\Omega} \vec{u}_n\,d\partial\Omega. \tag{23}$$

The last term in this chain of equations is a statement on the boundary velocities, but the next one to its left is a condition on the divergence itself, which is supposed to have been derived from the nonlinear advection step. In fact, adjusting the boundary velocities introduces a discontinuity in $\vec{u}_*$, and a delta function in the divergence that enforces the condition that it should have zero mean.

The discrete equivalent of Eq. (22) has the form,

$$DG\Delta p^{n+1} = D\vec{u}_*, \tag{24}$$

whose compatibility condition is

$$\psi^T DG\Delta p^{n+1} = \psi^T D\vec{u}_* = 0, \tag{25}$$

where the T superscript denotes transposition, and ψ is the left null eigenvector of DG. The two equations in (25) correspond to the first and last term in (23), and multiplication by ψ^T is the numerical equivalent of integration over $d\Omega$.

Assume for example that we enforce mass conservation by adding to the streamwise velocity u_* a constant correction u_p, which is zero everywhere except at the outflow boundary. Define $\mathbf{1}_e$ as the vector whose components are one over the points in which the correction is nonzero, and zero elsewhere, so that the new velocity is $u_* + u_p\mathbf{1}_e$. Introducing this velocity in Eq. (25), we obtain

$$u_p = -\frac{\psi^T D\vec{u}_*}{\psi^T D\mathbf{1}_e}, \tag{26}$$

which provides the magnitude of the required correction. Note that the denominator in this equation is a number that only needs to be computed once for each simulation. Non-constant corrections, or corrections over other boundaries, can be trivially accommodated.

In general, Eq. (26) can only be applied after the advection step in Eq. (4), because the denominator is not known beforehand, but the problem simplifies for the second-order pressure discretization used here. In most cases, the null vector ψ is nonzero only near the boundary, but, in the particular case of the staggered second-order scheme, it can be shown that the inner product in the numerator of Eq. (26) is the numerical flux,

$$\sum_i u_{*n,i}\varDelta_i, \tag{27}$$

where u_{*n} is the outwards normal velocity component, $\varDelta$ is the width of the grid cell centred on the normal velocities, and the summation extends only over known boundary values. For example, the contribution to Eq. (27) from the outflow boundary is $u_{N_x,j}h_j$, where $h_j = y_{j+1} - y_j$ and the y_j are located at the positions of v_j in Fig. 1. The correction u_p then satisfies

$$u_p \sum_j h_j = -\sum_i u_{*n,i}\varDelta_i, \tag{28}$$

which only involves boundary values, and can therefore be applied before the advective step. For more complicated schemes, the numerical equivalent of $\partial\Omega$ has to be obtained numerically from $(DG)^T\psi = 0$, and its support generally extends to points away from the boundary. In those cases, the correction can only be applied after $\vec{u}_*$ has been computed, which itself includes the boundary conditions. Either an iterative procedure should then be used, or $O(\Delta p)$ errors are introduced in the final normal velocities.

It is important to realize that in those cases in which the pressure is known at the boundary, instead of the velocities, Dirichlet boundary conditions can be applied to the Poisson pressure equation. The problem is then no longer singular, but the normal pressure gradient is no longer zero at the boundary, and the correction step changes the normal velocities. Global mass conservation is then automatically ensured. Such boundary conditions can be imposed within the framework of the fractional-step method described in Section 2 and in [2,19], but they have not been implemented in our code.

4. Code validation

In this section the code is applied to two problems whose results are of independent interest. Both will be discussed in future papers from the point of view of flow physics, and in the context of previous simulations and experiments. Here we only concern ourselves with the numerical issues that they raise.

4.1. A turbulent boundary layer with zero pressure gradient

As mentioned in the introduction, turbulent boundary layers have long been subjects of interest, particularly the canonical case with ZPG. This is therefore the subject of our first verification problem, with the goal of maximizing the Reynolds number that can be achieved for a given computational cost. This leads us to avoid transition, and to use a turbulent inflow generator. As already discussed in the introduction, the method chosen is the recycling scheme introduced by Lund in [13].

Both in that reference, where it was applied to drive an large-eddy simulations, and in some later direct simulations [31], the method is typically applied by running a smaller spatially-periodic simulation from which the inflow plane is extracted to the main inflow. If both the auxiliary and the main simulations belong to the same flow, we find little reason to separate them, and here we use a downstream plane of the main simulation, x_{ref}, as a reference from which to synthesize the inflow.

Lund's method is known to be delicate to apply [31,32], and Section 4.2 will be devoted to our experience with it. Before that, we summarize the results of the two sets simulations performed. Their parameters are given in Table 1. The 'short' simulation SB is actually representative of a family of tests used to tune the numerical scheme, whose numerical parameters, such as box height and grid resolution, were varied by factors of up to two from the nominal ones given in the table. Typical grids for these cases are in the range of 100 Mpoints.

The full simulation LB was intended as a production run. It is one of the highest-Reynolds-number boundary layer simulations presently available, but, as mentioned above, only its numerical issues are addresses here. The average streamwise pressure gradient is controlled by applying a constant uniform suction at the upper boundary, which is otherwise stress-free.

Table 1
Parameters of the zero-pressure-gradient simulations. N_x, N_y and N_z are the grid sizes along the three axes, expressed for z in terms of collocation points, and the Δ's are the corresponding resolutions, given at their coarsest points. The values in the top line are representative of the several short-box test simulations (SB), and those at the bottom belong to the production run (LB). The Kolmogorov length η is computed from the local energy dissipation. The coarsest resolution along x and z in terms of η is found at the wall, where $\eta^+ \approx 1.5$. The resolution given in the table for y is reached at $y \approx \delta_{99}/2$, where $\eta^+ \approx 3$. The reference momentum thickness, θ, is taken at the centre of the simulation box.

