



Application of physics-constrained data-driven reduced-order models to shape optimization

Hamid R. Karbasian^{1,†} and Brian C. Vermeire¹

¹Department of Mechanical, Industrial and Aerospace Engineering, Concordia University, Montreal, H3G 1M8, Canada

(Received 17 June 2021; revised 30 September 2021; accepted 22 November 2021)

This study proposes a novel approach for building surrogate models, in the form of reduced-order models (ROMs), for partial differential equation constrained optimization. A physics-constrained data-driven framework is developed to transform large-scale nonlinear optimization problems into ROM-constrained optimization problems. Unlike conventional methods, which suffer from instability of the forward sensitivity function, the proposed approach maps optimization problems to system dynamics optimization problems in Hilbert space to improve stability, reduce memory requirements, and lower computational cost. The utility of this approach is demonstrated for aerodynamic optimization of an NACA 0012 airfoil at $Re = 1000$. A drag reduction of 9.35% is obtained at an effective angle of attack of eight degrees, with negligible impact on lift. Similarly, a drag reduction of 20% is obtained for fully separated flow at an angle of attack of 25°. Results from these two optimization problems also reveal relationships between optimization in physical space and optimization of dynamical behaviours in Hilbert space.

Key words: computational methods, chaos, nonlinear instability

1. Introduction

Optimization arises in a wide range of applications from engineering design (Ravindran 2000; Bergmann, Cordier & Brancher 2005; Rozza & Manzoni 2010) to control systems (Lassila & Rozza 2010; Manzoni, Quarteroni & Rozza 2012). Increasing computational power has recently enabled the application of optimization techniques for industrial-scale problems. These problems are usually governed by large nonlinear partial differential equations (PDEs) with suitable initial and boundary conditions, which are often referred to as full-order models (FOMs). These FOMs arise in computational mechanics applications, such as computational fluid dynamics (CFD) (Orozco & Ghattas 1992; Newman *et al.* 1999; Nadarajah & Jameson 2000). In practice, these FOMs act as a constraint during

† Email address for correspondence: h.karbasian@utoronto.ca

optimization, referred to as PDE-constrained optimization. This type of optimization can be prohibitively expensive, since it requires one or more high-fidelity FOM solutions per design iteration (Yamaleev, Diskin & Nielsen 2008; Zahr & Persson 2016). Solving multiple large-scale FOMs using high-fidelity solvers in the context of CFD, such as large-eddy simulation and direct numerical simulation, can be prohibitively expensive. Therefore, the majority of optimization studies are performed using lower-fidelity models, such as the Reynolds-averaged Navier–Stokes approach.

To alleviate the prohibitive cost of PDE-constrained optimization using FOMs, a wide range of studies have developed surrogate modelling techniques (Moarref *et al.* 2014; Ling & Templeton 2015; Mohebujjaman, Reibold & Iliescu 2018). Surrogate models of nonlinear PDEs are usually referred to as reduced-order models (ROMs) (Oliveira & Patera 2007; Bui-Thanh, Willcox & Ghattas 2008a; Hinze & Matthes 2011). Dimensionality reduction using ROMs has become an attractive approach, and is employed successfully in optimal control (Bergmann & Cordier 2008; Kunisch & Volkwein 2008), optimization (Rozza & Manzoni 2010; Zahr & Farhat 2015) and sensitivity analysis (Bui-Thanh, Willcox & Ghattas 2008b; Zahr & Farhat 2015; Zahr & Persson 2016). Optimization combined with surrogate models introduces a new approach called ROM-constrained optimization. Generally speaking, ROMs can be embedded within machine learning frameworks (Bright, Lin & Kutz 2013; Rowley & Dawson 2017; Taira *et al.* 2017), and are based on a projection of FOMs from physical space into a lower-dimensional model space (i.e. Hilbert space, eigenspace, Banach space, etc.) and *vice versa* (Rowley & Dawson 2017; Taira *et al.* 2017, 2020). This projection typically employs trial basis functions. These trial basis functions typically use matrix representations, which define the reduced-order basis (ROB). For example, the ROB can be eigenvectors or bases of a dataset obtained via proper orthogonal decomposition (POD). The ROMs leveraging machine learning features are usually constructed by collecting data initially obtained from FOMs (Brunton & Kutz 2019; Brunton, Noack & Koumoutsakos 2020; Murata, Fukami & Fukagata 2020). After dimensionality reduction, other machine learning techniques can be applied to develop new models that can predict the behaviour of the dynamical system. Depending on the system's complexity, or nonlinearity, hidden layers can be used to build deep learning algorithms.

Despite recent advancements in machine learning, ROMs specialized for optimization have not yet achieved promising results for unsteady flows. This is because existing ROMs only estimate the state vector of the FOM at specified time intervals, but not its sensitivity to perturbations of design parameters. Furthermore, conventional ROMs can yield non-physical solutions when these design variables are perturbed (Epureanu 2003; Amsallem & Farhat 2008; Hay, Borggaard & Pelletier 2009). This issue arises from the fact that FOMs of nonlinear dynamical systems are highly sensitive to perturbations (Wang, Hu & Blonigan 2014; Karbasian & Vermeire 2022). Trust-region approaches, such as trust-region POD (Fahl 2000) and optimal system POD (Kunisch & Volkwein 2008), have been proposed to improve the robustness of ROMs with respect to perturbations of design parameters (Arian, Fahl & Sachs 2000; Carlberg & Farhat 2011). However, these methods still need multiple FOMs to be used for sensitivity analysis.

In addition to PDE-constrained optimization, ROMs for unsteady optimization often fail to obtain the sensitivity function. The main reason for this failure is that the behaviour of nonlinear dynamical systems governing high-fidelity FOMs leads to unconditionally unstable sensitivity functions for long-term data analysis (which is usually the case in unsteady CFD simulations). Secondly, a typical ROM ignores low-energy ranks that might significantly affect the evolutionary behaviour of the dynamical system. This issue results in missing information in the sensitivity functions. Therefore, adding extra terms

(or innovatively designing the surrogate model) to compensate for the effect of these low-energy ranks results in a closure model in the form of the ROM. Although a short-term data analysis might exhibit stability over a short period, improper ROM development ultimately leads to inaccurate results.

To the authors' knowledge, aerodynamic optimization in the presence of unsteady flows has not been considered thoroughly by other researchers. Although there has been some progress in this area, it has been limited to quasi-steady, periodic and laminar flows (Yoon 2016; Rubino *et al.* 2018; He *et al.* 2019; Apponsah & Zingg 2020; He & Martins 2021). Directly optimizing chaotic FOMs has also been considered, but is limited to relatively small-scale systems (Blonigan, Gomez & Wang 2014; Wang *et al.* 2014). In this study, we develop a closure model for a ROM by leveraging physics-constrained data-driven principles to overcome limitations for unsteady optimization of unstable flows. In the proposed algorithm, we derive sensitivity functions from the Navier–Stokes equations, such that the closure model can be represented as a physical (i.e. interpretable) surrogate model in Hilbert space. The proposed algorithm leverages the minimum-residual property to assure the optimality condition throughout the optimization procedure. This optimality condition means searching for optimal directions in subspaces to minimize the residual in the closure model. This condition is assessed via the least squares Petrov–Galerkin (LSPG) approach. Additionally, in the ROB, subspaces are built via POD modes. It is noteworthy that the trial basis function is updated at each design cycle to improve the predictions of the closure model.

The remainder of this paper is organized as follows. In § 2, a general form of a PDE-constrained optimization is defined, and principles of the forward sensitivity function are discussed. Section 3 discusses the intuition of solving minimization problems in low-dimensional subspaces, and it provides a basic concept for the ROM-constrained optimization. In § 4, a closure model in the form of the ROM is developed via the LSPG approach. Section 5 extends this closure model to the forward sensitivity function in Hilbert space. In § 6, we implement the physics-constrained ROM into a platform with which sensitivities are computed. In § 7, different strategies for building the ROB are considered. These strategies help shape the manifolds in Hilbert space. In § 8, unsteady aerodynamic optimization of an NACA 0012 at high angles of attack is considered, and the resulting designs are discussed in detail. Section 9 provides overall conclusions, and explains the utility of the ROM-constrained optimization for future unsteady aerodynamic designs.

2. PDE-constrained optimization

Let us consider a dynamical system represented as a set of PDEs over a smooth time-domain \mathbb{T} , as follows:

$$\frac{\partial \mathbf{u}_a}{\partial t} = \mathbf{f}_a(\mathbf{u}_a, t, \mathcal{S}), \quad \mathbf{u}_a(0, \mathcal{S}) = \mathbf{u}_{a,0}, \quad (2.1a,b)$$

where $t \in \mathbb{R}_+$ is time and $t \in \mathbb{T}$, $\mathbf{u}_a(t, \mathcal{S}) \in \mathbb{R}^\infty$ denotes the state vector, $\mathbf{u}_{a,0} \in \mathbb{R}^\infty$ is the initial state vector at $t = 0$. Additionally, \mathcal{S} is a vector of design variables $\mathcal{S} = [\mathcal{S}_1, \mathcal{S}_2, \dots, \mathcal{S}_{n_s}]^\top$, where n_s shows the total number of design variables in the design space, \mathcal{D} . Also, $\mathbf{f}_a : \mathbb{R}^\infty \rightarrow \mathbb{R}^\infty$ characterizes the dynamical behaviour of the system on the right-hand side of (2.1a,b). In the context of CFD, a spatial discretization can be applied to \mathbf{f}_a , yielding a discrete approximation with n_u dimensions, $\mathbf{f} : \mathbb{R}^{n_u} \rightarrow \mathbb{R}^{n_u}$, and a system of ODEs that approximate the original PDE. The approximate state vector found

using this system of ODEs is denoted $\mathbf{u}(t, \mathcal{S})$. Numerical error introduced by discretization is usually expressed as an additional residual term

$$\frac{\partial \mathbf{u}}{\partial t} = \mathbf{f}(\mathbf{u}, t, \mathcal{S}) + \mathbf{r}(\mathbf{u}, t, \mathcal{S}), \quad \mathbf{u}(0, \mathcal{S}) = \mathbf{u}_0, \tag{2.2a,b}$$

where $\mathbf{r}(\mathbf{u}, t, \mathcal{S}) : \mathbb{R}^{n_u} \rightarrow \mathbb{R}^{n_u}$ is the residual of the discretized PDEs. In practice, the dynamical behaviour of (2.2a,b) is investigated via a set of observables, which are called objective functions. Consider a scalar objective function dependent on the full state \mathbf{u} and design variables \mathcal{S} . This can be defined as $\mathcal{J}(\mathbf{u}, \mathcal{S}) : \mathbb{R}^{n_u} \rightarrow \mathbb{R}$. Usually, the objective function takes the form of a time average

$$\bar{\mathcal{J}}(\mathcal{S}) = \frac{1}{T} \int_0^T \mathcal{J}(\mathbf{u}, t, \mathcal{S}) dt, \tag{2.3}$$

where $\bar{\mathcal{J}}$ is the time-averaged objective function and $T \in \mathbb{T}$ is a time averaging period. Using (2.2a,b) and (2.3), we can define a PDE-constrained optimization

$$\left. \begin{aligned} &\text{minimize}_{\mathbf{u} \in \mathbb{R}^{n_u}, \mathcal{S} \in \mathcal{D}} \quad \bar{\mathcal{J}} = \frac{1}{T} \int_0^T \mathcal{J}(\mathbf{u}, t, \mathcal{S}) dt, \\ &\text{subject to} \quad \mathbf{r}(\mathbf{u}, t, \mathcal{S}) = 0, \\ &\quad \quad \quad \frac{d\mathbf{r}}{d\mathcal{S}}(\mathbf{u}, t, \mathcal{S}) = 0, \\ &\quad \quad \quad \mathbf{C}(\mathbf{u}, \mathcal{S}) \leq 0, \end{aligned} \right\} \tag{2.4}$$

where $d\mathbf{r}/d\mathcal{S}$ is the sensitivity of residual to a set of design variables, a critical constraint when the dynamical system is nonlinear, and $\mathbf{C}(\mathbf{u}, \mathcal{S})$ is a set of user-defined constraints (i.e. geometrical constraints). It is worth mentioning that adding the sensitivity of the residual to (2.4) is sometimes unusual. However, to keep consistency in formulations and have a better connection between PDE-constrained and ROM-constrained optimizations, we used this sensitivity function in the general definition of optimization problem. To solve large-scale minimization problems of this form via gradient-based methods, the time-averaged sensitivity of (2.3) with respect to \mathcal{S} is required:

$$\frac{d\bar{\mathcal{J}}}{d\mathcal{S}} = \left[\frac{\partial \mathcal{J}}{\partial \mathbf{u}} \right]^T \mathbf{v} + \frac{\partial \mathcal{J}}{\partial \mathcal{S}}, \tag{2.5}$$

where $\mathbf{v} = \partial \mathbf{u} / \partial \mathcal{S} \in \mathbb{R}^{n_u}$. Substituting (2.5) into (2.3), the time-averaged sensitivity of the objective function with respect to \mathcal{S} is

$$\frac{d\bar{\mathcal{J}}}{d\mathcal{S}} = \frac{1}{T} \int_0^T \left(\left[\frac{\partial \mathcal{J}}{\partial \mathbf{u}} \right]^T \mathbf{v} + \frac{\partial \mathcal{J}}{\partial \mathcal{S}} \right) dt. \tag{2.6}$$

Hence, the sensitivity of the objective function is directly related to the sensitivity of the solution. In order to find this, we use the gradient of the dynamical system from (2.2a,b):

$$\frac{\partial \delta \mathbf{u}}{\partial t} = \frac{\partial \mathbf{f}}{\partial \mathbf{u}} \delta \mathbf{u} + \frac{\partial \mathbf{f}}{\partial \mathcal{S}} \delta \mathcal{S} + \frac{d\mathbf{r}}{d\mathcal{S}} \delta \mathcal{S}, \tag{2.7}$$

where δ denotes perturbations. Premultiplying all terms in (2.7) by $1/\delta \mathcal{S}$, and assuming $\mathbf{v} = \partial \mathbf{u} / \partial \mathcal{S} \simeq \delta \mathbf{u} / \delta \mathcal{S}$, the following expression is obtained:

$$\frac{\partial \mathbf{v}}{\partial t} = \frac{\partial \mathbf{f}}{\partial \mathbf{u}} \mathbf{v} + \frac{\partial \mathbf{f}}{\partial \mathcal{S}} + \frac{d\mathbf{r}}{d\mathcal{S}}, \quad \mathbf{v}(0, \mathcal{S}) = \mathbf{v}_0. \tag{2.8a,b}$$

Because (2.8a,b) evolves in time (i.e. $t^{(n)} < t^{(n+1)}$), where n is the index of time intervals) from an initial condition $\mathbf{v}(0, \mathcal{S}) = \mathbf{v}_0$, it is called the forward sensitivity function. Finally, the sensitivity of the solution to the design variables can be obtained by solving (2.8a,b), and then, the gradient of the objective function can be computed via (2.6) for each design cycle. However, (2.8a,b) diverges for chaotic systems because of unstable modes in the dynamical behaviour of the system. In the next section, we illustrate the concept of ROM-constrained optimization to reduce computational costs, and address this stability issue for the forward sensitivity function.

3. ROM-constrained optimization

In ROM-constrained optimization, the governing equations are projected onto low-dimensional subspaces. Therefore, instead of solving massive PDEs for each optimization iteration, we only solve a compact form of the governing equations, as a set of ROMs, which has fewer degrees of freedom and complexity. Figure 1 shows a general approach for optimization problems. Conventionally, optimization problems are solved in physical space, \mathcal{P} . Consider the left-hand side object as the initial design in figure 1. To optimize the shape of this object, we solve the optimization problem subject to a Hamiltonian dynamical system defined as $\nabla \cdot \mathcal{H}(\mathbf{u}, \mathcal{S})$. In this example, \mathbf{u} and \mathcal{S} represent state and shape parameters, respectively. The right-hand side object in \mathcal{P} is the optimized shape. According to the properties of a Hamiltonian system, the dynamical system for each object has a unique representation in Hilbert space \mathcal{H} . Usually, this representation is a set of generalized coordinates, \mathbf{q} , that define the dynamical behaviour of the corresponding object in \mathcal{H} . Therefore, in order to find the representation of the reference object, we project its corresponding state vector from \mathcal{P} to \mathcal{H} using a projection function \mathbb{P} . We assume that \mathbb{F} is a surrogate model that represents the exact evolution of the state vector in \mathcal{H} , such that that $\mathbb{F} : \mathbf{q} \times \mathbb{T} \rightarrow \mathbf{q}$ with $(\mathbf{q}, t) \mapsto \mathbb{F}(\mathbf{q}, t)$ in $t \in \mathbb{T}$. Now if we apply this optimization problem subject to \mathbb{F} , we obtain the optimized set of generalized coordinates. In the end, we lift back these optimized generalized coordinates from \mathcal{H} to \mathcal{P} via \mathbb{P}' , obtaining the same optimized object in \mathcal{P} . It is worth mentioning that \mathbb{F} is unknown, and we should build a closure model with rank $r \in \mathbb{R}_+$ given by $\mathcal{F} : \mathbb{R}^r \rightarrow \mathbb{R}^r$, such that $\mathcal{F} : \mathbf{q} \times \mathbb{T} \rightarrow \mathbf{q}$, in the form of a ROM to approximate \mathbb{F} . The closer \mathcal{F} to \mathbb{F} , the closer the optimization results will be to the true optimal design in \mathcal{P} .

