Engineering with Computers
https://doi.org/10.1007/s00366-024-02062-3

ORIGINAL ARTICLE

Accelerating high order discontinuous Galerkin solvers through a clustering-based viscous/turbulent-inviscid domain decomposition

Kheir-Eddine Otmani[1] · Andrés Mateo-Gabín[1] · Gonzalo Rubio[1,2] · Esteban Ferrer[1,2]

Received: 11 May 2024 / Accepted: 3 September 2024

Abstract
We explore the unsupervised clustering technique introduced in Otmani et al. (Phys Fluids 35:027112, 2023) to identify viscous/turbulent from inviscid regions in incompressible flows. The separation of regions allows solving the Navier–Stokes equations including Large Eddy Simulation closure models only in the viscous/turbulent ones, while solving the Euler equations in the remaining of the computational domain. By solving different sets of equations, the computational cost is significantly reduced. This coupling strategy is implemented within a discontinuous Galerkin numerical framework, which allows discontinuous solutions (i.e., different sets of equations) in neighboring elements that interact through numerical fluxes. The proposed strategy maintains the same accuracy at lower cost, when compared to solving the full Navier–Stokes equations throughout the entire domain. Validation of this approach is conducted across diverse flow regimes, spanning from unsteady laminar flows to unsteady turbulent flows, including an airfoil section at Reynolds numbers $Re = 10^3$ and 10^4 and large angles of attack, and the flow past a wind turbine, modelled using actuator lines. The computational cost is reduced by 25% and 29% for the unsteady turbulent flow around an airfoil section and the flow past the wind turbine, respectively. In addition, to further accelerate the simulations, we combine the proposed decoupling with local *P*-adaptation, as proposed in Tlales et al. (Eng Comput, 2024). When doing so, we reduce the computational cost by 41% and 45% for the flow around the airfoil section and the flow past the wind turbine, respectively.

Keywords Computational fluid mechanics · Machine learning · Unsupervised clustering · Navier Stokes equations · Large eddy simulations · *P*-adaptation · NACA0012 · Wind turbine

1 Introduction

The Navier–Stokes equations describe the motion of fluids and include terms that account for convection and viscous diffusion. When supplemented with a turbulence subgrid model of Large Eddy Simulation (LES) type, the resulting equations are able to properly capture turbulence even in under-resolved, coarser meshes. Large eddy simulations can be expensive and finding means to decrease the cost is important as this enables the study of more complex physics through larger and longer simulations.

One possibility to reduce the cost is to refine the mesh in the flow regions where viscous and turbulent effects are important. Indeed, viscous/turbulent regions require finer meshes to capture large gradients (e.g., boundary layers) and vortex dynamics (e.g., wakes). The increased resolution can be obtained by means of mesh or polynomial refinements when using high order methods [14–16].

A less explored alternative is to modify the equations such that these only include the terms that are relevant to the flow regions at hand. In regions where intense viscous effects are present (e.g., boundary layers, wakes), it is necessary to solve the full LES Navier–Stokes equations with high accuracy, while far from walls (and in the absence of free stream turbulence) it is possible to solve only the Euler equations (neglecting viscosity and turbulence). Domain decomposition ideas have been explored in the past. Zhang

✉ Kheir-Eddine Otmani
otmani.kheir-eddine@alumnos.upm.es

[1] ETSIAE-UPM-School of Aeronautics, Universidad Politécnica de Madrid, Plaza Cardenal Cisneros 3, 28040 Madrid, Spain

[2] Center for Computational Simulation, Universidad Politécnica de Madrid, Campus de Montegancedo, Boadilla del Monte, 28660 Madrid, Spain

Published online: 21 September 2024

et al. [37] developed a non-parametric discontinuous Galerkin finite-element formulation for integral boundary layer equations, featuring strong viscous-inviscid coupling. In [27], a high-order viscous-inviscid interaction solver for aerodynamic flows was introduced, employing a split formulation where viscous and inviscid effects are separately solved in overlapping domains near solid walls or the wake centerline. A coupled viscous-inviscid interaction scheme that integrates the continuity equation for potential flow with the three-dimensional integral boundary layer equations was presented in [23]. To our knowledge, there has not been any work using clustering to separate viscous/turbulent from inviscid regions to accelerate high order solvers.

To apply any of the above mentioned techniques, it is critical to be able to identify/separate the regions where viscous/turbulent effects are important from the rest. In this work, we explore the capability of the clustering methodology proposed in [28] to decompose the mesh and simplify the equations when possible (away from walls and wakes). The full Navier–Stokes equations are exclusively solved in regions where the viscous effects are prominent. Away from them (i.e. in inviscid regions) the viscous and turbulent terms are neglected. The challenge is to retain accuracy, as to when solving the full set of equations everywhere, while reducing the computational cost.

In recent years, clustering techniques based on machine learning have emerged as promising tools to address the challenge of flow region separation in CFD. Clustering algorithms group data points into distinct clusters based on their similarities, enabling the identification of flow regions. These techniques leverage the power of data-driven approaches and can adapt to complex and dynamic flow patterns. By applying clustering techniques to CFD simulations, we can automatically detect and separate flow regions with different characteristics. In the context of flow region detection, various researchers have used machine learning models and especially clustering techniques, see [4, 22, 26, 30] and our previous work [28, 33].

Here, we propose to use a clustering technique along with the feature space proposed in [28] to identify viscous-dominated rotational regions (including turbulent flow), using the principal invariants of strain and rotational rate tensors as inputs to the Gaussian Mixture Model (GMM) to detect/separate viscous/turbulent-dominated areas from the rest. Furthermore, in [33], we showed that the detected regions can be tracked during the simulation and can be used to dynamically adjust the local resolution used to approximate the solution (*P*-adaptation for high order methods) with the aim of reducing the computational cost while maintaining the level of accuracy.

In this new work, the focus is to use the separated regions to simplify the equations to be solved, i.e., the full LES Navier–Stokes equations are to be solved only in the viscous/turbulent-dominated regions while the Euler equations are solved in the outer region. By doing so, we can concentrate the computational efforts in the regions where viscous/turbulent effects are important. The numerical method used to discretize the NS equations is the Discontinuous Galerkin Spectral Element Method (DGSEM) implemented in the open-source solver HORSES3D [11], since discontinuous Galerkin techniques allow to naturally handle discontinuous solutions. In the second part of the paper, we combine the proposed methodology (hybrid equations in the domain) with local *P*-adaptation to further accelerate the simulations.

The work is organized as follows. We first introduce the methodology in Sect. 2 including the viscous/turbulent region detection through a Gaussian mixture model, and details on the high order discontinuous Galerkin solver HORSES3D. Section 3.1 summarises the results obtained for the hybrid approach (solving LES-Navier–Stokes near walls and wakes and Euler in the rest) to show acceleration without loss of accuracy. In Sect. 3.2, we combine the new method with local *P*-adaptation to further accelerate the simulations. Finally, conclusions are summarised in Sect. 4.

2 Methodology

2.1 Viscous/turbulent region detection through Gaussian mixture models

Distinct physics are present in different flow regions. For example, the flow near walls is characterized by important viscous effects, which lead to the development of boundary layers and within wakes, rotation, viscous effects and turbulence will be important. Far from the solid boundaries and the wake region, viscous effects, rotation/vorticity and turbulence in the flow are negligible. To identify the viscous/turbulent flow regions, we proposed in [28] to use the principal invariants of the strain and rotational rate tensors as inputs to the Gaussian mixture unsupervised model to detect two different regions, a viscous, rotational/vortical region, on the one hand, and an inviscid outer region, on the other hand, more details about the Gaussian mixture models can be found in Appendix 3. The principal invariants of the strain and rotational rate tensor are defined as follows for incompressible flows:

$$Q_S = \frac{1}{2}\left(tr((\mathbf{S}))^2 - tr\left(\mathbf{S}^2\right)\right) \;\; ; \;\; R_S = -\frac{1}{3}det(\mathbf{S}),$$

where $\mathbf{S}$ is the strain rate tensor defined as $\mathbf{S} = \frac{1}{2}(\mathbf{J} + \mathbf{J}^T)$ and $\mathbf{J} = \nabla\mathbf{U}$ is the gradient tensor of the velocity field $\mathbf{U}$. The rotational tensor $\mathbf{\Omega} = \frac{1}{2}(\mathbf{J} - \mathbf{J}^T)$ has one invariant defined as:

$$Q_\Omega = -\frac{1}{2}tr\left(\mathbf{\Omega}^2\right).$$

The features Q_S,R_S and Q_Ω will be used to train the Gaussian mixture model (GMM) to identify two distinct flow regions: a viscous/turbulent region and an inviscid outer region. The clustering will provide a node-wise partitioning of the computational domain, and made of Gauss–Legendre points that constitute the the Degrees of Freedom (DoF) in our high order numerical method. The GMM clustering provides each node with two probability memberships p_v for the viscous/rotational region and p_i for the inviscid outer region. These probabilities describe the model's estimation of the likelihood that a given node belongs to a particular region. To obtain an element-wise representation of the detected region, the mean of probability memberships belonging to each region will be computed inside each element as follows:

$$\bar{p}_v = \frac{1}{N}\sum_{j=1}^{N} p_{vj}, \qquad p_{vj} \in [0,1], \tag{1}$$

$$\bar{p}_i = \frac{1}{N}\sum_{j=1}^{N} p_{ij}, \qquad p_{ij} \in [0,1], \tag{2}$$

where $N = (P+1)^3$ are the number of DoF of each element of the mesh, and P denotes the polynomial order associated with the high-order discretization. An element will be assigned to the region with the highest mean of membership probabilities $\bar{p} = \max(\bar{p}_i, \bar{p}_v)$. The same process will be applied for all elements to supply each one with a region ID, viscous/turbulent or inviscid.

