HAL
open science

Global and local POD models for the prediction of compressible flows with DG methods

Andrea Ferrero, Angelo Iollo, Francesco Larocca

To cite this version:

Andrea Ferrero, Angelo Iollo, Francesco Larocca. Global and local POD models for the prediction of compressible flows with DG methods. International Journal for Numerical Methods in Engineering, 2018, 10.1002/nme.5927 . hal-01908303

HAL Id: hal-01908303

https://inria.hal.science/hal-01908303v1

Submitted on 9 Jan 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

This is the accepted version of the following article: *Ferrero, Andrea, Angelo Iollo, and Francesco Larocca. "Global and local POD models for the prediction of compressible flows with DG methods." International Journal for Numerical Methods in Engineering 116.5 (2018): 332-357*, which has been published in final form at https://onlinelibrary.wiley.com/doi/full/10.1002/nme.5927
This article may be used for non-commercial purposes in accordance with the Wiley Self-Archiving Policy [https://authorservices.wiley.com/author-resources/Journal-Authors/licensing-open-access/open-access/self-archiving.html]

Global and local POD models for the prediction of compressible flows with DG methods

Andrea Ferrero*[1], Angelo Iollo[2,3,4], and Francesco Larocca[1]

[1] *Politecnico di Torino, 10129 Torino, Italy*
[2] *MEMPHIS Team, INRIA, F-33400 Talence, France*
[3] *Univ. Bordeaux, IMB, UMR 5251, F-33400 Talence, France*
[4] *CNRS, IMB, UMR 5251, F-33400 Talence, France*

Abstract

Proper Orthogonal Decomposition (POD) allows to compress information by identifying the most energetic modes obtained from a database of snapshots. In this work, POD is used to predict the behavior of compressible flows by means of global and local approaches which exploit some features of a discontinuous Galerkin spatial discretization. The presented global approach requires the definition of high-order and low-order POD bases which are built from a database of high-fidelity simulations. Predictions are obtained by performing a cheap low-order simulation whose solution is projected on the low-order basis. The projection coefficients are then used for the reconstruction with the high-order basis.
However, the non-linear behavior related to the advection term of the governing equations makes the use of global POD bases quite problematic. For this reason, a second approach is presented in which an empirical POD basis is defined in each element of the mesh. This local approach is more intrusive with respect to the global approach but it is able to capture better the non-linearities related to advection. The two approaches are tested and compared on the inviscid compressible flow around a gas-turbine cascade and on the compressible turbulent flow around a wind turbine airfoil.

Keywords— Reduced Order Model, POD, Compressible flows, RANS, Discontinuous Galerkin

1 Introduction

Reduced Order Models have been extensively investigated as low cost prediction methods in several fields of engineering. As far as fluid mechanics is concerned there are several different approaches: Proper Orthogonal Decomposition (POD) [15, 18, 20, 31, 36, 42, 43, 46, 47, 53], Proper Generalised Decomposition (PGD) [22], Reduced Basis (RB) [40, 41], Dynamic Mode Decomposition [45], Grassmannian interpolation [4], Empirical Interpolation Method (EIM) [9], Discrete Empirical Interpolation Method (DEIM) [21], hyper-reduction [6, 44, 54], regularized ROM [52], domain decomposition approaches [8, 13, 16]. In this work the attention is focused on the use of POD for the prediction of compressible steady flows described by the Euler equations or by the Reynolds Averaged Navier-Stokes (RANS) equations. The POD algorithm, also known as Principal Component Analysis in other fields, allows to compress the information contained in a database of snapshots by extracting the most energetic modes. After the definition of the POD basis there are several approaches to exploit it for performing predictive simulations. A classical approach consists in projecting the governing equations of the problem onto the POD basis by means of a Galerkin projection [11, 36, 43]: this leads to a set of ordinary differential equations which describe the evolution of the coefficients of the modes.
When steady problems are considered an effective approach is based on the evaluation of the POD coefficients for the configurations available in the database: the prediction of new configurations will be performed by interpolating the POD coefficients in the parameter space (for example by means of Kriging interpolation [14]).

*Corresponding author: andrea_ferrero@polito.it

In this work an alternative approach is proposed by defining a two levels description of the solutions in the database. In particular, each sampling point in the parameter space is investigated by performing a cheap low-order and an expensive high-order spatial discretization. The results are used to build two POD bases at different levels of accuracy. The prediction of new configurations is carried out by performing a cheap low-order simulation and projecting its solution on the low-order POD basis. The coefficients obtained by this projection are then used to reconstruct the solution with the high-order POD basis.
The space discretization adopted in the high-fidelity scheme used for this work is particularly suited to this multi-level description of the solution. In particular, a discontinuous Galerkin method is used to discretise the governing equations in space and a modal hierarchical orthonormal basis is employed to describe the solution in each element.

The previously described approach, which will be referred in the following as two levels global (TLG) method, exploits global POD modes which are defined on the entire computational domain. Since the POD modes are linear combinations of the snapshots and the predicted solution is a linear combination of POD modes, this approach assumes that the predicted solution can be reasonably approximated as a linear combination of the previously computed solutions. However, the presence of the non-linear advection term in the fluid mechanics equations makes this assumption difficult to be satisfied for several flows. In particular, the non-linear phenomena related to the advection of structures in the flow field can be accurately described by global POD modes only by using a very fine sampling of the parameter space. This means that, in order to get a reasonable accuracy in the low cost prediction, it is necessary to spend a lot of time in the offline phase for the generation of a very rich database. This is the price to pay for the description of a non-linear effect by means of a linear method. In such a situation the use of global modes could become questionable since the cost required to generate the database could be potentially larger than the cost that would be spent if the predictions were directly performed by using the high-fidelity solver.

An alternative approach is considered in this work by exploiting some properties of the chosen DG discretization. The idea is to substitute in each element the original DG basis with a smaller but still accurate empirical POD basis (LPOD-DG). Since the number of degrees of freedom contained in each DG element is quite large when high-order schemes are considered, there is the possibility to significantly reduce the cost of the predictive simulation. In other words, if the non-linearities are so strong that the global approach would require a prohibitive offline cost it would be natural to directly use the high-fidelity solver in the predictions. However, the flexibility of the DG discretization allows to use a different basis in each element: in this way it is possible to introduce local empirical bases which contain more degrees of freedom in the regions where non-linear effects are stronger. The predicted solution is still obtained by the integration of the discretized governing equations but the global number of degrees of freedom can be significantly reduced.

The LPOD-DG method presented in this paper shares the same spirit of the reduced basis local approaches proposed for the solution of elliptic problems by Maday and Ronquist [35] and Kaulmann et al. [33]. In particular, Maday and Ronquist [35] study a thermal problem by decomposing the domain in a set of elements and introducing a reduced basis in each elements of a continuous Galerkin discretization: the solutions at the interfaces between the elements are then coupled by means of a mortar-type method. Kaulmann et al. [33] study elliptic multiscale problems by means of a discontinuous Galerkin approach. In the LPOD-DG method proposed in this paper, both the original DG and the POD bases are defined in each element in order to make the method less intrusive and with the purpose of defining an error indicator based on the residuals in the original DG space. Furthermore, the present work is focused on non-linear convection dominated problems since Euler equations and RANS equations at high Reynolds numbers are considered.
Finally, the LPOD-DG method can be put in relationship with the domain decomposition method proposed by Baiges et al. [8] for the continuous Galerkin discretization of the incompressible Navier-Stokes equations. In particular, Baiges et al. [8] propose to decompose the domain in several regions and to define a local POD basis in each region. They suggest to define the local POD basis by a restriction of a global POD basis (LG-POD) or by a local application of the POD procedure (L-POD). In both cases, they observe some stability issues which they solve by introducing some techniques to match the

POD solutions at the interface between the different regions. In particular, they suggest to penalize the difference between the POD coefficients (in the LG-POD approach) or to introduce an overlapping (in the L-POD approach). In the present work, the use of local POD approaches is exploited in the framework of DG schemes which allow to intrinsically deal with the interface problem.
The LPOD-DG method described here exploits the locality concept in the physical space by defining a different POD basis for each element of the mesh. This approach is different to what is done in several local ROM approaches available in the literature[5, 7, 23, 55] where the locality concept is exploited in the parameter space: these approaches are local in the sense that the modes (defined on the full physical domain) are built from a subset of the database which contains only the sampling points closer to the prediction point. The LPOD-DG method presented in this work exploits the locality concept both in the physical and in the parameter space as shown in the following.

The paper is organized as follows. In Section 2 the governing equations are presented. In Section 3 the numerical methods used for performing the high-fidelity simulations are described. In Section 4 the POD algorithm is reported. In Sections 5 and 6 the two level global method and the local method are presented, respectively. In Section 7 the two methods are tested and compared on the prediction of the compressible inviscid flow around a gas turbine cascade which interacts with an upstream perturbation. Finally, in Section 8 the two methods are tested on the prediction of the turbulent flow around a wind turbine blade. The conclusions and the future perspectives are reported in Section 9.

2 Physical model

The problems considered in this work are described by the two-dimensional Euler or RANS equations defined on the spatial domain Ω. The RANS equations with the Spalart-Allmaras turbulence closure [2] are reported in the following:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \boldsymbol{u}) = 0 \tag{1}$$

$$\frac{\partial}{\partial t}(\rho \boldsymbol{u}) + \nabla \cdot (\rho \boldsymbol{u}\boldsymbol{u}) = -\nabla P + \nabla \cdot \boldsymbol{\tau} \tag{2}$$

$$\frac{\partial E}{\partial t} + \nabla \cdot (\boldsymbol{u}(E+P)) = \nabla \cdot (\tau \cdot \boldsymbol{u} - \boldsymbol{\Phi}) \tag{3}$$

$$\frac{\partial \rho \hat{\nu}}{\partial t} + \nabla \cdot (\rho \boldsymbol{u} \hat{\nu}) = \rho(\tilde{P} - \tilde{D}) + \frac{1}{\sigma}\nabla \cdot (\rho(\nu + \hat{\nu})\nabla \hat{\nu}) + \frac{c_{b2}}{\sigma}\rho(\nabla \hat{\nu})^2 - \frac{1}{\sigma}(\nu + \hat{\nu})\nabla \rho \cdot \nabla \hat{\nu} \qquad \boldsymbol{x} \in \Omega \subset \mathbb{R}^2, t \in 0^+ \tag{4}$$

where ρ, $\boldsymbol{u}$, P, E, ν, $\hat{\nu}$, $\boldsymbol{x}$ and t are density, velocity, pressure, total energy per unit volume, molecular viscosity, modified eddy viscosity, spatial position and time, respectively. The following equation for the energy is considered:

$$E = \frac{P}{\gamma - 1} + \frac{1}{2}\rho \boldsymbol{u} \cdot \boldsymbol{u} \tag{5}$$

where γ is the specific heat ratio.
The viscous stress tensor $\boldsymbol{\tau}$ includes both the molecular and eddy viscosity contributions and its components are given by:

$$\tau_{ij} = 2\rho(\nu + \hat{\nu} f_{v1})\left(\frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) - \frac{1}{3}\frac{\partial u_k}{\partial x_k}\delta_{ij}\right) \tag{6}$$

The model terms $\tilde{P}$, $\tilde{D}$ and f_{v1} and the constants σ, c_{b2} are defined in [2]. Finally, the heat flux $\boldsymbol{q}$ is described by the Fourier's law:

$$\boldsymbol{\Phi} = -\left(\frac{c_p \mu}{Pr} + \frac{c_p \rho \hat{\nu} f_{v1}}{Pr_t}\right)\nabla T \tag{7}$$

where T, c_p, Pr and Pr_t are the temperature, the constant pressure specific heat capacity, the Prandtl number and the turbulent Prandtl number.

The Euler equations can be obtained by considering Eqs. 1-3 and neglecting viscous terms. The previous equations can be written in the following compact form:

$$\frac{\partial \boldsymbol{q}}{\partial t} + \nabla \cdot \boldsymbol{F} = \boldsymbol{S} \qquad \boldsymbol{q}, \boldsymbol{S} \in \mathbb{R}^n, \qquad \boldsymbol{F} \in \mathbb{R}^{n \times 2} \tag{8}$$

where n is the number of governing equations ($n = 4$ for Euler equations and $n = 5$ for Spalart-Allmaras RANS equations), $\boldsymbol{q}$ is the vector of the conservative variables, $\boldsymbol{F}$ contains the fluxes defined in equations 1-4 and $\boldsymbol{S}$ is a source term (active only for Eq.4).