4. Dimensionality reduction

The full state can be defined as a linear combination of vectors embedded in the matrix of trial basis functions (a combination of subspaces in \mathcal{H} used as projection function \mathbb{P}). Therefore, the full state can be given by

$$\mathbf{u}(t, \mathcal{S}) = \bar{\mathbf{u}} + \tilde{\Phi} \tilde{\mathbf{q}}(t, \mathcal{S}) + \Phi' \mathbf{q}'(t, \mathcal{S}), \tag{4.1}$$

where $\tilde{\Phi}$ and Φ' indicate the coarse-scale and fine-scale modes of the dynamical system. Furthermore, $\tilde{\mathbf{q}}$ and \mathbf{q}' are generalized coordinates corresponding to $\tilde{\Phi}$ and Φ' , respectively, and $\bar{\mathbf{u}} \in \mathbb{R}^{n_u}$ is the vector of reference states. Writing the full state in the form of (4.1) will help distinguish between different types of variables, dominant modes, and the dynamical response of the full state. For instance, in turbulent flow, the total velocity magnitude can be decomposed into different modes to identify vortex shedding patterns. Using this kind of decomposition, it is possible to recognize each mode's effect on the turbulent flow. Since energetic modes are embedded in $\tilde{\Phi}$, we can truncate (4.1) and approximate the full

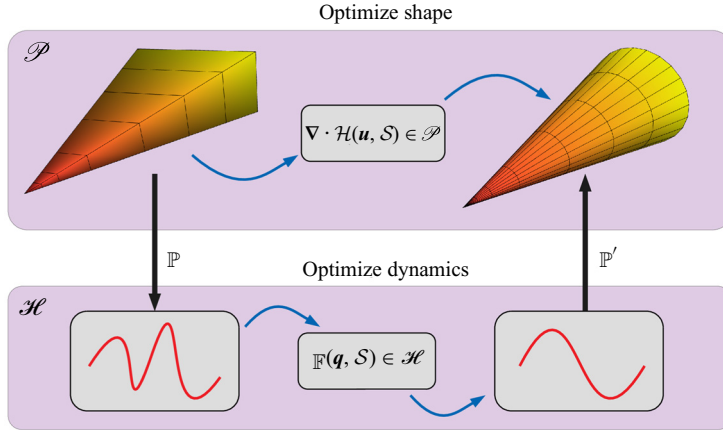


Figure 1. Optimization of the shape of an arbitrary object subject to a Hamiltonian system in physical space \mathcal{P} versus optimization of its dynamics subject to a surrogate model in Hilbert space \mathcal{H} .

state by

$$\mathbf{u}(t, \mathcal{S}) \approx \bar{\mathbf{u}} + \tilde{\Phi} \tilde{\mathbf{q}}(t, \mathcal{S}), \quad (4.2)$$

which is the reduced-order description of the full state \mathbf{u} in a lower-dimensional subspace. Note that $\tilde{\Phi}$, as a projection function \mathbb{P} , can accurately approximate the full state if the truncated rank $r \in \mathbb{R}_+$ is sufficiently large (the rank is the number of modes embedded in $\tilde{\Phi}$) such that $\tilde{\Phi}$ can capture the majority of the dynamical system’s energy. By substituting (4.2) into (2.2a,b), and constraining the nonlinear equations to be orthogonal to a set of test subspaces $\tilde{\Psi} \in \mathbb{R}^{n_u \times r}$, we can specify the ROM as

$$\tilde{\Psi}^\top \left(\tilde{\Phi} \frac{\partial \tilde{\mathbf{q}}}{\partial t} - \mathbf{f}(\bar{\mathbf{u}} + \tilde{\Phi} \tilde{\mathbf{q}}, t, \mathcal{S}) \right) + \tilde{\Psi}^\top \mathbf{r}(\bar{\mathbf{u}} + \tilde{\Phi} \tilde{\mathbf{q}}, t, \mathcal{S}) = 0, \quad \tilde{\mathbf{q}}(0, \mathcal{S}) = \tilde{\mathbf{q}}_0, \quad (4.3a,b)$$

where $\tilde{\mathbf{q}}_0$ is the initial condition of the ROM, and $\tilde{\Psi}$ is the test basis function, which is used to minimize the distance to the state of the FOM in some norm. Therefore, according to $\tilde{\Psi}^\top \mathbf{r}(\bar{\mathbf{u}} + \tilde{\Phi} \tilde{\mathbf{q}}, t, \mathcal{S}) \approx 0$, (4.3a,b) can be simplified to

$$\tilde{\Psi}^\top \left(\tilde{\Phi} \frac{\partial \tilde{\mathbf{q}}}{\partial t} - \mathbf{f}(\bar{\mathbf{u}} + \tilde{\Phi} \tilde{\mathbf{q}}, t, \mathcal{S}) \right) = 0, \quad \tilde{\mathbf{q}}(0, \mathcal{S}) = \tilde{\mathbf{q}}_0. \quad (4.4a,b)$$

If $\tilde{\Psi} = \tilde{\Phi}$, then the (4.4a,b) represents the Galerkin projection of the FOM (Carlberg, Barone & Antil 2017). For $\tilde{\Psi} = (\partial \mathbf{r} / \partial \mathbf{u}) \tilde{\Phi}$, (4.4a,b) is the LSPG projection (Bui-Thanh *et al.* 2008a,b; Carlberg, Bou-Mosleh & Farhat 2011; Carlberg *et al.* 2013, 2017). The distinguishing property of LSPG is that the projection remains optimal for dynamical systems with non-symmetric positive definite Jacobians (Zahr & Farhat 2015). Furthermore, It has shown that LSPG has the minimum-residual property (Zahr & Farhat 2015; Carlberg *et al.* 2017), which can be written as

$$\tilde{\mathbf{q}} = \operatorname{argmin}_{\mathbf{z} \in \mathbb{R}^r} \left\| \mathbf{r}(\bar{\mathbf{u}} + \tilde{\Phi} \mathbf{z}, t, \mathcal{S}) \right\|_2^2. \quad (4.5)$$

Generally speaking, LSPG adds a term to the Galerkin projection in order to stabilize it numerically. This stabilization becomes essential when FOMs possess nonlinear behaviour in the design space \mathcal{D} . Consequently, in this study, the closure model for the ROM is built based on LSPG projection.

5. Sensitivity analysis

The state sensitivity can be obtained by taking the derivative of (4.2) with respect to \mathcal{S} such that

$$\mathbf{v}(t, \mathcal{S}) \approx \frac{\partial \bar{\mathbf{u}}}{\partial \mathcal{S}} + \frac{\partial \tilde{\Phi}}{\partial \mathcal{S}} \tilde{\mathbf{q}}(t, \mathcal{S}) + \tilde{\Phi} \tilde{\mathbf{h}}(t, \mathcal{S}), \tag{5.1}$$

where $\partial \bar{\mathbf{u}}/\partial \mathcal{S} \in \mathbb{R}^{n_u}$ is the vector of reference sensitivities. According to (5.1), the solution of the sensitivity $\tilde{\mathbf{h}} = \partial \tilde{\mathbf{q}}/\partial \mathcal{S} \in \mathbb{R}^r$ can be approximated by the sensitivities of the prominent modes embedded in $\tilde{\Phi}$. Moreover, $\partial \tilde{\Phi}/\partial \mathcal{S}$ could be negligible if the perturbation of design parameters does not have a notable influence on the state. Otherwise, $\partial \tilde{\Phi}/\partial \mathcal{S}$ will be a required term in the optimization. To develop the sensitivity function, we assume that the ROM is identical to (2.2a,b). Hence, a set of ODEs in the form of a ROM can be written as

$$\frac{\partial \tilde{\mathbf{q}}}{\partial t} = \mathcal{F}(\tilde{\mathbf{q}}, t, \mathcal{S}) + \mathcal{R}(\tilde{\mathbf{q}}, t, \mathcal{S}), \quad \tilde{\mathbf{q}}(0, \mathcal{S}) = \tilde{\mathbf{q}}_0, \tag{5.2a,b}$$

where $\mathcal{R} : \mathbb{R}^r \rightarrow \mathbb{R}^r$ and $\mathcal{F} : \mathbb{R}^r \rightarrow \mathbb{R}^r$ with $\tilde{\mathbf{q}} \mapsto \mathcal{F}(\tilde{\mathbf{q}})$ are the residual and flux of the closure model, respectively. In the ideal case, the residual of the FOM and the ROM is zero. However, approximating the state of the FOM via dimensionality reduction (i.e. $\mathbf{u} \approx \bar{\mathbf{u}} + \tilde{\Phi} \tilde{\mathbf{q}}$) and frequent projection via \mathbb{P} will introduce non-zero values to the residual \mathcal{R} . Petrov–Galerkin projection minimizes $\mathcal{R}(\tilde{\mathbf{q}}, t, \mathcal{S})$ by projecting it onto $\mathbf{r}(\bar{\mathbf{u}} + \tilde{\Phi} \tilde{\mathbf{q}}, t, \mathcal{S})$ in \mathcal{P} by means of $\mathbb{P} = \tilde{\Psi}$. Therefore, we can write the Petrov–Galerkin projection of the residual as

$$\mathcal{R}(\tilde{\mathbf{q}}, t, \mathcal{S}) = \tilde{\Psi}^\top \mathbf{r}(\bar{\mathbf{u}} + \tilde{\Phi} \tilde{\mathbf{q}}, t, \mathcal{S}). \tag{5.3}$$

Substituting (2.2a,b) into (5.3), yields the same equation as (4.4a,b). Note that this equation in discrete form corresponds to LSPG. Similar to (2.4), to satisfy the constraints of the optimization problem, the derivative of the residual should be set to zero. By assuming that the rank of the ROM is sufficient, we can derive the sensitivity function from (4.4a,b) as

$$\begin{aligned} \frac{d\mathcal{R}}{d\mathcal{S}} &= \frac{\partial \tilde{\Psi}^\top}{\partial \mathcal{S}} \left(\frac{\partial (\bar{\mathbf{u}} + \tilde{\Phi} \tilde{\mathbf{q}})}{\partial t} - \mathbf{f}(\bar{\mathbf{u}} + \tilde{\Phi} \tilde{\mathbf{q}}, t, \mathcal{S}) \right) \\ &+ \tilde{\Psi}^\top \left(\frac{\partial \tilde{\Phi}}{\partial \mathcal{S}} \frac{\partial \tilde{\mathbf{q}}}{\partial t} + \tilde{\Phi} \frac{\partial \tilde{\mathbf{h}}}{\partial t} - \frac{\partial \mathbf{f}}{\partial \bar{\mathbf{u}}} \frac{\partial \bar{\mathbf{u}}}{\partial \mathcal{S}} - \frac{\partial \mathbf{f}}{\partial \mathbf{u}} \frac{\partial \tilde{\Phi}}{\partial \mathcal{S}} \tilde{\mathbf{q}} - \frac{\partial \mathbf{f}}{\partial \mathbf{u}} \tilde{\Phi} \tilde{\mathbf{h}} - \frac{\partial \mathbf{f}}{\partial \mathcal{S}} \right). \end{aligned} \tag{5.4}$$

Also, the vector of reference sensitivity is defined as $\bar{\mathbf{v}} = \partial \bar{\mathbf{u}}/\partial \mathcal{S}$. By leveraging the continuous representation of LSPG, we can define the test basis function as $\tilde{\Psi} = (\partial \mathbf{r}/\partial \mathbf{u}) \tilde{\Phi}$, and then substitute it into (5.4) to obtain the following equation:

$$\begin{aligned} \frac{d\mathcal{R}}{d\mathcal{S}} &= \left(\frac{\partial \tilde{\Phi}^\top}{\partial \mathcal{S}} \frac{\partial \mathbf{r}^\top}{\partial \mathbf{u}} + \tilde{\Phi}^\top \frac{\partial}{\partial \mathcal{S}} \frac{\partial \mathbf{r}^\top}{\partial \mathbf{u}} \right) \left(\tilde{\Phi} \frac{\partial \tilde{\mathbf{q}}}{\partial t} - \mathbf{f}(\bar{\mathbf{u}} + \tilde{\Phi} \tilde{\mathbf{q}}, t, \mathcal{S}) \right) \\ &+ \tilde{\Psi}^\top \left(\frac{\partial \tilde{\Phi}}{\partial \mathcal{S}} \frac{\partial \tilde{\mathbf{q}}}{\partial t} + \tilde{\Phi} \frac{\partial \tilde{\mathbf{h}}}{\partial t} - \frac{\partial \mathbf{f}}{\partial \bar{\mathbf{u}}} \bar{\mathbf{v}} - \frac{\partial \mathbf{f}}{\partial \mathbf{u}} \frac{\partial \tilde{\Phi}}{\partial \mathcal{S}} \tilde{\mathbf{q}} - \frac{\partial \mathbf{f}}{\partial \mathbf{u}} \tilde{\Phi} \tilde{\mathbf{h}} - \frac{\partial \mathbf{f}}{\partial \mathcal{S}} \right). \end{aligned} \tag{5.5}$$

The second-order derivative of the residual, $(\partial/\partial \mathcal{S})(\partial \mathbf{r}/\partial \mathbf{u})^\top$, is usually not available or is expensive to compute. Furthermore, $\partial \tilde{\Phi}/\partial \mathcal{S}$ cannot be determined from the trial

basis function. If the FOM is not very sensitive to the design variables, we can neglect it by setting $\partial \tilde{\Phi} / \partial \mathcal{S} = 0$. However, this is not acceptable in most cases. To compute $d\mathcal{R}/d\mathcal{S}$ efficiently, we need to remove unknown (or expensive) terms from (5.5). To this end, we rewrite (5.4) with reference to (5.2a,b) as

$$\frac{d\mathcal{R}}{d\mathcal{S}} = \frac{\partial \mathcal{R}}{\partial \tilde{\mathbf{q}}} \tilde{\mathbf{h}} + \frac{\partial \mathcal{R}}{\partial \mathcal{S}} + \frac{\partial \mathcal{R}}{\partial \Omega}, \tag{5.6}$$

where $\partial \mathcal{R} / \partial \Omega \in \mathbb{R}^r$ is derivative of the residual with respect to a source term. Also, derivatives in (5.6) have the following definition:

$$\left. \begin{aligned} \frac{\partial \mathcal{R}}{\partial \tilde{\mathbf{q}}} &= \frac{\partial}{\partial t} - \frac{\partial \mathcal{F}}{\partial \tilde{\mathbf{q}}}, \\ \frac{\partial \mathcal{R}}{\partial \mathcal{S}} &= -\frac{\partial \mathcal{F}}{\partial \mathcal{S}}, \\ \frac{\partial \mathcal{R}}{\partial \Omega} &= -\frac{\partial \mathcal{F}}{\partial \bar{\mathbf{u}}}. \end{aligned} \right\} \tag{5.7}$$

