Received: 20 June 2021 | Revised: 16 March 2022 | Accepted: 25 March 2022

DOI: 10.1002/fld.5089

RESEARCH ARTICLE

WILEY

An implicit *p*-adaptive discontinuous Galerkin solver for CAA/CFD simulations

Alessandro Colombo[1] | **Andrea Crivellini**[2] | **Antonio Ghidoni**[3] | **Alessandra Nigro**[2] | **Gianmaria Noventa**[3]

[1]Department of Engineering and Applied Sciences, University of Bergamo, Dalmine (BG), Italy

[2]Department of Industrial Engineering and Mathematical Sciences, Polytechnic University of Marche, Ancona, Italy

[3]Department of Mechanical and Industrial Engineering, University of Brescia, Brescia, Italy

Correspondence
Alessandro Colombo, Department of Engineering and Applied Sciences, University of Bergamo, viale Marconi 5, 24044 Dalmine, (BG), Italy.
Email: alessandro.colombo@unibg.it

Abstract

In the last decades flow simulations have become a routine practice in many industrial fields for the aerodynamic and noise prediction. Moreover, the ever increasing interest in simulating off-design operating conditions promoted the development of high-fidelity simulation tools to overcome the modeling and accuracy limits of standard industrial codes in predicting turbulent separated flows. The discontinuous Galerkin (DG) method is well suited for this class of simulations, but today DG-based CFD and CAA (Computational AeroAcoustics) solvers cannot yet reach the computational efficiency of well-established commercial codes. As a consequence, the present paper aims at exploiting some attractive strategies, such as the adaptation of the elemental polynomial degree (*p*-adaptation) and of the degree of exactness of quadrature rules, to enhance the computational efficiency of an implicit DG platform for CFD and CAA simulations. Moreover, a sponge layer non-reflecting boundary treatment has been also implemented for CAA. The predicting capabilities of the method have been assessed on classical CAA and CFD test cases. The proposed adaptive strategy guarantees a significant reduction (≈50%) of the computational effort for both CFD and CAA simulations, compared to uniform-order discretizations, while not spoiling the high accuracy requested to an high-fidelity simulation tool.

KEYWORDS

computational AeroAcoustics, discontinuous Galerkin, implicit LES, *p*-adaptation, SD7003 airfoil, sponge-layer

1 | INTRODUCTION

In many industrial fields standard second-order accurate codes are not able to satisfy the growing need for high-fidelity flow simulations. In this context higher-order methods, such as the discontinuous Galerkin (DG) methods,[1-6] can present a viable solution. DG methods, compared to industrial codes, can provide a greater accuracy, but are still too computational demanding for a routine use.[7] However, the industrial interest for these methods has been strongly promoting research efforts to increase their efficiency.[8-12] Previous works contributed to the development of efficient high-order

numerical methods for steady and unsteady flow problems involving adaptation of the spatial discretization by varying the order of the polynomial approximation throughout the domain, for example, References 13-17, by performing mesh adaptation, for example, References 18-22, or both, for example, References 23,24.

In this article, a *p*-adaptation strategy has been developed to increase the computational efficiency of a DG solver[25] for both Computational AeroAcoustics (CAA) and Computational Fluid Dynamics (CFD) simulations, for example, scale-resolving simulations performed according to the implicit large eddy simulation (ILES) approach. In the ILES,[4,5,25-31] the unfiltered Navier–Stokes equations are solved and the numerical dissipation introduced by the discretization plays the role of an explicit subgrid-scale (SGS) model that dissipates the smallest scale eddies.

The proposed order-adaptive strategy locally varies the polynomial degree of the solution representation within each element according to an error estimator, thus reducing the CPU time and memory usage, while not spoiling the spectral resolution required by CFD and CAA computations. CFD and CAA simulations are here based on the same set of governing equations, that is, the Navier–Stokes equations, but focusing on different features of the flow field. In fact, we are mainly interested in loads distributions and total pressure losses for the CFD applications, while we seek to accurately solve the acoustics perturbations over the domain for the CAA applications.

The idea behind this work is to use, as far as possible, the same framework for both applications to significantly reduce the efforts in code development and maintenance. To this end, we opted for the same error estimator, regardless of the type of application, which combines two contributions: (i) the solution jumps at grid cells interfaces;[32,33] (ii) the decay rate of the modal coefficients of the polynomial expansion.[34] This coupling guarantees a proper behavior both for high- and low-degree polynomial approximations. Despite not targeting any specific output quantity, the strategy performed reasonably well in the solution of both compressible and incompressible turbulent flows.[35-37] Although we formally rely on the same estimator, its value is calculated from two different solution statistics, computed at runtime, depending on whether we are interested in CFD or CAA aspects. In particular, for the turbulent flows proper of most of the CFD applications we use the time-averaged solution, while the estimator is computed from the root-mean-square (r.m.s.) of the solution for the CAA cases. To further optimize the computational effort, the degree of exactness for the quadrature rules is adapted over the mesh to avoid the possible over integration of straight-sided elements, while online load-balancing is applied to handle the degrees of freedom imbalance on each partition due to the adaptation algorithm.

When dealing with CAA, and, in general, with high-fidelity flow simulations, the use of a non-reflecting boundary treatment is mandatory for the correct prediction of some quantities and distributions, for example, the sound directivity. Supported by the results of Colombo and Crivellini,[38] we opted for an absorbing sponge layer approach and extended its implementation to a *p*-adaptive framework. The accuracy and efficiency of the adaptive strategies have been assessed on classical CAA test cases, such as the flow around a circular cylinder at Reynolds number $\mathrm{Re}_D = 100$ with Mach numbers $\mathrm{M}_\infty = \{0.15, 0.5\}$,[38] and on the ILES of the flow: (i) around a circular cylinder at $\mathrm{Re}_D = 3900$ and (ii) around a Selig-Donovan 7003 (SD7003) airfoil at $\mathrm{Re}_C = 60{,}000$ and angles of attack $\alpha = \{4°, 8°\}$.

The article is organized as follows. Section 2 introduces the DG spatial discretization and the implicit time integration of the governing equations. Section 3 briefly discusses the method implementation with particular attention to the definition of the orthonormal hierarchical polynomial basis functions, the algorithm used to adapt the degree of exactness for quadrature rules, the *p*-adaptation strategies and the load balancing algorithm. In Section 4 the results from the testcases are presented. Finally, the conclusions are given in Section 5.

2 | THE NUMERICAL METHOD

In this section we briefly describe the spatial and temporal discretization used by the DG code MIGALE (see References 5,39 for a more complete description).

2.1 | The DG discretization of the Navier–Stokes equations

The complete set of the Navier–Stokes equations for compressible flows can be written as

$$\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x_j}\left(\rho u_j\right) = 0$$

$$\frac{\partial}{\partial t}(\rho u_i) + \frac{\partial}{\partial x_j}\left(\rho u_j u_i\right) = -\frac{\partial p}{\partial x_i} + \frac{\partial \tau_{ji}}{\partial x_j}$$
$$\frac{\partial}{\partial t}(\rho E) + \frac{\partial}{\partial x_j}\left(\rho u_j H\right) = \frac{\partial}{\partial x_j}\left[u_i \tau_{ji} - q_j\right], \tag{1}$$

where ρ is the fluid density, E and H are the total energy and enthalpy, respectively. The pressure p, the stress tensor τ_{ij} and the heat flux vector q_j are given by

$$p = (\gamma - 1)\rho\left(E - \frac{u_k u_k}{2}\right), \tag{2}$$

$$\tau_{ji} = 2\mu\left[S_{ij} - \frac{1}{3}\frac{\partial u_k}{\partial x_k}\delta_{ij}\right], \tag{3}$$

$$q_j = -\frac{\mu}{Pr}\frac{\partial h}{\partial x_j}, \tag{4}$$

where $\gamma = C_p/C_v$ is the ratio of gas specific heats, Pr is the molecular Prandtl number, h is the enthalpy and

$$S_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right), \tag{5}$$

is the strain-rate tensor.

In compact form the set of governing equations can be written, for d dimensions and $m = 2 + d$ variables, as

$$\mathbf{P}(\mathbf{w})\frac{\partial \mathbf{w}}{\partial t} + \nabla \cdot \mathbf{F}_c(\mathbf{w}) + \nabla \cdot \mathbf{F}_v(\mathbf{w}, \nabla\mathbf{w}) = \mathbf{0}, \tag{6}$$

where $\mathbf{w} = [p, T, u_{i=1,\dots,d}] \in \mathbb{R}^m$ is the unknown solution vector, $\mathbf{F}_c, \mathbf{F}_v \in \mathbb{R}^m \otimes \mathbb{R}^d$ are the convective and viscous flux functions, and $\mathbf{P}(\mathbf{w}) \in \mathbb{R}^m \otimes \mathbb{R}^m$ is a transformation matrix. Notice that, when using the set of conservative variables, that is, $\mathbf{w} = \mathbf{w}_c = [\rho, \rho u_i, \rho E]^T$, $\mathbf{P}$ reduces to the identity matrix $\mathbf{I}$. For reading convenience we will also refer to the velocity vector as $\mathbf{u} = [u_1, u_2, u_3] = [u_x, u_y, u_z]$.

By multiplying Equation (6) by an arbitrary smooth test function $\mathbf{v} = \{v_1, \dots, v_m\}$, and integrating by parts, we obtain the weak formulation

$$\int_\Omega \mathbf{v} \cdot \left(\mathbf{P}(\mathbf{w})\frac{\partial \mathbf{w}}{\partial t}\right) d\mathbf{x} - \int_\Omega \nabla\mathbf{v} : \mathbf{F}(\mathbf{w}, \nabla\mathbf{w}) d\mathbf{x} + \int_{\partial\Omega} \mathbf{v} \otimes \mathbf{n} : \mathbf{F}(\mathbf{w}, \nabla\mathbf{w}) d\sigma = \mathbf{0}, \tag{7}$$

where $\mathbf{F}$ is the sum of the convective and viscous flux functions and $\mathbf{n}$ is the unit vector normal to the boundary.

Following the idea to define discrete polynomial spaces in physical coordinates, an approximation based on the space

$$\mathbb{P}_d^k(\mathcal{K}_h) \overset{\text{def}}{=} \left\{v_h \in L^2(\Omega) \mid v_{h|K} \in \mathbb{P}_d^k(K),\ \forall K \in \mathcal{K}_h\right\}, \tag{8}$$

is considered, where k is a non-negative integer and $\mathbb{P}_d^k(K)$ denotes the restriction of the polynomial functions of d variables and maximum degree k to the cell K, part of the computational mesh $\mathcal{K}_h = \{K\}$ of the flow domain $\Omega \in \mathbb{R}^d$. For the sake of presenting the discretization, we define the set $\mathcal{F}_h$ of the mesh faces $\mathcal{F}_h \overset{\text{def}}{=} \mathcal{F}_h^i \cup \mathcal{F}_h^b$, where $\mathcal{F}_h^b$ collects the faces located on the boundary of Ω_h and for each $F \in \mathcal{F}_h^i$ there exist two elements $K^+, K^- \in \mathcal{K}_h$ such that $F \in \partial K^+ \cap \partial K^-$. Moreover, for all $F \in \mathcal{F}_h^b$, $\mathbf{n}_F$ denotes the unit outward normal to Ω_h, whereas, for all $F \in \mathcal{F}_h^i$, $\mathbf{n}_F^-$ and $\mathbf{n}_F^+$ are unit outward normals pointing to K^+ and K^-, respectively.

To discretize Equation (7) we replace the solution $\mathbf{w}$ and the test function $\mathbf{v}$ with a finite element approximation $\mathbf{w}_h$ and a discrete test function $\mathbf{v}_h$, respectively, where $\mathbf{w}_h$ and $\mathbf{v}_h$ belong to the space $\mathbf{V}_h \overset{\text{def}}{=} [\mathbb{P}_d^k(\mathcal{K}_h)]^m$. As we deal with adaptive computations, the polynomial space will be denoted as $\mathbb{P}_d^{k_m \to k_M}$, where k_m and k_M are the minimum and maximum polynomial degree over the discretization, respectively. The symbol k_K denotes the polynomial degree local to any element $K \in \mathcal{K}_h$. For each mesh cell an orthonormal and hierarchical basis $\Phi_K^k = \left\{\phi_i^K\right\}$, where $i \in \{1, \dots, N_{dof}^K\}$, is obtained

by applying the modified Gram-Schmidt (MGS) orthogonalization algorithm to a set of monomials defined in a reference frame centered in the element barycenter and aligned with the principal axes of inertia of the element.[39] Details on the implementation of shape functions will be given in Section 3.1. Each component $j = 1, \dots, m$ of the numerical solution $\mathbf{w}_h$ over each element K can be expressed, in terms of the elements of the global vector $\mathbf{W}$ of unknown degrees of freedom, as $\mathbf{w}_{h,j}^K = \phi_l^K W_{j,l}^K$ where $\mathbf{W}^K$ is the local part of the global vector storing the degrees of freedom of the element K and the repeated index l implies summation over the range $l = 1, \dots, N_{dof}^K = \text{card}(\mathbb{P}_d^{k_K})$. As a function $w_h \in \mathbb{P}_d^k(\mathcal{K}_h)$ is double valued over a mesh face we introduce the jump $[\![w_h]\!] \overset{\text{def}}{=} w_{h|K^+}\mathbf{n}_F^+ + w_{h|K^-}\mathbf{n}_F^-$ and the average $\{w_h\} \overset{\text{def}}{=} (w_{h|K^+} + w_{h|K^-})/2$ trace operators. These operators act componentwise when applied to a vector.

Then, the DG discretization of the Navier–Stokes equations consists in seeking, for $j = 1, \dots, m$, the elements of $\mathbf{W}$ such that

$$\begin{aligned} &\sum_{K\in\mathcal{K}_h} \int_K \phi_i P_{j,k}(\mathbf{w}_h)\, \phi_l \frac{dW_{k,l}}{dt} d\mathbf{x} \\ &- \sum_{K\in\mathcal{K}_h} \int_K \frac{\partial \phi_i}{\partial x_n} F_{j,n}(\mathbf{w}_h, \nabla_h \mathbf{w}_h + \mathbf{r}([\![\mathbf{w}_h]\!]))\, d\mathbf{x} \\ &+ \sum_{F\in\mathcal{F}_h} \int_F [\![\phi_i]\!]_n \hat{F}_{j,n}\left(\mathbf{w}_h^{\pm}, (\nabla_h \mathbf{w}_h + \eta_F \mathbf{r}_F([\![\mathbf{w}_h]\!]))^{\pm}\right) dx = 0, \end{aligned} \tag{9}$$

for $i = 1, \dots, N_{dof}^K$, where repeated indices imply summation over the ranges $k = 1, \dots, m$, $l = 1, \dots, N_{dof}^K$ and $n = 1, \dots, d$.

The space discretization of the second-order terms uses the BR2 scheme,[40] where the viscous numerical flux is given by

$$\hat{\mathbf{F}}_v\left(\mathbf{w}_h^{\pm}, (\nabla_h \mathbf{w}_h + \eta_F \mathbf{r}_F([\![\mathbf{w}_h]\!]))^{\pm}\right) = \left\{\mathbf{F}_v(\mathbf{w}_h, \nabla_h \mathbf{w}_h + \eta_F \mathbf{r}_F([\![\mathbf{w}_h]\!]))\right\}, \tag{10}$$

$\mathbf{r}$ and $\mathbf{r}_F$ are the global and local lifting operators, respectively, and η_F is the stability parameter defined according to Reference 41. The convective numerical flux is computed as a Godunov flux using the exact Riemann solver of Gottlieb and Groth.[42]

2.2 | Implicit time integration

Numerical integration of Equation (9) by means of suitable Gauss quadrature rules leads to a system of nonlinear ODEs that can be written as

$$\mathbf{M_P}(\mathbf{W})\frac{d\mathbf{W}}{dt} + \mathbf{R}(\mathbf{W}) = \mathbf{0}, \tag{11}$$

where $\mathbf{R}(\mathbf{W})$ is the vector of residuals and $\mathbf{M_P}(\mathbf{W})$ is the global block diagonal matrix arising from the discretization of the first integral in Equation (9). For sets of variables different than $\mathbf{w}_c$, the transformation matrix $\mathbf{P}$ couples the degrees of freedom of the variables $\mathbf{w}_h$ within each block of $\mathbf{M_P}$. The high-order accurate time integration performed here is the multi-stage linearly implicit (Rosenbrock-type) Runge–Kutta scheme. This class of schemes requires the solution of a linear system at each stage, while the Jacobian matrix needs to be assembled only once per time step

$$\mathbf{W}^{n+1} = \mathbf{W}^n + \sum_{j=1}^{s} m_j \mathbf{Y}_j, \tag{12}$$

$$\left(\frac{\mathbf{I}}{\gamma \Delta t} + \tilde{\mathbf{J}}\right)^n \mathbf{Y}_i = -\tilde{\mathbf{R}}\left(\mathbf{W}^n + \sum_{j=1}^{i-1} a_{ij}\mathbf{Y}_j\right) + \sum_{j=1}^{i-1} \frac{c_{ij}}{\Delta t}\mathbf{Y}_j, \quad i = 1, \dots, s, \tag{13}$$

where, omitting the dependence on $\mathbf{W}$ for notational convenience,

$$\mathbf{J} = \frac{\partial \mathbf{R}}{\partial \mathbf{W}}, \qquad \tilde{\mathbf{R}} = \mathbf{M}_\mathbf{P}^{-1}\mathbf{R}, \qquad \tilde{\mathbf{J}} = \frac{\partial \tilde{\mathbf{R}}}{\partial \mathbf{W}} = \mathbf{M}_\mathbf{P}^{-1}\left(\mathbf{J} - \frac{\partial \mathbf{M}_\mathbf{P}}{\partial \mathbf{W}}\tilde{\mathbf{R}}\right), \tag{14}$$

and m_i, a_{ij}, c_{ij} are real coefficients. In this work the three stages, third-order (ROS3P) scheme of Lang and Verwer[43] was used; an extended review of several Rosenbrock schemes as well as their coefficients are reported in References 44,45. The Jacobian matrix $\mathbf{J}$ is computed analytically and a preconditioned GMRES method is used to solve Equation (12) at each stage. The code relies on PETSc[46] for the linear solvers and to manage data parallelism.

