Journal of Computational Physics 229 (2010) 5373–5393

Contents lists available at ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp

An implicit high-order spectral difference approach for large eddy simulation

M. Parsani [a,*], G. Ghorbaniasl [a], C. Lacor [a], E. Turkel [b]

[a] Vrije Universiteit Brussel, Department of Mechanical Engineering, Fluid Dynamics and Thermodynamics Research Group, Pleinlaan 2, 1050 Brussel, Belgium
[b] Tel-Aviv University, Department of Mathematics, Tel-Aviv, Ramat Aviv 69978, Israel

ARTICLE INFO

Article history:
Received 14 December 2009
Received in revised form 24 March 2010
Accepted 24 March 2010
Available online 31 March 2010

Keywords:
High-order spectral difference method
Large eddy simulation
Wall-adapting local eddy-viscosity model
Implicit LU-SGS algorithm

ABSTRACT

The filtered fluid dynamic equations are discretized in space by a high-order spectral difference (SD) method coupled with large eddy simulation (LES) approach. The subgrid-scale stress tensor is modelled by the wall-adapting local eddy-viscosity model (WALE). We solve the unsteady equations by advancing in time using a second-order backward difference formulae (BDF2) scheme. The nonlinear algebraic system arising from the time discretization is solved with the nonlinear lower–upper symmetric Gauss–Seidel (LU-SGS) algorithm. In order to study the sensitivity of the method, first, the implicit solver is used to compute the two-dimensional (2D) laminar flow around a NACA0012 airfoil at $Re = 5 \times 10^5$ with zero angle of attack. Afterwards, the accuracy and the reliability of the solver are tested by solving the 2D "turbulent" flow around a square cylinder at $Re = 10^4$ and $Re = 2.2 \times 10^4$. The results show a good agreement with the experimental data and the reference solutions.

1. Introduction

Spatially high-order accurate compact schemes are now well established as accurate methods to solve a variety of flow problems. In Computational Fluid Dynamics (CFD), they are used for Direct Numerical Simulation (DNS), Large Eddy Simulation (LES), Computational Aeroacoustics (CAA) etc., where the accurate resolution of small scales is required. Moreover, since CFD is increasingly used as an industrial design and analysis tool, high-order accuracy must be achieved on unstructured grids which are required for efficient meshing. These needs have been the driving force for the development of higher order schemes for unstructured meshes such as the Discontinuous Galerkin (DG) method [1–5], the Spectral Volume (SV) method [6–14] and the Spectral Difference (SD) method [15–17]. All these methods use piecewise continuous functions as the solution approximation space. They are capable of achieving high-order accuracy on unstructured grids and they have a compact stencil, which makes them easily parallelizable.

The SD method has an important advantage over the DG and SV methods, that no integrals have to be evaluated to compute the residuals, thus avoiding the need for costly high-order accurate quadrature formulas. Recently, there has been research on unifying several of the popular methods including the DG method, the SV method and the SD method with a technique that does not require the evaluation of the integral [18–20].

So far, the performance of the SD method has been investigated by Liang et al. [21] to solve the compressible turbulent channel flow at $Re_\tau = 400$ without modelling the contribution from the unresolved scales in the flow field. In [21] a good

* Corresponding author.
E-mail addresses: mparsani@vub.ac.be (M. Parsani), ghader.ghorbaniasl@vub.ac.be (G. Ghorbaniasl), clacor@vub.ac.be (C. Lacor), turkel@post.tau.ac.il (E. Turkel).

0021-9991/$ - see front matter © 2010 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2010.03.038

agreement with DNS results of Moser et al. [22] has been found. However, although spatially high-order accurate numerical schemes guarantee the accurate resolution of small scales, their application for the simulation of general turbulent flows implies that particular attention has still to be paid to subgrid models. In this framework, we couple, for the first time, a high-order SD scheme on unstructured quadrilateral grids with the local eddy-viscosity (WALE) model [23,24] to compute large eddy simulations.

One property of a good LES model is that its use in a laminar or low Reynolds number flow results in a solution which is very close to the solution obtained by solving the Navier–Stokes equations, i.e. the LES model should replicate laminar flows [25]. Therefore, the sensitivity of the method is first investigated for the flow 2D flow around a NACA0012 airfoil at $Re = 5 \times 10^5$, $M = 0.4$ and zero degrees angle of attack, with a fourth-order SD scheme and quadratic boundary elements. The accuracy and the reliability of the of the new coupling between the SD method and the WALE model are tested by solving the 2D "turbulent" flow around a square cylinder at $Re = 10^4$ and $Re = 2.2 \times 10^4$ with a third-order SD scheme. The obtained results are compared with DNS solutions, LES and experimental results reported in literature.

When high-order schemes are combined with classical solution methods, such as Explicit Runge–Kutta (E-RK) solvers, they suffer from a restrictive CFL condition and hence a relatively slow convergence rate. In addition, the solver must also be able to deal with the geometrical stiffness imposed by the Navier–Stokes grids where high-aspect ratios occur near walls. In the case of compressible solvers there is an additional stiffness when solving for low speed flows caused by the disparate eigenvalues of the system. Therefore, efficient solvers are needed to fully fulfill the potential of high-order methods. Implicit time-integration schemes can be used to deal with these problems. These schemes can advance the solution with significantly larger time steps compared with explicit methods. However, they may be more expensive than explicit schemes if the algebraic solver employed is not efficient. In the present study, the unsteady filtered fluid dynamic equations are solved by advancing in time using a second-order Backward Difference Formula (BDF2) scheme. The nonlinear algebraic system arising from the time discretization is then solved with the nonlinear Lower–Upper Symmetric Gauss–Seidel (LU-SGS) algorithm. The LU scheme was started by Jameson and Turkel [26] and later reformulated to use symmetric Gauss–Seidel by Jameson and Yoon in the context of second order central schemes [27]. It was recently rediscovered by Sun et al. [28] and adapted for use with SD schemes for steady state computations. In Parsani et al. [29] the nonlinear LU-SGS algorithm in combination with the BDF2 scheme was coupled with the SD method to solve the 2D unsteady Navier–Stokes equations for laminar/"turbulent" test cases.

The remainder of this article is organized as follows. A brief summary of the filtered fluid dynamic equation for a compressible flow and the description of the WALE model are given in Section 2. Section 3 is devoted to the description of the SD method and the treatment of the diffusive terms with a fully compact approach, i.e. the second approach of Bassi and Rebay [30]. In the same section the coupling of the SD method and the WALE model through a new definition of the grid filter width is given. The nonlinear LU-SGS algorithm combined with the BDF2 scheme is described in Section 4. Section 5 presents numerical results, before finally drawing conclusions in Section 6.

2. Governing equations for large eddy simulation

In this section the system of the LES fluid dynamic equations for a compressible flow are presented. The three physical conservation laws for a general Newtonian fluid, i.e. the continuity, the momentum and energy equations, are introduced using the following notation: ρ for the mass density, $\vec{u} \in \mathbb{R}^{dim}$ for the velocity vector in a physical space with *dim* dimensions, P for the static pressure and E for the specific total energy which is related to the pressure and the velocity vector field by

$$E = \frac{1}{\gamma - 1}\frac{P}{\rho} + \frac{|\vec{u}|^2}{2}, \tag{1}$$

where γ is the constant ratio of specific heats and it is 1.4 for air.

Define a vector $\boldsymbol{w}$ of all the filtered conservative variables, i.e.

$$\boldsymbol{w} = \begin{pmatrix} \overline{\rho} \\ \overline{\rho}\tilde{\vec{u}} \\ \overline{\rho}\tilde{E} \end{pmatrix}, \tag{2}$$

where the symbols $\bar{(\cdot)}$ and $\tilde{(\cdot)}$ represent respectively the spatially filtered field and the Favre filtered field defined as $\tilde{\vec{u}} = \overline{\rho\vec{u}}/\overline{\rho}$. The system of the filtered fluid dynamic equations for a compressible flow, written in divergence form and equipped with suitable initial-boundary conditions, is

$$\frac{\partial \boldsymbol{w}}{\partial t} + \vec{\nabla} \cdot \left(\vec{\mathbf{f}}_C(\boldsymbol{w}) - \vec{\mathbf{f}}_D(\boldsymbol{w}, \vec{\nabla}\boldsymbol{w}) \right) = \frac{\partial \boldsymbol{w}}{\partial t} + \vec{\nabla} \cdot \vec{\mathbf{f}} = 0, \tag{3}$$

$\vec{\mathbf{f}}_C(\boldsymbol{w}) = [\mathbf{f}_C, \mathbf{g}_C, \mathbf{h}_C]^T$ and $\vec{\mathbf{f}}_D\left(\boldsymbol{w}, \vec{\nabla}\boldsymbol{w}\right) = [\mathbf{f}_D, \mathbf{g}_D, \mathbf{h}_D]^T$ represent the convective and the diffusive fluxes, respectively. In a general 3D ($dim = 3$) Cartesian space, $\vec{x} = [x_1, x_2, x_3]^T$, the components of these fluxes are given by

$$\mathbf{f}_C = \begin{pmatrix} \overline{\rho}\tilde{u}_1 \\ \overline{\rho}\tilde{u}_1^2 + \overline{P} \\ \overline{\rho}\tilde{u}_1\tilde{u}_2 \\ \overline{\rho}\tilde{u}_1\tilde{u}_3 \\ \tilde{u}_1(\overline{\rho}\widetilde{E} + \overline{P}) \end{pmatrix}, \quad \mathbf{g}_C = \begin{pmatrix} \overline{\rho}\tilde{u}_2 \\ \overline{\rho}\tilde{u}_1\tilde{u}_2 \\ \overline{\rho}\tilde{u}_2^2 + \overline{P} \\ \overline{\rho}\tilde{u}_2\tilde{u}_3 \\ \tilde{u}_2(\overline{\rho}\widetilde{E} + \overline{P}) \end{pmatrix}, \quad \mathbf{h}_C = \begin{pmatrix} \overline{\rho}\tilde{u}_3 \\ \overline{\rho}\tilde{u}_1\tilde{u}_3 \\ \overline{\rho}\tilde{u}_2\tilde{u}_3 \\ \overline{\rho}\tilde{u}_3^2 + \overline{P} \\ \tilde{u}_3(\overline{\rho}\widetilde{E} + \overline{P}) \end{pmatrix},$$

$$\mathbf{f}_D = \begin{pmatrix} 0 \\ \tilde{\sigma}_{11} - \tau_{11}^{sgs} \\ \tilde{\sigma}_{21} - \tau_{21}^{sgs} \\ \tilde{\sigma}_{31} - \tau_{31}^{sgs} \\ (\tilde{\sigma}_{11} - \tau_{11}^{sgs})\tilde{u}_1 + (\tilde{\sigma}_{21} - \tau_{21}^{sgs})\tilde{u}_2 + (\tilde{\sigma}_{31} - \tau_{31}^{sgs})\tilde{u}_3 + c_P\frac{\mu}{Pr}\frac{\partial\tilde{T}}{\partial x_1} - q_1^{sgs} \end{pmatrix}, \quad \mathbf{g}_D = \begin{pmatrix} 0 \\ \tilde{\sigma}_{12} - \tau_{12}^{sgs} \\ \tilde{\sigma}_{22} - \tau_{22}^{sgs} \\ \tilde{\sigma}_{32} - \tau_{32}^{sgs} \\ (\tilde{\sigma}_{12} - \tau_{12}^{sgs})\tilde{u}_1 + (\tilde{\sigma}_{22} - \tau_{22}^{sgs})\tilde{u}_2 + (\tilde{\sigma}_{32} - \tau_{32}^{sgs})\tilde{u}_3 + c_P\frac{\mu}{Pr}\frac{\partial\tilde{T}}{\partial x_2} - q_2^{sgs} \end{pmatrix},$$

$$\mathbf{h}_D = \begin{pmatrix} 0 \\ \tilde{\sigma}_{13} - \tau_{13}^{sgs} \\ \tilde{\sigma}_{23} - \tau_{23}^{sgs} \\ \tilde{\sigma}_{33} - \tau_{33}^{sgs} \\ (\tilde{\sigma}_{13} - \tau_{13}^{sgs})\tilde{u}_1 + (\tilde{\sigma}_{23} - \tau_{23}^{sgs})\tilde{u}_2 + (\tilde{\sigma}_{33} - \tau_{33}^{sgs})\tilde{u}_3 + c_P\frac{\mu}{Pr}\frac{\partial\tilde{T}}{\partial x_3} - q_3^{sgs} \end{pmatrix},$$

where c_P, μ, Pr and T represent respectively the specific heat capacity at constant pressure, the dynamic viscosity, the Prandtl number and the temperature of the fluid. Moreover, σ_{ij} represents the ij-component of the resolved viscous stress tensor defined as

$$\tilde{\sigma}_{ij} = 2\mu\left(\tilde{S}_{ij} - \frac{\delta_{ij}}{3}\tilde{S}_{mm}\right) \quad i,j = 1,\ldots,dim, \tag{4}$$

$$\tilde{S}_{ij} = S_{ij}\left(\tilde{\vec{u}}\right) = \frac{1}{2}\left(\frac{\partial\tilde{u}_i}{\partial x_j} + \frac{\partial\tilde{u}_j}{\partial x_i}\right) \quad i,j = 1,\ldots,dim, \tag{5}$$

where δ_{ij} is the Kronecker Delta function.