The difference between (5.5) and (5.6) arises from the fact that (5.5) is derived from a closure model that only includes the state information in its hierarchical structures. Here we assume that $\tilde{\Phi}$ includes the state information. However, (5.6) assumes that the closure model uses other hierarchical structures and information to approximate the sensitivity functions. In other words, $\tilde{\mathbf{h}}$ is approximated by minimizing the residual of the solution of the sensitivity in some norm (which is usually the ℓ^2 norm),

$$\tilde{\mathbf{h}} = \underset{\mathbf{z} \in \mathbb{R}^r}{\operatorname{argmin}} \left\| \frac{\partial(\bar{\mathbf{u}} + \tilde{\Phi} \tilde{\mathbf{q}})}{\partial \mathcal{S}} - \bar{\mathbf{v}} - \tilde{\Phi} \mathbf{z} \right\|_2^2, \tag{5.8}$$

where $\mathbf{v} \approx \partial(\bar{\mathbf{u}} + \tilde{\Phi} \tilde{\mathbf{q}}) / \partial \mathcal{S} = \bar{\mathbf{v}} + \tilde{\Phi} \tilde{\mathbf{h}}$. It is worth pointing out that computing $\tilde{\mathbf{h}}$ requires embedding more information in $\tilde{\Phi}$, as shown in (5.8). Further details will be provided in § 7 to show how to revise $\tilde{\Phi}$ for sensitivity analysis. To solve (5.6), we discretize it using the backward differentiation formula (BDF), a linear multistep scheme. Therefore, a discrete form of (5.6) can be represented as

$$\frac{\xi_0}{\Delta t} \tilde{\mathbf{h}}^{(n)} + \left(\sum_{j=1}^k \frac{\xi_j}{\Delta t} \tilde{\mathbf{h}}^{(n-j)} \right) - \beta_0 \frac{\partial \mathcal{F}^{(n)}}{\partial \tilde{\mathbf{q}}^{(n)}} \tilde{\mathbf{h}}^{(n)} + \frac{\partial \mathcal{R}^{(n)}}{\partial \mathcal{S}} + \frac{\partial \mathcal{R}^{(n)}}{\partial \Omega^{(n)}} = 0, \tag{5.9}$$

such that

$$\lim_{\delta \mathcal{S} \rightarrow 0} \lim_{r \rightarrow n_u} \frac{\partial \mathcal{R}}{\partial \mathcal{S}} = 0. \tag{5.10}$$

Moreover, ζ and β are coefficients of the temporal scheme. Rearranging (5.9) yields

$$\left(\frac{\xi_0}{\Delta t} - \beta_0 \frac{\partial \mathcal{F}^{(n)}}{\partial \tilde{\mathbf{q}}^{(n)}} \right) \tilde{\mathbf{h}}^{(n)} + \sum_{j=1}^k \frac{\xi_j}{\Delta t} \tilde{\mathbf{h}}^{(n-j)} + \frac{\partial \mathcal{R}^{(n)}}{\partial \mathcal{S}} + \frac{\partial \mathcal{R}^{(n)}}{\partial \Omega^{(n)}} = 0. \tag{5.11}$$

Looking at (5.11), the first term is the derivative of the residual with respect to the vector of states at the n th time step, $\partial \mathcal{R}^{(n)} / \partial \tilde{\mathbf{q}}^{(n)}$. Additionally, the second term in this equation

is the derivative of the residual with respect to the vector of states at different time steps, $\forall p \neq n : \{\partial \mathcal{R}^{(n)} / \partial \tilde{\mathbf{q}}^{(p)} = \zeta_j / \Delta t\}$. Therefore, (5.11) can be written as

$$\frac{\partial \mathcal{R}^{(n)}}{\partial \tilde{\mathbf{q}}^{(n)}} \tilde{\mathbf{h}}^{(n)} + \sum_{j=1}^k \left(\frac{\partial \mathcal{R}^{(n)}}{\partial \tilde{\mathbf{q}}^{(n-j)}} \right) \tilde{\mathbf{h}}^{(n-j)} + \frac{\partial \mathcal{R}^{(n)}}{\partial \mathcal{S}} + \frac{\partial \mathcal{R}^{(n)}}{\partial \Omega^{(n)}} = 0. \quad (5.12)$$

Having all derivatives in the form of low-dimensional residuals allows us to convert them into smaller terms in the high-dimensional space. Therefore, we can obtain derivatives directly from the residual of the FOM. Using Petrov–Galerkin projection, the relationship between the ROM and the FOM can be expressed as $\mathcal{R}^{(n)} = \tilde{\Psi}^{(n)\top} \mathbf{r}^{(n)}$ in discrete time steps. Hence, the relation between derivatives of the residual can be represented as follows:

$$\left. \begin{aligned} \frac{\partial \mathcal{R}^{(n)}}{\partial \tilde{\mathbf{q}}^{(n)}} &= \left(\frac{\partial \tilde{\Psi}^{(n)\top}}{\partial \mathbf{u}^{(n)}} \mathbf{r}^{(n)} + \tilde{\Psi}^{(n)\top} \frac{\partial \mathbf{r}^{(n)}}{\partial \mathbf{u}^{(n)}} \right) \frac{\partial \mathbf{u}^{(n)}}{\partial \tilde{\mathbf{q}}^{(n)}}, \\ \frac{\partial \mathcal{R}^{(n)}}{\partial \tilde{\mathbf{q}}^{(p)}} &= \left(\frac{\partial \tilde{\Psi}^{(n)\top}}{\partial \mathbf{u}^{(p)}} \mathbf{r}^{(n)} + \tilde{\Psi}^{(n)\top} \frac{\partial \mathbf{r}^{(n)}}{\partial \mathbf{u}^{(p)}} \right) \frac{\partial \mathbf{u}^{(p)}}{\partial \tilde{\mathbf{q}}^{(p)}}, \quad n \neq p, \\ \frac{\partial \mathcal{R}^{(n)}}{\partial \mathcal{S}} &= \left(\frac{\partial \tilde{\Psi}^{(n)\top}}{\partial \mathcal{S}} \mathbf{r}^{(n)} + \tilde{\Psi}^{(n)\top} \frac{\partial \mathbf{r}^{(n)}}{\partial \mathcal{S}} \right). \end{aligned} \right\} \quad (5.13)$$

Furthermore, the derivative of the residual with respect to the source term is also obtained by

$$\frac{\partial \mathcal{R}^{(n)}}{\partial \Omega^{(n)}} = \tilde{\Psi}^{(n)\top} \sum_{j=0}^k \frac{\partial \mathbf{r}^{(n)}}{\partial \mathbf{u}^{(n-j)}} \bar{\mathbf{v}}. \quad (5.14)$$

Applying LSPG to obtain a ROM with enough POD modes in \mathcal{H} leads the residual of the FOM to be close to zero, such that $\forall n : \{\mathbf{r}^{(n)} : \tilde{\Phi} \tilde{\mathbf{q}} \rightarrow 0\}$. This definition leads to the first term in the derivatives being negligible. On the other hand, $\partial \mathbf{u}^{(n)} / \partial \tilde{\mathbf{q}}^{(n)} = \tilde{\Phi}$. Recall $\tilde{\Psi} = (\partial \mathbf{r} / \partial \mathbf{u}) \tilde{\Phi}$, we can simplify (5.13) as follows:

$$\left. \begin{aligned} \frac{\partial \mathcal{R}^{(n)}}{\partial \tilde{\mathbf{q}}^{(n)}} &= \tilde{\Psi}^{(n)\top} \tilde{\Psi}^{(n)}, \\ \frac{\partial \mathcal{R}^{(n)}}{\partial \tilde{\mathbf{q}}^{(p)}} &= \frac{\zeta_p}{\Delta t} \tilde{\Psi}^{(n)\top} \tilde{\Phi}, \quad n \neq p, \\ \frac{\partial \mathcal{R}^{(n)}}{\partial \mathcal{S}} &= \tilde{\Psi}^{(n)\top} \frac{\partial \mathbf{r}^{(n)}}{\partial \mathcal{S}}. \end{aligned} \right\} \quad (5.15)$$

In the end, by substituting (5.15) into (5.12), the discretized equation can be given by

$$\begin{aligned} &\tilde{\Psi}^{(n)\top} \tilde{\Psi}^{(n)} \tilde{\mathbf{h}}^{(n)} + \sum_{j=1}^k \frac{\zeta_j}{\Delta t} \tilde{\Psi}^{(n)\top} \tilde{\Phi} \tilde{\mathbf{h}}^{(n-j)} + \tilde{\Psi}^{(n)\top} \frac{\partial \mathbf{r}^{(n)}}{\partial \mathcal{S}} \\ &+ \tilde{\Psi}^{(n)\top} \left(\sum_{j=1}^k \zeta_j \mathcal{I} + \frac{\partial \mathbf{r}^{(n)}}{\partial \mathbf{u}^{(n)}} \right) \bar{\mathbf{v}} = 0, \end{aligned} \quad (5.16)$$

where \mathcal{I} is the identity matrix. From (5.16), all terms can now be obtained from derivatives of the FOM. Moreover, (5.16) lets us advance the solution of the sensitivity in time

with large time steps since this equation is solved implicitly. This capability has some significant advantages for optimization problems: (1) lower hard-disk requirements for a group of FOMs; (2) solving the ROM via an implicit scheme guarantees numerical stability (although poor datasets and configurations can lead to spurious results, proper configurations should always be taken into account); (3) LSPG performs efficiently for intermediate time steps (Carlberg *et al.* 2017). Ultimately, the minimization problem of the FOM in (2.4) can be transformed into a new minimization problem for a ROM in lower dimensions as

$$\left. \begin{aligned}
 & \underset{\tilde{\mathbf{q}} \in \mathbb{R}^r, \mathcal{S} \in \mathcal{D}}{\text{minimize}} && \bar{\mathcal{J}} = \frac{1}{T} \int_0^T \mathcal{J}(\bar{\mathbf{u}} + \tilde{\Phi} \tilde{\mathbf{q}}, t, \mathcal{S}) dt, \\
 & \text{subject to} && \mathcal{R}(\bar{\mathbf{u}} + \tilde{\Phi} \tilde{\mathbf{q}}, t, \mathcal{S}) = 0, \\
 & && \frac{d\mathcal{R}}{d\mathcal{S}}(\bar{\mathbf{u}} + \tilde{\Phi} \tilde{\mathbf{q}}, t, \mathcal{S}) = 0, \\
 & && \underset{\mathbf{z} \in \mathbb{R}^r}{\text{argmin}} \left\| \frac{\partial(\bar{\mathbf{u}} + \tilde{\Phi} \tilde{\mathbf{q}})}{\partial \mathcal{S}} - \bar{\mathbf{v}} - \tilde{\Phi} \mathbf{z} \right\|_2^2, \\
 & && \mathcal{C}(\bar{\mathbf{u}} + \tilde{\Phi} \tilde{\mathbf{q}}, \mathcal{S}) \leq 0,
 \end{aligned} \right\} \tag{5.17}$$

where the optimizer uses the ROM to search in the design space \mathcal{D} . A distinguishing feature of this work is that the optimization in (5.17) is solved in \mathcal{H} , as opposed to conventional optimization, which is solved in \mathcal{P} . According to (5.17), we need to obtain manifolds in \mathcal{H} to evolve the solution with time. These manifolds are responsible for the dynamical behaviour of the FOM in lower dimensions. Therefore, in the next section, we investigate ways to find such manifolds.

6. Physics-constrained ROMs

In this study, to build a closure model in lower dimensions, we explicitly derive the overall structures of manifolds in Hilbert space from the physical governing equations, and then shape them via a training dataset. This strategy is called a physics-constrained data-driven approach, and can result in accurate predictions of the closure model with less training data. Hence, we will leverage this approach to address downsides of conventional approaches, making optimization more robust with reduced requirements for the training dataset. To clarify what we seek when building a closure model, some essential points in this approach will be explained. If a closure model is represented as \mathcal{F} with a solution $\tilde{\mathbf{q}}$, then an ideal model should guarantee four criteria, represented as follows:

- (i) accurate prediction of the solution, $\tilde{\mathbf{q}} \mapsto \mathbf{u}$;
- (ii) accurate prediction of the dynamical response, $\mathcal{F} \mapsto \mathbf{f}$;
- (iii) strong numerical stability, $\|\mathcal{F}\| \leq \epsilon$, where ϵ is a bounded value;
- (iv) representation of the derivatives in lower dimensions, $\nabla \cdot \mathcal{F} \cong \nabla \cdot \mathbf{f}$.

The first and second criteria denote that the dimensionality reduction should not influence the prediction accuracy in the closure model over a finite time interval \mathbb{T} . This accuracy can be improved by adding an adequate number of subspaces (i.e. POD modes) to the ROB. In the third criterion, having a rich training dataset can improve prediction

accuracy in the closure model, but does not guarantee stability. The Galerkin projection is represented as a continuous form of minimization in the closure model (Carlberg *et al.* 2017). This projection becomes highly unstable for nonlinear dynamical systems (i.e. in high-Reynolds-number flows). The reason is that small flow structures play an essential role in the subsequent flow evolution (Nair *et al.* 2019). To overcome this issue, other implicit projection methods, such as LSPG, can be employed to build a stable closure model (Carlberg *et al.* 2011, 2017; Parish, Wentland & Duraisamy 2020). In the fourth criterion, derivatives of the closure model should closely approximate those of the FOM in lower dimensions. Most ROMs (linear, autoencoder or data-driven closure models) are only designed to approximate the state vector over $t \in \mathbb{T}$. In other words, there is no general model that represents the Jacobian of the ROM. Therefore, conventional ROMs may not be suitable for sensitivity analysis. Besides accurate prediction of the state vector, ROMs should project the most prominent features of the Jacobian and related derivatives of the FOM (i.e. $\partial \mathbf{r}^{(n)} / \partial \mathbf{u}^{(n)}$ and $\partial \mathbf{r}^{(n)} / \partial \mathcal{S}$) into lower dimensions.

In this study, the proposed algorithm relies on dimensionality reduction and dynamic prediction of a large-scale system using a ROM. Having a well-defined closure model for the ROM, sensitivity analysis of the FOM can then also be projected into lower dimensions. Figure 2 depicts the schematic of the proposed approach to build a ROM for approximating the sensitivity solution. The state vector \mathbf{u} from higher dimensions, \mathcal{P} , is projected into lower dimensions, \mathcal{H}_1 , via a projection function \mathbb{P} , where \mathcal{H}_1 represents a space embedding the dynamical system in (5.2a,b). In the next step, the closure model is built for the dynamical system via solving $\mathcal{R}(\bar{\mathbf{u}} + \tilde{\Phi} \tilde{\mathbf{q}}, t, \mathcal{S}) = 0$ for $\tilde{\mathbf{q}}$. These generalized coordinates, $\tilde{\mathbf{q}}$, evolve in \mathcal{H}_1 with time, and will be used in the next steps to build the Jacobian and other related matrices. If the model architecture and training procedure are fulfilled properly, the generalized coordinates typically will appear as a set of correlations. Subsequently, the steady-state forward sensitivity function is solved for \mathbf{v}_{ss} , which is in physical space \mathcal{P} , over a limited set of time intervals. The collected sensitivities are used to project these sensitivity solutions from \mathcal{P} into \mathcal{H}_2 via a projection function \mathbb{P}' . Note that \mathcal{H}_2 is a space adjusted to the sensitivity function in (5.16), and the evolution of the sensitivity solution lies into this space. In the next step, $(d\mathcal{R}/d\mathcal{S})(\bar{\mathbf{u}} + \tilde{\Phi} \tilde{\mathbf{q}}, t, \mathcal{S}) = 0$ is solved for $\tilde{\mathbf{h}}$, which is the sensitivity solution in \mathcal{H}_2 . Note that $\tilde{\mathbf{q}}$ and $\tilde{\mathbf{h}}$ could have different dimensions. Finally, the sensitivity solutions in \mathcal{H}_2 can be lifted back from lower to higher dimensions through \mathbb{P}'' to reconstruct the solution of the sensitivity in \mathcal{P} . An important note is that the ROM should be able to reconstruct the full state vector over \mathbb{T} . Also, the proposed model receives only a limited set of the sensitivities, sampled sporadically in time. Then, it predicts the full solution of the sensitivity for all time intervals in \mathbb{T} . In order to build a ROM, we need to use some portion of data obtained from the FOM. The data collected from this FOM are usually vectors of the state, represented as a set of snapshots. It is worth mentioning that the time period considered for each simulation, $T \in \mathbb{T}$, is a range in which we capture snapshots and use them to build the ROM. Therefore, this period needs to be long enough to obtain most of the evolutionary behaviour of the dynamical system. Otherwise, some portion of the dynamics will be missed in the ROM, leading to a spurious model. Hence, in this study, we selected T to be long enough with adequate time intervals such that the collection of snapshots can represent most of the flow features in this period.