Re_θ	$(L_x, L_y, L_z)/\theta$	$\Delta x^+, \Delta y^+, \Delta z^+$	$\Delta y/\eta$	N_x, N_y, N_z
600–950	$210 \times 41 \times 43$	$6.7 \times 0.70 \times 4.6$	1.2	$1282 \times 258 \times 386$
620–2140	$535 \times 29 \times 88$	$6.1 \times 0.30 \times 4.1$	1.4	$6145 \times 360 \times 1536$

The transpiration velocity is estimated from the known experimental growth of the displacement thickness, δ^*, in that range of Reynolds numbers. This keeps the acceleration coefficient to $\beta = \delta^* U_\infty^+ \partial_x U_\infty^+ \approx 2 \times 10^{-4}$, which is reasonably small, but the gradient increases sharply to $\beta \approx 5 \times 10^{-3}$ within the last 5% of the numerical domain. This corresponds to the last 1.5 boundary layer thicknesses, and is clearly due to the effect of the outflow, which uses no numerical sponge in this particular case. That region is discarded from the results.

The intensity of the free-stream velocity fluctuations turns out to be controlled by the ratio between the height of the computational box and the boundary layer thickness at the exit, δ_{99e} (see Fig. 4), and remains almost constant with x. The free-stream velocity fluctuations for the LB run are about $2.5 \times 10^{-3} U_\infty$, and the corresponding vorticity fluctuations are about $2 \times 10^{-3} U_\infty/\delta_{99e}$. They are also roughly independent of x, rather than scaling with the local δ_{99}. That scaling and constancy suggest that the residual freestream fluctuations consist mostly of large-scale vorticity waves advected by the free stream, and introduced at the inflow by the sloshing created by the interaction of the boundary layer with the exit. Such fluctuations are unavoidable. The streamwise derivatives $\partial_x v$ and $\partial_x w$ cannot be imposed at the inflow boundary, and there is no way of enforcing the strict irrotationality of the inflow. The incoming vorticity is fixed by the global pressure fluctuations, which are in turn created when the largest eddies leave the domain. The resulting vorticity fluctuations are weak, but because they have sizes comparable to those of the large outgoing eddies, they are not damped by viscosity, and cross the simulation domain essentially undisturbed.

Fig. 5(a) shows the development of the friction coefficients of the two simulations, given[1] in the form $U_\infty^+ = (2/C_f)^{1/2}$, as functions of the Reynolds number, $Re_\theta = U_\infty \theta/\nu$. They are compared with the simulations in [3], and with the experimental results in [33], which cover roughly the same Reynolds number range. The latter experiments are particularly interesting because they were designed to test the effect of the tripping, which is probably comparable to the effect of the numerical inflow, at least in the sense that both mechanisms introduce in the flow some approximation to turbulence that then has to evolve to equilibrium. All their measurements were repeated with three different tripping devices, which are plotted in Fig. 5 using different symbols. The result was that the effect of the trip survives up to $Re_\theta \approx 1500$, and only becomes small beyond that limit. It is seen in Fig. 5(a) that the same is true in our results, which initially diverge widely from the experiments, but eventually settle into excellent agreement with them at about the same location at which the experimental scatter begins to decrease.

Fig. 5(b) and (c) present mean and fluctuation streamwise velocities roughly at the centre of the computational domain of LB, together with the closest available experimental Reynolds numbers. The agreement is excellent in both cases. An older simulation at roughly similar Reynolds number is also included [3]. The minor discrepancies of its fluctuations with both the present results and with the experiments cannot be attributed to the Reynolds number difference, and are probably due to the mean-flow expansion used in [3] to approximate the flow.

4.2. The effect of the inflow

Most of the adjustments required by our code for this particular application were connected with the turbulent inflow condition. This was not due to the incorrect performance of the recycling method itself, which was applied much as described in the original publication [13]. For example, some authors have reported problems initializing simulations from random data [31], and have proposed solutions. Our simulations were initialized from a filtered field from [3], and extended gradually, and did not require any such modification.

The problem seems to arise from the dynamics of the flow itself, and is probably common to most methods of generating synthetic inflows. It is clear from Fig. 5(a) that a substantial initial part of the box has to be discarded. The evolution of flow properties that reside farther from the wall than the friction coefficient actually suggests that the problem affects the first 25% of the box (Fig. 6). This, together with the shorter segment contaminated by the outflow, means that about one third of the box has to be discarded, and that the range of useful Reynolds numbers is reduced from the values in Table 1 to about Re_θ = 1100–2050.