2.2 A high order discontinuous Galerkin solver

All simulations have been performed using the high-order spectral element CFD solver, HORSES3D [11]. Developed at ETSIAE-UPM (the School of Aeronautics of the Polytechnic University of Madrid), HORSES3D is a 3D parallel code designed for simulating fluid-flow phenomena. It employs the high-order discontinuous Galerkin spectral element method (DGSEM) and is implemented in modern Fortran 2003. The solver is adept at handling simulations governed by both compressible and incompressible Navier–Stokes equations and supports curvilinear hexahedral meshes of arbitrary order. The compressible Navier–Stokes equations together with details of the spatial DGSEM dicretisation are included in the appendix of this text.

HORSES3D stands out for its ability to handle anisotropic p-non-conforming elements, a key feature it offers. In our study, we exploit the adaptation capability of HORSES3D, adjusting the polynomial order uniformly within each element based on regions identified by the clustering technique. Most importantly, the solver incorporates a built-in Gaussian mixture model that is used to capture the viscous-dominated rotational regions during the run-time of the simulations, this built-in model allows the partitioning of the computational domain into two regions: viscous/turbulent and outer regions.

2.3 Viscous/turbulent-Inviscid interactions

The full LES Navier–Stokes equations incorporate convective, viscous diffusive and turbulent terms. Solving these equations in the entire computational domain can be expensive for flows that exhibit local features, such as in wakes of turbulent flows. In regions characterized by intense viscous effects, solving the full NS equations is warranted to accurately capture the intricate behavior. This ensures that viscous forces and rotational tendencies are properly accounted for a crucial understanding of boundary layer dynamics and near-wall interactions. Conversely, in regions where the flow is less influenced by viscous forces, an Euler equation, neglecting viscosity, can be solved. The Euler equation simplifies the computations by removing the need to account for viscous diffusion. The developed methodology exploits this behaviour and only computes viscous terms in the viscous domain, while in the inviscid domain, the viscous/turbulent terms will be neglected, which is equivalent to solving the Euler equation (inviscid flow). Let us note that the presented methodology affects only the viscous/turbulent terms in the governing equations, and therefore convective terms will be accounted for in the entire computational domain. In HORSES3D, there are two types of element interfaces, interior faces where the face is shared between two neighbor elements, and boundary faces which are localized at the boundaries of the computational domain. For every element we can highlight three different situations that occur when partitioning the domain into viscous/turbulent-dominated and inviscid regions:

- Inviscid region interactions: Far from boundaries we find flow regions where viscous and turbulent effects are negligible. Here, we solve the Euler equations (neglecting the viscous and turbulent terms in the equation) for both elements and we only allow for the computation of the

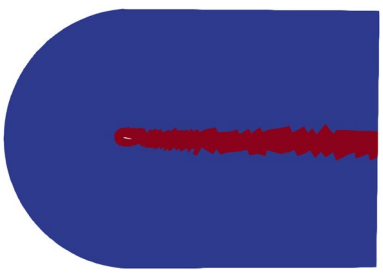


Fig. 1 Regions detected for the flow around a NACA0012 at $Re = 10^3$, $\alpha = 10°$ at $t = 42s$. Red: Viscous domain, Blue: Inviscid domain

Table 1 Comparison of numerical results from the literature for the unsteady flow past an airfoil NACA0012 at $Re = 10^3$ with $\alpha = 10°$

	C_d	C_l	St
Standard HORSES3D	0.16744	0.41658	0.870
Hybrid HORSES3D	0.16737	0.41631	0.869
Kouser et al. [19]	0.16608	0.41836	0.876
Di Ilio et al. [8]	0.15652	0.41470	–
Kurtulus [21]	0.16304	0.42058	–

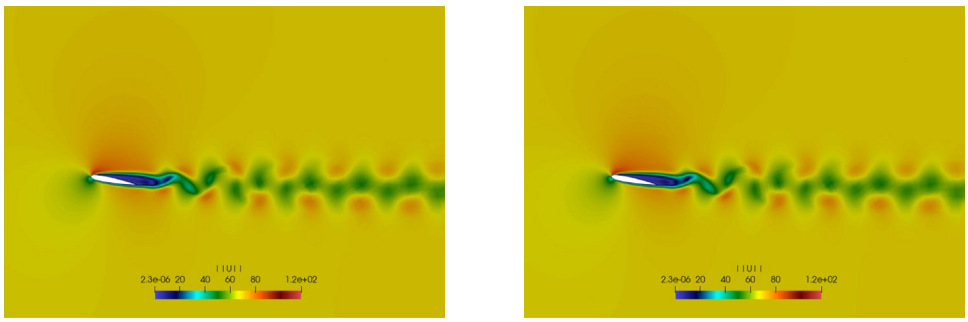
Table 2 Comparison of numerical results from the literature for the unsteady flow past an airfoil NACA0012 at $Re = 10^3$ with $\alpha = 20°$

	C_d	C_l	St
Standard HORSES3D	0.45200	0.91200	0.539
Hybrid HORSES3D	0.45196	0.91182	0.539
Kouser et al. [19]	0.44595	0.92811	0.531
Di Ilio et al. [8]	0.44705	0.90666	–
Kurtulus [21]	0.44117	0.89066	–

numerical convective flux to couple the elements. A Roe Riemann solver [29] is used as a numerical convective flux (but any other convective flux can be selected).

- Viscous and turbulent interactions: Near boundaries and within wakes, viscous and turbulent effects are important. In these elements, we solve the full LES Navier–Stokes equations for both elements and we couple the elements by computing both the numerical viscous and inviscid fluxes at the shared face. Similarly to the previous case, we use a Roe Riemann solver to compute the convective numerical flux and the BR1 viscous numerical flux [2].
- Inviscid-Viscous/Turbulent interactions: At the edge of the clustering regions, we find elements that contain inside some degrees of freedom that are inviscid and some viscous/turbulent. In these cases, the mean probability memberships (described previously) is used to decide the final nature of the element. The connection between elements with different character is naturally handled by the discontinuous Galerkin method and the fluxes between elements. We remind the reader that the discontinuous Galerkin method allows for discontinuous solutions between neighbouring elements.

Two approaches to performing the clustering are studied: Static clustering, where the identification of the viscous/turbulent regions is performed only once in the simulation after the flow is fully developed, and dynamic clustering where the identification is performed every n iterations, where n should be selected a-priori. If Δt is the time step used in the simulation then the clustering step is $\Delta c = \Delta t \times n$. First we use dynamic clustering in Sect. 3.1 to develop the viscous/turbulent-inviscid coupling strategy. Then static clustering is used in Sect. 3.2 where the viscous/turbulent-inviscid coupling strategy is complemented with the p-adaptation methodology. Both approaches yield similar levels of accuracy when detecting the regions, however dynamic clustering tends to be computationally more expensive than static clustering. In all the simulation shown in this work, time marching is conducted using a low-storage third order explicit



(a) Standard HORSES3D (b) Hybrid HORSES3D

Fig. 2 Velocity magnitude $||U||_2$ for the flow past an airfoil NACA0012 at $Re = 10^3$, $\alpha = 10°$

Table 3 Comparison of computational time of the Hybrid and Standard HORSES3D simulations for the flow around a NACA0012 at $Re = 10^3$ with a uniform polynomial order $P = 3$

Test case	Simulation type	Compt. time (s)	Reduction of Compt. time
NACA0012 $Re = 10^3$, $\alpha = 10°$	Standard HORSES3D	1.503×10^5	-
	Hybrid HORSES3D	1.298×10^5	14%
NACA0012 $Re = 10^3$, $\alpha = 20°$	Standard HORSES3D	1.554×10^5	-
	Hybrid HORSES3D	1.319×10^5	16%

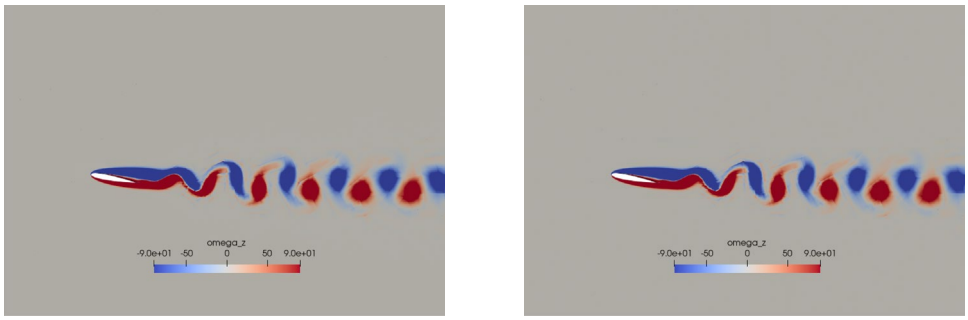


Fig. 3 Spanwise vorticity ω_z for the flow past an airfoil NACA0012 at $Re = 10^3$, $\alpha = 10°$