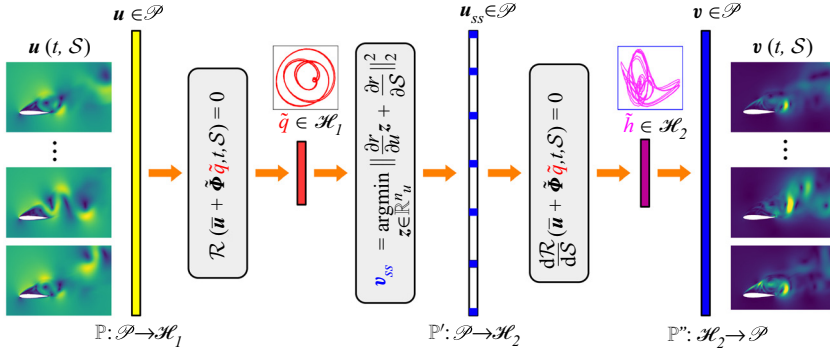


Figure 2. Schematic of the physics-constrained data-driven approach. The physics-constrained part uses information from the governing equations to apply strict constraints on the shape of manifolds in Hilbert space. Therefore, the overall structures of these manifolds are known, but their scale and details are unknown. The data-driven part helps fill this gap to find the shapes of these manifolds in Hilbert space.

7. ROB

Model reduction is usually based on the construction of low-dimensional subspaces in \mathcal{H} . Subsequently, a transformation function is applied to project the solution onto these low-dimensional subspaces. For the compressible flow problems considered here we build snapshots using the conserved variables $\mathbf{u} = [\rho, \rho u_i, \rho e]$, where ρ is density, u_i the velocity component in the i th direction and e is energy. Additionally, other quantities in the form of vectors can enrich the trial basis function in \mathcal{H} . These quantities can be solutions of the sensitivity, residuals or fluxes (Zahr & Farhat 2015). To define the trial basis function, first build a ROB, since the trial basis function is usually defined as a set of different ROB. As mentioned previously, having a well-trained ROB helps produce a robust trial basis function that can work in off-design conditions. Hence, the performance of a ROM with respect to any perturbation in \mathcal{D} will be ensured. We may execute several FOMs for a different set of design parameters, $\mathcal{S}_i = \{\mathcal{S}_1, \mathcal{S}_2, \dots, \mathcal{S}_{n_d}\} \subset \mathcal{D}$, where n_d indicates the total number of FOMs. Later, snapshots for each FOM are assembled according to

$$U(\mathcal{S}_i) = \begin{bmatrix} \mathbf{u}^{(1)}(\mathcal{S}_i) - \bar{\mathbf{u}} & \mathbf{u}^{(2)}(\mathcal{S}_i) - \bar{\mathbf{u}} & \dots & \mathbf{u}^{(m_u)}(\mathcal{S}_i) - \bar{\mathbf{u}} \end{bmatrix} \in \mathbb{R}^{n_u \times m_u}, \quad (7.1)$$

where m_u denotes the last snapshot. After building matrices using (7.1), we then collect them in a unified matrix

$$X_{state} = [U(\mathcal{S}_1) \quad U(\mathcal{S}_2) \quad \dots \quad U(\mathcal{S}_{n_d})], \quad (7.2)$$

where X_{state} is the dataset that will be used to train the ROB. Subsequently, each ROB is built via a set of POD modes

$$\tilde{\Phi}_u = \text{POD}(X_{state}), \quad \text{and} \quad U(\mathcal{S}_i) = \tilde{\Phi}_u Q_u(\mathcal{S}_i), \quad i = 1, 2, \dots, n_d, \quad (7.3a,b)$$

where $\tilde{\Phi}_u \in \mathbb{R}^{n_u \times r_{state}}$, and r_{state} shows the rank of the ROB for the state vector. Moreover, POD denotes a function that returns the orthogonal modes (i.e. POD modes). This function can be developed using snapshot-POD or singular value decomposition.

Additionally, $Q(\mathcal{S}_i)$ shows the generalized coordinates for each set of design parameters expressed as

$$Q_u(\mathcal{S}_i) = \begin{bmatrix} \tilde{q}_i^{(1)} & \tilde{q}_i^{(2)} & \dots & \tilde{q}_i^{(m_u)} \end{bmatrix} \in \mathbb{R}^{r_{state} \times m_u}. \quad (7.4)$$

To improve the trial basis function, it is also possible to train another ROB with the sensitivity dataset. The idea is that the sensitivity function is solved for a different set of design variables, $\mathcal{S}_i \subset [\mathcal{S}_1, \mathcal{S}_2, \dots, \mathcal{S}_{n_d}]$. Since the sensitivity function is usually unstable, we solve the steady-state sensitivity function for samples sporadically selected over different time intervals. Although steady-state solutions of the sensitivity, $\mathbf{v}_{ss} \in \mathbb{R}^{n_u}$, differ from unsteady ones, $\mathbf{v} \in \mathbb{R}^{n_u}$, they provide useful information about relevant structures in the dynamical behaviour of the forward sensitivity function. Therefore, we can use these results to approximate manifolds in Hilbert space to be able to solve this function in time. Therefore, snapshots of the solution of the sensitivity for all samples are defined as

$$V_{ss}(\mathcal{S}_i) = \begin{bmatrix} \mathbf{v}_{ss}^{(\mathbb{N}(1))}(\mathcal{S}_i) - \bar{\mathbf{v}}_{ss} & \mathbf{v}_{ss}^{(\mathbb{N}(2))}(\mathcal{S}_i) - \bar{\mathbf{v}}_{ss} & \dots & \mathbf{v}_{ss}^{(\mathbb{N}(m_p))}(\mathcal{S}_i) - \bar{\mathbf{v}}_{ss} \end{bmatrix} \in \mathbb{R}^{n_u \times m_p}, \quad (7.5)$$

where $\mathbb{N}(n') = \{\xi^{(1)}, \dots, \xi^{(m_p)}\}$ is the list of sampling time intervals, and m_p is the last sample for each case. Additionally, n' indicates an index for timing. Therefore, a dataset for all samples is represented as

$$X_{sens} = [V_{ss}(\mathcal{S}_1) \quad V_{ss}(\mathcal{S}_2) \quad \dots \quad V_{ss}(\mathcal{S}_{n_d})] \quad (7.6)$$

and, subsequently, the ROB for the solution of the sensitivity can be given by

$$\tilde{\Phi}_v = \text{POD}(X_{sens}), \quad \text{and} \quad V_{ss}(\mathcal{S}_i) = \tilde{\Phi}_v Q_v(\mathcal{S}_i), \quad i = 1, 2, \dots, n_d. \quad (7.7a,b)$$

Please note that the ROB of the solution of the sensitivity could have a different rank, $r_{sens} \in \mathbb{R}_+$, such that $\tilde{\Phi}_v \in \mathbb{R}^{n_u \times r_{sens}}$. Therefore, the generalized coordinates for the solutions of the sensitivity can be written as

$$Q_v = \begin{bmatrix} \tilde{h}^{(\mathbb{N}(1))} & \tilde{h}^{(\mathbb{N}(2))} & \dots & \tilde{h}^{(\mathbb{N}(m_p))} \end{bmatrix} \in \mathbb{R}^{r_{sens} \times m_p}. \quad (7.8)$$

Since the *POD* function is sensitive to the relative scale of the dataset in the snapshot matrix X , it cannot be used directly on the set of states and solutions of the sensitivity at the same time, $\text{POD}([X_{state}, X_{sens}])$ (Carlberg & Farhat 2011). To resolve this issue, a Gram–Schmidt-like procedure is used (Zahr & Farhat 2015) in which the ROB for state vectors and solutions of the sensitivity are a subset of the trial basis function

$$\Phi = [\tilde{\Phi}_u \quad \tilde{\Phi}_v] \in \mathbb{R}^{n_u \times r}, \quad (7.9)$$

where $r = r_{state} + r_{sens}$. It has been shown that building a trial basis function like (7.9) guarantees the minimum-residual property for the ROM without introducing excessive error (Zahr & Farhat 2015). Additionally, this type of trial basis function has been shown to be robust against perturbations in design parameters (Zahr & Farhat 2015). It is worth mentioning that the method for building the ROB in (7.6) and (7.7a,b) will remain efficient if all design variables have the same topology. For example, if all design variables

are coordinates of control points, this method can be useful. However, adding another variable with a different topology (i.e. angle of attack) may change the ROB significantly. Therefore, we recommend building the ROB as

$$\tilde{\Phi}_v = [POD(V_{ss}(S_1)), POD(V_{ss}(S_2)), \dots, POD(V_{ss}(S_{n_d}))], \tag{7.10}$$

where POD modes for each set of sensitivity snapshots are built separately and, consequently, they will be embedded into one unified matrix, $\tilde{\Phi}_v$. In (7.10), the POD modes are rearranged based on their eigenvalues.

In general, besides the aforementioned procedure, we can also use the following method if a specific application is required:

$$\tilde{\Phi} = \begin{cases} \tilde{\Phi}_u, & \text{if reconstruction of the state is required,} \\ \tilde{\Phi}_v, & \text{if forward sensitivity analysis is required.} \end{cases} \tag{7.11}$$

According to (7.11), the trial basis function could be the same as the ROB. If we use the ROM to approximate the state vector of the FOM, then $\tilde{\Phi}_u$ can be chosen since it still ensures the optimality condition over \mathbb{T} . Additionally, if we want to consider sensitivity analysis, then $\tilde{\Phi}_v$ can be used to obtain the sensitivity function of the closure model.

8. Application

In this section, the numerical model developed for the ROM is used for unsteady aerodynamic shape optimization. The objective is drag minimization of an NACA 0012 airfoil at $Re = 1000$ and $M_\infty = 0.1$. This is performed with multiple constraints to ensure the optimization is supervised in \mathcal{D} . Two different effective angles of attack, α_{eff} , are selected to investigate the performance of the optimization procedure for prestall and post-stall conditions. In this study, $\alpha_{eff} = 8^\circ$ and $\alpha_{eff} = 25^\circ$ correspond to moderate (prestall) and massive (post stall) flow separation, respectively. We deliberately chose these conditions, where the Navier–Stokes equations show strong nonlinear dynamics, to demonstrate the robustness of the optimization approach proposed in this study. The ROM-constrained minimization in continuous form can be written as

$$\left. \begin{aligned} &\text{minimize}_{\tilde{q} \in \mathbb{R}^r, S \in \mathcal{D}} \quad \bar{J} = \frac{1}{T} \int_0^T \beta (C_D)^2 dt + \frac{1}{T} \int_0^T (C_L - \bar{C}_{L,target})^2 dt, \\ &\text{subject to} \quad \left. \begin{aligned} &\mathcal{R}(\bar{u} + \tilde{\Phi}\tilde{q}, t, S) = 0, \\ &\frac{d\mathcal{R}}{dS}(\bar{u} + \tilde{\Phi}\tilde{q}, t, S) = 0, \\ &h = \operatorname{argmin}_{z \in \mathbb{R}^r} \left\| \frac{\partial(\bar{u} + \tilde{\Phi}\tilde{q})}{\partial S} - \bar{v} - \tilde{\Phi}z \right\|_2^2, \\ &C_{geom.}(S) \leq 0, \\ &B_l \leq C_{bound}(S) \leq B_u, \end{aligned} \right\} \tag{8.1} \end{aligned} \right.$$

where $C_L = \text{Lift}/(0.5\rho c u_\infty^2)$ and $C_D = \text{Drag}/(0.5\rho c u_\infty^2)$ are the lift and drag coefficients, respectively, ρ is the fluid density, u_∞ is the free stream velocity and c is the airfoil

chord length. Moreover, $C_{geom.}(\mathcal{S})$ and $C_{bound}(\mathcal{S})$ correspond to geometrical constraints and bounds included in the constraint function $C(\mathcal{S})$. Additionally, B_u and B_l are the upper and lower limits of C_{bound} , respectively, and β is a weighting factor. This weighting is important when using a united objective function. The second term of the objective function guides the optimization toward a target lift coefficient, $\bar{C}_{L,target}$. The fully discrete form of (8.1) is given by

$$\begin{aligned}
 & \underset{\tilde{\mathbf{q}} \in \mathbb{R}^r, \mathcal{S} \in \mathcal{D}}{\text{minimize}} \quad \bar{\mathcal{J}} = \frac{1}{m_u} \sum_{n=0}^{m_u} \beta \left(C_D(\bar{\mathbf{u}} + \tilde{\Phi} \tilde{\mathbf{q}}^{(n)}, t^{(n)}, \mathcal{S}) \right)^2 \\
 & \quad + \frac{1}{m_u} \sum_{n=0}^{m_u} \left(C_L(\bar{\mathbf{u}} + \tilde{\Phi} \tilde{\mathbf{q}}^{(n)}, t^{(n)}, \mathcal{S}) - \bar{C}_{L,target} \right)^2, \\
 & \text{subject to} \\
 & \mathcal{R}^{(n)}(\bar{\mathbf{u}} + \tilde{\Phi} \tilde{\mathbf{q}}^{(n)}, t^{(n)}, \mathcal{S}) = 0, \\
 & \frac{d\mathcal{R}^{(n)}}{d\mathcal{S}}(\bar{\mathbf{u}} + \tilde{\Phi} \tilde{\mathbf{q}}^{(n)}, t^{(n)}, \mathcal{S}) = 0, \\
 & \mathbf{v}_{ss}^{(\mathbb{N}(n'))} = \underset{\mathbf{z} \in \mathbb{R}^{n_u}}{\text{argmin}} \left\| \frac{\partial \mathbf{r}^{(\mathbb{N}(n'))}}{\partial \mathbf{u}^{(\mathbb{N}(n'))}} \mathbf{z} + \frac{\partial \mathbf{r}^{(\mathbb{N}(n'))}}{\partial \mathcal{S}} \right\|_2^2, \\
 & r_{state} = \underset{\mathbf{z} \in \mathbb{R}_{+\leq n_u}}{\text{argmin}} \left(\beta \mathbf{z} + \left\| \sum_{n=0}^{m_u} \mathbf{u}^{(n)} - \bar{\mathbf{u}} - \tilde{\Phi}_u \tilde{\mathbf{q}}^{(n)} \right\|_2^2 \right), \\
 & r_{sens} = \underset{\mathbf{z} \in \mathbb{R}_{+\leq n_u}}{\text{argmin}} \left(\beta \mathbf{z} + \left\| \sum_{n'=1}^{m_p} \mathbf{v}_{ss}^{(\mathbb{N}(n'))} - \bar{\mathbf{v}}_{ss} - \tilde{\Phi}_v \tilde{\mathbf{h}}^{(\mathbb{N}(n'))} \right\|_2^2 \right), \\
 & C_{geom.}(\mathcal{S}) \leq 0, \\
 & B_l \leq C_{bound}(\mathcal{S}) \leq B_u,
 \end{aligned} \tag{8.2}$$

where argmin in (8.1) is expanded to show the training process and the rank selection criteria in (8.2). The shape of the airfoil is changed using ‘B-spline’ shape functions, using control points shown in figure 3. These control points define smooth perturbations to the suction and pressure surfaces of the airfoil. Additionally, three fixed points at the leading edge, and another at the trailing edge, are used as constraints to avoid defective geometries in these regions. These fixed points are also responsible for keeping the angle of attack constant. As mentioned above, all of these constraints are included in $C(\mathcal{S})$.

8.1. Computational set-up

A two-dimensional structured computational domain with a C-topology is used. To solve a compressible flow field, an in-house CFD software, the high-order unstructured solver (HORUS) version 0.2.0, is used. The spatial discretization in HORUS uses the flux reconstruction approach (Huynh 2007), and we set the solution polynomial degree to $p_s = 2$, which recovers third-order accuracy. The second-order BDF (BDF2) is used for temporal discretization. The Prandtl number and Mach number are set to $Pr = 0.72$ and

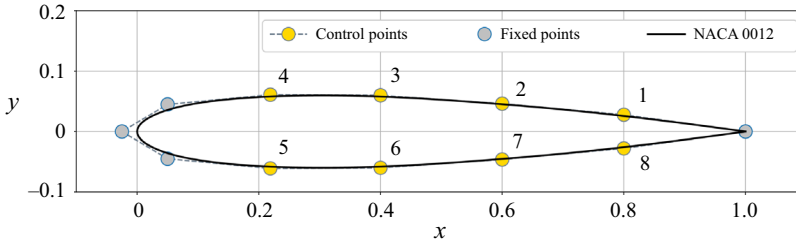


Figure 3. Geometrical definition of the airfoil using control and fixed points.