The reason for the long inflow length can be seen in Fig. 7(a), which displays the autocorrelation function

$$C_{uu}(x; x') = \frac{\langle u(x)u(x') \rangle}{(\langle u(x)^2 \rangle \langle u(x')^2 \rangle)^{1/2}}, \tag{29}$$

---

[1] Wall-scaled variables are defined in terms of the local friction velocity $u_\tau(x)$ and of the molecular viscosity ν, and denoted by a + superindex.
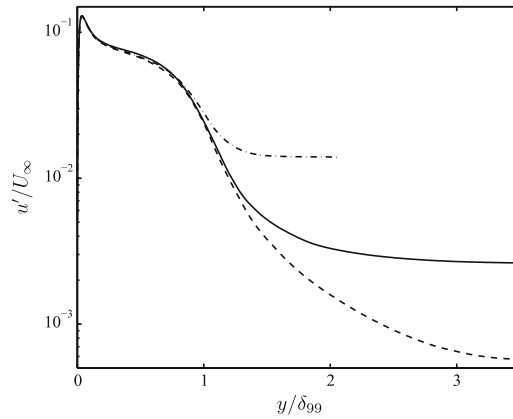
Fig. 4. Fluctuations of the streamwise velocity for three simulations with different box heights. $Re_\theta = 930$. $-\cdot-$, SB, $L_y/\delta_{99e} = 1.75$; —, LB, $L_y/\delta_{99e} = 2.4$; $----$, SB, $L_y/\delta_{99e} = 3.6$.
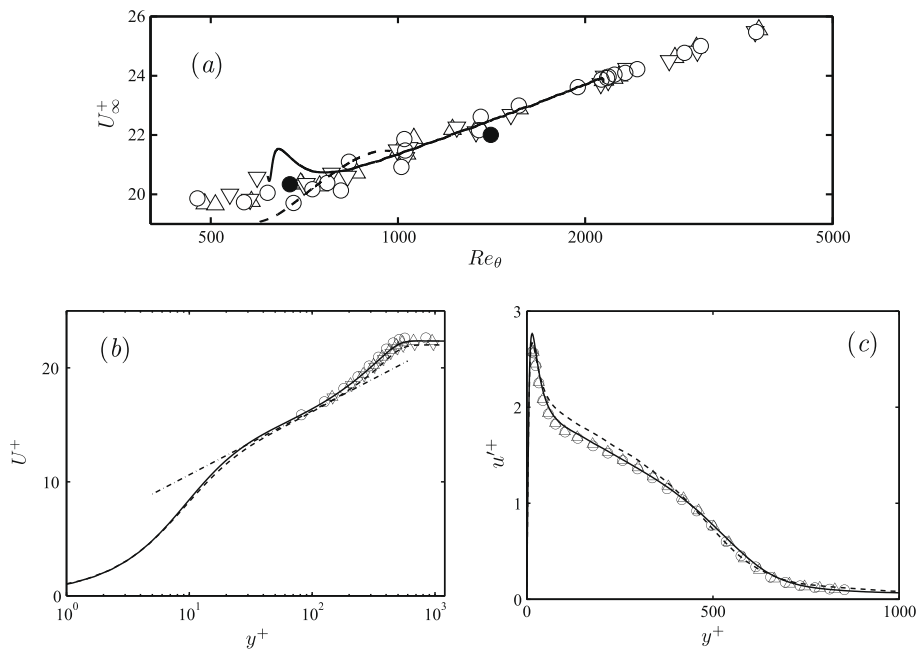

Fig. 5. (a) Friction coefficient versus Reynolds number. Open symbols are experiments from [33], tripped by $\triangle$, wire; $\triangledown$, grit; $\circ$, pins; $\bullet$, simulations in [3]. —, LB simulation; $----$, representative SB simulation. (b) Mean streamwise velocity. —, LB simulation at $Re_\theta = 1350$; $----$, simulation in [3], $Re_\theta = 1410$. Open symbols are as in (a), with $Re_\theta \approx 1350$. $-\cdot-\cdot-$, $\log(y^+)/0.41 + 5$. (c) Root-mean-squared streamwise velocity. Symbols as in (b), but $Re_\theta \approx 1550$, both for [33] and for LB.
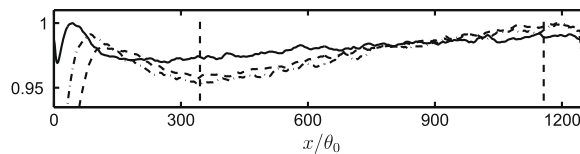

Fig. 6. Peak value of the three velocity-fluctuation intensities, normalized to a common level for the purpose of plotting. Simulation LB. —, u'/u_τ; $----$, v'/u_τ; $-\cdot-$, w'/u_τ. The dashed vertical lines are the limits of the 'useful' range.

where the averaging $\langle \rangle$ is taken over time and spanwise location, and the streamwise velocity u has been filtered to include only a certain band of spanwise Fourier coefficients. The correlation is given as a function of x, and, besides the primary peak at $x = x' \approx 1100\theta_0$, there is a secondary peak at $x = x' - x_{ref} \approx 150\theta_0$. Lund's method can be interpreted as a physical exper-
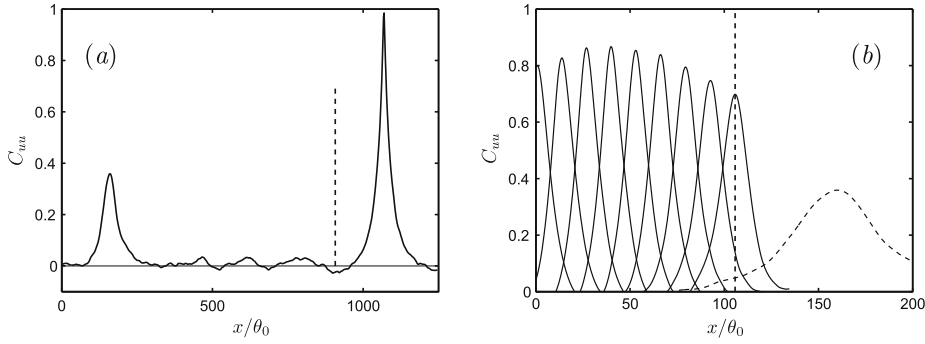
Fig. 7. (a) Correlation $C_{uu}(x; x')$ of the streamwise velocity in simulation LB, as a function of x. (b) Secondary peaks of C_{uu}, for various values of x'. —, SB; ----, LB. For both simulations the correlation is only computed for a band of spanwise wavenumbers $\lambda_z/\delta_{99,ref} = (1.2-2.3)$, at $y/\delta_{99,ref} \approx 0.4$. The dashed vertical line in (a) is the location of the inflow reference plane for LB. The one in (b) is for SB.