(a) Standard HORSES3D (b) Hybrid HORSES3D

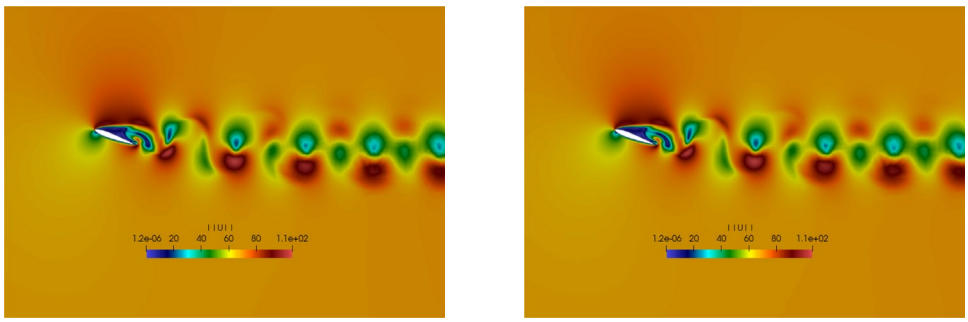


Fig. 4 Velocity magnitude $||U||_2$ for the flow past an airfoil NACA0012 at $Re = 10^3$, $\alpha = 20°$

(a) Standard HORSES3D (b) Hybrid HORSES3D

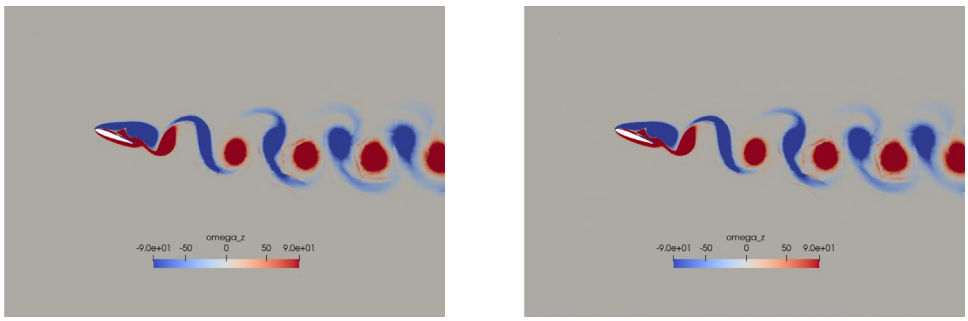


Fig. 5 Spanwise vorticity ω_z for the flow past an airfoil NACA0012 at $Re = 10^3$, $\alpha = 20°$

(a) Standard HORSES3D (b) Hybrid HORSES3D

Runge–Kutta scheme [5]. Finally, Appendix 5 includes a comparison between these two approaches.

3 Results and discussion

In this section, we first explore the advantages of hybrid simulations and then investigate the combination of hybrid simulations coupled with local polynomial adaptation.

3.1 Hybrid simulations

We apply the methodology to three distinct test cases, each increasing in complexity. Firstly, we demonstrate an unsteady laminar flow around a NACA0012 airfoil at $Re = 10^3$, considering two different angles of attack: $\alpha = 10°$ and $\alpha = 20°$. Then, we analyze an unsteady turbulent flow around a NACA0012 airfoil at $Re = 10^4$ with an angle of attack $\alpha = 10°$. Finally, we showcase the capabilities

Table 4 Comparison of numerical and experimental results from the literature for the unsteady turbulent flow past an airfoil NACA0012 at $Re = 10^4$ with $\alpha = 10°$

	Re	C_d	C_l
Standard HORSES3D	10^4	0.14586	0.55929
Hybrid HORSES3D	10^4	0.14513	0.56466
Exp: Zhou et al. [38]	1.05×10^4	0.13608	0.61043
Exp: Sunada et al. [32]	4×10^3	0.14262	0.36395
Exp: Wang et al. [36]	5.30×10^3	0.14836	0.48233

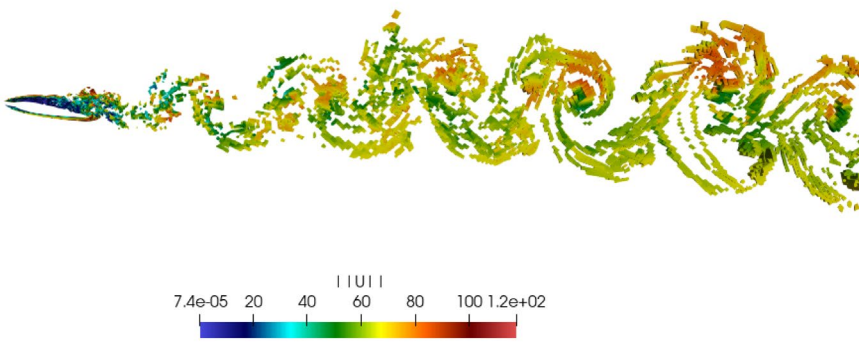
of the methodology with a flow past a wind turbine at $Re_c = 103600$. The results achieved through the use of viscous/turbulent-inviscid coupling are denoted as "Hybrid HORSES3D", while the results obtained from solving the full Navier–Stokes equations are referred to as "Standard HORSES3D".

3.1.1 Flow around a NACA0012 at $Re = 10^3$

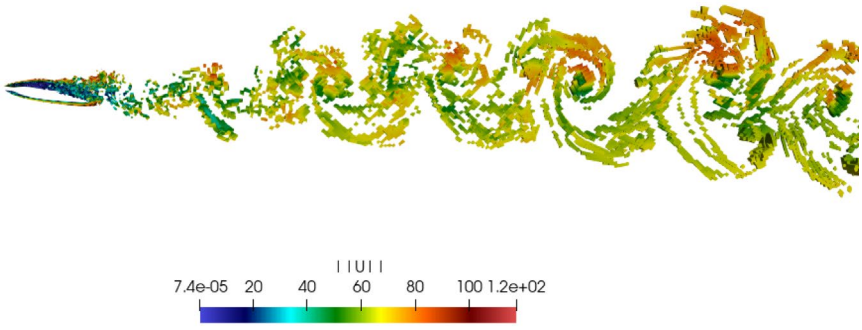
The viscous/turbulent-inviscid coupling methodology is first validated with a two-dimensional flow around a NACA0012 airfoil at $Re = 10^3$ at two different angles of attack $\alpha = 10°$ and $\alpha = 20°$. The mesh comprises 6039 elements and a uniform polynomial order of $P = 3$, resulting in a total of 193248 degrees of freedom (when accounting for the Gauss–Legendre points in each element). At this Reynolds and angles of attack, an unsteady wake will develop. We will challenge the capability of our methodology to track the regions of interest and solve the NS equations in the regions of high viscous effects as the angle of attack changes from $\alpha = 10°$ to $\alpha = 20°$. A non-dimensional time step of $\Delta t = 6 \times 10^{-5}$ is used throughout all the simulations, and the clustering step for this case is $\Delta c = 1.2 \times 10^{-3}$ resulting in performing the clustering every 20 iterations during the simulation. The Mach number for this test case is set to $M = 0.2$. Figure 1 illustrates the viscous dominated regions detected at $t = 42s$ for the flow around a NACA0012 at $Re = 10^3$ with $\alpha = 10°$, showing that the clustering methodology, detailed in 2, is able to detect the boundary layer and the wake of the airfoil where most of the viscous effects are concentrated. A convergence analysis under mesh refinement is performed in Appendix 4, showing that the clustered region is robust to mesh variation. Further details on the clustering methodology, including other examples can be found in our previous works [28, 33].

Table 5 Comparison of computational time of the Hybrid and Standard HORSES3D simulations for the flow around a NACA0012 at $Re = 10^4$ with a uniform polynomial order $P = 3$

Test case	Simulation type	Compt.time (s)	Reduction of Compt. time
NACA0012 $Re = 10^4$, $\alpha = 10°$	Standard HORSES3D	3.23×10^6	-
	Hybrid HORSES3D	2.422×10^6	25%



(a) Standard HORSES3D



(b) Hybrid HORSES3D

Fig. 6 Q-criterion contours, colored with L2 norm of velocity field $||U||_2$ for the flow around an airfoil NACA0012 at $Re = 10^4$, $\alpha = 10°$

To confirm the effectiveness of the viscous/turbulent-inviscid coupling methodology, the mean aerodynamic forces (mean drag C_d and mean lift C_l coefficients) and the Strouhal number St for $\alpha = 10°$ and $\alpha = 20°$ obtained with the "Hybrid HORSES3D" are compared in Tables 1 and 2, respectively, with those obtained with the "Standard HORSES3D" as well as the results reported in previous studies of this test case.

The average mean drag and lift coefficients are computed within an interval of time $T = 24U_\infty/c$, where U_∞ is the free-stream velocity and c is the airfoil chord. The mean drag and lift coefficients along with the Strouhal number predicted by the "Hybrid HORSES3D" for this test case exhibit close agreement with those obtained using the "Standard HORSES3D" and are consistent with findings from previous studies.

In Figs. 2, 3, 4 and 5, we compare the magnitude velocity $||U||_2$ and the spanwise vorticity ω_z for "Standard HORSES3D" and "Hybrid HORSES3D" for $\alpha = \{10°, 20°\}$.

