Journal of Computational Physics 241 (2013) 388–415

Contents lists available at SciVerse ScienceDirect

Journal of Computational Physics

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

A Spalart–Allmaras turbulence model implementation in a discontinuous Galerkin solver for incompressible flows

Andrea Crivellini [a,*], Valerio D'Alessandro [a], Francesco Bassi [b]

[a] Dipartimento di Ingegneria Industriale e Scienze Matematiche, Università Politecnica delle Marche, Via Brecce Bianche, 60100 Ancona (AN), Italy
[b] Dipartimento di Ingegneria Industriale, Università di Bergamo Viale Marconi 5, 24044 Dalmine (BG), Italy

ARTICLE INFO

Article history:
Received 16 February 2012
Received in revised form 23 December 2012
Accepted 28 December 2012
Available online 8 February 2013

Keywords:
CFD
High-order methods
Artificial compressibility
RANS equations

ABSTRACT

In this paper the artificial compressibility flux Discontinuous Galerkin (DG) method for the solution of the incompressible Navier–Stokes equations has been extended to deal with the Reynolds-Averaged Navier–Stokes (RANS) equations coupled with the Spalart–Allmaras (SA) turbulence model. DG implementations of the RANS and SA equations for compressible flows have already been reported in the literature, including the description of limiting or stabilization techniques adopted in order to prevent the turbulent viscosity $\tilde{\nu}$ from becoming negative. In this paper we introduce an SA model implementation that deals with negative $\tilde{\nu}$ values by modifying the source and diffusion terms in the SA model equation only when the working variable or one of the model closure functions become negative. This results in an efficient high-order implementation where either stabilization terms or even additional equations are avoided. We remark that the proposed implementation is not DG specific and it is well suited for any numerical discretization of the RANS-SA governing equations. The reliability, robustness and accuracy of the proposed implementation have been assessed by computing several high Reynolds number turbulent test cases: the flow over a flat plate ($Re = 10^7$), the flow past a backward-facing step ($Re = 37400$) and the flow around a NACA 0012 airfoil at different angles of attack ($\alpha = 0^\circ, 10^\circ, 15^\circ$) and Reynolds numbers ($Re = 2.88 \times 10^6, 6 \times 10^6$).

1. Introduction

Discontinuous Galerkin (DG) methods for Computational Fluid Dynamics have been first applied to strictly hyperbolic problems as the Euler equations [1,2]. After the pioneering work of Bassi and Rebay [3], DG methods have received much attention also for the high-order discretization of elliptic problems and, since then, many stable DG discretization schemes for the diffusive terms of the compressible Navier–Stokes equations have been developed. More recently, DG methods suited for the incompressible Navier–Stokes (INS) equations have been proposed and analyzed in a number of papers [4–11].

Papers about the high-order DG space discretization of the Reynolds-Averaged Navier–Stokes (RANS) equations are far less numerous and they only deal with compressible flows. This is to be related to severe problems of numerical stiffness induced by the highly non-linear source terms of the turbulence model equations and by grid stretching needed to resolve the near-wall behaviour of the turbulent quantities. Thus, high-order DG methods for the RANS and turbulence model equations must be carefully set up and always include some form of limiting or stabilization terms in the turbulence model equations in order to prevent blow-up of computations.

* Corresponding author.
E-mail addresses: a.crivellini@univpm.it (A. Crivellini), v.dalessandro@univpm.it (V. D'Alessandro), francesco.bassi@unibg.it (F. Bassi).

0021-9991/$ - see front matter © 2013 Elsevier Inc. All rights reserved.
http://dx.doi.org/10.1016/j.jcp.2012.12.038

At the time of this writing, the authors are aware of few DG implementations of the RANS equations coupled with two turbulence models, the one-equation SA and the two-equation k—ω models. Bassi et al. [12] solved the RANS and k—ω equations with some modifications in the original model consisting of (i) rewriting the ω equation in terms of $\log(\omega)$ instead of ω, (ii) fulfilling the realizability constraints for the turbulent stresses (obtaining a lower bound for ω), and (iii) enforcing a zero lower bound for the turbulent kinetic energy in the source terms and in the turbulent viscosity constitutive relation. Persson et al. [13] solved the RANS and the SA turbulence model equations using an artificial viscosity term in order to stabilize the discretization of the turbulence model equation. Oliver and Darmofal [14,15] also developed a DG solver for the RANS equations and the SA model. They added an artificial dissipation, governed by an additional PDE, to avoid numerical oscillations in under-resolved regions. Landmann et al. [16] solved the RANS equations coupled with both the SA and k—ω turbulence models. For the k—ω model they used the same approach adopted by Bassi et al. [12] while for the SA model a strong instability for negative values of eddy viscosity was observed. Therefore they implemented a limiting technique that after each Newton iteration resets to zero negative values of the computed turbulence variable. Finally, Hartmann et al. [17] developed a DG code for turbulent flow computations using the same modifications proposed by Bassi et al. in [12] for the k—ω equations.

This work continues the development of the DG method for the INS equations introduced in [18] and extended in [19] to natural convection flows and in [20] to unsteady flows. The main feature of the method is an original formulation of the inviscid interface numerical fluxes based on the solution of Riemann problems with a relaxed incompressibility constraint, [18]. The numerical experiments reported in Bassi et al. [18] show that the convergence rate, using polynomial approximations of degree k for all the variables, is $k+1$ for the velocity components and at least k for the pressure.

In this paper the method has been extended to the numerical solution of the incompressible RANS equations coupled with the SA model. One advantage of using this model is that the turbulent variable $\tilde{\nu}$ has less stringent requirements on near-wall grid resolution needed to capture velocity profiles with respect to other models, especially of the k—ϵ type. For high-order computations the dimensionless height y^+ of elements next to the wall can be taken of order 10, see for example [21] or the results reported in Section 5.1. Despite of its relative simplicity, the robustness and accuracy of the SA model for aerodynamic flows has been demonstrated in very many papers and its extension to more advanced levels of turbulence modelling such as the Detached Eddy Simulation (DES), which is one of our goals, has already been attempted with promising results. Finally, the ideal boundary conditions for the turbulent variable at far-field and wall boundaries are simply homogeneous boundary conditions. In the next sections we present and analyse an implementation of the SA model that addresses the above mentioned issues related to high-order DG implementations. In particular, we will show how few careful corrections in the source and diffusive terms of the model are enough to stabilise its numerical behaviour and to avoid unphysical turbulent quantities.

The paper is organised as follows. Section 2 presents the governing equations with a detailed analysis of the proposed modifications in the SA model. Sections 3 and 4 describe the DG discretization and the implicit Newton–Krylov solution algorithm. In the final section, Section 5, numerical results of several test cases are reported.

2. Governing equations

The governing equations of the fluid flow model considered in this paper are the incompressible RANS equations coupled with the one-equation SA turbulence model. Let $\Omega \subset \mathbb{R}^N,\ N \in \{2,3\}$, be a bounded and connected Lipschitz domain. Adopting the standard SA nomenclature, the complete set of governing equations, in $[0,t] \times \Omega$, reads:

$$\begin{aligned} &\nabla \cdot \mathbf{u} = 0,\\ &\frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot (\mathbf{u} \otimes \mathbf{u}) + \nabla p - \nabla \cdot \left((\nu + \nu_t)\left(\nabla \mathbf{u} + \nabla \mathbf{u}^T\right)\right) = \mathbf{0},\\ &\frac{\partial \tilde{\nu}}{\partial t} + \nabla \cdot (\mathbf{u}\tilde{\nu}) - \frac{1}{\sigma}\nabla \cdot ((\nu + \tilde{\nu})\nabla\tilde{\nu}) = s, \end{aligned} \tag{1}$$

where $p = P/\rho$ is the pressure dived by density and the turbulent viscosity, ν_t, is computed according to the equation

$$\nu_t = f_{v1}\tilde{\nu}. \tag{2}$$

In Eq. (1) the source term, s, is given by

$$s \stackrel{\text{def}}{=} c_{b1}\tilde{S}\tilde{\nu} + \frac{c_{b2}}{\sigma}\nabla\tilde{\nu} \cdot \nabla\tilde{\nu} - c_{w1}f_w\left(\frac{\tilde{\nu}}{d}\right)^2, \tag{3}$$

where d is the minimum distance from the wall and $\tilde{S}$, which is a function of both the vorticity magnitude S and the turbulent variable $\tilde{\nu}$, is the following production term

$$\tilde{S} = S + \frac{\tilde{\nu}}{k^2 d^2} f_{v2}, \quad \mathbf{\Omega} = \frac{\nabla \mathbf{u} - \nabla \mathbf{u}^T}{2}, \quad S = \sqrt{2\mathbf{\Omega} : \mathbf{\Omega}}. \tag{4}$$

To completely define the PDEs the closure functions

$$\begin{aligned}
&\chi = \frac{\tilde{\nu}}{\nu}, \quad f_{v1} = \frac{\chi^3}{(\chi^3 + c_{v1}^3)},\\
&f_{v2} = 1 - \frac{\chi}{(1 + \chi f_{v1})}, \quad f_w = g\left[\frac{1 + c_{w3}^6}{g^6 + c_{w3}^6}\right]^{\frac{1}{6}},\\
&g = r + c_{w2}(r^6 - r), \quad r = \frac{\tilde{\nu}}{\tilde{S}k^2d^2}
\end{aligned} \tag{5}$$

and closure constants

$$\begin{aligned}
&c_{b1} = 0.1355, \quad c_{b2} = 0.622, \quad c_{v1} = 7.1,\\
&\sigma = 2/3, \quad c_{w1} = \frac{c_{b1}}{k^2} + \frac{(1 + c_{b2})}{\sigma},\\
&c_{w2} = 0.3, \quad c_{w3} = 2, \quad k = 0.41
\end{aligned} \tag{6}$$

are required. The resulting turbulence model is the standard SA model without the trip function, hence employed in a fully turbulent mode.

2.1. SA modifications

The SA model described in [22] was designed so as to incorporate four nested versions. The first version, which is the simplest one, is applicable only to free shear flows; the last version, the most complex one, is applicable to viscous flows past solid bodies with a specified turbulent transition location. The model complexity is increased adding new terms, passive in all the lower versions of the model. This means that near the edge of boundary layers as well as in free-shear layers only the simplest model is active. This class of problems admits typical ramp solutions related to the following sub-model:

$$\frac{\partial \tilde{\nu}}{\partial t} + \nabla \cdot (\mathbf{u}\tilde{\nu}) - \frac{1}{\sigma}\nabla \cdot ((\nu + \tilde{\nu})\nabla\tilde{\nu}) = \frac{c_{b2}}{\sigma}\nabla\tilde{\nu} \cdot \nabla\tilde{\nu} + c_{b1}S\tilde{\nu}. \tag{7}$$

It is already known that the eddy viscosity decreases rapidly from a positive value to the free-stream value (ideally zero but usually set to a positive very small value) changing its curvature in a narrow layer. For high Reynolds number flows this layer is extremely thin, its thickness being determined by the laminar viscosity. Moreover in the limit of $Re \to \infty$, as noted by Spalart and Allmaras [22], the equation admits weak solutions, with discontinuous $\nabla\tilde{\nu}$.

The numerical stability of high-order DG solvers is closely related to this phenomenon. Persson et al. [13] and Oliver and Darmofal [14,15] have recognised in this behaviour the cause of highly oscillatory, or even unstable, solutions characterised by negative $\tilde{\nu}$ values. It is clear that a negative turbulent eddy viscosity is meaningless and that the original closure functions, Eq. (5), and closure constants, Eq. (6), were developed only for positive $\tilde{\nu}$ (the function f_{v1}, for example, is singular for $\chi = -c_{v1}$).

As outlined in Section 1, several authors have proposed different numerical approaches to address the stability issues of high-order DG implementations of the SA model. In this work only corrections for negative values of $\tilde{\nu}$ and for the closure function r have been adopted, resulting in an efficient implementation of the RANS high-order solver since neither additional equations/terms nor complex limiting procedures, which for an high-order approximation may be computationally expensive, are employed. Hence, the key ingredients for a stable and reliable high-order implementation of the SA model are here strictly related to the treatment of the source and diffusive terms.

2.1.1. Negative $\tilde{\nu}$ modifications

In the proposed algorithm when $\tilde{\nu}$ is negative the turbulence model equation is modified in order to ensure an increasing turbulent viscosity. A first version of this approach has been suggested by Allmaras himself in [23] while the current version is unpublished and it was only reported, as a personal communication, by Oliver [24]. Following Oliver presentation, one multiplies the model equation by its independent variable to obtain the following partial differential equation for the energy of $\tilde{\nu}$

$$\tilde{\nu}\frac{\partial \tilde{\nu}}{\partial t} + \tilde{\nu}\nabla \cdot (\mathbf{u}\tilde{\nu}) = \frac{1}{2}\left(\frac{\partial \tilde{\nu}^2}{\partial t} + \nabla \cdot (\mathbf{u}\tilde{\nu}^2)\right) = \frac{\tilde{\nu}}{\sigma}\nabla \cdot (\xi\nabla\tilde{\nu}) + \tilde{\nu}s, \tag{8}$$

where for convenience we have set the diffusion coefficient $\xi = \nu + \tilde{\nu} = \nu(1 + \chi)$. Defining Ω^+ and Ω^- the sub-domains of $\Omega = \Omega^+ \cup \Omega^-$, in which $\tilde{\nu}$ is positive and negative, respectively, and assuming that $\partial\Omega \cap \partial\Omega^- = 0$, *i.e.* Ω^- is internal to Ω, and that the function $\tilde{\nu}$ is $C_0(\Omega)$, the integration of Eq. (8) over the domain Ω^- gives:

$$\frac{1}{2}\int_{\Omega^-}\frac{\partial \tilde{\nu}^2}{\partial t}\mathrm{d}\mathbf{x} + \underbrace{\frac{1}{2}\int_{\partial\Omega^-}\mathbf{u}\tilde{\nu}^2 \cdot \mathbf{n}\mathrm{d}\sigma}_{0} - \underbrace{\int_{\partial\Omega^-}\frac{\tilde{\nu}}{\sigma}\tilde{\nu}\nabla\tilde{\nu} \cdot \mathbf{n}\mathrm{d}\sigma}_{0} + \int_{\Omega^-}\frac{\nabla\tilde{\nu}}{\sigma} \cdot \xi\nabla\tilde{\nu}\mathrm{d}\mathbf{x} = \int_{\Omega^-}\tilde{\nu}s\mathrm{d}\mathbf{x},$$

where the Gauss divergence theorem is used in order to simplify the equation, being for continuity $\tilde{\nu}|_{\partial\Omega^-} = 0$. By using the source term definition, Eq. (3), one obtains

$$\frac{1}{2}\int_{\Omega^-}\frac{\partial\tilde{\nu}^2}{\partial t}d\mathbf{x} = \int_{\Omega^-}\underbrace{\frac{c_{b2}\tilde{\nu}-\xi}{\sigma}\nabla\tilde{\nu}\cdot\nabla\tilde{\nu}}_{\tilde{\nu}s_C}d\mathbf{x} + \int_{\Omega^-}\underbrace{c_{b1}\tilde{S}\tilde{\nu}^2}_{\tilde{\nu}s_P}d\mathbf{x} - \int_{\Omega^-}\underbrace{c_{w1}f_w\frac{\tilde{\nu}^3}{d^2}}_{\tilde{\nu}s_D}d\mathbf{x}.$$

Since the convective fluxes are null, this equation represents the rate of change of the $\tilde{\nu}$ "energy" associated with a control volume moving together with the boundary of Ω^-. The terms $\tilde{\nu}s_C$, $\tilde{\nu}s_P$ and $\tilde{\nu}s_D$ represent diffusion plus cross term, production term and destruction term, respectively. Thus, if $\tilde{\nu}$ is negative, $\tilde{\nu}s_C$ and $\tilde{\nu}s_P$ must be less than zero while $\tilde{\nu}s_D$ must be greater than zero in order to avoid an increasing "energy".

The $\tilde{\nu}s_C$ contribution, given by

$$\tilde{\nu}s_C = \frac{c_{b2}\tilde{\nu}-\xi}{\sigma}\nabla\tilde{\nu}\cdot\nabla\tilde{\nu},$$

is positive if $\tilde{\nu} < 0$ and $(c_{b2}\tilde{\nu}-\xi) > 0$ and thus the following alternative definition of ξ, ensuring the desired sign, is adopted

$$\xi = \begin{cases} \nu(1+\chi) & \chi \geqslant 0, \\ \nu\left(1+\chi+\frac{1}{2}\chi^2\right) & \chi < 0. \end{cases} \tag{9}$$

The original $\tilde{\nu}s_P$ term is a function of $S,\ d^2$ and χ, but for negative χ it is always positive for $\chi > -1.940 \times 10^1$; therefore the production term is modified as

$$s_P = \begin{cases} c_{b1}\tilde{S}\tilde{\nu} & \chi \geqslant 0, \\ c_{b1}S\tilde{\nu}g_n & \chi < 0, \end{cases} \tag{10}$$

where

$$g_n = 1 - \frac{10^3\chi^2}{1+\chi^2}. \tag{11}$$

If $\chi < -\sqrt{1/999}$ the g_n function is negative and $\tilde{\nu}s_P$ is positive. This region in which the production term has the "wrong" sign is unavoidable if the continuity of Jacobian is requested. Since $s_P|_{\chi=0} = 0$ and $\frac{\partial s_P}{\partial\chi}|_{\chi=0} = c_{cb1}S\nu$ with $S > 0$ the production term must change sign in $\chi = 0$. The function in Eq. (11) clusters this undesirable behaviour in a small interval, for $-\sqrt{1/999} < \chi < 0$, and limits the maximum $\tilde{\nu}s_P$ "adverse" value.

Finally, the sign of $\tilde{\nu}s_D$ depends on the sign of $\tilde{\nu}$ and f_w, the latter being negative for $-1.185 < r < 0$; thus, once again, a new formulation is introduced

$$s_D = \begin{cases} c_{w1}f_w\frac{\tilde{\nu}^2}{d^2} & \chi \geqslant 0, \\ -c_{w1}\frac{\tilde{\nu}^2}{d^2} & \chi < 0. \end{cases} \tag{12}$$

Note that all these negative $\tilde{\nu}$ modifications have been devised so as to ensure the continuity of the function and of its first derivative with respect to $\tilde{\nu}$, *i.e.* the resulting Jacobian matrix is continuous even in $\tilde{\nu} = 0$.