From the definitions of the fluxes components it is seen that both the momentum and the energy equations differ from the classical fluid dynamic equations only for two terms which take into account the contributions from the unresolved scales. These contributions, represented by the specific subgrid-scale stress tensor τ_{ij}^{sgs} and by the subgrid-scale heat-flux vector defined q_i^{sgs}, appear when the spatial filter is applied to the convective terms and they are defined as follows

$$\tau_{ij}^{sgs} = \overline{\rho}\left(\widetilde{u_i u_j} - \tilde{u}_i\tilde{u}_j\right) \quad i,j = 1,\ldots,dim, \tag{6}$$

$$q_i^{sgs} = c_P\overline{\rho}\left(\widetilde{Tu_i} - \widetilde{T}\tilde{u}_i\right) \quad i = 1,\ldots,dim. \tag{7}$$

The interactions of τ_{ij}^{sgs} and q_i^{sgs} with the resolved scales have to be modeled through a subgrid-scale model because they cannot be determined using only the resolved flow field $\boldsymbol{w}$.

2.1. The wall-adapted local eddy-viscosity model

In the previous section, it was seen that the smaller scales and their interaction with the resolved scales have to be modeled through the subgrid-scale term τ_{ij}^{sgs}. The tensor τ_{ij}^{sgs} can be modeled at different levels of complexity. The most common approach is based on the eddy-viscosity concept in which one assumes that the residual stress is proportional to the filtered rate of strain, which is defined as follows:

$$\tau_{ij}^{sgs} - \tau_{kk}^{sgs}\delta_{ij} = -2\overline{\rho}\nu_t\left(\widetilde{S}_{ij} - \frac{\delta_{ij}}{3}\widetilde{S}_{kk}\right) = -2\overline{\rho}\nu_t\widetilde{S}_{ij}^D, \tag{8}$$

In the wall-adapted local eddy-viscosity (WALE) proposed by Nicoud and Ducros [23], it is assumed that the eddy-viscosity ν_t is proportional to the square of the length scale of the filter and the filtered local rate of strain. Although the model was originally developed for incompressible flows, it can also be used for variable density flows by giving the formulation as follows [24]

$$\nu_t = (C\Delta)^2|\widetilde{S}|. \tag{9}$$

Here $|\widetilde{S}|$ is defined as

$$|\widetilde{S}| = \frac{\left[\widetilde{S}_{ij}^d\widetilde{S}_{ij}^d\right]^{3/2}}{\left[\widetilde{S}_{ij}\widetilde{S}_{ij}\right]^{5/2} + \left[\widetilde{S}_{ij}^d\widetilde{S}_{ij}^d\right]^{5/4}}, \tag{10}$$

where $\widetilde{S}_{ij}^{d}$ is given by

$$\widetilde{S}_{ij}^{d}=\frac{1}{2}\left(\tilde{g}_{ij}^{2}+\tilde{g}_{ji}^{2}\right)-\frac{\delta_{ij}}{3}\tilde{g}_{kk}^{2}. \tag{11}$$

with

$$\tilde{g}_{ij}^{2}=\frac{\partial\tilde{u}_i}{\partial x_k}\frac{\partial\tilde{u}_k}{\partial x_j}. \tag{12}$$

Note that in Eq. (9) Δ, i.e. the width of the grid filter, is an unknown function. Often the grid filter width is taken proportional to the smallest resolvable length scale of the discretization. In the present work, the definition of the grid filter function is given in Section 3.2, where the spectral difference method is discussed.

The WALE model is specifically designed to return the correct wall-asymptotic y^{+3}– variation of the subgrid-scale viscosity ν_t [23] and the constant model coefficient C can be adjusted so that the correct amount of subgrid dissipation is obtained. This model has the following properties:

- It is invariant to any coordinate translation or rotation.
- It is easily computed on any kind of computational grid.
- It is a function of both the strain and the rotation rates.
- It naturally goes to zero at the wall: neither damping function nor dynamic procedure is needed to reproduce the effect of the no-slip condition.

For the subgrid heat-flux vector q_i^{sgs}, if an eddy diffusivity model [31] is used, the following expression is obtained

$$q_i^{sgs}=C_p\bar{\rho}(\widetilde{Tu_i}-\widetilde{T}\tilde{u}_i)=-\bar{\rho}\frac{\nu_t C_p}{Pr_t}\frac{\partial\widetilde{T}}{\partial x_i}, \tag{13}$$

where the value of the turbulent Prandtl number Pr_t is set to 0.72 [32] and the eddy-viscosity is computed by Eq. (9).

3. Spectral difference method

In this section the spectral difference (SD) method is presented for the discretization of the filtered fluid dynamics equations for compressible flow. Consider a problem governed by a general system of conservation laws given by Eq. (3) and valid on a domain V for the filtered conservative variable vector $\boldsymbol{w}$ defined in Eq. (2). The domain is divided into N non-overlapping cells, with cell index i. In each cell, a 3D mapped coordinate system $\vec{\xi}=[\xi_1,\xi_2,\xi_3]^T$ is introduced, with the transformation to Cartesian coordinates defined by

$$\vec{x}_i=\begin{bmatrix}x_{1,i}\\ x_{2,i}\\ x_{3,i}\end{bmatrix}=\begin{bmatrix}x_{1,i}(\xi_1,\xi_2,\xi_3)\\ x_{2,i}(\xi_1,\xi_2,\xi_3)\\ x_{3,i}(\xi_1,\xi_2,\xi_3)\end{bmatrix}=\vec{x}_i(\vec{\xi}). \tag{14}$$

The Jacobian matrix for this transformation is denoted as $\overrightarrow{\mathrm{J}}_i$ and the Jacobian determinant as J_i. The fluxes projected in the mapped coordinate system $\vec{\mathbf{f}}^{\vec{\xi}}$ are related to the Cartesian flux components $\vec{\mathbf{f}}$ for the cell i by

$$\vec{\mathbf{f}}_i^{\vec{\xi}}=\begin{bmatrix}\mathbf{f}_i^{\vec{\xi}}\\ \mathbf{g}_i^{\vec{\xi}}\\ \mathbf{h}_i^{\vec{\xi}}\end{bmatrix}=J_i\overrightarrow{\mathrm{J}}_i^{-1}\begin{bmatrix}\mathbf{f}_i\\ \mathbf{g}_i\\ \mathbf{h}_i\end{bmatrix}=J_i\overrightarrow{\mathrm{J}}_i^{-1}\vec{\mathbf{f}}_i. \tag{15}$$

Therefore, the governing Eq. (3) can be written in the mapped coordinate system as

$$\frac{\partial(J_i\boldsymbol{w})}{\partial t}\equiv\frac{\partial\mathbf{w}_i^{\vec{\xi}}}{\partial t}=-\frac{\partial\mathbf{f}_i^{\vec{\xi}}}{\partial\xi_1}-\frac{\partial\mathbf{g}_i^{\vec{\xi}}}{\partial\xi_2}-\frac{\partial\mathbf{h}_i^{\vec{\xi}}}{\partial\xi_3}=-\vec{\nabla}^{\vec{\xi}}\cdot\vec{\mathbf{f}}_i^{\vec{\xi}}, \tag{16}$$

with $\boldsymbol{w}_i^{\vec{\xi}}\equiv J_i\boldsymbol{w}$ the conservative variables in the mapped coordinate system.

For a dim-dimensional $(p+1)$th order scheme, $N^s(p,dim)$ solution points with index j are introduced at positions $\vec{\xi}_j^s$, supporting a set of Lagrangian basis polynomials $L_j^s(\vec{\xi})$ of degree p. Based on these basis polynomials, one can approximate the solution in cell i with a pth order polynomial as follows

$$\boldsymbol{w}_i\approx\boldsymbol{W}_i(\vec{\xi})=\sum_{j=1}^{N^s}\boldsymbol{W}_{i,j}L_j^s\left(\vec{\xi}\right), \tag{17}$$

where the conservative variables at the solution points $\boldsymbol{W}_{i,j}$ denote the solution variables of the SD method. The evolution of these variables is governed by Eq. (16) evaluated at the solution points.

To estimate the divergence of the mapped fluxes $\vec{\nabla}^{\vec{\xi}} \cdot \vec{\mathbf{f}}_i^{\vec{\xi}}$ at the solution points, a set of N^f flux points with index l and at positions $\vec{\xi}_l^f$, supporting a polynomial of degree $p+1$, is introduced. The evolution of the mapped flux vector $\vec{\mathbf{f}}^{\vec{\xi}}$ in cell i is then approximated by a flux polynomial $\vec{\mathbf{F}}_i^{\vec{\xi}}$, which is obtained by reconstructing the solution variables at the flux points and evaluating the fluxes $\vec{\mathbf{F}}_{i,l}^{\vec{\xi}}$ at these points. To ensure a coupling between the cells, a number of flux points needs to lie at the faces or the corners of the cell. In order to maintain conservation at a cell level, the flux component normal to a face must be continuous between two neighbouring cells. Since the solution at a face is in general not continuous, this requires the introduction of Riemann solvers at those points. Two different approaches were discussed in Liu et al. [15] and in Wang et al. [16]. The first approach involves the definition of multi-dimensional Riemann solvers, while the second one uses multiple 1D Riemann solvers. One can find more information in the references mentioned above. The flux polynomial is then defined by

$$\vec{\mathbf{F}}_i^{\vec{\xi}}\left(\vec{\xi}\right) = \sum_{l=1}^{N^f} \vec{\mathbf{F}}_{i,l}^{\vec{\xi}} L_l^f\left(\vec{\xi}\right), \tag{18}$$

where the $L_l^f(\vec{\xi})$ are the Lagrangian basis polynomials associated with the flux points. Taking the divergence of the flux polynomial $\vec{\nabla}^{\vec{\xi}} \cdot \vec{\mathbf{F}}_i^{\vec{\xi}}$ in the solution points results in the following modified form of Eq. (16), describing the evolution of the conservative variables in the solution points:

$$\frac{d\boldsymbol{W}_{i,j}}{dt} = -\vec{\nabla} \cdot \vec{\mathbf{F}}_i\Big|_j = -\frac{1}{J_{i,j}} \vec{\nabla}^{\vec{\xi}} \cdot \vec{\mathbf{F}}_i^{\vec{\xi}}\Big|_j = \mathbf{R}_{i,j}, \tag{19}$$

where $\mathbf{R}_{i,j}$ is the SD residual associated to $\boldsymbol{W}_{i,j}$. This is a system of ordinary differential equations in time for the solution unknowns $\boldsymbol{W}_{i,j}$, which can be solved numerically using any classical method for such system.