3 | INGREDIENTS FOR AN EFFICIENT *p*-adaptive IMPLEMENTATION

In this section we list the basics building blocks used for an efficient implementation of the *p*-adaptive capabilities in the compressible version of the DG code MIGALE.

3.1 | Efficient evaluation of the basis functions

Basis functions must be used so many times during operators assembly that their efficient evaluation is mandatory to avoid the reduction of the overall solver performance. In this work a set of orthonormal and hierarchical basis functions has been adopted, built according to Reference 39 by means of the modified Gram-Schmidt (MGS) algorithm. In particular, a set of orthonormal basis functions, $\Phi_K^k = \left\{\phi_i^K\right\}$ with $i \in \left\{1, ..., N_{dof}^K\right\}$ is obtained by applying the MGS procedure to an initial set of monomials, $\hat{\Phi}_K^k = \left\{\hat{\phi}_i^K\right\}$ with $i \in \left\{1, ..., N_{dof}^K\right\}$, defined in a reference frame relocated in the element barycenter and aligned with the principal axes of inertia of K.

The basis functions can be evaluated according to three possible strategies:

- Full storing (|PreShape|): the basis functions and, if needed, their derivatives are evaluated at each Gauss point during pre-processing and stored in memory.
- Pre-computed orthonormalization coefficients (|PreCoef|): the MGS coefficients are evaluated and stored during pre-processing for each element, while the monomials are evaluated and orthonormalized on-the-fly, using the pre-computed coefficients.
- On-the-fly evaluation (|OTF|): both the monomials and the orthonormalization coefficients are computed on-the-fly during assembly.

As shown in Reference 35 for a prototype problem, the computational time increases from the |PreShape| to |OTF|, due to a larger amount of operations performed at runtime, while memory increases from |OTF| to |PreShape|. Although the three strategies are all implemented in our solver, we only use the |PreShape| or the |PreCoef| approaches for real life computations, depending on the choice to optimize the computational time or the allocated memory. We remark that the |PreShape| strategy is particularly suited when using matrix-free implementations of the GMRES iterative linear solver, where the spatially discretized operator, that is, the residual function, needs to be evaluated at each linear system iteration, see Reference 37.

3.2 | A *p*-adaptation strategy for the CAA and CFD

A *p*-adaptation strategy aims at varying the polynomial degree of the solution over the mesh, and the local lack/excess of spatial resolution is adjusted according to some error estimator. CFD and CAA simulations are here based on the same set of governing equations, that is, the Navier–Stokes equations, but different features of the flow field are considered. While we are mainly interested in force distributions and total pressure losses for the CFD applications here considered, we seek to accurately solve the acoustics perturbations over the domain for the CAA applications. However, the main idea behind this work is to rely on, as far as possible, the same framework for both applications to significantly reduce the efforts in code development and maintenance. To this end, we opted for the same error estimator, regardless of the type of application, which combines two contributions: (i) the solution jumps at grid cells interfaces;[32,33] (ii) the decay rate of

the modal coefficients of the polynomial expansion.[34] Although we formally rely on the same estimator, this is calculated from two different solution statistics, computed at runtime, depending on whether we are interested in CFD or CAA aspects. In particular, for the CFD applications we compute the estimator from the time-averaged solution, while for the CAA cases the estimator is computed from the root-mean-square (r.m.s.) of the solution. For the sake of compactness we will refer to these strategies as *p-CFD* and *p-CAA*, respectively.

The r.m.s. of the solution is here stored in terms of coefficients of the polynomial basis. In a modal context, this implies a L^2-projection of the r.m.s. on a finite element space. However, to collect this statistic at runtime we would need to actually project and store the squared solution components. This would imply the use of a much larger polynomial space, that is, $\mathbb{P}_d^{2k}(K) \;\; \forall K \in \mathcal{K}_h$, with a substantial impact on the CPU time and memory usage. Supported by some numerical experiments and considering that this statistic takes only part into the evaluation of the error estimator, we decided to project the r.m.s. on the same polynomial space of the solution, that is, $\mathbb{P}_d^{k}(K) \;\; \forall K \in \mathcal{K}_h$, with significant computational savings.

The jump indicator is calculated as the maximum normalized pressure (mean or r.m.s) difference at the interfaces of the element K

$$\eta_K^{JMP} = \max_{if} \max_{j} \left| \frac{\left(p\left(x_j\right) - p\left(x_j\right)^+ \right)_{if}}{\left(p\left(x_j\right) + p\left(x_j\right)^+ \right)_{if}} \right|, \tag{15}$$

where $p(x_j)$ is the pressure value at the j th surface quadrature point x_j on the if th element's interface and computed with the degrees of freedom of the solution belonging to the cell K, while $p(x_j)^+$ is the pressure value at the same location but computed with the degrees of freedom belonging to the cell sharing the face if with K. Our numerical experiments revealed that this indicator, probably due to its stencil, marks for adaptation large regions of the domain, especially when dealing with low-order solutions.

The spectral decay indicator (SDI) correlates the amplitudes of highest modes of the solution to the amplitude of the total modes, and is defined as

$$\eta_K^{SDI} = \frac{\int_K (p^K - \tilde{p}^K)^2 d\mathbf{x}}{\int_K (p^K)^2 d\mathbf{x}}, \tag{16}$$

$$p^K(\mathbf{x}) = \sum_{l=1}^{N_{dof}^K} W_{p,l} \phi_l^K(\mathbf{x}), \qquad \tilde{p}^K(\mathbf{x}) = \sum_{l=1}^{L_{dof}^K} W_{p,l} \phi_l^K(\mathbf{x}), \tag{17}$$

where $W_{p,l}$ are the coefficients of the modal expansion related to the pressure (mean or r.m.s) variable, N_{dof}^K the number of degrees of freedom associated to the local polynomial degree k_K, and L_{dof}^K the number of degrees of freedom associated to the polynomial degree $k_K - 1$.

Unlike the jump indicator, the SDI does not involve in its definition the solution on the neighboring cells. Moreover, while the jump indicator can be computed for any polynomial degree, the SDI definition is not suited for piece-wise constant approximations, that is, $k_K = 0$.

The error estimator, inspired by Gassner et al.,[33] has been implemented by combining the two indicators as

$$\eta_K^{TOT} = \eta_K^{SDI} + \frac{1}{\max(1, k_K)} \eta_K^{JMP}, \quad \forall K \in \mathcal{K}_h, \tag{18}$$

where, according to its definition and our numerical experiments, η_K^{SDI} is set to 0 for $k_K = \{0, 1\}$. Before the coupling, both indicators η^{JMP} and η^{SDI} are normalized over the domain according to their maximum and minimum values.

The adaptation process is triggered by a simple indicator, *trg*, defined as the norm of the relative increment, computed at each time step, of the vector of the degrees of freedom of the runtime time-averaged solution $\overline{\mathbf{W}}$.

The pseudo code for the *p*-adaptation strategy is reported in the Algorithm 1, where ℓ is the index denoting the ℓ th adaptive cycle. The user-defined parameters for the process are:

- k_{ini}, the value for the uniform initialization of the polynomial degree over the mesh;
- n_{adp}, the maximum number of adaptation cycles;

- tol_{trg}, a threshold tolerance for the activation of the adaptation process;
- $\mathcal{N}_{adp}$, the minimum number of time steps, i_{cyc}, between two adaptation cycles;
- $k_{\max}$, the maximum allowable polynomial degree;
- k_K, the polynomial degree on element K;
- $\mathcal{G}_r$, the percentage of elements with the higher estimated error that are marked for refinement at each adaptation cycle;
- $\mathcal{G}_c$, the percentage of elements with the lower estimated error that are marked for coarsening at each adaptation cycle;
- POS_K, the position of the element K in an array numbered from zero and sorted in increasing order according to the estimator η_K^{TOT} value;
- N_{cyc}, the total number of time steps.

Algorithm 1. p-adaptation algorithm

$\ell = 0$
$j_{cyc} = 1$
$k_K = k_{ini}\ \forall K \in \mathcal{K}_h$
for $i_{cyc} = 1 \rightarrow N_{cyc}$ **do**
  advance the solution in time
  evaluate the runtime time-averaged solution, $\overline{\mathbf{W}}$
  compute trg: the relative increment of $\overline{\mathbf{W}}$
  if ($j_{cyc} \geq \mathcal{N}_{adp}$ **and** $trg \leq tol_{trg}$ **and** $\ell \leq n_{adp}$) **then**
    $\ell \leftarrow \ell + 1$
    if ($\ell = 1$ **and** $k_{ini} = 0$) **then**
      for $K \in \mathcal{K}_h$ **do**
        $k_K \leftarrow 1$
      end for
    else
      compute and normalize the estimators $\eta_K^{TOT}\ \forall K \in \mathcal{K}_h$
      for $K \in \mathcal{K}_h$ **do**
        if $POS_K \geq (1 - \mathcal{G}_r) card(\mathcal{K}_h)$ **then**
          $k_K \leftarrow \min(k_K + 1, k_{\max})$
        else if $POS_K < (\mathcal{G}_c) card(\mathcal{K}_h)$ **then**
          $k_K \leftarrow \max(k_K - 1, 1)$
        end if
      end for
      balance the load among processors via re-partitioning
    end if
    L_2 projection of the solution on the new polynomial space
    $j_{cyc} = 0$
  end if
  $j_{cyc} \leftarrow j_{cyc} + 1$
end for

The algorithm has been written in a general form, as we consider both refining and coarsening. However, all simulations performed in this articles use very coarse meshes, where polynomial degree coarsening is not necessary, that is, $\mathcal{G}_c = 0$. As suggested by numerical experiments, the following initialization strategy has been adopted to optimize the effectiveness and the computational efficiency of the algorithm: the computations start from a piece-wise free-stream solution ($k_{ini} = 0$) and the polynomial degree is increased uniformly to $k_K = 1, \forall K \in \mathcal{T}_h$, during the first adaptation cycle ($\ell = 1$).

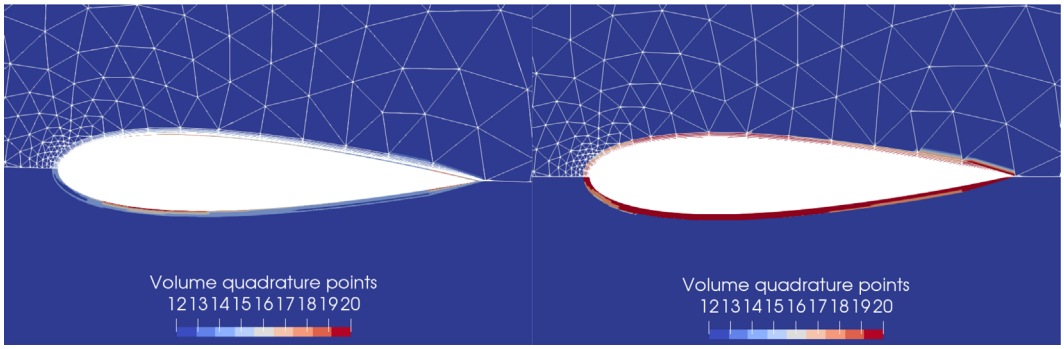

FIGURE 1 Quadrature reduction—Distribution of the number of volume quadrature points for the $tol_q = 10^{-2}$ (left) and $tol_q = 10^{-8}$ (right) case on the NACA0018 airfoil mesh, $\mathbb{P}^6$ approximation [Colour figure can be viewed at wileyonlinelibrary.com]

3.3 | Adaptation of the degree of exactness for quadrature rules

Many approaches for the generation of meshes with curved edges rely on the agglomeration of underlying fine linear meshes and can produce grids accommodating both curved- and straight-sided cells. Following this approach, the linear elements within the mesh are often not known *a priori* from the grid format and cells can be marked as curved although they are (almost) straight-sided. As the number of points needed for an accurate integration drastically increases when dealing with high-order approximations of the solution especially on curved cells, the over-integration of straight-sided elements should be avoided when an efficient implementation of a DG method is pursued. Code MIGALE implements an adaptive procedure to identify the curved elements by using a simple measure for the quadrature error, defined as $\varepsilon_{i,K} = |m^*_{ii} - m^{ex}_{ii}|/|m^{ex}_{ii}|$, where m^*_{ii} and m^{ex}_{ii} are the value of the i th diagonal entry of the local mass matrix ($m_{ii} = \int_{K\in\mathcal{K}_h} \phi_i(\mathbf{x})\phi_i(\mathbf{x})\, d\mathbf{x}$, where $i = 1, \dots, N^K_{dof}$) computed with a "reduced" and an "exact" quadrature rule, respectively. We use the term "exact" when the quadrature formula has a degree of exactness equal to $2k^K + k_m$, where $2k$ represents the shape-shape product within the mass matrix and k^m the polynomial degree of the possibly nonlinear mapping from the reference element, where quadrature points are defined, for example, a unit square, to the mesh face or cell. According to a user-defined tolerance tol_q, an integration rule with the minimum degree of exactness that satisfy $\max_{i\in 1,\dots,N^K_{dof}} \varepsilon_{i,K} \leq tol_q \forall K \in \mathcal{K}_h$ is used on elements (and faces) with significant computational time savings. The quadrature rules adaptation process is performed only once during pre-processing and after each adaptation step of the solution.

A proof of concept for the present algorithm is reported below. The error for the L^2-projection of a given polynomial function with respect to its analytical definition is compared for the "exact" and "reduced" cases when varying the tol_q value. For this test we selected a mesh representative of an external aerodynamics case, that is, the tessellation of a cylindrical domain, with farfield located at $R = 5$, surrounding a NACA0018 wing. The computational grid was generated by extruding a two-dimensional mesh built with the high-order version of an in-house hybrid meshing tool based on the advancing-Delaunay strategy.[47] This coarse mesh is made of 2590 hybrid elements, hexahedra in the boundary layer and prism outside. The mesh generator produces, by design, "real" curved elements only inside the boundary layer while in the rest of the domain high-order points are simply added on straight edges. Figure 1 shows the distribution of the quadrature points number over the mesh, when using the tolerance $tol_q = 10^{-2}$ and $tol_q = 10^{-8}$, and a uniform polynomial degree $k = 6$.

It is worth noting that quadrature rules with the highest degree of exactness are only used in the boundary layer region, where curved elements are created by the mesh generation process. The polynomial function $f(x, y, z) = (x^3y^3z^3)/R^2$ was used to assess the error on the integrals computation due the adaptive reduction of the number of the quadrature points. Table 1 reports the total number of quadrature points for the volume integral and the integral error for different tolerance values tol_q, where $\|err\|_{L^2}$ is calculated with respect to a distribution with an exact quadrature.

These results suggested to use $tol_q = 10^{-10}$, which allows reducing significantly the number of Gauss points without spoiling the integral accuracy. Examples of DNS computations for incompressible flows that benefit of this adaptation strategy are reported in Reference 37.

TABLE 1 Quadrature reduction—L^2-projection test on a the NACA0018 wing mesh

tol_q	#points	$\|err\|_{L^2}$
10^{-2}	$1.01 \cdot 10^6$	$3.37 \cdot 10^{-10}$
10^{-4}	$1.18 \cdot 10^6$	$1.35 \cdot 10^{-12}$
10^{-6}	$1.38 \cdot 10^6$	$1.44 \cdot 10^{-14}$
10^{-10}	$1.79 \cdot 10^6$	$1.48 \cdot 10^{-14}$
10^{-16}	$1.27 \cdot 10^7$	$1.17 \cdot 10^{-14}$

Note: Total number of the volume quadrature points and the integral error for different tol_q values. $\|err\|_{L^2}$ is calculated with respect to a distribution with an exact quadrature.

3.4 | Load balancing

After each adaptation cycle, an imbalance of the computational load among the processes is induced, requiring a technique to evenly re-distribute the effort over the processes. To this purpose, an online balanced re-partitioning of the computational mesh is performed by exploiting the ability of the Metis library[48] to enforce multiple constraints on a weighted graph. In the graph, weights are associated to the vertices and corresponds to the cells of the computational grid. The values of weights are properly set to balance the computational cost of (i) the residual and the Jacobian assembly, (ii) the matrix-vector product of GMRES, (iii) the local number of quadrature points resulting from the adaptive algorithm of Section 3.3. The parallel efficiency of the proposed p-adaptive framework (up to 2178 cores) and the details on the definition of the weights can be found in Reference 49.

3.5 | Non-reflecting sponge-layer

When dealing with CAA, and, in general, with high-fidelity flow simulations, the use of a non-reflecting boundary treatment is mandatory for the correct prediction of some quantities and distributions, for example, the sound directivity. Supported by the results of Colombo and Crivellini,[38] we opted for an absorbing sponge layer approach and extended its implementation to a p-adaptive context. The good performance of this simple numerical approach will be demonstrated in Section 4.1. The non-reflecting treatment consists in explicitly adding to Equation (9) the term

$$-\sum_{K\in\mathcal{K}_h}\int_K \phi_i\sigma\left(w_{c,j}-w^r_{c,j}\right)d\mathbf{x}, \tag{19}$$

where $w_{c,j}$ and $w^r_{c,j}$ are the components of the vector of the conservative variables and the reference state computed from the runtime time-averaged solution, respectively. The sponge layer profile σ is defined as $\sigma = \sigma_0[(\Delta - d)/\Delta]^2$ where Δ is the layer width, d the distance from the free-stream boundary, σ_0 the maximum value. Moreover, the strength of the sponge-layer S_t is an integral measure of σ and can be related to a sponge target-damping parameter η_{target} measured in decibels (dB). All the details on this reflection-absorbing method and its parameters are presented in References 38,50. Finally, as this damping term only depends on elemental unknowns without coupling with neighbor elements, we remark that the implicit implementation of the method is strongly simplified.