Assembling all the modifications in a single source term, the Allmaras approach reads

$$s = \begin{cases} c_{b1}\tilde{S}\tilde{\nu} + \frac{c_{b2}}{\sigma}\nabla\tilde{\nu}\cdot\nabla\tilde{\nu} - c_{w1}f_w\left(\frac{\tilde{\nu}}{d}\right)^2 & \chi \geqslant 0, \\ c_{b1}S\tilde{\nu}g_n + \frac{c_{b2}}{\sigma}\nabla\tilde{\nu}\cdot\nabla\tilde{\nu} + c_{w1}\frac{\tilde{\nu}^2}{d^2} & \chi < 0. \end{cases} \tag{13}$$

The second approach to deal with negative $\tilde{\nu}$ values follows the DG implementation of the k–ω turbulence model introduced in Bassi et al. [12]. If we interpret this approach in the framework of the above described modifications of negative turbulence model quantities, it takes the meaning of a simple algorithm ensuring a positive diffusion coefficient (equal to the molecular viscosity) and zero source terms if the turbulent viscosity quantities happen to be negative. In formulas, this approach reads

$$\xi = \begin{cases} \nu(1+\chi) & \chi \geqslant 0, \\ \nu & \chi < 0, \end{cases} \tag{14}$$

$$s = \begin{cases} c_{b1}\tilde{S}\tilde{\nu} + \frac{c_{b2}}{\sigma}\nabla\tilde{\nu}\cdot\nabla\tilde{\nu} - c_{w1}f_w\left(\frac{\tilde{\nu}}{d}\right)^2 & \chi < 0, \\ 0 & \chi < 0, \end{cases} \tag{15}$$

which guarantees the fulfilment of conditions for an increasing negative $\tilde{\nu}$. In this case, however, the Jacobian is not continuous anymore. Unlike what Landmann et al. claim in [16], we have found that this algorithm works well without any stability issue. As already observed by Allmaras [23] for his implementation, both approaches reduce to original SA model for a positive (*i.e.* meaningful) turbulent viscosity solution, being this behaviour enforced by the constraint

$$\nu_t = f_{v1}\max(0, \tilde{\nu}). \tag{16}$$

For this reason the two approaches to deal with negative $\tilde{\nu}$ values should be viewed just as implementation variants of the original SA model.

To highlight typical numerical issues related to inadequate treatment of negative $\tilde{\nu}$ values, Fig. 1(a) shows the turbulent viscosity profile in the shear-layer behind a NACA 0012 airfoil. The effect of negative $\tilde{\nu}$ correction for the diffusion coefficient can be easily appreciated: compared to the standard model, the amplitude of solution oscillations is drastically reduced, and the numerical error does not significantly propagate outside the turbulent region. Nevertheless, solutions inside this region appear almost indistinguishable, proving that this technique does not affect positive $\tilde{\nu}$ values, see Fig. 1(a) and (b).

2.1.2. The negative r modifications

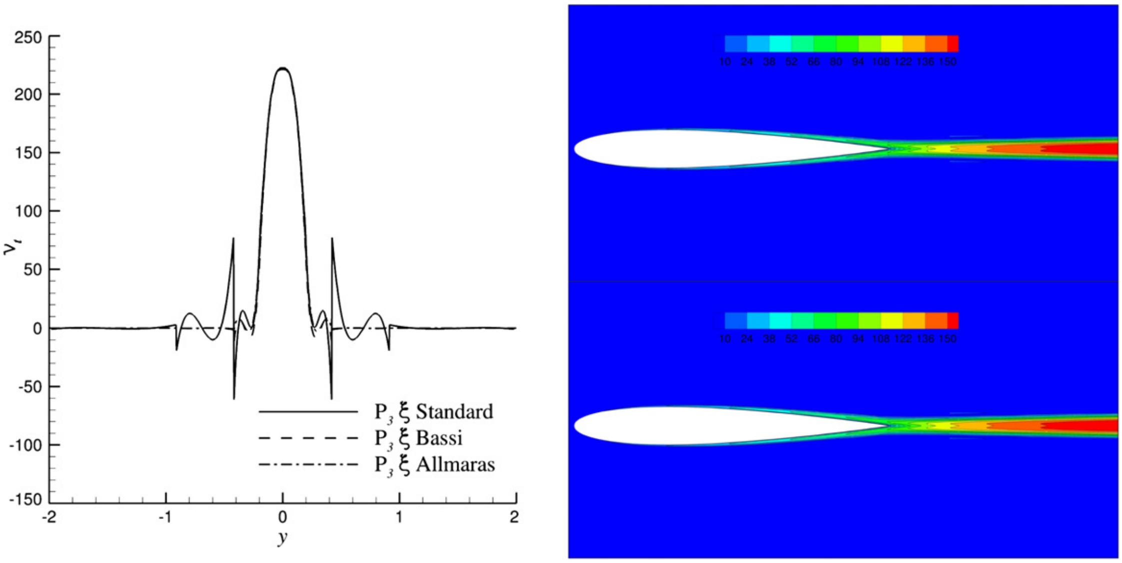
The second key ingredient of the proposed high-order SA implementation can be considered a kind of realizability condition for the r closure function, Eq. (5). In this section $\tilde{\nu}$, hence χ, is assumed to be positive, since in the negative $\tilde{\nu}$ case the r function is not evaluated at all. In the paper where the SA model was presented for the first time, [22], this quantity was introduced according to the definition $r = l^2/\left(k^2d^2\right)$, where l is a mixing length inspired to algebraic turbulence models. Substituting Eq. (4) in the r closure function of Eq. (5) yields

$$r = \frac{\tilde{\nu}}{\tilde{S}(kd)^2} = \frac{\chi}{\frac{S(kd)^2}{\nu} + \chi f_{v2}}, \tag{17}$$

which means that r is a function of the two, positive defined, non-dimensional groups, $S(kd)^2/\nu$ and χ, related to the vorticity magnitude and to the turbulent eddy-viscosity, respectively. By definition the r function should be positive, however, for $S(kd)^2/\nu \leqslant 6.088$, negative values can occur in Eq. (5) even with $\tilde{\nu} \geqslant 0$, see Fig. 2(a). This behaviour is related to the function f_{v2} which is negative for $1.003 \leqslant \chi \leqslant 1.840 \times 10^1$, Fig. 2(b). Moreover r is singular for $\chi f_{v2} = -S(kd)^2/\nu$. The contours plot of the r function, Fig. 3(a), shows that the region with $r < 0$ is bounded by a line along which the function is singular.

To avoid the singularity here r is limited to a positive value, r_{max}. Moreover, since the original function r is negative only if it crosses the curve where $r \rightarrow \pm\infty$, negative values are handled in the same way as large values. This means that the following new r definition takes the place of the original one in Eq. (5):

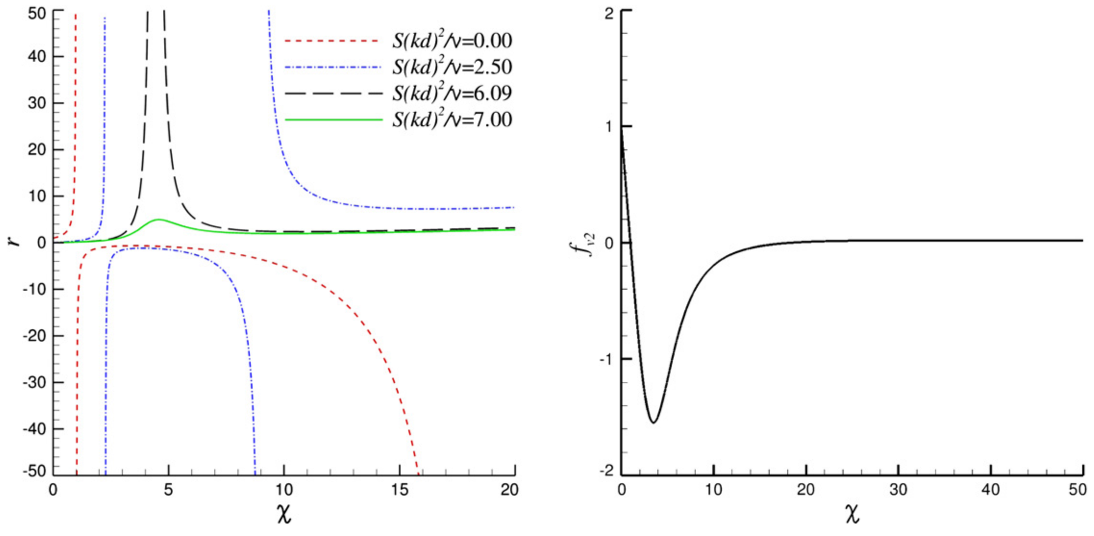
$$r^* = \frac{\tilde{\nu}}{\tilde{S}k^2d^2}, \quad r = \begin{cases} r_{max} & r^* < 0, \\ r^* & 0 \leqslant r^* < r_{max}, \\ r_{max} & r^* \geqslant r_{max}. \end{cases} \tag{18}$$



(a) $\tilde{\nu}$ profile in the wake, $x = 25$.

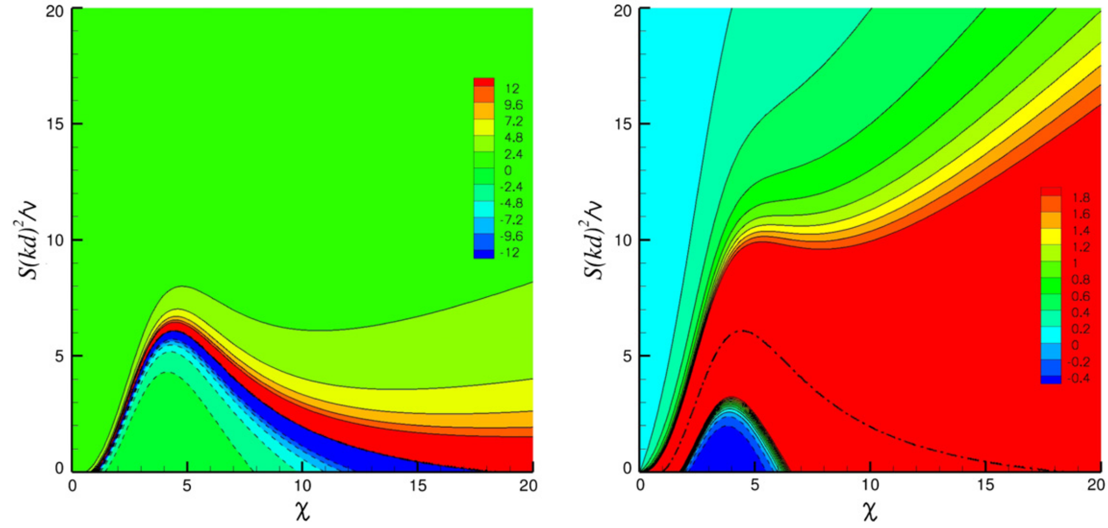
(b) $\mathbb{P}_3$ positive $\tilde{\nu}$. Top: ξ standard; Bottom: ξ Allmaras.

Fig. 1. Effect of the correction equation (9) and (14) on the numerical solution, $\tilde{\nu}$ behaviour on the NACA 0012 wake. $Re = 10^6$, $\alpha = 0^\circ$.



(a) r. (b) f_{v2}.

Fig. 2. Closure functions, standard SA model.



(a) r. (b) f_w.

Fig. 3. Contour plot of the closure functions, standard SA model, solid lines for positive iso-value curves, dashed lines for negative iso-value curves, dot-dashed line for singular curve.

Furthermore, in order to avoid a negative production term, as will become more clear in the following, here the s_P is rearranged as

$$s_P = c_{b1}\frac{\tilde{\nu}^2}{rk^2d^2}, \tag{19}$$

which is equivalent to the original formulation if the standard r definition is assumed. Eq. (19) indicates that when $r < 0$ the standard s_P term is negative; moreover, even the standard destruction term, $s_D = c_{w1}f_w\tilde{\nu}^2/d^2$, may change its original sign for $r < 0$. As a matter of the fact, Fig. 3(b) shows that, for $r < 0$, the f_w closure function can be negative in the region below the dot-dashed line.

It is very important to realize that also this modification does not affect the final solution. In fact, the SA equation is built to obtain $r = 1$ close to the wall, where $\tilde{\nu} \sim u_\tau kd$ and $\tilde{S} \sim u_\tau/(kd)^2$ (u_τ is the friction velocity), and $r = 0$ at the boundary edge where $\tilde{\nu}$ goes to zero faster than S. Obviously far from the wall, in free shear region, r should not influence the solution since it was originally introduced in the turbulence model only as the argument of the f_w function which serves the purpose of the model calibration for the near-wall region. Furthermore, quoting the Spalart and Allmaras paper, [22], "the region $r > 1$ is exercised only in adverse pressure gradient, and then rarely beyond $r = 1.1$".

However, from the numerical point of view, at the edge of boundary layer both $\tilde{\nu}$, as noticed in previous section, and S, see Fig. 4(a), display steep profiles resulting in oscillating solutions around zero or around a free stream value close to zero. This phenomenon can possibly lead the r function to very high or even negative values, see Fig. 4(b).

During this work we have observed that this undesired behavior of the r function is not very important for attached flow simulations, while it becomes crucial when flows with large adverse gradients and large flow separations are addressed. In these cases during the non linear solution process, using for example the implicit pseudo transient approach here adopted, the solution displays wildly oscillating residuals which can lead to blow-up of computations or compromise the non linear convergence history and thus the computational efficiency.

To a better understand of the difficulties related to the numerical treatment of the original SA model, we first rewrite the source terms, involved in the near-wall region modifications, as follows

$$s_{PD} = s_P + s_D = \left(\frac{\nu\chi}{d}\right)^2 \left(\frac{c_{b1}}{k^2 r} - c_{w1} f_w\right). \tag{20}$$

Being this source term a function of r multiplied by a positive quantity, $(\nu\chi/d)^2$, its sign is completely determined by the r value. In Fig. 5(a) the f_w function, which is negative for $-1.185 < r < 0$, is reported.

Now we can express the partial derivative of S_{pd}, with respect to $\tilde{\nu}$, as

$$\frac{\partial s_{PD}}{\partial \tilde{\nu}} = \frac{\nu\chi}{d^2}\left[2\left(\frac{c_{b1}}{k^2 r} - c_{w1} f_w\right) - \nu\chi\left(\frac{c_{b1}}{k^2 r^2} + c_{w1}\frac{df_w}{dr}\right)\frac{\partial r}{\partial \tilde{\nu}}\right],$$

where, introducing the following relations

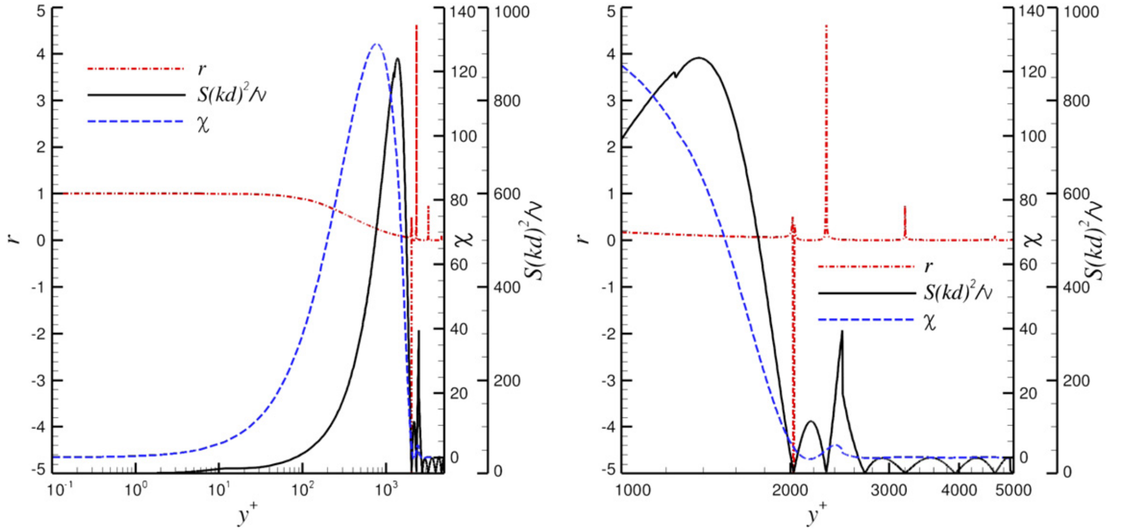
$$\frac{\partial r}{\partial \tilde{\nu}} = \left(\frac{1}{\nu\chi} - \frac{1}{\tilde{S}}\frac{\partial \tilde{S}}{\partial \tilde{\nu}}\right) r, \quad \frac{\partial \tilde{S}}{\partial \tilde{\nu}} = \frac{1}{(kd)^2}\left(f_{v2} + \chi\frac{df_{v2}}{d\chi}\right),$$

one obtains

$$\frac{\partial s_{PD}}{\partial \tilde{\nu}} = \frac{\nu\chi}{d^2}\left\{2\left(\frac{c_{b1}}{k^2 r} - c_{w1} f_w\right) - \left(\frac{c_{b1}}{k^2 r^2} + c_{w1}\frac{df_w}{dr}\right)\left[r - r^2\left(f_{v2} + \chi\frac{df_{v2}}{d\chi}\right)\right]\right\}. \tag{21}$$

Note that if $r \to \infty$ the s_{PD} source term tends to $-2.005 c_{w1}(\nu\chi/d)^2$, since $\lim_{r\to\infty} f_w = 2.005$, while if $r \to 0$ even if $c_{b1}/(k^2 r) \to \infty$ the equation is still well behaved since this is possible only for $\chi \to 0$ or for $d \to \infty$. For the Jacobian the same consideration holds true at $r = 0$, while for $r \to \infty$ it is still well defined since $\lim_{r\to\infty} r^2 df_w/dr = 0$.

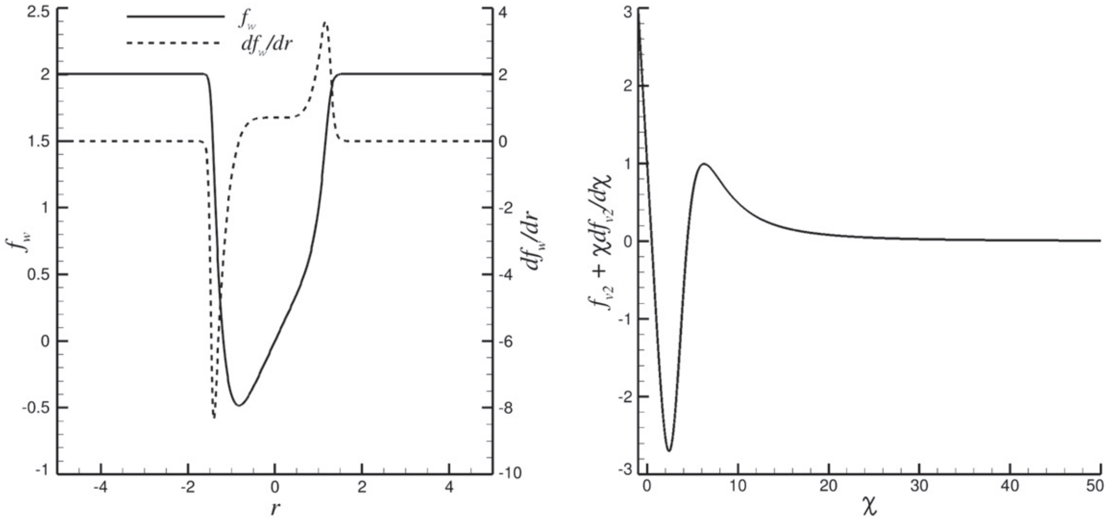
Eqs. (20) and (21) together with Eq. (17) are particularly useful for our investigation because they allow to represent the source term and its Jacobian at each point of the computational domain, thus for a particular d, and for a particular flow regime, thus for a fixed value of ν kinematic viscosity, as a function of the two non-dimensional groups introduced above. In



(a) Boundary layer.

(b) Edge of boundary layer.