The goal function evaluated in the first test case considered in this work is related to the total pressure (P°) which is defined as follows:

$$P^\circ = P\left(1 + \frac{\gamma - 1}{2} M^2\right)^{\frac{\gamma}{\gamma-1}} \tag{9}$$

where $M = |\boldsymbol{u}|/\sqrt{\gamma \mathcal{R} T}$ is the Mach number and $\mathcal{R}$ is the gas constant. The fluid is supposed to follow the ideal gas law ($P = \rho \mathcal{R} T$).

All the equations and the results in the following will be considered dimensionless after normalization with respect to a reference pressure (P_{ref}), a reference temperature (T_{ref}), a reference length (L_{ref}), a reference viscosity (μ_{ref}) and a reference speed ($U_{ref} = \sqrt{\mathcal{R} T_{ref}}$).

3 High-fidelity discretization

The governing equations are discretized by means of the method of lines: the discontinuous Galerkin method is adopted for the space discretization while a linearized implicit Euler scheme is used for the time integration.

3.1 Discontinuous Galerkin space discretization

The spatial domain Ω is discretized with a collection of non-overlapping elements Ω_e. The numerical solution is searched inside a finite dimensional functional space V_h which is spanned by polynomial functions of degree at most p continuous only inside each element. Consider a basis for V_h with functions $\phi(\boldsymbol{x})$. Denote as N_e the size of the basis in the e-th element. The vector of conservative variables $\boldsymbol{q} \in \mathbb{R}^n$ can be approximated as:

$$q_l(\boldsymbol{x}, t) = \sum_{i=1}^{N_e} \tilde{q}_{li}(t) \phi_i(\boldsymbol{x}) \qquad 1 \leq l \leq n \tag{10}$$

and $\tilde{\boldsymbol{q}} \in \mathbb{R}^{n \times N_e}$ contains the degrees of freedom inside the element.
In this work a modal basis obtained by the application of the modified Gram-Schmidt orthonormalisation to a set of monomials defined in the physical space is adopted, following the approach of Bassi et al. [10]. The size of the basis is related to the reconstruction order by the following relation:

$$N_e = \frac{(p+1)(p+2)}{2} \tag{11}$$

Note that the same basis functions are used for all the conservative variables and so there is the same number of degrees of freedom for the different conservative variables.
In the following, a DG scheme which uses a reconstruction order p will be identified as DGp.
The weak formulation of the problem is obtained by multiplying the governing equations by an arbitrary test function belonging to V_h and then integrating by parts. This leads to the following set of ordinary differential equations on each element:

$$\sum_{i=1}^{N_e} \frac{d\tilde{q}_{li}}{dt} \int_{\Omega_e} \phi_i \phi_j d\Omega = \int_{\Omega_e} \sum_{m=1}^{2} \frac{\partial \phi_j}{\partial x_m} F_{lm} d\Omega - \int_{\partial \Omega_e} \phi_j \sum_{m=1}^{2} \bar{F}_{lm} n_m d\sigma + \int_{\Omega_e} S_l \phi_j d\Omega \qquad 1 \leq l \leq n, \quad 1 \leq j \leq N_e \tag{12}$$

where $\boldsymbol{n}$ is the outward unit normal vector. The flux F_{lm} in the volume integral is computed by means of the element's degrees of freedom. Since the solution is not continuous across the interfaces between the elements also the fluxes are not continuous. In order to make the scheme conservative a common numerical flux $\bar{F}_{lm}$ is imposed across the interface. In this work the convective part of the numerical flux is computed by solving a Riemann problem with the method proposed by Osher and Solomon [37] and implemented according to Pandolfi [38]. The diffusive part of the numerical flux is computed by means of a recovery based approach [26].
The volume and surface integrals in Eq. 12 are discretized by means of quadrature formulas with a number of points related to the degree of the reconstruction and the shape of the element.

3.2 Implicit time integration

In this work steady problems are considered. The solution is obtained by means of a time marching approach in which Eq. 12 is integrated until a steady solution is reached. In particular, a linearized implicit Euler scheme is adopted. Consider the global vector $\tilde{\boldsymbol{Q}} \in \mathbb{R}^{N_{DOF}}$ which contains all the N_{DOF} degrees of freedom of the problem. The time evolution of this vector is obtained by putting together all the elemental contributions from Eq. 12:

$$[\boldsymbol{M}]\frac{d\tilde{\boldsymbol{Q}}}{dt} = \tilde{\boldsymbol{R}}(\tilde{\boldsymbol{Q}}) \tag{13}$$

where $\tilde{\boldsymbol{R}} \in R^{N_{DOF}}$ contains all the elemental contributions from the right hand side of Eq. 12 and the block diagonal mass matrix $[\boldsymbol{M}]$ is the identity because the chosen basis is orthonormal. The linearized implicit Euler method allows to update the solution from the time step k to the time step $k+1$ in the following way:

$$[\boldsymbol{M}]\frac{\tilde{\boldsymbol{Q}}^{k+1} - \tilde{\boldsymbol{Q}}^{k}}{\Delta t} = \tilde{\boldsymbol{R}}(\tilde{\boldsymbol{Q}}^{k}) + \tilde{\boldsymbol{J}}(\tilde{\boldsymbol{Q}}^{k})(\tilde{\boldsymbol{Q}}^{k+1} - \tilde{\boldsymbol{Q}}^{k}) \tag{14}$$

where Δt is the time step size and $\tilde{\boldsymbol{J}} \in R^{N_{DOF}\times N_{DOF}}$ is the jacobian matrix. Here the jacobian matrix is computed numerically with the following approximation:

$$\tilde{J}_{ij} = \frac{\partial \tilde{R}_i}{\partial \tilde{Q}_j} = \frac{\tilde{R}_i(\tilde{\boldsymbol{Q}} + \epsilon \boldsymbol{e}_j) - \tilde{R}_i(\tilde{\boldsymbol{Q}})}{\epsilon} \tag{15}$$

where $\epsilon = 10^{-8}$ is the perturbation magnitude and $\boldsymbol{e}_j$ is the j-th versor in $\mathbb{R}^{N_{DOF}}$.
The resulting linear system is solved in parallel by means of the GMRES solver with ILU0 preconditioner provided by the PARALUTION library [34].

4 Proper Orthogonal Decomposition

The POD approach requires to build a database with N_s high-fidelity solutions (snapshots) obtained by changing the working conditions which are identified by a parameter z. The information in the database is compressed by the algorithm in order to identify the most significant modes. This will allows to describe the $l-th$ conservative variable q_l as a linear combination of $\hat{N}$ POD modes defined in the region Ω_{POD}:

$$q_l(\boldsymbol{x}, t) = \sum_{i=1}^{\hat{N}} \hat{a}_{li}(t)\psi_{li}(\boldsymbol{x}) \tag{16}$$

In the following, Ω_{POD} will be set equal to the full domain Ω (see Section 5) or to a single element Ω_e (see Section 6). The POD modes (ψ_{li}) can be expressed as a linear combination of the snapshots following the approach of Sirovich [47]

$$\psi_{li}(\boldsymbol{x}) = \sum_{j=1}^{N_s} b_{lj} q_l(\boldsymbol{x}, z_j) \tag{17}$$

where z_j is the value of the design parameter for the $j-th$ working condition in the database.

Let define the discrete scalar product operator $\langle \cdot, \cdot \rangle_{\Omega_{POD}}$:

$$\langle f(\mathbf{x}), g(\mathbf{x}) \rangle_{\Omega_{POD}} = \int_{\Omega_{POD}} f(\mathbf{x}) \cdot g(\mathbf{x}) d\Omega \tag{18}$$

where the integral is approximated by means of the same quadrature formula used for the volume integral in Eq.12. Let $|| \cdot ||_{\Omega_{POD}}$ be the norm related to this scalar product.

The coefficients b_{lj} are computed by solving a maximization problem. In particular, the coefficients b_{lj} are found such that the projection of the snapshots on the modes is maximized:

$$\max_{b_{lj}} \left[\sum_{k=1}^{N_s} \langle q_{lk}(\mathbf{x}), \psi_i(\mathbf{x}) \rangle^2_{\Omega_{POD}} \right] \quad \text{subject to} \, ||\psi_{li}(\mathbf{x})||_{\Omega_{POD}} = 1 \qquad 1 \leq i \leq N_s, 1 \leq l \leq n. \tag{19}$$

An eigenvalue problem is obtained by substituting Eq. 17 in Eq. 19. The coefficients b_{lj} which appear in Eq. 17 are obtained from the eigenvectors of the system. Note that a different POD basis must be computed for each conservative variable. This is in contrast with the DG basis defined in Section 3.1 which is used for all the conservative variables.

The contribution given by each mode to the reconstruction of the snapshots can be quantified by looking to the eigenvalues (λ_{li}) associated to the modes. The sum of all the eigenvalues represents the Total Information Content of the database. In order to limit the size of the POD basis it is possible to cut the POD expansion by considering the Relative Information Content (RIC) indicator:

$$RIC(\hat{N}_l) = \frac{\sum_{i=1}^{\hat{N}_l} \lambda_{li}}{\sum_{i=1}^{N_s} \lambda_{li}} \qquad 1 \leq \hat{N}_l \leq N_s, \qquad 1 \leq l \leq n \tag{20}$$

where N_s is the number of snapshots and hence the maximum possible size of the POD basis.

5 Two levels global method (TLG)

In this Section the two levels global (TLG) method is illustrated. Consider the POD algorithm described in Section 4 and define the POD modes as global modes on the full computational domain, i.e. $\Omega_{POD} = \Omega$.
The modal basis used in the chosen DG scheme is hierarchical and orthonormal: this makes very easy to project a low-order solution on an higher order basis. For this reason, the steady solution is usually found by performing a sequence of simulations in which the reconstruction order (and the cost) is increased from first order to the required accuracy level. In particular, this sequence of simulations can be performed on the same mesh: the increased accuracy is related to the increased number of degrees of freedom which are introduced in each element during the p-refinement procedure.

In this framework, it is natural to keep in the database both the low-order (p_L) and the high-order (p_H) simulations related to each sampling point. This makes it possible to build two different global POD bases related to the different discretization levels (ψ^L_{li} and ψ^H_{li}). More details on the choice of p_L and p_H will be given in Section 5.2. This completes the offline stage of the procedure.

5.1 Low order projection and high-order reconstruction

Suppose now to need a prediction for a new configuration not included in the database. The high fidelity DG solver can be called with the low reconstruction order (p_L) to get the low order solution q^L_l: this makes it possible to perform a very fast prediction of this new configuration because of the low number

of degrees of freedom in each element. The obtained solution can then be projected on the low-order POD basis in order to get the POD coefficients:

$$\hat{a}_{li}^L = \int_\Omega q_l^L(\boldsymbol{x})\psi_{li}^L(\boldsymbol{x})d\Omega = \sum_{e=1}^{N_T}\int_{\Omega_e} q_l^L(\boldsymbol{x})\psi_{li}^L(\boldsymbol{x})d\Omega \qquad 1 \leq i \leq \hat{N}_l, \quad 1 \leq l \leq n \tag{21}$$

where N_T is the number of elements in the mesh and the integral on each element is approximated by the same quadrature formula used for the volume integral in Eq. 12.
The POD coefficients obtained from the low-order POD basis are then used for the reconstruction of the predicted high-order solution ($q_l^{H'}$) by using the high-order POD basis:

$$q_l^{H'}(\boldsymbol{x}, z) = \sum_{i=1}^{\hat{N}_l} \hat{a}_{li}^L(z)\psi_{li}^H(\boldsymbol{x}) \qquad 1 \leq l \leq n \tag{22}$$

This approach is based on the assumption that the low and the high accuracy discretizations give POD bases such that it is possible to identify a correlation between each mode of the low-order basis and a corresponding mode in the high-order basis: the validity of this assumption will be discussed in the following Section 5.2.

The TLG method described here must not be confused with the two levels method proposed in [51] in which only the nonlinear terms of a POD-Galerkin model are computed on a coarse mesh. Furthermore, there are some connections between the TLG method and the ideas behind multiresolution methods used in optimization problems [39]. Multiresolution optimizations allow to reduce the cost of an optimization procedure by using low-fidelity models to accelerate the search but with occasional and systematic use of high-fidelity models to monitor the optimization progress [1]. An example of multifidelity approach is given by the Efficient Global Optimization (EGO) method in which a low-fidelity surrogate model is built by performing Kriging interpolation of the goal function values obtained by an high-fidelity model [32]: in the TLG method the role of the Kriging interpolation is played by the projection-reconstruction procedure described by Eqs. 21 and 22. Furthermore, while the EGO method works directly on the goal function, the TLG method allows to reconstruct the flow fields.