4 | NUMERICAL RESULTS

The simulations for all the testcases have been initialized by computing, in sequence, uniform $\mathbb{P}^0$ and $\mathbb{P}^1$ solutions. After initialization, the adaptation process was activated with $\mathcal{G}_c = 0$, that is, disabling coarsening, and $\mathcal{G}_r = 0.2$ to refine the 20% of elements with the highest estimated error values. An adaptation cycle is triggered according to the tolerance $tol_{trg} = 0.02$ and a minimum number of time-steps between two adaptation cycles $\mathcal{N}_{adp}$ corresponding to 5 Tc, where Tc is a reference convective time defined according to the freestream velocity u_∞ and the reference length (D or C). The maximum allowable polynomial degree was $k_{\max} = 5$ for the cylinders test cases and $k_{\max} = 3$ for the SD7003 airfoil.

Span-wise periodicity is assumed for all testcases. With the exception of the CFD cylinder testcase, we remark that we did not increased the mesh density in the wake region, as we aim at automatically refining this region by placing higher-order cells.

The computations have been integrated in time with the ROS3-3 scheme of Lang et al.[43] and performed using 15 KNL nodes, with 68 CPUs each, on the MARCONI A2 HPC system provided by CINECA, the Italian Supercomputing Center.

4.1 | CAA test cases

The accuracy and efficiency of the proposed adaptation strategies have been assessed on a canonical CAA test case, that is, the flow around a circular cylinder at Reynolds number $\mathrm{Re}_D = 100$ and two different Mach numbers, that is, $\mathrm{M}_\infty = 0.15$ and $\mathrm{M}_\infty = 0.5$. These testcases deal with the prediction of the noise generated by the vortex shedding behind the cylinder. The flow problem was also used to demonstrate the reliability of a sponge layer non-reflecting treatment extended to a high-order p-adaptive discretization. Colombo and Crivellini[38] performed a parametric study to assess the performance of this technique in case of uniform-order discretizations. The results indicated that a theoretically exponential damping for waves travelling within the layer can be achieved by properly setting the sponge width. This section will demonstrate that also in a p-adaptive framework, where a discretization coarsening could be naturally considered as an attempt to damp oscillations, the use of a non-reflecting boundary treatment is mandatory for the correct prediction of those quantities, like sound directivity, which would be spoiled by the boundary reflections. Finally, in this section the effects of the time-step, which is defined as a fraction f of the convective time, that is, $\Delta t = f\,(D/u_{\mathrm{ref}})$ (with D the circular cylinder diameter), and of the CAA and CFD error indicators on the computed results are investigated. To compare the error indicators, the lack of symmetry for the r.m.s. of pressure fluctuations is considered, and it is defined as $err_{r.m.s} = 1/n \sum_i^n \left| p'_{rms,i}|_{y>0} - p'_{rms,i}|_{y<0} \right|$.

4.1.1 | Flow past a circular cylinder, $\mathrm{Re}_D = 100$, $\mathrm{M}_\infty = 0.15$

The first CAA case is the laminar flow around the circular cylinder at $\mathrm{M}_\infty = 0.15$ and $\mathrm{Re}_D = 100$. This regime is characterized by an alternate vortex shedding behind the cylinder. As acoustic pulses radiate from the body, an appropriate non-reflecting treatment for boundary conditions is needed to correctly predict the sound dipole.

A hybrid unstructured mesh with a circular farfield boundary located at $40D$ from the cylinder center with 10,206 quadratic prismatic and hexahedral elements has been used. The two dimensional section of the mesh is made of 5103 elements, as in Reference 38, but it was extruded with two layer in the spanwise direction with periodic boundary conditions. The mesh and the near wall detail are shown in Figure 2. All simulations have been advanced in time with a time-step equals to a fraction $f = 0.1628$ of the convective time. The sponge layer has been used with $\eta_{target} = 40$ and $\Delta = 20D$, as suggested in Reference 38. In fact, the adopted η_{target} value reduces the incident sound waves by a factor 100 that can be

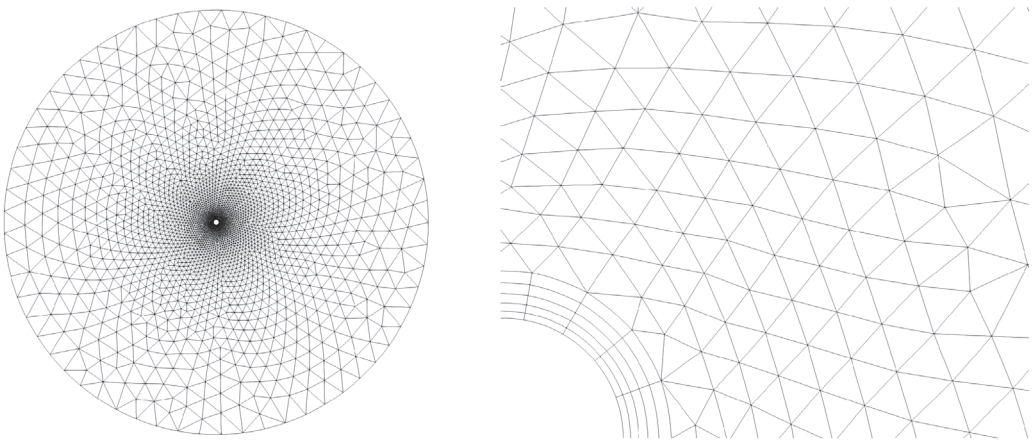

FIGURE 2 Laminar flow past a circular cylinder $M_\infty = 0.15$ —The computational mesh consists of 10, 206 triangular and quadrilateral elements with quadratic edges with a circular farfield boundary located at $40D$

considered appropriate for a practical use. After achieving a periodic behavior of the force coefficients, time-averaged and r.m.s. statistics have been computed on a period $\sim 35fL/u_{\text{ref}}$, where $L = 2 \times 40D$ is the domain extension.

Simulations have been performed with uniform polynomial degree discretizations, *DG*-$\mathbb{P}^3$ ($2.04 \cdot 10^5$ DoFs) and *DG*-$\mathbb{P}^4$ ($3.57 \cdot 10^5$ DoFs), and with both the *p-CFD* and *p-CAA* adaptation strategies. The total number of adaptation cycles performed during each computation was set to $n_{adp} = 6$ to reach a comparable number of DoFs per equation at the end of the process, that is, $1.24 \cdot 10^5$ and $1.33 \cdot 10^5$ DoFs for *p-CFD* and *p-CAA*, respectively. Both error estimation strategies, that is, based on the time-averaged pressure or the r.m.s. of the pressure fluctuation, were able to refine the discretization in the shear layer and in the wake regions. However, high-order polynomial cells are differently distributed over the domain, as shown in Figure 3A,C. In particular, we observed that *p-CAA* places high-order cells even upstream of the cylinder, differently from *p-CFD*. Moreover, Figure 3B,D show the base-ten logarithm of the r.m.s. pressure fluctuations contour for *p-CAA* and *p-CFD* strategies. The former allows to predict a smoother distribution of the r.m.s, while the latter show some oscillations in the wake and upstream of the cylinder.

Figure 4 shows the pressure fluctuation p'_{rms}, measured in dB, as a function of the θ angle on a circle centered in $\{x/D, y/D\} = \{0, 0\}$ with radius $r/D = 30$. Figure 4A shows the distribution computed by extending the solutions at radius $r/D = 15$, according to the inverse proportionality between pressure waves amplitude and the square root of the distance, while in Figure 4b the results are directly computed at $r/D = 30$. All the computations are in reasonable agreement with Desquesnes et al. [51] and Colombo et al. [38] The p -adaptive strategy exploits only $\sim 65\%$ of DoFs used for the uniform $\mathbb{P}^3$ computation. Moreover, *p-CAA* approach provides smoother profiles than *p-CFD*, especially at stagnation point.

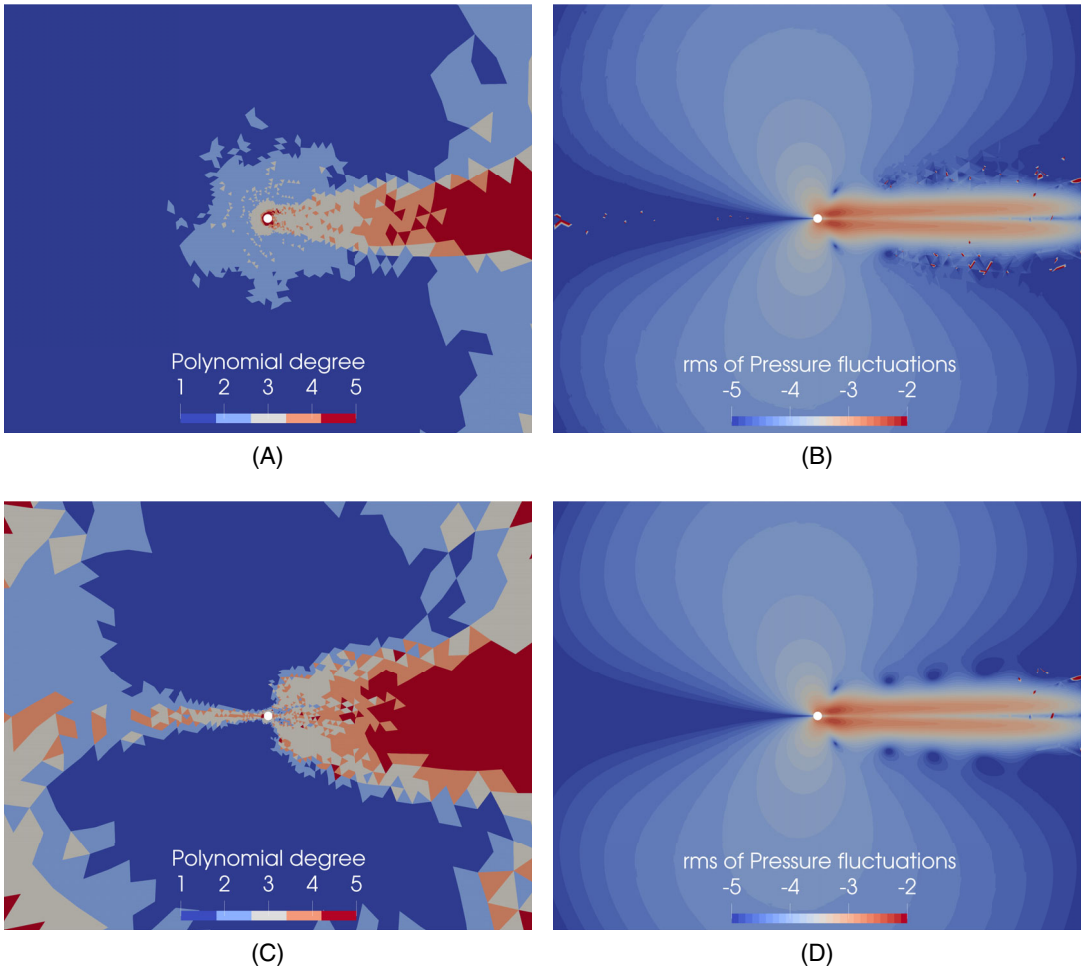

FIGURE 3 Laminar flow past a circular cylinder $M = 0.15$ —Polynomial degree distribution and base-ten logarithm of r.m.s. pressure fluctuations, 20 contour levels from −5 to −2, for *p-CFD* ($1.24 \cdot 10^5$ DoFs) and *p-CAA* ($1.32 \cdot 10^5$ DoFs), *DG*-$\mathbb{P}^{1\to5}$ solution. (A) Polynomial degree distribution, *p-CFD*; (B) base-ten logarithm of r.m.s. pressure fluctuations, *p-CFD*; (C) polynomial degree distribution, *p-CAA*; (D) base-ten logarithm of r.m.s. pressure fluctuations, *p-CAA* [Colour figure can be viewed at wileyonlinelibrary.com]

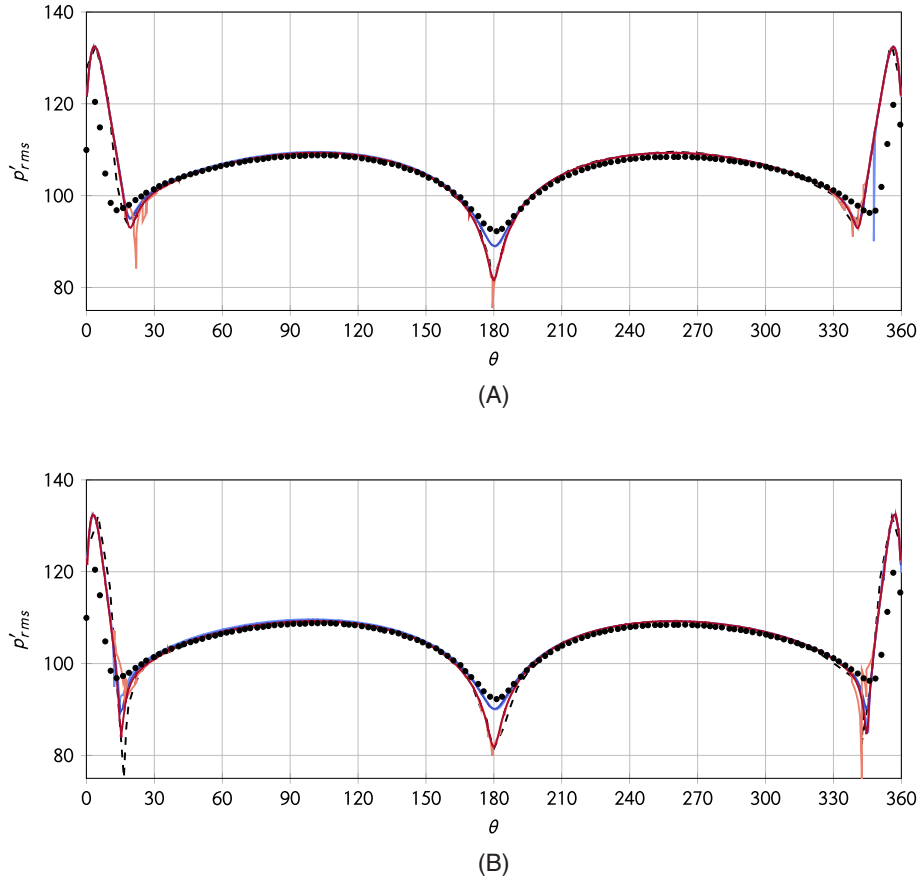

FIGURE 4 Laminar flow past a circular cylinder $M = 0.15$ —Directivity plot of sound, spanwise-averaged pressure fluctuation p'_{rms} in dB, located on a circle centered in $\{x/D, y/D\} = \{0, 0\}$ with radius $r/D = 30$. • Desquesnes et al., [51] − − − Colombo and Crivellini, [38] ___ *DG*-$\mathbb{P}^3$ ($2.04 \cdot 10^5$ DoFs), ___ *DG*-$\mathbb{P}^4$ ($3.57 \cdot 10^5$ DoFs), ___ *DG*-$\mathbb{P}^{1\to5}$ *p-CFD* ($1.24 \cdot 10^5$ DoFs), ___ *DG*-$\mathbb{P}^{1\to5}$ *p-CAA* ($1.33 \cdot 10^5$ DoFs). (A) Distribution at $r/D = 30$, extended from the adaptive computation at $r/D = 15$ according to the inverse proportionality between pressure waves amplitude and the square root of the distance (B) Distribution at $r/D = 30$ from the adaptive computation [Colour figure can be viewed at wileyonlinelibrary.com]

It is worth noting that the non-reflecting treatment was able to deliver a solution without pressure waves reflections, and the adaptive algorithm marks for refinement also cells in the sponge layer region. Finally, the lack of symmetry for the r.m.s. of pressure fluctuations, $err_{r.m.s.}$ (with $n = 1000$), is evaluated and tabulated in Table 2. The measure was computed on circles centered in $\{x/D, y/D\} = \{0, 0\}$ with radii $r/D = \{10, 15, 20, 30\}$. The *p-CAA* algorithm delivers the best solution in terms of symmetry (also with respect to *p-CFD*) with a number of DoFs roughly one order of magnitude lower than $\mathbb{P}^{3,4}$ computations.

TABLE 2 Laminar flow past a circular cylinder $M = 0.15$—Evaluation of the non-symmetric behavior of the r.m.s. pressure fluctuations, $err_{r.m.s.}$, for *p-CFD*and *p-CAA*, *DG*-$\mathbb{P}^{1\to5}$, *DG*-$\mathbb{P}^3$, and *DG*-$\mathbb{P}^4$solutions on circles centered in $\{x/D, y/D\} = \{0, 0\}$with radii $r/D = \{10, 15, 20, 30\}$

	DoFs	$r/D = 10$	$r/D = 15$	$r/D = 20$	$r/D = 30$
p-CAA	132, 000	3.37e-7	2.98e-7	2.89e-7	2.06e-7
p-CFD	124, 000	1.12e-6	7.30e-7	5.87e-7	5.34e-7
DG - $\mathbb{P}^3$	204, 000	1.76e-6	2.17e-6	2.80e-6	3.24e-6
DG - $\mathbb{P}^4$	357, 000	1.61e-6	1.67e-6	1.63e-6	1.68e-6

4.1.2 | Flow past a circular cylinder, $\mathrm{Re}_D = 100$, $\mathrm{M}_\infty = 0.5$

The flow around a circular cylinder at $\mathrm{Re}_D = 100$ and $\mathrm{M}_\infty = 0.5$ was computed to assess the influence of Mach number on the combined performance of the non-reflecting sponge layer and the *p*-adaptive strategy.