$M_\infty = 0.1$, respectively. Since the governing equations are solved implicitly, the time step is chosen to be $\Delta t = 0.05t^*$, where $t^* = c/u_\infty$ is the convective time.

The number of elements for the initial mesh is approximately 3.7×10^3 . With a solution polynomial degree $p_s = 2$ the total number of solution points is $n_u = 1.35 \times 10^5$. Figure 4 shows a comparison of time-averaged lift and drag coefficients at several angles of attack, showing good agreement with available reference data. The baseline NACA 0012 airfoil was then modified using sequential least squares programming as an optimizer from the Scipy package (Virtanen 2020). Additionally, the singular value decomposition, as an eigenvalue problem is solved via SLEPc (Hernandez, Roman & Vidal 2005) in parallel, and PETSc (Dalcin *et al.* 2011; Balay 2020), compiled with OpenMPI (Dalcín, Paz & Storti 2005), is used for parallelization. We simulated each FOM case until $40t^*$. It was observed that this period is sufficient to obtain statistically converged results. Data snapshots were then collected over a period of $T = 15t^*$. The number of subspaces (the rank of the ROB) is a crucial parameter that should be assessed before starting optimization. Having an inappropriate rank may cause a bias in the ROM, creating a poor dynamical model. Therefore, a ROB was built for the reference case (NACA 0012 airfoil at $Re = 1000$). Then the arrangement and accumulation of singular values were analysed according to (8.2). It is observed that if the rank of the ROM is selected to be $r_{state} = 50$, the ROB contains most of the interpretive information of flow structures. Additionally, $m_p = 20$ samples for each control point, P(1) to P(8), are collected to develop the ROB of the sensitivity function with the rank of $r_{sens} = 80$. These derivatives are computed with a second-order central finite difference scheme. The sensitivity function is also discretized via the first-order BDF (BDF1) scheme for sensitivity analysis. It is worth mentioning that r_{sens} should be modified since some modes in $\tilde{\Phi}_v$ might be unnecessary, leading to an increase in the bias of the sensitivity. On the other hand, $\tilde{\Phi}_v$ should have a high enough rank to minimize the detrimental effect of variance on the results. Therefore, the residual of the sensitivity function was monitored to ensure it was lower than 10^{-4} . Using this approach, we will now consider shape optimization at two different angles of attack.

8.2. Shape optimization of an NACA0012 at $\alpha_{eff} = 8^\circ$

8.2.1. Set-up

An NACA 0012 airfoil at $\alpha_{eff} = 8^\circ$ is in the prestall condition with $\bar{C}_L = 0.323$ and $\bar{C}_D = 0.14$. For drag minimization we select $\bar{C}_{L,target} = 0.32$ and $\beta = 10$ in (8.2). We set the effective angle of attack as a variable that can be changed by unlocking the fixed points at the leading edge. In this case, $C(S)$ includes 10 geometrical constraints with 16 bounds. Additionally, one extra constraint is considered to limit \bar{C}_L . The minimization is then advanced for 10 design iterations.

Unsteady aerodynamic optimization

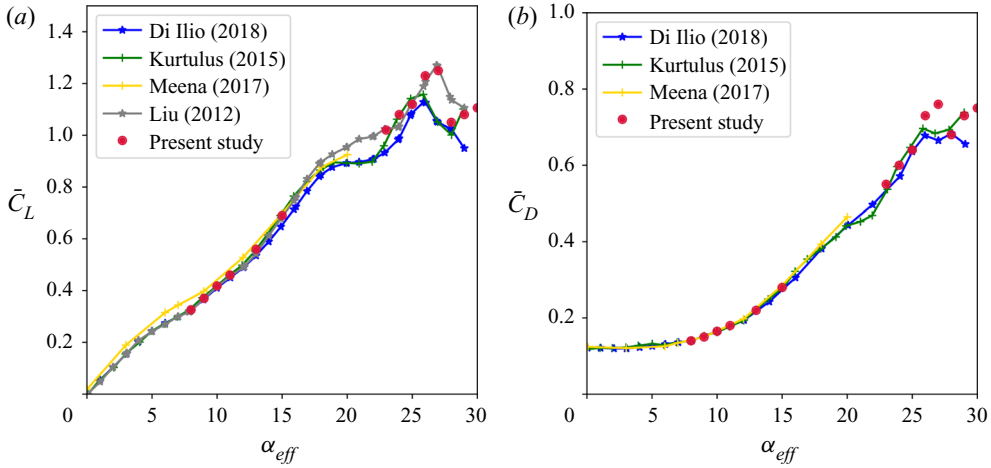


Figure 4. Comparison of present numerical simulations with reference data (Liu *et al.* 2012; Kurtulus 2015; Meena, Taira & Asai 2017; Di Ilio *et al.* 2018): (a) time-averaged lift coefficient; (b) time-averaged drag coefficient.

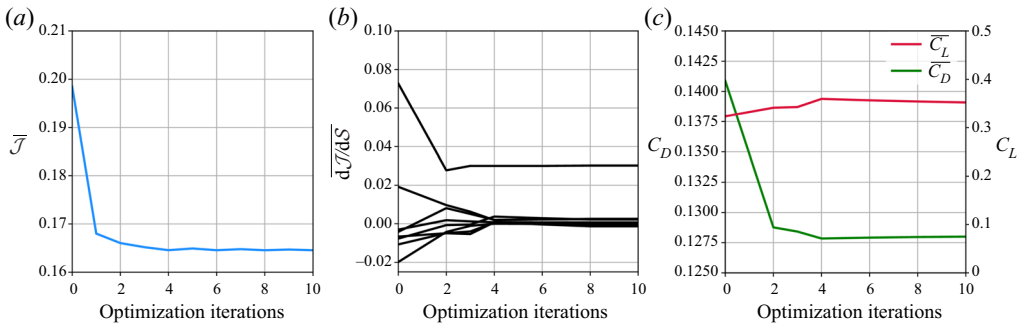


Figure 5. Optimization progress in the shape optimization of the airfoil at $Re = 1000$ and $\alpha_{eff} = 8^\circ$: (a) objective function; (b) gradients and (c) lift and drag coefficients.

8.2.2. Optimization results

Figure 5 shows the optimization results. The time-averaged objective function, $\bar{\mathcal{J}}$, continuously decreases until the fourth iteration. After that, the constraints slow down reduction of $\bar{\mathcal{J}}$. Moreover, time-averaged gradients of the objective function are displayed in figure 5(b). All gradients converge to zero, except one that remains constant at approximately 0.03. This cannot be driven lower due to the imposed constraints. Figure 5(c) illustrates how the time-averaged lift and drag coefficients are changed during optimization. The drag coefficient is reduced by approximately 9.35%, while producing 9% higher lift than the reference airfoil.

To compare the reference and optimized designs in terms of geometry and flow characteristics, figure 6 is provided. The pressure changes rapidly at the leading edge of the optimized airfoil. These variations on the suction side produce additional increased lift, while the rest of the optimized airfoil has no notable contribution to lift generation. Furthermore, the optimized airfoil has a nose slightly lower than the reference airfoil, which influences the effective angle of attack. Interestingly, the optimized airfoil produces higher lift at lower effective angle of attack.

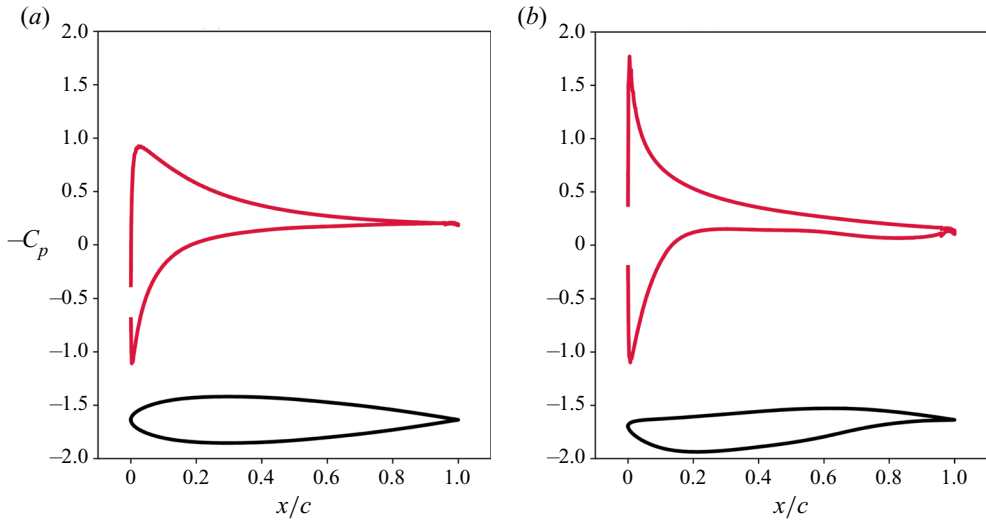


Figure 6. The geometry of the airfoil and the time-averaged pressure coefficient along the airfoil surface: (a) reference case and (b) optimized case.

8.3. Flow structures

The slender leading edge of the optimized airfoil has a notable role for the separation point on the suction side. As shown in figure 7(a), the separation point, where the flow detaches from the surface due to an adverse pressure gradient, is at $x \simeq 0.5c$. After the separation point, the flow confined on the suction side generates a large pressure difference and, subsequently, higher drag. This confined area is also directly related to the wake of the airfoil. However, the optimized airfoil's nose induces a sudden adverse pressure gradient near the leading edge, which leads to laminar separation bubble (LSB) formation, as shown in figure 7(b). Consequently, it leads to a delayed separation point located at approximately $x \sim 0.8c$. This delayed flow separation leads to a smaller wake behind the airfoil, which reduces drag.

8.4. Shape optimization of an NACA0012 at $\alpha_{eff} = 25^\circ$

8.4.1. Set-up

There are different reports on the stall point of an NACA 0012 at $Re = 1000$. Liu *et al.* (2012) reported that the stall angle of attack is 27° with $\bar{C}_L = 1.28$. Kurtulus (2015) also showed the stall lift coefficient as $\bar{C}_L = 1.25$ with a stall angle of 26° , and Di Ilio *et al.* (2018) detected the same stall angle, but with $\bar{C}_L = 1.1$. The present study shows that stall occurs at 27° with $\bar{C}_L = 1.25$. Irregular vortex shedding is expected due to massive flow separation on the airfoil's suction side. Our objective is to minimize the drag coefficient of the airfoil at $\alpha_{eff} = 25^\circ$. The lift and drag coefficients for the NACA 0012 at this angle of attack are $\bar{C}_L = 1.17$ and $\bar{C}_D = 0.66$, respectively.

To define the optimization problem, we selected $\beta = 2$ and $\bar{C}_{L,target} = 1.25$ to force the objective function to keep the lift coefficient relatively constant while reducing drag. Having a slightly higher $\bar{C}_{L,target}$ than $\bar{C}_L = 1.17$ was found to prevent the magnitude of the lift coefficient dropping significantly in early design iterations. Additionally, $C(S)$ includes 12 geometrical constraints and 16 bounds for the control points. Limiting the lift

Unsteady aerodynamic optimization

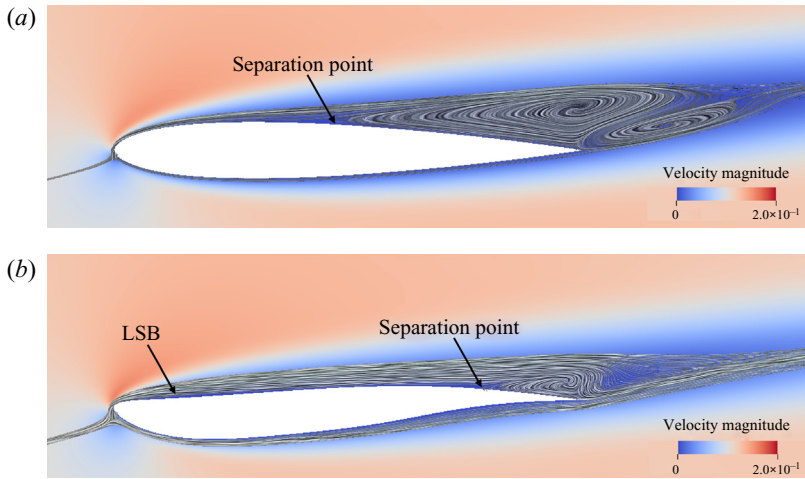


Figure 7. Velocity contours with superimposed streamlines for $Re = 1000$ and $\alpha_{eff} = 8^\circ$:
(a) reference and (b) optimized airfoils.

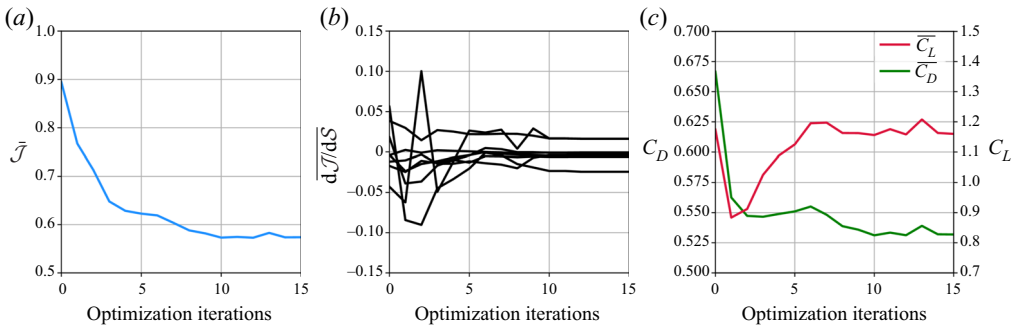


Figure 8. Optimization progress in the shape optimization of the airfoil at $Re = 1000$ and $\alpha_{eff} = 25^\circ$:
(a) objective function; (b) gradients and (c) lift and drag coefficients.

coefficient and the angle of attack are two other constraints. Therefore, this optimization problem contains 30 constraints in total.

8.4.2. Optimization results

To show how the design changes during optimization, figures 8 and 9 are provided. In figure 8(a), the time-averaged objective function, \bar{J} , begins to reduce continuously until the 10th optimization iteration. Later the objective function reduces slightly or oscillates when the design parameters are close to the local optimum point in the design space. As shown in figure 8(b), gradients of the objective function either plateau or converge to zero. According to the gradient-based formulation, gradients of the objective function ideally should be zero at the local/global optimum. However, bounding the control points with geometrical constraints may restrict gradients from reaching identically zero. Figure 8(c) also represents the variation of time-averaged lift and drag coefficients. The lift coefficient drops at the first optimization iteration and, later, it rises toward the target value $\bar{C}_{L,target}$. The drag coefficient diminishes significantly.

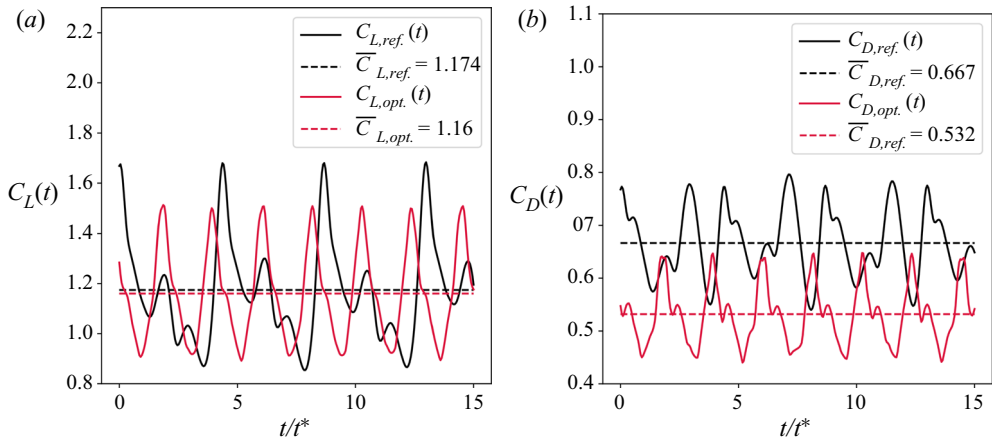


Figure 9. (a) Instantaneous lift and (b) drag coefficients for both reference and optimized cases.

