Registration-based model reduction of parameterized two-dimensional conservation laws.

Andrea Ferrero[1] **Tommaso Taddei**[2], **Lei Zhang**[2]

[1] *Department of Mechanical and Aerospace Engineering, Politecnico di Torino, Corso Duca degli Abruzzi 24, 10129 Torino, Italy* andrea_ferrero@polito.it
[2] *IMB, UMR 5251, Univ. Bordeaux; 33400, Talence, France. Inria Bordeaux Sud-Ouest, Team MEMPHIS; 33400, Talence, France,*
tommaso.taddei@inria.fr,lei.a.zhang@inria.fr

Abstract

We propose a nonlinear registration-based model reduction procedure for rapid and reliable solution of parameterized two-dimensional steady conservation laws. This class of problems is challenging for model reduction techniques due to the presence of nonlinear terms in the equations and also due to the presence of parameter-dependent discontinuities that cannot be adequately represented through linear approximation spaces. Our approach builds on a general (i.e., independent of the underlying equation) registration procedure for the computation of a mapping Φ that tracks moving features of the solution field and on an hyper-reduced least-squares Petrov-Galerkin reduced-order model for the rapid and reliable computation of the solution coefficients. The contributions of this work are twofold. First, we investigate the application of registration-based methods to two-dimensional hyperbolic systems. Second, we propose a multi-fidelity approach to reduce the offline costs associated with the construction of the parameterized mapping and the reduced-order model. We discuss the application to an inviscid supersonic flow past a parameterized bump, to illustrate the many features of our method and to demonstrate its effectiveness.

Keywords: parameterized hyperbolic partial differential equations; model order reduction; registration methods; nonlinear approximations.

1 Introduction

1.1 Model order reduction for steady conservation laws

Despite the recent advances in high-performance computing and numerical analysis, approximation of the solution to fluid problems remains a formidable task that requires extensive computational resources. The lack of fast and reliable computational fluid dynamics (CFD) solvers limits the use of high-fidelity (`hf`) simulations to perform extensive parametric studies in science and engineering. Parameterized model order reduction (pMOR) aims at constructing a low-dimensional surrogate (or reduced-order) model (ROM) over a range of parameters, and ultimately speed up parametric studies. The goal of this paper

is to develop a nonlinear registration-based MOR procedure for steady two-dimensional conservation laws and to demonstrate its effectiveness for applications in aerodynamics.

We denote by μ the vector of model parameters in the parameter region $\mathcal{P} \subset \mathbb{R}^P$; we denote by $\Omega \subset \mathbb{R}^2$ the open computational domain — to simplify the presentation, in the introduction we assume that the domain does not depend on the parameters; however, in the numerical examples, we shall consider the case of parameterized geometries. Given the parametric field $w : A \times \mathcal{P} \to B$ for some open sets $A \subset \mathbb{R}^n$ and $B \subset \mathbb{R}^m$ with $m, n \in \mathbb{N}$, we denote by $w_\mu = w(\cdot;\mu) : A \to B$. We further denote by $\overline{A}$ the closure of the open set A in $\mathbb{R}^n$. We denote by $U : \Omega \times \mathcal{P} \to \mathbb{R}^D$ the parametric solution field satisfying the conservation law:

$$\nabla \cdot F_\mu(U_\mu) = S_\mu(U_\mu) \quad \text{in } \Omega, \tag{1}$$

where $F : \mathbb{R}^D \times \mathcal{P} \to \mathbb{R}^{D,2}$ is the physical flux and $S : \mathbb{R}^D \times \mathcal{P} \to \mathbb{R}^D$ is the source term. The problem is completed with suitable boundary conditions that depend on the number of incoming characteristics (cf. [68, Chapter 12]). We denote by $\mathcal{M} := \{U_\mu : \mu \in \mathcal{P}\}$ the solution manifold associated with (1). We further define the Hilbert space $\mathcal{X} = [L^2(\Omega)]^D$, endowed with the inner product $(\cdot,\cdot)$ and the induced norm $\|\cdot\| := \sqrt{(\cdot,\cdot)}$, such that $(w, v) = \int_\Omega w \cdot v \, dx$ for all $w, v \in \mathcal{X}$.