In the present paper, only meshes with quadrilateral (2D) are considered. For such cells, different sets of flux points are used for the different components of the mapped flux vector. For the $\mathbf{f}^{\vec{\xi}}$-component, a set of flux points that supports a polynomial of degree $p+1$ in ξ_1 and of degree p in ξ_2 and ξ_3 is defined. These flux points are labeled the 'ξ_1-flux points' in the remainder of this paper. For the $\mathbf{g}^{\vec{\xi}}$- and $\mathbf{h}^{\vec{\xi}}$-components, 'ξ_2-' and 'ξ_3-flux points' are introduced analogously. The solution and flux points distributions for second- and third-order accurate quadrilateral SD cells are illustrated in Fig. 1 and 2 respectively.

In Van den Abeele et al. [33] it was shown that the distribution of the solution points has very little influence on the properties of the SD schemes, and in fact, for linear problems, the different distributions in the figures lead to identical results. It is easily seen that the rightmost distributions in the figures allow for a significant reduction in the solution reconstruction cost. Notice that for the third-order scheme, there is a free parameter α_3 for the flux point distributions. In general, the flux points are not uniquely defined for schemes with $p > 1$. The choice of the flux point distribution was shown to have an important influence on the stability and accuracy of the SD schemes in Van den Abeele et al. [33]. In Huynh et al.[18], it was proven that for quadrilateral and hexahedral cells, tensor product flux point distributions based on a 1D flux point distribution consisting of the end points and the Legendre–Gauss quadrature points, lead to stable schemes for arbitrary values of p. Consequently, in the present paper, symmetrical distributions with most solution points at flux points, shown in Fig. 1c and Fig. 2c, are used, where the flux point distribution is defined using Legendre–Gauss quadrature points. Finally, since only one component of the projected flux vectors is reconstructed in each set of flux points, a traditional 1D Riemann flux, like the Rusanov or the Roe flux [34,35], suffices for the treatment of the convective fluxes at face flux points in quadrilateral and hexahedral SD cells.

3.1. Spectral difference formulation for diffusive terms

Generally speaking, the different treatments used for the diffusive terms for the SD methods are all adopted from approaches developed for the DG method. An overview of them can be found in Arnold et al. [36]. However, in the present paper only one treatment is considered, i.e. the second approach of Bassi and Rebay (BR2) in [30].

For the evaluation of the diffusive flux vector $\vec{\mathbf{f}}_D\left(\boldsymbol{w}, \vec{\nabla}\boldsymbol{w}\right)$, the gradients of the conservative variables must be available at the flux points. Defining the vectors $\vec{\mathrm{J}}_i^{\vec{\xi}_1}$, $\vec{\mathrm{J}}_i^{\vec{\xi}_2}$ and $\vec{\mathrm{J}}_i^{\vec{\xi}_3}$ as

$$J_i \vec{\mathrm{J}}_i^{-1} = \left[\vec{\mathrm{J}}_i^{\vec{\xi}_1}, \quad \vec{\mathrm{J}}_i^{\vec{\xi}_2}, \quad \vec{\mathrm{J}}_i^{\vec{\xi}_3}\right]^T, \tag{20}$$

a gradient approximation polynomial $\vec{\boldsymbol{\Phi}}_i(\boldsymbol{\xi})$ is obtained by computing the values at the solution points as follows:

$$\vec{\nabla}\boldsymbol{w}|_{i,j} \approx \frac{1}{J_{i,j}} \left[\frac{\partial \breve{\boldsymbol{W}}_i \vec{\mathrm{J}}_i^{\vec{\xi}_1}}{\partial \xi_1} + \frac{\partial \breve{\boldsymbol{W}}_i \vec{\mathrm{J}}_i^{\vec{\xi}_2}}{\partial \xi_2} + \frac{\partial \breve{\boldsymbol{W}}_i \vec{\mathrm{J}}_i^{\vec{\xi}_3}}{\partial \xi_3}\right]_j = \vec{\boldsymbol{\Phi}}_{i,j}, \tag{21}$$

where $\breve{\boldsymbol{W}}_i$ is a polynomial of degree $p+1$, defined by its values at the flux points. At an internal flux point, this is just the value of the polynomial $\boldsymbol{W}_i$. At a face flux point, it is equal to an average value $\hat{\boldsymbol{W}}$, of the two available values $\boldsymbol{W}_L$ and $\boldsymbol{W}_R$, which is defined in Section 3.1.1. The gradients in cell i are then approximated by $\vec{\boldsymbol{\Phi}}_i(\vec{\xi})$, given by

$$\vec{\nabla}\boldsymbol{w}|_i \approx \vec{\boldsymbol{\Phi}}_i(\vec{\xi}) = \sum_{j=1}^{N^s} \vec{\boldsymbol{\Phi}}_{i,j} L_j^s(\vec{\xi}). \tag{22}$$

The diffusive flux vector is approximated as $\vec{\mathbf{f}}_D\left(\boldsymbol{W}_i, \vec{\boldsymbol{\Phi}}_i\right)$ in an internal flux point. At a face, two values, $\vec{\boldsymbol{\Phi}}_L$ and $\vec{\boldsymbol{\Phi}}_R$, of $\vec{\boldsymbol{\Phi}}$ are available. An averaged value $\hat{\vec{\boldsymbol{\Phi}}}$ is then used, also defined in Section 3.1.1. The normal component of the diffusive flux vector is thus evaluated as $\vec{\mathbf{f}}_D(0.5(\boldsymbol{W}_L + \boldsymbol{W}_R), \hat{\vec{\boldsymbol{\Phi}}}) \cdot \vec{1}_n$ in a face flux point, where the unit normal vector in the physical space $\vec{1}_n$ to a face points from the left (L) to the right (R) neighbouring cell.

3.1.1. The second approach of bassi and rebay

The BR2 definition of $\hat{\boldsymbol{W}}$ and $\hat{\vec{\boldsymbol{\Phi}}}$ is

$$\hat{\boldsymbol{W}} = \frac{\boldsymbol{W}_L + \boldsymbol{W}_R}{2}, \tag{23}$$

$$\hat{\vec{\boldsymbol{\Phi}}} = \frac{\vec{\nabla}\boldsymbol{W}_L + \vec{\nabla}\boldsymbol{W}_R}{2} + \varphi \frac{\vec{\boldsymbol{\Lambda}}_L + \vec{\boldsymbol{\Lambda}}_R}{2}, \tag{24}$$

where φ defines the amount of damping added to the gradients. It is always equal to one in the present work. The lifting operators $\vec{\boldsymbol{\Lambda}}_L$ and $\vec{\boldsymbol{\Lambda}}_R$, associated to a face, can be interpreted as corrections to the gradients of the solution polynomials in the neighbouring cells. These lifting operators are polynomials defined in the neighbouring cells by their values at the solution points:

$$\vec{\boldsymbol{\Lambda}}_{L(R),j} = \frac{1}{J_{L(R),j}} \left[\vec{\nabla}^{\vec{\xi}} \delta\boldsymbol{W}_{L(R)}\right]_j. \tag{25}$$

In this expression, $\delta\boldsymbol{W}$ is a polynomial of degree $p+1$, defined by its values at the flux points:

$$\delta\boldsymbol{W}_{L(R),l} = \begin{cases} (\boldsymbol{W}_{R,l} - \mathbf{W}_{L,l}) \left| \left(\vec{\vec{J}}^{-1}\right)^T \vec{1}_n^{\vec{\xi}} \right|_{L(R),l} \vec{1}_n & l \in \text{curr. face}, \\ 0 & \text{elsewise}, \end{cases} \tag{26}$$

where $\vec{1}_n^{\vec{\xi}}$ is the unit normal to the face in the mapped coordinate system. Unlike the Local Spectral Difference approach [37], the BR2 approach is fully compact, as only the immediate neighbors are required for the computation of the residuals in a cell. For $p = 0$, the gradients of the solution polynomials are always zero, $\vec{\nabla}\boldsymbol{W}_L = \vec{\nabla}\boldsymbol{W}_R = 0$, and the gradients at a face are approximated by the lifting operators alone as shown in Eq. (24).

3.2. Grid filter width for the subgrid-scale model

In Section 2.1 it is seen that in the WALE model the grid filter width Δ is used to compute the turbulent eddy-viscosity, i.e. Eq. (9). In general Δ is an unknown function and it has to be defined to have a closed model. Often, the grid filter width is taken proportional to the smallest resolvable length scale of the discretization and for a general cell with index i is usually approximated by

$$\Delta_i = \left(\prod_{k=1}^{dim} h_k\right)^{1/dim}, \tag{27}$$

where h_k is the size of the cell in the k-direction. However, at the face flux points, two values of Δ are available, labeled Δ_L and Δ_R. Consequently, an averaged value for the filter width is generally used,

$$\hat{\Delta}_i = \frac{\Delta_L + \Delta_R}{2}. \tag{28}$$

For classical finite volume (FV) methods, Eq. (28) uniquely defines the grid filter width because for these schemes the flux points always lie at the cell face. However, the same reasoning applied to a general SD method implies a natural formulation of the grid filter width based on the Jacobian determinant of the transformation defined by Eq. (14). However, since in the SD scheme each cell has interior solution points and a high-order polynomial approximation occurs in the cell, it is natural to choose the filter width depending on the order of the polynomial. Therefore, in the definition of the grid filter width, the order of the polynomial is taken into account through the division of the Jacobian determinant by the number of solution points, i.e. $N^s(p, dim)$. Consequently, for each cell with index i and each flux points with index l and positions $\vec{\xi}_l^f$, we propose the following new definition of the grid filter width

$$\Delta_{i,l} = \left[\frac{1}{N^s} det\left(\vec{\vec{J}}_i|_{\vec{\xi}_l^f}\right)\right]^{1/dim} = \left(\frac{J_{i,l}}{N^s}\right)^{1/dim}. \tag{29}$$

Eq. (29) uniquely defines the filter width for the internal flux points but for the face flux points two values of the Jacobians determinant are available, labeled $J_{L,l}$ and $J_{R,l}$. Consequently, an averaged value is again used, i.e.

$$\widehat{\Delta}_{i,l} = \left[\frac{det\left(\vec{\mathbf{J}}_L|_{\xi_l^{gf}}\right) + det\left(\vec{\mathbf{J}}_R|_{\xi_l^{gf}}\right)}{2N^s}\right]^{1/dim} = \left(\frac{J_{L,l}+J_{R,l}}{2N^s}\right)^{1/dim}. \tag{30}$$

Notice that with this approach, the cell filter width is not constant in one cell, but it varies because the Jacobian matrix is a function of the positions of the flux points. Moreover, for a given mesh, the number of solution points depends on the order of the SD scheme, so that the grid filter width vary by varying the order of the scheme. The proposed approach is valid for high-order schemes and it is consistent because the filter width is a function of the polynomial order through the number of solution points. In fact, the grid filter width decreases by increasing the polynomial order of the solution approximation.