5.2 Choice of the resolution levels p_L and p_H

The choice of the resolution levels p_L and p_H is a key point of the TLG method. The choice of p_H can be based on a preliminary convergence study. In particular it is useful to start the offline phase by performing a set of first order ($p = 0$) simulations on very coarse meshes. Then the reconstruction order has to be increased until the solution reaches convergence and the goal function does not change any more. This fixes the required p_H level. This approach is particularly efficient when smooth problems are considered because in this case the DG method gives exponential convergence when p is increased.
The choice of p_L is less evident. In particular, it is necessary to guarantee a sufficient accuracy level in the prediction performed during the on-line phase such that the projection coefficients are correctly placed in the space spanned by the p_H basis. This condition can be studied systematically during the offline phase by adopting an iterative algorithm. Consider a database of solutions available at different accuracy levels ($0 \leq p \leq p_H$) which are used to build several corresponding POD bases at different accuracy levels. The analysis can be initialized by setting $p_L = 0$. For each point in the parameter space, it is possible to project the p_L solution to the p_L basis using Eq.21: this projection is exact since the projected solution is one of the snapshots used to generate the p_L basis. However, when the projection coefficients are used to predict the $q_l^{H'}$ solution by Eq.22 an approximation is introduced: the larger the distance between p_L and p_H, the larger the approximation. In particular, if the p_L solution is too inaccurate to properly describe some structures in the flow field which represent peculiar differences between the p_H snapshots then the projection coefficients obtained by Eq.21 will not be able to correctly identify the position of the predicted solution in the space of the p_H basis. In order to quantify this error it is possible to measure the relative distance $E_{HH'}$ between the high-order solution q_l^H in the database

and the predicted solution $q_l^{H'}$ obtained by using Eqs. 21 and 22:

$$E_{HH'} = \max_{1 \leq l \leq n} \left[\frac{\left|\left| q_l^H - q_l^{H'} \right|\right|_\Omega}{\left|\left| q_l^H \right|\right|_\Omega} \right] \tag{23}$$

If this distance is larger than a certain threshold σ then the p_L level must be increased and the test repeated. The algorithm is schematically reproduced by the following pseudo-code:

Algorithm 1 Choice of p_L

1: Set $p_L = -1$
2: **do**
3: Set $p_L = p_L + 1$
4: Low order projection: $\hat{a}_{li}^L = \int_\Omega q_l^L(\boldsymbol{x}) \psi_{li}^L(\boldsymbol{x}) d\Omega \qquad 1 \leq i \leq \hat{N}_l, \quad 1 \leq l \leq n$
5: High order reconstruction: $q_l^{H'} = \sum_{i=1}^{\hat{N}_l} \hat{a}_{li}^L \psi_{li}^H \qquad 1 \leq l \leq n$
6: Distance evaluation: $E_{HH'}$
7: **while** $E_{HH'} > \sigma$

The algorithm has to be applied during the offline phase to all the points of the database (or to a representative subset) and the most restrictive (the largest) value of p_L should be chosen among all the governing equations. The error indicator introduced in Eq. 23 could be substituted by an error on a goal function.

6 Local POD-DG method (LPOD-DG)

The global approach is based on the use of POD modes which are defined on the full computational domain Ω. This means that the predicted solution will be described as a linear combination of the solutions in the database. However, the governing equations contain advection terms which lead to non-linear phenomena. In order to understand this effect it is sufficient to think to a database of two solutions which are characterized by different positions of a particular structure in the fluid. If a linear combination of these snapshots is performed then it will not be possible to get a solution with the structure in an intermediate positions: the linear interpolation will give the superposition of the structures in their original positions.
In order to deal with this problem a different approach is proposed. Consider the POD algorithm described in Section 4 and apply it on each element of the mesh: this will give a POD basis for each element (and for each conservative variable). Since the POD basis is now defined locally there is the possibility to predict better non linear phenomena related to the advection of fluid structures in space.

6.1 Substituing the DG basis with the POD bases

In the TLG approach the POD bases are just used to compress the information of the database and to perform projections and reconstructions. In the local approach the POD bases can be directly exploited in the framework of the DG discretization: the general purpose DG modal basis used in Eq. 10 is substituted by an empirical local POD basis. The solution will still be obtained by the integration of the discretized governing equations (Eq. 12) but, since the POD is able to compress the information, there is the possibility to reduce the number of degrees of freedom required by each element with respect to the original DG discretization. This is particularly true when high-order DG methods are considered since the number of degrees of freedom inside each element grows quickly with the order (quadratically in 2D) and so there are large margins of reduction for high-order schemes. The price to pay is related to the fact that the original modal DG basis was the same for all the conservative variables while in the local approach it is necessary to introduce a different basis for each conservative variable. The size of the local POD basis can be decided by setting a threshold for the *RIC* indicator defined in Eq.20. This will lead to a locally changing size of the POD basis in the same spirit of p-adaptive techniques usually adopted for DG methods [3, 17, 25, 26, 29, 50]. In order to maximize the cost reduction introduced by the local

approach it is possible to compute the element POD basis by selecting a reduced set of snapshots from the database: in particular, only the first sampling points in the parameter space closer to the prediction point should be considered as snapshots. This allows to reduce the size of the element POD basis by keeping only the information relevant to the chosen prediction point. In this way the method will be local not only in the physical space but also in the parameter space. Several studies on the benefits related to the use of the locality concept in the parameter space are available in the literature [5, 7, 23, 55].

6.2 A less intrusive implementation

The proposed LPOD-DG method is intrusive because requires to modify the solver. However, it is possible to make some choices which minimize the modifications which have to be introduced. In particular, steady problems are considered in this work and so implicit time integration schemes are used. A simple profiling analysis shows that most of the computational time is related to the solution of the linear system in Eq. 14 which is required at each time step. For this reason, the goal will be to reduce the size of this system without changing the structure of the solver: this can be done by keeping both the original DG basis and the POD local basis descriptions.

The vector $\hat{\boldsymbol{A}} \in \mathbb{R}^{N_{DOF}^{POD}}$ contains all the POD degrees of freedom $\hat{\boldsymbol{a}}$ in the entire mesh and is significantly smaller than the corresponding vector $\tilde{\boldsymbol{Q}} \in \mathbb{R}^{N_{DOF}}$ of the original DG discretization. In particular the size of the problem is now given by

$$N_{DOF}^{POD} = \sum_{e=1}^{N_T} \sum_{l=1}^{n} \hat{N}_{le} \tag{24}$$

where $\hat{N}_{le}$ represents the number of degrees of freedom related to the l-th conservative variable in the element e and N_T is the number of elements. The linear system in Eq. 14 is substituted by the following one:

$$[\boldsymbol{M}] \frac{\hat{\boldsymbol{A}}^{n+1} - \hat{\boldsymbol{A}}^{n}}{\Delta t} = \hat{\boldsymbol{R}}(\hat{\boldsymbol{A}}^{n}) + \hat{\boldsymbol{J}}(\hat{\boldsymbol{A}}^{n})(\hat{\boldsymbol{A}}^{n+1} - \hat{\boldsymbol{A}}^{n}) \tag{25}$$

where the global mass matrix $[\boldsymbol{M}]$ is still the identity because the POD modes are orthonormal by definition and the jacobian matrix $\hat{\boldsymbol{J}} \in R^{N_{DOF}^{POD} \times N_{DOF}^{POD}}$ is evaluated numerically:

$$\hat{J}_{ij} = \frac{\partial \hat{R}_i}{\partial \hat{A}_j} = \frac{\hat{R}_i(\hat{\boldsymbol{A}} + \epsilon \boldsymbol{e}_j) - \hat{R}_i(\hat{\boldsymbol{A}})}{\epsilon} \tag{26}$$

where $\hat{R}_i$ is the time derivative of the i-th degrees of freedom in the POD representation.

Let consider the following element-wise projection operators from the DG basis to the POD basis and vice versa:

$$\hat{a}_{li} = \mathcal{P}_i(q_l^{DG}) = \int_{\Omega_e} q_l \psi_{li} d\Omega = \int_{\Omega_e} \sum_{j=1}^{N_e} (\tilde{q}_{lj}\phi_j)\psi_{li} d\Omega \qquad 1 \leq i \leq \hat{N}_{le}, \quad 1 \leq l \leq n \tag{27}$$

$$\tilde{q}_{li} = \mathcal{S}_i(q_l^{POD}) = \int_{\Omega_e} q_l \phi_i d\Omega = \int_{\Omega_e} \sum_{j=1}^{\hat{N}_{le}} (\hat{a}_{lj}\psi_{lj})\phi_i d\Omega \qquad 1 \leq i \leq N_e, \quad 1 \leq l \leq n \tag{28}$$

These operators contain only space integrals and so they can be applied to both the solution (q_l) and its time derivative ($\partial q_l / \partial t$). In particular the time derivative of the POD degrees of freedom ($\hat{R}_i$) which appears in Eq.25 can be computed by projecting the time derivative of the DG degrees of freedom. Consider the i-th POD degrees of freedom which is related to the element e and the conservative variable l. Its time derivative can be computed as:

$$\hat{R}_i = \mathcal{P}_i \left(\frac{\partial q_l^{DG}}{\partial t} \right) = \mathcal{P}_i \left(\sum_{j=j_{le}^1}^{j_{le}^2} \tilde{R}_j \phi_j \right) \tag{29}$$

where j^1_{le} and j^2_{le} identify the range of the degrees of freedom of the DG basis in the element e for the conservative variable l.

In this way most of the solver remains unchanged: both the DG and the POD basis are defined in each element. At each time step the POD solution is projected on the DG basis and the solution on the DG basis is used to compute the time derivative of the DG degrees of freedom by exploiting the original structure of the solver. Then the time derivative of the DG degrees of freedom are projected with Eq. 29 in order to get the time derivative of the POD degrees of freedom which are required by Eq. 25. All the spatial integrals which are computed in the process are approximated by the same quadrature formula required by the original DG basis.

6.3 An approach to judge the quality of a LPOD-DG solution

When a classical DG simulation of steady problems is performed the computation is stopped when the time derivative of the DG degrees of freedom reaches a sufficiently low value such that the goal function does not change anymore. This means that the engineer knows from experience the admissible value of the time derivative at which the simulation can be stopped. This know-how can be exploited in the framework of the LPOD-DG method to judge whether the obtained solution is sufficiently close to the steady solution of a full DG simulation.
The availability of the time derivatives for both the DG and POD degrees of freedom can be exploited to get an estimation of the distance from the solutions obtained by the LPOD-DG method and the full DG method. In particular, the LPOD-DG simulation is stopped when the time derivative of the POD degrees of freedom which appears in Eq. 25 reaches a sufficiently low value (10^{-10} in this work). However, the time derivative of the DG degrees of freedom does not fall to machine precision but goes asymptotically to a finite value. The magnitude of this finite value can be considered as a measure of the distance from the DG basis and the POD basis: if the POD basis would contain the same information of the full DG basis then both time derivatives would reach machine precision.
The norm-2 of the DG time derivatives ($||r||$) can be computed as:

$$||r|| = \sqrt{\frac{\int_\Omega \sum_{l=1}^{n} \sum_{i=1}^{N_e} (\frac{\partial \tilde{q}_{li}}{\partial t})^2 d\Omega}{n N_e \int_\Omega d\Omega}} \tag{30}$$

and can be used to judge the distance from the LPOD-DG solution and a corresponding full DG solution. If the prediction obtained by the LPOD-DG method gives a value of $||r||$ too high there are too possible remedies: the database could be improved or the LPOD-DG solution can be used as an initial condition for a full DG simulation which will be quite cheap since it would start from a relatively good initial condition.

7 Perturbed inviscid flow around a turbine airfoil

In this Section the global and local approaches are tested and compared on the study of the inviscid compressible flow around the MTU T106 gas turbine airfoil. This airfoil is representative of the geometry of low pressure turbines in modern aeronautical turbofans and can show large separations in certain working conditions [27, 30]. In this work the inviscid flow around this geometry is studied and a perturbation in the inlet total pressure is introduced. In particular the undisturbed inlet total pressure ($P^0_i = 1$, as it is chosen as reference pressure P_{ref}) is reduced by a Gaussian perturbation $\Delta P^0(y)$:

$$\Delta P^0(y) = \Delta e^{-\frac{(y-y_w)^2}{\sigma^2}} \tag{31}$$

where Δ, σ and y_w are related to the intensity, the thickness and the position of the perturbation, respectively. The presence of the perturbation in the inlet total pressure distribution can represent the wake of a a fixed obstacle upstream of the profile. This configuration can be observed at the interface between high-pressure and low-pressure turbines in aeronautical turbofans. The inlet total temperature ($T^0 = 1$) and the inlet angle (α) are uniform. Since the flow is inviscid and subsonic there is no entropy

generation and so the entropy wake related to the perturbation is transported by the flow around the airfoil.
The static pressure is imposed at the subsonic outlet and periodic boundary conditions are imposed on the lateral boundaries. The computational domain and the mesh with 3063 elements are shown in Figure 1.