Figure 9 depicts the lift and drag coefficients versus non-dimensional time t/t^* . From figure 9(a), the time-averaged lift coefficient reduces slightly from $\bar{C}_{L,ref.} = 1.17$ to $\bar{C}_{L,opt.} = 1.16$. However, a significant difference between the time-averaged drag coefficients yields an approximately 20% reduction in drag. Figure 10 shows the distribution of the time-averaged pressure coefficient along the airfoil surface, where the shaded area shows pressure variations over time. From figure 10(a), the pressure on the reference airfoil suction side fluctuates across the majority of the chord length, particularly near the trailing edge. However, pressure fluctuations for the optimized design are significantly reduced, as shown in figure 10(b). Because the intensity of large-scale coherent structures in the flow field tends to amplify pressure fluctuations on the suction side, this indicates that the optimized airfoil creates weaker wake vortices.

The spatial integral of the time-averaged pressure on the surface yields the time-average pressure force exerted on the airfoil. The majority of this pressure force originates in the vicinity of the leading edge. However, in the optimized case, the time-averaged pressure force near the trailing edge is significantly reduced. The pressure curves in this region ($x/c = 0.8 \sim 1$) are similar, resulting in minimal lift and drag production in this region. Despite generating significantly less lift near the trailing edge, the airfoil is still able to satisfy the lift coefficient constraint. This is achieved via the shear layer produced at the leading edge of the airfoil. As shown in figure 10(b), sharp variations in the pressure coefficient around the airfoil's leading edge arise from an intense, but relatively stable, shear layer. Furthermore, the airfoil's suction surface is relatively flat and tangent to the flow, leading to the enhanced lift. In contrast, the sharp difference of the pressure coefficient at the leading edge of the reference airfoil is less intense than the optimized one (figure 10a). The pressure variations on the suction side near the nose of the reference airfoil fluctuate with a large amplitude, indicating an unstable shear layer at the leading edge.

8.4.3. Flow structures

Here we will explore the behaviour of coherent structures in the wakes of the reference and optimized designs. Figures 11 and 12 show time histories of the lift and drag coefficients, and contours of vorticity at several instants during a shedding cycle. Green arrows show the

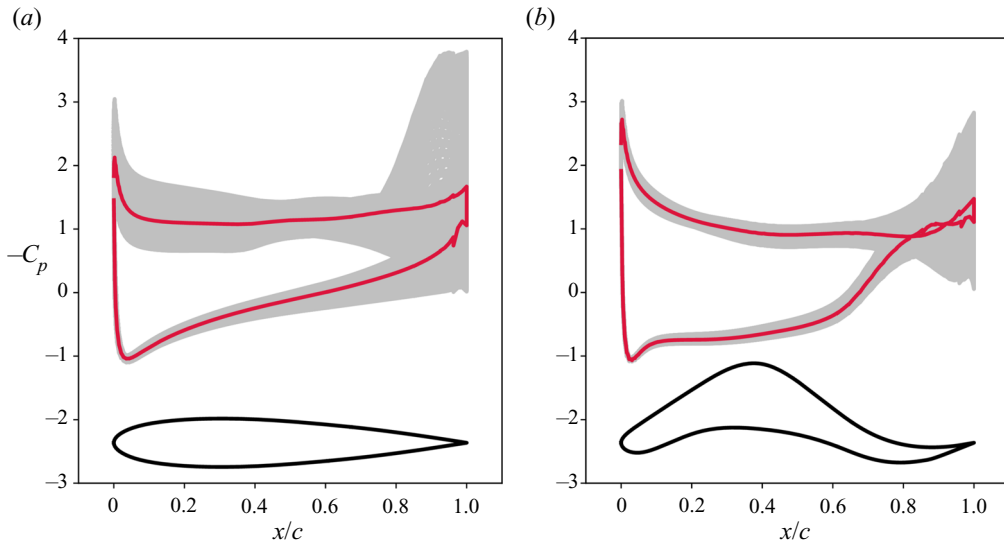


Figure 10. The geometry of the airfoil and the time-averaged pressure coefficient (red) along the airfoil surface. The shaded area also displays the variations of the pressure coefficient over time. (a) Reference case and (b) optimized case.

direction of the jet formed at the interface of two counter-rotating vortices. In frame ‘A’, the first trailing edge vortex (TEV), Γ_1^+ , leaves the airfoil surface and travels downstream. Moreover, the first leading edge vortex (LEV), Γ_1^- , increases in intensity and envelopes the suction surface of the airfoil. Hence, the pressure coefficient on the suction surface reaches a minimum, yielding high lift. The shedding of LEV and TEV manifests as a von Kármán street in the wake of the airfoil. In frame ‘B’, the aerodynamic forces decrease, since Γ_2^- is not strong enough to produce a notable low-pressure zone on the suction surface of the airfoil. Furthermore, Γ_2^+ does not allow Γ_2^- to grow. According to Kelvin’s circulation theorem, after Γ_1^- separates, the second TEV, Γ_2^+ , forms and gains energy by absorbing the kinetic energy coming from Γ_2^- , and the shear layer at the trailing edge of the airfoil. This energy transfer indicates that if two counter-rotating vortices are located beside each other, the low-energy vortex absorbs energy from the high-energy vortex through shear. In frame ‘C’, the pair vortex, Γ^* , has developed on the suction surface. This counter-rotating vortex reduces the connection of Γ_2^- with the airfoil surface. It is observed that Γ^* in the reference case plays an important role in the quasi-periodic behaviour of vortical structures in the wake. Different intensities and growth rates for Γ^* leads to different vortex shedding patterns, which influences the LEV and TEV dynamics. This is evident in time variations of the lift and drag coefficients.

Figure 12 shows different instants during one shedding cycle for the optimized design. In frame ‘A’, Γ_1^+ has detached from the airfoil, while Γ_1^- envelops the suction surface. When Γ_1^- reaches its maximum intensity, the lift is maximized. In frame ‘B’, a shear layer forms a LSB, which initiates Γ^* . As Γ^* increases in intensity, it ejects Γ_1^- away from the surface with the help of Γ_2^+ (as shown in frame ‘C’). It is observed that LEV and TEV enter a harmonic vortex shedding pattern for the optimized design, where the aerodynamic loads repeat cyclically. The flow field becomes more complicated in the post-stall region, where at least two shedding frequencies appear. Hence, we observe that the optimized

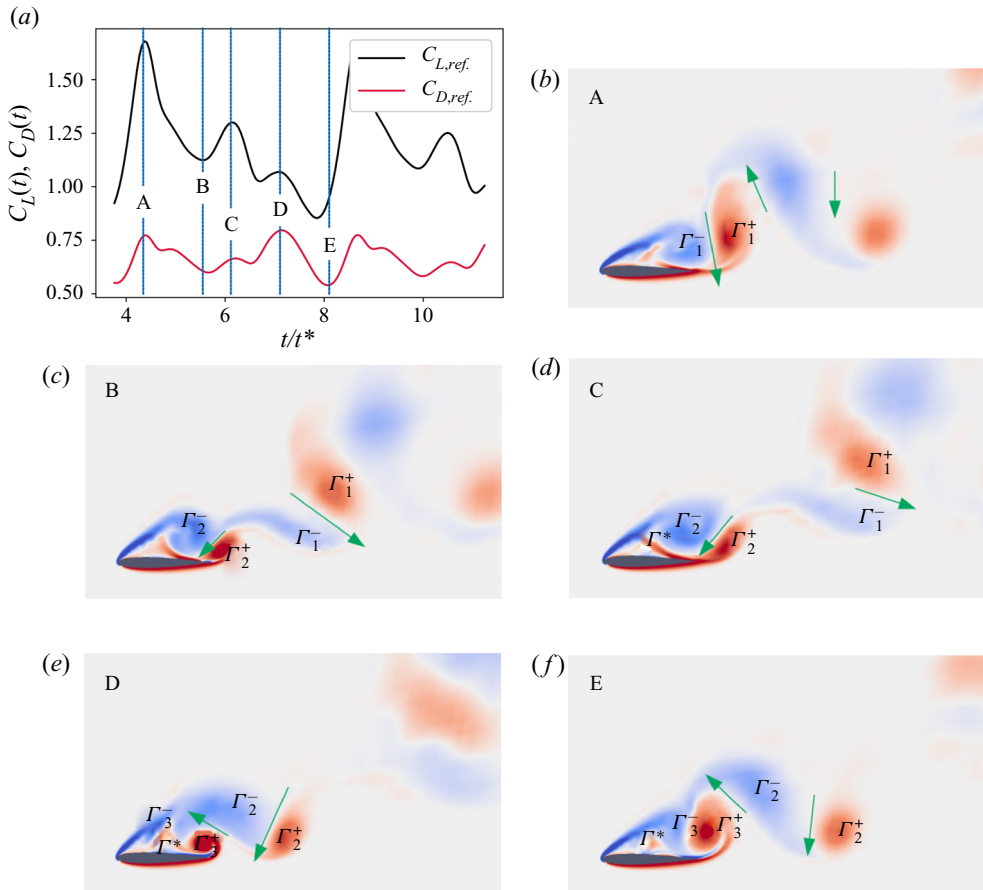


Figure 11. Flow structures around the reference airfoil.

design has a simpler shedding pattern, with harmonic interaction between the LEV and TEV.

To quantify the strength of these vortices, a control surface is placed around the airfoil over which an integral of $\nabla \times \mathbf{u}$ is computed, yielding circulation. The control surface boundaries were placed far enough away from the airfoil such that this circulation was not sensitive to them. Figure 13(a) shows the total circulation magnitude of the LEV over $15t^*$ for the reference case, Γ_{ref}^- , and the optimized case, Γ_{opt}^- . The strength of Γ_{ref}^- is notably higher than Γ_{opt}^- . Furthermore, the oscillation of Γ_{ref}^- is greater, while Γ_{opt}^- has smaller oscillations with nearly constant peaks. The same observation is made from figure 13(b) showing the strength of Γ_{ref}^+ and Γ_{opt}^+ . This indicates that the optimized airfoil produces a weaker LEV. This is because the optimized airfoil guides the flow over the surface more smoothly, which leads to simpler interaction between the LEV and TEV structures in the wake.

8.4.4. Sensitivity analysis

This section analyses the sensitivity of the state vector and the objective function with respect to the design parameters. According to (8.1), the total gradient of the objective

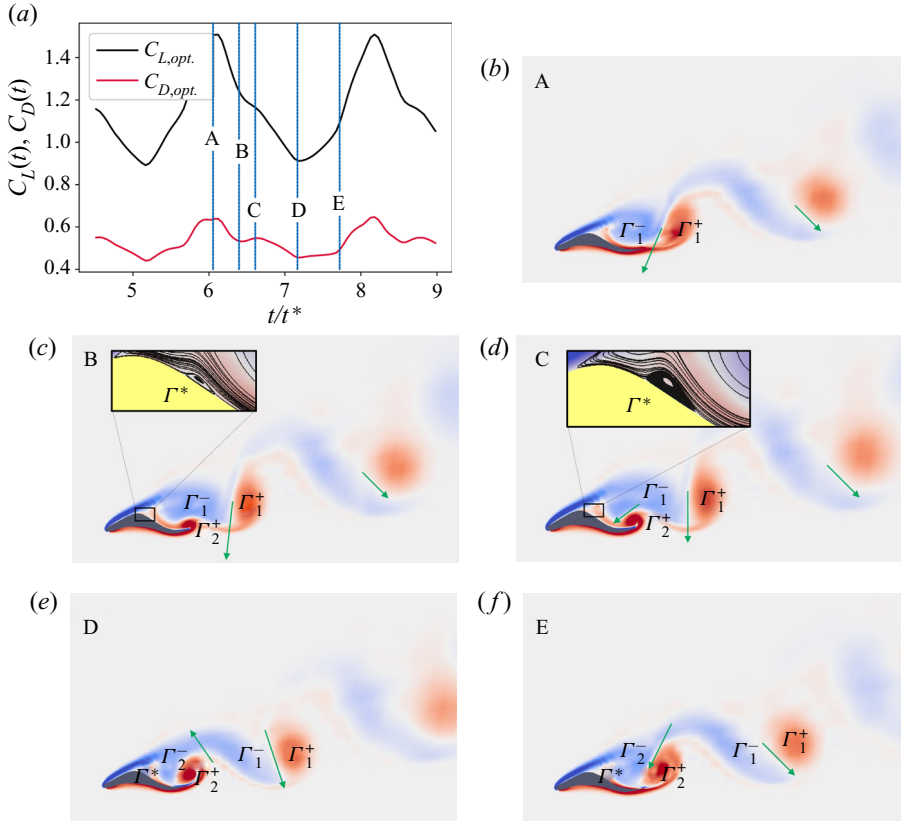


Figure 12. Flow structures around the optimized airfoil.

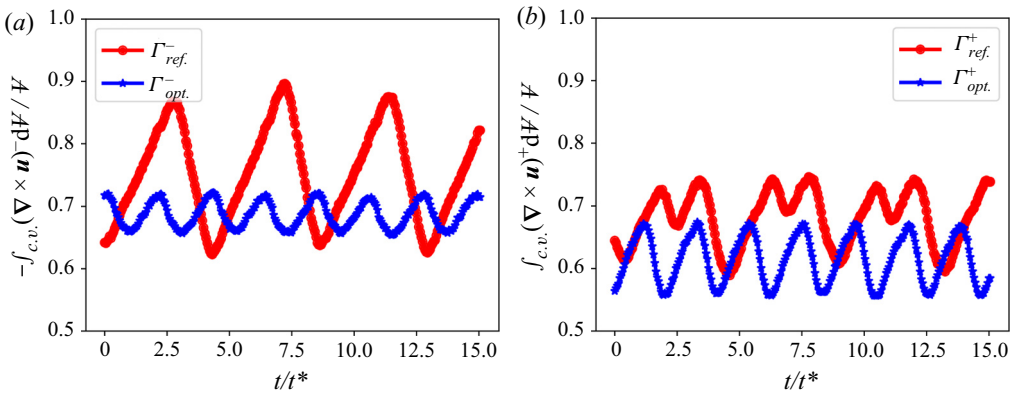


Figure 13. Time variation of the total strength of core vortices (i.e. LEV and TEV) over $15t^*$ for both reference and optimized cases: (a) instantaneous circulation generated by LEV; (b) instantaneous circulation generated by TEV.

function, $\bar{\mathcal{J}}$, can be written as

$$\frac{d\bar{\mathcal{J}}}{dS} = \frac{2}{T} \int_0^T \beta \frac{dC_D}{dS} C_D dt + \frac{2}{T} \int_0^T \frac{dC_L}{dS} (C_L - \bar{C}_{L,target}) dt, \quad (8.3)$$

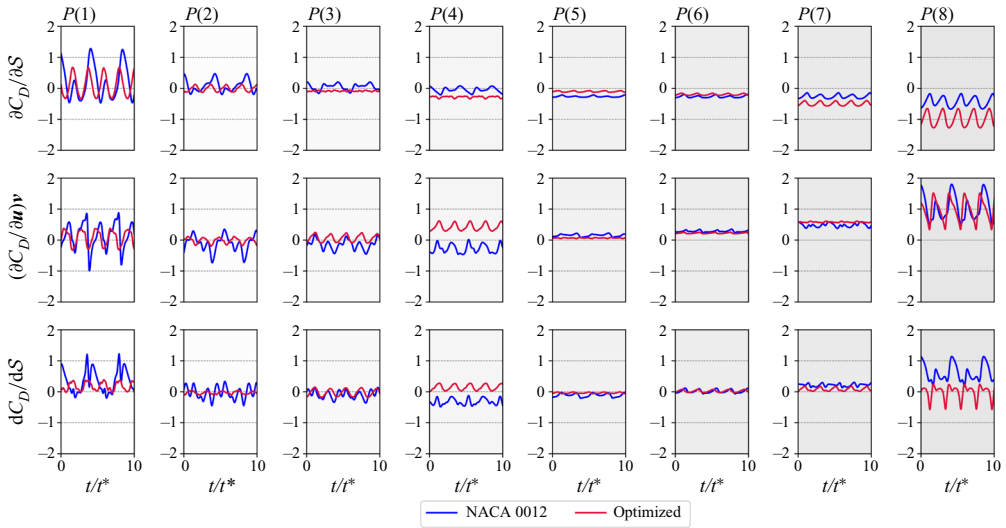


Figure 14. Instantaneous gradients of the drag coefficient (geometrical, state and total sensitivities) with respect to the control points.

where

$$\left\{ \begin{array}{l} \frac{dC_L}{dS} = \frac{\partial C_L}{\partial \mathbf{u}} \mathbf{v} + \frac{\partial C_L}{\partial S} \\ \frac{dC_D}{dS} = \frac{\partial C_D}{\partial \mathbf{u}} \mathbf{v} + \frac{\partial C_D}{\partial S} \end{array} \right. \quad (8.4)$$

and \mathbf{v} is approximated by $\mathbf{v} \approx \bar{\mathbf{v}} + \tilde{\Phi} \tilde{\mathbf{h}}$. Since the design objective is drag minimization, we will isolate this term. The total sensitivity of the drag coefficient, dC_D/dS , is dependent on the geometrical sensitivity, $\partial C_D/\partial S$, and the state sensitivity, $(\partial C_D/\partial S)\mathbf{v}$. The geometrical sensitivity is only dependent on the shape of the airfoil, independent of the flow field. Also, the state sensitivity is only dependent on the flow field, independent of perturbations in the airfoil geometry. Therefore, these two sensitivities will be explored separately.