iment in which eddies at the reference plane are approximately (except for rescaling) copied to a different position in the boundary layer (the inflow), and evolved. The correlation between the reference and the inflow planes is always large, because one is almost a copy of the other. When the correlation is computed with respect to some location downstream of the reference plane, it reflects, besides the local structure of the eddies, the correspondence between eddies that have been advected from the reference and from the inflow planes. This is the origin of the secondary peak in Fig. 7(a), and its decay with $x' - x_{ref}$ is a measure of the Lagrangian decorrelation time of the eddies as they are advected by the mean velocity. Both the range of spanwise wavenumbers in Fig. 7, and the y location of the correlations, have been chosen to maximize the amplitude of the secondary peak, and it is interesting that they correspond to spanwise wavelengths of the order of $\lambda_z/\delta_{99,ref} \approx 1.5$, which is somewhat wider than, but of the same order as, the large-scale structures identified in this region of experimental boundary layers [34].

The decay of the secondary peak as it moves away from the inflow is shown in Fig. 7(b), and is exceedingly slow. The peaks from the short simulation, in which the reference plane is close to the inflow, have decayed very little by the time they reach x_{ref}. A secondary peak from the LB simulation, in which $x_{ref}/\theta_0 = 850$, and for which the correlation of the inflow with the reference plane is presumably lower than in SB because of the larger rescaling ratio, is still clearly visible at $x/\theta_0 \approx 160$. It has been known for a long time that the decay of the space-time velocity correlation in boundary layers is much slower when the two points are at "optimum" separations than when they are at the same location [35], showing that the eddies stay coherent while being advected for much longer distances than their lengths. For example, Favre et al. [35] found that, while the correlation length of u was only a few δ_{99}, the correlation along the optimum advection speed had only decayed to about 0.8 after $4\delta_{99} (\approx 35\theta)$. Fig. 7(b) suggests that eddies would stay coherent for $200-300\theta$, which is consistent with the lengths found for the persistence of the tripping influence in [33]. The physical reason for the long inflow length is that the turnover time for a large eddy of size $O(\delta)$, with internal velocities $O(u_\tau)$, is δ/u_τ, during which time the eddy is advected by about $U_\infty \delta/u_\tau$. The implied accommodation lengths, of order $U_\infty^+ \delta$, are consistent with the values found above.

Note that the actual persistence of the eddies is probably about twice longer than what can be deduced from Fig. 7(b), because the inflow 'experiment' tests the decorrelation of two eddies as they diverge from each other, while a single eddy probably decorrelates with itself about twice slower. Note also that the apparent growth of the correlation at intermediate locations of Fig. 7(b) is due to the correlation being computed at a constant y, while the maximum correlation is known to occur between eddies at similar values of y/δ_{99} [35].

The practical consequence from the point of view of numerical inflow conditions is that the first few hundred momentum thicknesses of a boundary layer are controlled by the inflow, and that they cannot therefore be used to investigate the structure of turbulence. Better inflow fields may be useful to create more realistic flows, which could for example be used to investigate the effects of turbulence on some other aspect of the flow, but when the physics of the turbulence itself is what is being investigated, the first few hundred momentum thicknesses essentially reflect the hypotheses made for the inflow conditions. Note that this is a problem of any inflow, not only of the recycling method, and that the inflow length estimated here would have to be added to any other length contaminated by local numerical manipulations (e.g. fringes).

In the case of Lund's method, the eddy persistence can give rise to numerical artifacts if the reference plane is chosen too close to the inlet. An example is shown in Fig. 8, which corresponds to an early example of the SB simulations in which $x_{ref}/\theta_0 = 32$. The inflow couples to the reference plane, creating an artificial periodicity that can be seen even in instantaneous velocity maps. Doubling the reference distance to $x_{ref}/\theta_0 = 60$ decreases the effect, although it is still visible in the correlations. No periodicity can be detected in the LB simulation, where $x_{ref}/\theta_0 = 850$.

When the inflow coupling is added to other numerical problems, the effect can be even more marked. In another SB test, with an reference plane close to the inflow and a low upper boundary, the coupling became strong enough to create periodic vorticity 'chimneys' extending across the whole height of the box, raising the free-stream turbulence level to $O(1)$. Similar effects in a pipe were reported in [36].
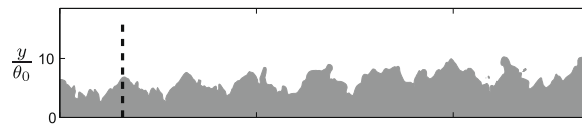
Fig. 8. Streamwise $(x - y)$ section of an SB simulation in which the inflow reference plane is at $x_{ref}/\theta_0 = 32$, represented by the dashed vertical line. The gray area is $U < 0.9U_\infty$. Note the spurious periodic structures.

4.3. Turbulent boundary layer with a strong adverse pressure gradient

A more demanding case than the one just described is the turbulent boundary layer under a strong APG, which occurs, for example, in flows around airfoils and other obstacles. In this section we describe a simulation over a flat plate with boundary conditions modelling the flow typically found on the suction side of low-pressure turbine blades.