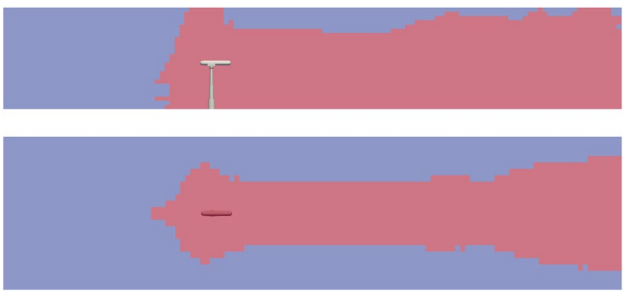


Fig. 7 Regions detected for the flow around a wind turbine: top figure shows side view of the turbine and the bottom figure shows a view from the top. Red: Viscous/turbulent domain. Blue: Inviscid domain

There are no observable differences in the velocity and vorticity profiles between the "Hybrid HORSES3D" and "Standard HORSES3D" simulations, implying that the employment of the viscous/turbulent-inviscid coupling methodology could potentially yield comparable accuracy to the "Standard HORSES3D" simulation, which solves the Navier–Stokes equations across the entire computational domain. The "Hybrid HORSES3D" solution was obtained in 14% and 16% less computational time than the "Standard HORSES3D" solution, for the angles of attack $\alpha = 10°$ and $\alpha = 20°$, respectively as shown in Table 3.

3.1.2 Flow around a NACA0012 at $Re = 10^4$

We now challenge the methodology with the unsteady turbulent three-dimensional flow around a NACA0012 at $Re = 10^4$ and an angle of attack $\alpha = 10°$. This test case has been studied by various researchers [10, 38]. Now, we consider a mesh comprised of 81420 elements and a uniform polynomial order $P = 3$ resulting in a total of degrees of freedom $DoF = 5.1 \times 10^6$. A total of 10 elements have been used to extrude the mesh in the spanwise direction with a length $L_z/c = 1$, where c is the chord of the airfoil. A large eddy simulation with a Vreman sub-grid closure turbulence model (see Appendix of this text for details) and a Mach number $M = 0.2$ has been conducted to show the effectiveness of the viscous/turbulent-inviscid coupling methodology.

Both simulations "Standard HORSES3D" and "Hybrid HORSES3D" were conducted with a non-dimensional time step $\Delta t = 5 \times 10^{-5}$. The clustering for this test case is made every 20 iterations which is equivalent to a clustering step $\Delta c = 10^{-3}$ as explained in Sect. 2. The drag and lift coefficients are averaged within an interval of time $T = 15U_\infty/c$. The obtained results of "Standard HORSES3D" and "Hybrid HORSES3D" simulations are presented in

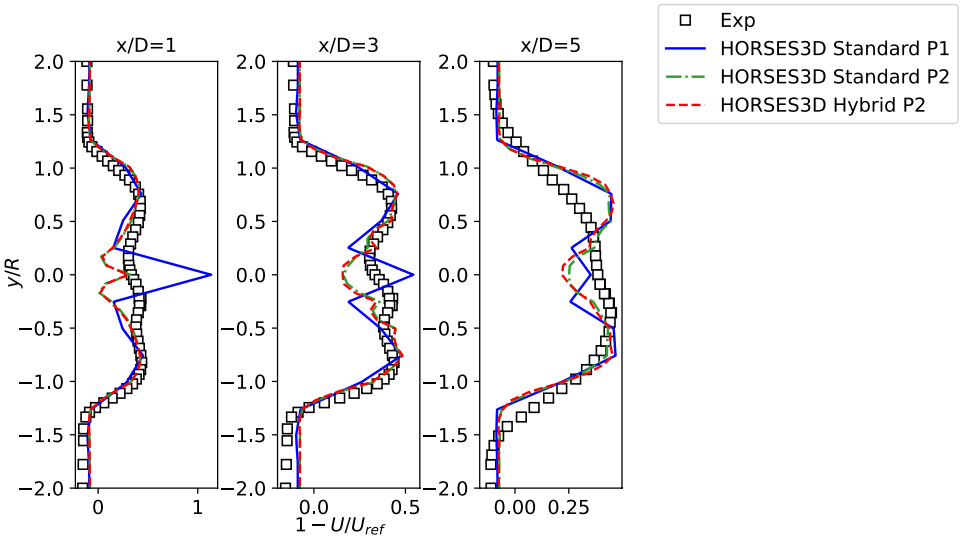
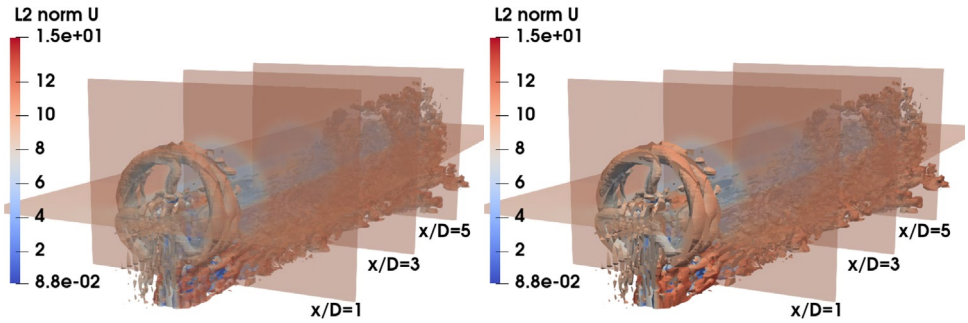


Fig. 8 Horizontal profiles of mean streamwise velocity deficit at three downstream positions x/D for "Standard HORSES3D" $P = 1$ and $P = 2$ and "Hybrid HORSES3D" $P = 2$ compared against experimental data (Exp) [20]



(a) Standard HORSES3D (b) Hybrid HORSES3D

Fig. 9 Contours of the Q-criterion, with colors representing the L2 norm of the velocity, for the flow past a wind turbine

Table 6 Comparison of computational time of the Hybrid and Standard HORSES3D simulations for the flow past a wind turbine

Test case	Simulation type	Polynomial order	Compt. time (s)	Reduction of Compt.time	$\|\|U_{P_3} - U_{P_x}\|\|_2$
Wind turbine at $Re_c = 103600$	Standard HORSES3D	2	6.25×10^5	-	0.10
	Standard HORSES3D	1	3.508×10^5	54%	1.30
	Hybrid HORSES3D	2	4.44×10^5	29%	0.11

Table 4, and compared against experimental data [32, 36, 38]. The drag coefficient obtained from "Hybrid HORSES3D" closely matches that of "Standard HORSES3D" with an error of 7×10^{-4}, and falls well in line with the results reported in the literature. The lift coefficient calculated using "Hybrid HORSES3D" aligns closely with the one predicted by "Standard HORSES3D". As pointed out by Ferrer [10], the mean lift predictions for this test case at this angle of attack are highly sensitive to the testing conditions (e.g., Reynolds number, turbulence intensity) which explains the scattering of results for experiments at comparable Reynolds numbers. In Fig. 6, the Q-criterion contours colored with velocity field magnitude are shown for both "Hybrid" and "Standard" HORSES3D. Comparing Figs. 6b, a reveals that the viscous/turbulent-inviscid coupling yields a solution with the same level of accuracy as the "Standard HORSES3D" solution. When applying the proposed methodology, we could reduce the computational cost by 25%, as summarised in Table 5.

3.1.3 Flow past a wind turbine

The viscous/turbulent-inviscid coupling strategy is now applied to the three-dimensional turbulent flow past a wind turbine at $Re_c = 103600$ (based on the blade chord). This flow has been tested experimentally at the Norwegian University of Science and Technology, the wind turbine has a diameter $D = 0.894$m, the blades are made up of NREL S826 airfoils [20]. The blind test used a wind tunnel of dimensions $[L \times W \times H] = [11.15 \times 2.71 \times 1.8]$ m, a low turbulent intensity of 0.3%, and a uniform inflow velocity. Various tip speed ratios (δ) were used in the blind test. In this study, we use the optimal tip speed ratio $\delta = \gamma D/2U_\infty = 6$, with $U_\infty = 10$ m/s and $\gamma = 134.228$ rad/s. The blade tip Reynolds number for this case is $Re_c = \delta U_\infty c_{tip}/\nu = 103600$, where $c_{tip} = 0.025926$ m is the tip chord length and ν is the kinematic viscosity of air. An immersed boundary method [11, 17, 18] has been used to model the tower and the nacelle, the rotating blades were modeled using an actuator line method [3, 25, 31]. Two large eddy simulations with a Vreman subgrid closure model and polynomial order $P \in \{1, 2\}$ have been conducted using the HORSES3D numerical framework [11]. A Cartesian mesh is generated with the same size as the wind tunnel, consisting of $[128 \times 24 \times 24]$ elements. This results in a $D/\Delta x$ ratio of approximately 10, being D the diameter of the turbine. It is important to note that, since we use a high-order method, the spatial resolution is also increased by raising the polynomial order. The Mach number is set to $M \approx 0.03$, a free slip boundary condition is used in the wind tunnel walls and no inlet turbulence is used. The total number of degrees of freedom when considering $P = 2$, is $DoF = 1.99 \times 10^6$. We run this test case using a

Table 7 Comparison of numerical and experimental results from the literature for the unsteady turbulent flow past an airfoil NACA0012 at $Re = 10^4$ with $\alpha = 10°$

	Re	C_d	C_l
Standard HORSES3D	10^4	0.14586	0.55929
Hybrid Adapted HORSES3D	10^4	0.14206	0.54822
Exp: Zhou et al. [38]	1.05×10^4	0.13608	0.61043
Exp: Sunada et al. [32]	4×10^3	0.14262	0.36395
Exp: Wang et al. [36]	5.30×10^3	0.14836	0.48233

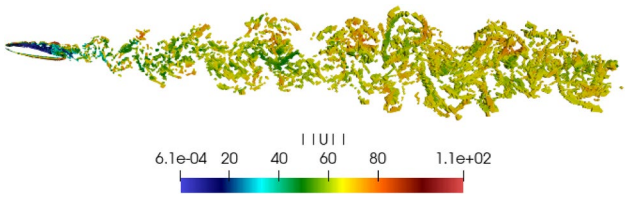


Fig. 10 Q-criterion contours, colored with L2 norm of velocity field $||U||_2$, for "Hybrid adapted HORSES3D" simulation of flow around NACA0012 airfoil at $Re = 10^4$, $\alpha = 10°$

polynomial order $P = 1$ and $P = 2$ for the "Standard HORSES3D" simulations, while we only use $P = 2$ for the "Hybrid HORSES3D" simulation. A non-dimensional time step of $\Delta t = 1.6 \times 10^{-4}$ has been used for all the simulations. For the "Hybrid HORSES3D" simulation, the clustering step is $\Delta c \approx 3.34 \times 10^{-3}$. A visualization of the viscous/turbulent areas detected for this case is presented in Fig. 7

The simulations are conducted for $T = 2\,s$. After that, the statistics are then gathered within an interval of length of $0.5\,s$. Figure 8 depicts the horizontal profiles of time-averaged streamwise velocity deficit at different downstream positions for the "Standard HORSES3D" and "Hybrid HORSES3D". These findings are compared with experimental data (Exp) [20].