We introduce the finite element (FE) mesh $\mathcal{T}_{\rm hf} := \left(\{x_j^{\rm hf}\}_{j=1}^{N_{\rm hf,v}}, \mathtt{T}\right)$ where $\{x_j^{\rm hf}\}_j \subset \overline{\Omega}$ are the nodes of the mesh, $N_{\rm hf,v}$ is the total number of nodes, and $\mathtt{T} \in \mathbb{N}^{n_{\rm lp}, N_{\rm e}}$ is the connectivity matrix, where $n_{\rm lp}$ is the number of degrees of freedom in each element and $N_{\rm e}$ is the total number of elements. We denote by $\mathcal{X}_{\rm hf} \subset \mathcal{X}$ a FE discretization associated with $\mathcal{T}_{\rm hf}$ and we set $N_{\rm hf} = \dim(\mathcal{X}_{\rm hf})$. Given $w \in \mathcal{X}_{\rm hf}$, we denote by $\mathbf{w} \in \mathbb{R}^{N_{\rm hf}}$ the vector representation of w with respect to a suitable basis: note that the pair mesh-coefficients $(\mathcal{T}_{\rm hf}, \mathbf{w})$ uniquely identifies the field $w \in \mathcal{X}_{\rm hf}$. Finally, we denote by $U_\mu^{\rm hf} \in \mathcal{X}_{\rm hf}$ the `hf` estimate of the solution $U_\mu \in \mathcal{X}$ to (1) for a given $\mu \in \mathcal{P}$.

Hyperbolic problems with moving fronts are extremely challenging for state-of-the-art model reduction procedures. First, the vast majority of MOR methods rely on linear approximations: as shown in several studies (e.g., [43]), linear methods are fundamentally ill-suited to deal with parameter-dependent sharp gradients that naturally arise in the solutions to hyperbolic conservation laws. Another major issue concerns the construction of accurate meshes for parametric studies. For advection-dominated problems, adaptive mesh refinement (AMR) is of paramount importance to reduce the size of the mesh required to achieve a given accuracy. However, if parametric variations strongly affect the location of sharp-gradient regions, AMR should be applied to each system configuration and will lead to `hf` discretizations of intractable size. Effective MOR procedures for conservation laws should thus embed an effective parametric AMR strategy to track moving structures.

1.2 Registration methods for parameterized problems

Registration-based (or Lagrangian) methods for pMOR (e.g., [29, 42, 54, 58, 57]) rely on the introduction of a parametric mapping $\Phi : \Omega \times \mathcal{P} \to \Omega$ such that (i) Φ_μ is a bijection from Ω in itself for all $\mu \in \mathcal{P}$, and (ii) the mapped manifold $\widetilde{\mathcal{M}} = \{U_\mu \circ \Phi_\mu : \mu \in \mathcal{P}\}$ is more amenable for linear compression methods. In

the FE framework, or equivalently in the finite volume context, this corresponds to considering approximations of the form

$$\mu \in \mathcal{P} \mapsto \left(\Phi_\mu(\mathcal{T}_{\mathrm{hf}}),\ \widehat{\mathbf{U}}_\mu = \mathbf{Z}\,\widehat{\boldsymbol{\alpha}}_\mu\right), \tag{2}$$

where $\Phi_\mu(\mathcal{T}_{\mathrm{hf}}) := \left(\{\Phi_\mu(x_j^{\mathrm{hf}})\}_{j=1}^{N_{\mathrm{hf,v}}}, \mathtt{T}\right)$, $\mathbf{Z} \in \mathbb{R}^{N_{\mathrm{hf}},N}$ is the N-dimensional reduced-order basis (ROB). Note that the mapped mesh $\Phi_\mu(\mathcal{T}_{\mathrm{hf}})$ shares with $\mathcal{T}_{\mathrm{hf}}$ the same connectivity matrix, while $\widehat{\mathbf{U}}_\mu = \mathbf{Z}\,\widehat{\boldsymbol{\alpha}}_\mu$ can be viewed as an approximation of U_μ if paired with the mesh $\Phi_\mu(\mathcal{T}_{\mathrm{hf}})$, or as an approximation of $U_\mu \circ \Phi_\mu$ if paired with the mesh $\mathcal{T}_{\mathrm{hf}}$.