The characteristic linear decay of the free-stream velocity in such flows is obtained by imposing a vertical suction velocity at the top simulation boundary. Fig. 10(a) shows the suction velocity, V_c, which is constant except near the endpoints. The constant $V_c = aL_y$ is calculated approximately from the desired linearly decaying streamwise velocity $U_\infty(x) = U_\infty(0) - ax$, and the acceleration a is chosen to match a typical low-pressure turbine APG.

As shown in Fig. 10(a), the suction matches the v-inflow profile at $x = 0$, and decreases and becomes briefly negative near the exit of the numerical domain. These deviations from constant suction at both ends of the computational domain were found necessary to prevent numerical artifacts. For example, when the suction was kept constant up to the exit plane, separation occurred over the last few grid points, and the associated upstream velocities interacted with the outflow boundary condition to cause numerical divergence.

The u and w velocities at the top boundary satisfy free slip conditions, $\partial_y u = \partial_y w = 0$, which are equivalent to irrotationality except where v is not constant. The wall satisfies no-slip for u, w, and impermeability for v.

Under the conditions of the simulation, the flow is initially laminar, separates, transitions within the short separation bubble, reattaches as a consequence of the transition, and develops into an attached turbulent APG boundary layer over roughly half of the simulation domain (Fig. 9).

The discretization uses $N_x \times N_y \times N_z = 1539 \times 301 \times 768$ collocation points over a box of size $L_x/\delta_{99e} \times L_y/\delta_{99e} \times L_z/\delta_{99e} \approx 8.7 \times 2.5 \times 4.1$. The strength of the imposed APG is illustrated by the difference of these values with those expressed in terms of the boundary layer thickness measured at the inlet, $L_x/\delta_{99i} \times L_y/\delta_{99i} \times L_z/\delta_{99i} \approx 173 \times 49 \times 82$.

Since the flow is close to separation over much of the domain, wall units based on the friction velocity are not particularly relevant as resolution criteria. The grid size compared with the Kolmogorov scale is a function of both x and y, and varies from $\Delta x = 6.3\eta$ at the transition point, to around 2η in the fully turbulent part. The spanwise resolution is $\Delta z \approx \Delta x$, expressed in terms of collocation points. The wall-normal resolution is $\Delta y/\eta \approx 1$ near the wall at reattachment, and $\Delta y/\eta \approx 0.5$ in the fully turbulent part. Farther from the wall the maximum of this ratio remains approximately equal to 2.8 over the whole domain length. Note that the choice of grid spacings departs from the usual practice in attached flows, $\Delta z \approx \Delta x/2$. In fact, Δz was chosen following the same criteria as in the ZPG simulation, but Δx had to be made finer to capture the small scales created at transition. For reasons explained when discussing the calculation of the pressure, this fine spacing was then maintained throughout the simulation.

A laminar Hiemenz profile [37] is imposed at the inflow to approximate u and v, and a stationary three dimensional perturbation is generated by adding
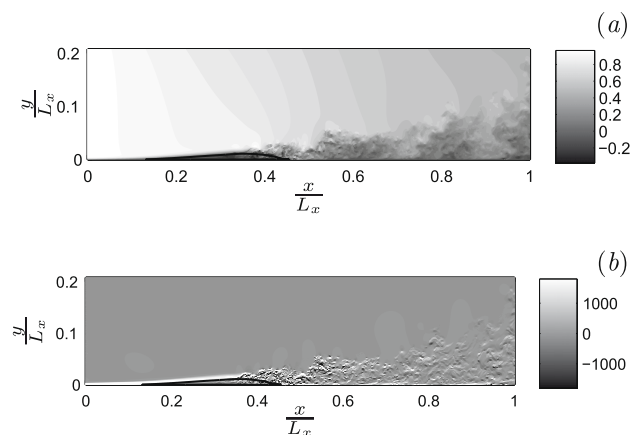

Fig. 9. Instantaneous x–y sections of, (a) the streamwise velocity, and (b) the spanwise vorticity. The solid line in both figures is the zero contour of u, and marks the location of the separation bubble.

$$u_{pb}(y,z) = 0.038 U_\infty \phi(y) \left[\frac{\sin \kappa_1 z - \cos \kappa_1 z}{2} + \sin \kappa_2 z + \sin \kappa_3 z \right], \tag{30}$$

to the streamwise velocity. The perturbation wavenumbers are $\kappa_1 = 3$, $\kappa_2 = 7$ and $\kappa_3 = 80$, corresponding to wavelengths of approximately $27\delta_{99i}$, $12\delta_{99i}$ and δ_{99i}. The function