The problem is parametrized with two variables: the inlet flow angle ($31.7° \leq \alpha \leq 33.7°$) and the vertical position of the inlet total pressure perturbation ($0.03 \leq y_w \leq 0.04$). The range for α is centered on the design value of this airfoil which is 32.7° [30]. These parameters have been chosen because when they change the perturbation moves in space: this represents a typical advection phenomenon which is challenging for POD methods because of its non-linearity. The entropy and Mach number fields for two different configurations ($\alpha = 31.7°, y_w = 0.03$ and $\alpha = 33.7°, y_w = 0.04$) are reported in Figures 2 and 3, respectively.

A set of high fidelity solutions are computed by means of the DG solver by choosing a fifth order accurate scheme ($p = 4$). This requires 15 degrees of freedom per equation in each element (see Eq. 11). Since there are 4 governing equations and 3063 elements the total number of degrees of freedom is 183780. A Cartesian sampling of the parameter space is performed and three databases are considered with 2×2, 3×3 and 5×5 sampling points, respectively (see Figure 4). The uniform sampling is sufficient for the purpose of this study but there are several more efficient sampling techniques in the literature: adaptive sampling for POD-based surrogates [49], adaptive sampling based on a leave-one-out error indicator [14, 55], compact POD bases [19], greedy approaches based on Voronoi tessellation [12].

The goal function for this test case is the total pressure loss distribution in the vertical control section L located 0.5 axial chords behind the trailing edge. Figure 5 shows the total pressure loss distribution $(1 - P^0(y))$ in the section L for the high-fidelity simulations of the 2×2 database.

The performances of the global and local solvers will be tested by trying to predict the solution for 10 pseudo-random points in the parameter space generated by the Sobol sequence (see Figure 4). The quality of the predictive simulations is evaluated by considering the following error on the total pressure distribution loss $(1 - P^0(y))$ along the vertical control section L:

$$E_r = \sqrt{\frac{\int_L \left(P^0(y) - P^0_{exa}(y)\right)^2 dy}{\int_L (1 - P^0_{exa}(y)) dy}} \tag{32}$$

where P^0_{exa} is the total pressure value computed by means of a standard DG4 simulation.

Two levels global method The low accuracy level p_L required by the TLG method is chosen according to the strategy reported in Section 5.2. In particular, the iterative procedure is applied to the points of the 2×2 database which represent the limits of the parameter space. The results reported in Table 1 show that there is a large improvement when moving from $p_L = 0$ to $p_L = 1$ while a further increase to $p_L = 2$ or $p_L = 3$ gives smaller benefits. For this reason, p_L is set to 1.
The TLG method is tested on the 10 prediction points. The solutions in the database at $p_L = 1$ and at $p_H = 4$ are used to generate two global POD bases in which the modes are defined in the full domain Ω. The number of modes kept in the POD bases is chosen by setting $RIC = 99.9999\%$.
Each prediction is obtained by performing a second order accurate DG1 simulation and then projecting the solution on the p_L POD basis. The obtained POD coefficients are then used for the reconstruction of the solution with the p_H POD basis. The reduction in the computational cost with respect to a classical DG4 prediction is significant because the ratio between the number of degrees of freedom at the two accuracy levels is equal to 5.

Local POD-DG method The POD bases used by the local method for each prediction point are built by considering the snapshots from the first 4 closest sampling points in the parameter space. The threshold on the RIC is set to 99.9999%. This assures that almost all the information in the original snapshots is kept in the basis. Even with this high value of RIC there are some elements in which the

Table 1: Effect of p_L on $E_{HH'}$ (database 2x2)

y_w	$\alpha[^\circ]$	$E_{HH'}(p_L=0)$	$E_{HH'}(p_L=1)$	$E_{HH'}(p_L=2)$	$E_{HH'}(p_L=3)$	$E_{HH'}(p_L=4)$
0.03	31.7	0.224	0.127	0.099	0.071	0.
0.04	31.7	0.206	0.113	0.095	0.051	0.
0.03	33.7	0.221	0.120	0.095	0.049	0.
0.04	33.7	0.202	0.112	0.093	0.074	0.

local POD basis contains just 1 or 2 modes. In Figure 6 a map of the number of degrees of freedom per element is shown: the reduction with respect to the original DG4 discretization is evident. The regions of the domain in which the map has the value 4 require just 1 mode per element for all the conservative variables. In order to simplify the implementation the same number of degrees of freedom is used for all the conservative variables in each element. In particular, the size of the basis is chosen by taking the maximum number of degrees of freedom required by the imposition of the RIC threshold between all the four conservative variables.

Comparison of the methods The comparison between the TLG and the LPOD-DG methods is done by evaluating the prediction error and the cost of the predictions. The cost includes the generation of the database (which is the same for the two methods in this example) and the cost of the predictive simulation. The cost of the predictive simulation can be related to the number of degrees of freedom required by a DG1 simulation ($3063 \times 3 \times 4 = 36756$) for the TLG method and to the total number of POD degrees of freedom (N_{DOF}^{POD}) for the LPOD-DG method, respectively. The results are reported in Tables 2, 3 and 4 for the 2×2, 3×3 and 5×5 databases, respectively. The results show that the number of degrees of freedom required by a prediction with the TLG method is larger with respect to the LPOD-DG method. However, the error obtained by the LPOD-DG method is significantly smaller with respect to the TLG method. Furthermore, both methods allow a strong reduction of the cost with respect to the full DG4 simulation which requires 183780 degrees of freedom.
It is interesting to note that when the sampling of the database is improved, the LPOD-DG method can benefit of the better POD basis and gives a lower prediction error. This is true also for the TLG method but only when the database is increased from 2×2 to 3×3. A further improvement in the sampling (to the 5×5 database) does not reduce the prediction error: this means that now the error of the TLG method is fixed by the error of the DG1 predictive simulation. Any further improvement of the database is useless. It would be necessary to increase the order of the low level simulation in order to exploit the new information added in the database.
It is interesting to compare the error obtained by the two proposed methods with a classical DG discretization with approximately the same number of degrees of freedom. The total number of degrees of freedom for a classical DG discretization can be obtained by Eq. 11(36756 for DG1, 73512 for DG2, 122520 for DG3, 183780 for DG4) which shows that the DG1 discretization requires a number of degrees of freedom equal to the TLG method and comparable to the LPOD-G method. In Table 5 the prediction error obtained by a classical DG1 simulation is reported: it is significantly higher than the errors obtained by the TLG and LPOD-DG methods.
In Figure 7 the predicted total pressure loss distribution in the control section is reported for a reference DG4 simulation, for the TLG and LPOD-DG methods (with database 3×3) and for a standard DG1 discretization. The first configuration ($y_w = 0.035599$, $\alpha = 32.1016^\circ$) of Table 3 is considered. The Figure shows clearly the better agreement obtained by the local method. The global method shows negative losses in the region out of the wake where the losses should be zero: this could be related to the fact that the projections used by the global method are done on the conservative variables and not on the total pressure. The standard DG1 prediction which has the same number of degrees of freedom of the TLG method gives a significantly higher error. This is due to the fact that the POD-based methods work with POD modes obtained from DG4 snapshots: when the order of reconstruction is increased from DG1 to DG4 with the same mesh there is an exponential convergence to the exact solution and this explain why the standard DG1 simulation gives poor results with respect to the POD-based methods.

Table 2: Comparison of global and local methods in predictive simulations (database 2x2)

y_w	$\alpha[^\circ]$	N_{DOF} (TLG)	E_r (TLG)	N_{DOF} (LPOD-DG)	E_r (LPOD-DG)
0.03559	32.10	36756	0.367	25860	0.183
0.03078	33.39	36756	0.327	25860	0.191
0.03979	33.39	36756	0.375	25860	0.064
0.03382	32.50	36756	0.292	25860	0.252
0.03712	33.16	36756	0.311	25860	0.187
0.03227	31.96	36756	0.463	25860	0.122
0.03770	32.94	36756	0.277	25860	0.229
0.03273	33.61	36756	0.394	25860	0.111
0.03606	32.33	36756	0.304	25860	0.231
0.03012	32.12	36756	0.443	25860	0.156

Table 3: Comparison of global and local methods in predictive simulations (database 3x3)

y_w	$\alpha[^\circ]$	N_{DOF} (TLG)	E_r (TLG)	N_{DOF} (LPOD-DG)	E_r (LPOD-DG)
0.03559	32.10	36756	0.274	22788	0.0861
0.03078	33.39	36756	0.290	22416	0.0849
0.03979	33.39	36756	0.275	21820	0.0490
0.03382	32.50	36756	0.269	22812	0.0471
0.03712	33.16	36756	0.269	21820	0.0825
0.03227	31.96	36756	0.307	22812	0.109
0.03770	32.94	36756	0.266	21820	0.0696
0.03273	33.61	36756	0.298	22416	0.0633
0.03606	32.33	36756	0.263	22788	0.145
0.03012	32.12	36756	0.310	22812	0.106

In Table 6 the norm-2 of the time derivative of the DG degrees of freedom ($||r||$ defined in Eq.30) is reported for the predictive simulations with the LPOD-DG method. The Table shows that, in average, the value of $||r||$ tends to decrease when the sampling of the parameter space is improved.

8 Turbulent flow around a wind turbine airfoil

The TLG and the LPOD-DG methods are evaluated by considering the prediction of the flow around a wind turbine airfoil. In particular the DU91-W2-250 profile is considered. The geometry and some experimental results are available in [48]. The computational domain and the mesh are shown in Figure 10. The computational domain is a circle and the far field boundary is at 10 chords from the airfoil. The mesh, generated by the high-order meshing tool Gmsh [28], is composed by a structured region close to the wall surrounded by an unstructured triangular grid. It contains 7416 elements. Details of the mesh close to the airfoil and in the trailing edge region are reported in Figures 11 and 12.

The experimental data shows that the stall for this airfoil is reached at approximately 10° of incidence when the Reynolds number is 3 millions. Here we study the flow field for different values of Reynolds number (Re) and angle of attack (α). The following ranges are chosen: $0.5 \cdot 10^6 \leq Re \leq 2 \cdot 10^6$ and $3^\circ \leq \alpha \leq 12^\circ$. The far field Mach number and turbulent intensity are set to $M_\infty = 0.2$ and $\hat{\nu}/\nu = 1$, respectively.

A database of four high-fidelity simulations ($p = 3$) corresponding to the limits of the parameter space is built. The database points are reported in Figure 13. All the database solutions are obtained as a sequence of steady simulations from $p = 0$ to $p = 3$. Since the mesh is quite coarse the boundary layer and the separated shear layer are strongly unresolved for the lower reconstruction orders: this can lead to instability problems when large time steps are performed during the implicit time integration of

Table 4: Comparison of global and local methods in predictive simulations (database 5x5)

y_w	$\alpha[^\circ]$	N_{DOF} (TLG)	E_r (TLG)	N_{DOF} (LPOD-DG)	E_r (LPOD-DG)
0.03559	32.10	36756	0.281	20060	0.0369
0.03078	33.39	36756	0.289	19604	0.0403
0.03979	33.39	36756	0.275	19000	0.0232
0.03382	32.50	36756	0.270	20252	0.0815
0.03712	33.16	36756	0.259	19648	0.0989
0.03227	31.96	36756	0.307	19568	0.0492
0.03770	32.94	36756	0.251	19216	0.0440
0.03273	33.61	36756	0.296	19176	0.0254
0.03606	32.33	36756	0.266	20228	0.0440
0.03012	32.12	36756	0.302	19568	0.0456

Table 5: Standard DG1 predictions

y_w	$\alpha[^\circ]$	N_{DOF} (DG1)	E_r (DG1)
0.03559	32.10	36756	1.86
0.03078	33.39	36756	1.82
0.03979	33.39	36756	1.78
0.03382	32.50	36756	1.85
0.03712	33.16	36756	1.79
0.03227	31.96	36756	1.89
0.03770	32.94	36756	1.80
0.03273	33.61	36756	1.80
0.03606	32.33	36756	1.84
0.03012	32.12	36756	1.89

Table 6: DG residuals in LPOD-DG prediction

y_w	$\alpha[^\circ]$	$\lvert R\rvert$ (db 2×2)	$\lvert R\rvert$ (db 3×3)	$\lvert R\rvert$ (db 5×5)
0.03559	32.10	2.97E-04	2.04E-04	6.78E-05
0.03078	33.39	2.52E-04	3.52E-04	6.39E-05
0.03979	33.39	3.21E-04	1.80E-04	5.01E-05
0.03382	32.50	3.25E-04	1.65E-04	4.46E-04
0.03712	33.16	2.83E-04	9.96E-05	1.77E-04
0.03227	31.96	2.94E-04	1.32E-04	7.21E-05
0.03770	32.94	3.10E-04	1.13E-04	6.86E-05
0.03273	33.61	2.31E-04	1.01E-03	7.27E-05
0.03606	32.33	3.13E-04	3.54E-04	9.22E-05
0.03012	32.12	3.16E-04	1.54E-04	7.24E-05