Figure 14 displays the sensitivity of the drag coefficient with respect to control points (shown in figure 3), where each column corresponds to a control point. The first, second and third rows show the time variation of the geometrical sensitivity, state sensitivity and total sensitivity. From figure 10, the pressure coefficient has large fluctuations near the trailing edge, which suggests that the aerodynamic loads are highly sensitive here (P(1) and P(8)). This implies that the trailing edge has a significant influence on TEV formation, affecting the drag coefficient. Furthermore, the drag coefficient is sensitive to the suction surface of the airfoil (P(2) to P(4)), where sensitivities fluctuate with the flow. However, since the flow field on the pressure side of the airfoil is relatively steady, the sensitivities for P(5) and P(6) are also relatively steady. Moreover, the time variation of dC_D/dS reveals that the reference airfoil is more sensitive than the optimized design.

Figure 15 shows the primary mode of the sensitivity, which corresponds to the fluid momentum in x direction, $\tilde{\Phi}_x$. For the reference case, P(1) has a notable effect on the TEV and drag reduction. Also, P(1) affects a wide range of secondary structures in the wake. For the optimized case, P(1) has less sensitivity to the TEV. On the other hand, the optimized airfoil has higher sensitivity to the LEV. In general, for both the reference and optimized

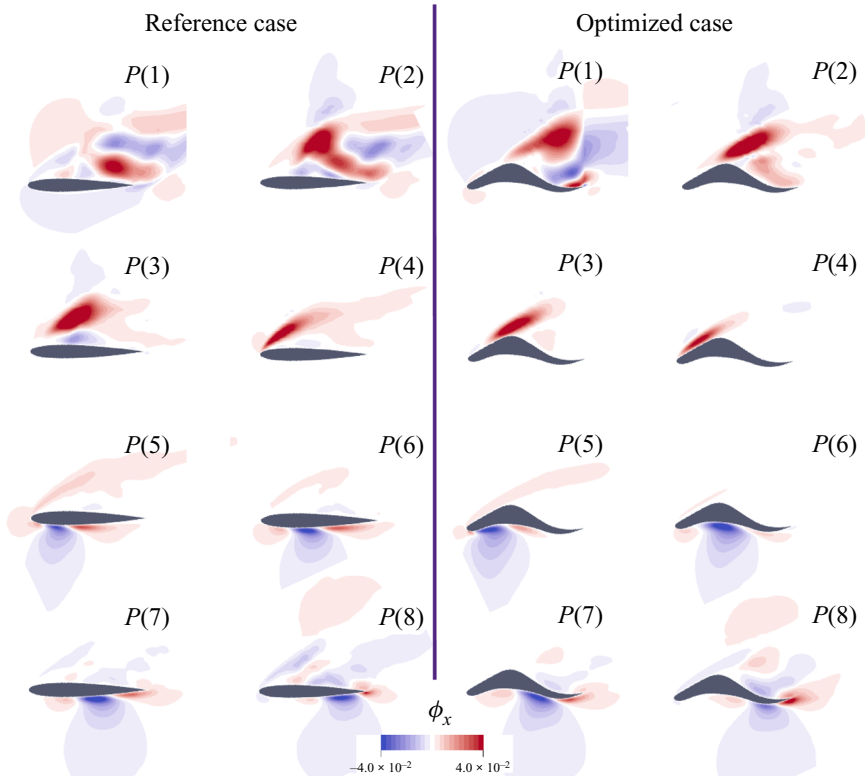


Figure 15. The primary modes of the solutions of the sensitivity for both reference and optimized cases.

cases, P(2) and P(3) control the shear layer at the leading edge, which directly changes the LEV and high-pressure differences at the suction surface of the airfoil. However, P(5) to P(7) indicate that the pressure surface of the airfoil does not significantly affect vortical structures in the wake of the airfoil. On the other hand, P(8) influences these vortical structures on the suction surface of the airfoil. The conclusion drawn from figure 15 is that control points in the vicinity of the trailing edge have a significant effect on drag minimization. On the other hand, control points adjacent to the leading edge have a notable influence on lift.

8.4.5. Dynamics

In the previous sections, we explained how optimization changes the airfoil shape and, hence, flow structures in the wake. Then, we focused on the behaviour of the sensitivity solution to understand each control point's effect on the flow field. Here, we consider the optimization procedure from a different perspective in Hilbert space. We transform the optimization problem from \mathcal{P} to \mathcal{H} , and then we identify the characteristics of the underlying dynamical system.

Figure 16 shows a discrete Fourier transform of the generalized coordinates. Figure 16(a) shows the Strouhal number distribution for 10 subsequent modes, indexed by their rank r . The Strouhal number is defined as $St = f_s l_c / u_\infty$, where f_s , and l_c are the frequency and characteristic length, taken here to be the wake width, respectively. Dominant Strouhal numbers are evident in modes with high spectrum magnitudes. For the reference case, several dominant shedding modes are observed. However, the optimized case has only

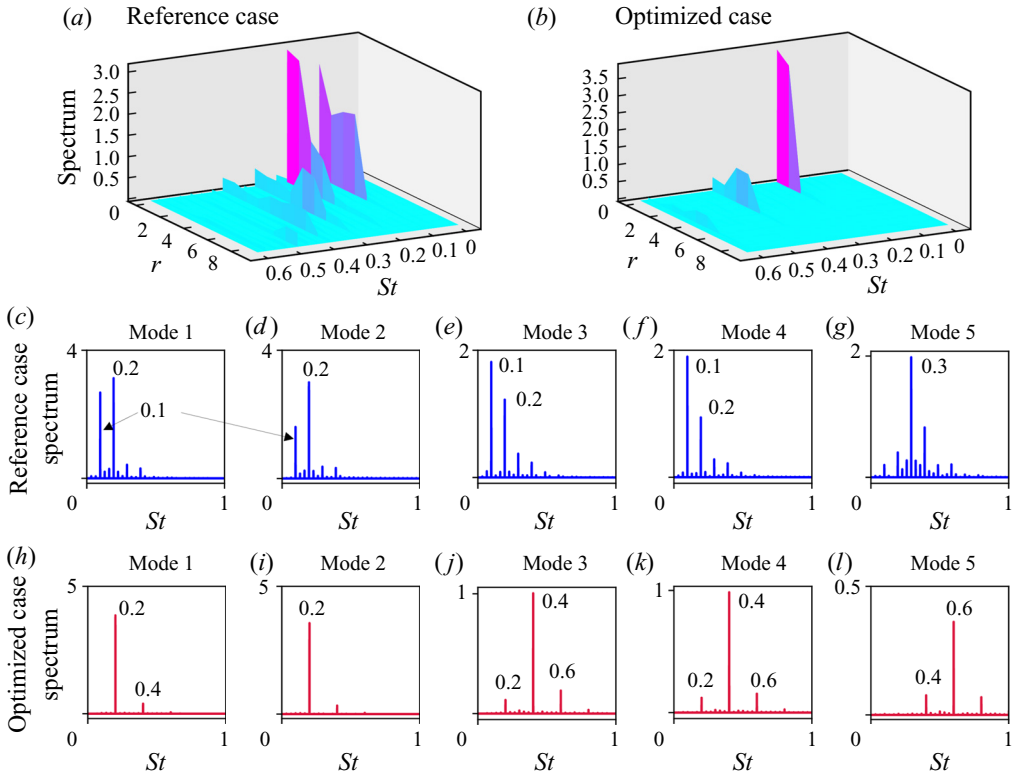


Figure 16. Variation of frequency for different modes in the reference and optimized cases.

three dominant modes, as shown in [figure 16\(b\)](#). This suggests that the wake is significantly simpler in nature for the optimized case. [Figures 16\(c\) to 16\(g\)](#) display the amplitude spectrum versus Strouhal number for the first five dominant modes for the reference case. Frequencies detected in the first four modes are $St_1 = 0.2$ and $St_2 = 0.1$. [Figures 16\(h\) and 16\(i\)](#) also show that the dominant Strouhal number is $St_1 = 0.2$ for the optimized case. [Yarusevych, Sullivan & Kawall \(2009\)](#) performed an extensive experiment on the wake of the reference airfoil, and the reported Strouhal number was $St \approx 0.2$ for an airfoil at $\alpha_{eff} = 10^\circ$. Additionally, similar observations can be found in the results of [Huang & Lin \(1995\)](#). They considered wake structures of an NACA 0012 airfoil, and mentioned that the Strouhal number of the dominant vortex shedding for angles of attack higher than $\alpha_{eff} = 25^\circ$ remains approximately constant with a value of $St \approx 0.2$.

Correlations can be defined as interactions between the generalized coordinates, $q_i \in \mathbf{q} = [q_1, q_2, \dots, q_r]^T$, to understand the dynamical behaviour of the system. [Figure 17](#) shows correlations for the first 10 modes in the reference and optimized cases. These 10 modes correspond to five ‘pair’ modes in the dynamical system. The pair modes are two subsequent modes that show the same, or similar, dynamical behaviour with a certain offset. [Figures 17\(a\), 17\(c\), 17\(e\), 17\(g\) and 17\(i\)](#) show correlations of ‘pair’ modes. The remaining figures show correlations between ‘non-pair’ modes. Each correlation loop, specifically in ‘pair’ modes, belongs to a specific vortical structure. [Figure 17\(a\)](#) shows that the correlation of the reference case has two loops, which indicates that two different dominant patterns in the vortex shedding occur one after another. However, the optimized case has only one loop indicating one dominant pattern occurs cyclically.

Unsteady aerodynamic optimization

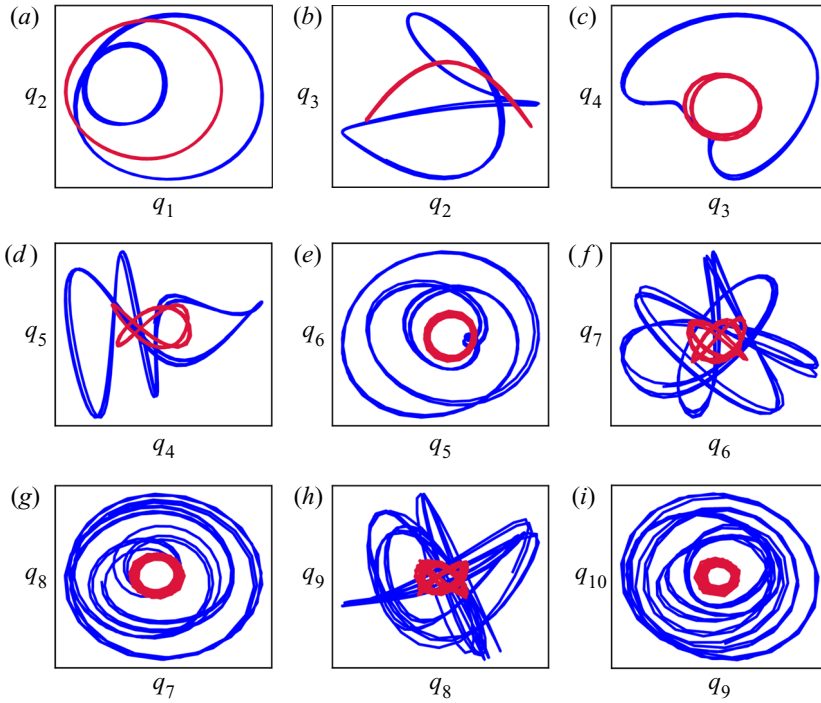


Figure 17. Correlations between modes: reference (blue) and optimized (red) cases.

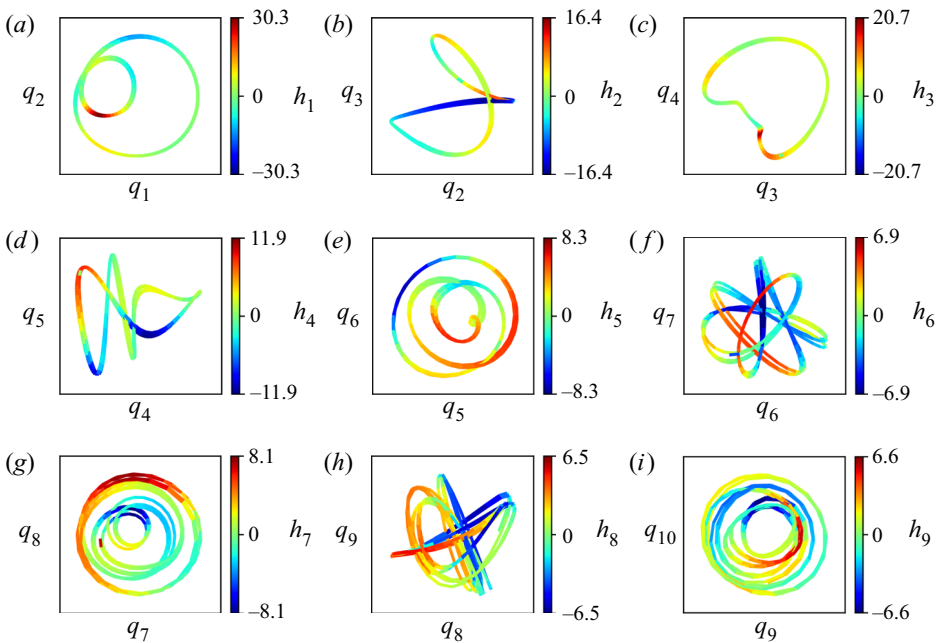


Figure 18. Correlations of modes coloured by sensitivity magnitude for the reference case.

High rank correlations also contain more loops, which indicates that each dominant pattern has different forms that repeat harmonically in different shedding cycles. It is worth pointing out that as the dynamical system experiences higher nonlinearity, the number

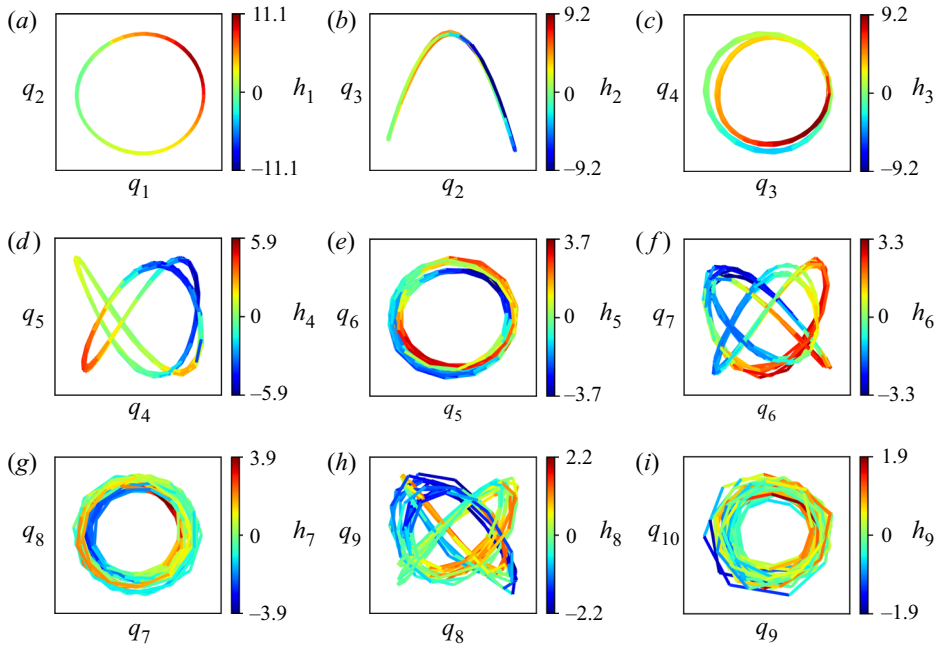


Figure 19. Correlations of modes coloured by sensitivity magnitude for the optimized case.

of loops in corresponding correlations increases. Interestingly, in the optimized case, high rank correlations have multiple similar loops, indicating linear behaviour of the vortical structures in the wake of the optimized airfoil.