We can observe a close agreement between the "Hybrid HORSES3D P2" results and the "Standard HORSESD3D P2" simulation suggesting that the viscous/turbulent-inviscid coupling yields a very similar level of accuracy, when compared to "Standard HORSES3D". The differences observed between the numerical results and the experimental data can be attributed, in part, to the requirement for increased resolutions ($P > 2$) and the utilization of more sophisticated blade modeling techniques (such as the incorporation of sliding meshes). The contours of the Q-criterion for "Hybrid HORSES3D P2" and "Standard HORSESD3D P2" are shown in Fig. 9.

We can see that the "Hybrid HORSES3D" simulation is able to capture the vortical structures in the the wake of the turbine with a similar level of accuracy compared to the "Standard HORSES3D P2", proving that the viscous/turbulent-inviscid strategy was able to recreate a solution as accurate as when solving the full NS+LES equations in the entire domain while reducing the computational cost by 29% as shown in Table 6. Finally, the table also includes the error in the outer inviscid region (relative to a P3 solution) for the Standard and Hybrid approaches. It can be seen that the errors are very similar, showing that no additional errors are included when hybridizing the simulations.

3.2 Hybrid and *P*-adapted simulations

The proposed methodology has been applied successfully to different flow regimes showing accelerations in all cases. In this section, we will combine the hybrid approach with P-adaptation. In the inviscid region, we will reduce the polynomial order while only solving the Euler equations, with the aim of further accelerating the simulations while preserving the accuracy.

Dynamic clustering has been employed to identify the viscous/turbulent regions during run-time simulations. This approach is effective when the targeted regions for detection change significantly with time. When the flow exhibits time-periodic behavior, such as in flows characterized by finite number of vortex shedding frequencies, static clustering can be used. In static clustering, the flow regions detection is performed only once in the simulation, and a representative clustering of the regions can be obtained within a vortex shedding cycle, static clustering is often preferred since it reduces the computational overhead that may result from performing the clustering dynamically during run-time simulations. A comparison between both approaches is provided in Appendix 5. There, we include a comparison between dynamic and static clustering and report computational

Table 8 Comparison of computational time of the Hybrid Adapted and Standard HORSES3D simulations for the flow past a NACA0012 at $Re_c = 10^4$

Test case	Simulation type	$P_{cluster}$	$P_{inviscid}$	Compt.time (s)	Reduction of DoF	Reduction of Compt. time
NACA0012 at $Re_c = 10^4$	Standard HORSES3D	3	3	3.23×10^6	-	-
	Hybrid Adapted HORSES3D	3	1	1.923×10^6	51%	41%

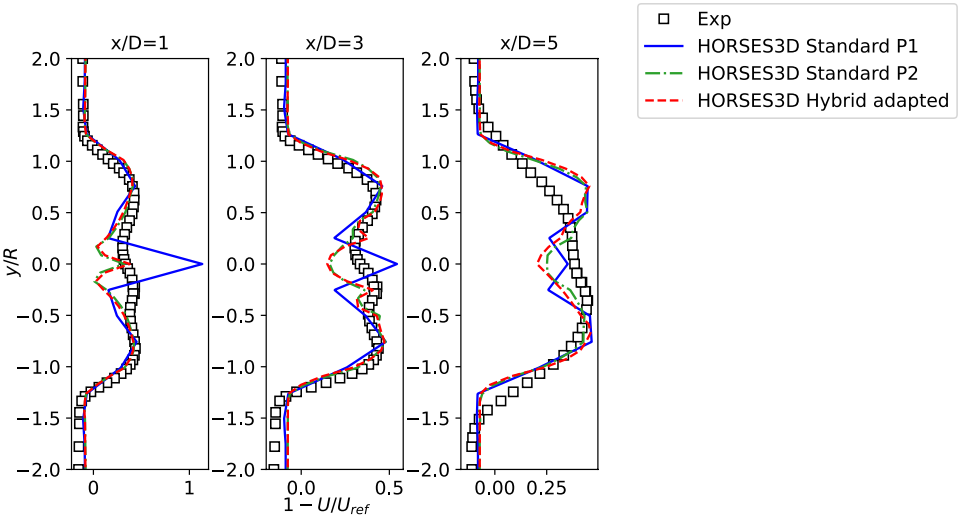


Fig. 11 Horizontal profiles of mean streamwise velocity deficit at three downstream positions x/D for "Standard HORSES3D" $P = 1$ and $P = 2$ and "Hybrid Adapted HORSES3D" $P_{cluster} = 2, P_{inviscid} = 1$ compared against experimental data (Exp) [20]

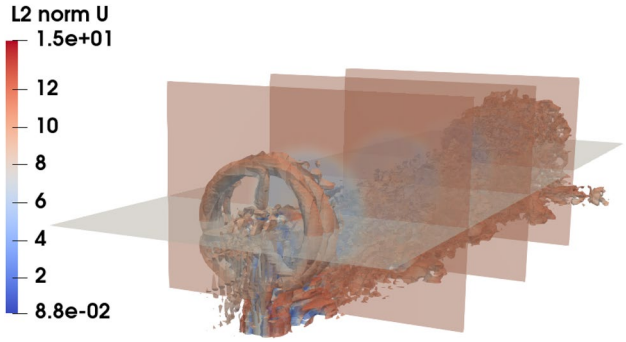


Fig. 12 Contours of the Q-criterion, with colors representing the L2 norm of the velocity field, for the "Hybrid adapted HORSES3D" flow past a wind turbine

times for both, highlighting the advantage of static clustering. In our previous work [33], we used static clustering for P-adaptation and it has been shown that once the flow is fully developed, a single snapshot can be used to cluster the flow regions. In this section and with the aim of obtaining optimal performance, we select static clustering to detect the viscous/turbulent regions.

We again consider the flow around a NACA0012 at $Re = 10^4$ with angle of attack $\alpha = 10°$ and the flow past a turbine. In the text, we refer to the combination of different sets of equations together with adaptation as: "Hybrid adapted HORSES3D" while the results obtained using uniform polynomial order as well as solving the full NS equations in the entire domain are denoted as: "Standard HORSES3D".

3.2.1 Flow around a NACA0012 at $Re = 10^4$

Following the same setup described in 3.1.2, we conduct the "Standard HORSES3D" simulation with a uniform polynomial order $P = 3$. For the "Hybrid adapted HORSES3D" simulation, we run the simulation until the flow is fully developed using $P_{init} = 3$, we perform the clustering to detect the viscous/turbulent regions in the flow field. For this simulation, we set $P_{cluster} = 3$ and $P_{inviscid} = 1$ and we restart the simulation until reaching a statistical convergence state. The diffusive and turbulent terms are accounted for only in the elements assigned to the viscous domain using the viscous/turbulent-inviscid coupling strategy presented in 2.3. In Table 7, we present the mean drag and lift coefficients predicted by the "Standard HORSES3D" and "Hybrid adapted HORSES3D", these results are compared against previous experimental and numerical studies. The mean drag predictions of "Hybrid adapted HORSES3D" match closely the ones predicted by "Standard HORSES3D".

In Sect. 3.1.2, we discussed how the mean lift predictions for this specific test case can be highly affected by the

Table 9 Comparison of computational time of the Hybrid Adapted and Standard HORSES3D simulations for the flow past a wind turbine

Test case	Simulation type	$P_{cluster}$	$P_{inviscid}$	Compt.time (s)	Reduction of DoF	Reduction of Compt. time
Wind turbine at $Re_c = 103600$	Standard HORSES3D	2	2	6.497×10^5	-	-
	Standard HORSES3D	1	1	3.508×10^5	70%	54%
	Hybrid Adapted HORSES3D	2	1	3.58×10^5	46%	45%

testing conditions. This is the reason why the experimental data shows a considerable amount of variability, which can explain the variations observed in the predictions when the Reynolds number is changed. A visualization of the Q-criterion colored with velocity magnitude for the "Hybrid adapted HORSES3D" is provided in Fig. 10.

In Table 8, we present the computational time, the reduction in the number of Degrees of Freedom (DoF), and the reduction in computational cost achieved when using the "Hybrid adapted HORSES3D" strategy.