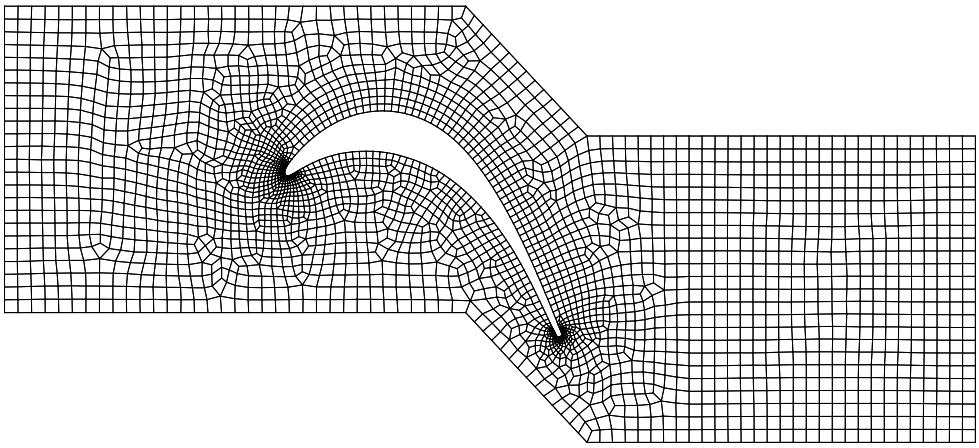

Figure 1: Mesh with 3063 elements

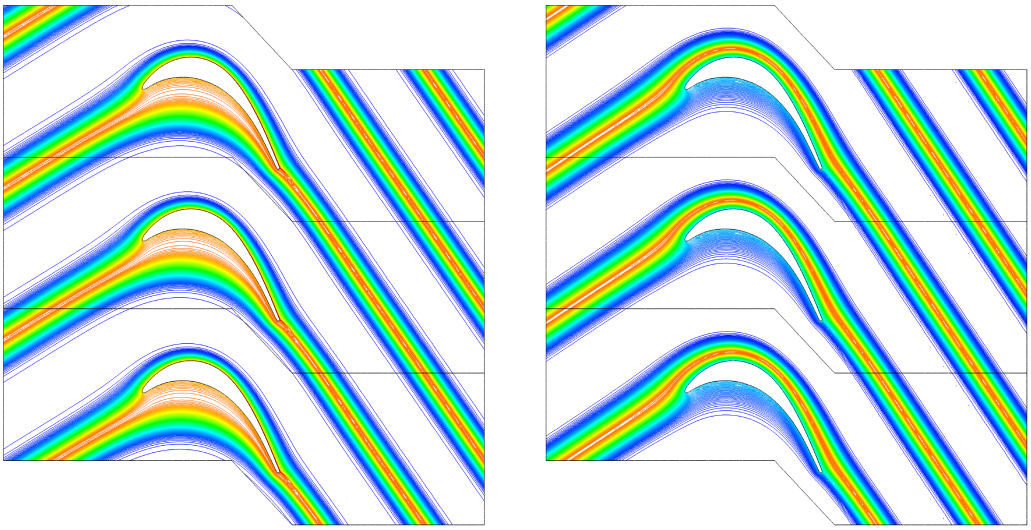

Figure 2: Entropy field for $\alpha = 31.7^o$ and $y_w = 0.03$ (left) and for $\alpha = 33.7^o$ and $y_w = 0.04$ (right)

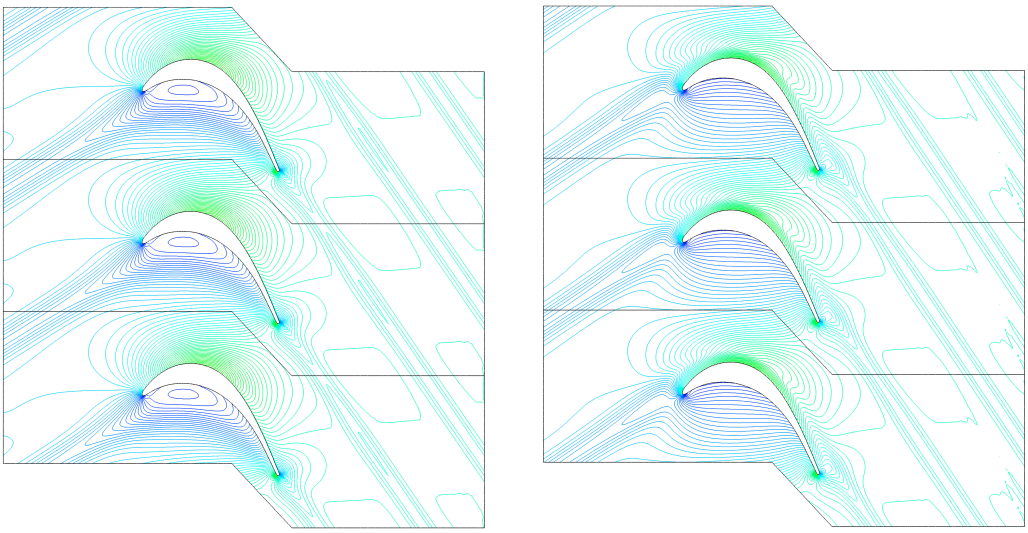

Figure 3: Mach field for $\alpha = 31.7^o$ and $y_w = 0.03$ (left) and for $\alpha = 33.7^o$ and $y_w = 0.04$ (right)

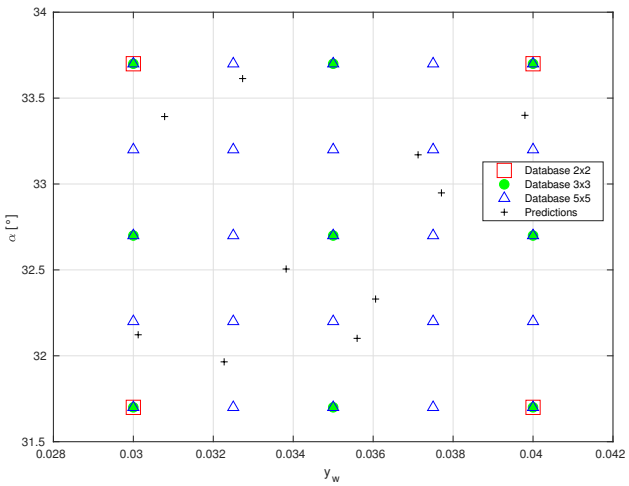

Figure 4: Sampling points and test points in the parameter space

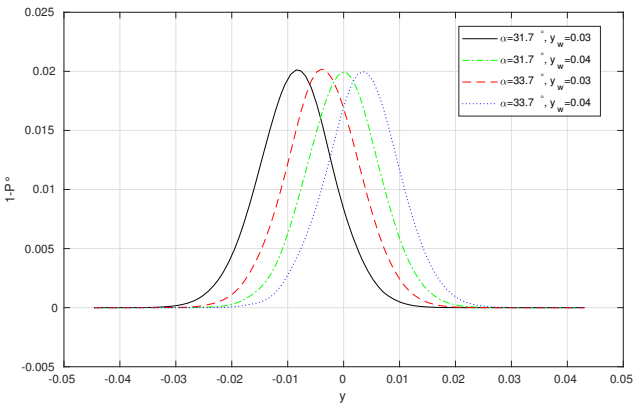

Figure 5: Total pressure loss distribution for the sampling points in the database 2x2

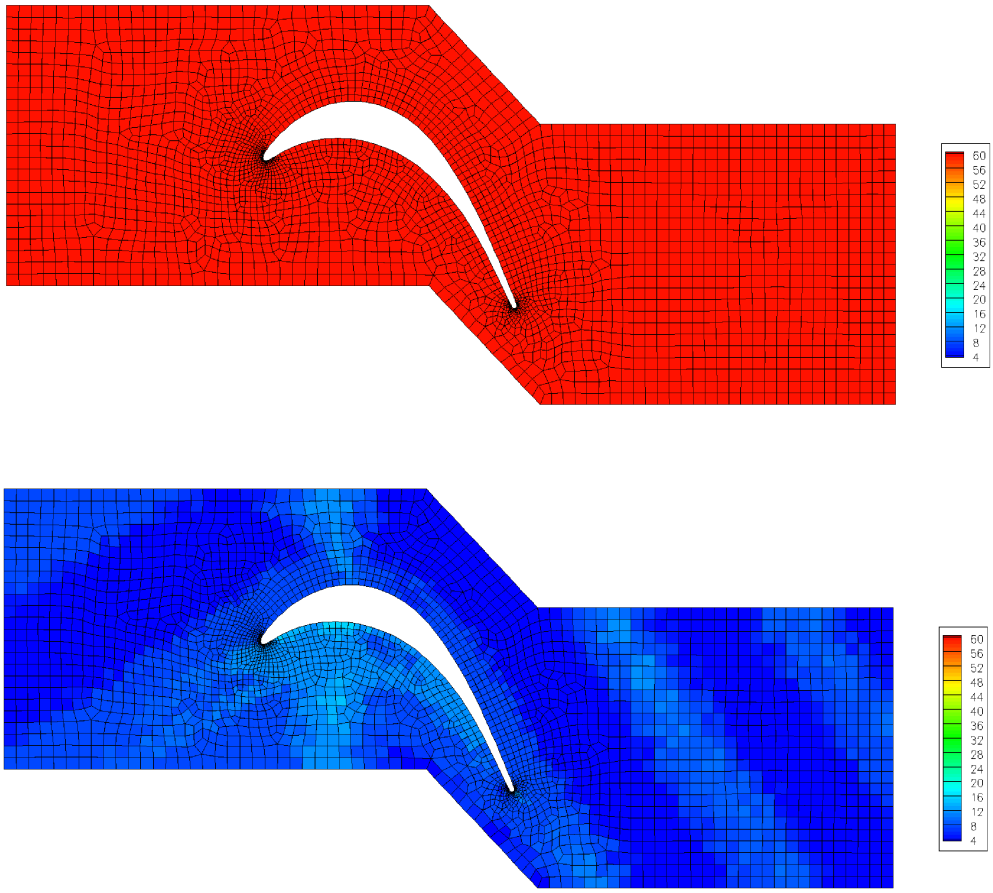

Figure 6: Distribution of the number of degrees of freedom per element: original DG4 (top) and local POD approach (bottom)

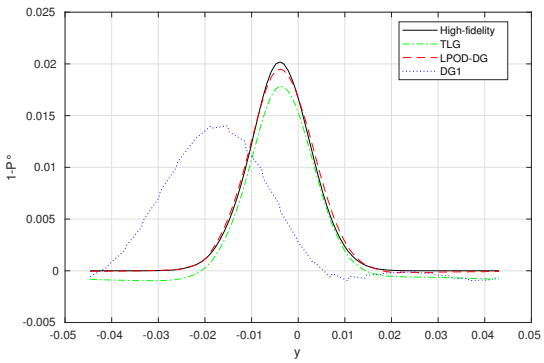

Figure 7: Prediction of the total pressure loss distribution with TLG and LPOD-DG methods

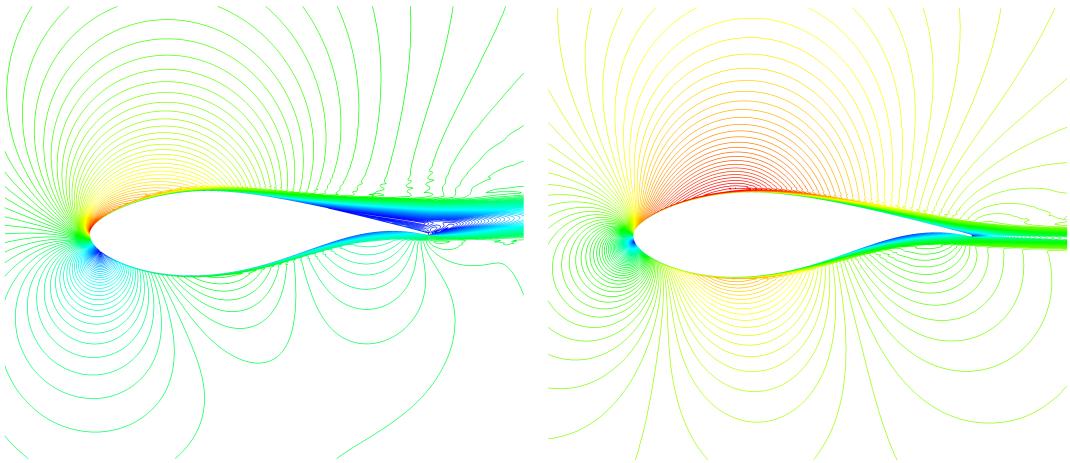

Figure 8: Mach field at $\alpha = 12°$ and $Re = 0.5 \cdot 10^6$ (left) and at $\alpha = 3°$ and $Re = 2 \cdot 10^6$ (right)

the equations. In order to make the integration more robust, a filtering approach is adopted during the initial transient following the approach proposed in [24]. This filtering stabilization is then deactivated when the solution is close to the steady state.