The hybrid mesh is made of 13,042 prismatic and hexahedral elements with quadratic edges and a circular farfield boundary located at $35D$. The two dimensional section of the mesh is made of 6521 elements and is the same used in Reference 38 here extruded with two layer in the spanwise direction. The time-averaged and r.m.s. solutions have been computed on a period $\sim 60fL/u_{\mathrm{ref}}$. The sponge layer has been used with $\eta_{target} = 40$ and $\Delta = 20D$, as suggested in Reference 38. The total number of adaptation cycles performed has been set to $n_{adp} = 6$ to guarantee a comparable number of DoFs per equation at the end of the process, that is, $1.54 \cdot 10^5$ and $1.76 \cdot 10^5$ DoFs for *p-CFD* and *p-CAA*, respectively. Simulations have been also performed with uniform *DG* - $\mathbb{P}^3$ elements discretization, resulting in $2.60 \cdot 10^5$ DoF, $\sim$59% more DoF than the adaptive cases.

Figure 5 shows the polynomial degree distribution and the base-ten logarithm of r.m.s. pressure fluctuations contour for both the *p-CFD* (Figure 5A,B) and *p-CAA* (Figure 5C,D). The polynomial degree distributions are quite different. In particular, *p-CAA* algorithm uses high-order polynomial degree cells upstream of the cylinder to accurately capture the interaction between waves of opposite sign, generated above and below the cylinder (see Figure 6). Both methods mark for adaptation the wake, where strong perturbations characterized by a short wave length occur. However, this region is ineffective in sound radiation and relative larger wave lengths, which are not constant for the Doppler effect, occur in the

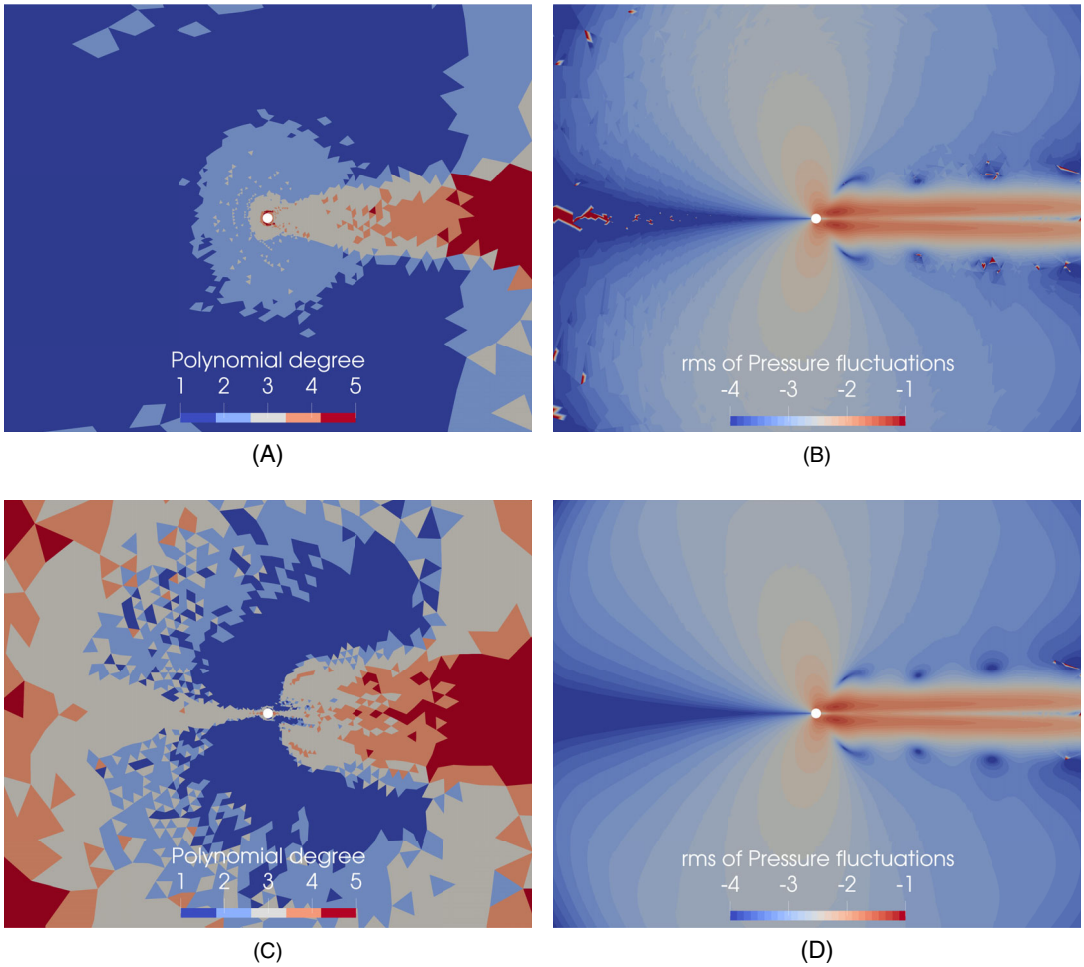

FIGURE 5 Laminar flow past a circular cylinder $M = 0.5$ —Polynomial degree distribution (left) and base-ten logarithm of r.m.s. pressure fluctuations, 20 contour levels from -4 to -1, for *p-CFD* ($1.54 \cdot 10^5$ DoF) and *p-CAA* ($1.76 \cdot 10^5$ DoF), *DG*-$\mathbb{P}^{1\to5}$ solution, $f = 0.1628$. (A) Polynomial degree distribution, *p-CFD*; (B) base-ten logarithm of r.m.s. pressure fluctuations, *p-CFD*; (C) Polynomial degree distribution, *p-CAA* (D) base-ten logarithm of r.m.s. pressure fluctuations, *p-CAA* [Colour figure can be viewed at wileyonlinelibrary.com]

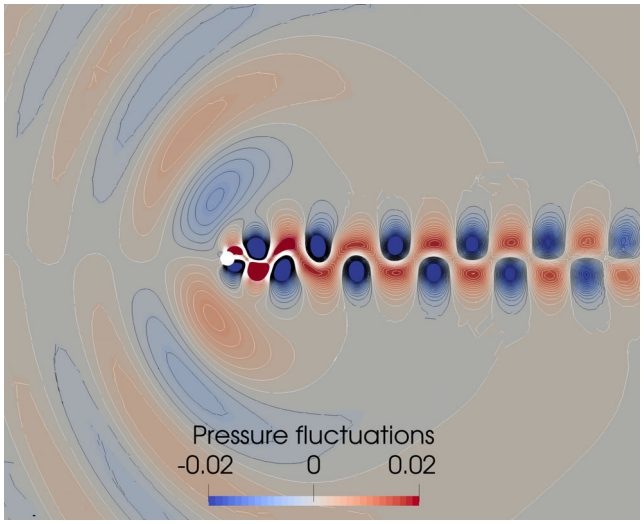

FIGURE 6 Laminar flow past a circular cylinder $M = 0.5$ —Pressure fluctuations, 32 contour levels from −0.02 to +0.02, *p-CAA* with *DG*-$\mathbb{P}^{1\to5}$ solution, $f = 0.1628$ [Colour figure can be viewed at wileyonlinelibrary.com]

rest of the domain. These perturbations are caused by the pressure fluctuations induced by vortex shedding, that is, the problem is characterized by a dipole source, see Reference 52.

To assess the influence of the time-step on the adaptation strategy, the flow fields obtained with two different values of f ($f = 0.1628$ and $f = 0.0814$) are compared in Figure 7. Notice that the final adapted polynomial degree distribution is similar for both time-steps, as confirmed by the number of DoFs, $1.72 \cdot 10^5$ and $1.75 \cdot 10^5$ for $f = 0.0814$ and $f = 0.1628$, respectively. As expected, also the r.m.s. pressure fluctuations contours are almost identical.

Figure 8 shows the pressure fluctuation p'_{rms} as a function of the θ angle on a circle centered in $\{x/D, y/D\} = \{0, 0\}$ with radius $r/D = 30$ and a detail of the back of the cylinder. The uniform $\mathbb{P}^3$ simulation exhibits spurious oscillations due to the lack of spatial accuracy far from the body, while adaptive results, especially the *p-CAA* algorithm, are in a reasonable agreement with Colombo and Crivellini.[38]

In this section an alternative approach to the sponge layer to damp spurious oscillations incoming from the farfield boundary is investigated. In fact, in an adaptive solver a straightforward and simple approach to damp oscillation can be represented by the reduction of the local polynomial degree far from the body. However, this approach demonstrated not to perform properly, as shown in Figure 9, where the polynomial degree is reduced to one in the region corresponding to the sponge layer, but without using any damping term. Spurious oscillations can be observed, as for the solution without the anti-reflecting treatment shown in Figure 10. For both cases, the adaptive algorithm refines the local polynomial degree where non-physical oscillations occur. The results demonstrate that the *p-CAA* algorithm is better suited for CAA flow problems than the *p-CFD* alternative and that a proper treatment for possible boundary reflections is mandatory also in a *p*-adaptive context.

As for the $M_\infty = 0.15$ case, a measure of the lack of symmetry of the r.m.s. of pressure fluctuations, $err_{r.m.s.}$ (with $n = 18,000$), is evaluated on circles centered in $\{x/D, y/D\} = \{0, 0\}$ with radii $r/D = \{10, 20, 30\}$ for *p-CAA* and *p-CFD* , after six adaptation cycles and with $f = 0.1628$. Figure 11 shows the behavior of $err_{r.m.s}$ as a function of the DoFs (corresponding to different values of $n_{adp} = \{4, 5, 6\}$). The results are summarized in Table 3, where uniform $\mathbb{P}^3$ computations with the same value of f are also reported. The *p-CAA* algorithm confirmed to deliver the overall best solution in terms of symmetry with less DoFs than the uniform $\mathbb{P}^3$ computation. Moreover, Table 3 reports also the value of $err_{r.m.s.}$ with *p-CAA* for different values of n_{adp}, and of the fraction, f, of the convective time. Notice that already with $n_{adp} = 4$, a lower lack of symmetry with respect to *DG*-$\mathbb{P}^3$ is obtained at $r/D = \{20, 30\}$ with less than 50% of the DoFs. Also in this analysis the influence of the time-step seems to be negligible. Finally, Table 4 reports a comparison between the most promising approach for CAA, that is, the *p-CAA*, and the uniform $\mathbb{P}^3$ computation, which is taken as reference, both in terms of required DoFs and computing time for different values of the adaptation cycles, n_{adp}, and $f = 0.1628$. For all values of n_{adp} we can observe a saving both in terms of DoFs and computing time. In particular, the results with $n_{adp} = 4$ show a reduction of roughly the 60% of the required DoFs and computing time, but with higher symmetry with respect to the uniform $\mathbb{P}^3$ computation at $r/D = \{20, 30\}$.

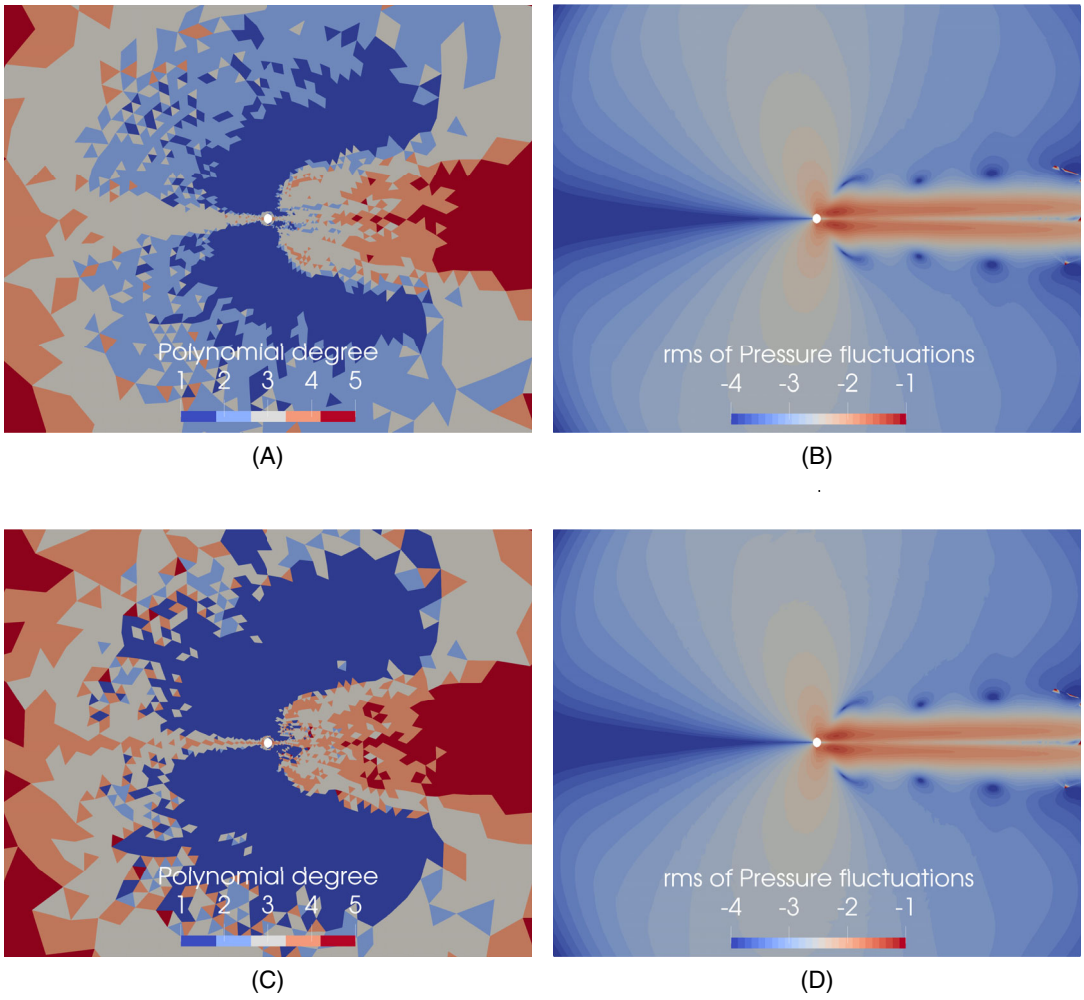

FIGURE 7 Laminar flow past a circular cylinder $M = 0.5$ —Polynomial degree distribution and base-ten logarithm of r.m.s. pressure fluctuations, 20 contour levels from -4 to -1, for *p-CAA* with $f = 0.0814$ ($1.72 \cdot 10^5$ DoFs) and $f = 0.1628$ ($1.75 \cdot 10^5$ DoFs), *DG* - $\mathbb{P}^{1\to5}$ solution, $n_{adp} = 6$. (A) Polynomial degree distribution, *p-CAA*, $f = 0.0814$; (B) base-ten logarithm of r.m.s. pressure fluctuations, *p-CAA*, $f = 0.0814$; (C) Polynomial degree distribution, *p-CAA*, $f = 0.1628$; (D) base-ten logarithm of r.m.s. pressure fluctuations, *p-CAA*, $f = 0.1628$ [Colour figure can be viewed at wileyonlinelibrary.com]

4.2 | CFD test cases

The accuracy and efficiency of the *p-CFD* adaptive strategy have been assessed by computing the ILES of: *i*) the flow past a circular cylinder at Reynolds number $\mathrm{Re}_D = 3900$; (ii) the flow around a SD7003 airfoil at $\mathrm{Re}_c = 60,000$ and angles of attack $\alpha = \{4°, 8°\}$.