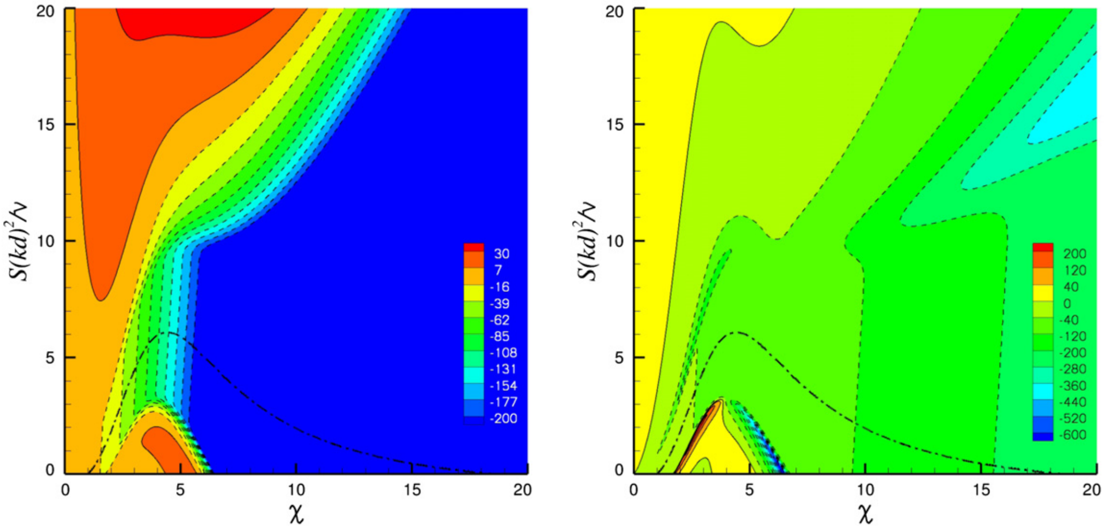
Fig. 4. Flat plate turbulent profiles at $Re_x = 2.7 \times 10^6$, computation with $\mathbb{P}_6$ and $ne = 384$.



(a) f_w and $\frac{df_w}{dr}$.

(b) $f_{v2} + \chi \frac{df_{v2}}{d\chi}$.

Fig. 5. Closure functions, standard SA model.



(a) s_{PD}.

(b) $\frac{\partial s_{PD}}{\partial \tilde{\nu}}$.

Fig. 6. Source term and its Jacobian contour plots, standard SA model, solid lines for positive iso-value curves, dashed lines for negative iso-value curves, dot-dashed line for singular curve.

Fig. 6(a) and (b) the contour plots of s_{PD} and of $\partial s_{PD}/\partial \tilde{\nu}$ are reported and it is quite easy to observe that there are two zones in which the Jacobian is positive.

The first one is where the production term is dominated by the free shear production term $c_{b1} S \tilde{\nu}$. The second is instead inside the negative r zone, below the dot-dashed line in Fig. 6(a), where the following term, related to wall destruction, may be large

$$c_{w1} \frac{\nu \chi}{d^2} r^2 \frac{df_w}{dr} \left(f_{v2} + \chi \frac{df_{v2}}{d\chi} \right).$$

In fact for $r < -8.289 \times 10^{-1}$, Fig. 5(a), and for $5.003 \times 10^{-1} < \chi < 4.421$, Fig. 5(b), both df_w/dr and $f_{v2} + \chi df_{v2}/d\chi$ are negative and, since $c_{w1} \nu \chi r^2 / d^2$ is a positive quantity, the previous term is greater than zero.

As has been already noticed for example by Allmaras [23] or Wilcox [25] a positive Jacobian source term, in the RANS closure equations, can compromise the convergence proprieties of an implicit solution algorithm. The diagonal dominance of the matrix is in fact reduced and the linearised SA equation may admit an exponential growth of the turbulence working variable. As a matter of fact, we have verified that only if the previous term is damped in the Jacobian matrix, *i.e.* is multiplied by a function which is zero at the beginning of the computation and is equal to one when the norm vector of the residuals is extremely small, the stability of the method dramatically improves, see Fig. 7(a). However this approach requires to make a choice about the shape of damping function while the r modification here proposed completely avoids this issue. Moreover the r modification does not renounce to the exact Jacobian formulation, which may be not so important in a steady state algorithm but is relevant in a URANS (Unsteady Reynolds Averaged Navier Stokes) or in a DES framework.

The idea of a cutoff for r is not completely new since Spalart and Allamars, [22], suggested to take 10 as a maximum for this function. However their observation was based only on the fact that f_w approaches a constant value, Fig. 5(a), for a sufficiently large r and they did not consider the negative case. Nevertheless, since both s_{PD} and its Jacobian are well behaved for $r \to \infty$, we adopt the r clipping only in order to avoid numerical problems in the floating point operations, as for example an overflow, or the loss of many significant digits in the closure functions evaluations. Therefore we have tested several r_{max} values much larger than 10 without any trouble or significant difference of computed results.

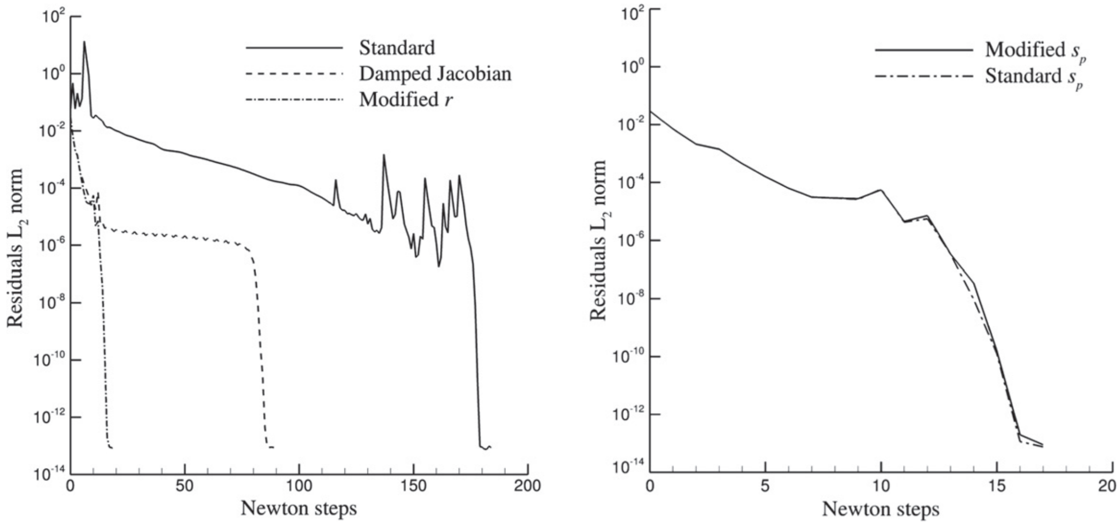
In [15,24] it was already noted that, due to the f_{v2} definition, $\tilde{S}$ can be negative even for a positive $\tilde{\nu}$ and this implies a negative value of $c_{b1}\tilde{S}\tilde{\nu}$. This behaviour is considered the responsible for stability issues and slow steady state convergence; thus a new formulation for $\tilde{S}$ was there introduced. Differently from what is claimed in those works we believe that the main numerical difficulties are strictly related to the destruction term which, in certain circumstances, behaves as a production term with an increasing production rate. Despite the different considerations which have inspired the SA modifications, even that variant of the $\tilde{S}$ definition implicitly avoids the negative r and reads:

$$\bar{S} = \frac{\tilde{\nu} f_{v2}}{k^2 d^2}, \quad \tilde{S} = \begin{cases} S + \bar{S}, & \bar{S} \geqslant -c_{v2} S, \\ S + \frac{S\left(c_{v2}^2 S + c_{v3}\bar{S}\right)}{(c_{v3} - 2c_{v2})S - \bar{S}}, & \bar{S} < -c_{v2} S, \end{cases} \quad c_{v2} = 0.7, \quad c_{v3} = 0.9.$$

This formulation is referenced in [15,24] as an unpublished work of Johnson and Allmaras. However only few details are given about how the new closure function for $\tilde{S}$ has been developed, but it should be quite clear that it was designed in order to obtain $\tilde{S} \geqslant (1 - c_{v2})S$. Using this $\tilde{S}$ definition the new source term is C_1 continuous for both $\tilde{\nu}$ and S while with our simpler approach the Jacobian is no more continuous at $r = r_{max}$. However the discontinuity, Fig. 8(b), is quite small, being, as previously noted, f_w almost constant for large r and both df_w/dr and $c_{b1}\nu\chi/(k^2 d^2 r)$ null for $r \to \infty$. Thus the only term that significantly leads to a jump in the Jacobian is

$$c_{b1}\frac{\nu\chi}{k^2 d^2}\left(f_{v2} + \chi\frac{df_{v2}}{d\chi}\right).$$

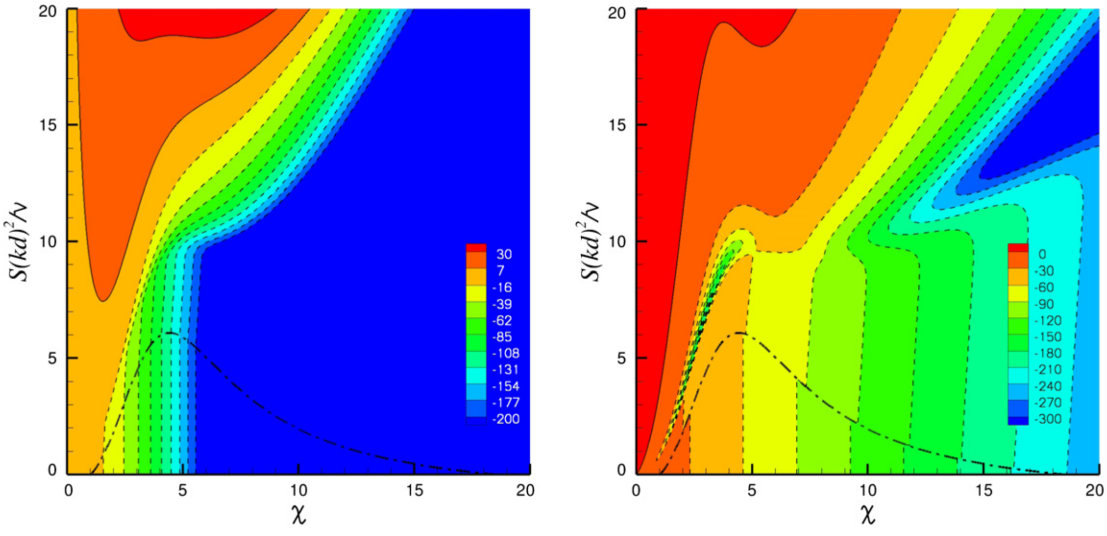
A possible cure to the discontinuous Jacobian, Fig. 8(b), is to keep the original formulation for the production term, $c_{b1}\tilde{S}\tilde{\nu}$, and to adopt the new r definition inside the destruction term. Looking at Figs. 8(b) and 9(b) it is easy to observe that the source



(a) standard SA model, standard SA with damped Jacobian matrix, modified r.

(b) Modified s_D, standard and modified s_P term.

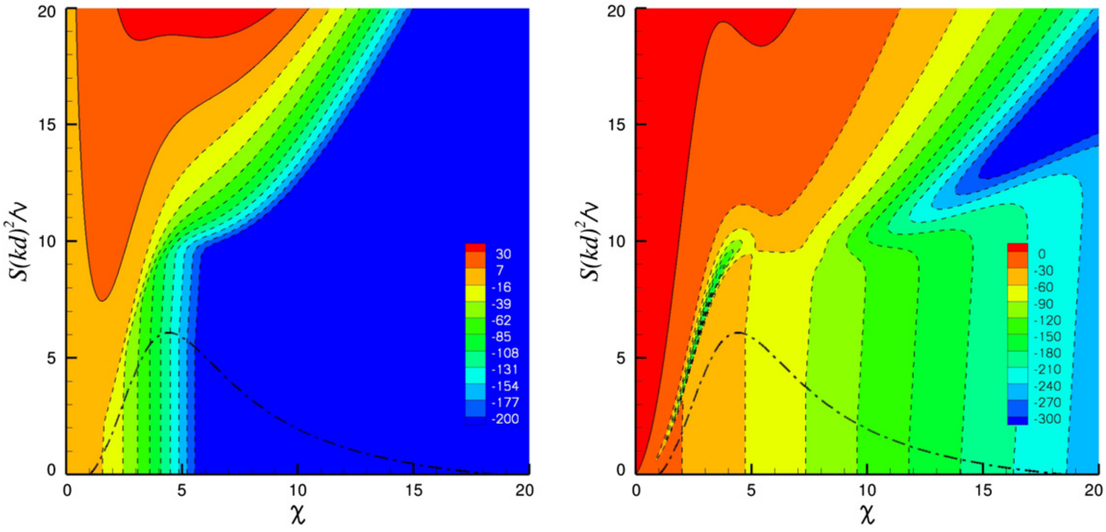
Fig. 7. Convergence histories, $\mathbb{P}_3$, NACA 0012 $Re = 2.88 \times 10^6$, $\alpha = 15^\circ$.



(a) s_{PD}.

(b) $\frac{\partial s_{PD}}{\partial \tilde{\nu}}$.

Fig. 8. Source term and its Jacobian contour plots, modified r for both s_D and s_P terms, solid lines for positive iso-value curves, dashed lines for negative iso-value curves, dot-dashed line for singular curve.



(a) s_{PD}.

(b) $\frac{\partial s_{PD}}{\partial \tilde{\nu}}$.

Fig. 9. Source term and its Jacobian contour plots, modified r for s_D, standard s_P term, solid lines for positive iso-value curves, dashed lines for negative iso-value curves, dot-dashed line for singular curve.

term s_{PD} is almost identical between the two cases, proving that for $r < 0$ the negative s_P values are negligible compared to the destruction term. Numerical experiments show that there is no remarkable difference using the original s_P or the new formulation, Eq. (19), with almost indistinguishable convergence histories, Fig. 7(b), proving once more that the responsible for instabilities and computational inefficiency is the destruction term. Anyway here we prefer to retain the physically meaningful r positive values for both source terms, accepting the new source term is only C_0.

Finally, since all the above discussed approach consists only in a particular clipping of r, we believe that the here proposed modification is extremely well suited for a straightforward implementation in many existing CFD codes. While we have verified that in a high-order framework is almost mandatory to handle in some way the negative r case, we believe that this approach can also improve the stability and the computational efficiency of standard second order codes.

3. Discontinuous Galerkin discretization

In this section we present the DG discretization of Eq. (1) considering inviscid, viscous, source terms and boundary conditions treatment separately.

3.1. Convective terms

Introducing the following convective fluxes:

$$\mathbf{F}(\mathbf{u},p)\overset{\text{def}}{=}\mathbf{u}\otimes\mathbf{u}+p\mathbf{I}=u_iu_j+p\delta_{ij},$$

$$\mathbf{g}(\mathbf{u},\tilde{v})\overset{\text{def}}{=}\mathbf{u}\tilde{v}=u_i\tilde{v}$$

and neglecting the diffusive and source terms, the weak form of Eq.(1) reads:

$$\begin{aligned}
&-\int_\Omega \nabla q\cdot\mathbf{u}\mathrm{d}\mathbf{x}+\int_{\partial\Omega} q\mathbf{u}\cdot\mathbf{n}\mathrm{d}\sigma=0,\\
&\int_\Omega \mathbf{v}\cdot\frac{\partial\mathbf{u}}{\partial t}\mathrm{d}\mathbf{x}-\int_\Omega \nabla\mathbf{v}:\mathbf{F}(\mathbf{u},p)\mathrm{d}\mathbf{x}+\int_{\partial\Omega}\mathbf{v}\otimes\mathbf{n}:\mathbf{F}(\mathbf{u},p)\mathrm{d}\sigma=\mathbf{0},\\
&\int_\Omega q\frac{\partial\tilde{v}}{\partial t}\mathrm{d}\mathbf{x}-\int_\Omega \nabla q\cdot\mathbf{g}(\mathbf{u},\tilde{v})\mathrm{d}\mathbf{x}+\int_{\partial\Omega} q\mathbf{g}(\mathbf{u},\tilde{v})\cdot\mathbf{n}\mathrm{d}\sigma=0
\end{aligned}\tag{22}$$

for arbitrary test functions $\mathbf{v}$ and q.

In order to construct the DG discretization of Eq. (22), we consider a triangulation $\mathcal{T}_h=\{K\}$ of an approximation Ω_h of Ω, that is we partition Ω_h into a set of non-overlapping elements K (not necessarily simplices). We denote with $\mathcal{E}_h^0$ the set of internal element faces, with $\mathcal{E}_h^\partial$ the set of boundary element faces and we let $\mathcal{E}_h=\mathcal{E}_h^0\cup\mathcal{E}_h^\partial$. We moreover set:

$$\Gamma_h^0\overset{\text{def}}{=}\bigcup_{e\in\mathcal{E}_h^0}e,\quad \Gamma_h^\partial\overset{\text{def}}{=}\bigcup_{e\in\mathcal{E}_h^\partial}e,\quad \Gamma_h\overset{\text{def}}{=}\Gamma_h^0\cup\Gamma_h^\partial.\tag{23}$$

The solution is approximated on $\mathcal{T}_h$ as a piecewise polynomial function possibly discontinuous on element interfaces, *i.e.* we assume the following space settings for the approximate solution $(\mathbf{u}_h,p_h,\tilde{v}_h)$:

$$\mathbf{u}_h\in\mathbf{V}_h\overset{\text{def}}{=}[V_h]^N,\quad p_h,\tilde{v}_h\in Q_h\overset{\text{def}}{=}V_h,\tag{24}$$

where, for all integers $k\geqslant1$,

$$V_h\overset{\text{def}}{=}\left\{v_h\in L^2(\Omega):v_h|_K\in\mathbb{P}_k(K)\forall K\in\mathcal{T}_h\right\}$$

being $\mathbb{P}_k(K)$ the space of polynomials of global degree at most k on the element K. To overcome the ill-conditioning of elemental mass matrices for higher-order polynomials on high aspect ratio and curved elements we have chosen a hierarchical and orthogonal set of shape functions defined in physical space. This set is obtained using a modified Gram-Schmidt procedure considering as a starting point a set of monomial functions of the same degree k. For all $e\in\mathcal{E}_h^0$, for all vector quantities $\mathbf{v}$ and scalar quantities q such that a (possibly two-valued) trace is available on e, we define:

$$[\![\mathbf{v}]\!]\overset{\text{def}}{=}\mathbf{v}^+\otimes\mathbf{n}^++\mathbf{v}^-\otimes\mathbf{n}^-,\quad [\![q]\!]\overset{\text{def}}{=}q^+\mathbf{n}^++q^-\mathbf{n}^-,\tag{25}$$

where we adopt the notation illustrated in Fig. 10. According to these definitions, $[\![\mathbf{v}]\!]$ is a tensor quantity, and $[\![q]\!]$ is a vector quantity, i.e. the jump operator always increases the tensor rank by one. Similarly, for all scalar, vector or tensor quantities such that a (possibly two-valued) trace is available on e, we introduce the average operator:

$$\{\cdot\}\overset{\text{def}}{=}\frac{(\cdot)^++(\cdot)^-}{2}.\tag{26}$$

These definitions can be extended to faces intersecting $\partial\Omega$ accounting for the weak imposition boundary conditions as discussed in Section 3.4.