The results of the high-fidelity simulations at $Re = 0.5 \cdot 10^6$ / $\alpha = 12°$ (left) and $Re = 2 \cdot 10^6$ / $\alpha = 3°$ (right) are reported in Figures 8 and 9 which show the Mach field and the modified eddy viscosity ($\rho\hat{\nu}$) field, respectively. The field at the higher angle of attack is characterized by a large separation while the flow is attached in the other configuration. This means that the range explored in this study is sufficiently large to include both pre-stall and near-stall working conditions. The wall pressure coefficient (c_p) distribution is reported in Figure 14 for the database points: it is possible to identify a plateau on the suction side close to the trailing edge at $Re = 0.5 \cdot 10^6$ / $\alpha = 12°$ which corresponds to the separated region.

The TLG and the LPOD-DG methods are tested by predicting the solution in nine configurations reported in Figure 13. The RIC threshold is set to 99.9999% for both the TLG and LPOD-DG methods. The local bases used by the LPOD-DG method are obtained by the $p = 3$ high-fidelity simulations while the low and high-accuracy levels required by the TLG method are set to $p_L = 1$ and $p_H = 3$, according to the algorithm proposed in Section 5.2.
These choices lead to a total number of degrees of freedom equal to 101645 and 111240 for the LPOD-DG and the TLG methods, respectively. These values represent a significant reduction with respect to the number of degrees of freedom in the high-fidelity DG3 discretization (370800). Figure 15 shows the spatial distribution of degrees of freedom per element in the original DG and in the LPOD-DG discretisations.

The wall pressure coefficient (c_p) distribution predicted by the TLG and LPOD-DG methods are compared with the reference values obtained by the DG3 high-fidelity simulations and by standard DG1 simulations (which have the same online cost of TLG predictions and approximately the same cost of LPOD-DG predictions). These results are reported in Figures 16-24. The results of the LPOD-DG method are very close to the high-fidelity results and systematically better than those obtained by the TLG method. Furthermore, the cost of the LPOD-DG method is significantly lower than the cost of the high-fidelity simulations since the implicit integration of the LPOD-DG equations requires to solve a system which is approximately 3.7 times smaller than the system required by the original DG3 scheme. The results obtained by the TLG method show a prediction error which is comparable to the standard DG1 computations: in some predictions the TLG performs better than the DG1 and vice versa. However, the DG1 results are characterized by large oscillations which are typical of DG methods in the presence of strongly unresolved boundary layers. On the contrary, the TLG results do not show these oscillations because they are built on DG3 snapshots which have enough resolution in the boundary layer.

Figure 9: Turbulent variable ($\rho\hat{\nu}$) field at $\alpha = 12°$ and $Re = 0.5 \cdot 10^6$ (left) and at $\alpha = 3°$ and $Re = 2 \cdot 10^6$ (right)

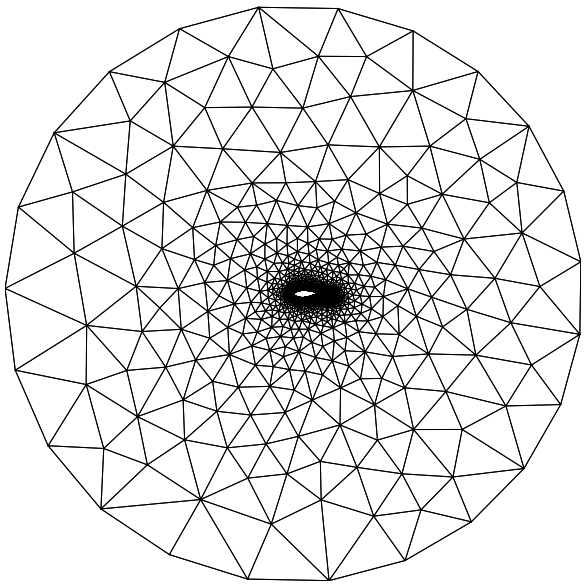

Figure 10: Computational mesh for wind turbine airfoil

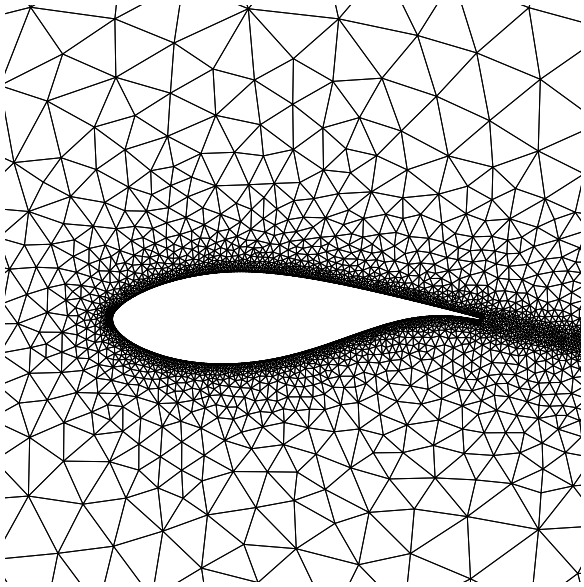

Figure 11: Computational mesh for wind turbine airfoil: region close to the wall

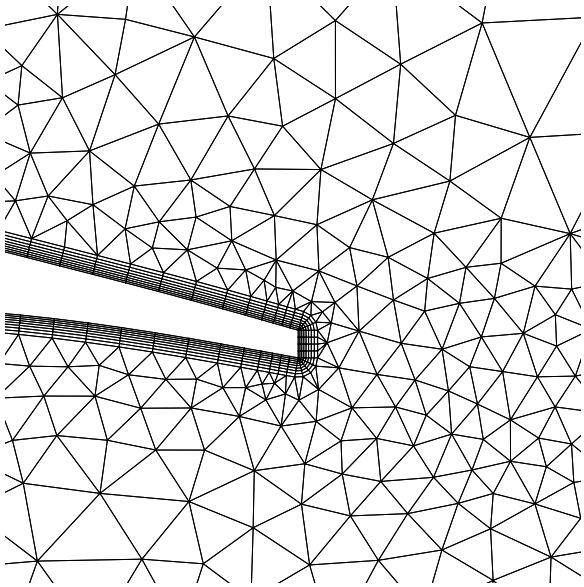

Figure 12: Computational mesh for wind turbine airfoil: detail of the trailing edge

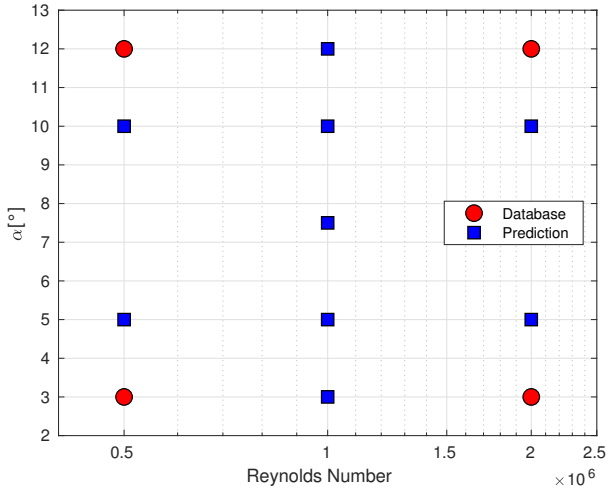

Figure 13: Database and predictions points in the parameter space

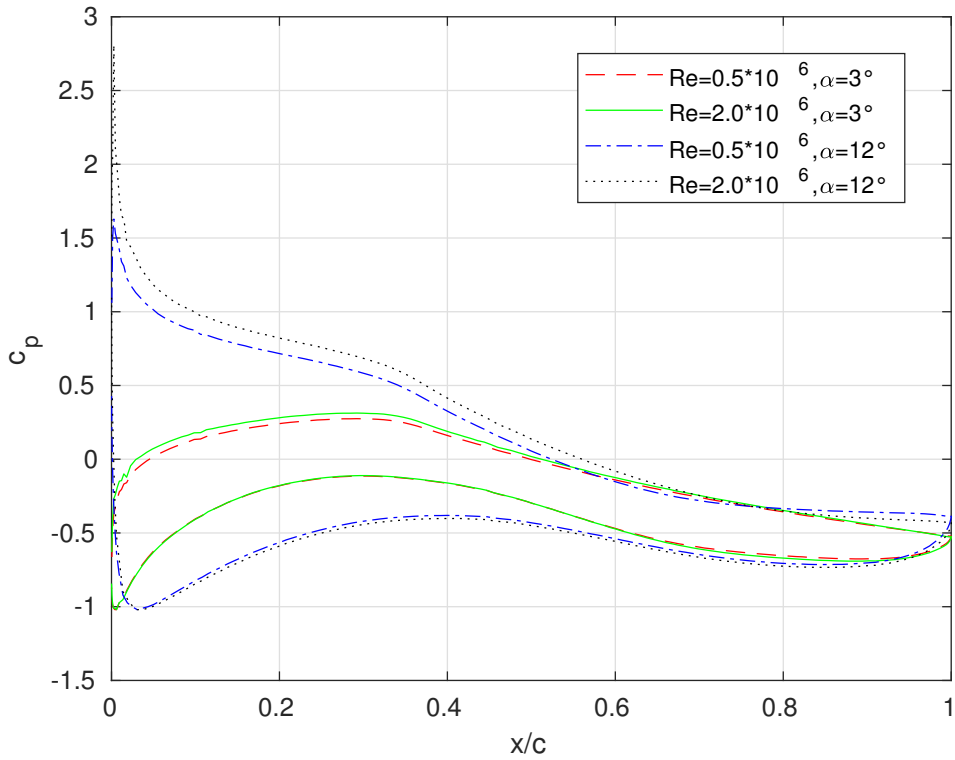

Figure 14: Wall pressuce coefficient distribution for database points

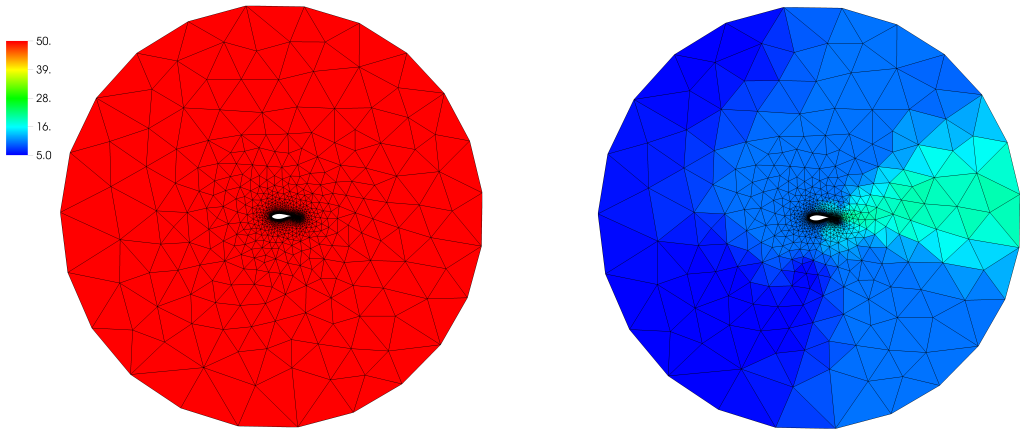

Figure 15: Number of degrees of freedom per element in the DG3 scheme (left) and LPOD-DG scheme (right)