$$\phi(y) = \frac{1}{2} U_0(y) \tanh(0.5 - 0.25j + 0.02N_y), \tag{31}$$

where j is the grid index, was chosen by trial and error to confine the perturbation to the neighbourhood of the wall. Note that, in codes with a spanwise spectral representation like the present one, some spanwise variation has to be explicitly added for transition to occur, since otherwise the flow would remain strictly two-dimensional. Since the goal of the simulation is to study the flow in the APG turbulent range, rather than to control transition, no effort was made to optimize the
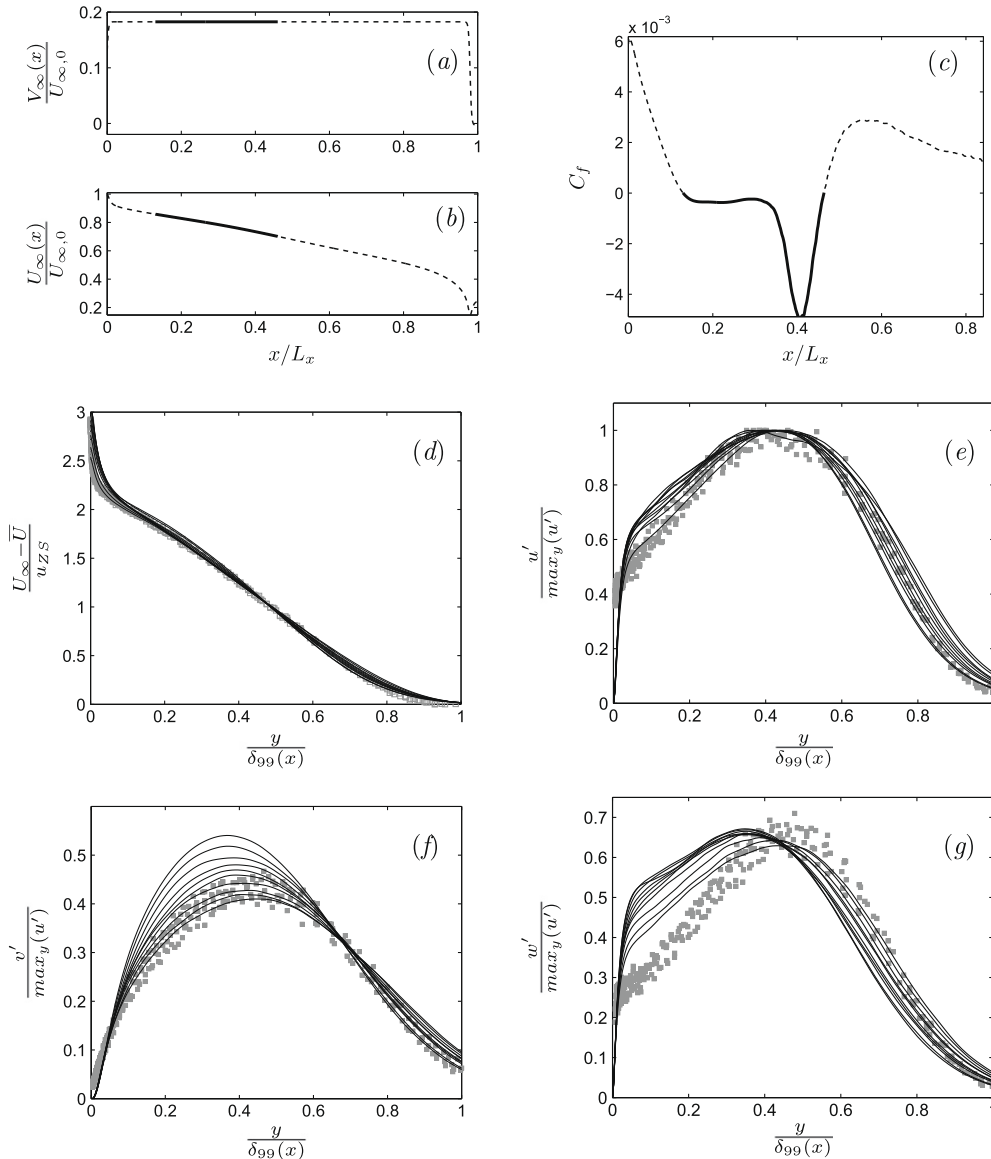

Fig. 10. (a) Vertical suction velocity used to generate the APG. (b) Streamwise velocity at the upper boundary as a function of x. (c) Friction coefficient as a function of x. (d) Mean velocity in outer scaling. (e) Root-mean-squared streamwise velocity. (f) Root-mean-squared wall-normal velocity. (g) Root-mean-squared spanwise velocity. For figures (a–c), dashed segments represent areas of attached flow, and solid ones, separated flow. For figures (d, e-g), the solid lines are numerical data from the present simulation, and the symbols are experimental data from [40] at $Re_\theta = 4-5 \times 10^4$. Figures (a–b) show data for the whole streamwise length, while figures (c–g) show only data for the useful region defined in the text.

spanwise inflow perturbation, except to make sure that its wavelengths span the order of magnitude in which separated shear layers are known to be sensitive to three-dimensionality [38,39], and to include incommensurate wavelengths to prevent spurious symmetries.

Typical instantaneous cross sections of the flow are shown in Fig. 9. They illustrate the excellent resolution in the transition region, and show that the outflow boundary condition, although very close to being separated, does not introduce observable spurious modes, even if no numerical viscous buffer is used in the simulation.

The suction at the top boundary is intended to approximate a linearly decreasing free-stream velocity $U_\infty(x)$. Fig. 10(b) shows that this is approximately achieved, and that the influence of the separation bubble on the external potential flow is negligible. Earlier simulations had shown that in lower domains the separation bubble causes $U_\infty(x)$ to develop a strong plateau.

The influence of negative suction applied near the exit is also clearly seen in Fig. 10(b), and extends over approximately $1.5\delta_{99e}$, as in the zero-pressure-gradient case. Once this region is discarded, the useful range of turbulent Reynolds numbers is about $Re_\theta = 1000-1600$.

The friction coefficients at the wall is shown in Fig. 10(c). Its most striking feature is the large negative peak close to reattachment. This is typical of laminar and transitional separation bubbles and was also reported in [9]. After the layer reattaches, its friction coefficient decreases again almost linearly and becomes very close to zero at the exit. The present APG $(-\delta^*\partial_x U_\infty/U_\infty = 0.015-0.066)$ is probably very close to the strongest one that can be imposed in this domain without going through a second turbulent separation.