The discrete counterpart of Eq. (22) for a generic element $K\in\mathcal{T}_h$ then reads:

$$\begin{aligned}
&-\int_K \nabla_h q_h\cdot\mathbf{u}_h\mathrm{d}\mathbf{x}+\int_{\partial K} q_h\mathbf{u}_h|_K\cdot\mathbf{n}\mathrm{d}\sigma=0,\\
&\int_K \mathbf{v}_h\cdot\frac{\partial\mathbf{u}_h}{\partial t}\mathrm{d}\mathbf{x}-\int_K \nabla_h\mathbf{v}_h:\mathbf{F}(\mathbf{u}_h,p_h)\mathrm{d}\mathbf{x}+\int_{\partial K}\mathbf{v}_h\otimes\mathbf{n}:\mathbf{F}(\mathbf{u}_h|_K,p_h|_K)\mathrm{d}\sigma=\mathbf{0},\\
&\int_K q_h\frac{\partial\tilde{v}_h}{\partial t}\mathrm{d}\mathbf{x}-\int_K \nabla_h q_h\cdot\mathbf{g}(\mathbf{u}_h,\tilde{v}_h)\mathrm{d}\mathbf{x}+\int_{\partial K} q_h\mathbf{g}(\mathbf{u}_h|_K,\tilde{v}_h|_K)\cdot\mathbf{n}\mathrm{d}\sigma=0.
\end{aligned}\tag{27}$$

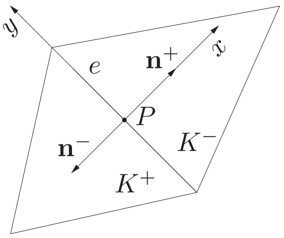


Fig. 10. Normals and local frame at quadrature point P on edge e.

To introduce a coupling between the degrees of freedom belonging to adjacent elements and to ensure conservation, we substitute the fluxes $\mathbf{u}_h|_K$, $\mathbf{F}(\mathbf{u}_h|_K, p_h|_K)$ and $\mathbf{g}(\mathbf{u}_h|_K, \tilde{\nu}_h|_K)$ with suitably defined numerical fluxes $\hat{\mathbf{u}}(\mathbf{u}_h^\pm, p_h^\pm)$, $\hat{\mathbf{F}}(\mathbf{u}_h^\pm, p_h^\pm)$ and $\hat{\mathbf{g}}(\mathbf{u}_h^\pm, p_h^\pm, \tilde{\nu}_h^\pm)$. We remark that the stability and accuracy properties of the method strongly depend on the choice of such numerical fluxes. Summing Eq. (27) over the elements we obtain the DG formulation of problem (22) which then requires to find $\mathbf{u}_h \in \mathbf{V}_h$ and $p_h, \tilde{\nu}_h \in Q_h$ such that:

$$\begin{aligned}
&-\int_{\Omega_h} \nabla_h q_h \cdot \mathbf{u}_h \mathrm{d}\mathbf{x} + \int_{\Gamma_h} [\![q_h]\!] \cdot \hat{\mathbf{u}}(\mathbf{u}_h^\pm, p_h^\pm) \mathrm{d}\sigma = 0, \\
&\int_{\Omega_h} \mathbf{v}_h \cdot \frac{\partial \mathbf{u}_h}{\partial t} \mathrm{d}\mathbf{x} - \int_{\Omega_h} \nabla_h \mathbf{v}_h : \mathbf{F}(\mathbf{u}_h, p_h) \mathrm{d}\mathbf{x} + \int_{\Gamma_h} [\![\mathbf{v}_h]\!] : \hat{\mathbf{F}}(\mathbf{u}_h^\pm, p_h^\pm) \mathrm{d}\sigma = \mathbf{0}, \\
&\int_{\Omega_h} q_h \frac{\partial \tilde{\nu}_h}{\partial t} \mathrm{d}\mathbf{x} - \int_{\Omega_h} \nabla_h q_h \cdot \mathbf{g}(\mathbf{u}_h, \tilde{\nu}_h) \mathrm{d}\mathbf{x} + \int_{\Gamma_h} [\![q_h]\!] \cdot \hat{\mathbf{g}}(\mathbf{u}_h^\pm, p_h^\pm, \tilde{\nu}_h^\pm) \mathrm{d}\sigma = 0
\end{aligned} \tag{28}$$

for all $\mathbf{v}_h \in \mathbf{V}_h$ and $q_h \in Q_h$. The key idea to devise $\hat{\mathbf{u}}$, $\hat{\mathbf{F}}$ and $\hat{\mathbf{g}}$ is to reduce the problem of flux computation to the solution of a planar Riemann problem as in the compressible case. In order to recover the hyperbolic character of the equations, the incompressibility constraint is relaxed by adding an artificial compressibility term to the continuity equation. At each quadrature point P on Γ_h we therefore solve the Riemann problem for the equations:

$$\begin{aligned}
&\frac{1}{c^2}\frac{\partial p}{\partial t} + \frac{\partial u}{\partial x} = 0, \\
&\frac{\partial u}{\partial t} + \frac{\partial (u^2 + p)}{\partial x} = 0, \\
&\frac{\partial v}{\partial t} + \frac{\partial u v}{\partial x} = 0, \\
&\frac{\partial \tilde{\nu}}{\partial t} + \frac{\partial u \tilde{\nu}}{\partial x} = 0,
\end{aligned} \tag{29}$$

with initial datum

$$(\mathbf{u}, p, \tilde{\nu}) = \begin{cases} (\mathbf{u}_h^-, p_h^-, \tilde{\nu}_h^-) & \text{if } x < 0, \\ (\mathbf{u}_h^+, p_h^+, \tilde{\nu}_h^+) & \text{if } x > 0, \end{cases}$$

where x denotes a locally defined axis oriented as the normal vector $\mathbf{n}^+$ pointing out of K^+ and located in such a way that $x = 0$ at P (see Fig. 10).

Denoting with $(\mathbf{u}_*, p_*, \tilde{\nu}_*)$ the solution of the Riemann problem on the space–time line $x/t = 0$, we finally set:

$$\hat{\mathbf{u}}(\mathbf{u}_h^\pm, p_h^\pm) = \mathbf{u}_*, \quad \hat{\mathbf{F}}(\mathbf{u}_h^\pm, p_h^\pm) = \mathbf{F}(\mathbf{u}_*, p_*), \quad \hat{\mathbf{g}}(\mathbf{u}_h^\pm, p_h^\pm, \tilde{\nu}_h^\pm) = \mathbf{g}(\mathbf{u}_*, \tilde{\nu}_*).$$

The details of the procedure adopted to determine the state $(\mathbf{u}_*, p_*, \tilde{\nu}_*)$ are thoroughly discussed in the Appendix A of [18]. Here we remark that the transported scalar quantities, like turbulent viscosity but also tangential velocity components or temperature (if the energy equation is used), can be computed according to the relation

$$\phi_* = \begin{cases} \phi^+ f(\mathbf{u}_h^\pm \cdot \mathbf{n}^+, p_h^\pm) & \mathbf{u}_* \cdot \mathbf{n}^+ \geqslant 0, \\ \phi^- f(\mathbf{u}_h^\pm \cdot \mathbf{n}^+, p_h^\pm) & \mathbf{u}_* \cdot \mathbf{n}^+ < 0, \end{cases}$$

where ϕ stands for any transport quantities and the function f, depending on the structure of the Riemann, is the same for all the transported quantities.

3.2. Diffusive terms

Many techniques are available for the DG space discretization of the diffusive terms: a complete survey can be found in [26]. In this paper we choose the form, first proposed in [2], which gives the following space discretization of the complete governing equations: find $\mathbf{u}_h \in \mathbf{V}_h$ and $p_h, \tilde{\nu}_h \in Q_h$ such that

$$\begin{aligned}
&-\int_{\Omega_h} \nabla_h q_h \cdot \mathbf{u}_h \mathrm{d}\mathbf{x} + \int_{\Gamma_h} [\![q_h]\!] \cdot \hat{\mathbf{u}}(\mathbf{u}_h^{\pm}, p_h^{\pm}) \mathrm{d}\sigma = 0,\\
&\int_{\Omega_h} \mathbf{v}_h \cdot \frac{\partial \mathbf{u}_h}{\partial t} \mathrm{d}\mathbf{x} - \int_{\Omega_h} \nabla_h \mathbf{v}_h : (\mathbf{F}_\nu(\nabla_h \mathbf{u}_h, \mathbf{r}([\![\mathbf{u}_h]\!]), \tilde{\nu}_h) + \mathbf{F}(\mathbf{u}_h, p_h)) \mathrm{d}\mathbf{x} + \int_{\Gamma_h} [\![\mathbf{v}_h]\!] : \left(\hat{\mathbf{F}}_\nu(\nabla_h \mathbf{u}_h^{\pm}, \mathbf{r}_e^{\pm}([\![\mathbf{u}_h]\!]), \tilde{\nu}_h^{\pm}) + \hat{\mathbf{F}}(\mathbf{u}_h^{\pm}, p_h^{\pm})\right) \mathrm{d}\sigma = \mathbf{0},\\
&\int_{\Omega_h} q_h \frac{\partial \tilde{\nu}_h}{\partial t} \mathrm{d}\mathbf{x} - \int_{\Omega_h} \nabla_h q_h \cdot (\mathbf{g}_\nu(\tilde{\nu}_h, \nabla_h \tilde{\nu}_h, \mathbf{r}([\![\tilde{\nu}_h]\!])) + \mathbf{g}(\mathbf{u}_h, \tilde{\nu}_h)) \mathrm{d}\mathbf{x} + \int_{\Gamma_h} [\![q_h]\!] \cdot (\hat{\mathbf{g}}_\nu(\tilde{\nu}_h^{\pm}, \nabla_h \tilde{\nu}_h^{\pm}, \mathbf{r}_e^{\pm}([\![\tilde{\nu}_h]\!])) + \hat{\mathbf{g}}(\mathbf{u}_h^{\pm}, p_h^{\pm}, \tilde{\nu}_h^{\pm})) \mathrm{d}\sigma = 0
\end{aligned} \tag{30}$$

for all $\mathbf{v}_h \in \mathbf{V}_h$ and $q_h \in Q_h$. The diffusive fluxes in the above equations are defined as:

$$\begin{aligned}
&\mathbf{F}_\nu(\nabla_h \mathbf{u}_h, \mathbf{r}([\![\mathbf{u}_h]\!]), \tilde{\nu}_h) \overset{\text{def}}{=} -(\nu + \nu_t(\tilde{\nu}_h))\left((\nabla_h \mathbf{u}_h + \nabla_h \mathbf{u}_h^T) + \left(\mathbf{r}([\![\mathbf{u}_h]\!]) + \mathbf{r}([\![\mathbf{u}_h]\!])^T\right)\right),\\
&\hat{\mathbf{F}}_\nu(\nabla_h \mathbf{u}_h^{\pm}, \mathbf{r}_e^{\pm}([\![\mathbf{u}_h]\!]), \tilde{\nu}_h^{\pm}) \overset{\text{def}}{=} -\left\{(\nu + \nu_t(\tilde{\nu}_h))\left((\nabla_h \mathbf{u}_h + \nabla_h \mathbf{u}_h^T) - \eta_e\left(\mathbf{r}_e([\![\mathbf{u}_h]\!]) + \mathbf{r}_e([\![\mathbf{u}_h]\!])^T\right)\right)\right\},\\
&\mathbf{g}_\nu(\tilde{\nu}_h, \nabla_h \tilde{\nu}_h, \mathbf{r}([\![\tilde{\nu}_h]\!])) \overset{\text{def}}{=} -\bar{\nu}_h(\nabla_h \tilde{\nu}_h + \mathbf{r}([\![\tilde{\nu}_h]\!])),\\
&\hat{\mathbf{g}}_\nu(\tilde{\nu}_h^{\pm}, \nabla_h \tilde{\nu}_h^{\pm}, \mathbf{r}_e^{\pm}([\![\tilde{\nu}_h]\!])) \overset{\text{def}}{=} -\{\xi_h(\nabla_h \tilde{\nu}_h + \eta_e \mathbf{r}_e([\![\tilde{\nu}_h]\!]))\},
\end{aligned} \tag{31}$$

where ξ_h is defined according to Eq. (9) or Eq. (14). For all $e \in \mathcal{E}_h$ and for all $\psi \in [L^2(e)]^{N^2}$ the lifting operator $\mathbf{r}_e$ is defined as the solution of the following problem:

$$\int_{\Omega_h} \mathbf{r}_e(\psi) : \tau_h \mathrm{d}\mathbf{x} = -\int_e \psi : \{\tau_h\} \mathrm{d}\sigma, \quad \forall \tau_h \in [V_h]^{N^2}. \tag{32}$$

Similarly, for all $\mathbf{w} \in [L^2(e)]^N$,

$$\int_{\Omega_h} \mathbf{r}_e(\mathbf{w}) \cdot \mathbf{v}_h \mathrm{d}\mathbf{x} = -\int_e \mathbf{w} \cdot \{\mathbf{v}_h\} \mathrm{d}\sigma, \quad \forall \mathbf{v}_h \in [V_h]^N. \tag{33}$$

Finally,

$$\mathbf{r}(\cdot) \overset{\text{def}}{=} \sum_{e \in \mathcal{E}_h} \mathbf{r}_e(\cdot). \tag{34}$$

It is possible to find lower bounds for the parameter $\eta_e \in \mathbb{R}^+$ ensuring stability of the method [26].

3.3. Source terms

To complete the DG spatial discretization a source term is added to the right hand side of the equation for $\tilde{\nu}_h$ in the system (30):

$$\begin{aligned}
&-\int_{\Omega_h} \nabla_h q_h \cdot \mathbf{u}_h \mathrm{d}\mathbf{x} + \int_{\Gamma_h} [\![q_h]\!] \cdot \hat{\mathbf{u}}(\mathbf{u}_h^{\pm}, p_h^{\pm}) \mathrm{d}\sigma = 0,\\
&\int_{\Omega_h} \mathbf{v}_h \cdot \frac{\partial \mathbf{u}_h}{\partial t} \mathrm{d}\mathbf{x} - \int_{\Omega_h} \nabla_h \mathbf{v}_h : (\mathbf{F}_\nu(\nabla_h \mathbf{u}_h, \mathbf{r}([\![\mathbf{u}_h]\!]), \tilde{\nu}_h) + \mathbf{F}(\mathbf{u}_h, p_h)) \mathrm{d}\mathbf{x} + \int_{\Gamma_h} [\![\mathbf{v}_h]\!] : \left(\hat{\mathbf{F}}_\nu(\nabla_h \mathbf{u}_h^{\pm}, \mathbf{r}_e^{\pm}([\![\mathbf{u}_h]\!]), \tilde{\nu}_h^{\pm}) + \hat{\mathbf{F}}(\mathbf{u}_h^{\pm}, p_h^{\pm})\right) \mathrm{d}\sigma = \mathbf{0},\\
&\int_{\Omega_h} q_h \frac{\partial \tilde{\nu}_h}{\partial t} \mathrm{d}\mathbf{x} - \int_{\Omega_h} \nabla_h q_h \cdot (\mathbf{g}_\nu(\tilde{\nu}_h, \nabla_h \tilde{\nu}_h, \mathbf{r}([\![\tilde{\nu}_h]\!])) + \mathbf{g}(\mathbf{u}_h, \tilde{\nu}_h)) \mathrm{d}\mathbf{x} + \int_{\Gamma_h} [\![q_h]\!] \cdot (\hat{\mathbf{g}}_\nu(\tilde{\nu}_h^{\pm}, \nabla_h \tilde{\nu}_h^{\pm}, \mathbf{r}_e^{\pm}([\![\tilde{\nu}_h]\!])) + \hat{\mathbf{g}}(\mathbf{u}_h^{\pm}, p_h^{\pm}, \tilde{\nu}_h^{\pm})) \mathrm{d}\sigma\\
&= \int_{\Omega_h} q_h s_h(\nabla_h \mathbf{u}_h + \mathbf{r}([\![\mathbf{u}_h]\!]), \tilde{\nu}_h, \nabla_h \tilde{\nu}_h + \mathbf{r}([\![\tilde{\nu}_h]\!])) \mathrm{d}\mathbf{x},
\end{aligned} \tag{35}$$

where the term s_h is defined according to Eq. (13), for the Allmaras approach, or to Eq. (15), for the Bassi one, and according to the new r definition, Eq. (18). All integrals appearing in the space discrete problem (35) are computed by means of Gauss integration rules with a number of integration points suited for the required accuracy. The theoretical degree of integration must account for the degrees of (i) polynomial approximation, (ii) mapping, and (iii) Jacobian of the mapping. If available, cheaper non-product formulae, taken from [27], are preferred to tensor product ones.

3.4. Boundary conditions

The DG discretization is best suited for a weak enforcement of boundary conditions. This can be easily achieved by properly defining a boundary state which, together with the internal state, allows to compute the numerical fluxes and the lifting operator on the portion Γ_h^{∂} of the boundary Γ_h.

For Dirichlet type boundary data this can be easily achieved by properly defining a boundary state $\mathbf{u}_h^b,\ p_h^b$ and $\tilde{\nu}_h^b$ which, together with the internal state $\mathbf{u}_h^+$, p_h^+ and $\tilde{\nu}_h^+$ is suitable for the evaluation of the numerical fluxes $\hat{\mathbf{u}},\ \hat{\mathbf{F}},\ \hat{\mathbf{g}}$ and the function $\mathbf{r}_e^+$. Moreover the averages $\{\nabla_h \mathbf{u}_h\}, \{\mathbf{r}_e([\![\mathbf{u}_h]\!])\}, \{\nabla_h \tilde{\nu}_h\}$ and $\{\mathbf{r}_e([\![\tilde{\nu}_h]\!])\}$ are set in $\hat{\mathbf{F}}_v$ and $\hat{\mathbf{g}}_v$ equal to the internal values. The wall-type boundary conditions have been implemented by defining the boundary state on the exterior of boundary faces as $\mathbf{u}_h^b = -\mathbf{u}_h^+,\ \nabla\mathbf{u}_h^b = \nabla\mathbf{u}_h^+,\ p_h^b = p_h^+,\ \tilde{\nu}_h^b = -\tilde{\nu}_h^+$ and $\nabla\tilde{\nu}_h^b = \nabla\tilde{\nu}_h^+$. In this case the external boundary state exactly replaces $\mathbf{u}_h^-, \tilde{\nu}_h^-$ and p_h^- in the jump operators, in the numerical fluxes and in the lifting operator.