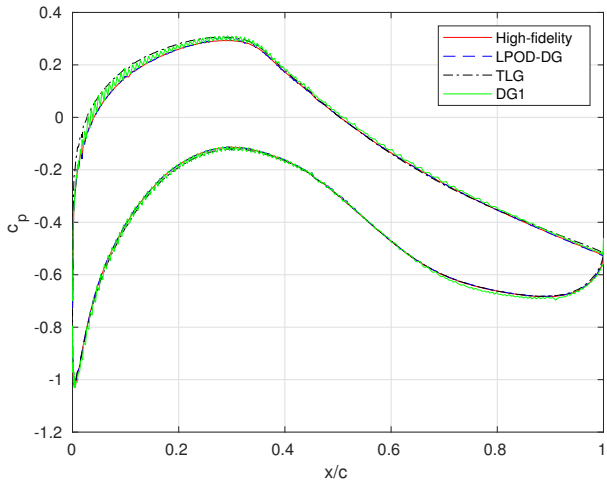

Figure 16: Predicted c_p distribution for $Re = 1.0 \cdot 10^6$ / $\alpha = 3°$

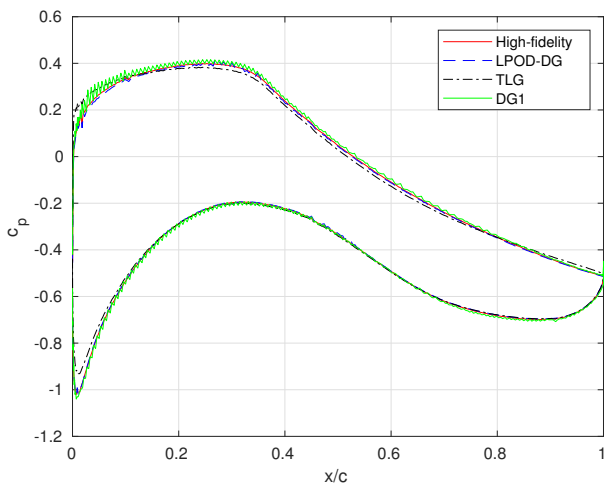

Figure 17: Predicted c_p distribution for $Re = 1.0 \cdot 10^6$ / $\alpha = 5°$

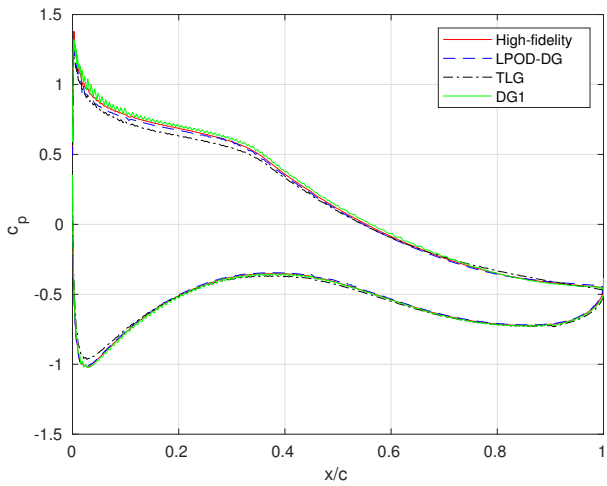

Figure 18: Predicted c_p distribution for $Re = 1.0 \cdot 10^6$ / $\alpha = 10°$

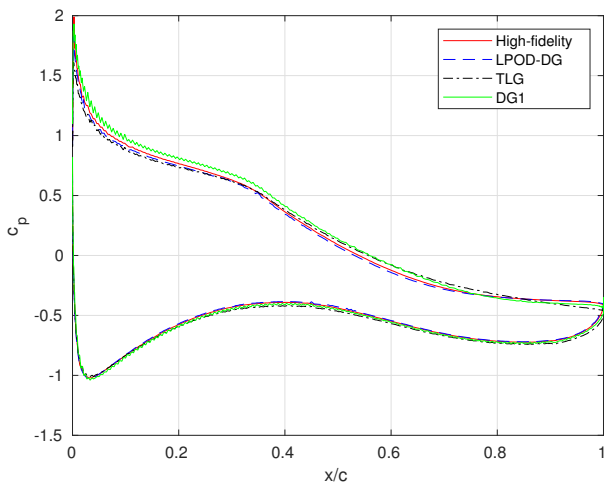

Figure 19: Predicted c_p distribution for $Re = 1.0 \cdot 10^6$ / $\alpha = 12°$

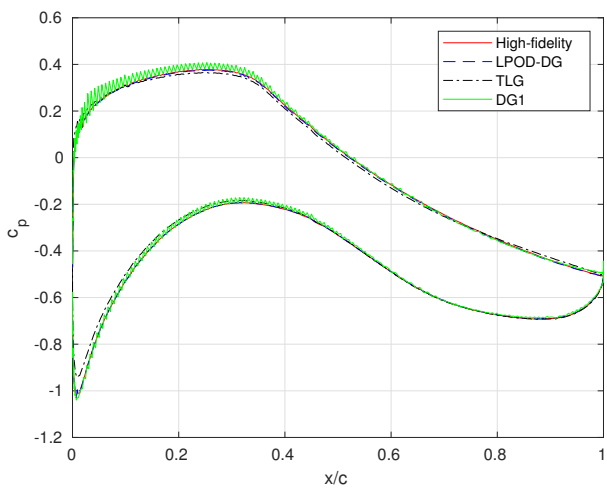

Figure 20: Predicted c_p distribution for $Re = 0.5 \cdot 10^6$ / $\alpha = 5°$

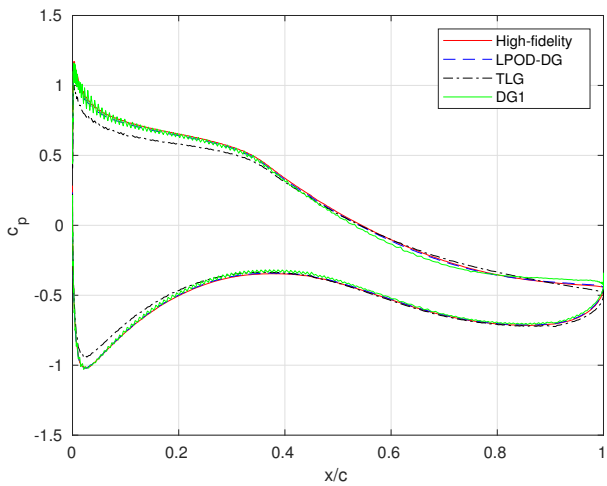

Figure 21: Predicted c_p distribution for $Re = 0.5 \cdot 10^6$ / $\alpha = 10°$

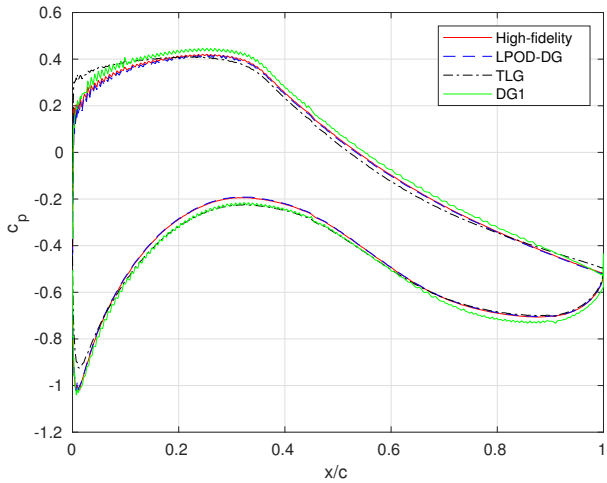

Figure 22: Predicted c_p distribution for $Re = 2.0 \cdot 10^6$ / $\alpha = 5°$
l

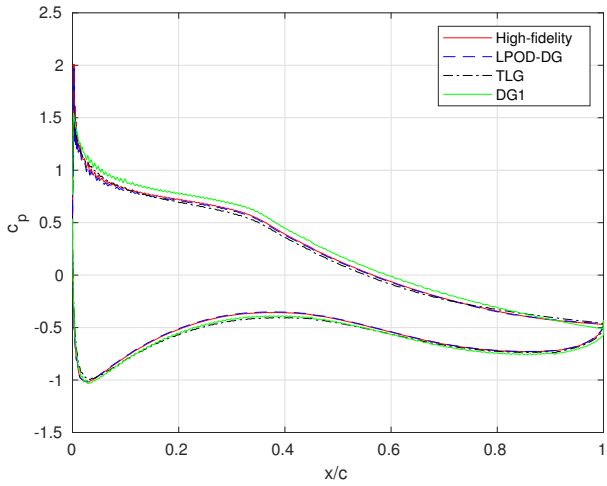

Figure 23: Predicted c_p distribution for $Re = 2.0 \cdot 10^6$ / $\alpha = 10°$

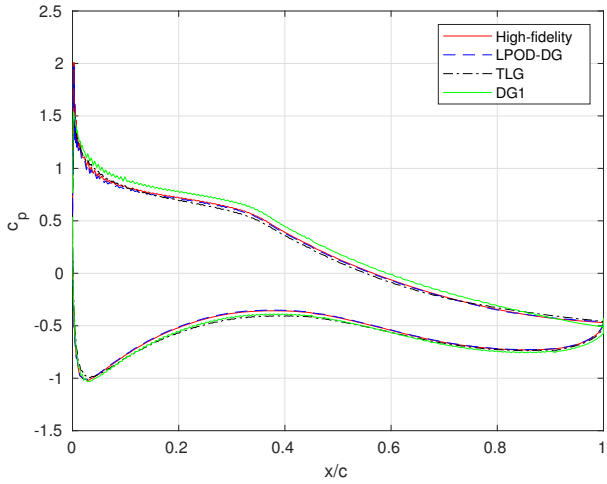

Figure 24: Predicted c_p distribution for $Re = 1.0 \cdot 10^6$ / $\alpha = 7.5°$

9 Conclusions

Two different ROM approaches are presented in the framework of a DG modal discretization.
The TLG method is based on global POD modes which are defined on the full computational domain. It requires to compute a database of solutions which are used to build low-order and high-order POD bases. This step can be easily accomplished in the framework of DG schemes because the accuracy of the discretization can be incremented by adding more degrees of freedom in the elements without changing the mesh. The TLG prediction is obtained by performing a low-order simulation, projecting the result on the low-order basis and using the projection coefficients for the reconstruction with the high-order basis. An offline algorithm for the choice of the reconstruction order required by the two levels is proposed.
The LPOD-DG method is based on the definition of a local POD basis in each element which can substitute the original DG basis. The ability of the POD to compress the information allows to reduced the number of degrees of freedom inside each element with respect to the original DG scheme. The price to pay for this reduction is an increase in the memory requirements related to the fact that a different POD basis must be used for each conservative variable. The results show that the LPOD-DG method allows to significantly reduce the simulation cost with respect to a full order DG solution.
A direct comparison between the two methods shows that the LPOD-DG method gives better predictions at lower cost with respect to the TLG method when the same sampling of the parameter space is used. Furthermore, the two methods behave differently when the sampling of the database is improved: on one hand, the LPOD-DG method benefits from the new information which is added in the POD bases. On the other hand, the TLG method shows some improvements when the sampling of the database is increased until a limit error is reached. This limit of the TLG method seems to be related to the discretization error imposed by the low-order prediction. While the LPOD-DG method gives a prediction error always lower than a comparable (in terms of degrees of freedom) DG1 simulation on the same mesh, the TLG method shows smaller benefits, depending on the particular test case considered.
A particular implementation of the LPOD-DG method is suggested in order to make the approach less intrusive. In particular, both the original DG basis and the POD basis can be kept in each element: in this way the implementation reduces to the definition of projection operators but still allows to reduce the size of the linear system required by the implicit time integration which represents the bottleneck of the discretization. Furthermore, the availability of both bases allows to estimate the quality of a predictive LPOD-DG steady simulation: while a steady solution is reached in the POD space, the time derivatives of the DG degrees of freedom goes asymptotically to a finite value which gives indications on the distance between the POD basis and the full DG basis. This error indicator could be exploited in future works to locally adapt the solution by substituting the POD basis with the original high-fidelity DG basis in those elements where the POD basis is not able to properly predict the solution.
The potential reduction in the number of degrees of freedom introduced by the LPOD-DG method increases for high-order DG schemes and decreases for high-dimensional parameter spaces. So the method

is efficient for smooth problems in which high-order schemes can be efficiently employed on coarse meshes. The presence of shock waves, which limit the regularity of the solution, suggests the use of low-order schemes with fine meshes and so the LPOD-DG method does not seem to be convenient for this kind of problems because the number of POD modes in each element would become equivalent to the size of the original DG basis. However, it would be interesting to evaluate the behavior of the LPOD-DG method in the framework of hp-adaptive schemes: in the regions characterized by shock waves fine meshes with low-order DG bases could be used for both high-fidelity and low-fidelity simulations. On the contrary, smooth regions could be described by coarse meshes which use high-order DG bases in the high-fidelity simulations and LPOD-DG bases in the low-fidelity predictions.
The proposed methods have been tested on the prediction of turbulent flows by means of RANS equations in a test case characterized by a fixed geometry studied under several different working conditions. This kind of study could be used to accelerate RANS simulations in robust design analysis of industrial components.