4.2.1 | Flow past a circular cylinder, $\mathrm{M}_\infty = 0.1$, $\mathrm{Re}_D = 3900$

The transitional flow around a circular cylinder at Mach number $\mathrm{M}_\infty = 0.1$ and $\mathrm{Re}_D = 3900$ is a very common flow problem used as benchmark in the scientific community. This test case was also part of the suite of the International Workshop on high-order methods.[53]

The simulations have been performed on two different meshes made of 44,856 (mesh A) and 67,466 (mesh B) quadratic elements, hexahedra in the boundary layer and prism outside. Mesh A has been generated without wake refinement, see Figure 12, to assess the performance of the adaptive strategy in recovering this lack of spatial resolution. A periodic boundary condition over a width $s = 2D$ is set in the spanwise direction to mimic an infinite domain discretized

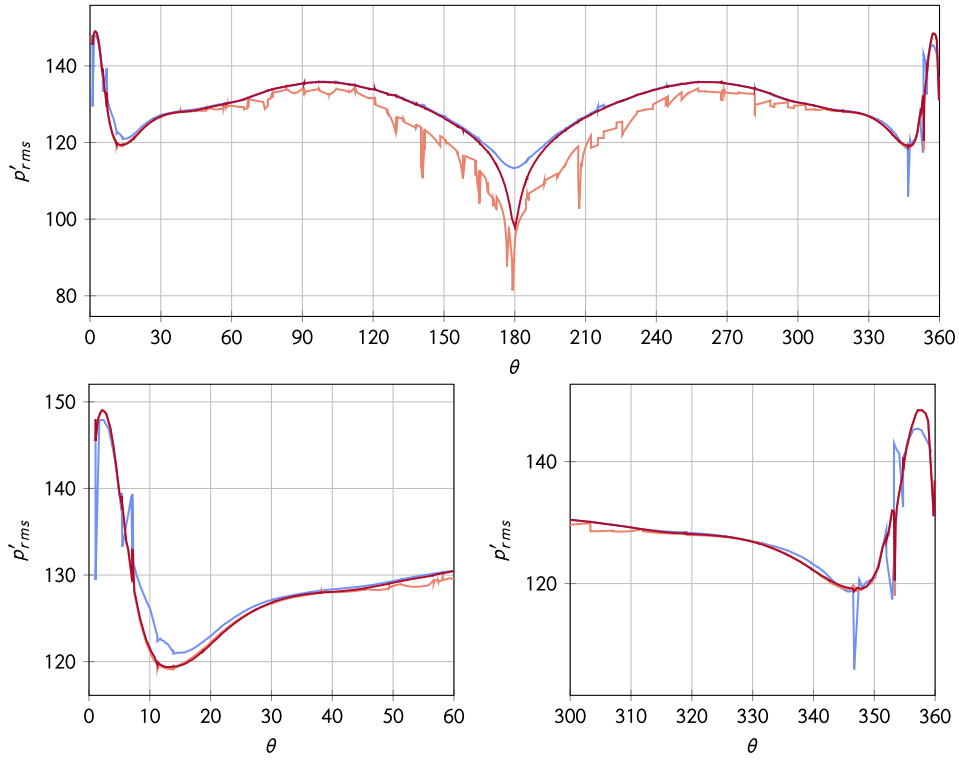

FIGURE 8 Laminar flow past a circular cylinder $M = 0.5$ —Plot of the sound directivity, pressure fluctuation p'_{rms} in dB, located on a circle centered in $\{x/D, y/D\} = \{0, 0\}$ and radius $r/D = 30$. —— *DG*-$\mathbb{P}^3$ ($2.60 \cdot 10^5$ DoFs), —— *DG*-$\mathbb{P}^{1\to5}$ *p-CFD* ($1.54 \cdot 10^5$ DoFs), —— *DG*-$\mathbb{P}^{1\to5}$ *p-CAA* ($1.76 \cdot 10^5$ DoFs) [Colour figure can be viewed at wileyonlinelibrary.com]

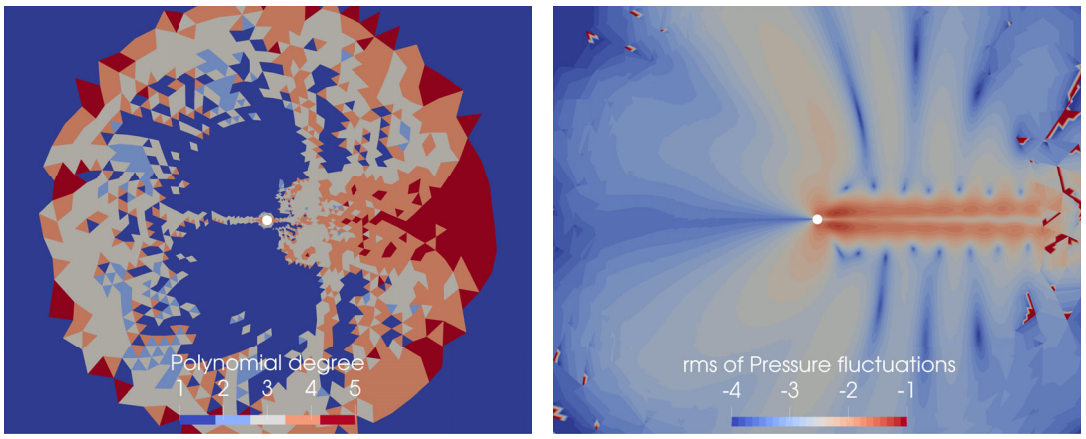

FIGURE 9 Laminar flow past a circular cylinder $M = 0.5$ —Polynomial degree distribution and base-ten logarithm of r.m.s. pressure fluctuations, 20 contour levels from -4 to -1, (right) for *p-CAA* ($1.72 \cdot 10^5$ DoFs), *DG*-$\mathbb{P}^{1\to5}$ solution without anti-reflecting treatment and using $\mathbb{P}^1$ elements only in the outer region [Colour figure can be viewed at wileyonlinelibrary.com]

by 14 elements. At the circular farfield, located at 50 D from the cylinder, characteristic-based boundary conditions are imposed.

The maximum number of adaptation cycles has been set equal to $n_{adp} = 7$ and to $n_{adp} = 6$ for the mesh A and mesh B, respectively. These values guarantee a comparable number of DoFs per equation at the end of the process, that is, 653, 679 and 681, 327 for mesh A and B, respectively. The final polynomial degree distribution has been used to compute the average solution over 100 shedding cycles, see Figures 12 and 13. On both meshes the adaptive algorithm refines the spatial discretization in the shear layer and wake regions of the domain. However, as expected, the high-order polynomials

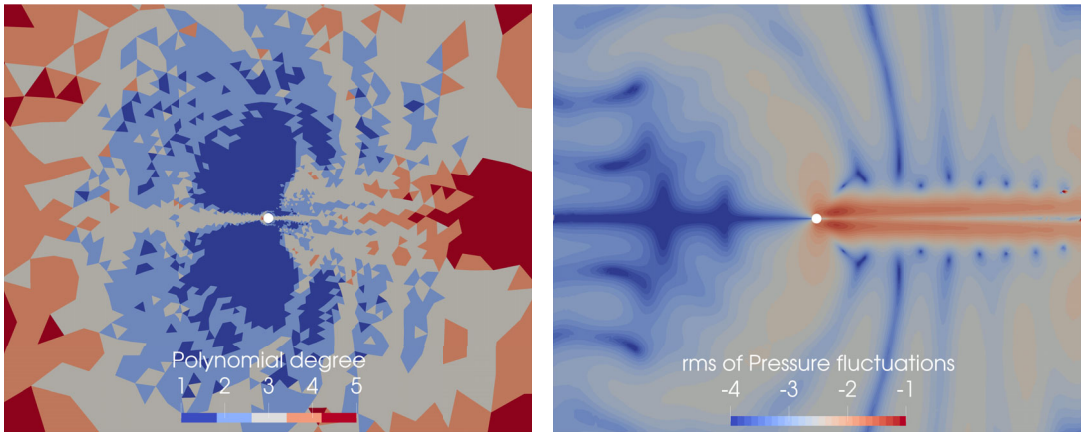

FIGURE 10 Laminar flow past a circular cylinder $M = 0.5$ —Polynomial degree distribution and base-ten logarithm of r.m.s. pressure fluctuations, 20 contour levels from -4 to -1, (right) for *p-CAA* ($1.58 \cdot 10^5$ DoFs), *DG*-$\mathbb{P}^{1\to5}$ solution without using the sponge layer [Colour figure can be viewed at wileyonlinelibrary.com]

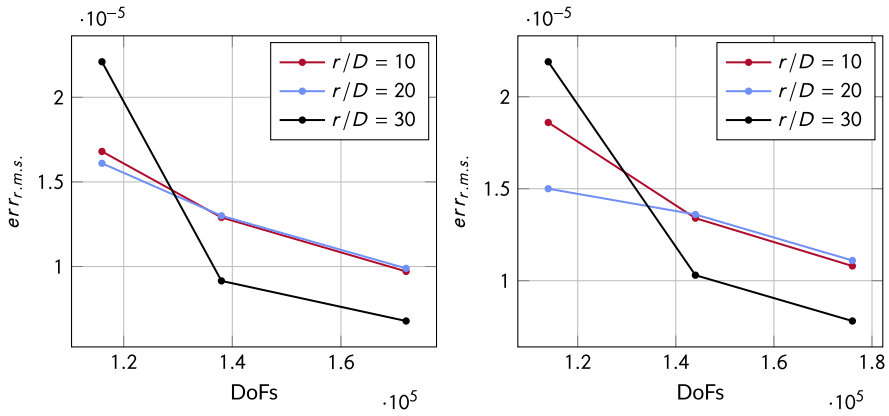

FIGURE 11 Laminar flow past a circular cylinder $M = 0.5$ —Evaluation of the non-symmetric behavior of the r.m.s. pressure fluctuations, $err_{r.m.s.}$, for *p-CAA* and different $n_{adp} = 4, 5, 6$ values and fractions f, that is, $f = 0.0814$ (left) and $f = 0.1628$ (right), of the convective time, on circles centered in $\{x/D, y/D\} = \{0, 0\}$ with radius $r/D = \{10, 20, 30\}$ [Colour figure can be viewed at wileyonlinelibrary.com]

TABLE 3 Laminar flow past a circular cylinder $M = 0.5$—Evaluation of the non-symmetric behavior of the r.m.s. pressure fluctuations, $err_{r.m.s.}$, for *p-CAA*, *p-CFD* and *DG*-$\mathbb{P}^3$with different n_{adp}, and fraction $f$of the convective time

Method	n_{adp}	f	DoFs	$r/D = 10$	$r/D = 20$	$r/D = 30$
p-CAA	4	0.0814	116,000	1.68e-05	1.61e-05	2.21e-05
p-CAA	5	0.0814	138,000	1.29e-05	1.30e-05	9.15e-06
p-CAA	6	0.0814	172,000	9.71e-06	9.89e-06	6.78e-06
p-CAA	4	0.1628	114,000	1.86e-5	1.50e-5	2.19e-5
p-CAA	5	0.1628	144,000	1.34e-5	1.36e-5	1.03e-5
p-CAA	6	0.1628	176,000	1.08e-5	1.11e-5	7.81e-6
p-CFD	6	0.1628	154,000	1.11e-5	2.19e-5	3.67e-5
DG-$\mathbb{P}^3$	-	0.1628	260,000	8.10e-6	1.85e-5	3.05e-5

TABLE 4 Laminar flow past a circular cylinder $M = 0.5$—Computational time measured over 10 time-steps for *p-CAA*and different n_{adp}in comparison with uniform $\mathbb{P}^3$ computation (taken as a reference), *DG*-$\mathbb{P}^{1\to5}$ solution, $f = 0.1628$

Method	n_{adp}	DoFs/DoFs$_r$(%)	Time/time$_r$(%)
p-CAA	4	43.8	42.1
p-CAA	5	55.4	60.5
p-CAA	6	67.7	86.7
DG-$\mathbb{P}^3$	-	1	1

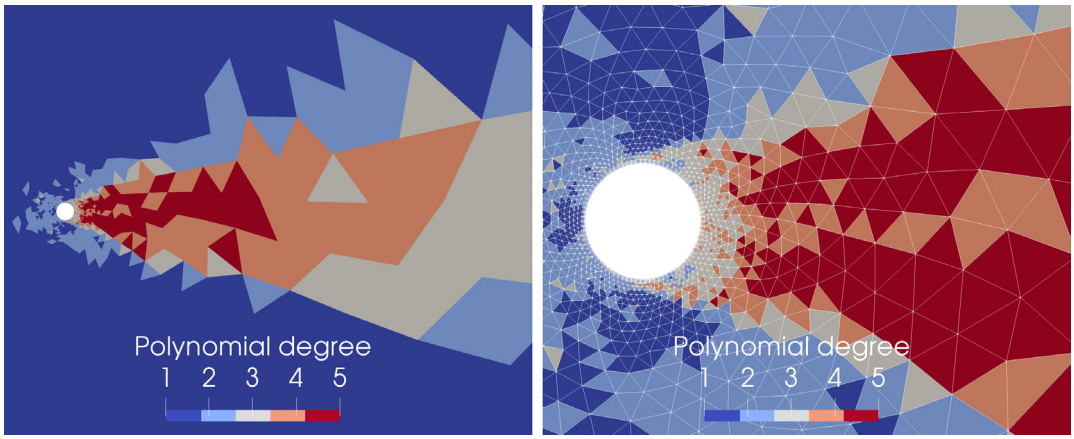

FIGURE 12 Flow around a circular cylinder—Polynomial degree distribution over mesh A ($6.53 \cdot 10^5$ DoFs), $\mathbb{P}^{1\to5}$ solution [Colour figure can be viewed at wileyonlinelibrary.com]

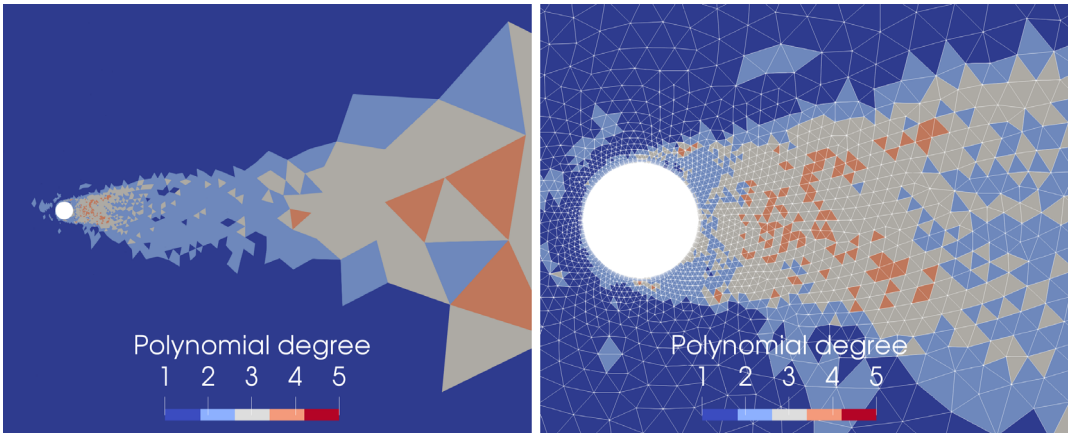

FIGURE 13 Flow around a circular cylinder—Polynomial degree distribution over mesh B ($6.81 \cdot 10^5$ DoFs), $\mathbb{P}^{1\to5}$ solution [Colour figure can be viewed at wileyonlinelibrary.com]

cells are placed differently on mesh A and mesh B. On mesh A (Figure 12), characterized by coarse cells in the wake region, high-order polynomials are uniformly distributed in order to balance the initial lack of spatial resolution. On mesh B (Figure 13) the error indicator localizes high-order elements more selectively in the wake region and shear layer, consistently with the target overall number of degrees of freedom. Figure 14 shows the instantaneous Mach and pressure coefficient contours for the two meshes.

Figure 15 shows the spanwise- and time-averaged pressure coefficients c_p and the non-dimensional wall vorticity $\Omega/2Re^{0.5}$ distribution on the cylinder, which compare favorably with experimental data from Norberg[54] and other

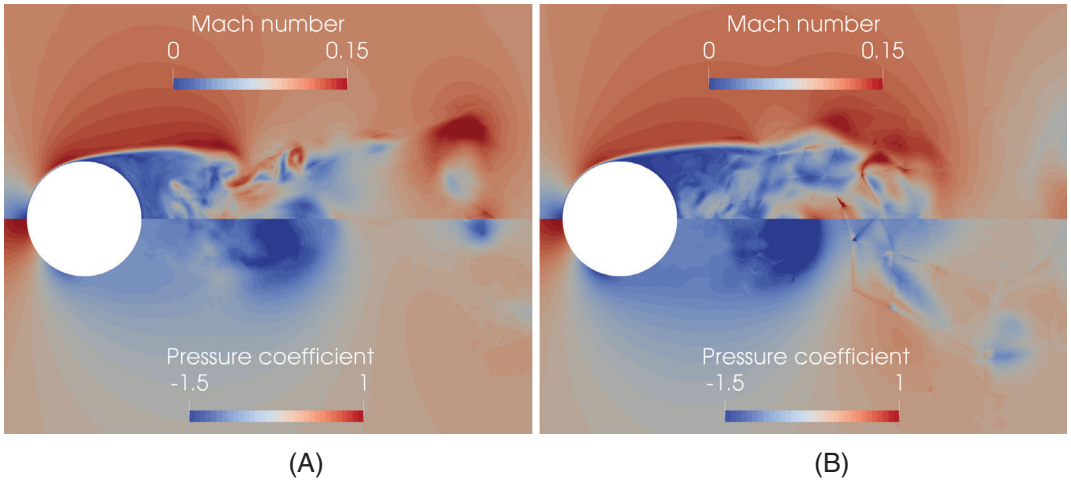

FIGURE 14 Flow around a circular cylinder—Instantaneous Mach number and pressure coefficient contours for (A) mesh A and (B) mesh B, $\mathbb{P}^{1\to5}$ solution [Colour figure can be viewed at wileyonlinelibrary.com]

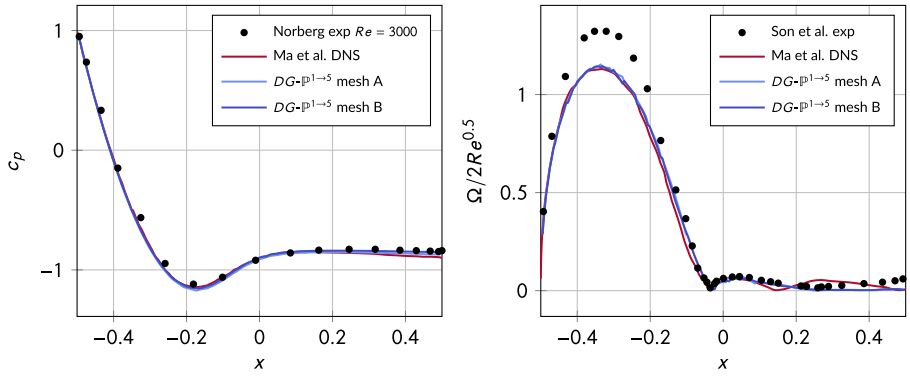

FIGURE 15 Flow around a circular cylinder—Spanwise and time-averaged pressure coefficients c_p (left) and non-dimensional wall vorticity $\Omega/2Re^{0.5}$ (right) distribution on the cylinder, $\mathbb{P}^{1\to5}$ solution (Ma et al. DNS $\sim 100 \cdot 10^6$ DoFs[56]) [Colour figure can be viewed at wileyonlinelibrary.com]

numerical simulations.[55,56] Thanks to the proposed p-adaptation strategy the computed distributions can be considered almost mesh independent.

Figure 16 shows the averaged streamwise u_x/u_{ref} and crosswise u_y/u_{ref} velocity profiles at different locations in the wake of the cylinder $x/D = \{1.06, 1.54, 2.02\}$, while Figure 17 the streamwise u_x/u_{ref} velocity along the centerline in the wake of the cylinder. The results have been compared with the experiments from Parnaudeau et al.[57] and Ong et al.,[58] and other numerical results.[55,59,60] These quantities confirm a good matching between experimental and our numerical results. Furthermore, comparing the results of grids A and B, an almost mesh independent behavior is shown with the exception of the first station. At the first station, the polynomial distribution of the mesh A seems to guarantee a better resolution.