4. Time discretization and linear system solution

The discrete problem corresponding to Eq. (35) can be written as:

$$\mathbf{M}\frac{\mathrm{d}\mathbf{U}}{\mathrm{d}t} + \mathbf{R}(\mathbf{U}) = \mathbf{0}, \tag{36}$$

where $\mathbf{U}$ is the global vector of unknowns and $\mathbf{M}$ is the global block diagonal mass matrix. Eq. (36) defines a system of non-linear ODEs which is discretized by means of linearly implicit (Rosenbrock-type) Runge–Kutta schemes, see [20] for time accurate unsteady Navier–Stokes computations. In this work only steady problems are considered hence only the first order backward Euler scheme was adopted:

$$\left(\frac{\mathbf{M}}{\Delta t} + \mathbf{J}\right)\Delta\mathbf{U} = -\mathbf{R}(\mathbf{U}^n), \tag{37}$$

where $\mathbf{J} = \partial\mathbf{R}(\mathbf{U}^n)/\partial\mathbf{U}$ is the Jacobian matrix of the DG space discretization. Eq. (37) requires to solve a linear system of the form $\mathbf{Ax} + \mathbf{b} = \mathbf{0}$ in which the matrix $\mathbf{A}$ can be regarded as an $N_K \times N_K$ block sparse matrix where N_K is the number of elements in $\mathcal{T}_h$ and the rank of each block is $N_{DOF}^K \times (N+2)$, being N_{DOF}^K the number of degrees of freedom for each variable in the generic element K. Thanks to the DG discretization here adopted, the degrees of freedom of a generic element K are only coupled with those of the neighbouring elements and the number of nonzero blocks for each (block) row K of the matrix $\mathbf{A}$ is therefore equal to the number of elements surrounding the element K plus one. The Jacobian matrix of the DG discretization is computed analytically and, using very large time steps, the method can therefore achieve quadratic convergence in the computation of steady state solutions. In the limit $\Delta t \to \infty$ Eq. (37) is in fact identical to one iteration of the Newton method applied to the steady discrete problem. Thus a fully implicit Newton–Krylov algorithm is accomplished with a pseudo-transient continuation strategy. The growth of the local CFL number is related to the residual vector L_2 and L_∞ norms according to the following relations:

$$\begin{aligned} \mathrm{CFL} &= \min\left(\frac{\mathrm{CFL}_{\min}}{f^{\beta}}, \mathrm{CFL}_{\max}\right) \\ f &= \max\left(\frac{\|\mathbf{R}(\mathbf{U}^n)\|_{L_2}}{\|\mathbf{R}\left(\mathbf{U}^0\right)\|_{L_2}}, \frac{\|\mathbf{R}(\mathbf{U}^n)\|_{L_\infty}}{\|\mathbf{R}\left(\mathbf{U}^0\right)\|_{L_\infty}}\right), \end{aligned} \tag{38}$$

where $\mathrm{CFL}_{\min}$, $\mathrm{CFL}_{\max}$ and β are user-defined input parameters. This means that, since Δt is linearly proportional to the CFL value, during the computation the time step size continuously grows with the decreasing of the residuals norms. A remarkable consequence is the achievement of a quadratic convergence at least in the last part of the solution process when, being the residuals small, large time steps are employed.

If a local time derivative of pressure is introduced in the continuity equation, the matrix condition number can be significantly decreased, as well as the number of iterations required to achieve a prescribed residuals tolerance in the Krylov linear solver. This is particularly relevant during the first pseudo time steps when, being the residuals large, the Δt is small. Thus the idea is to solve

$$\int_{\Omega_h} q_h \frac{1}{a^2}\frac{\partial p_h}{\partial t}\mathrm{d}\mathbf{x} - \int_{\Omega_h} \nabla_h q_h \cdot \mathbf{u}_h \mathrm{d}\mathbf{x} + \int_{\Gamma_h} [\![q_h]\!] \cdot \hat{\mathbf{u}}(\mathbf{u}_h^{\pm}, p_h^{\pm})\mathrm{d}\sigma = 0, \tag{39}$$

instead of the first equation of the system of Eq. (35). Even if Eq. (39) has exactly the same form of a standard artificial compressibility approach, in our algorithm the coefficient a^2 can safely differs from the artificial compressibility coefficient, c^2, adopted in the Riemann solver, Eq. (29). Eqs. (39) and (38) merely define a globalisation strategy of the Newton–Krylov algorithm preventing its lack of robustness and allowing to efficiently seek the solution of the steady state problem of Eq. (35). This solution algorithm enjoys the flexibility of the two compressibility coefficients setting: the one related to the convective flux terms can be fixed to a small value, in [18] it is shown that is preferable for both the accuracy and the Jacobian condition number, while the second to a larger value, resembling the incompressible constrain which implies $a \to \infty$. Note that in the original global mass matrix **M**, since the time derivative of pressure does not appear in the governing equations, the block corresponding to the pressure degrees of freedom are null while this is not the case with the application of Eq. (39).

To solve Eq. (37) we resort to the preconditioned GMRES (Generalized Minimal RESidual) linear solvers available in the PETSc library, [28], the software upon which our DG code relies for the purpose of parallelization.

The results here reported are obtained with the standard algorithm, nevertheless even a matrix-free version [29], in which the Jacobian matrix **J** is replaced by its first order finite difference approximation ensuring a significant memory saving, has been already verified.

The parallelization is based on grid partitioning accomplished by means of the METIS package, [30]. Each processor owns the data related to its local portion of the grid and the data on remote processors are accessed through MPI, the standard for message-passing communication. Thanks to the compactness of our DG method only the data owned by the near neighbour elements at partition boundaries need to be shared among different processes. An effective choice for the preconditioner is the incomplete lower–upper factorization, ILU(0), of the analytically computed matrix **A**. However, since the global ILU(0) is not a highly scalable algorithm, the block Jacobi (BJ) method, with one block per process each of which is solved with ILU(0), is often used in the parallel computing context. Moreover several computations have shown that the Additive Schwarz Method (ASM), for which the local block is enlarged with the matrix entries corresponding to one or more strips of overlapping elements, is more effective than the BJ method.

5. Numerical results

In this section we present high-order DG solutions of steady problems in order to assess the reliability and robustness as well as the accuracy of the proposed SA model implementation. If not stated otherwise, the following results have been obtained using the Allmaras negative $\tilde{\nu}$ corrections and the new r definition. The validation cases here considered are included in the proposals of the Turbulence Model Benchmarking Working Group, a subcommittee of the American Institute of Aeronautics and Astronautics (AIAA) Fluid Dynamics Technical Committee [31].

The main aim of the first test case, the flow over a flat plate, was to verify the accuracy of our DG implementation while the second and the third test cases, the flow past a backward-facing step and around a NACA 0012 airfoil, have been chosen to assess the solver performance in dealing with large flow separation zones and with highly stretched and curved elements, respectively.

For all the computations here reported the freestream turbulent viscosity has been set to $\tilde{\nu}_\infty = 10^{-3}\nu$. This value is considered perfectly acceptable, see for example [22,32], but, unlike the ideal condition $\tilde{\nu}_\infty = 0$, prevents the occurrence of an apparent transition, see Section 5.1.

The initial flow field of all the $\mathbb{P}_1$ solutions has been set to uniform freestream conditions, with $a = 10$ in Eq. (39) and $\mathrm{CFL_{min}} = 10^{-1} \div 1$, $\mathrm{CFL_{max}} = 10^{20}$, $\beta = 1$ in Eq. (38). For computational efficiency, each converged lower-order solution has been taken as the initial flow field of the next higher-order solution. Thanks to the good initial field approximation, the $\mathrm{CFL_{min}}$ parameter could be gradually increased up to 10^2, rising the degree of polynomial approximation.

The artificial compressibility parameter of the Riemann solver has been set in the range of 10^{-2} to 10^{-1} for all the cases considered. However, as can be seen in Section 5.3, the results were not found to be significantly affected by the c^2 value.

The parameters of the restarted GMRES solver have been set to 120 Krylov space vectors, 240 maximum iterations and 10^{-5} relative convergence tolerance. For parallel computations, a one-level Additive Schwarz preconditioner has been used. The above parameters have been chosen in order to avoid any issues related to under-solving the linear systems. However, we have already verified that all the test cases here reported could also be computed with less expensive GMRES settings.

All the solutions have been computed on a Linux Cluster with 8 Opteron nodes for a total of 64 cores operating at 2.3 GHz.

5.1. High Reynolds number flow over a flat plate

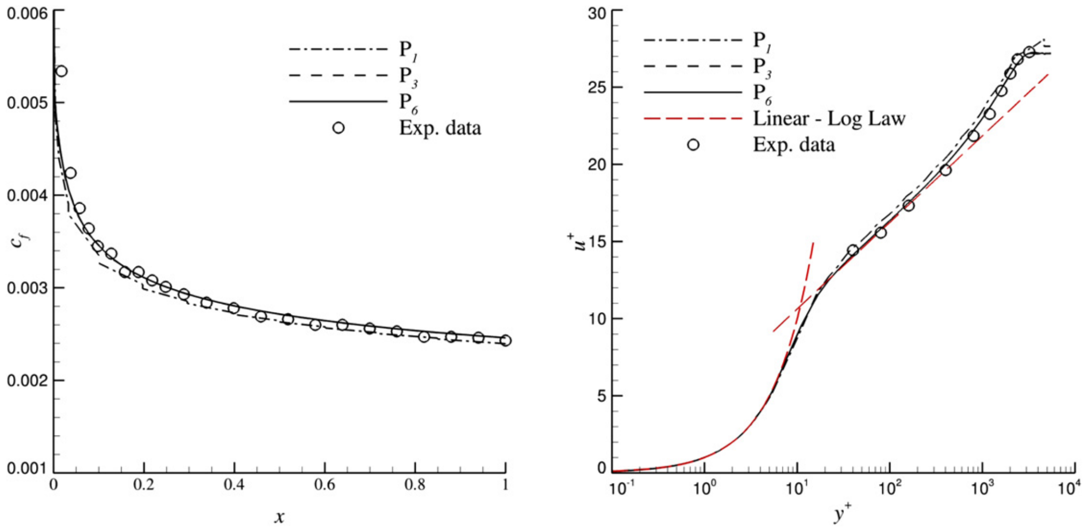
The first test case is the flow over a smooth flat plate with Reynolds number equal to 10^7 based on the plate length. The flow was experimentally investigated in [33] and included in the 1969 ASOFR-IFP Stanford Conference [34]. It was also used as a validation case of CFD codes by the NPARC Alliance [35].

The solution has been computed up to $\mathbb{P}_6$ polynomial approximation on two sets of quadrilateral grids suitably refined at the wall and around the leading edge. The first set has been used to perform a convergence study and consists of five nested grids with a number of elements in the range of $n_e = 96$ to $n_e = 24576$. The second set has been used to investigate the effect of grid spacing close to the wall and includes three grids with $n_e = 384$ and different height of the elements next to the wall.

The computed skin friction coefficients are shown in Figs. (a)(a)11–13(a) for increasingly higher values of the non-dimensional centroid distance $y_c^+ = y_c u_\tau/\nu$ of the elements next to the wall. The viscous stress τ_w tangential to the wall, needed to compute the friction velocity $u_\tau = \sqrt{\tau_w/\rho}$, has been evaluated taking into account the lifting contribution of the velocity jump to the velocity gradient internal to the elements, according to the equation

$$\boldsymbol{\tau} = \rho(\nu + \nu_t(\tilde{\nu}_h))[(\nabla_h \mathbf{u}_h + \mathbf{r}([\![\mathbf{u}_h]\!])) + (\nabla_h \mathbf{u}_h^T + \mathbf{r}([\![\mathbf{u}_h]\!])^T].$$

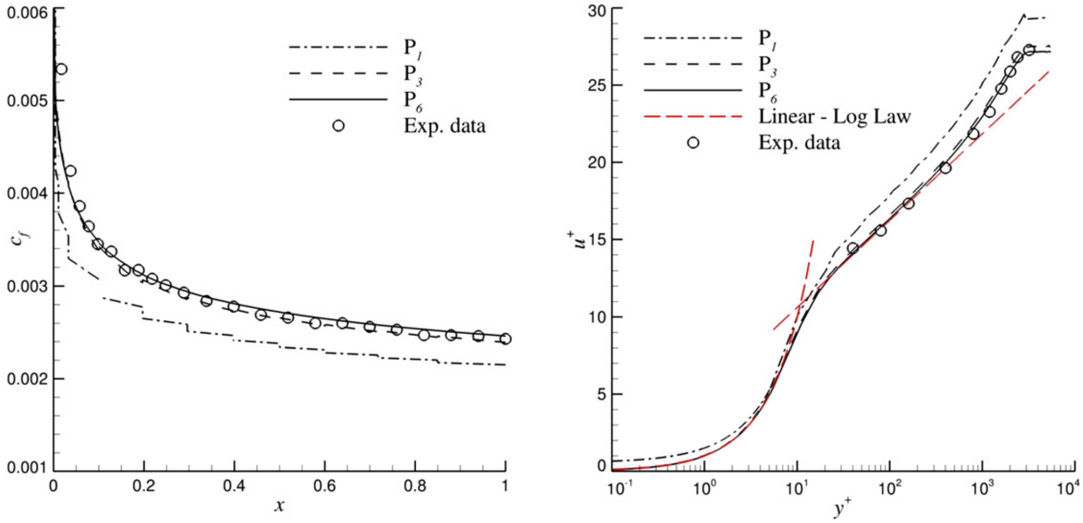
These Figures show that using $\mathbb{P}_3$ polynomial approximation the computed skin friction distributions look quite accurate on the grid with near-wall spacing as high as $O(y_c^+) \approx 10$.



(a) Skin friction coefficient.

(b) Velocity profile.

Fig. 11. Flat plate, $Re = 10^7$. Grid with $O(y_c^+) \approx 1$ and $ne = 384$.



(a) Skin friction coefficient.

(b) Velocity profile.

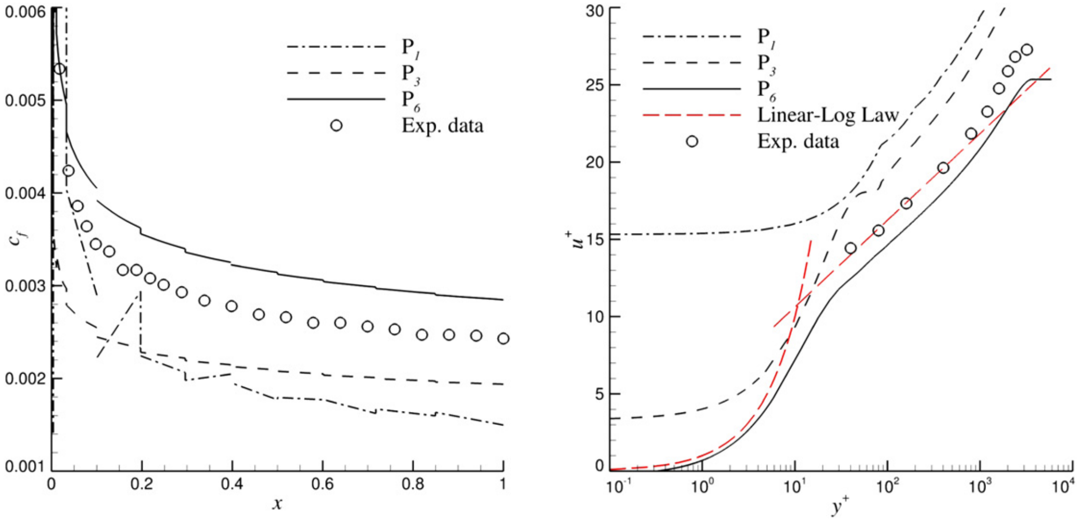
Fig. 12. Flat plate, $Re = 10^7$. Grid with $O(y_c^+) \approx 10$ and $ne = 384$.

The near-wall behaviour of the numerical solutions is presented in terms of dimensionless velocity profiles $u^+ = u/u_\tau = f(y^+)$ for $Re_x = 5 \times 10^6$. The velocity profiles are compared with experimental data and with the theoretical profiles in the viscous sub-layer and in the log-law region. Similarly to skin friction distributions, Figs. 11(b), 12(b) and 13(b) show that high-order $\mathbb{P}_3$ solutions agree very well with the experimental data on the grid with $O(y_c^+) \approx 10$ near-wall spacing. Higher degree polynomial approximations allow to obtain pretty accurate solutions even on very coarse meshes with only $n_e = 24$ elements, see Fig. 14(a).

For the purpose of studying the convergence of the method, the computed drag coefficients have been compared with the value of the most accurate $\mathbb{P}_6$ solution on the finest grid with $n_e = 24576$ elements. Assuming such value as the "exact" one, Fig. 14(b) shows that the k theoretical order of convergence has been achieved indeed.

A typical convergence history of residuals using up to $\mathbb{P}_6$ polynomial approximations is displayed in Fig. 15(a). For this simple test case the two negative $\tilde{\nu}$ modifications and r formulations presented above provide quite similar convergence histories and numerical results.

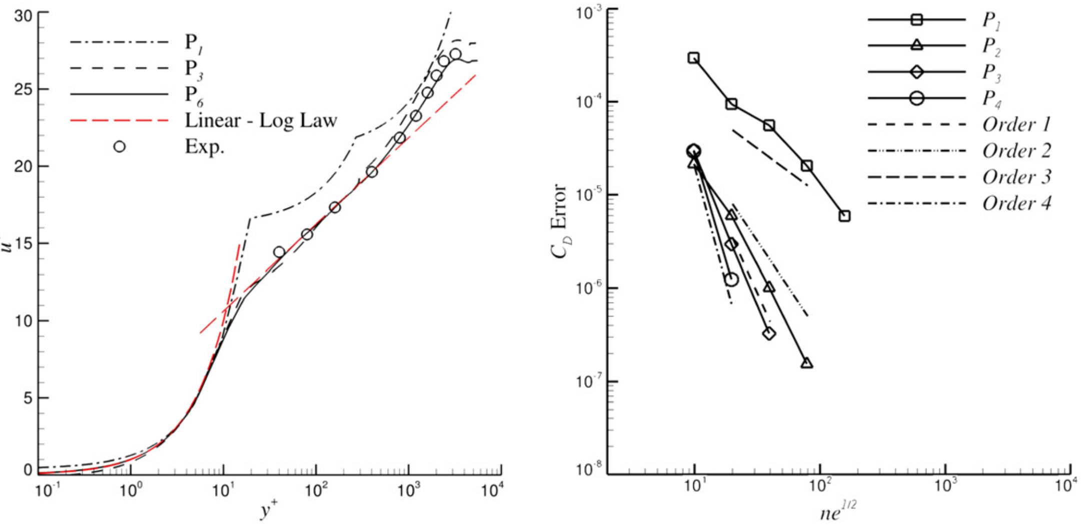
Finally, a remark on the boundary condition for $\tilde{\nu}_\infty$. Employing the ideal value $\tilde{\nu}_\infty = 0$, an apparent transition, that has no relationship with any physics intentionally built in the SA model, has been observed in the solutions. This behaviour was



(a) Skin friction coefficient.

(b) Velocity profile.

Fig. 13. Flat plate, $Re = 10^7$. Grid with $O(y_c^+) \approx 100$ and $ne = 384$.



(a) $n_e = 24$, $O(y_c^+) \approx 1$.