Several features of registration methods are attractive for applications to hyperbolic problems with moving fronts. First, registration methods are effective to track sharp gradients of the solution field, and ultimately improve performance of linear compression methods in the reference configuration. Second, since the mesh should be refined in the proximity of the jumps of the solution, reducing shocks' and contact discontinuities' sensitivity to parameter variations leads to a significant reduction of the mesh size required for a given accuracy. Third, after having built the mapping Φ, Lagrangian methods reduce to linear methods in parameterized domains: this class of methods has been widely studied in the MOR literature (see the reviews [33, 53] and also [60]) and is now well-understood. In particular, we can rely on standard training algorithms to build $\mathbf{Z}$ in (2) – in particular, proper orthogonal decomposition (POD, [7, 63]) and the weak-Greedy algorithm [52] — and on effective hyper-reduced projection-based techniques to compute the solution coefficients $\widehat{\boldsymbol{\alpha}}_\mu$.

In this work, we consider the registration procedure first proposed in [57] and then extended in [59, 61] to generate the mapping; then, similarly to [59], we rely on a projection-based least-squares Petrov-Galerkin (LSPG, [13, 11]) formulation with elementwise empirical quadrature (EQ, [22, 69]) to estimate the coefficients $\widehat{\boldsymbol{\alpha}}_\mu$ for any new value of the parameters. Furthermore, we rely on the *discretize-then-map* framework (cf. [17, 60, 65]) to deal with geometry variations. The contribution of the paper is twofold.

- We show performance of registration-based model reduction for a representative problem in aerodynamics with shocks: we discuss performance of registration, and we also address the combination with projection-based MOR techniques. In particular, we investigate in detail the offline-online computational decomposition and we also comment on hyper-reduction, which is key for online efficiency.
- We present work toward the implementation of a multi-fidelity approach for registration-based model reduction. As explained in [57, 59, 61], a major issue of our registration procedure is the need for extensive explorations of the parameter domain: in this work, we show that we can rely on a significantly less accurate `hf` discretization to generate the snapshots used for registration and ultimately greatly reduce the cost of offline training. In the numerical results, we further show that multi-fidelity training might help reduce the size of the `hf` discretization required to properly track moving features — in effect, spatio-parameter mesh adaptivity.

The outline of the paper is as follows. In section 2, we introduce the model problem; in section 3, we present the methodology: first, we introduce the regis-

tration algorithm proposed in [61], then, we discuss the projection-based scheme and finally we present the offline/online computational decomposition based on a two-fidelity sampling. In section 4, we present extensive numerical investigations to illustrate the performance of our proposal. In the remainder of this section, we discuss relation to previous works (cf. section 1.3), we briefly comment on the many nonlinear approximation methods appeared in the literature to better clarify the interest for registration-based methods (cf. section 1.4), and we present relevant notation (cf. section 1.5).

1.3 Relation to previous works

Several authors have applied MOR techniques to aerodynamics problems including inviscid flows: we refer to [71] for a review; we further refer to the early work by Zimmermann et al, [23] and to the more recent work by Carlberg et al, [8] for application to aerodynamics of techniques based on nonlinear approximations. Simultaneous adaptivity in space — via AMR — and in parameter — via Greedy sampling — has been considered by Yano in [70] and more recently in [56]. Methods in [56, 70] rely on h-refinement to adapt the spatial mesh, while we exploit a solution-aware parameterized mapping to deform the mesh without changing its topology (r-adaptivity): we thus envision that the two strategies might be combined with mutual benefits.

Multifidelity methods have been extensively studied in the MOR literature: we refer to [45] for a thorough review and also to the more recent work by Kast et al. [31]. As explicitly stated in section 1.2, the present study offers a proof of concept of the application of multifidelity schemes in combination with registration methods; it also shows the importance of multifidelity schemes for spatio-parameter adaptivity.

As discussed in [57, 59, 61], the fundamental building block of our registration procedure is a nonlinear non-convex optimization statement for the computation of the mapping Φ for the parameters in the training set. Our optimization statement minimizes an L^2 reconstruction error plus a number of terms that control the smoothness of the map and the mesh distortion: minimization of the L^2 reconstruction error has been previously considered in several works (e.g., [40, 50, 51, 55]); on the other hand, penalization of mesh distortion has been considered in [73] in a related context.