Figure 18 extends the comparison to the spanwise- and time-averaged streamwise $u'_x u'_x/u^2_{\text{ref}}$ and crosswise $u'_y u'_y/u^2_{\text{ref}}$ velocity fluctuations, showing a good agreement with the results from the literature. Finally, Figure 19 shows the contours of the time-averaged x -component of the velocity fluctuation.

Table 5 compares the results for mesh A and B with the references in terms of the root-mean-square lift coefficient $c_{L\,rms}$, the time-averaged drag coefficient c_D, the Strouhal number St, the time-averaged separation angles θ_{sep}, the time-averaged base suction coefficient $-c_{pb}$, the time-averaged peak of the streamwise velocity in the wake $-u_{\min}/u_{\text{ref}}$ and the time-averaged recirculation zone length L_r/D. The time-averaged separation angle θ_{sep} is calculated from the condition of vanishing wall-shear stress in the time-averaged field around the cylinder, while the time-averaged base suction coefficient $-c_{pb}$ as the time-averaged pressure coefficient on the cylinder's surface at $\theta = 180°$. The time-averaged

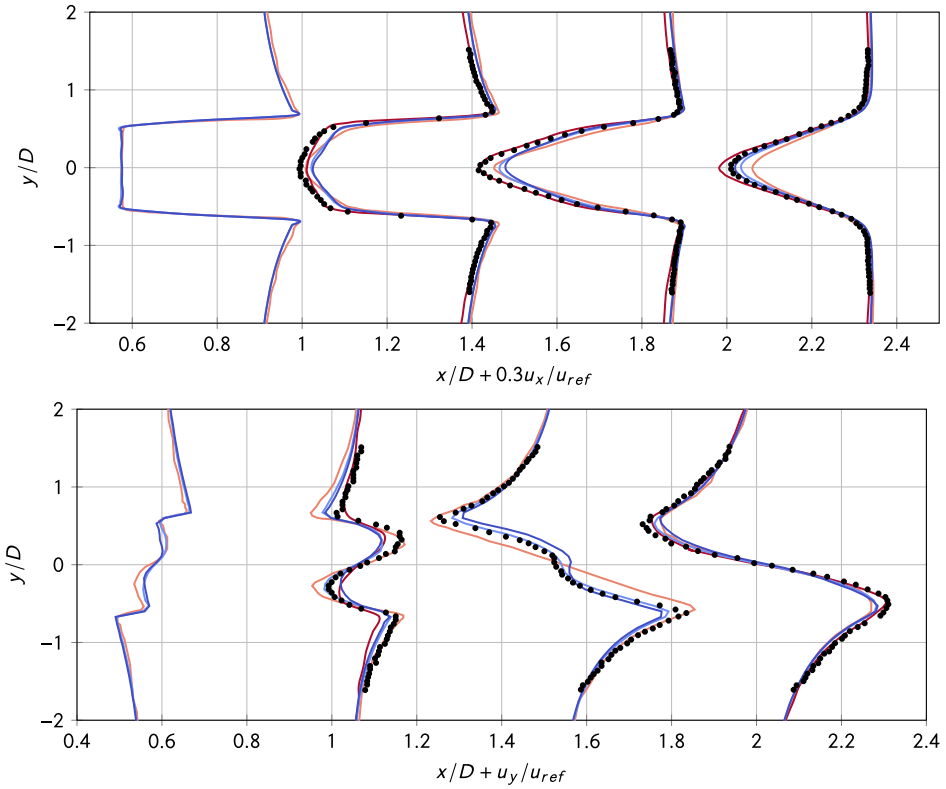

FIGURE 16 Flow around a circular cylinder—Spanwise and time-averaged stream-wise u_x/u_{ref} and cross-wise u_y/u_{ref} velocity profiles in the wake of the cylinder. • Parnaudeau et al. exp,[57] —— Wissink and Rodi DNS ($\sim 256 \cdot 10^6$ DoFs),[60] —— Meyer et al. ILES ($6.5 \cdot 10^6$ DoFs),[59] —— $DG - \mathbb{P}^{1\to5}$ mesh A, —— $DG - \mathbb{P}^{1\to5}$ mesh B [Colour figure can be viewed at wileyonlinelibrary.com]

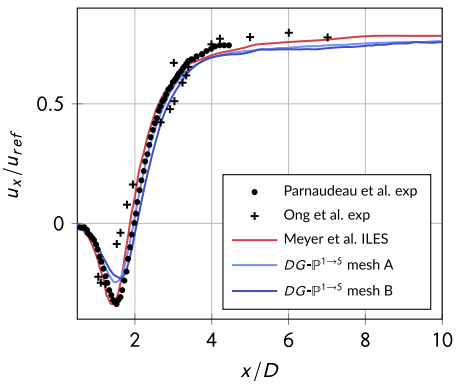

FIGURE 17 Flow around a circular cylinder—Spanwise and time-averaged stream-wise velocity u_x/u_{ref} profile along the centerline ($y/D = 0$) in the wake of the cylinder (Meyer et al. ILES $6.5 \cdot 10^6$ DoF[59]) [Colour figure can be viewed at wileyonlinelibrary.com]

suction coefficient is strongly related to the drag coefficient. The time-averaged recirculation length L_r/D corresponds to the distance between the surface of the cylinder and the sign change of the time-averaged streamwise velocity profile along the centerline in the wake of the cylinder. The Strouhal number is defined as $St = f_{\text{shedding}} D/u_{\text{ref}}$, where f_{shedding} is the frequency of the vortex shedding instability. A comparison between the results obtained on meshes A and B shows a good agreement; some differences can be appreciated only for $c_{L\,rms}$ and L_r/D. The comparison of our predicted results with available reference data, that is, the experimental measurements of Norberg[54] and Parnaudeau et al.,[57] shows a good agreement for the Strouhal number, c_D, $c_{L\,rms}$, $c_{p\,b}$, L_r/D, and θ_{sep}. Only $-u_{\min}/u_{ref}$ is slightly under estimated.

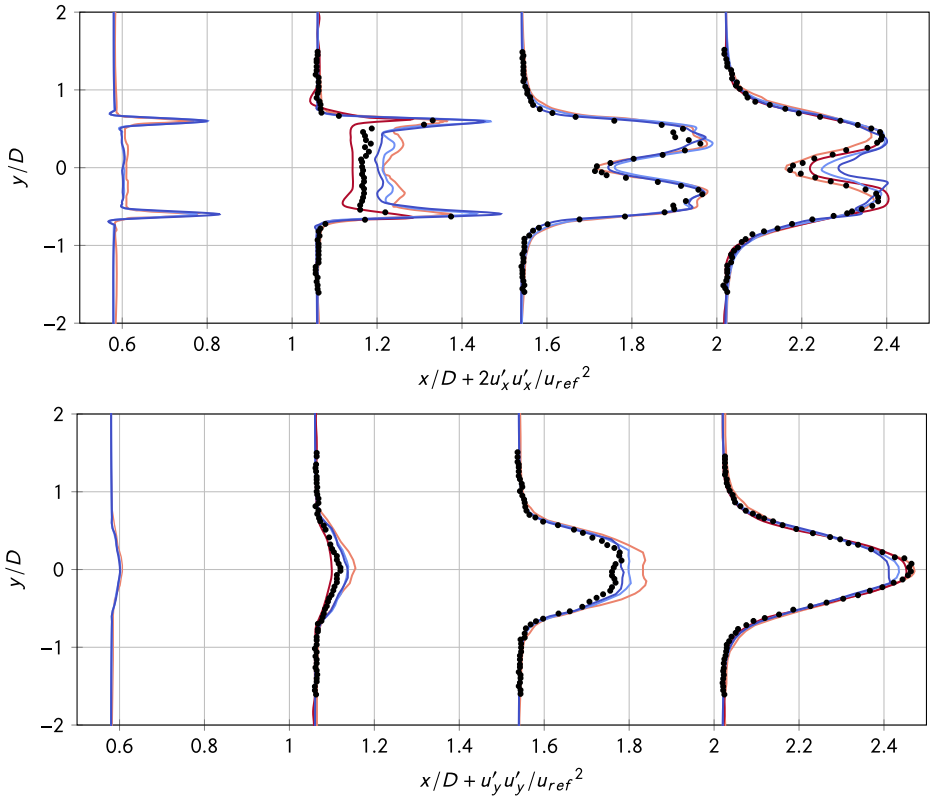

FIGURE 18 Flow around a circular cylinder—Spanwise and time-averaged stream-wise $u'_x u'_x / u^2_{\text{ref}}$ and cross-wise $u'_y u'_y / u^2_{\text{ref}}$ velocity fluctuations profiles in the wake of the cylinder. • Parnaudeau et al. exp,[57] Wissink and Rodi DNS ($\sim 256 \cdot 10^6$ DoFs),[60] Meyer et al. ILES ($6.5 \cdot 10^6$ DoFs),[59] $DG - \mathbb{P}^{1\to5}$ mesh A, $DG - \mathbb{P}^{1\to5}$ mesh B [Colour figure can be viewed at wileyonlinelibrary.com]

4.2.2 | Flow around the SD7003: $M_\infty = 0.2$, $Re_C = 60{,}000$, $\alpha = \{4°, 8°\}$

The transitional compressible flow around the Selig-Donovan 7003 (SD7003) airfoil with Mach number $M_\infty = 0.2$, angles of attack $\alpha = \{4°, 8°\}$, Reynolds number $Re_C = 60,000$ based on the chord profile, C, and free-stream conditions, and zero free-stream turbulence intensity at farfield, is considered here. The main difficulty of this testcase is related to the correct prediction of a laminar separation bubble, that is, a transitional shear layer followed by turbulent reattachment due to the enhanced momentum transport. The low Mach number has been chosen to compare our solutions with the incompressible numerical results presented in the literature[63-70] and experimental data.[71-73]

The mesh has 160,512 hexahedral quadratic elements and a first cell height of $\Delta n/C = 1.45 \cdot 10^{-4}$. A periodic boundary condition over a width $s = 0.2C$ is set in the spanwise direction to mimic an infinite span wing. At the circular farfield, located at 100 C from the airfoil, a characteristic-based boundary condition is imposed. Computations with different no-slip wall boundary conditions, adiabatic and fixed temperature $T_{\text{wall}}/T_{\text{ref}} = 1.002$,[63] were performed in order to evaluate their influence on the solution. As no significant differences were found for the wall distributions (pressure and skin friction coefficients) and the first- and second-order statistics (the velocity and velocity fluctuations profile), only the results for the adiabatic wall are here presented. During the simulation the adaptation was performed 6 times with a final number of DoFs equal to 1, 613,452 and 1, 627,750 for $\alpha = 4°$ and $\alpha = 8°$, respectively (see Figures 20 and 21). Snapshots of the instantaneous pressure coefficient and Mach number contours are reported in Figure 22.

The performance of the proposed p-adaptation strategy is assessed by comparing the predicted results with reference numerical results[63-70] and experimental measurements by Selig et al.,[71] Ol et al.[72] and Hain et al.[73] Table 6 summarizes the comparison in terms of the integral load coefficients, and of the parameters of the laminar separation bubble, that is, the separation and reattachment points coordinates, x_s/C and x_r/C, the separation bubble length and height, L/C and H/C. The predicted results shows an early separation and a delayed reattachment of the laminar separation bubble with respect to the experimental data. This discrepancy can be ascribed to the lower free-stream turbulence intensity. Only the drag coefficient for the $\alpha = 4°$ case seems to be quite different from the experimental value, but similar to other numerical

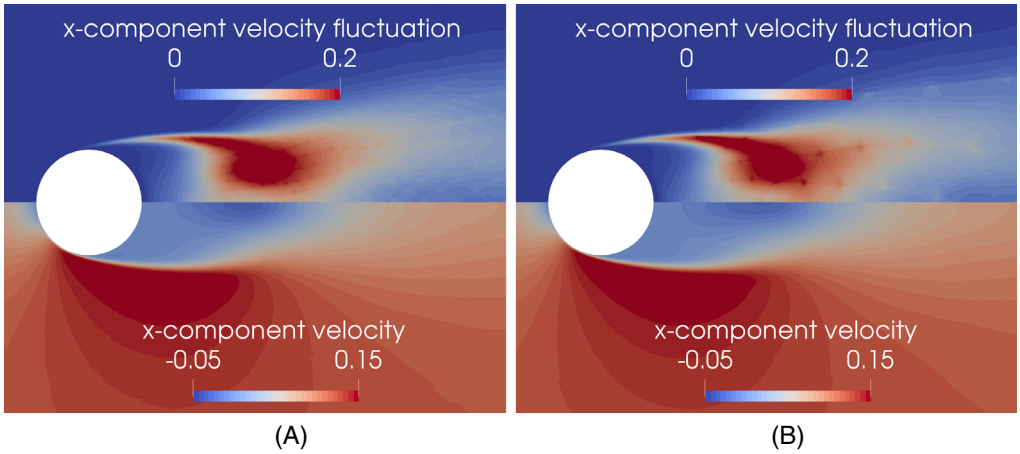

FIGURE 19 Flow around a circular cylinder—Time-averaged x -component velocity u_x/u_{ref} and velocity fluctuations $u_x'^2/u_{\text{ref}}^2$ for (A) mesh A and (B) mesh B, $\mathbb{P}^{1\to5}$ solution [Colour figure can be viewed at wileyonlinelibrary.com]

TABLE 5 Flow around a circular cylinder—Comparison with the literature of the root-mean-square lift coefficient $c_{L\,rms}$, the time-averaged drag coefficient c_D, the Strouhal number St, the time-averaged separation angles θ_{sep}, the time-averaged base suction coefficient $-c_{p\,b}$, the time-averaged peak of the streamwise velocity in the wake $-u_{\min}/u_{\text{ref}}$ and the time-averaged recirculation zone length L_r/D for mesh A and B, $\mathbb{P}^{1\to5}$ solution

Method	c_D	$c_{L\,rms}$	St	$-c_{p\,b}$	L_r/D	$-u_{\min}/u_{\text{ref}}$	θ_{sep}
Exp.[61]	0.99	-	0.22	-	1.19	0.24	86°
Exp.[54]	0.98	0.04–0.15	-	0.84	-	-	-
Exp.[58]	-	-	0.21	-	-	-	-
DNS[56]	0.84	-	0.22	0.89	1.59	-	-
Exp.[57]	-	-	0.21	-	1.48	0.34	-
LES[57]	-	-	0.21	-	1.56	0.28	-
DNS[60]	-	-	0.22	-	1.588	0.33	87°
LES[62]	1.04	-	0.2	0.94	1.35	0.37	88°
LES $_{Smag.}$ [55]	1.18	0.44	0.19	0.8	0.9	0.26	89°
LES $_{k-eq.}$ [55]	0.97	0.09	0.21	0.91	1.67	0.27	88°
$\mathbb{P}^{1\to5}$ mesh A	0.98	0.15	0.21	0.87	1.46	0.25	88°
$\mathbb{P}^{1\to5}$ mesh B	0.95	0.09	0.21	0.85	1.54	0.23	88°

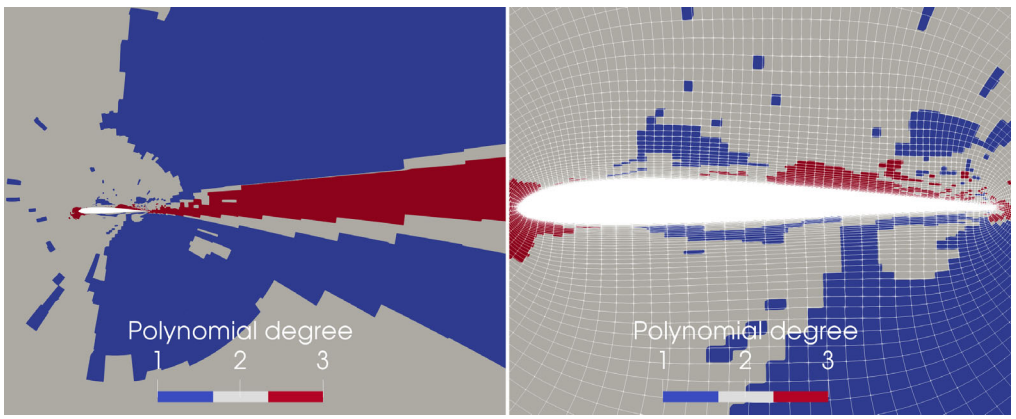

FIGURE 20 Flow around a SD7003 airfoil—Polynomial degree distribution with near wall detail for the $\alpha = 4°$ case ($1.61 \cdot 10^6$ DoFs), $\mathbb{P}^{1\to3}$ solution [Colour figure can be viewed at wileyonlinelibrary.com]

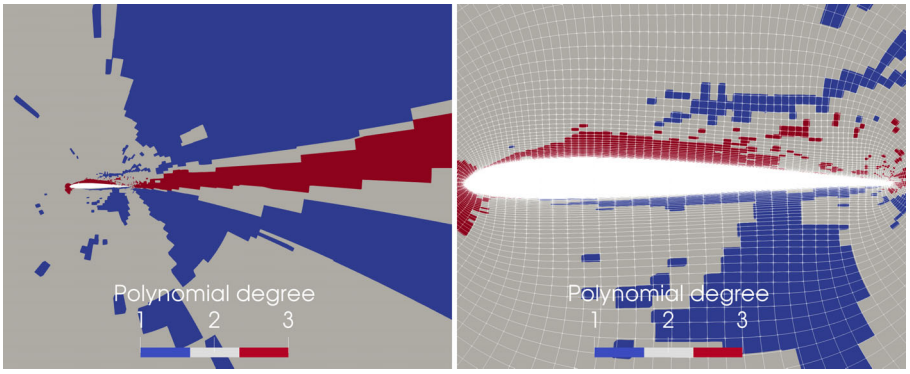

FIGURE 21 Flow around a SD7003 airfoil—Polynomial degree distribution with near wall detail for the $\alpha = 8°$ case ($1.63 \cdot 10^6$ DoFs), $\mathbb{P}^{1\to3}$ solution [Colour figure can be viewed at wileyonlinelibrary.com]