Fig. 10(d) shows the mean streamwise velocity for the present simulation and for the experiments in [40], scaled with the reference velocity $u_{ZS} = U_{\infty,x}\delta^*/\delta_{99}$ suggested by [41]. The agreement is remarkable, considering the very different flow development and Reynolds numbers of the two cases, although part of the reason is doubtlessly that u_{ZS} is designed to make the area underneath the profiles equal to unity. No equivalent velocity scale is known for the fluctuations, and the comparisons in Fig. 10(e)–(g) between our results for u', v', w' and the experiments are normalized so that the maximum of the streamwise velocity fluctuation profile is unity for all cases. It shows that the two layers have qualitatively similar fluctuation profiles. The agreement is not as good as in the ZPG case discussed above, but there are no experiments in this case at Reynolds numbers comparable to the simulation, or even with matching acceleration parameters. For example, the differences near the wall most probably reflect the much higher Reynolds number of the experiments.

5. Summary and conclusions

We have presented a new numerical code for boundary layer simulations that can be used with and without pressure gradients, and with arbitrary time-dependent boundary conditions for the velocity. It uses a fractional-step primitive-variable formulation with fourth-order compact finite differences on a staggered grid for the convective and viscous terms, and a standard second-order scheme for the divergence of the velocity and for the gradient of the pressure. It is shown that, although the code is formally only second-order in space, the resolution limits for the velocities are those of the fourth-order derivatives. This is validated on a Taylor–Green decay simulation.

Because the pressure is only second-order accurate, its Poisson equation is solved using a direct method in combination with Fourier decomposition. A concise way is presented to impose arbitrary mass fluxes over the domain boundaries, and to conserve mass globally for general schemes.

The high resolution achieved by mixing different order operators for the advective at pressure terms, and the concise way to impose mass conservation, are the two features that made the use of the fractional step method feasible for our purposes.

The code has been verified on two relatively large-scale simulations of turbulent boundary layers. The first is a zero-pressure-gradient case with a useful range of $Re_\theta \approx 1100-2050$, within which the agreement with experimental results is excellent.

The turbulent inflow is synthesized using Lund's [13] recycling method. It is found that the first several hundred momentum thicknesses have to be discarded because they retain the effect of the synthetic inflow. It is argued that this is a property of the flow itself, similar to the persistence of tripping effects, and not of the particular inflow scheme. Together with a shorter contaminated segment at the outflow, it amounts to about one third of our longest simulation box, and spans the full extent of shorter boxes used for code tuning. A similar fraction was mentioned for the fringe method in [4], but the present limit is not connected with the numerical manipulation of the affected region, and our results suggest that both lengths would have to be added in cases in which part of the flow is locally contaminated by the numerics.

The same eddy persistence responsible for the inflow length implies that the reference plane used to synthesize the incoming turbulence should be as far as possible from the inflow, or at least farther than 20–30 boundary layer thicknesses. Otherwise the two planes tend to couple, creating spurious flow periodicities that, in extreme cases, may result in large free-stream oscillations, or even in numerical divergence. An obvious constraint is that the plane used as reference should not be within the part of the flow corrupted by the exit condition. A less obvious one is that the rescaling ratio increases with the distance from the reference to the inflow, so that the incoming turbulence becomes less realistic as that distance is increased. The value chosen for the present simulation, $x_{ref} = 850\theta_0 \approx 90\delta_{99,0}$ is probably close to the largest practical one.

The second example is a simulation of a laminar separation bubble, followed by transition to turbulence, reattachment, and the development of a turbulent boundary layer under strong adverse pressure gradient. The mean velocity and the

streamwise velocity fluctuations compare very well with experimental results at higher Reynolds numbers, although no scaling velocity is known in this case to make the comparison quantitative.

The code is parallelized in MPI, using a global transpose from y–z to x—y planes, and has been run on up to 512 processors in IBM and Intel clusters, with essentially linear scaling. It is easily transportable to other similar systems. Typical running times for the large ZPG simulation on 512 processors is 90 s/step, with 35% of the time devoted to communications.