For completeness, as already discussed in [61], we remark that registration-based methods are tightly linked to a number of techniques in related fields. First, registration is central in image processing: in this field, registration — more precisely, non-rigid point set registration [35, 36, 76] — refers to the process of computing a spatial transformation that optimally aligns pairs of point sets. In computational mechanics, Persson and Zahr have proposed in [73] an r-adaptive optimization-based high-order discretization method to deal with shocks/sharp gradients of the solutions to advection-dominated problems. In uncertainty quantification, several authors (see, e.g., [37]) have proposed measure transport approaches to sampling: transport maps are used to "push forward" samples from a reference configuration and ultimately facilitate sampling from non-Gaussian distributions. Finally, the notion of registration is also at the core of diffeomorphic dimensionality reduction ([64]) in the field of machine learning.

1.4 Methods based on nonlinear approximations: expressivity and learnability

In recent years, there has been a growing interest in nonlinear model reduction techniques, particularly for CFD applications. A first class of methods relies on adaptive partitioning of the parameter domain, [19]. Another class of methods relies on online basis update and/or refinement: relevant works that fit in this category might rely on low-rank updates (e.g., [12, 21, 44]), or might rely on Grasmannian learning to construct parameter-dependent reduced-order bases [1, 75]. A third class of methods relies on the introduction in the offline/online workflow of a preprocessing stage to reformulate the problem in a form that is more amenable for linear approximations: representative methods in this category are the approach in [25] based on approximate Lax pairs, and the method of freezing in [42]. We remark that such preprocessing stage might be performed once during the offline stage, at the beginning of the online stage for any new $\mu \in \mathcal{P}$, or at each time step in combination with a suitable time-marching scheme. A fourth class of methods considers directly nonlinear approximations in combination with specialized methods to compute the solution during the online stage: to provide concrete references, we refer to the approaches based on convolutional autoencoders, [24, 30, 32, 34], and to the approach in [20] based on optimal transport and nonlinear interpolation. As explained below (cf. (3b)), Lagrangian methods lead to predictions $\widehat{U}$ that are linear in the solution coefficients $\widehat{\boldsymbol{\alpha}}_\mu$ and nonlinear in the mapping coefficients $\widehat{\mathbf{a}}_\mu$: depending on the way mapping coefficients are computed, Lagrangian methods fit in the third category (e.g., [57, 59] and this work) or in the fourth category (e.g., [39]).

To analyze the many nonlinear proposals and ultimately perform an informed decision for the specific problem of interest, we shall interpret pMOR techniques as the combination of two fundamental blocks: a low-rank parameter-independent operator $\mathtt{Z} : \mathcal{A} \subset \mathbb{R}^Q \to \mathcal{X}$ for some $Q \in \mathbb{N}$ that will be clarified below in (3) and a ROM for the reduced coefficients $\widehat{\boldsymbol{\beta}} : \mathcal{P} \to \mathcal{A}$. To build $\mathtt{Z}$, we first identify a class of approximations (see (3) below) and then we proceed to use offline data to identify the proper (quasi-optimal) approximation within that class; after having built $\mathtt{Z}$, we rely on intrusive (projection-based) or non-intrusive (data-fitted) methods to rapidly find the coefficients $\widehat{\boldsymbol{\beta}}_\mu \in \mathcal{A}$ for any new value of the parameters in $\mathcal{P}$. Examples of approximation classes include the aforementioned linear methods, Lagrangian methods, convolutional methods, and transported methods.

- Linear methods can be written as

$$\widehat{U}_\mu = \mathtt{Z}(\widehat{\boldsymbol{\beta}}_\mu = \widehat{\boldsymbol{\alpha}}_\mu) = \sum_{n=1}^{N} (\widehat{\boldsymbol{\alpha}}_\mu)_n \zeta_n, \tag{3a}$$

  with $Q = N$, $\mathcal{A} = \mathbb{R}^N$, and $\zeta_1, \dots, \zeta_N \in \mathcal{X}$.