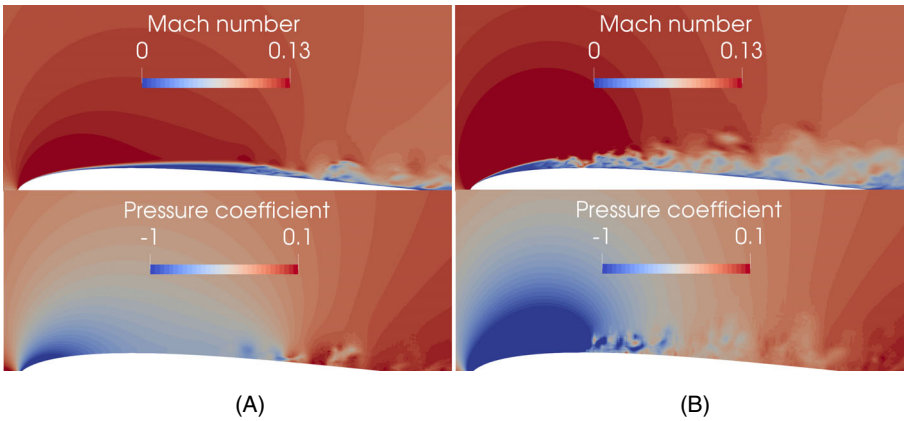

FIGURE 22 Flow around a SD7003 airfoil—Instantaneous pressure coefficient and Mach number contours for (A) $\alpha = 4°$ and (B) $\alpha = 8°$, $\mathbb{P}^{1\to3}$ solution [Colour figure can be viewed at wileyonlinelibrary.com]

TABLE 6 Flow around a SD7003 airfoil—Comparison of the laminar separation bubble dimensions and the aerodynamic loads for $\alpha = 4°$(top) and $\alpha = 8°$(bottom)

	Method	Tu (%)	DoFs (10^6)	x_s/C	x_r/C	L/C	H/C	c_D	c_L
$\alpha = 4°$	Exp.[72]	0.1	-	0.30	0.62	0.53	0.028	-	-
	Exp.[71]	-	-	-	-	-	-	$\approx$0.016	$\approx$0.60
	ILES DG- $\mathbb{P}^{1\to3}$	0	1.61	0.208	0.662	0.454	0.018	0.173	0.634
	ILES-FD[63]	0	5.70	0.230	0.675	-	0.030	-	-
	ILES-FD[65]	0	54.37	0.207	0.649	0.442	0.017	0.021	0.599
	ILES-DG[67]	-	10.94	0.209	0.654	-	-	0.196	0.602
	ILES-DG[68]	0	1.06	0.240	0.600	-	-	0.220	0.603
$\alpha = 8°$	Exp. (TUBS[73])	0.1	-	0.078	0.205	0.147	0.016	-	-
	Exp.[71]	-	-	-	-	-	-	$\approx$0.029	$\approx$0.92
	ILES-DG- $\mathbb{P}^{1\to3}$	0	1.62	0.028	0.332	0.304	0.023	0.024	0.998
	ILES-FD[63]	0	5.70	0.040	0.280	-	0.027	-	-
	ILES-FD[65]	0	54.37	0.031	0.303	0.272	0.020	0.044	0.917
	ILES-DG[66]	-	4.55	0.030	0.336	-	-	0.050	0.932
	ILES-FD[69]	0	4.48	0.037	0.200	-	-	0.034	0.968
	ILES-FR[70]	-	-	0.038	0.331	-	-	0.042	0.949

Note: *Tu* is the freestream turbulence intensity, x_s/C and x_r/C are the separation and reattachment points coordinates, L/C and H/C the separation bubble length and height, c_D the drag coefficient and c_L the lift coefficient.

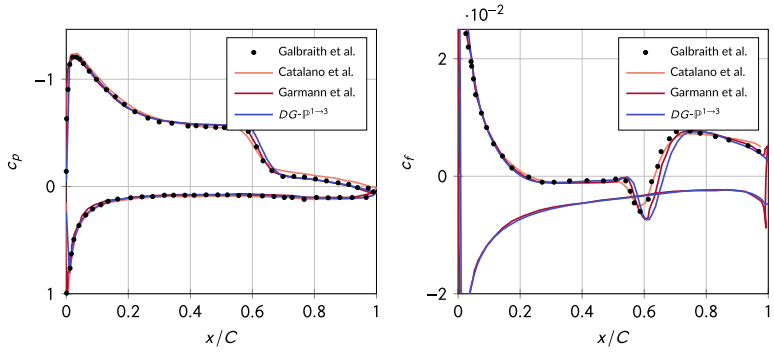

FIGURE 23 Flow around a SD7003 airfoil—Spanwise- and time-averaged pressure c_p (left) and skin friction c_f (right) coefficients distribution on the airfoil for $\alpha = 4°$, $\mathbb{P}^{1\to3}$ solution (Galbraith et al. $5.70 \cdot 10^6$ DoFs,[63] Catalano et al. $8.63 \cdot 10^6$ DoFs,[64] Garmann et al. $54.37 \cdot 10^6$ DoFs[65]) [Colour figure can be viewed at wileyonlinelibrary.com]

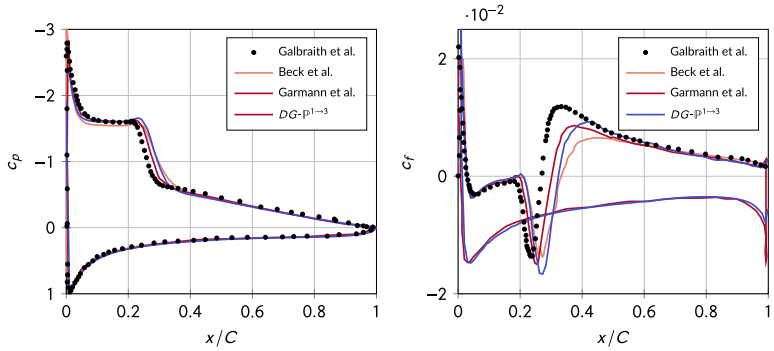

FIGURE 24 Flow around a SD7003 airfoil—Spanwise- and time-averaged pressure c_p (left) and skin friction c_f (right) coefficients distribution on the airfoil for $\alpha = 8°$, $\mathbb{P}^{1\to3}$ solution (Galbraith et al. $5.70 \cdot 10^6$ DoFs,[63] Beck et al. $4.55 \cdot 10^6$ DoFs,[66] Garmann et al. $54.37 \cdot 10^6$ DoFs[65]) [Colour figure can be viewed at wileyonlinelibrary.com]

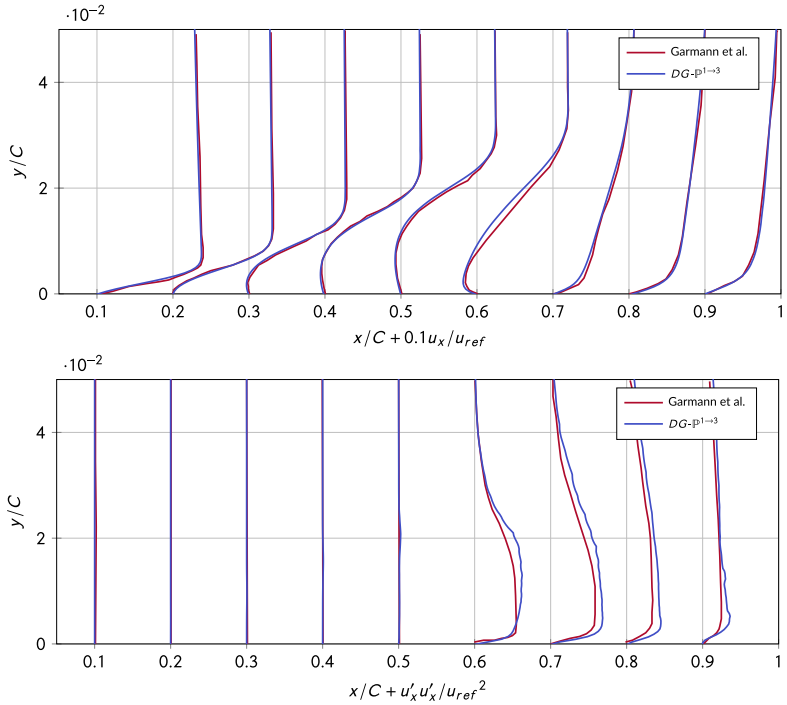

FIGURE 25 Flow around a SD7003 airfoil—Span-wise and time-averaged stream-wise velocity u_x/u_{ref} and velocity fluctuations $u'_x u'_x/u^2_{\text{ref}}$ profiles for $\alpha = 4°$, $DG - \mathbb{P}^{1\to3}$ solution (Garmann et al. $54.37 \cdot 10^6$ DoFs[65]) [Colour figure can be viewed at wileyonlinelibrary.com]

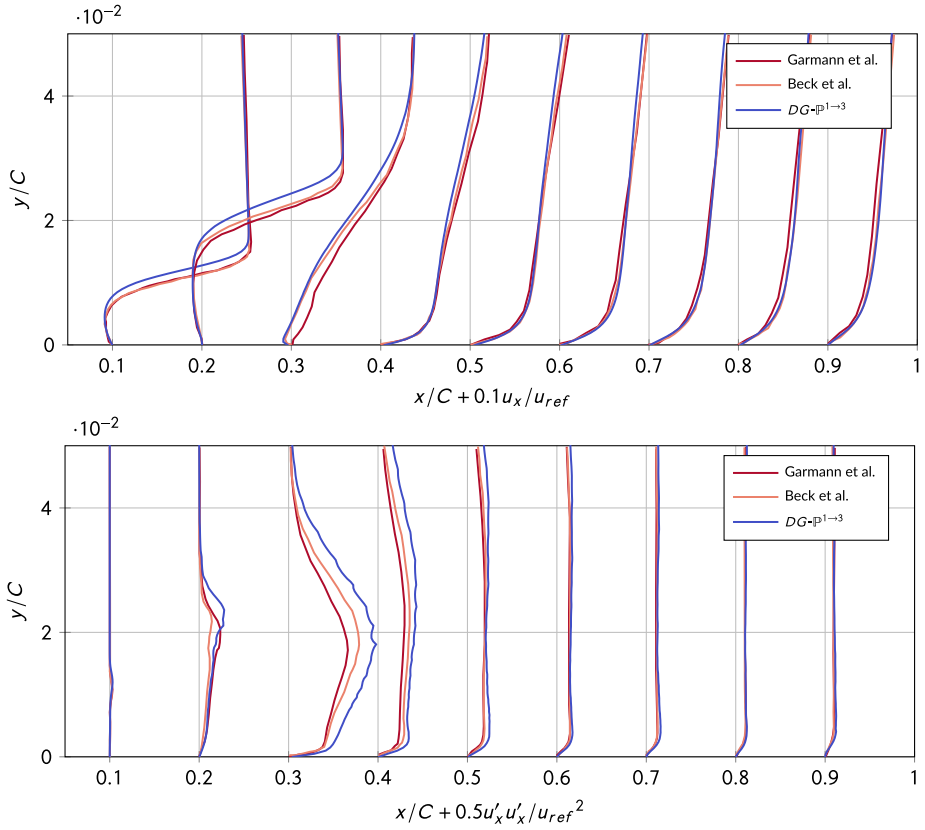

FIGURE 26 Flow around a SD7003 airfoil—Span-wise and time-averaged stream-wise velocity u_x/u_{ref} and velocity fluctuations $u_x'^2/u_{\text{ref}}^2$ profiles for $\alpha = 8°$, $DG - \mathbb{P}^{1\to3}$ solution (Garmann et al. $54.37 \cdot 10^6$ DoFs,[65] Beck et al. $4.55 \cdot 10^6$ DoFs[66]) [Colour figure can be viewed at wileyonlinelibrary.com]

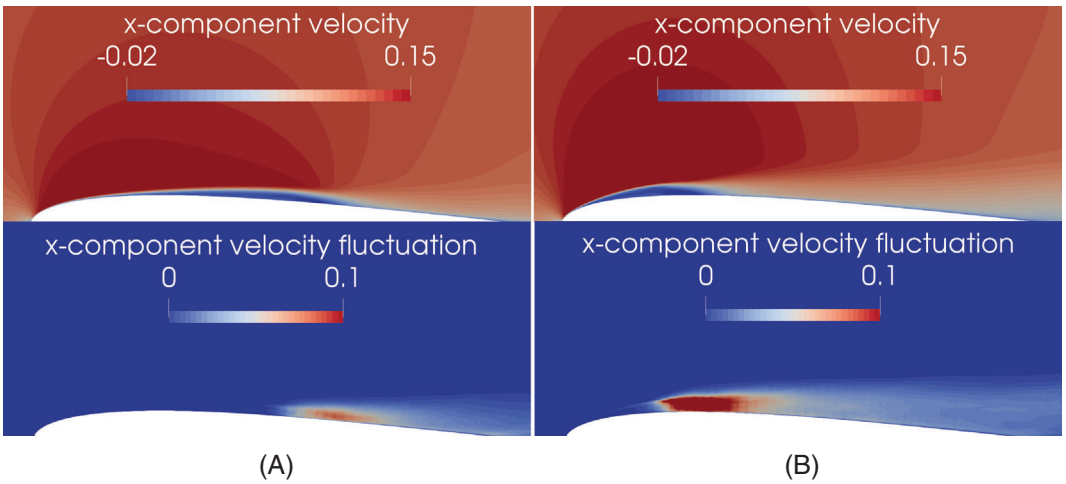

FIGURE 27 Flow around a SD7003 airfoil—Time-averaged x -component velocity u_x/u_{ref} and velocity fluctuations $u_x'^2/u_{\text{ref}}^2$ (A) $\alpha = 4°$ and (B) $\alpha = 8°$, $\mathbb{P}^{1\to3}$ solution [Colour figure can be viewed at wileyonlinelibrary.com]

results. Only Garmann et al. shows a c_D value comparable with experiments, but using a number of DoFs $\approx$ 33 times higher with respect to our simulation. Although a general dispersion is observed in the published data, our results are within a reasonable range, giving some confidence on the ability of adaptation in delivering good results with a significant saving of DoFs.

Figures 23 and 24 show the spanwise- and time-averaged pressure and skin friction coefficients distribution for $\alpha = 4°$ and $\alpha = 8°$, respectively. As expected, some variability is observed in the published results but our $\mathbb{P}^{1\to3}$ computations compare well with the references. A reasonable agreement for the velocity and fluctuations profiles with the results of Garmann et al.[65] and Beck et al.[66] is observed for both $\alpha = 4°$ and $\alpha = 8°$ cases, as shown in Figures 25 and 26, respectively. Finally, Figure 27 shows the contours of the time-averaged x -component of the velocity fluctuation.

5 | CONCLUSIONS

The article presents an implicit p -adaptive DG method designed for both CAA and CFD simulations. The solver takes advantage of the same scalable computational framework, which uses a single simple definition for the error estimator computed with different statistics, that is, the time-average or r.m.s. pressure distribution, whether the focus is on aerodynamics loads and total pressure losses (CFD) or on the acoustics perturbations originated by flow unsteadiness (CAA). Our implementation also exploits an adaptive selection for the quadrature rules degree-of-exactness to avoid over-integration of straight-sided elements contributing to the overall efficiency of the method.

Numerical results on both CAA and CFD flow problems demonstrated the capability of the algorithm to deliver accurate solutions, comparable or even better than uniform-order discretizations, with a significant reduction ($\approx$ 50%) of the computational effort. As expected, for CAA applications the use of a non-reflecting treatment at boundaries was mandatory. The present p-adaptive method was combined, without any ad-hoc tuning, with the simple sponge-layer approach proposed in Reference 38 to damp the spurious reflection showing very good performance.

Future work will involve the development of advanced linear solvers designed to further reduce the memory footprint and the solution time. Particular attention will be devoted to efficient matrix-free iterative solvers suited for the use in DG discretizations of compressible flows in primitive variables.

ACKNOWLEDGMENT

We acknowledge CINECA for the availability of high performance computing resources under the Italian Super-Computing Resource Allocation (ISCRA) initiative. Open Access Funding provided by Universita degli Studi di Bergamo within the CRUI-CARE Agreement.

DATA AVAILABILITY STATEMENT

Authors elect to not share data.

ORCID

Alessandro Colombo https://orcid.org/0000-0002-6527-8148
Andrea Crivellini https://orcid.org/0000-0002-3995-742X
Antonio Ghidoni https://orcid.org/0000-0001-5513-4784
Alessandra Nigro https://orcid.org/0000-0002-0644-1670
Gianmaria Noventa https://orcid.org/0000-0002-7668-4539

References

1. Bassi F , Botti L, Colombo A, et al. Very high-order accurate discontinuous Galerkin computation of transonic turbulent flows on aeronautical configurations. In: Kroll N, Bieler H, Deconinck H, Couaillier V, van derVen H, Sørensen K, eds. *ADIGMA - A European Initiative on the Development of Adaptive Higher-Order Variational Methods for Aerospace Applications*. Notes on Numerical Fluid Mechanics and Multidisciplinary Design. Vol 113. Springer; 2010:25-38.
2. Hartmann R, Held J, Leicht T, Prill F. Error estimation and adaptive mesh refinement for aerodynamic flows. In: Norbert Kroll, Heribert Bieler, Herman Deconinck, Vincent Couaillier, Harmen van der Ven, Kaare Sørensen, eds. *ADIGMA-A European Initiative on the Development of Adaptive Higher-Order Variational Methods for Aerospace Applications*. Notes on Numerical Fluid Mechanics and Multidisciplinary Design. Vol 113. Springer; 2010:339-354.
3. Wang L, Mavriplis DJ. Adjoint-based *h-p* adaptive discontinuous Galerkin methods for the 2D compressible Euler equations. *J Comput Phys*. 2009;228(20):7643-7661.