4. Nonlinear lower–upper symmetric Gauss–Seidel method

Applying the SD method to a set of convection–diffusion equations results in a semi-discretization in space of the form (19). This expression can be discretized in time using any classical method. If an explicit time stepping algorithm, like a Runge–Kutta method, is used, then, because of the Courant–Friedrichs–Lewy stability condition for the convective terms, there is a maximum value, proportional to the cell size and inversely proportional to the maximum wave propagation velocity, for the associated time step Δt. This stability limit is often too restrictive, especially for high-order schemes and Navier–Stokes grids. Implicit time-integration schemes can be used to deal with these problems because they do not suffer from such a stability limit. In the present work, the second-order backward difference formula (BDF2) with variable time-step,

$$\frac{1+2\tau_n}{1+\tau_n}\boldsymbol{W}|_{t_{n+1}} - (1+\tau_n)\mathbf{W}|_{t_n} + \frac{\tau_n^2}{1+\tau_n}\boldsymbol{W}|_{t_{n-1}} = \Delta t_n \mathbf{R}|_{t_{n+1}}, \tag{31}$$

is employed to integrate, in time, the system of ODEs which arises from the spatial discretization of the Navier–Stokes equation using the SD method. In Eq. (31) n is the index of the time iteration, $\Delta t_n = t_{n+1} - t_n$ and $\tau_n = \frac{\Delta t_n}{\Delta t_{n-1}}$. The BDF2 method is A-stable [38]. So if BDF2 is employed to solve the linear Cauchy problem,

$$\begin{cases} \frac{dy}{dt} = \lambda y(t), \quad t > 0, \\ y(0) = y_0, \end{cases}$$

its absolute stability region is

$$A = \{z = \Delta t_n \lambda \in \mathbb{C}^-\}.$$

The aforementioned property is important to solve systems of stiff ODEs which often arise from the discretization of the fluid dynamic equations with a spatially high-order numerical scheme. In fact, the Fourier footprint of a spatial scheme, which gives an indication of the stiffness of the problem, grows with increasing the order of the method.

Expression (31) is a nonlinear algebraic system, which has to be solved each time iteration to find the solution at the next iteration t_{n+1} starting from the two previous time iterations t_n and t_{n-1}. Writing Eq. (31) for one cell (with the current cell denoted by the subscript cc, the neighbouring cells that contribute to its residual denoted by the subscript nb) omitting the solution point index j, and linearizing the residual about time iteration t_n, one obtains

$$\frac{\Delta \boldsymbol{W}_{cc}}{\Delta t_n} - \beta_1\left(\frac{\partial \mathbf{R}_{cc}}{\partial \mathbf{W}_{cc}}\bigg|_{t_n}\Delta \boldsymbol{W}_{cc} + \sum_{nb}\frac{\partial \mathbf{R}_{cc}}{\partial \boldsymbol{W}_{nb}}\bigg|_{t_n}\Delta \boldsymbol{W}_{nb}\right) = \beta_2\left(\mathbf{W}_{cc}|_{t_n} - \boldsymbol{W}_{cc}|_{t_{n-1}}\right) + \beta_1 \mathbf{R}_{cc}|_{t_n}, \tag{32}$$

with $\Delta \boldsymbol{W} = \boldsymbol{W}|_{t_{n+1}} - \boldsymbol{W}|_{t_n}$, $\beta_1 = \frac{1+\tau_n}{1+2\tau_n}$ and $\beta_2 = \frac{\tau_n^2}{\Delta t_n(1+2\tau_n)}$. Applying a (symmetric) Gauss–Seidel (SGS) algorithm to solve this linear algebraic system results in

$$\left[-\beta_1\frac{\partial \mathbf{R}_{cc}}{\partial \mathbf{W}_{cc}}\bigg|_{t_n} + \frac{\mathbf{I}}{\Delta t_n}\right]\Delta \mathbf{W}_{cc}^{m+1} = \beta_1\left(\mathbf{R}_{cc}|_{t_n} + \sum_{nb}\frac{\partial \mathbf{R}_{cc}}{\partial \boldsymbol{W}_{nb}}\bigg|_{t_n}\Delta \boldsymbol{W}_{nb}^{*}\right) + \beta_2\left(\boldsymbol{W}_{cc}|_{t_n} - \mathbf{W}_{cc}|_{t_{n-1}}\right), \tag{33}$$

where the approximation of the linear system solution at the iteration m is written as $\Delta \boldsymbol{W}^m$, and the superscript * signifies the latest available solution.

To avoid the storage of the off-diagonal block matrices $\frac{\partial \mathbf{R}_{cc}}{\partial \boldsymbol{W}_{nb}}|_{t_n}$, expression (33) is further manipulated to obtain

$$\left[-\beta_1\frac{\partial \mathbf{R}_{cc}}{\partial \mathbf{W}_{cc}}\bigg|_{t_n} + \frac{\mathbf{I}}{\Delta t_n}\right]\Delta \mathbf{W}_{cc}^{m+1} = \beta_1\left(\mathbf{R}_{cc}^{*} - \frac{\partial \mathbf{R}_{cc}}{\partial \boldsymbol{W}_{cc}}\bigg|_{t_n}\Delta \mathbf{W}_{cc}^{*}\right) + \beta_2\left(\boldsymbol{W}_{cc}|_{t_n} - \boldsymbol{W}_{cc}|_{t_{n-1}}\right), \tag{34}$$

where the following linearization about the time t_n is used,

$$\mathbf{R}_{cc}|_{t_n} + \sum_{nb} \left.\frac{\partial \mathbf{R}_{cc}}{\partial \mathbf{W}_{nb}}\right|_{t_n} \Delta \mathbf{W}^*_{nb} \approx \mathbf{R}_{cc}\left(\mathbf{W}^*_{cc}, \{\mathbf{W}^*_{nb}\}\right) - \left.\frac{\partial \mathbf{R}_{cc}}{\partial \mathbf{W}_{cc}}\right|_{t_n} \Delta \mathbf{W}^*_{cc}. \tag{35}$$

Defining $\delta \mathbf{W}^{m+1}_{cc} = \Delta \mathbf{W}^{m+1}_{cc} - \Delta \mathbf{W}^*_{cc} = \mathbf{W}^{m+1}_{cc} - \mathbf{W}^*_{cc}$, the final expression of the nonlinear LU-SGS algorithm for the BDF2 scheme with variable time step is then:

$$\left[-\beta_1 \left.\frac{\partial \mathbf{R}_{cc}}{\partial \mathbf{W}_{cc}}\right|_{t_n} + \frac{\mathbf{I}}{\Delta t_n}\right] \delta \mathbf{W}^{m+1}_{cc} = \beta_1 \mathbf{R}^*_{cc} + \beta_2 (\mathbf{W}_{cc}|_{t_n} - \mathbf{W}_{cc}|_{t_{n-1}}) - \frac{\Delta \mathbf{W}^*_{cc}}{\Delta t_n}. \tag{36}$$

The inverse of the small Jacobian matrices in the left hand side of this expression can be computed using a LU decomposition at the beginning of each time iteration, which makes the solution of the small linear algebraic systems much more efficient during subsequent SGS sweeps. Note, that this LU-SGS algorithm acts directly on the nonlinear algebraic system to be solved, which is the right hand side of expression (36). For this reason the algorithm is called nonlinear LU-SGS algorithm.

Since only the diagonal block Jacobians is stored, the total number of variables N needed for these Jacobians, in 3D, is

$$N[(p+1)^3 \times \text{ physical variables}]^2. \tag{37}$$

From expression (37), it is clear that the nonlinear LU-SGS method requires significantly less memory than the classical method that use the full Jacobian matrix (for instance the GMRES algorithm [39,40]), but the required amount still increases with p to the power six. In this study, the Jacobian matrices are numerically obtained using the following numerical differencing

$$\frac{\partial \mathbf{R}_{cc}}{\partial \mathbf{W}_{cc}} = \frac{\mathbf{R}_{cc}(\mathbf{W}_{cc} + \delta \mathbf{W}_{cc}, \mathbf{W}_{nb}) - \mathbf{R}_{cc}(\mathbf{W}_{cc}, \mathbf{W}_{nb})}{\delta \mathbf{W}_{cc}}, \tag{38}$$

where $\delta \mathbf{W}_{cc}$ is a small value computed as

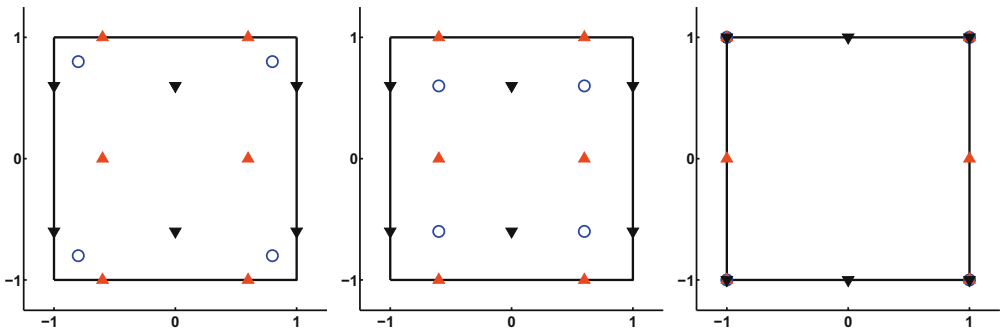
$$\delta \mathbf{W}_{cc} = \|\mathbf{W}_{cc} - \mathbf{W}_{ref}\| \varepsilon. \tag{39}$$

$\mathbf{W}_{ref}$ and ε are respectively the reference conservative variables and a small constant set equal to something larger than the square root of machine ε_{mach} [40]. In the present work, with 64-bit double precision machine, ε was set to 10^{-5}.

5. Numerical results

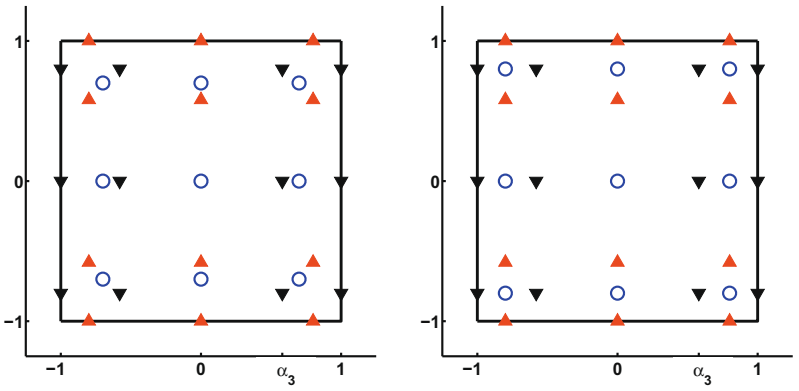
All test cases considered here are governed by the 2D fluid dynamic equations. In order to study the quality of the new coupling approach, the results obtained with and without the subgrid-scale model are compared with the reference solutions. All quantities are non-dimensionalized using the flow quantities at the free-stream/inflow location, which is indicated by the subscript ∞.

As previously mentioned, one property of a good LES model is that its use in a laminar or low Reynolds number flow results in a solution which is very close to the solution obtained by solving the Navier–Stokes equations [25]. Therefore, the sensitivity of the method is first investigated by solving the flow 2D flow around a NACA0012 airfoil at $Re = 5 \times 10^5, M = 0.4$ and zero angle of attack. For this test case, a DNS reported in Sandberg et al. is available [41]. Afterwards, the new coupling between the SD method and LES is tested by solving the 2D "turbulent" flow around a square cylinder at $Re = 10^4$ and $Re = 2.2 \times 10^4$. The obtained results are compared with the DNS results of Wissink [43] and the LES and experimental results reported in Bouris et al. [44].



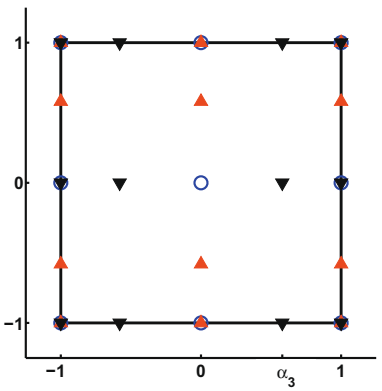
(a) General. (b) Locally 1D. (c) Solution points at flux points.

Fig. 1. Second-order quadrilateral SD cells. Solution points (○) and ξ_1- (▼) and ξ_2-flux points (▲).

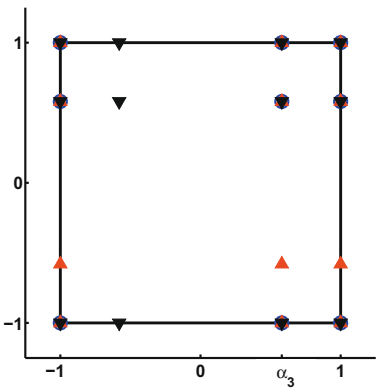


(a) General.