- Lagrangian (or registration-based) methods can be written as

$$\widehat{U}_\mu = \mathtt{Z}(\widehat{\boldsymbol{\beta}}_\mu = [\widehat{\boldsymbol{\alpha}}_\mu, \widehat{\mathbf{a}}_\mu]) = \sum_{n=1}^{N} (\widehat{\boldsymbol{\alpha}}_\mu)_n \zeta_n \circ \mathtt{N}(\widehat{\mathbf{a}}_\mu)^{-1} \tag{3b}$$

  where $\zeta_1, \dots, \zeta_N \in \mathcal{X}$, $\mathcal{A} = \mathbb{R}^N \times \mathcal{A}_{\mathrm{bj}}$, $\mathtt{N} : \mathbb{R}^M \to \mathrm{Lip}(\Omega; \mathbb{R}^2)$ such that $\mathtt{N}(\mathbf{a})$ is a bijection in Ω for all $\mathbf{a} \in \mathcal{A}_{\mathrm{bj}}$, $Q = N + M$.

- Convolutional approximations ([24, 30, 32, 34]) with $L > 0$ layers can be stated as

$$\mathtt{Z}(\widehat{\boldsymbol{\beta}}_\mu = [\widehat{\boldsymbol{\alpha}}_{1,\mu}, \dots, \widehat{\boldsymbol{\alpha}}_{L,\mu}]) = \mathtt{N}_L\left(\mathtt{N}_{L-1}\left(\dots,\ \boldsymbol{\alpha}_{L-1,\mu}\right), \boldsymbol{\alpha}_{L,\mu}\right) \tag{3c}$$

  where $\mathtt{N}_\ell : \mathbb{R}^{D_\ell} \times \mathbb{R}^{N_\ell} \to \mathbb{R}^{D_{\ell+1}}$ with $D_1 = 2$ (number of spatial dimensions) and $D_{L+1} = D$ (number of state variables), $Q = \sum_{\ell=1}^{L} N_\ell$ and $\mathcal{A} = \mathbb{R}^Q$.

- Finally, transported (or transformed) snapshot methods ([10, 41, 49, 66]) with $N > 0$ terms can be stated as

$$\mathtt{Z}(\widehat{\boldsymbol{\beta}}_\mu = [\widehat{\boldsymbol{\alpha}}_\mu, \widehat{\mathbf{a}}_{1,\mu}, \dots, \widehat{\mathbf{a}}_{N,\mu}]) = \sum_{n=1}^{N} (\widehat{\boldsymbol{\alpha}}_\mu)_n \zeta_n \circ \mathtt{N}_n(\widehat{\mathbf{a}}_{n,\mu}) \tag{3d}$$

  where $\mathtt{N}_1, \dots, \mathtt{N}_N : \mathbb{R}^M \to \mathrm{Lip}(\Omega : \mathbb{R}^2)$, $\zeta_1, \dots, \zeta_N \in \mathcal{X}_{\mathrm{ext}} := \{v \in L^2(\mathbb{R}^2) : v|_\Omega \in \mathcal{X}\}$.

Note that, while in Lagrangian methods we require that $\mathtt{N}$ is bijective, transformed methods do not explicitly require bijectivity of $\mathtt{N}_1, \dots, \mathtt{N}_N$. Note also that linear methods are a subset of Lagrangian methods — in the sense that they reduce to linear methods for $\mathtt{N} = \mathtt{id}$, $\mathtt{id}(x) = x$. Similarly, Lagrangian methods are a subset of convolutional and transported methods.

The choice of the class of approximations should be a compromise between *expressivity* and *learnability*. In statistical learning, expressivity (or expressive power) of a network refers to the approximation properties for a given class of functions, [27]. Given the class of approximations $\mathcal{C} \subset C(\mathcal{A}; \mathcal{X})$ for some $\mathcal{A} \subset \mathbb{R}^Q$ — $C(\mathcal{A}; \mathcal{X})$ is the space of continuous applications from $\mathcal{A}$ to $\mathcal{X}$ — we measure the expressivity of $\mathcal{C}$ for $\mathcal{M}$ in terms of the nonlinear width ([18]):

$$\inf_{\mathtt{Z} \in \mathcal{C}} \sup_{w \in \mathcal{M}} \inf_{\boldsymbol{\beta} \in \mathcal{A}} \|\mathtt{Z}(\boldsymbol{\beta}) - w\|. \tag{4}$$

On the other hand, learnability depends on two distinct factors: (i) the performance of available training algorithms to identify an approximation map $\mathtt{Z}$ in $\mathcal{C}$ that approximately realizes the optimum of (4); and (ii) the performance of available methods to rapidly and reliably compute the coefficients $\widehat{\boldsymbol{\beta}}_\mu$ during the online stage. Note that the training algorithm in (i) is fed with a finite set of snapshots from $\mathcal{M}$: due to the large cost of `hf` CFD simulations, reduction of the number of required offline simulations is key for practical applications.