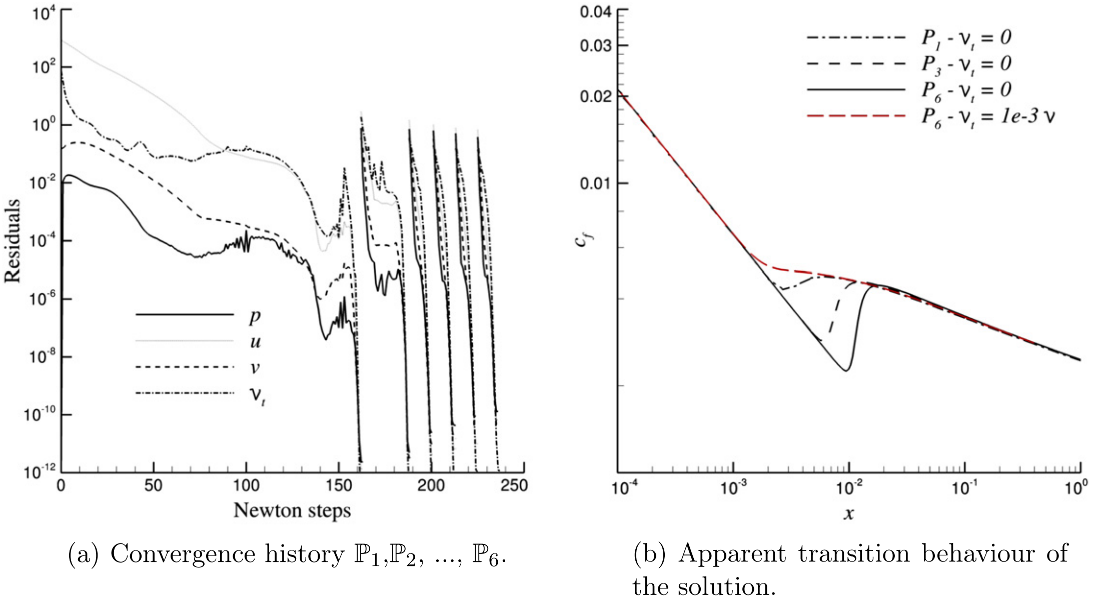
(b) Convergence study.

Fig. 14. Flat Plate, $Re = 10^7$. Solution accuracy.

already investigated by Rumsey [36] and, as found by this author, we have verified that more accurate numerical solutions, either on refined grids or with higher degree of polynomial approximation, produce larger ‘‘laminar regions’’, see Fig. 15(b). In our results this artificial transition appears even if the trip term, considered by Rumsey the main responsible for this undesirable behaviour, is missing, being it part of the transitional model, here not considered. However, the value $\tilde{\nu}_\infty = 10^{-3}\nu$, much smaller than $\tilde{\nu}_\infty = 3\nu$ suggested by Rumsey as a cure for the full model, was sufficient to avoid this numerical artifact. According to this observation, the following computations have been run with the freestream condition $\tilde{\nu}_\infty = 10^{-3}\nu$.

5.2. Turbulent flow past a backward-facing step

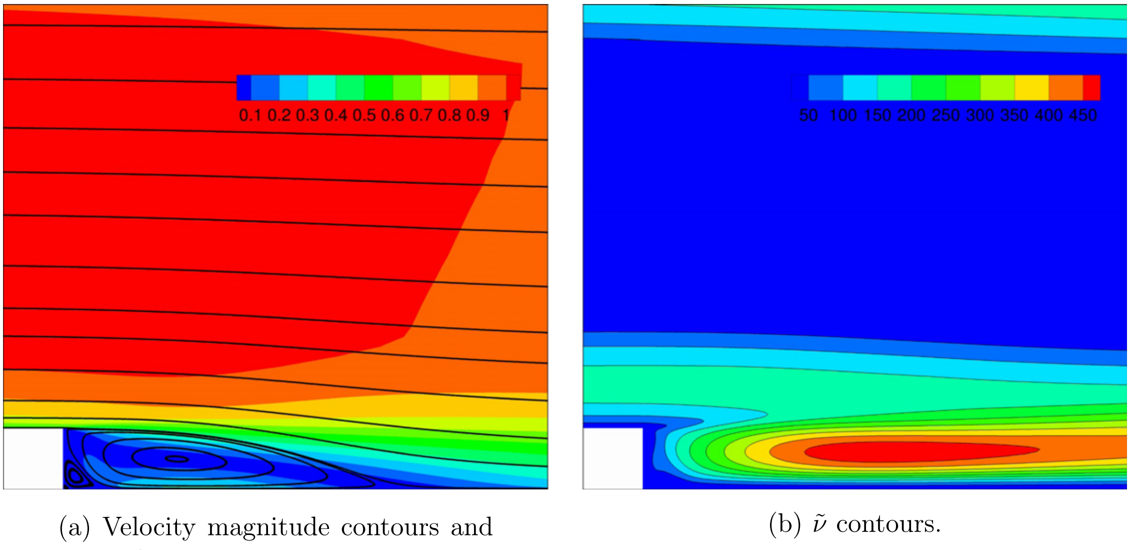
The second test case considered in this work is the flow past a backward-facing step. This is a quite popular test case for studying the physics of separating and reattaching turbulent flows and for assessing the capabilities of turbulence models and numerical methods. Here we have considered the experimental setup of [37] with step height H, inlet channel height $8H$ and Reynolds number based on the step height $Re = 37400$. The flow has been computed on a set of five Cartesian grids



(a) Convergence history $\mathbb{P}_1$,$\mathbb{P}_2$, ..., $\mathbb{P}_6$.

(b) Apparent transition behaviour of the solution.

Fig. 15. Flat plate, $Re = 10^7$.



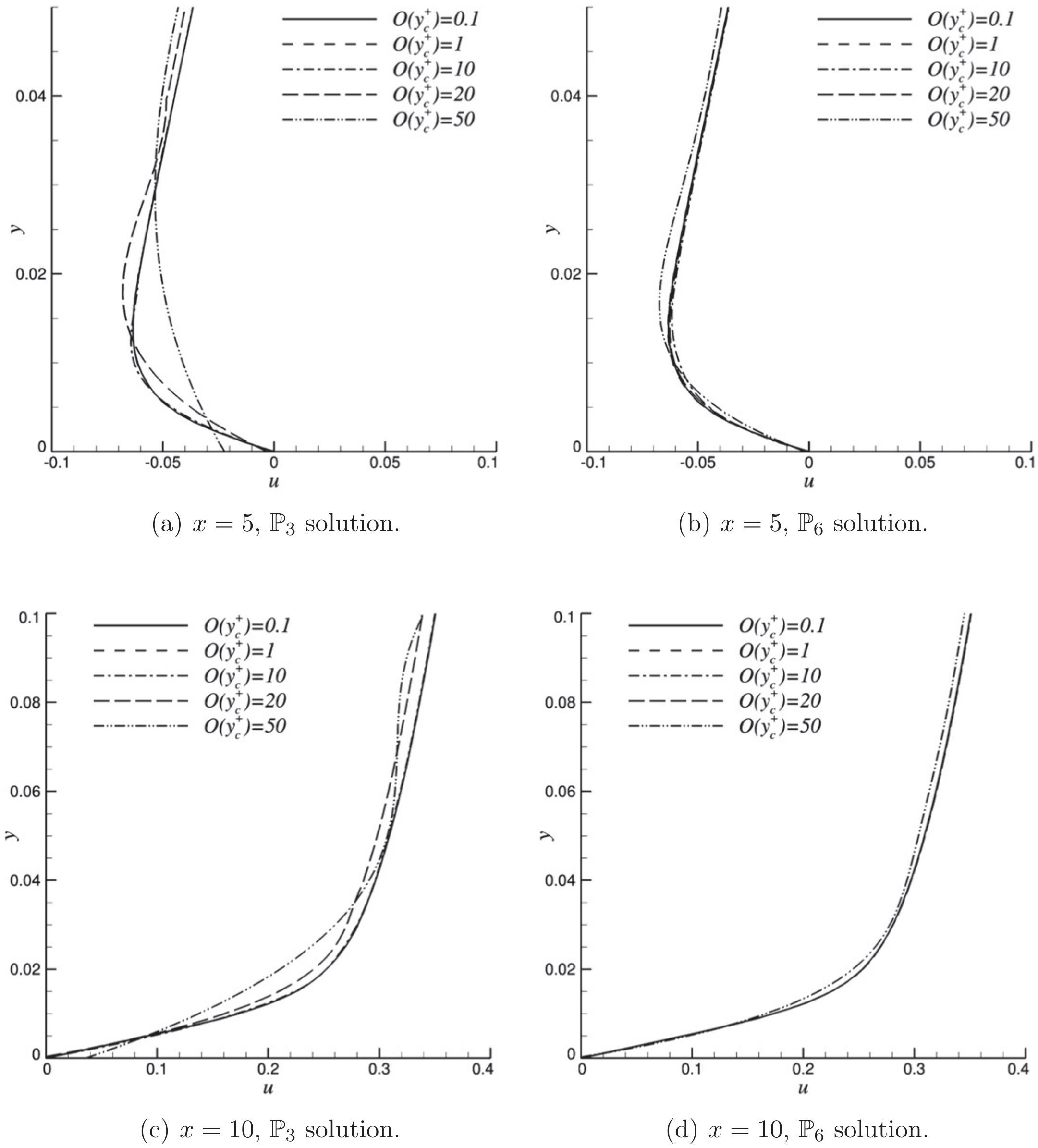
(a) Velocity magnitude contours and streamlines.

(b) $\tilde{\nu}$ contours.

Fig. 16. Backward-facing step case, $Re = 37400$. $\mathbb{P}_6$ solution. Grid with $O(y_c^+) \approx 0.1$.

with 2800 elements, grid lines clustered at the walls and around the separation point and y_c^+ first cell height values of the order of 0.1, 1, 10, 20 and 50. The example of velocity contours and streamlines of a high-order solution, shown in Fig. 16, highlights the capability of the method to capture the secondary vortex on the bottom of the step. As was the case for the flat plate problem, high-order (at least $\mathbb{P}_3$) solutions on the grid with $O(y_c^+) \approx 10$ near-wall grid spacing already provide almost converged results, see Fig. 17.

Figs. 18 and 19(a) compare the DG results with the experimental velocity profiles and skin friction coefficients reported in [37]. Velocity profiles, obtained using the $O(y_c^+) \approx 0.1$ mesh, at some dimensionless x values match well, especially in the reattached zone, as well as the skin friction coefficient along the bottom wall. Fig. 19(b) shows the $\tilde{\nu}$ profile at $x = 32$. It is clear that almost indistinguishable solutions are achieved with the two negative $\tilde{\nu}$ modifications here implemented. Little differences can be observed only in the region where $\tilde{\nu}$ is close to zero, with the Allmaras approach ensuring smaller oscillations amplitude around the freestream value.



(a) $x = 5$, $\mathbb{P}_3$ solution.

(b) $x = 5$, $\mathbb{P}_6$ solution.

(c) $x = 10$, $\mathbb{P}_3$ solution.

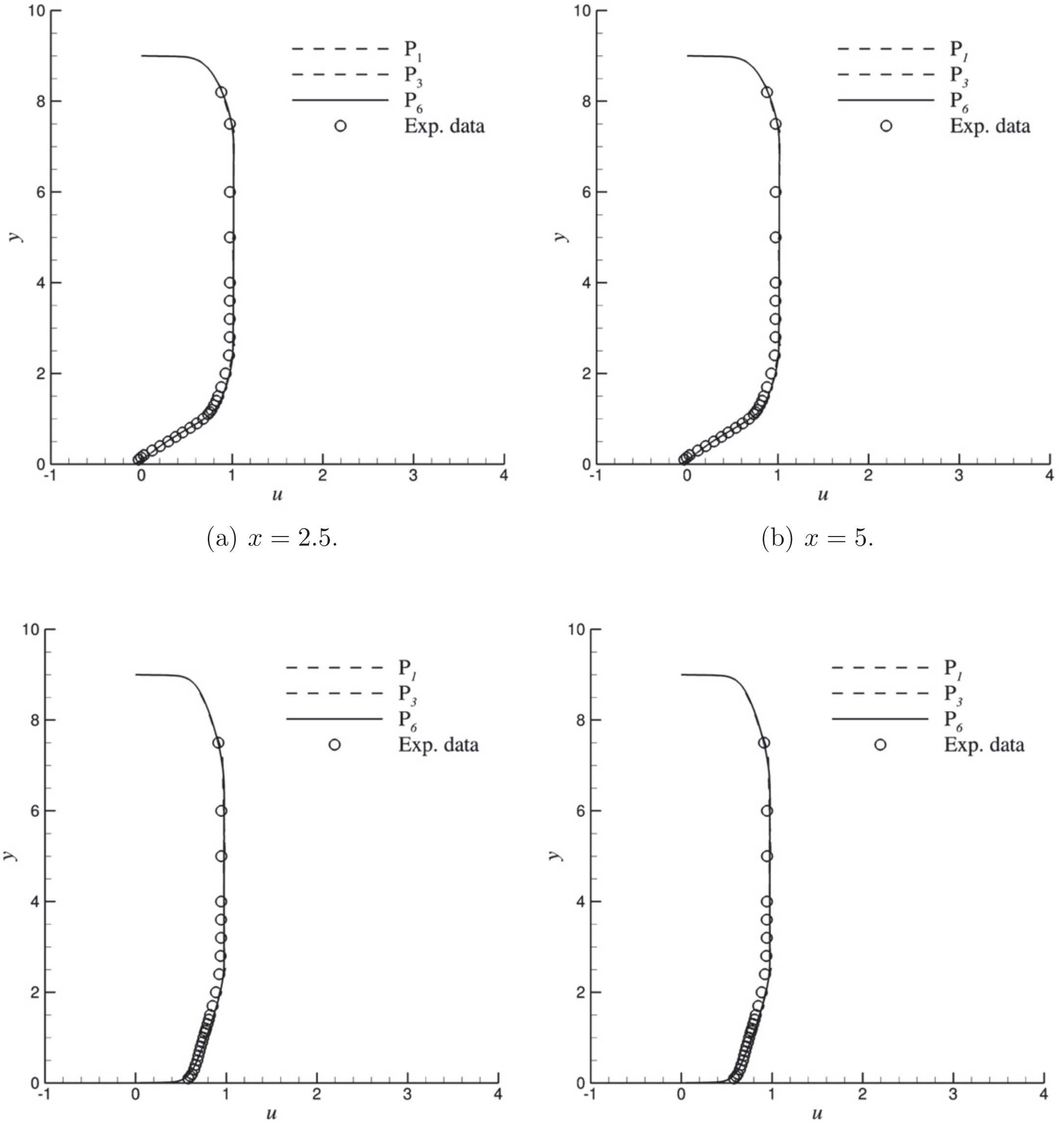
(d) $x = 10$, $\mathbb{P}_6$ solution.

Fig. 17. Backward-facing step, $Re = 37400$. Numerical solutions for different near-wall grid spacing.

Considering the predicting capabilities of the SA model on massively separated flow fields, the reattachment length x_r, is a sensitive parameter that has been historically used to asses the overall performance of turbulence models. In Table 1 are reported the reattachment locations, computed using up to sixth order polynomial approximations, which are in good agreement with the experimental data available in [37] and are almost identical to that, reported in [25], obtained with the same turbulence model.

The table shows also the x_r values obtained adopting both the Allmaras and the Bassi negative $\tilde{\nu}$ corrections and the standard or the modified r closure function. It is worth to mention that, apart the first order cases, all the values fall inside the same computational element. Furthermore it is easy to observe that, for the same order of polynomial approximation, all the results are within the 1% of tolerance proving that all the modifications here employed do not affect the final solutions.

As regards the solver computational efficiency this test case reveals some interesting features about the corrections adopted for the negative working variable of the turbulence model. Convergence histories are really similar for the two negative modifications, see Fig. 20. However the Allmaras approach is faster, in terms of Newton steps needed to reach the full convergence up to machine precision, at low and at moderate polynomial order, Fig. 21(a), while the Bassi one seems to be preferable for higher order computations, Fig. 21(b). In our opinion this behaviour can be justified since with the first



(a) $x = 2.5$.

(b) $x = 5$.

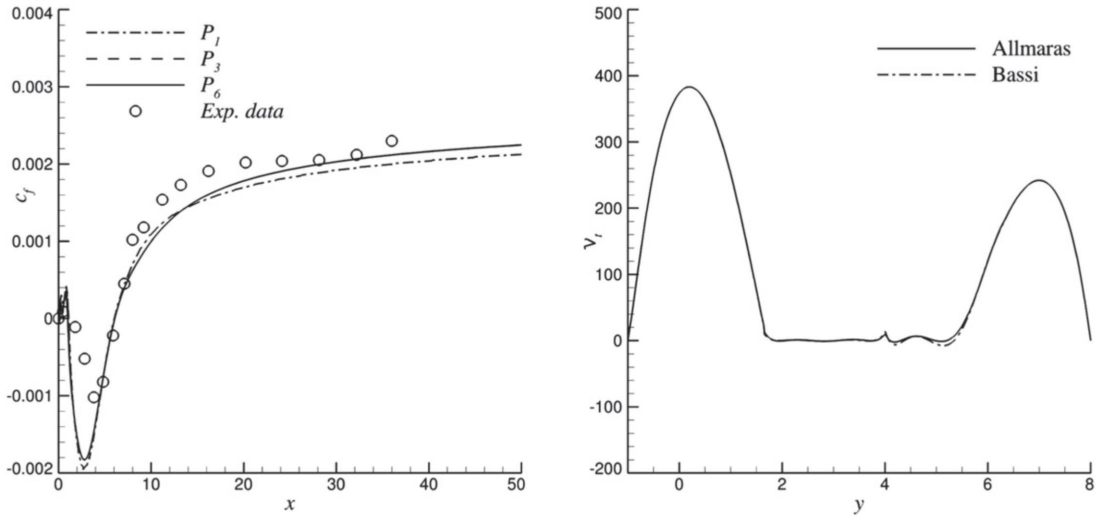
(c) $x = 10$.

(d) $x = 32$.

Fig. 18. Backward-facing step, $Re = 37400$. $O(y_c^+) \approx 0.1$. Numerical and experimental velocity profiles.

approach more terms are able to raise a negative $\tilde{\nu}$ value, thus a lower number of zone characterised by $\tilde{\nu} < 0$ are computed both during the solution process and in the final solution. However when a higher-order approximation is involved the non-linear system of equation becomes stiffer compared to that achieved following the Bassi approach. In fact in the latter case, when $\tilde{\nu} < 0$, the SA turbulence model is simply reduced into an advection–diffusion equation which does not involve any non-linear source terms as it happens in the Allmaras case. In Fig. 22 $\mathbb{P}_6$ convergence histories are depicted. It is quite easy to observe that relevant oscillations of the L_2 norms of the residuals vector can occur when this value is quite small thus, according to Eq. (38), Δt is large. In our computational experience even when this happens the computations do not blow-up and still converge to the steady state solution. Moreover the residuals oscillation amplitude can be safely reduced increasing the accuracy of the linear solver, for instance enhancing the total number of iteration or the Krylov sub-space size etc. Fig. 22(a), or reducing the time step size Δt lowering the β parameter in Eq. (38), Fig. 22(b).

Finally we mention that, in order to verify the solver robustness, in this case all the simulations were performed with $a^2 \to \infty$ in Eq. (39). As expected the lack of the block corresponding to the pressure DOFs in the mass matrix $\mathbf{M}$ is in some sense paid only with an increased number of the inner GMRES iterations within the first Newton steps.