(b) Locally 1D.



(c) Most solution points at flux points, symmetrical.



(d) All solution points at flux points, asymmetrical.

Fig. 2. Third-order quadrilateral SD cells. Solution points (○) and ξ_1- (▼) and ξ_2-flux points (▲).

The nonlinear system (36) is solved with multiple cell-wise symmetric forward and backward sweeps with a prescribed tolerance of 10^{-6} on the change of the L_2 norm of the solution variation $\delta \boldsymbol{W}_{cc}^{(m+1)}$ and/or a maximum number of 100 symmetric Gauss–Seidel sweeps. During the calculations, the maximum number of symmetric Gauss–Seidel sweeps is never require. However, during the initial time steps (depending on the test cases and on the initial solution) the number of inner LU-SGS sweeps is between forty and forty-five, for the present test cases. Afterwards, the number decreases and it reaches a values which is between eight and twelve (depending again on the test cases). The meshes were created using Gmsh [45] which allows second-order ($p = 2$) polynomial approximation of the boundary elements. The computations were done on a machine with eight Dual Core AMD Opteron™processors with a clock-speed of 2412 MHz, using the COOLFluiD collaborative simulation environment [46]. Twenty-four gigabytes of RAM were available.

5.1. 2D flow around a NACA0012 airfoil at low Reynolds number

The compressible laminar flow simulation over a symmetric NACA0012 airfoil is conducted at $\alpha = 0°$ and $M = 0.4$. The characteristic Reynolds number based on the chord c and the module of free-stream velocity $|\vec{u}_\infty|$ is specified as $Re = 5 \times 10^5$. In Fig. 3 the configuration of the test case is illustrated. The airfoil is placed on $\chi_2 = 0$. At the left-hand-side boundary (the inflow) the flow is prescribed to be uniform and the same boundary conditions are applied to the upper and lower boundaries. At the right-hand-side boundary (the outflow), far enough from the profile, only the pressure is prescribed.

The test case is solved using a fourth-order ($p = 3$) SD scheme and a mesh with 19874 quadrilateral cells and quadratic ($p = 2$) boundary elements. The maximum aspect ratio AR of the first layer of the cells close to the airfoil is large. The time-step used for the computation starts from 0.00001 and increases linearly till 0.00125. This time-step allows ≈ 300 time samples per period of the vortex shedding.

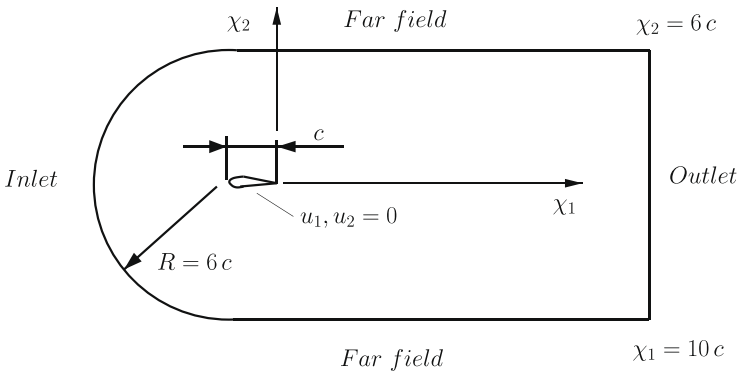


Fig. 3. Configuration of the 2D NACA0012 test case.

The instantaneous contour lines of the entropy obtained using the subgrid-scale model, are shown in Fig. 4 to give an impression of the flow field around the airfoil. The flow around the body is almost symmetric on the top and the bottom sides of the airfoil and an unsteady wake develops downstream of the trailing edge. In Fig. 5 the fast Fourier transform (FFT) of the lift coefficient c_l at a statistically steady state solution with the subgrid-scale model is shown. Notice that the c_l for the airfoil is computed as the ratio between the instantaneous global force L in the direction perpendicular to the free-stream velocity $\vec{u}_\infty$ and the dynamic pressure $\frac{1}{2}\rho|\vec{u}_\infty|^2$ times the chord c, i.e. $c_l = \frac{L}{\frac{1}{2}\rho|\vec{u}_\infty|^2 c}$. In Fig. 5 one distinct peak can be observed in the lift coefficient spectrum, which corresponds to the frequency f of the unsteady wake. The non-dimensional values of f, i.e. the reduced frequencies

$$f_r = \frac{f\frac{c}{2}}{a_\infty(1-M^2)}, \tag{40}$$

where a_∞ is the free-stream speed of sound, are listed in Table 1 together with the 2D DNS result of Sandberg et al. [41], where a fourth-order numerical scheme without upwinding, artificial dissipation or explicit filtering [42] is used. For

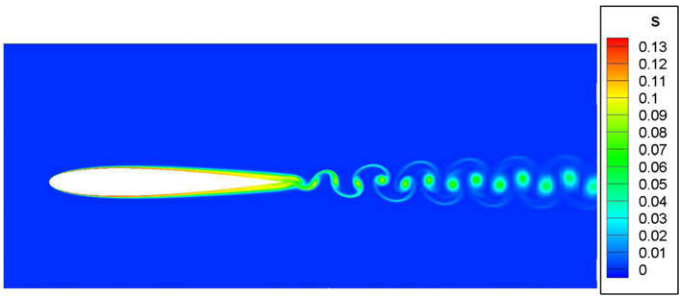


Fig. 4. Instantaneous entropy contours of the 2D NACA0012 airfoil at $Re = 5 \times 10^5$, $M = 0.4$, α=0° and subgrid-scale model.

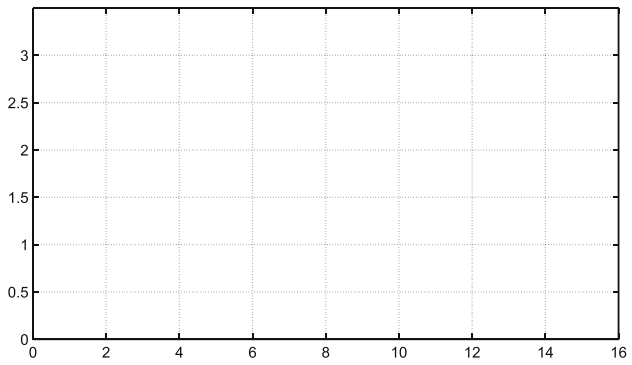


Table 1
Reduced frequency of the 2D NACA0012 airfoil at $Re = 5 \times 10^5$, $M_c = 0.4$ and $\alpha = 0°$.

Reduced frequency f_r		
DNS, [41]	SD no model	SD–LES
6.580	6.650	6.649

DNS, a mesh with 2243×692 non-equidistantly spaced points in the tangential and in the normal direction with respect to the airfoil surface was employed. From Table 1 it can be seen that the reduced frequency is in good agreement with the DNS value for both computations. In fact, the error with and without subgrid-scale model is respectively 1.04% and 1.06%. This is a good achievement since the grid employed is very coarse compared to the grid used for a DNS computation. Moreover, as was expected, the subgrid-scale model does not affect the frequency of the vortex shedding when the laminar flow is well resolved. To conclude this study, in Fig. 6 the time-averaged center-line velocity component $\langle \tilde{u}_1 \rangle$ is compared with the 2D DNS solution. Fig. 6 shows that after an initial peak, $\langle \tilde{u}_1 \rangle$ becomes negative before increasing to positive values further downstream, i.e. a reverse flow region is present. This reverse flow region is caused by flow oscillating around the trailing-edge corner at the wake frequency. From this figure, it is seen that the solutions obtained with and without subgrid-scale model are almost indistinguishable. Therefore, the new coupling between the SD method and LES replicates the laminar flows obtained by solving the pure Navier–Stokes equations. Moreover, the comparisons show a good agreement between the present numerical results and the 2D DNS solution of Sandberg et al. [41].

5.2. 2D flow around a square cylinder at $Re = 10^4$

The purpose of this test case is to compare the quality of the present numerical approach with a reference 2D DNS solution presented by Wissink in [43]. In Fig. 7, the configuration of the test case is illustrated. The cylinder is placed on $\chi_2 = 0$. At the left-hand-side boundary (the inflow) the flow is prescribed to be uniform. The same conditions are applied to the upper boundary and to the lower boundary. At the right-hand-side boundary, far enough from the cylinder, only the pressure is prescribed. The Mach number is set to 0.05, so that the flow is almost incompressible. Notice that, although the Mach number is rather small, no specific difficulties of convergence and/or accuracy have been observed. An investigation of the effects

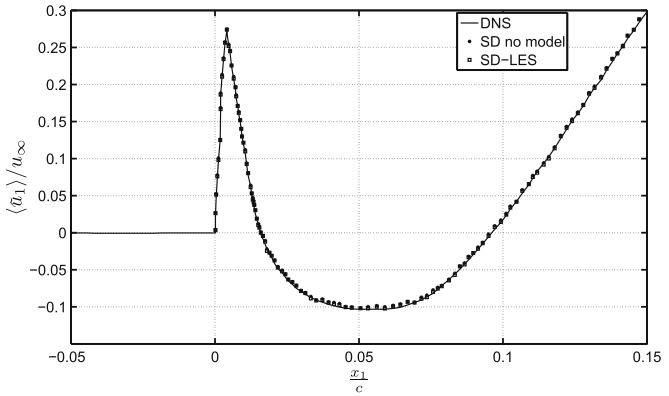


Fig. 6. Time averaged center-line ($x_2 = 0$) velocity component $\frac{\langle \tilde{u}_1 \rangle}{|\tilde{u}_\infty|}$ of the 2D NACA0012 airfoil at $Re = 5 \times 10^5$, $M = 0.4$ and α=0°. Comparison with the DNS solution of Sandberg et al. [41].

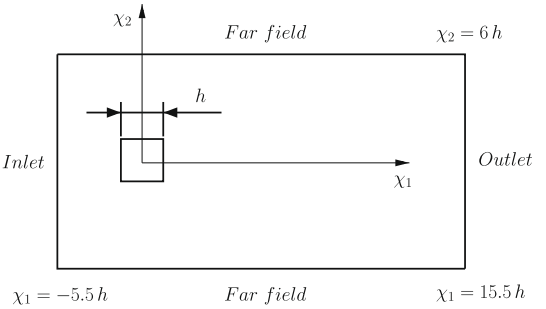
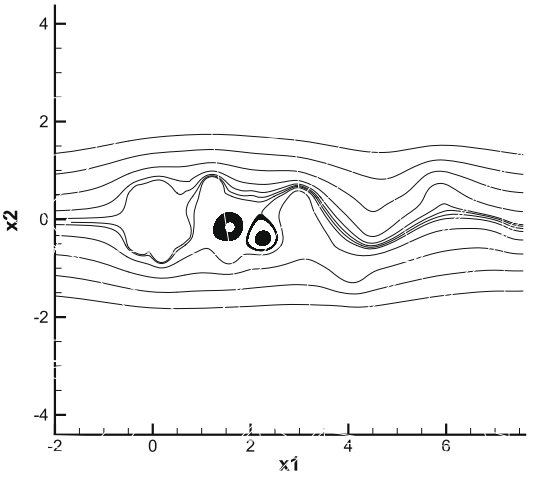


Fig. 7. Configuration of the 2D square cylinder test case.

of the Mach number on the convergence of the implicit LU-SGS was carried out in [47], where the algorithm has been shown accurate at low Mach. The Reynolds number based on the module of free-stream velocity $|\vec{u}_\infty|$ and height of the square cylinder h is 10^4. The DNS of Wissink [43] was performed on a structured mesh with 400×400 grid points using a spatial discretization that consists of a combination of a seventh-order upwind-biased method for the convective terms and an eighth-order central method for the diffusive terms. In the present work, the test case is solved using third-order ($p = 2$) SD schemes with and without the subgrid-scale model. A mesh with 12,622 hexahedral elements with a maximum aspect ratio (AR_{max}) of 33 close to the square cylinder is employed. The first cell size normal to the wall in turbulent units is $y^+ = 3.2$. Notice that the total number of degrees of freedom (DOFs) is 113598 which is almost 71% of the DOFs used in [43]. The time-step used for the computation starts from 0.00001 and increases linearly up to 0.0025. At the beginning of the simulation (starting with uniform flow field), rather small vortices of opposite sign are shed in parallel. However, this symmetric flow is not a stable solution at $Re = 10^4$. After the breaking of the symmetry, the flow behind the cylinder behaves very chaotically and is fully turbulent. A lot of very small vortices and filaments appear just behind the cylinder as shown in Fig. 8. Further away from the cylinder these small vortices merge to form larger vortical structures. In Fig. 9, the number of inner LU-SGS sweeps as a function of the time iteration is shown, for the calculation with the subgrid-scale model. As can be clearly seen, the maximum number of symmetric Gauss–Seidel sweeps is never used. However, during the initial time steps the number of inner LU-SGS sweeps increases up to forty-six. Afterwards, it decreases to an averaged value of ten. In Fig. 9 the evolution of the LU-SGS sweeps is shown up to hundred and fifty time iterations. Note, that the transitional behavior in Fig. 9 depends on the initial solution. In the present work, an uniform velocity field equals to the inlet boundary condition is imposed at the beginning of the computation.