References

[1] J. Kim, P. Moin, Application of a fractional-step method to incompressible Navier–Stokes equations, J. Computat. Phys. 59 (1985) 308–323.
[2] J.B. Perot, An analysis of the fractional step method, J. Computat. Phys. 108 (1993) 51–58.
[3] P.R. Spalart, Direct simulation of a turbulent boundary layer up to Re_θ = 1410, J. Fluid Mech. 187 (1988) 61–98.
[4] P.R. Spalart, J.H. Watmuff, Experimental and numerical study of a turbulent boundary layer with pressure gradients, J. Fluid Mech. 249 (1993) 337–371.
[5] J. Kim, P. Moin, R.D. Moser, Turbulence statistics in fully developed channel flow at low Reynolds number, J. Fluid Mech. 177 (1987) 133–166.
[6] S. Hoyas, J. Jiménez, Scaling of the velocity fluctuations in turbulent channels up to R_τ = 2003, Phys. Fluids 18 (2006) 011702.
[7] P.R. Spalart, Numerical study of sink-flow boundary layers, J. Fluid Mech. 172 (1986) 307–328.
[8] J. Nordström, N. Nordin, D. Henningson, The fringe region technique and the Fourier method used in the direct numerical simulation of spatially evolving viscous flows, SIAM J. Sci. Comput. 90 (1999) 1365–1393.
[9] M. Alam, N.D. Sandham, Direct numerical simulation of short laminar separation bubbles with turbulent reattachment, J. Fluid Mech. 410 (2000) 1–28.
[10] M. Skote, D.S. Henningson, Direct numerical simulation of a separated turbulent boundary layer, J. Fluid Mech. 471 (2002) 107–136.
[11] G. Khurajadze, M. Oberlack, DNS and scaling laws from new symmetry groups of zpg turbulent boundary layer flow, Theor. Computat. Fluid Dyn. 18 (2004) 391–441.
[12] A. Keat, U. Piomelli, E. Ballaras, H. Kaltenbach, A priori and a posteriori tests of inflow conditions for large-eddy simulations, Phys. Fluids 16 (2004) 4696–4712.
[13] T.S. Lund, X. Wu, K.D. Squires, Generation of turbulent inflow data for spatially-developing boundary layer simulations, J. Computat. Phys. 140 (1998) 233–258.
[14] H. Le, P. Moin, Direct numerical simulation of turbulent flow over a backwards-facing step, Ph.D. Thesis, Thermosciences Division, Department of Mechanical Engineering, Stanford University, 1994.
[15] Y. Na, P. Moin, Direct numerical simulation of turbulent boundary layers with adverse pressure gradient and separation, Ph.D. Thesis, Thermosciences Division, Department of Mechanical Engineering, Stanford University, 1996.
[16] X. Wu, R.G. Jacobs, J.C.R. Hunt, P.A. Durbin, Simulation of boundary layer transition induced by periodically passing wakes, J. Fluid Mech. 398 (1999) 109–153.
[17] J.A. Chorin, Computational Fluid Mechanics: Selected Papers, Academic Press, 1989.
[18] R.C.D.L. Brown, M.L. Minion, Accurate projection methods for the incompressible Navier–Stokes equations, J. Computat. Phys. 168 (2001) 464–499.
[19] J.K. Dukowicz, A.S. Dvinsky, Approximate factorization as a high order splitting for the implicit incompressible flow equations, J. Computat. Phys. 102 (1992) 336–347.
[20] S. Armfield, R. Street, The fractional-step method for the Navier–Stokes equations on staggered grids: the accuracy of three variations, J. Computat. Phys. 153 (1999) 660–665.
[21] P. Orlandi, Fluid Flow Phenomena, Kluwer, 2000.
[22] S.K. Lele, Compact finite difference schemes with spectral-like resolution, J. Computat. Phys. 103 (1992) 16–42.
[23] S. Nagarajan, S.K. Lele, J.H. Ferziger, A robust high-order compact method for large eddy simulation, J. Computat. Phys. 191 (2003) 329–419.
[24] H.L. Meitz, H.F. Fasel, A compact-difference scheme for the Navier–Stokes equations in vorticity–velocity formulation, J. Computat. Phys. 157 (2000) 371–403.
[25] U. Rist, H.F. Fasel, Direct numerical simulation of controlled transition in a flat-plate boundary layer, J. Fluid Mech. 298 (1995) 211–248.
[26] M.P. Simens, The study and control of wall-bounded flows, Ph.D. thesis, Aeronautics, U. Polit. Madrid. <http://oa.upm.es/1047/>, 2008.
[27] K. Akselvoll, P. Moin, An efficient method for temporal integration of the Navier–Stokes equations in confined axisymmetric geometries, J. Computat. Phys. 125 (1996) 454–463.
[28] S. Orszag, Numerical simulation of incompressible flows within simple boundaries: accuracy, J. Fluid Mech. 49 (1971) 75–112.
[29] P. Moin, Fundamentals of Engineering Numerical Analysis, Cambridge University Press, 2001.
[30] J.H. Ferziger, M. Peric, Computational Methods for Fluid Dynamics, Springer, 1996.
[31] A. Ferrante, S.E. Elgobashi, A robust method for generating inflow conditions for direct simulations of spatially-developing turbulent boundary layers, J. Computat. Phys. 198 (2004) 372–387.
[32] K. Liu, R.H. Pletcher, Inflow conditions for the large-eddy simulation of turbulent boundary layers: a dynamic procedure, J. Computat. Phys. 219 (2006) 1–6.
[33] L.P. Erm, P.N. Joubert, Low-Reynolds-number turbulent boundary layers, J. Fluid Mech. 230 (1991) 1–44.
[34] C.D. Tomkins, R.J. Adrian, Energetic spanwise modes in the logarithmic layer of a turbulent boundary layer, J. Fluid Mech. 545 (2005) 141–162.
[35] A.J. Favre, J.J. Gaviglio, R.J. Dumas, Further space-time correlations of velocity in a turbulent boundary layer, J. Fluid Mech. 3 (1958) 344–356.
[36] N. Nikitin, Spatial periodicity of spatially evolving turbulent flow caused by inflow boundary condition, Phys. Fluids 19 (2007) 091703.
[37] H. Schlichting, Boundary Layer Theory, sixth ed., McGraw-Hill, 1968.
[38] R. Pierrehumbert, S. Widnall, The two- and three-dimensional instabilities of a spatially periodic shear layer, J. Fluid Mech. 114 (1982) 59–82.
[39] J. Jiménez, A spanwise structure in the plane mixing layer, J. Fluid Mech. 132 (1983) 319–336.
[40] P.E. Skare, P.A. Krogstad, A turbulent equilibrium boundary layer near separation, J. Fluid Mech. 272 (1994) 319–348.
[41] M.V. Zagarola, A.J. Smits, Mean-flow scaling of turbulent pipe flow, J. Fluid Mech. 373 (1998) 33–79.