Figures 18 and 19 show correlations coloured by sensitivity magnitudes, $h_i \in \mathbf{h} = [h_1, h_2, \dots, h_r]^\top$, for the reference and optimized cases, respectively. The maximum sensitivity belongs to the first rank and, as the rank increases, the sensitivity magnitude reduces. The primary weight of the sensitivity belongs to prominent vortical structures in the wake of the airfoil. As we discussed earlier, the sensitivity of the optimized case is less than that of the reference case, which is observed by comparing sensitivity magnitudes in figures 18 and 19. It is worth mentioning that the maximum sensitivity magnitude for each mode happens when the generalized coordinates in correlations move to another loop. This exchange occurs when an LEV or TEV sheds downstream. Therefore, based on these observations, the sensitivity of the generalized coordinates in Hilbert space is strongly related to the sensitivity of the state vector in physical space. Consequently, applying optimization to the dynamical system in Hilbert space is analogous to optimization in physical space.

9. Conclusions

In this study, we proposed a new approach for PDE-constrained optimization of nonlinear systems. The intuition of this study is that, instead of directly optimizing FOMs in physical space, we transform the physical governing equations into an unphysical space, where the dynamics of the system evolve on manifolds. This allows us to optimize the evolutionary behaviour of dynamical systems in lower dimensions. Hence, in optimization problems, the optimizer is able to change the dynamical behaviour of the system, such that its effect in physical space minimizes the objective function. To this end, we developed a

closure model in the form of a ROM. This closure model was explicitly derived from the FOM using the LSPG approach, and leverages the minimum-residual property over a discrete temporal domain. Using this ROM, we omitted low energy unstable modes, while maintaining accuracy of the ROM. Additionally, sampling techniques were considered for building the trial and test basis functions, which shape the manifolds in Hilbert space. This procedure is referred to as a physics-constrained data-driven approach. The main feature of developing a ROM using this approach is that it provides derivatives of the FOM (i.e. Jacobian) in lower dimensions. Furthermore, sequential least-squares minimizations were used to solve this closure model, leading to a robust framework, specialized for unsteady optimization.

Shape optimization of an NACA 0012 airfoil in the presence of flow separation at $Re = 1000$ was then considered. It was shown that the proposed framework results in a significant improvement in the aerodynamic performance of the airfoil, with a drag reduction of approximately 20% observed at $\alpha_{eff} = 25^\circ$. Unlike the reference case with the nonlinear dynamical system, the results illustrate that the dynamical system of the optimized case exhibit a linear-like dynamical behaviour. Therefore, we expect that implementing the proposed optimization approach can be applied to large-scale nonlinear problems, such as unsteady flows, and this approach significantly reduces computational cost and data storage requirements. Future work will focus on applying this approach to larger and more complicated turbulent flows.

Supplementary material. Supplementary material is available at <https://doi.org/10.1017/jfm.2021.1051>.

Funding. The authors acknowledge the support of the Natural Sciences and Engineering Research Council of Canada (NSERC): RGPAS-2017-507988, RGPIN-2017-06773. This research was enabled in part by support provided by Calcul Quebec (www.calculquebec.ca), WestGrid (www.westgrid.ca), SciNet (www.scinethpc.ca) and Compute Canada (www.computeCanada.ca) via a Resources for Research Groups allocation. H.K. also acknowledges support from Concordia University via the Concordia Graduate Fellowship and Concordia University International Tuition Award of Excellence.

Data statement. Data relating to the results in this manuscript can be downloaded from the publication's website under a CC BY-NC-ND 4.0 license.

Declaration of interests. The authors report no conflict of interest.

Author ORCIDs.

 Hamid R. Karbasian <https://orcid.org/0000-0003-3067-4359>.

REFERENCES

- AMSALLEM, D. & FARHAT, C. 2008 Interpolation method for adapting reduced-order models and application to aeroelasticity. *AIAA J.* **46** (7), 1803–1813.
- APPONSAH, K.P. & ZINGG, D.W. 2020 Aerodynamic shape optimization for unsteady flows: some benchmark problems. In *AIAA Scitech 2020 Forum*, p. 0541.
- ARIAN, E., FAHL, M. & SACHS, E.W. 2000 Trust-region proper orthogonal decomposition for flow control. *Tech. Rep.* Institute for computer application in science and engineering, Hampton, VA.
- BALAY, S., *et al.* 2020 PETSc users manual. *Tech. Rep.* ANL-95/11 – Revision 3.14. Argonne National Laboratory.
- BERGMANN, M. & CORDIER, L. 2008 Optimal control of the cylinder wake in the laminar regime by trust-region methods and POD reduced-order models. *J. Comput. Phys.* **227** (16), 7813–7840.
- BERGMANN, M., CORDIER, L. & BRANCHER, J.-P. 2005 Optimal rotary control of the cylinder wake using proper orthogonal decomposition reduced-order model. *Phys. Fluids* **17** (9), 097101.
- BLONIGAN, P.J., GOMEZ, S.A. & WANG, Q. 2014 Least squares shadowing for sensitivity analysis of turbulent fluid flows. In *52nd Aerospace Sciences Meeting*, p. 1426.
- BRIGHT, I., LIN, G. & KUTZ, J.N. 2013 Compressive sensing based machine learning strategy for characterizing the flow around a cylinder with limited pressure measurements. *Phys. Fluids* **25** (12), 127102.

- BRUNTON, S.L. & KUTZ, J.N. 2019 *Data-Driven Science and Engineering: Machine Learning, Dynamical Systems, and Control*. Cambridge University Press.
- BRUNTON, S.L., NOACK, B.R. & KOUMOUTSAKOS, P. 2020 Machine learning for fluid mechanics. *Annu. Rev. Fluid Mech.* **52**, 477–508.
- BUI-THANH, T., WILLCOX, K. & GHATTAS, O. 2008a Model reduction for large-scale systems with high-dimensional parametric input space. *SIAM J. Sci. Comput.* **30** (6), 3270–3288.
- BUI-THANH, T., WILLCOX, K. & GHATTAS, O. 2008b Parametric reduced-order models for probabilistic analysis of unsteady aerodynamic applications. *AIAA J.* **46** (10), 2520–2529.
- CARLBERG, K., BARONE, M. & ANTIL, H. 2017 Galerkin v. least-squares Petrov–Galerkin projection in nonlinear model reduction. *J. Comput. Phys.* **330**, 693–734.
- CARLBERG, K., BOU-MOSLEH, C. & FARHAT, C. 2011 Efficient non-linear model reduction via a least-squares Petrov–Galerkin projection and compressive tensor approximations. *Intl J. Numer. Meth. Engng* **86** (2), 155–181.
- CARLBERG, K. & FARHAT, C. 2011 A low-cost, goal-oriented ‘compact proper orthogonal decomposition’ basis for model reduction of static systems. *Intl J. Numer. Meth. Engng* **86** (3), 381–402.
- CARLBERG, K., FARHAT, C., CORTIAL, J. & AMSALLEM, D. 2013 The GNAT method for nonlinear model reduction: effective implementation and application to computational fluid dynamics and turbulent flows. *J. Comput. Phys.* **242**, 623–647.
- DALCÍN, L., PAZ, R. & STORTI, M. 2005 MPI for Python. *J. Parallel Distrib. Comput.* **65** (9), 1108–1115.
- DALCIN, L.D., PAZ, R.R., KLER, P.A. & COSIMO, A. 2011 Parallel distributed computing using Python. *Adv. Water Resour.* **34** (9), 1124–1139.
- 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.
- EPUREANU, B.I. 2003 A parametric analysis of reduced order models of viscous flows in turbomachinery. *J. Fluids Struct.* **17** (7), 971–982.
- FAHL, M. 2000 Trust-region methods for flow control based on reduced order modelling. PhD thesis, Universität Trier.
- HAY, A., BORGGGAARD, J.T. & PELLETIER, D. 2009 Local improvements to reduced-order models using sensitivity analysis of the proper orthogonal decomposition. *J. Fluid Mech.* **629**, 41–72.
- HE, P., MADER, C.A., MARTINS, J.R.R.A. & MAKI, K.J. 2019 Aerothermal optimization of a ribbed U-bend cooling channel using the adjoint method. *Intl J. Heat Mass Transfer* **140**, 152–172.
- HE, P. & MARTINS, J. 2021 A hybrid time-spectral approach for aerodynamic shape optimization with unsteady flow. In *AIAA Scitech 2021 Forum*, p. 0278.
- HERNANDEZ, V., ROMAN, J.E. & VIDAL, V. 2005 SLEPc: a scalable and flexible toolkit for the solution of eigenvalue problems. *ACM Trans. Math. Softw.* **31** (3), 351–362.
- HINZE, M. & MATTHES, U. 2011 Model order reduction for networks of ODE and PDE systems. In *IFIP Conference on System Modeling and Optimization*, pp. 92–101. Springer.
- HUANG, R.F. & LIN, C.L. 1995 Vortex shedding and shear-layer instability of wing at low-Reynolds numbers. *AIAA J.* **33** (8), 1398–1403.
- HUYNH, H.T. 2007 A flux reconstruction approach to high-order schemes including discontinuous Galerkin methods. In *18th AIAA Computational Fluid Dynamics Conference*, p. 4079.
- KARBASIAN, H.R. & VERMEIRE, B.C. 2022 Gradient-free aerodynamic shape optimization using large eddy simulation. *Comput. Fluids* **232**, 105185.
- KUNISCH, K. & VOLKWEIN, S. 2008 Proper orthogonal decomposition for optimality systems. *ESAIM* **42** (1), 1–23.
- KURTULUS, D.F. 2015 On the unsteady behavior of the flow around NACA 0012 airfoil with steady external conditions at $Re = 1000$. *Intl J. Micro Air Veh.* **7** (3), 301–326.
- LASSILA, T. & ROZZA, G. 2010 Parametric free-form shape design with PDE models and reduced basis method. *Comput. Meth. Appl. Mech. Engng* **199** (23–24), 1583–1592.
- LING, J. & TEMPLETON, J. 2015 Evaluation of machine learning algorithms for prediction of regions of high Reynolds averaged Navier–Stokes uncertainty. *Phys. Fluids* **27** (8), 085103.
- LIU, Y., LI, K., ZHANG, J., WANG, H. & LIU, L. 2012 Numerical bifurcation analysis of static stall of airfoil and dynamic stall under unsteady perturbation. *Commun. Nonlinear Sci. Numer. Simul.* **17** (8), 3427–3434.
- MANZONI, A., QUARTERONI, A. & ROZZA, G. 2012 Shape optimization for viscous flows by reduced basis methods and free-form deformation. *Intl J. Numer. Meth. Fluids* **70** (5), 646–670.
- MEENA, M., TAIRA, K. & ASAI, K. 2017 Airfoil-wake modification with Gurney flap at low Reynolds number. *AIAA J.* **56** (4), 1348–1359.
- MOARREF, R., JOVANOVIĆ, M.R., TROPP, J.A., SHARMA, A.S. & MCKEON, B.J. 2014 A low-order decomposition of turbulent channel flow via resolvent analysis and convex optimization. *Phys. Fluids* **26** (5), 051701.

- MOHEBUJJAMAN, M., REBHOLZ, L.G. & ILIESCU, T. 2018 Physically constrained data 'driven correction for reduced' order modeling of fluid flows. *Intl J. Numer. Meth. Fluids* **89** (3), 103–122.
- MURATA, T., FUKAMI, K. & FUKAGATA, K. 2020 Nonlinear mode decomposition with convolutional neural networks for fluid dynamics. *J. Fluid Mech.* 882.
- NADARAJAH, S. & JAMESON, A. 2000 A comparison of the continuous and discrete adjoint approach to automatic aerodynamic optimization. In *38th Aerospace Sciences Meeting and Exhibit*, p. 667.
- NAIR, A.G., YEH, C.-A., KAISER, E., NOACK, B.R., BRUNTON, S.L. & TAIRA, K. 2019 Cluster-based feedback control of turbulent post-stall separated flows. *J. Fluid Mech.* **875**, 345–375.
- NEWMAN, J.C. III, TAYLOR, A.C. III, BARNWELL, R.W., NEWMAN, P.A. & HOU, G.J.-W. 1999 Overview of sensitivity analysis and shape optimization for complex aerodynamic configurations. *J. Aircraft* **36** (1), 87–96.
- OLIVEIRA, I.B. & PATERA, A.T. 2007 Reduced-basis techniques for rapid reliable optimization of systems described by affinely parametrized coercive elliptic partial differential equations. *Optim. Engng* **8** (1), 43–65.
- OROZCO, C.E. & GHATTAS, O.N. 1992 Massively parallel aerodynamic shape optimization. *Comput. Syst. Engng* **3** (1–4), 311–320.
- PARISH, E.J., WENTLAND, C.R. & DURAISAMY, K. 2020 The adjoint Petrov–Galerkin method for non-linear model reduction. *Comput. Meth. Appl. Mech. Engng* **365**, 112991.
- RAVINDRAN, S.S. 2000 Reduced-order adaptive controllers for fluid flows using POD. *J. Sci. Comput.* **15** (4), 457–478.
- ROWLEY, C.W. & DAWSON, S.T.M. 2017 Model reduction for flow analysis and control. *Annu. Rev. Fluid Mech.* **49**, 387–417.
- ROZZA, G. & MANZONI, A. 2010 Model order reduction by geometrical parametrization for shape optimization in computational fluid dynamics. In *Proceedings of the ECCOMAS CFD, V European Conference on Computational Fluid Dynamics*, pp. 1–20.
- RUBINO, A., PINI, M., COLONNA, P., ALBRING, T., NIMMAGADDA, S., ECONOMON, T. & ALONSO, J. 2018 Adjoint-based fluid dynamic design optimization in quasi-periodic unsteady flow problems using a harmonic balance method. *J. Comput. Phys.* **372**, 220–235.
- TAIRA, K., BRUNTON, S.L., DAWSON, S.T.M., ROWLEY, C.W., COLONIUS, T., MCKEON, B.J., SCHMIDT, O.T., GORDEYEV, S., THEOFILIS, V. & UKEILEY, L.S. 2017 Modal analysis of fluid flows: an overview. *AIAA J.* **55** (12), 4013–4041.
- TAIRA, K., HEMATI, M.S., BRUNTON, S.L., SUN, Y., DURAISAMY, K., BAGHERI, S., DAWSON, S.T.M. & YEH, C.-A. 2020 Modal analysis of fluid flows: applications and outlook. *AIAA J.* **58** (3), 998–1022.
- VIRTANEN, P., *et al.* 2020 SciPy 1.0: fundamental algorithms for scientific computing in Python. *Nat. Meth.* **17**, 261–272.
- WANG, Q., HU, R. & BLONIGAN, P. 2014 Least squares shadowing sensitivity analysis of chaotic limit cycle oscillations. *J. Comput. Phys.* **267**, 210–224.
- YAMALEEV, N., DISKIN, B. & NIELSEN, E. 2008 Adjoint-based methodology for time-dependent optimization. In *12th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, p. 5857.
- YARUSEVYCH, S., SULLIVAN, P.E. & KAWALL, J.G. 2009 On vortex shedding from an airfoil in low-Reynolds-number flows. *J. Fluid Mech.* **632**, 245.
- YOON, G.H. 2016 Topology optimization for turbulent flow with Spalart–Allmaras model. *Comput. Meth. Appl. Mech. Engng* **303**, 288–311.
- ZAHR, M.J. & FARHAT, C. 2015 Progressive construction of a parametric reduced-order model for PDE-constrained optimization. *Intl J. Numer. Meth. Engng* **102** (5), 1111–1135.
- ZAHR, M.J. & PERSSON, P.-O. 2016 An adjoint method for a high-order discretization of deforming domain conservation laws for optimization of flow problems. *J. Comput. Phys.* **326**, 516–543.