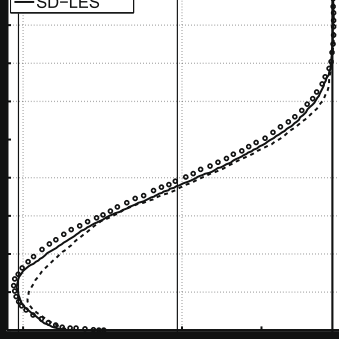
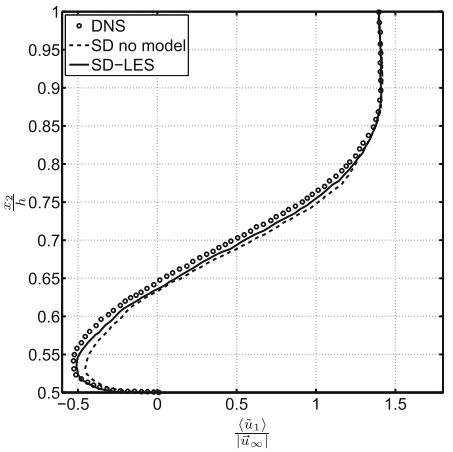
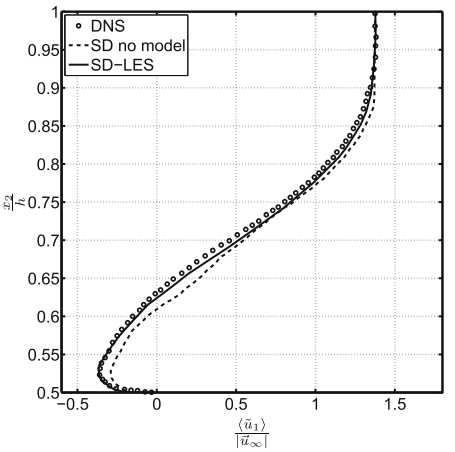


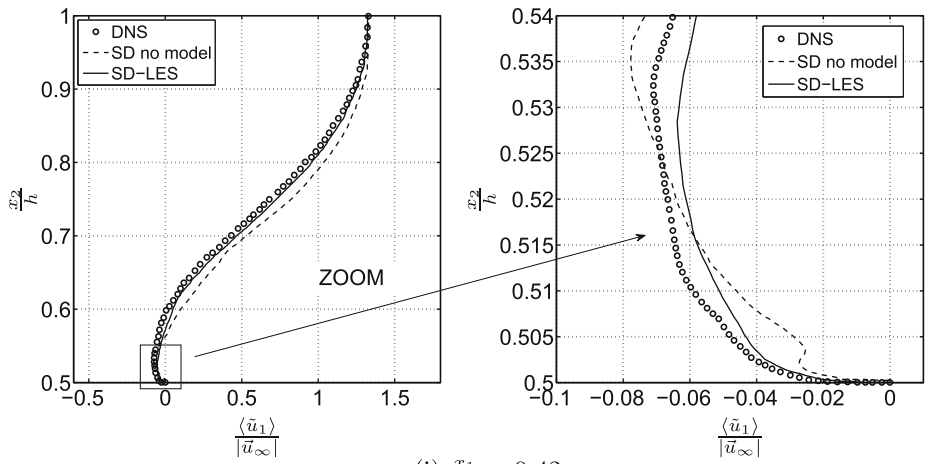
Fig. 10. Time averaged velocity profiles $\frac{\langle u_1 \rangle}{|U_\infty|}$ at various stations along the upper side of the square cylinder at $Re = 10^4$. Comparison
of Wissink [40].



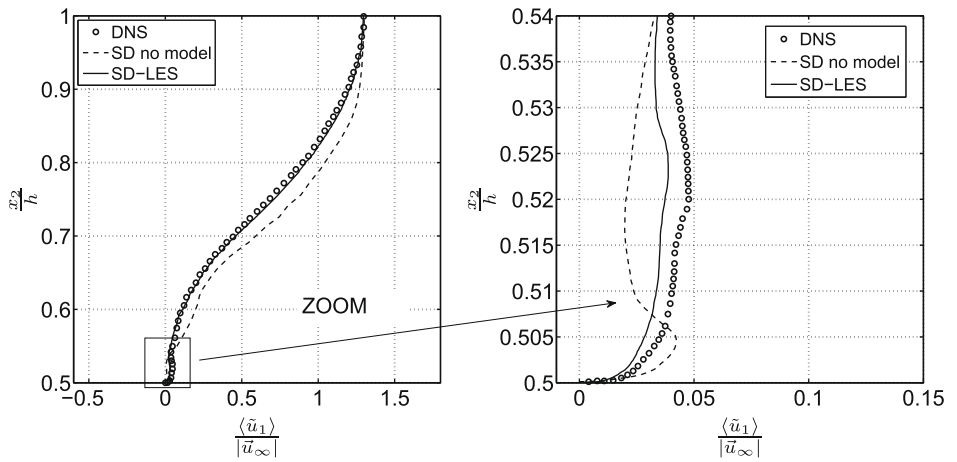
(g) $\frac{x_1}{h} = 0.21$.



(h) $\frac{x_1}{h} = 0.32$.



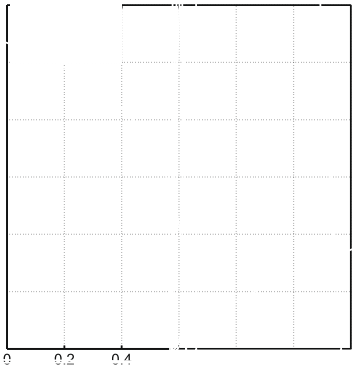
(i) $\frac{x_1}{h} = 0.42$.



(j) $\frac{x_1}{h} = 0.49$.

Fig. 10 (*continued*)

ted. The 2D DNS solution is also plotted in this figure for comparison. It can be observed that the SD method, in combination with LES works very well and improves the accuracy of the results obtained with the SD method without the subgrid-scale model. The improvement is clearly visible at trailing edge of the cylinder (Fig. 10i and j), where the LES solution captures



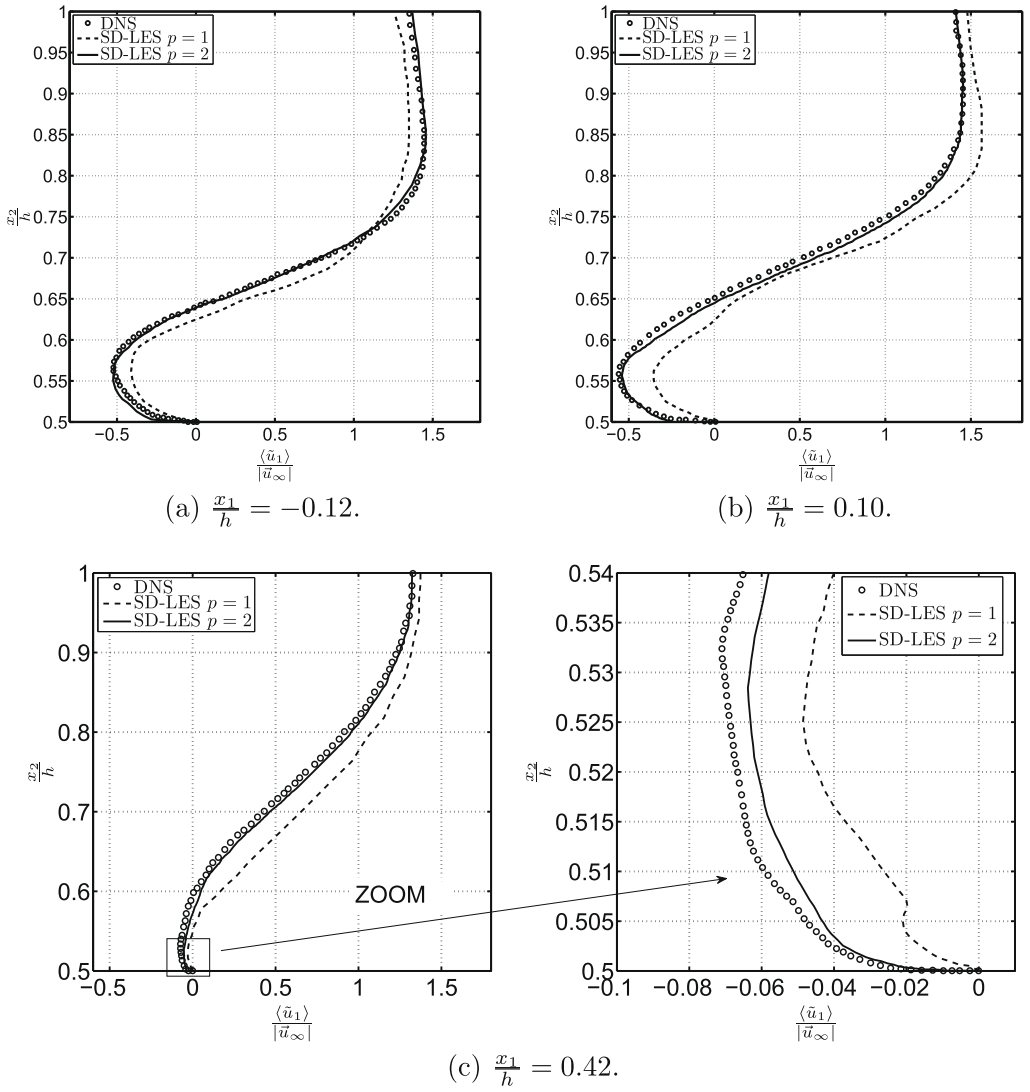
quite well the variation of the velocity profiles close to the wall, whereas the solution obtained without subgrid-scale model differs clearly from the DNS solution. The $\frac{\langle \bar{u}_1 \rangle}{|\bar{u}_\infty|}$ profiles plotted in Fig. 11 have been gathered at various stations x_1 = constant behind the cylinder. Fig. 11 shows once more that the SD–LES simulation works well and improves the accuracy of the results obtained solving the pure Navier–Stokes equations. In fact, the velocity profiles obtained with the subgrid-scale model are close to those computed with the DNS.

In order to study the effect of the polynomial order on the LES solution, we also have performed a simulation with a second-order spatial discretization ($p = 1$) on the same grid. Thus, the number of DOFs was reduced by a factor of 2.25. In Figs. 12 and 13, the results obtained at three locations on the side of the square cylinder and at two locations downstream in the wake are shown, respectively. One can compare the solutions and see that the solution obtained with the second-order SD scheme is much less accurate than the one obtained with the third-order SD scheme. Although, the second-order accurate solution is able to capture the main features of the flow, the velocity profiles show an large deviation from the DNS solution both on the side of the square cylinder and in the wake.