When using the "Hybrid adapted HORSES3D" methodology, the number of DoF is reduced by 51% resulting in a mean polynomial order $P = 1.88$, and a reduction in computational cost of 41%, when compared to "Standard HORSES3D" simulation.

3.2.2 Flow past a wind turbine

To simulate this case, we utilized the setup described in 3.1.3, and run the simulations for 2 s with $P_{init} \in \{1, 2\}$. For the hybrid adapted simulation, we set $P = 2$ until the wake is fully developed, the clustering is then performed to obtain a partitioning of the domain. The simulation is restarted using the polynomial order distribution P with $P_{cluster} = 2$ and $P_{inviscid} = 1$, we continue the simulation for an additional 1 s, but only updating the viscous/turbulent terms in the elements belonging to the viscous region, following the same approach proposed in [33]. Finally, the statistics are collected within an interval of 0.5 seconds. As in 3.1.3, the horizontal profiles of mean streamwise velocity at the positions $x/D = 1, 3, 5$ downstream the turbine are illustrated in Fig. 11.

The "Hybrid adapted HORSES3D" showed close agreement with those of the "Standard HORSES3D P2", proving that the combination of the hybrid approach with local P-adaptation does not damage the accuracy of the solver. The new approach was able to reduce the number of DoF by 46% and the computational cost by 45%. The Q-criterion contours colored with the velocity magnitude for "Hybrid adapted HORSES3D" are depicted in Fig. 12.

A computational time reduction of 16% has been achieved in obtaining the numerical solution for "Hybrid Adapted HORSES3D", when compared to the computational time for "Hybrid HORSES3D P2" reported in Table 6 (see Table 9).

4 Conclusion

We have proposed a novel clustering-based viscous/turbulent-inviscid coupling strategy to accelerate high order discontinuous Galerkin solvers. This methodology can achieve a similar level of numerical accuracy as solving the full LES NS in the entire computational domain. The element-wise treatment of the DG numerical framework facilitates the implementation of such a strategy. We have shown that for solving different sets of equations (NS and Euler equations), one has to care only about the viscous numerical flux used to couple the elements at the interface between both regions. Keeping this numerical viscous flux single-valued ensures consistency and conservative properties of the scheme. By applying the viscous/turbulent-inviscid strategy, we could reduce the complexity of the governing equations in the inviscid-outer region which results in reducing the computational cost of the considered simulations, remarkably, the computational times of the NACA0012 at $Re = 10^4$ and the flow past a wind turbine at $Re_c = 103600$ were reduced by 25% and 29%, respectively.

Additionally, we have proposed to merge the viscous/turbulent-inviscid methodology with the P-adaptation strategy introduced in [33]. By combining the viscous/turbulent-inviscid coupling with P-adaptation, we achieve a significant reduction in computational costs without compromising the results accuracy. The computational times for the NACA0012 airfoil at $Re = 10^4$ were remarkably reduced by 41%. Similarly, for the flow past a wind turbine at $Re_c = 103600$, we have achieved a significant reduction in computational times, with a decrease of 45%. Finally, let us note that the presented methodology is optimal for external aerodynamics, including boundary layers and wakes. In other scenarios, where no clear distinctions exist between viscous/inviscid regions (for example internal flows) the presented methodology could perform unsuccessfully.

Compressible Navier–Stokes

In this work we solve the 3D Navier–Stokes equations for laminar cases and we supplement the equations with the Vreman LES model for turbulent flows. The 3D Navier–Stokes equations when including the Vreman model can be compactly written as:

$$\boldsymbol{u}_t + \nabla \cdot F_e = \nabla \cdot F_{v,turb}, \tag{3}$$

where $\boldsymbol{u}$ is the state vector of large scale resolved conservative variables $\boldsymbol{u} = [\rho, \rho v_1, \rho v_2, \rho v_3, \rho e]^T$, F_e are the inviscid, or Euler fluxes,

$$F_e = \begin{bmatrix} \rho v_1 & \rho v_2 & \rho u_3 \\ \rho v_1^2 + p & \rho v_1 v_2 & \rho v_1 v_3 \\ \rho v_1 v_2 & \rho v_2^2 + p & \rho v_2 v_3 \\ \rho v_1 v_3 & \rho v_2 v_3 & \rho v_3^2 + p \\ \rho v_1 H & \rho v_2 H & \rho v_3 H \end{bmatrix}, \tag{4}$$

where ρ, e, $H = E + p/\rho$, and p are the large scale density, total energy, total enthalpy and pressure, respectively, and $\vec{v} = [v_1, v_2, v_3]^T$ is the large scale resolved velocity

components. Additionally, $F_{v,turb}$ defines the viscous and turbulent fluxes,

$$F_{v,turb} = \begin{bmatrix} 0 & 0 & 0 \\ \tau_{xx} & \tau_{xy} & \tau_{xz} \\ \tau_{yx} & \tau_{yy} & \tau_{yz} \\ \tau_{zx} & \tau_{zy} & \tau_{zz} \\ \sum_{j=1}^{3} v_j \tau_{1j} + \kappa T_x & \sum_{j=1}^{3} v_j \tau_{2j} + \kappa T_y & \sum_{j=1}^{3} v_j \tau_{3j} + \kappa T_z \end{bmatrix}, \tag{5}$$

where κ is the thermal conductivity, T_x, T_y and T_z denote the temperature gradients and the stress tensor $\boldsymbol{\tau}$ is defined as $\boldsymbol{\tau} = (\mu + \mu_t)(\nabla\vec{v} + (\nabla\vec{v})^T) - 2/3(\mu + \mu_t)\boldsymbol{I}\nabla \cdot \vec{v}$, with μ the dynamic viscosity, μ_t the turbulent viscosity (in this work defined through the Vreman model) and $\boldsymbol{I}$ the three-dimensional identity matrix. Note that when solving laminar flows, it suffices to set $\mu_t = 0$ and re-interpret the large scale resolved components as the only components (there is no under-resolved components). The dynamic turbulent viscosity using the Vreman [35] model is given by:

$$\begin{aligned} \mu_t &= C_v \rho \sqrt{\frac{B_\beta}{\alpha_{ij}\alpha_{ij}}}, \\ \alpha_{ij} &= \frac{\partial u_j}{\partial x_i}, \\ \beta_{ij} &= \Delta^2 \alpha_{mi}\alpha_{mj}, \\ B_\beta &= \beta_{11}\beta_{22} - \beta_{12}^2 + \beta_{11}\beta_{33} - \beta_{13}^2 + \beta_{22}\beta_{33} - \beta_{23}^2, \end{aligned} \tag{6}$$

where $C_v = 0.07$ is the constant of the model. The Vreman LES model adjusts the model parameters based on the local flow characteristics and automatically reduces the turbulent viscosity in laminar, transitional, and near-wall regions allowing to capture the correct physics.

Spatial discretisation: discontinuous Galerkin

HORSES3D discretises the Navier–Stokes equations using the Discontinuous Galerkin Spectral Element Method (DGSEM), which is a particularly efficient nodal version of DG schemes [6]. For simplicity, here we only introduce the fundamental concepts of DG discretisations. More details can be found in [10, 24].

The physical domain is tessellated with non-overlapping curvilinear hexahedral elements, e, which are geometrically transformed to a reference element, el. This transformation is performed using a polynomial transfinite mapping that relates the physical coordinates $\vec{x}$ and the local reference coordinates $\vec{\xi}$. The transformation is applied to (3) resulting in the following:

$$J\boldsymbol{u}_t + \nabla_\xi \cdot F_e = \nabla_\xi \cdot F_{v,turb}, \tag{7}$$

where J is the Jacobian of the transfinite mapping, ∇_ξ is the differential operator in the reference space and F are the contravariant fluxes [6].

To derive DG schemes, we multiply (7) by a locally smooth test function ϕ_j, for $0 \le j \le P$, where P is the polynomial degree, and integrate over an element el to obtain the weak form

$$\int_{el} J\boldsymbol{u}_t\phi_j + \int_{el} \nabla_\xi \cdot F_e\phi_j = \int_{el} \nabla_\xi \cdot F_{v,turb}\phi_j. \tag{8}$$

We can now integrate by parts the term with the inviscid fluxes, F_e, to obtain a local weak form of the equations (one per mesh element) with the boundary fluxes separated from the interior

$$\int_{el} J\boldsymbol{u}_t\phi_j + \int_{\partial el} F_e \cdot \hat{\mathbf{n}}\phi_j - \int_{el} F_e \cdot \nabla_\xi\phi_j = \int_{el} \nabla_\xi \cdot F_{v,turb}\phi_j, \tag{9}$$