10 Acknowledgements

This work was partially supported by the AEROGUST project which has been funded from the European Union's Horizon 2020 research and innovation programme under grant agreement No 636053. The partners in AEROGUST are: University of Bristol, INRIA, NLR, DLR, University of Cape Town, NUMECA, Optimad Engineering S.r.l., University of Liverpool, Airbus Defence and Space, Dassault Aviation, Piaggio Aerospace and Valeol.
Experiments presented in this paper were carried out using the PLAFRIM experimental testbed, being developed under the Inria PlaFRIM development action with support from Bordeaux INP, LABRI and IMB and other entities: Conseil Régional d'Aquitaine, Université de Bordeaux and CNRS (and ANR in accordance to the programme Investissements d'Avenir (see https://www.plafrim.fr/).
Computational resources were also provided by HPC@POLITO, a project of Academic Computing within the Department of Control and Computer Engineering at the Politecnico di Torino (http://www.hpc.polito.it).

References

[1] Natalia M Alexandrov, Robert Michael Lewis, Clyde R Gumbert, Lawrence L Green, and Perry A Newman. Approximation and model management in aerodynamic optimization with variable-fidelity models. *Journal of Aircraft*, 38(6):1093–1101, 2001.

[2] Steven R Allmaras and Forrester T Johnson. Modifications and clarifications for the implementation of the spalart-allmaras turbulence model. In *Seventh international conference on computational fluid dynamics (ICCFD7)*, pages 1–11, 2012.

[3] Enrico Ampellio, Francesco Bertini, Andrea Ferrero, Francesco Larocca, and Luca Vassio. Turbomachinery design by a swarm-based optimization method coupled with a cfd solver. *Advances in aircraft and spacecraft science*, 3(2):149–170, 2016.

[4] David Amsallem and Charbel Farhat. Interpolation method for adapting reduced-order models and application to aeroelasticity. *AIAA journal*, 46(7):1803–1813, 2008.

[5] David Amsallem, Matthew J Zahr, and Charbel Farhat. Nonlinear model order reduction based on local reduced-order bases. *International Journal for Numerical Methods in Engineering*, 92(10): 891–916, 2012.

[6] David Amsallem, Matthew Zahr, Youngsoo Choi, and Charbel Farhat. Design optimization using hyper-reduced-order models. *Structural and Multidisciplinary Optimization*, 51(4):919–940, 2015.

[7] Alberto Badías, David González, Iciar Alfaro, Francisco Chinesta, and Elias Cueto. Local proper generalized decomposition. In *AIP Conference Proceedings*, volume 1896, page 170007. AIP Publishing, 2017.

[8] Joan Baiges, Ramon Codina, and Sergio Idelsohn. A domain decomposition strategy for reduced order models. application to the incompressible navier–stokes equations. *Computer Methods in Applied Mechanics and Engineering*, 267:23–42, 2013.

[9] Maxime Barrault, Yvon Maday, Ngoc Cuong Nguyen, and Anthony T Patera. An 'empirical interpolation' method: application to efficient reduced-basis discretization of partial differential equations. *Comptes Rendus Mathematique*, 339(9):667–672, 2004.

[10] Francesco Bassi, Lorenzo Botti, Alessandro Colombo, Daniele A Di Pietro, and Pietro Tesini. On the flexibility of agglomeration based physical space discontinuous galerkin discretizations. *Journal of Computational Physics*, 231(1):45–65, 2012.

[11] Michel Bergmann, C-H Bruneau, and Angelo Iollo. Enablers for robust pod models. *Journal of Computational Physics*, 228(2):516–538, 2009.

[12] Michel Bergmann, Thierry Colin, Angelo Iollo, Damiano Lombardi, Olivier Saut, and Haysam Telib. Reduced order models at work in aeronautics and medicine. In *Reduced Order Methods for Modeling and Computational Reduction*, pages 305–332. Springer, 2014.

[13] Michel Bergmann, Andrea Ferrero, Angelo Iollo, Edoardo Lombardi, Angela Scardigli, and Haysam Telib. A zonal galerkin-free pod model for incompressible flows. *Journal of Computational Physics*, 352:301–325, 2018.

[14] T Braconnier, M Ferrier, J-C Jouhaud, M Montagnac, and P Sagaut. Towards an adaptive pod/svd surrogate model for aeronautic design. *Computers & Fluids*, 40(1):195–209, 2011.

[15] Marcelo Buffoni and Karen Willcox. Projection-based model reduction for reacting flows. In *40th Fluid Dynamics Conference and Exhibit*, page 5008, 2010.

[16] Marcelo Buffoni, Haysam Telib, and Angelo Iollo. Iterative methods for model reduction by domain decomposition. *Computers & Fluids*, 38(6):1160–1167, 2009.

[17] A Burbeau and P Sagaut. A dynamic p-adaptive discontinuous galerkin method for viscous flow with shocks. *Computers & fluids*, 34(4-5):401–417, 2005.

[18] Alfonso Caiazzo, Traian Iliescu, Volker John, and Swetlana Schyschlowa. A numerical investigation of velocity–pressure reduced order models for incompressible flows. *Journal of Computational Physics*, 259:598–616, 2014.

[19] Kevin Carlberg and Charbel Farhat. A compact proper orthogonal decomposition basis for optimization-oriented reduced-order models. In *12th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, page 5964, 2008.

[20] Kevin Carlberg, Charbel Bou-Mosleh, and Charbel Farhat. Efficient non-linear model reduction via a least-squares petrov–galerkin projection and compressive tensor approximations. *International Journal for Numerical Methods in Engineering*, 86(2):155–181, 2011.

[21] Saifon Chaturantabut and Danny C Sorensen. Nonlinear model reduction via discrete empirical interpolation. *SIAM Journal on Scientific Computing*, 32(5):2737–2764, 2010.

[22] Francisco Chinesta, Adrien Leygue, Felipe Bordeu, Jose Vicente Aguado, Elías Cueto, David González, Iciar Alfaro, Amine Ammar, and Antonio Huerta. Pgd-based computational vademecum for efficient design, optimization and control. *Archives of Computational Methods in Engineering*, 20(1):31–59, 2013.

[23] Romain Dupuis, Jean-Christophe Jouhaud, and Pierre Sagaut. Aerodynamic data predictions for transonic flows via a machine-learning-based surrogate model. In *2018 AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, page 1905, 2018.

[24] Andrea Ferrero and Francesco Larocca. Feedback filtering in discontinuous galerkin methods for euler equations. *Progress in Computational Fluid Dynamics, an International Journal*, 16(1):14–25, 2016.

[25] Andrea Ferrero and Francesco Larocca. Adaptive cfd schemes for aerospace propulsion. *Journal of Physics: Conference Series*, 841(1), 2017.

[26] Andrea Ferrero, Francesco Larocca, and Gabriella Puppo. A robust and adaptive recovery-based discontinuous galerkin method for the numerical solution of convection–diffusion equations. *International Journal for Numerical Methods in Fluids*, 77(2):63–91, 2015.

[27] Andrea Ferrero, Francesco Larocca, and Verena Bernaschek. Unstructured discretisation of a non-local transition model for turbomachinery flows. *Advances in aircraft and spacecraft science*, 4(5): 555–571, 2017.

[28] Christophe Geuzaine and Jean-François Remacle. Gmsh: A 3-d finite element mesh generator with built-in pre-and post-processing facilities. *International journal for numerical methods in engineering*, 79(11):1309–1331, 2009.

[29] Ralf Hartmann and Paul Houston. Adaptive discontinuous galerkin finite element methods for the compressible euler equations. *Journal of Computational Physics*, 183(2):508–532, 2002.

[30] Koen Hillewaert, Corentin Carton de Wiart, Guillaume Verheylewegen, and Tony Arts. Assessment of a high-order discontinuous galerkin method for the direct numerical simulation of transition at low-reynolds number in the t106c high-lift low pressure turbine cascade. In *ASME Turbo Expo 2014: Turbine Technical Conference and Exposition*, pages V02BT39A034–V02BT39A034. American Society of Mechanical Engineers, 2014.

[31] Miloš Ilak and Clarence W Rowley. Modeling of transitional channel flow using balanced proper orthogonal decomposition. *Physics of Fluids*, 20(3):034103, 2008.

[32] Donald R Jones, Matthias Schonlau, and William J Welch. Efficient global optimization of expensive black-box functions. *Journal of Global optimization*, 13(4):455–492, 1998.

[33] Sven Kaulmann, Mario Ohlberger, and Bernard Haasdonk. A new local reduced basis discontinuous galerkin approach for heterogeneous multiscale problems. *Comptes Rendus Mathematique*, 349(23-24):1233–1238, 2011.

[34] Dimitar Lukarski and Nico Trost. Paralution project. *URL http://www. paralution. com. Accessed: December*, 2014.

[35] Yvon Maday and Einar M Ronquist. The reduced basis element method: application to a thermal fin problem. *SIAM Journal on Scientific Computing*, 26(1):240–258, 2004.

[36] Bernd R Noack, Paul Papas, and Peter A Monkewitz. The need for a pressure-term representation in empirical galerkin models of incompressible shear flows. *Journal of Fluid Mechanics*, 523:339–365, 2005.

[37] Stanley Osher and Fred Solomon. Upwind difference schemes for hyperbolic systems of conservation laws. *Mathematics of computation*, 38(158):339–374, 1982.

[38] Maurizio Pandolfi. A contribution to the numerical prediction of unsteady flows. *AIAA journal*, 22 (5):602–610, 1984.

[39] Benjamin Peherstorfer, Karen Willcox, and Max Gunzburger. Survey of multifidelity methods in uncertainty propagation, inference and optimization. Technical report, Technical Report TR-16-1. Aerospace Computational Design Laboratory, Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, MA, USA, 2016.

[40] Alfio Quarteroni and Gianluigi Rozza. *Reduced order methods for modeling and computational reduction*, volume 9. Springer, 2014.

[41] Alfio Quarteroni, Gianluigi Rozza, and Andrea Manzoni. Certified reduced basis approximation for parametrized partial differential equations and applications. *Journal of Mathematics in Industry*, 1 (1):3, 2011.

[42] Clarence W Rowley. Model reduction for fluids, using balanced proper orthogonal decomposition. *International Journal of Bifurcation and Chaos*, 15(03):997–1013, 2005.

[43] Clarence W Rowley, Tim Colonius, and Richard M Murray. Model reduction for compressible flows using pod and galerkin projection. *Physica D: Nonlinear Phenomena*, 189(1):115–129, 2004.

[44] David Ryckelynck. A priori hyperreduction method: an adaptive approach. *Journal of computational physics*, 202(1):346–366, 2005.

[45] Peter J Schmid. Dynamic mode decomposition of numerical and experimental data. *Journal of fluid mechanics*, 656:5–28, 2010.

[46] Sirod Sirisup, George Em Karniadakis, Dongbin Xiu, and Ioannis G Kevrekidis. Equation-free/galerkin-free pod-assisted computation of incompressible flows. *Journal of Computational Physics*, 207(2):568–587, 2005.

[47] Lawrence Sirovich. Turbulence and the dynamics of coherent structures part i: coherent structures. *Quarterly of applied mathematics*, 45(3):561–571, 1987.

[48] WA Timmer and RPJOM Van Rooij. Summary of the delft university wind turbine dedicated airfoils. *Journal of solar energy engineering*, 125(4):488–496, 2003.

[49] Massimiliano Vasile, Edmondo Minisci, Domenico Quagliarella, Marc Guénot, Ingrid Lepot, Caroline Sainvitu, Jordan Goblet, and Rajan Filomeno Coelho. Adaptive sampling strategies for non-intrusive pod-based surrogates. *Engineering computations*, 30(4):521–547, 2013.

[50] Li Wang and Dimitri J Mavriplis. Adjoint-based h–p adaptive discontinuous galerkin methods for the 2d compressible euler equations. *Journal of Computational Physics*, 228(20):7643–7661, 2009.

[51] Zhu Wang, Imran Akhtar, Jeff Borggaard, and Traian Iliescu. Two-level discretizations of nonlinear closure models for proper orthogonal decomposition. *Journal of Computational Physics*, 230(1): 126–146, 2011.

[52] D Wells, Z Wang, X Xie, and T Iliescu. An evolve-then-filter regularized reduced order model for convection-dominated flows. *International Journal for Numerical Methods in Fluids*, 84(10): 598–615, 2017.

[53] Karen Willcox and Jaime Peraire. Balanced model reduction via the proper orthogonal decomposition. *AIAA journal*, 40(11):2323–2330, 2002.

[54] Matthew J Zahr, Philip Avery, and Charbel Farhat. A multilevel projection-based model order reduction framework for nonlinear dynamic multiscale problems in structural and solid mechanics. *International Journal for Numerical Methods in Engineering*, 112(8):855–881, 2017.

[55] Zhao Zhan, Wagdi G Habashi, and Marco Fossati. Local reduced-order modeling and iterative sampling for parametric analyses of aero-icing problems. *AIAA Journal*, 53(8):2174–2185, 2015.