To conclude the study of this test case, a simple estimation of the Reynolds number scaling of the computational cost of the SD–LES is given. From literature (see [48]), the computational cost of the DNS for isotropic turbulence, i.e. the total number of modes to resolve the entire energy spectrum in 3D, is proportional to a power of the Reynolds number

$$N^{mod,3D} \sim Re_L^{9/4},$$

where Re_L is the Reynolds number based on the length scale L characterizing the large eddies. For the present test cases L can be chosen equal to the height of the square cylinder, i.e. h. For a classical finite volume scheme, $N^{mod,\ 3D}$ corresponds to the number of DOFs. Using the same relation presented above, the number of DOFs needed to perform a DNS with SD can be roughly estimated as



(a) $\frac{x_1}{h} = -0.12$. (b) $\frac{x_1}{h} = 0.10$.

(c) $\frac{x_1}{h} = 0.42$.

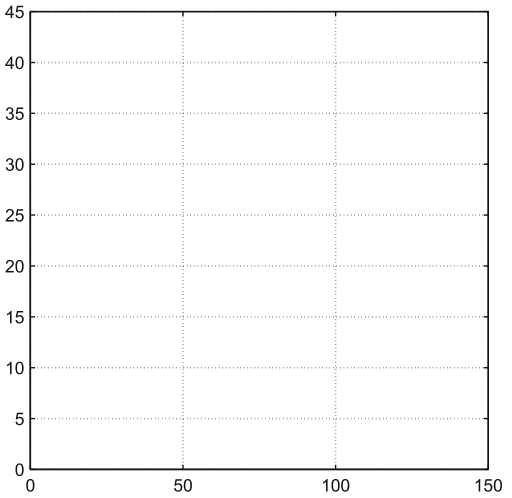
Fig. 12. Time averaged velocity profiles $\frac{\langle \bar{u}_1 \rangle}{|\vec{u}_\infty|}$ at two selected locations along the upper side of the square cylinder at $Re = 10^4$, for second- ($p = 1$) and third-order ($p = 2$) SD–LES scheme. Comparison with the 2D DNS solution of Wissink [43].

$$N_{SD-DNS}^{cells,3D} \sim \frac{Re_L^{9/4}}{N^s(p,dim)} \quad dim = 3, \quad N_{SD-DNS}^{cells,2D} \sim \frac{Re_L^{6/4}}{N^s(p,dim)} \quad dim = 2.$$

Consequently, for the flow past a square cylinder at $Re = 10^4$, in order to perform a 2D DNS with a third-order SD scheme, 3.7×10^4 cells are needed. Hence, the averaged CPU time per iteration, to perform the 2D DNS simulation can be computed as

$$CPU_{SD-DNS}^{iter} = \frac{N_{SD-DNS}^{cells,2D}}{N_{SD-LES}^{cells,2D}} \frac{CPU_{SD-no\,model}^{iter}}{CPU_{SD-LES}^{iter}} \approx 2.5$$

Therefore, from our estimate a SD–LES simulation is almost 2.5 times cheaper than a SD-DNS simulation. The reasoning that we have presented gives just a rough estimation of the CPU time needed for the DNS simulation. However, LES computed with a compact high-order method is a new topic and more investigations on the properties of the SD for LESs will be done in future works. In fact, to estimate in more detail the cost of the SD-DNS and SD–LES, an analysis of the number of Fourier



modes in each direction for adequate resolution of the isotropic turbulence should be performed, as it is shown in [48]. Moreover, this analysis should be carried out considering the dispersion and the dissipative properties of the SD scheme.

5.3. 2D flow around a square cylinder at $Re = 2.2 \times 10^4$

The purpose of the next test case is to evaluate the quality of the proposed numerical method by comparing the mean flow field and the turbulent kinetic energy results with some reference solution. In the literature 2D and 3D large eddy simulations of the flow past a square cylinder have been performed, for instance see Murakami et al. [53] and Rodi [52]. Moreover, Breuer and Pourquie [54] and Bouris et al. [44] also performed 2D and 3D LES computations and compared the results with the experimental measurements of Lyn [49] and Durao et al. [50]. It was demonstrated that the characteristic of this type of flow is its quasi-two-dimensional character and the presence of periodic vortex shedding from the front corners of the square rod which introduces a low-frequency variation of the velocity field behind the rod in addition to the high-frequency turbulence fluctuations. It has been stated in the past [44,54] that 2D LES calculations are clearly inferior to 3D ones since certain important features of 3D turbulence are not resolved. The three dimensionality of turbulence cannot be questioned, however, in the present paper, we want to show that the importance of detailed simulation of the quasi-two-dimensional mechanisms can be achieve performing a 2D LES with the new combination of SD and subgrid-scale model.

The same domain, boundary conditions and mesh used for the previous test case are employed. The Mach number is set to 0.05 so that the flow is again almost incompressible. The Reynolds number based on the module of the free-stream velocity $|\vec{u}_\infty|$ and height of the square cylinder h is 2.2×10^4. The flow is computed using third-order ($p = 2$) SD schemes with and without the subgrid-scale model. The time-step used for the computation starts from 0.00001 and increases linearly up to 0.0020. Fig. 14 shows the number of inner LU-SGS sweeps as a function of the time iteration. As can be clearly seen, the maximum number of symmetric Gauss–Seidel sweeps is never reached. However, during the initial steps the number of inner LU-SGS sweeps increases up to forty-five. Afterwards, the number of sweeps decreases to nine and it oscillates around this value. For this reason, the evolution of the LU-SGS sweeps in Fig. 14 is shown up to hundred and fifty time iterations. Note once more, that the transitional behavior in Fig. 14 depends on the initial solution. A uniform velocity field equals to the inlet boundary condition is imposed at the beginning of the computation.

In the present test case, the 2D LES calculation with a second-order finite volume scheme and the Smagorisky-Lilly model of Bouris et al. [44] will be used as reference 2D LES solution. In the latter work, a mesh with 350×300 points is employed. Therefore, the total number of DOFs of the present SD calculation is practically the same as the number of DOFs used in [44].

As for the previous calculation, inlet conditions are based on a uniform flow with non-fluctuating velocity profiles. Table 2 shows the predictions of the dominant vortex shedding frequency in non-dimensional form reported in [44], as well as the results from the present simulations. In this table, the abbreviation $(k-\varepsilon)^*$, RSE1 and RSE2 stand, respectively, for $k-\varepsilon$ model with 2 layers, Reynolds stress equation with wall function and Reynolds stress equation with two layers [51,52]. The re-

Table 2
Numerical calculation of the Strouhal number and the mean drag coefficient from various turbulence models, the present calculations and experimental measurements, for the flow over a square cylinder at $Re = 2.2 \times 10^4$.

	$(k-\varepsilon)^*$ [51]	RSE1 [51]	RSE2 [51]	2D LES [44]	Present no model	Present LES-WALE	Exps. [49,50]
St	0.124	0.136	0.159	0.134	0.121	0.133	0.132 ± 0.004 0.139
$\langle c_d \rangle$	1.179	2.15	2.43	2.18	1.98	2.21	2.05 − 2.23

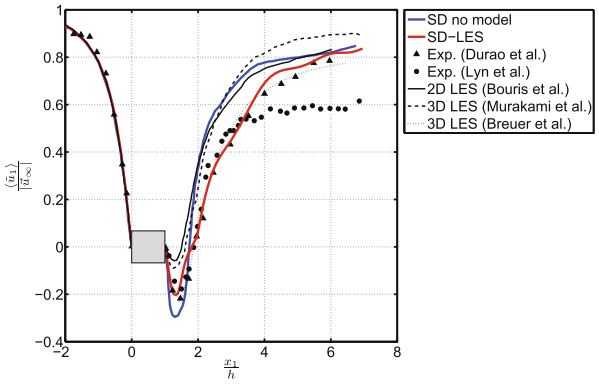


Fig. 15. Time averaged velocity $\frac{\langle \bar{u}_1 \rangle}{|\vec{u}_\infty|}$ along the center-line behind the square cylinder. Comparison with the numerical solution reported in Bouris et al. [44].

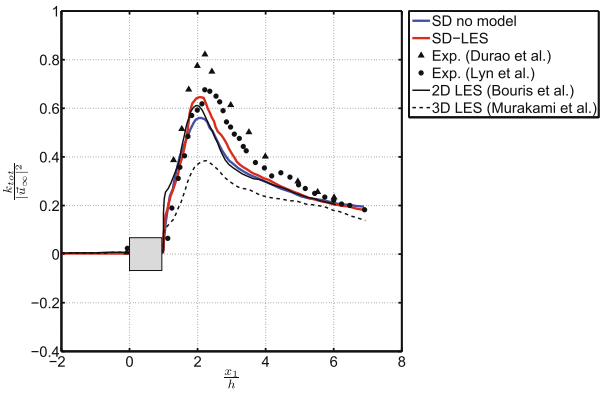


Fig. 16. Total fluctuation energy $\frac{k_{tot}}{|\vec{u}_\infty|^2}$ of the velocity along the center-line behind the square cylinder. Comparison with the numerical solution reported in Bouris et al. [44].

sults of the SD–LES are in good agreement with the experimental measurements while the solution without the subgrid-scale model underestimates both the Strouhal number and the time averaged drag coefficient $\langle c_d \rangle$.

Time averaged results are obtained integrating the data over approximately 20 shedding cycles and the mean center-line velocity is presented in Fig. 15. Although the SD scheme without subgrid-scale model overpredicts the value of the reverse velocity, it already gives a solution which has somewhat the same accuracy as the 2D LES of Bouris et al. [44]. Moreover, it can be clearly seen that the modeling of the subgrid-scale stress tensor improves the accuracy of the results. In fact, a good agreement between the predicted results and experimental data of Durao et al. [49] throughout the comparison domain shows that the quality of a high-order SD method increases when it is coupled with large eddy simulation. In the other parts of the comparison domain the method without subgrid-scale model is in a good agreement with LES of Bouris et al. [44]. Note that, the SD–LES method captures the peak of the mean stream-wise velocity considerably better than the others.

Fig. 16 shows the resolved total turbulent kinetic energy. One can compare the results with the experimental data and see that the SD method with the subgrid-scale model undershoots the peak experimental value somewhat less than the 2D LES of Bouris et al. [44] and predicts the results well elsewhere. As is observed, the present approach without subgrid-scale model overshoots the peak experimental value less than does the 2D LES of Bouris et al. [44], whereas throughout most of the domain both approaches are on the top of each other. Overall, our results indicate that although the SD method without subgrid-scale model is already able to predict all the features of the flow, the SD scheme combined with subgrid-scale model is more accurate than the one without subgrid-scale model, for the stream-wise mean velocity and the total kinetic energy.

6. Conclusions

In this work we have presented the high-order spectral difference method in combination with large eddy simulation. The approach used a third-order scheme for the spectral difference method and the WALE model for large eddy simulation. We have proposed a new procedure to calculate the filter width based on the Jacobian determinant of the transformation of the mapped coordinate system and on the order of the spectral difference method. With this approach, the cell filter width is not constant in one cell, but it varies because the Jacobian matrix is a function of the positions of the flux points. Moreover, for a given mesh, the distribution of the flux points depends on the order of the SD scheme, so that the Jacobian and hence the grid filter width varies when varying the order of the scheme. We have compared the results for the laminar flow around a NACA0012 airfoil at $Re = 5 \times 10^5$ and $M = 0.4$ and for the turbulent flow over a square cylinder with two different Reynolds numbers of 10^4 and 2.2×10^4, respectively. The first test case was employed to show that the use of numerical approach in a laminar or low Reynolds number flows results in a solution which is very close to the solution obtained by solving the Navier–Stokes equations. For the two turbulent cases, for a given number of degrees of freedom, the SD–LES method gives better results than the 2D simulation without the subgrid-scale model. Moreover, for the square cylinder at $Re = 10^4$ the present LES results are in good agreement with those of the DNS solution. For the square cylinder at $Re = 2.2 \times 10^4$, the solution obtained with the combination of the SD method and the subgrid-scale model is in good agreement with reference numerical solutions and the experiment reported in the literature. Overall, the SD scheme in combination with large eddy simulation improves the quality of the results for turbulent flows.