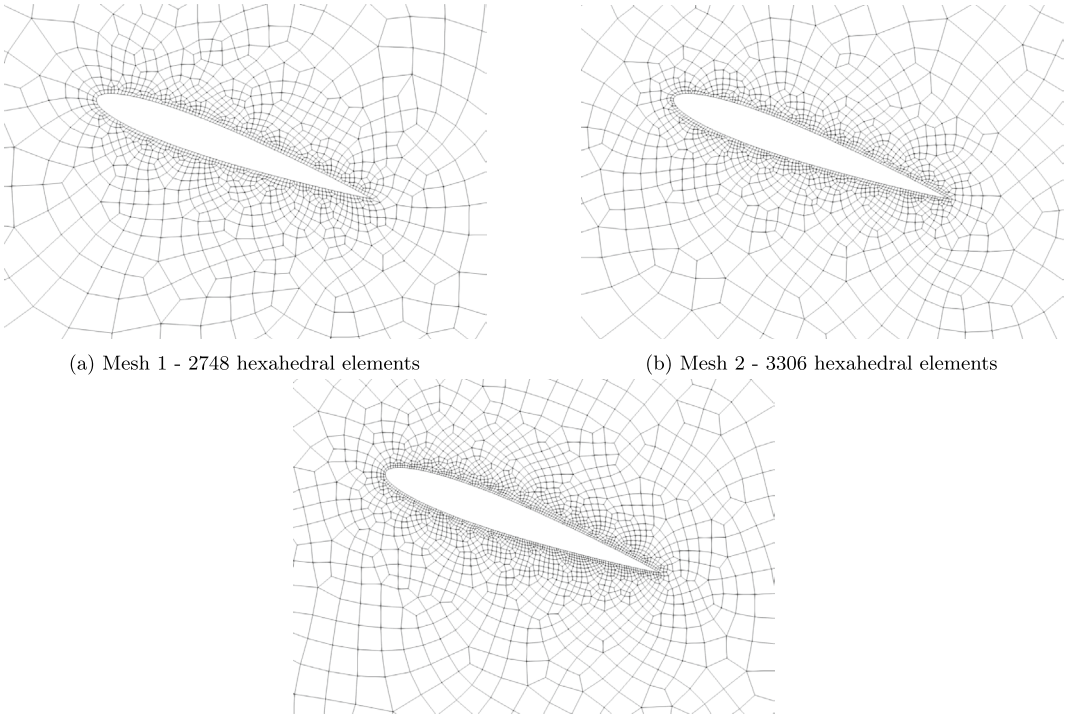
where $\hat{n}$ is the unit outward vector of each face of the reference element ∂el. We replace discontinuous fluxes at inter–element faces by a numerical inviscid flux, $F_e^\star$, to couple the elements,

$$\int_{el} J\boldsymbol{u}_t \cdot \phi_j + \int_{\partial el} F_e^\star \cdot \hat{\mathbf{n}}\phi_j - \int_{el} F_e \cdot \nabla_\xi\phi_j = \int_{el} \nabla_\xi \cdot F_{v,turb}\phi_j. \tag{10}$$

This set of equations for each element is coupled through the Riemann fluxes $F_e^\star$, which governs the numerical characteristics, see for example the classic book by Toro [34]. Note that one can proceed similarly and integrate the viscous terms by parts (see, for example, [1, 9, 12, 13]). The viscous terms require further manipulations to obtain usable discretisations (Bassi Rebay 1 and 2 or Interior Penalty). Viscous and turbulent terms are discretised following the same spatial discretisation and in this work we retain the Bassi Rebay 1 scheme. For simplicity, here we retain the volume form:

$$\begin{aligned} &\int_{el} J\boldsymbol{u}_t \cdot \phi_j + \int_{\partial el} \underbrace{F_e^\star \cdot \hat{\mathbf{n}}}_{\text{Convective fluxes}} \phi_j - \int_{el} F_e \cdot \nabla_\xi\phi_j \\ &= \int_{el} \Big(\underbrace{\nabla_\xi \cdot F_{v,turb}}_{\text{Viscous and Turbulent fluxes}} \Big) \cdot \phi_j. \end{aligned} \tag{11}$$

The final step, to obtain a usable numerical scheme, is to approximate the numerical solution and fluxes by polynomials (of order P) and to use Gaussian quadrature rules to numerically approximate volume and surface integrals. In HORSES3D we allow for Gauss–Legendre or Gauss–Lobatto quadrature points, but we only use Gauss–Legendre in this work.



(a) Mesh 1 - 2748 hexahedral elements

(b) Mesh 2 - 3306 hexahedral elements

(c) Mesh 3 - 4637 hexahedral elements

Fig. 13 Close-up of the three meshes for the NACA0012 airfoil

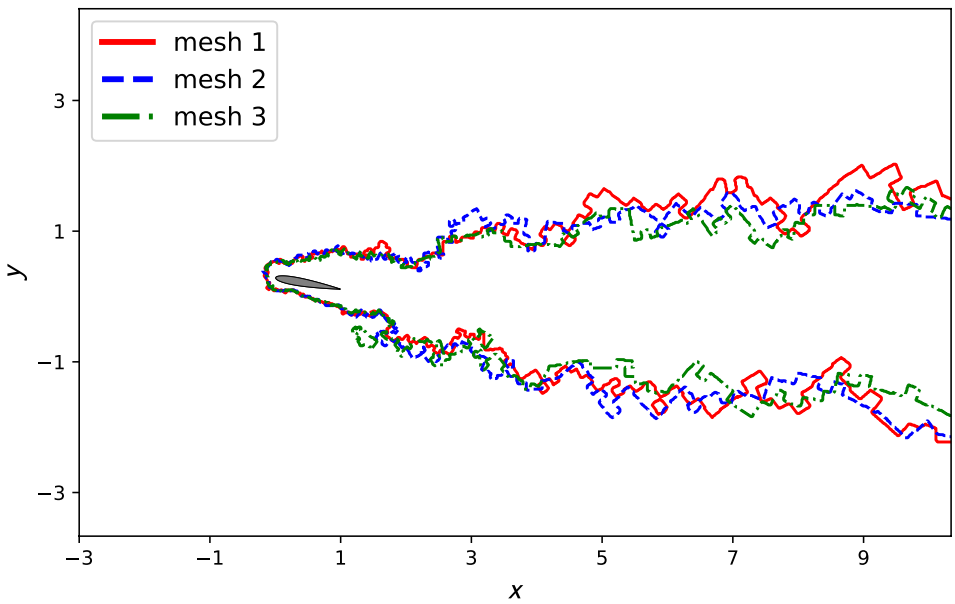


Fig. 14 Boundaries of the viscous/rotational clusters obtained for the flow around a NACA0012 at $Re = 1000$ and $\alpha = 10°$

Gaussian Mixture Models (GMM)

Gaussian mixture is an unsupervised learning algorithm designed to represent the data as a combination of multiple Gaussian distributions (one per cluster). Given an m-dimensional random vector y, the probability density function f of a finite mixture of normal distributions can be written as

Table 10 Comparison between dynamical and static clustering

Test case	Dynamical clustering time over Comp.time	Static clustering time over Comp. time
NACA0012, $Re = 10^3$, $\alpha = 10°$	2×10^{-3}	3.7×10^{-10}
NACA0012, $Re = 10^3$, $\alpha = 20°$	7×10^{-3}	7×10^{-8}
NACA0012, $Re = 10^4$, $\alpha = 10°$	8×10^{-3}	7.4×10^{-5}
Wind turbine, $Re_c = 103600$	3×10^{-2}	1.5×10^{-7}

$$f(y, \Theta) = \sum_{i=1}^{c} p_i N_i(y, \theta_i), \tag{12}$$

where $\Theta = (\theta_1, \theta_2, ..., \theta_c, p_1, p_2, ..., p_c)$ denotes the vector of unknown parameters, c denotes the number of normal distributions, and p_i is the mixing proportion illustrating the contribution of the normal distributions, N_i, when describing the data. In the context of CFD simulations, the vector y is created from a finite set of n variables $y = \{y_i\}_{i=1}^{n}$ evaluated at each degree of freedom. The aim is to assign each degree of freedom by evaluating the function f and the probability membership τ_i given by:

$$\tau_i(y_j, \hat{\Theta}) = \hat{p}_i \frac{N_i(y_j, \hat{\theta}_i)}{f(y_j, \hat{\Theta})} \; \forall j = 1, ..., n. \tag{13}$$

Each degree of freedom is assigned to the cluster N_i with the highest probability membership τ_i. $\hat{\Theta}$ is an estimation of Θ obtained by solving the following likelihood equation:

$$\frac{\partial \log(L(\Theta))}{\partial \Theta} = 0, \tag{14}$$

where L is the likelihood function of Θ:

$$\log L(\Theta) = \sum_{j=1}^{n} \log \left(\sum_{i=1}^{c} p_i N_i(y_j, \theta_i) \right). \tag{15}$$

Eq. 14 is solved by applying the Expectation-Maximization (EM) method [7].

Mesh resolution (h-mesh)

In this section, we study how the initial h-mesh affects the clustering of viscous/rotational regions. We perform the analysis for the flow around the NACA0012 airfoil at $Re = 1000$ and $\alpha = 10°$. We use three meshes with varying numbers of hexahedral elements: $N_e \in \{2748, 3306, 4637\}$. The minimum element sizes for the three meshes are: 6×10^{-3}, 5×10^{-3} and 4×10^{-3}, respectively. Figure 13 shows a close-up of the airfoil profile for the three meshes. The boundaries of the viscous/rotational clusters are shown in Fig. 14. The results obtained show that the initial h-mesh resolution has minimal effect on the obtained clustering. These results confirm our previous findings in [33], where we conducted a similar analysis for varying polynomial orders (on a fixed h-mesh).

A comparison between dynamic and static clustering

In this section, we show the impact of employing dynamic clustering vs using static clustering. Unlike static clustering, which relies on a single flow snapshot (after the flow has been fully developed or statistically converged), dynamic clustering is performed, with predefined intervals, during the simulation. Table 10, shows the ratios of dynamical and static clustering costs from the total computational time of the simulations considered in this work are presented.