4. de Wiart CC, Hillewaert K, Bricteux L, Winckelmans G. Implicit LES of free and wall-bounded turbulent flows based on the discontinuous Galerkin/symmetric interior penalty method. *Int J Numer Methods Fluids*. 2015;78(6):335-354. doi:10.1002/fld.4021
5. Bassi F, Botti L, Colombo A, Crivellini A, Ghidoni A, Massa F. On the development of an implicit high-order discontinuous Galerkin method for DNS and implicit LES of turbulent flows. *Eur J Mech B/Fluids*. 2016;55:367-379.
6. Cagnone JS, Zeren Z, Châtel A, Rasquin M, Hillewaert K, Bricteux L. Assessment of high-order discontinuous Galerkin methods for LES of transonic flows. In: Salvetti MV, Armenio V, Fröhlich J, Geurts BJ, Kuerten H, eds. *Direct and Large-Eddy Simulation XI*. Springer International Publishing; 2019:83-90.
7. Sørensen KA, Johnston C, Leicht T, et al. External aerodynamic test cases. In: Kroll N, Hirsch C, Bassi F, Johnston C, Hillewaert K, eds. *IDIHOM: Industrialization of High-Order Methods - A Top-Down Approach*. Notes on Numerical Fluid Mechanics and Multidisciplinary Design. Vol 128. Springer International Publishing; 2015:553-606.
8. Bassi F, Franchina N, Ghidoni A, Rebay S. A numerical investigation of a spectral-type nodal collocation discontinuous Galerkin approximation of the Euler and Navier-Stokes equations. *Int J Numer Methods Fluids*. 2013;71:1322-1339.
9. Ghidoni A, Colombo A, Bassi F, Rebay S. Efficient *p*-multigrid discontinuous Galerkin solver for complex viscous flows on stretched grids. *Int J Numer Methods Fluids*. 2014;75(2):134-154.
10. Wallraff M , Hartmann R, Leicht T. Multigrid solver algorithm for DG methods and applications to aerodynamic flows. In: Kroll N, Hirsch C, Bassi F, Johnston C, Hillewaert K, eds. *IDIHOM: Industrialization of High-Order Methods - A Top-Down Approach*. Notes on Numerical Fluid Mechanics and Multidisciplinary Design. Vol 128. Springer International Publishing; 2015. doi:10.1007/978-3-319-12886-3_11
11. Pazner W , Persson P-O. Approximate tensor-product preconditioners for very high order discontinuous Galerkin methods. *J Comput Phys*. 2018;354:344-369. doi:10.1016/j.jcp.2017.10.030
12. Diosady LT, Murman SM. Scalable tensor-product preconditioners for high-order finite-element methods: scalar equations. *J Comput Phys*. 2019;394:759-776. doi:10.1016/j.jcp.2019.04.047
13. Tugnoli M, Abbà A, Bonaventura L, Restelli M. A locally *p*-adaptive approach for large eddy simulation of compressible flows in a DG framework. *J Comput Phys*. 2017;349:33-58. doi:10.1016/j.jcp.2017.08.007
14. Naddei F , de laLlave Plata M, Couaillier V, Coquel F. A comparison of refinement indicators for *p*-adaptive simulations of steady and unsteady flows using discontinuous Galerkin methods. *J Comput Phys*. 2019;376:508-533. doi:10.1016/j.jcp.2018.09.045
15. Kompenhans M , Rubio G, Ferrer E, Valero E. Comparisons of *p*-adaptation strategies based on truncation- and discretisation-errors for high order discontinuous Galerkin methods. *Comput Fluids*. 2016;139:36-46. doi:10.1016/j.compfluid.2016.03.026
16. Rueda-Ramírez AM, Rubio G, Ferrer E, Valero E. Truncation error estimation in the *p*-anisotropic discontinuous Galerkin spectral element method. *J Sci Comput*. 2019;78(1):433-466. doi:10.1007/s10915-018-0772-0
17. Abbà A , Recanati A, Tugnoli M, Bonaventura L. Dynamical *p*-adaptivity for LES of compressible flows in a high order DG framework. *J Comput Phys*. 2020;420:109720. doi:10.1016/j.jcp.2020.109720
18. Hartmann R , Houston P. Adaptive discontinuous Galerkin finite element methods for the compressible Euler equations. *J Comput Phys*. 2002;183(2):508-532. doi:10.1006/jcph.2002.7206
19. Fidkowski KJ, Luo Y. Output-based space–Time mesh adaptation for the compressible Navier–Stokes equations. *J Comput Phys*. 2011;230(14):5753-5773. doi:10.1016/j.jcp.2011.03.059
20. Ceze M , Fidkowski KJ. Drag prediction using adaptive discontinuous finite elements. *J Aircr*. 2014;51(4):1284-1294. doi:10.2514/1.C032622
21. Fidkowski KJ. Output-based space–Time mesh optimization for unsteady flows using continuous-in-time adjoints. *J Comput Phys*. 2017;341:258-277. doi:10.1016/j.jcp.2017.04.005
22. Fidkowski KJ, Chen G. Metric-based, goal-oriented mesh adaptation using machine learning. *J Comput Phys*. 2021;426:109957. doi:10.1016/j.jcp.2020.109957
23. Basile F, Chapelier J-B, Laraufie R, Frey P. *hp*-adaptive hybrid RANS/LES simulations for unstructured meshes with the discontinuous Galerkin method. AIAA science and technology forum and exposition. *AIAA SciTech Forum*. 2022;2022:1207. doi:10.2514/6.2022-1207
24. Basile F, Chapelier J-B, de laLlave Plata M, Laraufie R, Frey P. Unstructured *h*- and *hp*-adaptive strategies for discontinuous Galerkin methods based on a posteriori error estimation for compressible flows. *Comput Fluids*. 2022;233:105245. doi:10.1016/j.compfluid.2021.105245
25. Bassi F, Botti L, Colombo A, et al. Time integration in the discontinuous Galerkin code MIGALE - Unsteady problems. In: Norbert Kroll, Charles Hirsch, Francesco Bassi, Craig Johnston, Koen Hillewaert, eds.*IDIHOM: Industrialization of High-Order Methods-A Top-Down Approach*. Springer; 2015:205-230.
26. Uranga A, Persson P-O, Drela M, Peraire J. Implicit large eddy simulation of transition to turbulence at low Reynolds numbers using a discontinuous Galerkin method. *Int J Numer Methods Eng*. 2011;87(1-5):232-261. doi:10.1002/nme.3036
27. Moura RC, Sherwin SJ, Peiró J. Linear dispersion–Diffusion analysis and its application to under-resolved turbulence simulations using discontinuous Galerkin spectral/*hp* methods. *J Comput Phys*. 2015;298:695-710.
28. de la Llave Plata M, Couaillier V, lePape M-C. On the use of a high-order discontinuous Galerkin method for DNS and LES of wall-bounded turbulence. *Comput Fluids*. 2018;176:320-337. doi:10.1016/j.compfluid.2017.05.013
29. Moura RC, Mengaldo G, Peiró J, Sherwin SJ. On the eddy-resolving capability of high-order discontinuous Galerkin approaches to implicit LES / under-resolved DNS of Euler turbulence. *J Comput Phys*. 2017;330:615-623. doi:10.1016/j.jcp.2016.10.056
30. Park JS, Witherden FD, Vincent PE. High-order implicit large-eddy simulations of flow over a NACA0021 aerofoil. *AIAA J*. 2017;55(7):2186-2197. doi:10.2514/1.J055304

31. A. Garai, L.T. Diosady, S.M. Murman, and N.K. Madavan. Scale-resolving simulations of bypass transition in a high-pressure turbine cascade using a spectral element discontinuous Galerkin method. *J Turbomach*, 140 (3), 2018:13. 10.1115/1.4038403
32. Krivodonova L , Xin J, Remacle J-F, Chevaugeon N, Flaherty JE. Shock detection and limiting with discontinuous Galerkin methods for hyperbolic conservation laws. *Appl Numer Math*. 2004;48(3-4):323-338.
33. Gassner G , Staudenmaier M, Hindenlang F, Atak M, Munz C-D. A space–Time adaptive discontinuous Galerkin scheme. *Comput Fluids*. 2015;117:247-261. doi:10.1016/j.compfluid.2015.05.002
34. Persson PO, Peraire J. Sub-cell shock capturing for discontinuous Galerkin methods. Proceedings of the 44th AIAA Aerospace Sciences Meeting and Exhibit; 2006:112.
35. A. Colombo, G.Manzinali, A.Ghidoni, G.Noventa, M.Franciolini, A.Crivellini, and F.Bassi. A *p*-adaptive implicit discontinuous Galerkin method for the under-resolved simulation of compressible turbulent flows. Proceedings of the 7nd European Conference on Computational Fluid Dynamic; 2020:4159-4170.
36. F. Bassi, A.Colombo, A.Crivellini, M.Franciolini, A.Ghidoni, G.Manzinali, and G.Noventa. Under-resolved simulation of turbulent flows using a *p*-adaptive discontinuous Galerkin method. In RamisÖrlü, AlessandroTalamelli, JoachimPeinke, and MartinOberlack, editors, *Progress in Turbulence VIII*, pages 157–162. Springer International Publishing, 2019. 10.1007/978-3-030-22196-6_25
37. Bassi F , Botti L, Colombo A, et al. A *p*-adaptive matrix-free discontinuous Galerkin method for the implicit LES of incompressible transitional flows. *Flow Turbul Comb*. 2020;105(2):437-470. doi:10.1007/s10494-020-00178-2
38. Colombo A , Crivellini A. Assessment of a sponge layer non-reflecting boundary treatment for high-order CAA/CFD computations. *Comput Fluids*. 2016;140:1339-1351. doi:10.1016/j.compfluid.2016.09.019
39. Bassi F , Botti L, Colombo A, Di Pietro DA, Tesini P. On the flexibility of agglomeration based physical space discontinuous Galerkin discretizations. *J Comput Phys*. 2012;231(1):45-65.
40. F. Bassi, S.Rebay, G.Mariotti, S.Pedinotti, and M.Savini. A high-order accurate discontinuous finite element method for inviscid and viscous turbomachinery flows. In R. Decuypere and G. Dibelius, editors, *Proceedings of the 2nd European Conference on Turbomachinery Fluid Dynamics and Thermodynamics*, pages 99–108, 1997. Technologisch Instituut.
41. Arnold DN, Brezzi F, Cockburn B, Marini LD. Unified analysis of discontinuous Galerkin methods for elliptic problems. *SIAM J Numer Anal*. 2002;39(5):1749-1779.
42. Gottlieb JJ, Groth CPT. Assessment of Riemann solvers for unsteady one-dimensional inviscid flows of perfect gases. *J Comput Phys*. 1988;78(2):437-458.
43. Lang J, Verwer J. ROS3P—An accurate third-order Rosenbrock solver designed for parabolic problems. *BIT*. 2001;41(4):731-738.
44. Bassi F, Botti L, Colombo A, Ghidoni A, Massa F. Linearly implicit Rosenbrock-type Runge–Kutta schemes applied to the discontinuous Galerkin solution of compressible and incompressible unsteady flows. *Comput Fluids*. 2015;118:305-320.
45. Massa FC, Noventa G, Lorini M, Bassi F, Ghidoni A. High-order linearly implicit two-step peer schemes for the discontinuous Galerkin solution of the incompressible Navier–Stokes equations. *Comput Fluids*. 2018;162:55-71.
46. Balay S, Abhyankar S, Adams M. PETSc Web page, 2018. http://www.mcs.anl.gov/petsc
47. Ghidoni A, Pelizzari E, Rebay S, Selmin V. 3D anisotropic unstructured grid generation. *Int J Numer Methods Fluids*. 2006;51(9-10):1097-1115.
48. Karypis G, Kumar V. METIS: unstructured graph partitioning and sparse matrix ordering system, version 5.0. Technical report; 2009. http://www.cs.umn.edu/metis
49. Bassi F , Colombo A, Crivellini A, et al. Entropy-adjoint *p*-adaptive discontinuous Galerkin method for the under-resolved simulation of turbulent flows. *AIAA J*. 2020;58(9):3963-3977. doi:10.2514/1.J058847
50. Crivellini A. Assessment of a sponge layer as a non-reflective boundary treatment with highly accurate gust–Airfoil interaction results. *Int J Comput Fluid Dyn*. 2016;30(2):176-200. doi:10.1080/10618562.2016.1167193
51. Desquesnes G, Terracol M, Manoha E, Sagaut P. On the use of a high order overlapping grid method for coupling in CFD/CAA. *J Comput Phys*. 2006;220(1):355-382. doi:10.1016/j.jcp.2006.05.019
52. Inoue O, Hatakeyama N. Sound generation by a two-dimensional circular cylinder in a uniform flow. *J Fluid Mech*. 2002;471:285-314. doi:10.1017/S0022112002002124
53. HioCFD-3 webpage. Proceedings of the 3rd International Worskhop on High-Order CFD methods; January 3–4, 2015; Orlando, Florida. https://www.grc.nasa.gov/hiocfd
54. Norberg C. Flow around a circular cylinder: aspects of fluctuating lift. *J Fluids Struct*. 2001;15(3):459-469.
55. Lysenko DA, Ertesvåg IS, Rian KE. Large-eddy simulation of the flow over a circular cylinder at Reynolds number 3900 using the OpenFOAM toolbox. *Flow Turbul Comb*. 2012;89(4):491-518. doi:10.1007/s10494-012-9405-0
56. Ma X, Karamanos G-S, Karniadakis GE. Dynamics and low-dimensionality of a turbulent near wake. *J Fluid Mech*. 2000;410:29-65.
57. Parnaudeau P, Carlier J, Heitz D, Lamballais E. Experimental and numerical studies of the flow over a circular cylinder at Reynolds number 3900. *Phys Fluids*. 2008;20:08. doi:10.1063/1.2957018
58. Ong L, Wallace J. The velocity field of the turbulent very near wake of a circular cylinder. *Exp Fluids*. 1996;20(6):441-453. doi:10.1007/BF00189383
59. Meyer M, Hickel S, Adams NA. Assessment of implicit large-eddy simulation with a conservative immersed interface method for turbulent cylinder flow. *Int J Heat Fluid Flow*. 2010;31(3). doi:10.1016/j.ijheatfluidflow.2010.02.026 Proceedings of the Sixth International Symposium on Turbulence and Shear Flow Phenomena. pp. 368-377.
60. Wissink J, Rodi W. Numerical study of the near wake of a circular cylinder. *Int J Heat Fluid Flow*. 2008;29:1060-1070. doi:10.1016/j.ijheatfluidflow.2008.04.001

61. Lourenco LM, Shih C. Characteristics of the plane turbulent near wake of a circular cylinder. a particle image velocimetry study. 1994. Published in P. Beaudan, P. Moin. Numerical experiments on the flow past a circular cylinder at sub-critical Reynolds number. Report TF-62, Stanford University.
62. Breuer M. Large eddy simulation of the subcritical flow past a circular cylinder: numerical and modeling aspects. *Int J Numer Methods Fluids*. 1998;28(9):1281-1302.
63. Galbraith M, Visbal M. Implicit large eddy simulation of low Reynolds number flow past the SD7003 airfoil; Vol. 225, 2008. doi:10.2514/6.2008-225
64. Catalano P, Tognaccini R. Large eddy simulations of the flow around the SD 7003 airfoil. Proceedings of the AIMETA Conference; 2011:1-10.
65. Garmann DJ, Visbal MR. C3.3: implicit large eddy-simulations of transitional flow over the SD7003 airfoil using compact finite-differencing and filtering. Proceedings of the 2nd International Workshop on High-Order CFD Methods; May 27–28 2013; Cologne, Germany.
66. Beck A, Bolemann T, Flad D, et al. High-order discontinuous Galerkin spectral element methods for transitional and turbulent flow simulations. *Int J Numer Methods Fluids*. 2014;76:11. doi:10.1002/fld.3943
67. de Wiart CC, Hillewaert K. DNS and ILES of transitional flows around a SD7003 using a high order discontinuous Galerkin method. Proceedings of the 7th International Conference on Computational Fluid Dynamics (ICCFD7); 01 2012.
68. Uranga A, Persson PO, Drela M, Peraire J. Implicit large eddy simulation of transitional flows over airfoils and wings; 06, 2009. doi:10.2514/6.2009-4131
69. Boom P, Zingg D. Time-accurate flow simulations using an efficient Newton-Krylov-Schur approach with high-order temporal and spatial discretization. Proceedings of the 51st AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition; 2013. doi:10.2514/6.2013-383
70. Vermeire BC, Vincent PE. On the properties of energy stable flux reconstruction schemes for implicit large eddy simulation. *J Comput Phys*. 2016;327:368-388. doi:10.1016/j.jcp.2016.09.034
71. Selig MS, Guglielmo JJ, Broeren AP, Giguere P. Summary of low-speed airfoil data. Technical report, Vol. 1. Soartech Publications; 1995.
72. Ol M, McCauliffe B, Hanff E, Scholz U, Kaehler C. Comparison of laminar separation bubble measurements on a low Reynolds number airfoil in three facilities. Proceedings of the 35th AIAA Fluid Dynamics Conference and Exhibit; 2005:1-12. doi:10.2514/6.2005-5149
73. Hain R, Kähler CJ, Radespiel R. Dynamics of laminar separation bubbles at low-Reynolds-number aerofoils. *J Fluid Mech*. 2009;630:129-153. doi:10.1017/S0022112009006661

How to cite this article: Colombo A, Crivellini A, Ghidoni A, Nigro A, Noventa G. An implicit *p*-adaptive discontinuous Galerkin solver for CAA/CFD simulations. *Int J Numer Meth Fluids*. 2022;1-29. doi: 10.1002/fld.5089