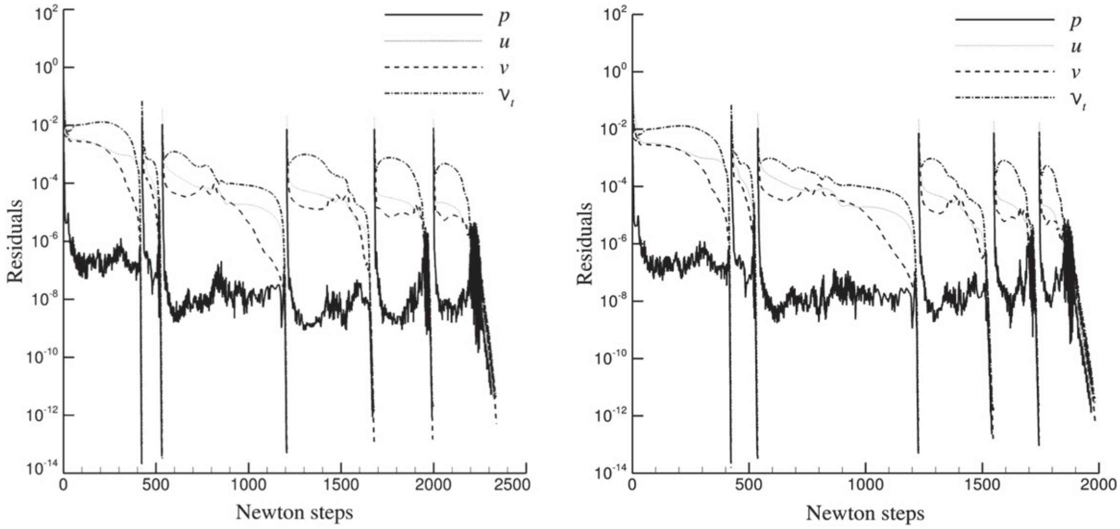
(a) Skin friction coefficients distribution

(b) $\tilde{\nu}$ profile

Fig. 19. Backward-facing step, $Re = 37400$. $O(y_c^+) \approx 0.1$.

Table 1
Backward-facing step, $Re = 37,400$. Computed reattachment location, x_r, for different polynomial approximations, $O(y_c^+) \approx 0.1$, Exp. $x_r = 6.26$ [37], Wilcox [25] $x_r = 6.1$.

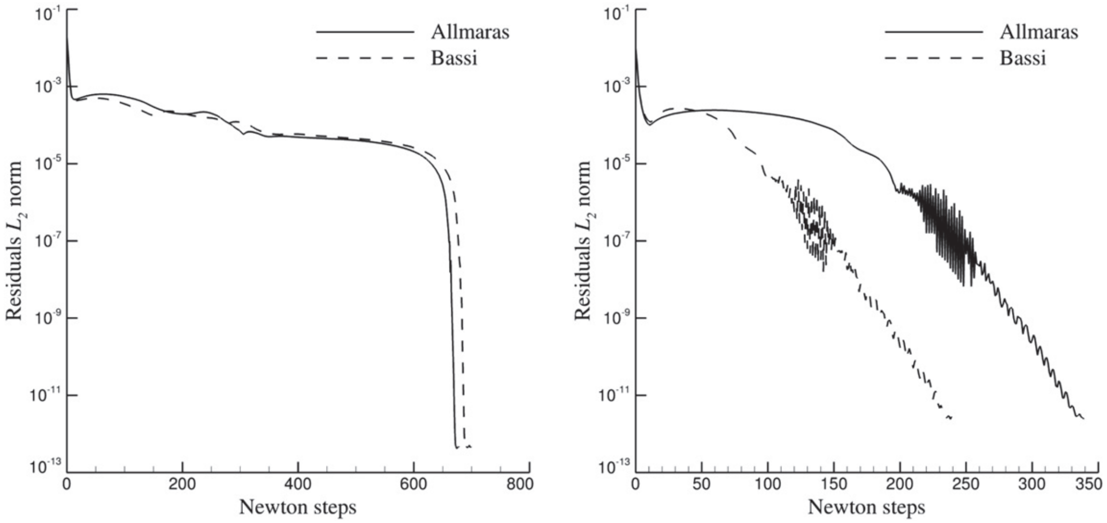
	$\mathbb{P}_1$	$\mathbb{P}_2$	$\mathbb{P}_3$	$\mathbb{P}_4$	$\mathbb{P}_5$	$\mathbb{P}_6$
Standard r, Allmaras $\tilde{\nu}$	5.6892	6.0084	6.0167	6.0159	6.0338	6.0105
Standard r, Bassi $\tilde{\nu}$	5.6932	6.0238	6.0372	6.0174	6.0350	6.0112
Modified r, Allmaras $\tilde{\nu}$	5.7113	6.0642	6.0661	6.0655	6.0656	6.0654
Modified r, Bassi $\tilde{\nu}$	5.7111	6.0687	6.0684	6.0669	6.0666	6.0660



(a) Allmaras modifications.

(b) Bassi modifications.

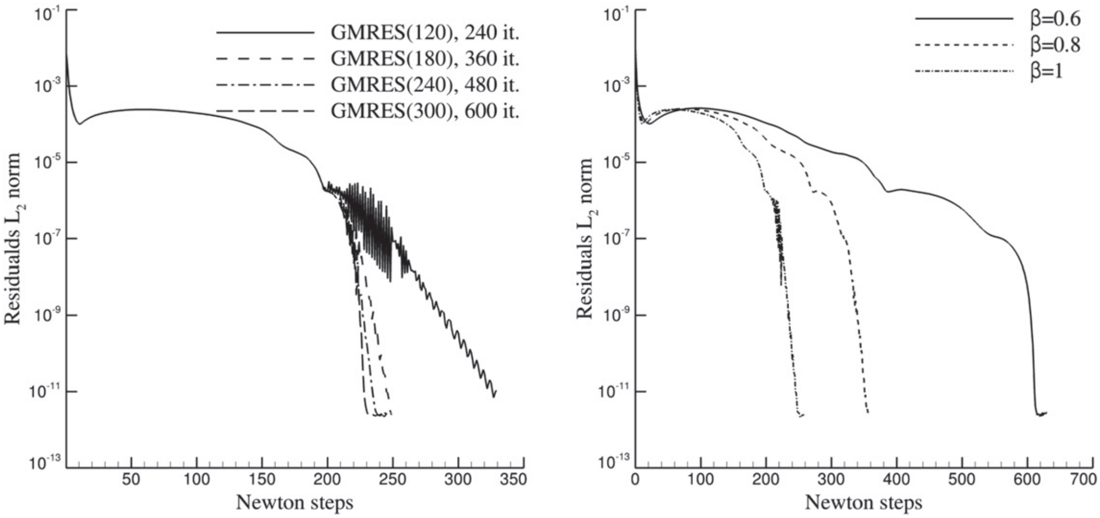
Fig. 20. Backward-facing step, $Re = 37400$. $\mathbb{P}_1, \mathbb{P}_2, \ldots, \mathbb{P}_6$ convergence histories.



(a) $\mathbb{P}_3$ residuals.

(b) $\mathbb{P}_6$ residuals.

Fig. 21. Backward-facing step $Re = 37400$. Convergence histories.



(a) Effect of the GMRES(m) settings.

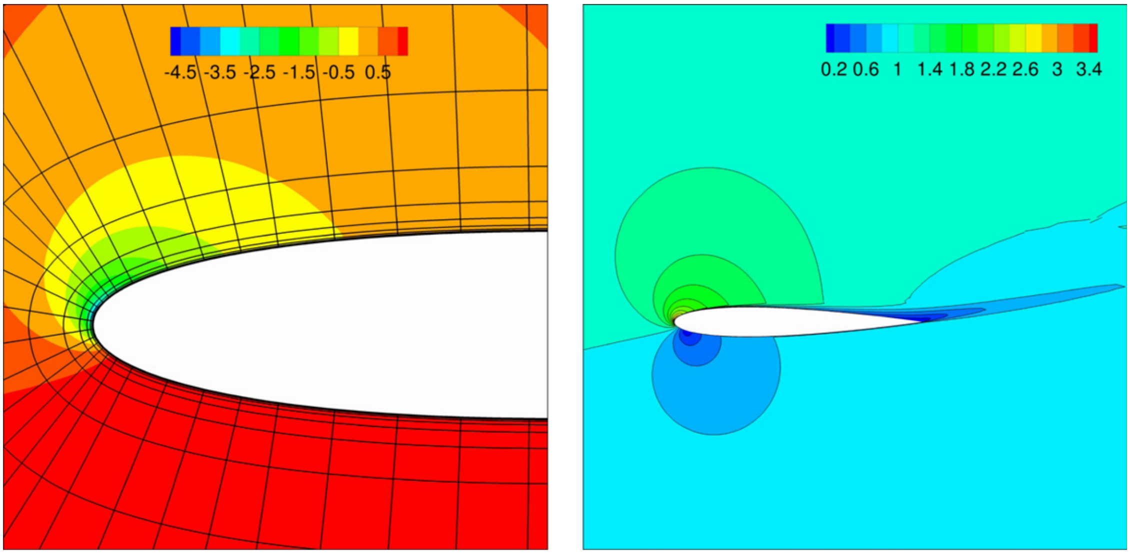
(b) Effect of the β parameter of eq. (38).

Fig. 22. Backward-facing step, $Re = 37400$. $\mathbb{P}_6$ convergence histories.

5.3. Turbulent flow around a NACA 0012 airfoil

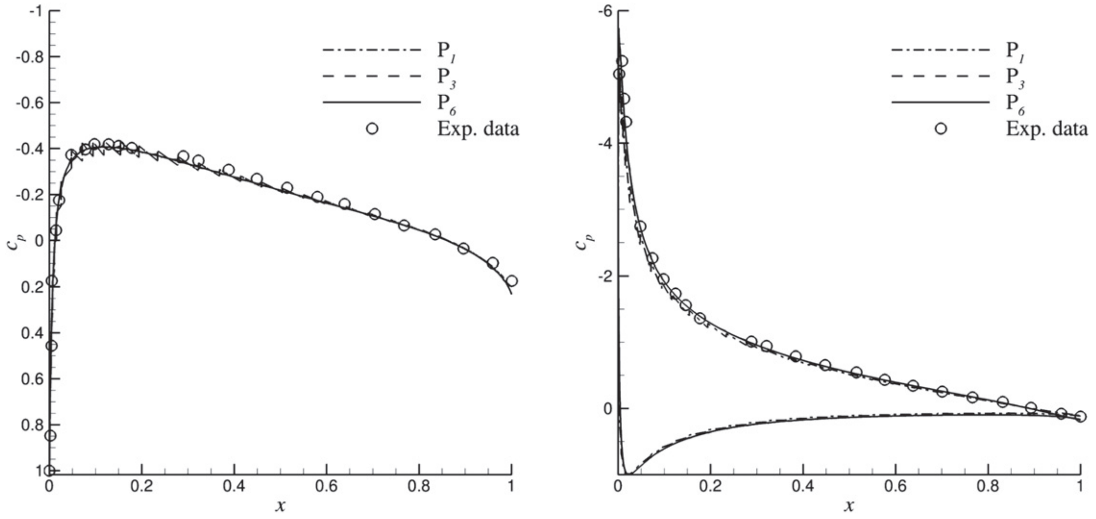
As a final test case we have computed the incompressible turbulent flow around a NACA 0012 airfoil at different Reynolds numbers ($Re = 2.88 \times 10^6$, 6×10^6) and angles of attack ($\alpha = 0^\circ$, 10°, 15°). A C-type grid with 2472 elements and piecewise cubic representation of the edges has been used. The farfield boundary has been placed at 100 chords length from the airfoil section. Grid lines have been suitably clustered around the leading edge and near the airfoil surface. First cell hight guarantees a maximum y_c^+ value of about 20 in proximity of the airfoil trailing edge (for the case $Re = 2.88 \times 10^6$ and $\alpha = 15^\circ$). This test case was computed up to $\mathbb{P}_6$ polynomial approximation, see Fig. 23 for some representative $\mathbb{P}_6$ results.

Our numerical results have been compared with the experimental data of Gregory and O'Reilly [38] ($Re = 2.88 \times 10^6$) and of Ladson [39,40] ($Re = 6 \times 10^6$). Figs. 24 and 25 highlight the excellent agreement between experimental data and numerical results.



(a) Pressure contours. (b) Velocity magnitude contours.

Fig. 23. NACA 0012, $Re = 2.88 \times 10^6$ and $\alpha = 15^\circ$. $\mathbb{P}_6$ solution.

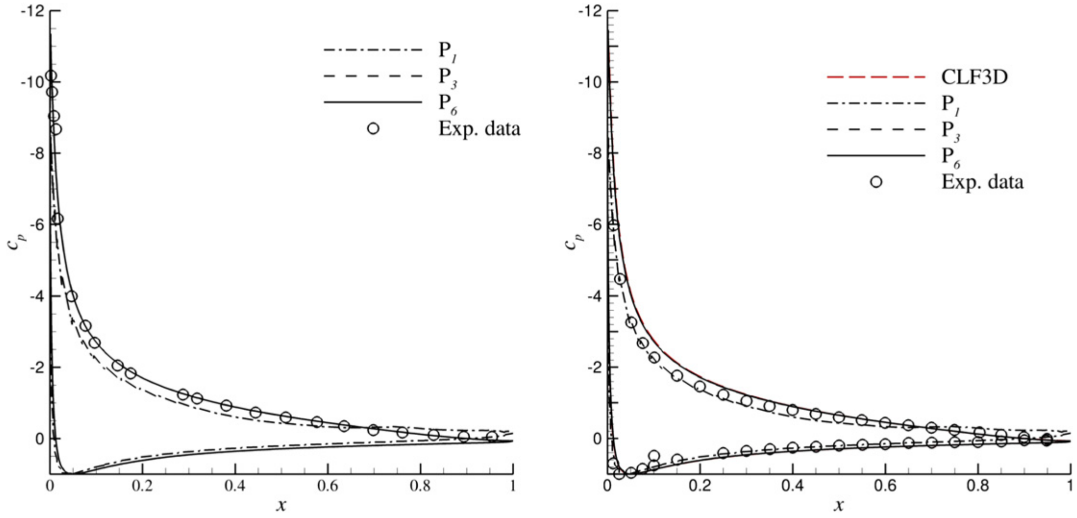


(a) $\alpha = 0^\circ$. (b) $\alpha = 10^\circ$.

Fig. 24. NACA 0012, $Re = 2.88 \times 10^6$. Pressure coefficient.

The DG pressure distributions for $Re = 6 \times 10^6$ were also compared with a reference SA solution computed by means of the well established CFL3D code [31] on a very fine 859 × 257 C-type mesh. This comparison, see Fig. 25(b), shows that our converged DG solutions are indistinguishable from the reference one, thus confirming once more that the proposed modifications of the SA model, while improving the robustness of the implementation, do not impair the quality of solutions. This aspect can also be seen looking at the force coefficients reported in Tables 2 and 3 for all the degrees of polynomial approximations and the various SA modifications proposed in this paper. The maximum difference for C_D is below 1.5%, while the lift coefficient C_L compares even better with a difference less than 0.2% for $\mathbb{P}_k$ solutions with $k \geqslant 2$.

For the case $Re = 6 \times 10^6$ and $\alpha = 0^\circ$ we have also performed a sensitivity analysis on the effect of the artificial compressibility parameter c^2 of the Riemann solver. Fig. 26 shows that even changing c^2 by five orders of magnitude, the computed pressure and skin friction distributions are almost identical even changing c^2 by several orders of magnitude, thus confirming the reliability and robustness of the method.



(a) $\mathrm{Re} = 2.88 \cdot 10^6$.

(b) $\mathrm{Re} = 6 \cdot 10^6$.

Fig. 25. NACA 0012, $\alpha = 15^\circ$. Pressure coefficient.

Table 2
NACA 0012, $Re = 2.88 \times 10^6$ at $\alpha = 15^\circ$. Computed $C_D \cdot 10^2$ for different polynomial approximations, $ne = 2472$.

	$\mathbb{P}_1$	$\mathbb{P}_2$	$\mathbb{P}_3$	$\mathbb{P}_4$	$\mathbb{P}_5$	$\mathbb{P}_6$
Standard r, Allmaras $\tilde{\nu}$	9.0546	2.6754	2.4722	2.4372	2.4188	2.4433
Standard r, Bassi $\tilde{\nu}$	8.9073	2.6541	2.4609	2.4185	2.4143	2.4175
Modified r, Allmaras $\tilde{\nu}$	8.6525	2.6445	2.4529	2.4140	2.4113	2.4114
Modified r, Bassi $\tilde{\nu}$	8.7111	2.6498	2.4558	2.4153	2.4122	2.4141

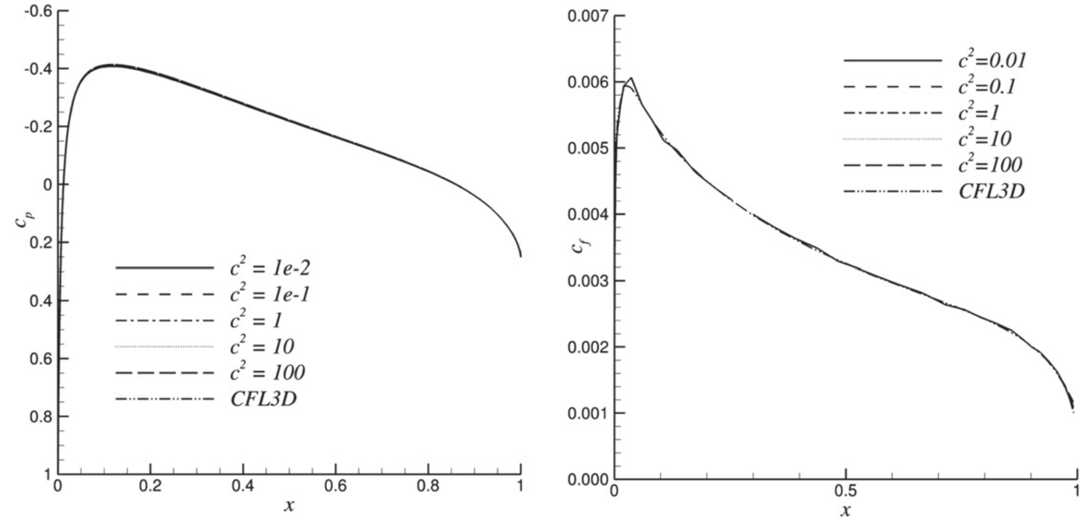
Table 3
NACA 0012, $Re = 2.88 \times 10^6$ at $\alpha = 15^\circ$. Computed C_L for different polynomial approximations, $ne = 2472$.

	$\mathbb{P}_1$	$\mathbb{P}_2$	$\mathbb{P}_3$	$\mathbb{P}_4$	$\mathbb{P}_5$	$\mathbb{P}_6$
Standard r, Allmaras $\tilde{\nu}$	1.1124	1.4823	1.4992	1.5023	1.5041	1.5014
Standard r, Bassi $\tilde{\nu}$	1.1207	1.4842	1.5000	1.5042	1.5046	1.5041
Modified r, Allmaras $\tilde{\nu}$	1.1369	1.4612	1.5017	1.5051	1.5052	1.5049
Modified r, Bassi $\tilde{\nu}$	1.1330	1.4848	1.5009	1.5047	1.5050	1.5047

Concerning the computational efficiency, Fig. 27 shows that the modified r function ensures really faster convergence to steady state. The solver is in fact significantly more stable, thus allowing the standard values of $\mathrm{CFL}_{\min}$ and β parameters in Eq. (38) to be safely used. Instead, the stability of the original r formulation requires, especially for high angles of attack, lower values of such parameters and a larger number of Newton steps to converge. Moreover, the increased stability of the new r formulation, together with the analytical residual linearisation employed in the code, allows to increase the $\mathrm{CFL}_{\min}$ parameter up to 100 after the first or second lower order solutions, thus strongly reducing the number of Newton steps to reach convergence in the following higher order solutions, see Fig. 27(b). We believe that this behaviour of the solver, already seen in the convergence histories of the flat plate problem shown in Fig. 15(a), is important for the overall efficiency of high-order solutions, since larger number of Newton steps are required only for the relatively less costly low-order computations.