These results are obtained by dividing the clustering cost for both dynamic and static cases over the total computational time of the simulations. The results reveal that dynamic clustering is costly compared to static clustering for all the considered test cases. However, let us note that performing static clustering requires the flow to be fully developed, moreover, static clustering is only valid for the flows that have time-periodic behavior whereas dynamic clustering can be applied to a more general category of fluid flows (e.g. fluids that do not have main vortex shedding frequency). Even in the cases where static clustering is applicable, we typically collect several flow snapshots to confirm that all can be represented by a single cluster, see examples in our previous work [28]. In general, static clustering (when applicable), is more efficient while providing similar levels of accuracy, see also [33].

Acknowledgements Kheir-Eddine Otmani aknowledges the Grant 072 Bis/PG/Espagne/2020-2021 of Ministère de l'Enseignement Supérieur et de la Recherche Scientifique, République Algérienne Démocratique et Populaire. Gonzalo Rubio and Esteban Ferrer acknowledge the funding received by the Grant DeepCFD (Project No. PID2022-137899OB-I00) funded by MICIU/AEI/10.13039/501100011033 and by ERDF, EU. Esteban Ferrer would like to thank the support of Agencia Estatal de Investigación (for the grant "Europa Excelencia 2022" Proyecto EUR2022-134041/AEI/10.13039/501100011033) y del Mecanismo de Recuperación y Resiliencia de la Unión Europea. This research has been cofunded by the European Union (ERC, Off-coustics, project number 101086075). Views and opinions expressed are, however, those of the author(s) only and do not necessarily reflect those of the European Union or the European Research Council. Neither the European Union nor the granting authority can be held responsible for them. Finally, all authors gratefully acknowledge the Universidad Politécnica

de Madrid (www.upm.es) for providing computing resources on Magerit Supercomputer.

Author contributions All authors contributed to the conceptualization and methodology of this work. K.O and E.F wrote the main manuscript, K.O prepared all the figures in the text, all the authors reviewed and corrected the text

Funding Open Access funding provided thanks to the CRUE-CSIC agreement with Springer Nature.

Data Availability The data that support the findings of this study are available from the corresponding author upon reasonable request.

Declarations

Competing interests The authors declare no competing interests.

References

1. Arnold D, Brezzi F, Cockburn B, Marini L (2001) Unified analysis of discontinuous Galerkin methods for elliptic problems. SIAM J Numer Anal 39(5):1749–1779
2. Bassi F, Rebay S (1997) A high-order accurate discontinuous finite element method for the numerical solution of the compressible Navier-Stokes equations. J Comput Phys 131(2):267–279
3. Botero-Bolívar L, Marino OA, Venner CH, de Santana LD, Ferrer E (2024) Low-cost wind turbine aeroacoustic predictions using actuator lines. Renew Energy 227:120476
4. Callaham JL, Koch JV, Brunton BW, Kutz JN, Brunton SL (2021) Learning dominant physical processes with data-driven balance models. Nat Commun 12(1):8–9
5. Carpenter M, Kennedy C (1994) Fourth-order 2n-storage Runge-Kutta schemes. Nasa Rep TM 109112:07
6. Kopriva DA (2009) Implementing spectral methods for partial differential equations. Springer Netherlands
7. Dempster AP, Laird NM, Rubin DB (1977) Maximum likelihood from incomplete data via the em algorithm. J R Stat Soc Ser B (Methodol) 39(1):1–22
8. Di Ilio G, Chiappini D, Ubertini S, Bella G, Succi S (2018) Fluid flow around naca 0012 airfoil at low-Reynolds numbers with hybrid lattice Boltzmann method. Comput Fluids 166:200–208
9. Ferrer E (2012) A high order Discontinuous Galerkin-Fourier incompressible 3D Navier-Stokes solver with rotating sliding meshes for simulating cross-flow turbines. PhD thesis, Oxford University, UK
10. Ferrer E (2017) An interior penalty stabilised incompressible discontinuous Galerkin-Fourier solver for implicit large eddy simulations. J Comput Phys 348:754–775
11. Ferrer E, Rubio G, Ntoukas G, Laskowski W, Mariño O, Colombo S, Mateo-Gabín A, Marbona H, Manrique de Lara F, Huergo D, Manzanero J, Rueda-Ramírez A, Kopriva D, Valero E (2023) Horses3d: a high-order discontinuous Galerkin solver for flow simulations and multi-physics applications. Comput Phys Commun 287:108700
12. Ferrer E, Willden R (2011) A high order discontinuous Galerkin finite element solver for the incompressible Navier-Stokes equations. Comput Fluids 46(1):224–230. In: 10th ICFD Conference Series on Numerical Methods for Fluid Dynamics (ICFD 2010)
13. Ferrer E, Willden R (2012) A high order discontinuous Galerkin-Fourier incompressible 3D Navier-Stokes solver with rotating sliding meshes. J Comput Phys 231(21):7037–7056
14. Ims J, Wang Z (2023) A comparison of three error indicators for adaptive high-order large eddy simulation. J Comput Phys 490:112312
15. Kompenhans M, Rubio G, Ferrer E, Valero E (2016) Adaptation strategies for high order discontinuous Galerkin methods based on tau-estimation. J Comput Phys 306:216–236
16. Kompenhans M, Rubio G, Ferrer E, Valero E (2016) Comparisons of p-adaptation strategies based on truncation- and discretisation-errors for high order discontinuous Galerkin methods. Comput Fluids 139:36–46. In: 13th USNCCM International Symposium of High-Order Methods for Computational Fluid Dynamics—a special issue dedicated to the 60th birthday of Professor David Kopriva
17. Kou J, de Mendoza AH, Joshi S, Le Clainche S, Ferrer E (2022) Eigensolution analysis of immersed boundary method based on volume penalization: applications to high-order schemes. J Comput Phys 449:110817
18. Kou J, Joshi S, de Mendoza AH, Puri K, Hirsch C, Ferrer E (2022) Immersed boundary method for high-order flux reconstruction based on volume penalization. J Comput Phys 448:110721
19. Kouser T, Xiong Y, Yang D, Peng S (2021) Direct numerical simulations on the three-dimensional wake transition of flows over naca0012 airfoil at $re = 1000$. Int J Micro Air Veh 13:17568293211055656
20. Krogstad P, Eriksen P (2013) "blind test'' calculations of the performance and wake development for a model wind turbine. Renew Energy 50:325–333
21. Kurtulus DF (2019) Unsteady aerodynamics of a pitching naca 0012 airfoil at low Reynolds number. Int J Micro Air Veh 11:1756829319890609
22. Li B, Yang Z, Zhang X, He G, Deng B-Q, Shen L (2020) Using machine learning to detect the turbulent region in flow past a circular cylinder. J Fluid Mech 905:A10
23. Lokatt M, Eller D (2017) Robust viscous-inviscid interaction scheme for application on unstructured meshes. Comput Fluids 145:37–51
24. Manzanero J, Rubio G, Kopriva DA, Ferrer E, Valero E (2020) Entropy–stable discontinuous Galerkin approximation with summation–by–parts property for the incompressible Navier–Stokes/Cahn–Hilliard system. J Comput Phys 408:109363
25. Marino OA, Sanz R, Colombo S, Sivaramakrishnan A, Ferrer E (2024) Modelling wind turbines via actuator lines in high-order h/p solvers. arXiv:2406.09993
26. Mateo-Gabín A, Tlales K, Valero E, Ferrer E, Rubio G (2023) Unsupervised machine-learning shock-capturing technique for high-order solvers. arXiv:2308.00086
27. Moro D, Nguyen C, Peraire J, Drela M (2013) Advances in the development of a high order, viscous-inviscid interaction solver. In: 21st AIAA Computational Fluid Dynamics Conference, p 2943
28. Otmani K-E, Ntoukas G, Mariño OA, Ferrer E (2023) Toward a robust detection of viscous and turbulent flow regions using unsupervised machine learning. Phys Fluids 35(2):027112

29. Roe P (1981) Approximate Riemann solvers, parameter vectors, and difference schemes. J Comput Phys 43(2):357–372
30. Saetta E, Tognaccini R (2023) Identification of flowfield regions by machine learning. AIAA J 61(4):1503–1518
31. Sorensen J, Shen W (2002) Numerical modeling of wind turbine wakes. J Fluids Eng 124(2):393–399
32. Sunada S, Sakaguchi A, Kawachi K (1997) Airfoil section characteristics at a Low Reynolds number. J Fluids Eng 119(1):129–135
33. Tlales K, Otmani KE, Ntoukas G et al (2024) Machine learning mesh-adaptation for laminar and turbulent flows: applications to high-order discontinuous Galerkin solvers. Engineering with Computers. https://doi.org/10.1007/s00366-024-01950-y
34. Toro E (2009) Riemann solvers and numerical methods for fluid dynamics: a practical introduction. Springer, Berlin
35. Vreman AW (2004) An eddy-viscosity subgrid-scale model for turbulent shear flow: algebraic theory and applications. Phys Fluids 16(10):3670–3681
36. Wang S, Zhou Y, Alam MM, Yang H (2014) Turbulent intensity and Reynolds number effects on an airfoil at low Reynolds numbers. Phys Fluids 26(11):115107
37. Zhang S, Galbraith M, Allmaras S, Drela M, Darmofal DL (2017) A non-parametric discontinuous Galerkin formulation of the integral boundary layer equations with strong viscous/inviscid coupling. In: 23rd AIAA computational fluid dynamics conference. https://doi.org/10.2514/6.2017-4278
38. Zhou Y, Alam MM, Yang H, Guo H, Wood D (2011) Fluid forces on a very low Reynolds number airfoil and their prediction. Int J Heat Fluid Flow 32(1):329–339