Acknowledgment

This research was funded by IWT under Project SBO 050163. This funding is gratefully acknowledged.

References

[1] B. Cockburn, C.-W. Shu, TVB Runge–Kutta local projection discontinuous Galerkin finite element method for conservation laws II: general framework, Mathematics of Computation 52 (1989) 411–435.
[2] B. Cockburn, S.-Y. Lin, C.-W. Shu, TVB Runge–Kutta local projection discontinuous Galerkin finite element method for conservation laws III: one-dimensional systems, Journal of Computational Physics 84 (1989) 90–113.
[3] B. Cockburn, S. Hou, C.-W. Shu, TVB Runge–Kutta local projection discontinuous Galerkin finite element method for conservation laws IV: the multi-dimensional case, Mathematics of Computation 54 (1990) 545–581.
[4] B. Cockburn, C.-W. Shu, The Runge–Kutta discontinuous Garlerkin method for conservation laws V: multi-dimensional systems, Journal of Computational Physics 141 (1998) 199–224.
[5] B. Cockburn, C.-W. Shu, The local discontinuous Galerkin method for time-dependent convection–diffusion systems, SIAM Journal of Numerical Analysis 35 (6) (1998) 2440–2463.
[6] Z.J. Wang, Spectral (finite) volume method for conservation laws on unstructured grids: basic formulation, Journal of Computational Physics 178 (2002) 210–251.
[7] Z.J. Wang, Y. Liu, Spectral (finite) volume method for conservation laws on unstructured grids II: extension to two-dimensional scalar equation, Journal of Computational Physics 179 (2002) 665–697.
[8] Z.J. Wang, Y. Liu, Spectral (finite) volume method for conservation laws on unstructured grids III: one-dimensional systems and partition optimization, Journal of Scientific Computing 20 (2004) 137–157.
[9] Z.J. Wang, L. Zhang, Y. Liu, Spectral (finite) volume method for conservation laws on unstructured grids IV: extension to two-dimensional Euler equations, Journal of Computational Physics 194 (2) (2004) 716–741.
[10] Y. Liu, M. Vinokur, Z.J. Wang, Spectral (finite) volume method for conservation laws on unstructured grids V: extension to three-dimensional systems, Journal of Computational Physics 212 (2006) 454–472.
[11] Y. Sun, Z.J. Wang, Y. Liu, Spectral (finite) volume method for conservation laws on unstructured grids VI: extension to viscous flow, Journal of Computational Physics 215 (2006) 41–58.
[12] Z.J. Wang, Y. Liu, Extension of the spectral volume method to high-order boundary representation, Journal of Computational Physics 211 (2006) 154–178.
[13] K. Van den Abeele, T. Broeckhoven, C. Lacor, Dispersion and dissipation properties of the 1D spectral volume method and application to a *p*-multigrid algorithm, Journal of Computational Physics 224 (2) (2006) 616–636.
[14] K. Van den Abeele, C. Lacor, An accuracy and stability study of the 2D spectral volume method, Journal of Computational Physics 226 (2007) 1007–10026.
[15] Y. Liu, M. Vinokur, Z.J. Wang, Spectral difference method for unstructured grids I: basic formulation, Journal of Computational Physics 216 (2006) 780–801.
[16] Z.J. Wang, Y. Liu, G. May, A. Jameson, Spectral difference method for unstructured grids II: extension to the Euler equations, Journal of Scientific Computing 32 (1) (2007) 449–454.
[17] Y. Sun, Z.J. Wang, Y. Liu, High-order multidomain spectral difference method for the Navier–Stokes equations on unstructured hexahedral grids, Communications in Computational Physics 2 (2) (2007) 310–333.
[18] H.T. Huynh, A flux reconstruction approach to high-order schemes including discontinuous Galerkin methods, in: 18th AIAA Computational Fluid Dynamics Conference; AIAA 2007-4079, Miami, Florida, 25–28 June, 2007.
[19] Z.J. Wang, H. Gao, A unifying collocation penalty formulation for the Euler equations on mixed grids, in: 47th AIAA Aerospace Sciences Meeting Including The New Horizons Forum and Aerospace Exposition; AIAA 2009-401, Orlando, Florida, 5–8 January, 2009.
[20] H. Gao, Z.J. Wang, A high-order lifting collocation penalty formulation for the Navier–Stokes equations on 2D mixed grids, in: 19th AIAA Computational Fluid Dynamics Conference; AIAA 2009-3784, San Antonio, Texas, 22–25 June, 2009.
[21] C. Liang, S. Premasuthan, A. Jameson, Z.J. Wang, Large eddy simulation of compressible turbulent channel flow with spectral difference method, in: 47th AIAA Aerospace Sciences Meeting Including The New Horizons Forum and Aerospace Exposition; AIAA 2009-402, Orlando, Florida, 5–8 January, 2009.
[22] R.D. Moser, J. Kim, N.N. Mansour, DNS of turbulent channel flow up to $Re_\tau = 590$, Physics of Fluids 11 (1999) 943–945.
[23] F. Nicoud, F. Ducros, Subgrid-scale stress modelling based on the square of the velocity gradient tensor, Flow, Turbulence and Combustion 62 (3) (1999) 183–200.
[24] G. Ghorbaniasl, Computational Aeroacoustic-Noise Prediction Using Hybrid Methodologies, Ph.D. Thesis, Vrije Universiteit Brussel, Bruxelles, September 2009.
[25] T. Iliescu, V. Jhon, W.J. Layton, G. Matthies, L. Tobiska, A numerical study of a class of LES models, International Journal of Computational Fluid Dynamics 17 (1) (2003) 75–85.
[26] A. Jameson, E. Turkel, Implicit schemes and LU decompositions, Mathematics of Computations 37 (1981) 385–397.
[27] A. Jameson, S. Yoon, Lower–upper implicit schemes with multiple grids for the Euler equations, AIAA Journal 25 (7) (1987) 929–935.
[28] Y. Sun, Z.J. Wang, Y. Liu, C.-L. Chen, Efficient implicit LU-SGS algorithm for high-order spectral difference method on unstructured hexahedral grids, in: 45th AIAA Aerospace Sciences Meeting; AIAA 2007-313, Reno, Nevada, 8–11 January, 2007.
[29] M. Parsani, K. Van den Abeele, C. Lacor, E. Turkel, Simulation of compressible turbulent flows with an implicit LU-SGS algorithm for high-order spectral difference method on unstructured grids, in: 19th AIAA Computational Fluid Dynamics Conference; AIAA 2009-4143, San Antonio, Texas, 22–25 June, 2009.
[30] F. Bassi, S. Rebay, G. Mariotti, S. Pedinotti, M. Savini, A high-order accurate discontinuous finite element method for inviscid and viscous turbomachinery flows, Proceedings of Second European Conference on Turbomachinery, vol. 1, Technologisch Instituut, Antwerpen, Belgium, 1997, pp. 99–108.
[31] C. Fureby, G. Tabor, Mathematical and physical constraits on large-eddy simulations, Theoretical and Computational Fluid Dynamics 9 (2) (1997) 85–102.
[32] T.A. Zang, P.R. Dahlburg, J.P. Dahlburg, Direct and large eddy simulation of three dimensional compressible Navier–Stokes turbulence, Physics of Fluids A4 (1) (1992) 127–140.
[33] K. Van den Abeele, C. Lacor, Z.J. Wang, On the stability and accuracy of the spectral difference method, Journal on Scientific Computing 37 (2) (2008) 162–188.
[34] P.D. Lax, Weak solutions of non-linear hyperbolic equations and their numerical computation, Communications on Pure and Applied Mathematics VII (1954) 159–193.
[35] P.L. Roe, Approximate Riemann solvers, parameter vectors, and difference scheme, Journal of Computational Physics 43 (1981) 357–372.
[36] D.N. Arnold, F. Brezzi, B. Cockburn, L.D. Marini, Unified analysis of discontinuous Galerkin methods for elliptic problems, SIAM Journal on Numerical Analysis 39 (5) (2002) 1749–1779.
[37] K. Van den Abeele, M. Parsani, C. Lacor, An implicit spectral difference Navier–Stokes solver for unstructured hexahedral grids, in: 47th AIAA Aerospace Sciences Meeting Including The New Horizons Forum and Aerospace Exposition; AIAA 2009-0181, Orlando, Florida, 5–9 January, 2009.
[38] A. Quarteroni, R. Sacco, F. Saleri, Numerical Mathematics, Springer, Berlin, 2002.
[39] Y. Saad, M.H. Schultz, GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems, SIAM Journal on Scientific and Statistical Computing 7 (1986) 856–869.

[40] D.A. Knoll, D.A. Keyes, Jacobian-free Newton–Krylov methods: a survey of approaches and applications, Journal of Computational Physics 193 (2004) 357–397.
[41] R. Sandberg, L. Jones, N. Sandham, P. Joseph, Direct numerical simulations of tonal noise generated by laminar flow past airfoils, Journal of Sound and Vibration 320 (2008) 838–858.
[42] N.D. Sandham, Q. Li, H.C. Yee, Entropy splitting for high-order numerical simulation of compressible turbulence, Journal of Computational Physics 178 (2002) 307–322.
[43] J.G. Wissink, DNS of 2D turbulent flow around a square cylinder, International Journal for Numerical Methods in Fluids 25 (1997) 51–62.
[44] D. Bouris, G. Bergeles, 2D LES of vortex shedding from a square cylinder, Journal of Wind Engineering and Industrial Aerodynamics 80 (1999) 31–46.
[45] C. Geuzaine, J.-F. Remacle, Gmsh: a three-dimensional finite element mesh generator with built-in pre- and post-processing facilities, International Journal for Numerical Methods in Engineering 79 (11) (2009) 1309–1331.
[46] T. Quintino, A Component Environment for High-Performance Scientific Computing: Design and Implementation, Ph.D. Thesis, Katholike Universiteit Leuven, von Karman Institute for Fluid Dynamics, Leuven, December 2008.
[47] M. Parsani, K. Van den Abeele, C. Lacor, E. Turkel, Implicit LU-SGS algorithm for high-order methods on unstructured grid with *p*-multigrid strategy for solving the steady Navier–Stokes equations, Journal of Computational Physics 229 (2010) 828–850.
[48] S.B. Pope, Turbulent Flows, Cambridge University Press, 2000.
[49] D. Durão, M. Heitor, J. Pereira, Measurements of turbulent and periodic flows around a square cross-section cylinder, Experiments in Fluids 6 (1988) 298–304.
[50] D. Lyn, S. Einav, W. Rodi, J. Park, A laser doppler velocimetry study of ensemble-averaged characteristics of the turbulent near wake of a square cylinder, Journal of Fluid Mechanics 304 (1995) 285–319.
[51] R. Franke, W. Rodi, Calculation of vortex shedding past a square cylinder with various turbulence models, Proceedings of Eighth Symposium Turbulent Shear Flows, Springer, Berlin; Munich, 1991. 9–11 September.
[52] W. Rodi, On the simulation of turbulent flow past bluff bodies, Journal of Wind Engineering and Industrial Aerodynamics 46–47 (1993) 3–19.
[53] S. Murakami, A. Mochida, On turbulent vortex shedding flow past a square cylinder predicted by CFD, Journal of Wind Engineering and Industrial Aerodynamics 54 (1995) 191–211.
[54] M. Breuer, M. Poriquie, First experiences with LES of flows past bluff bodies, Third International Symposium on Engineering Turbulence Modelling and Measurements, Elsevier, Amsterdam, 1996.