As regards the behaviour of the Allmaras and Bassi treatments of negative $\tilde{\nu}$ values, this test case confirms the comments done in the previous section, see Fig. 28.

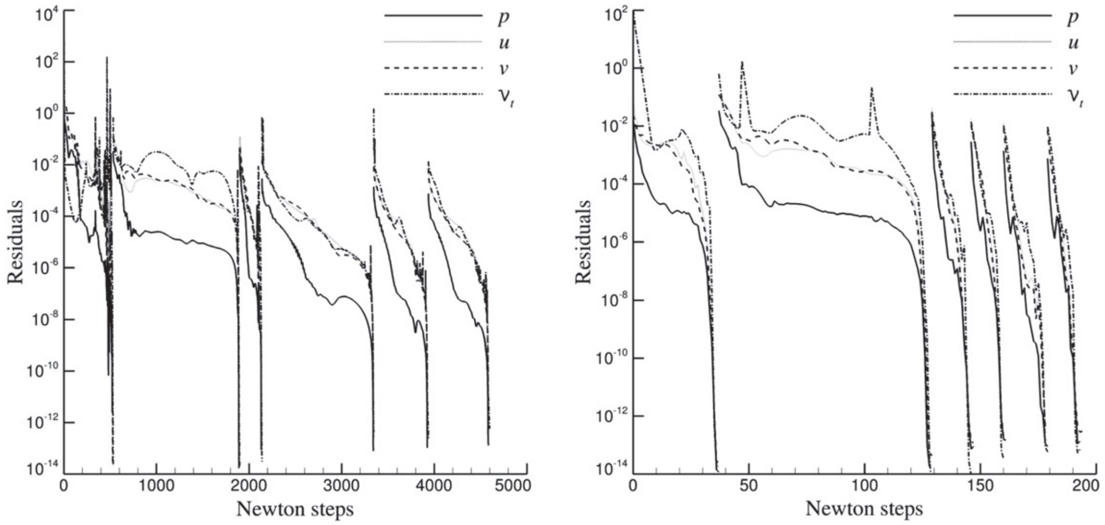
Fig. 29 shows the zones with negative eddy-viscosity and highlights that the Allmaras corrections result in a lower number of elements with negative eddy viscosity with respect to the approach of Bassi. However, both approaches produce quite extended zones with non positive $\tilde{\nu}$ values but their reliability to assure a smooth transition of the eddy viscosity to its freestream value allows to prevent the blow-up of the computations.



(a) Pressure coefficient.

(b) Skin friction coefficient.

Fig. 26. NACA 0012, $Re = 6 \times 10^6$ at $\alpha = 0^\circ$, $\mathbb{P}_6$ solutions. Effect of the artificial compressibility coefficient.



(a) Bassi $\tilde{\nu}$, standard r.

(b) Bassi $\tilde{\nu}$, modified r.

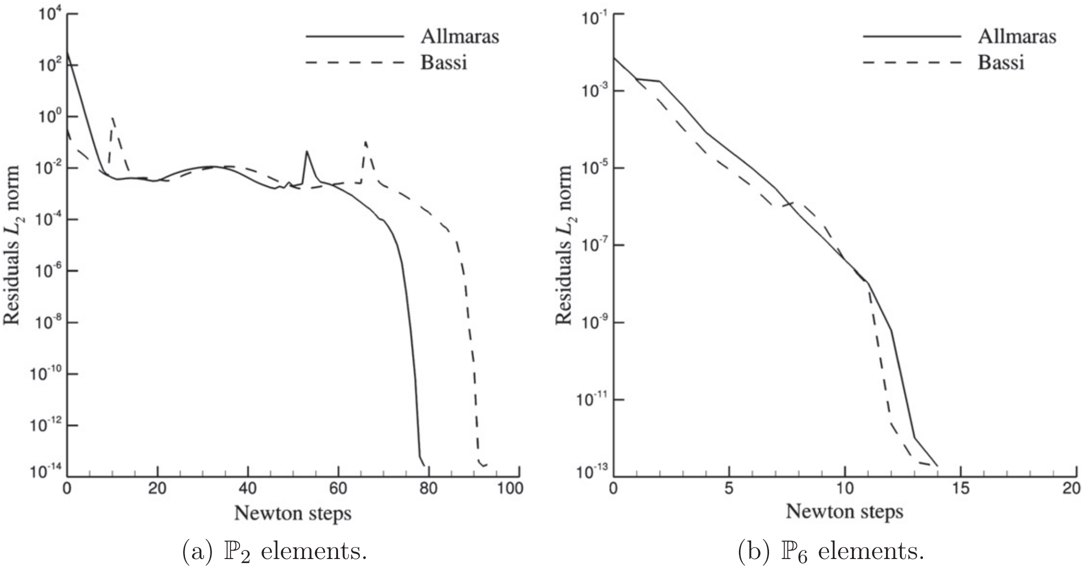
Fig. 27. NACA 0012, $Re = 2.88 \times 10^6$ at $\alpha = 15^\circ$. $\mathbb{P}_1, \mathbb{P}_2, \ldots, \mathbb{P}_6$ convergence histories.

6. Conclusions

In this work a high-order DG method for the incompressible RANS equations coupled with the SA turbulence model has been presented. The DG discretization of the nonlinear convective terms is based on the artificial compressibility flux formulation introduced few years ago for the incompressible Navier–Stokes equations. This approach has proved to be very well suited to deal with the numerical stiffness induced by highly stretched grids, with possibly curved elements, and by the strongly nonlinear character of the governing equations, typical of high-Reynolds number turbulent flow computations.

Two are the main contributions of this paper about the SA model implementation. The first one is how to deal with unphysical, but likely to occur numerically, negative eddy viscosity values. The second one is the proposal, based on a detailed analysis, of a new limiting procedure for one of the model closure functions that strongly affects the numerical stability of the implementation, even for positive values of $\tilde{\nu}$.

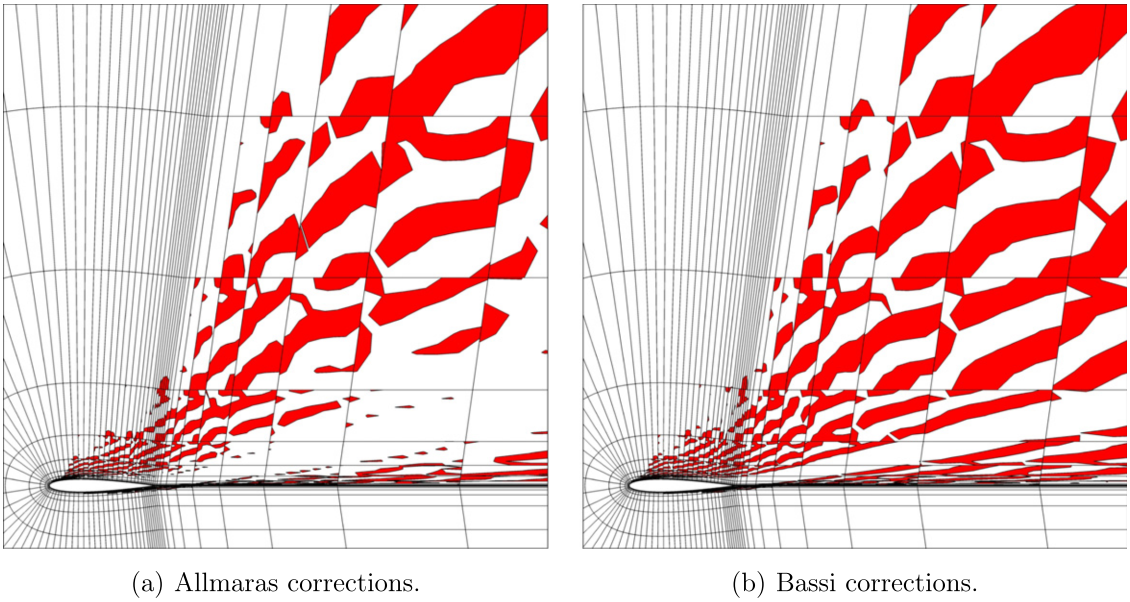
The occurrence of negative eddy viscosity values has been addressed and numerically evaluated by means of two approaches. The first one was suggested by Allmaras himself while the second one was inspired to the DG implementation



(a) $\mathbb{P}_2$ elements.

(b) $\mathbb{P}_6$ elements.

Fig. 28. NACA 0012, $Re = 2.88 \times 10^6$ at $\alpha = 15°$. Convergence histories.



(a) Allmaras corrections.

(b) Bassi corrections.

Fig. 29. NACA 0012 airfoil, $Re = 2.88 \times 10^6$ at $\alpha = 15°$. $\mathbb{P}_6$ solution. Negative $\tilde{\nu}$ zones.

of the k–ω turbulence model reported in Bassi et al. [12]. The two approaches differ in that the corrections proposed by Allmaras are continuous and ensure continuity of the Jacobian, while in the approach of Bassi et al. [12] the Jacobian is discontinuous. We remark that, since the Allmaras modifications include more terms that could lead to the growth of an unphysical $\tilde{\nu}$ value, usually the zones with negative eddy viscosities are smaller in size. However, as shown in Section 5, the overall quality of results and the behaviour of the solution algorithm are quite similar in the two approaches.

As a second contribution to a robust and efficient implementation of the SA model, we have introduced a new limiting procedure for the closure function r of the model. The behaviour of this function in the original SA model sometimes leads to a positive Jacobian of the source term which impairs the convergence to steady state of an implicit solver. The procedure introduced in this work prevents the r function, whose square root is in fact a length scale, from attaining negative values, and avoids the sign inversion of production and destruction source terms. Numerical experiments have shown that this modification of the SA model becomes essential when computing flows with strong adverse pressure gradients or large recirculation and separation zones.

By comparing our results with reference results available in the literature we have found that all the SA model modifications here proposed do not affect the RANS-SA solution, while they significantly improve the reliability of the model and the robustness of the code. Furthermore, these SA modifications are not specific to the DG method, thus they can be easily implemented in any RANS-SA solver with significant advantage. Finally, the proposed modifications do not require any additional computational effort and thus do not impair the efficiency of the implicit solver, which is able to provide high-order solutions in a small number of Newton steps.

The code for three-dimensional flows in complex geometries has already demonstrated numerical robustness similar to the two-dimensional case. The implementation and the demonstration of the method for the three-dimensional case will be the subject of a forthcoming paper. Future work will be devoted to the implementation of DES in our high-order incompressible DG solver.

References

[1] B. Cockburn, C.W. Shu, The Runge–Kutta discontinuous Galerkin finite element method for conservation laws V: multidimensional systems, J. Comput. Phys. 141 (1998) 199–224.

[2] F. Bassi, S. Rebay, G. Mariotti, S. Pedinotti, M. Savini, A high-order accurate discontinuous finite element method for inviscid and viscous turbomachinery flows, in: R. Decuypere, G. Dibelius, Proceedings of the 2nd European Conference on Turbomachinery Fluid Dynamics and Thermodynamics, Technologisch Instituut, Antwerpen, Belgium, 1997, pp. 99–108.

[3] F. Bassi, S. Rebay, A high-order accurate discontinuous finite element method for the numerical solution of the compressible Navier–Stokes equations, J. Comput. Phys. 131 (1997) 267–279.

[4] J.G. Liu, C.W. Shu, A high-order discontinuous Galerkin method for 2D incompressible flows, J. Comput. Phys. 160 (2) (2000) 577–596.

[5] B. Cockburn, G. Kanschat, D. Schötzau, C. Schwab, Local discontinuous Galerkin methods for the Stokes system, SIAM J. Numer. Anal. 40 (1) (2002) 319–343.

[6] B. Cockburn, G. Kanschat, D. Schötzau, The local discontinuous Galerkin method for linearized incompressible fluid flow: a review, Comput. Fluids 34 (2005) 491–506.

[7] B. Cockburn, G. Kanschat, D. Schötzau, A locally conservative LDG method for the incompressible Navier–Stokes equations, Math. Comput. 74 (2005) 1067–1095.

[8] P.F. Fischer, K. Shahbazi, C.R. Ethier, A high-order discontinuous Galerkin method for the unsteady incompressible Navier–Stokes equations, J. Comput. Phys. 222 (1) (2007) 391–407.

[9] N.C. Nguyen, J. Peraire, B. Cockburn, An implicit high-order hybridizable discontinuous Galerkin method for the incompressible Navier–Stokes equations, J. Comput. Phys. 230 (4) (2011) 1147–1170.

[10] L. Botti, D.A. Di Pietro, A pressure-correction scheme for convection-dominated incompressible flows with discontinuous velocity and continuous pressure, J. Comput. Phys. 230 (3) (2011) 572–585.

[11] R. Henniger, D. Obrist, L. Kleiser, High-order accurate solution of the incompressible Navier–Stokes equations on massively parallel computers, J. Comput. Phys. 229 (10) (2010) 3543–3572.

[12] F. Bassi, A. Crivellini, S. Rebay, M. Savini, Discontinuous Galerkin solution of the Reynolds averaged Navier–Stokes and k–ω turbulence model equations, Comput. Fluids 34 (2005) 507–540.

[13] P.O. Persson, N. Nguyen, J. Peraire, RANS solutions using high order discontinuous Galerkin methods, in: 45th AIAA Aerospace Sciences Meeting and Exhibit, Reno, NV, AIAA-2007-914, January 8–11, 2007.

[14] T.A. Oliver, D.L. Darmofal, An unsteady adaptation algorithm for discontinuous Galerkin discretizations of the RANS equations, in: 18th AIAA Computational Fluid Dynamics Conference, Miami, FL, AIAA-2007-3940, June 25–28, 2007.

[15] T.A. Oliver, D.L. Darmofal, Impact of turbulence model irregularity on high-order discretization, in: 47th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition, Orlando, FL, AIAA-2009-953, January 5–8 2009.

[16] B. Landmann, M. Kessler, S. Wagner, E. Krämer, A parallel, high-order discontinuous Galerkin code for laminar and turbulent flows, Comput. Fluids 37 (4) (2008) 427–438 (Turbulent flow and noise generation).

[17] R. Hartmann, J. Held, T. Leicth, F. Prill, Discontinuous Galerkin methods for computational aerodynamics – 3D adaptive flow simulation with the DLR PADGE code, Aerosp. Sci. Technol. 14 (7) (2010) 512–519.

[18] F. Bassi, A. Crivellini, D.A. Di Pietro, S. Rebay, An artificial compressibility flux for the discontinuous Galerkin solution of the incompressible Navier–Stokes equations, J. Comput. Phys. 218 (2) (2006) 794–815.

[19] F. Bassi, A. Crivellini, A high-order discontinuous Galerkin method for natural convection problems, in: P. Wesseling, E. Onate, J. Périaux (Eds.), Electronic Proceedings of the ECCOMAS CFD 2006 Conference, Egmond aan Zee, The Netherlands, TU Delft, September 5–8, 2006.

[20] F. Bassi, A. Crivellini, D.A. Di Pietro, S. Rebay, An implicit high-order discontinuous Galerkin method for steady unsteady incompressible flows, Comput. Fluids 36 (10) (2007) 1529–1546 (Special issue dedicated to Professor Michele Napolitano on the occasion of his 60th birthday).

[21] J. Peraire, P.O. Persson, The compact discontinuos Galerkin (CDG) method for elliptic problems, SIAM J. Sci. Comput. 30 (4) (2008) 1806–1824.

[22] P.R. Spalart, S.R. Allmaras, A one-equation turbulent model for aerodynamic flows, La Recherche Aérospatiale 1 (1994) 5–21.

[23] S.R. Allmaras, Multigrid for the 2-d compressible Navier–Stokes equations, in: 14th AIAA Computational Fluid Dynamics Conference, Norfolk, VA, AIAA-2007-914, June 28–July 1, 1999.

[24] T.A. Oliver, A high-order, adaptive, discontinuous Galerkin finite element method for the Reynolds-averaged Navier–Stokes equations, Ph.D. Thesis, MIT, 2008.

[25] D.C. Wilcox, Turbulence modeling for CFD. DCW Industries, third ed., Incorporated, November 1, 2006.

[26] D.N. Arnold, F. Brezzi, B. Cockburn, D. Marini, Unified analysis of discontinuous Galerkin methods for elliptic problems, SIAM J. Numer. Anal. 39 (5) (2002) 1749–1779.

[27] R. Cools, An encyclopedia of cubature formulas, J. Complex. 19 (2003) 445–453.

[28] S. Balay, K. Buschelman, W.D. Gropp, D. Kaushik, M.G. Knepley, L.C. McInnes, B.F. Smith, H. Zhang. PETSc Web page, 2001, <http://www.mcs.anl.gov/petsc>.

[29] A. Crivellini, F. Bassi, An implicit matrix-free discontinuous Galerkin solver for viscous and turbulent aerodynamic simulations, Comput. Fluids 50 (1) (2011) 81–93.

[30] G. Karypis, V. Kumar. METIS, a software package for partitioning unstructured graphs, partitioning meshes, and computing fill-reducing orderings of sparse matrices, Technical Report Version 4.0, University of Minnesota, Department of Computer Science/Army HPC Research Center, 1998.

[31] C.L. Rumsey, B.R. Smith, G.P. Huang, Description of a website resource for turbulence modeling verification and validation, in: 40th AIAA Computational Fluid Dynamics Conference and Exhibit, Chicago, IL, AIAA-2010-4742, June 28–July 1, 2010.

[32] Tuncer Cebeci, Analysis of Turbulent Flows, Elsevier, 2004.

[33] K. Wieghardt, W. Tillman, On the turbulent friction layer for rising pressure, Technical Report 1314, NACA, 1951.

[34] D.E. Coles, E.A. Hirst, Computation of turbulent boundary layer, in: AFORS-IFP-Stanford Conference, 1969.

[35] D.A. Yoder, N.J. Georgiadis, Implementation and validation of the chien $k-\epsilon$ turbulence model in the wind Navier–Stokes code, AIAA Paper 99-0745, 1999.

[36] C.L. Rumsey, Apparent transition behavior of widely-used turbulence models, Int. J. Heat Fluid Flow 28 (6) (2007) 1460–1471 (Revised and extended papers from the 5th Conference in Turbulence, Heat and Mass Transfer).
[37] D. Driver, H. Seegmiller, Features of a reattaching turbulent shear layer in divergent channel flow, AIAA J. 23 (1985) 163–171.
[38] N. Gregory, C.L. O'Reilly, Low-speed aerodynamic characteristics of NACA 0012 aerofoil section, including the effects of upper-surface roughness simulating hoar frost, Technical Report, Aeronautical Research Council, 1970.
[39] C.L. Ladson, Low-speed aerodynamic characteristics of NACA 0012 Aerofoil Sections, including the Effects of Upper-Surface Roughness Simulation Hoar Frost, Technical Report TM 4074, NASA, October 1988.
[40] C.L. Ladson, S.H. Hill, W.G. Johnson, Pressure distributions from high Reynolds number transonic tests of an NACA 0012 airfoil in the Langley 0.3-Meter Transonic Cryogenic Tunnel, Technical Report TM 100526, NASA, December 1987.