Since expressivity depends on the particular manifold of interest, while learnability depends on the PDE model under consideration, it seems difficult to offer a definitive answer concerning the optimal choice of the approximation class $\mathcal{C}$. The aim of this work is to show that Lagrangian approximations have high expressive power for a representative problem in aerodynamics and that they can be learned effectively based on sparse datasets: further theoretical and numerical investigations are needed to clarify the scope of the present class of methods and ultimately offer guidelines for the choice of the class of approximations.

1.5 Notation

We estimate the solution to (1) using a nodal-based discontinuous Galerkin (DG) finite element (FE) discretization of degree $\mathtt{p}$. Similarly to [61], we resort

to a FE isoparametric discretization. We define the reference element $\widehat{\mathtt{D}} = \{X \in [0,1]^2 : \sum_{d=1}^{2} X_d < 1\}$ and the Lagrangian basis $\{\ell_i\}_{i=1}^{n_{\rm lp}}$ of the polynomial space $\mathbb{P}_{\mathtt{p}}(\widehat{\mathtt{D}})$ associated with the nodes $\{X_i\}_{i=1}^{n_{\rm lp}}$; then, recalling the definition of $\mathcal{T}_{\rm hf}$ in section 1.1, we define the elemental mappings $\{\Psi_k^{\rm hf}\}_{k=1}^{N_{\rm e}}$ such that

$$\Psi_k^{\rm hf}(X) = \sum_{i=1}^{n_{\rm lp}} x^{\rm hf}_{\mathtt{T}_{i,k}}\, \ell_i(X), \tag{5}$$

and the elements of the mesh $\{\mathtt{D}_k := \Psi_k(\widehat{\mathtt{D}})\}_k$. We further define the basis functions $\ell_{i,k} : \Omega \to \mathbb{R}$ such that $\ell_{i,k}(x) = 0$ for all $x \notin \mathtt{D}_k$ and $\ell_{i,k} = \ell_i \circ \Psi_k^{-1}(x)$ for $x \in \mathtt{D}_k$, $i = 1, \dots, n_{\rm lp}$, $k = 1, \dots, N_{\rm e}$.

We define the FE space $\mathcal{X}_{\rm hf} = \text{span}\{\ell_{i,k} e_d : i = 1, \dots, n_{\rm lp}, k = 1, \dots, N_{\rm e}, d = 1, \dots, D\}$ where $e_1, \dots, e_D$ are the canonical basis of $\mathbb{R}^D$. Given $w \in \mathcal{X}_{\rm hf}$, we denote by $\mathbf{w} \in \mathbb{R}^{N_{\rm hf}}$, $N_{\rm hf} = n_{\rm lp} \cdot N_{\rm e} \cdot D$, the corresponding vector of coefficients such that

$$w(x) = \sum_{k=1}^{N_{\rm e}} \sum_{i=1}^{n_{\rm lp}} \sum_{d=1}^{D} (\mathbf{w})_{i+n_{\rm lp}(k-1)+n_{\rm lp}N_{\rm e}(d-1)}\ \ell_{i,k}(x)\, e_d, \qquad \forall\, x \in \Omega. \tag{6}$$

Note that (6) introduces an isomorphism between $\mathbb{R}^{N_{\rm hf}}$ and $\mathcal{X}_{\rm hf}$. Following the discussion in [60], we can extend the previous definitions to the mapped mesh and mapped FE space. We omit the details.

In view of the FE approximation, it is important that the deformed mesh $\Phi_\mu(\mathcal{T}_{\rm hf})$ (cf. (2)) does not have inverted elements. In this respect, we say that the mapping $\Phi : \Omega \times \mathcal{P} \to \mathbb{R}^2$ is bijective with respect to $\mathcal{T}_{\rm hf}$ (*discrete bijectivity*, [61, Definition 2.2]) if the elemental mappings of the deformed mesh are invertible.

2 Model problem

We consider the problem of approximating the solution to the parameterized compressible Euler equations. The compressible Euler equations are a widely-used model to study aerodynamic flows: we refer to [62] for a thorough discussion; we here consider the non-dimensional form of the equations. We denote by ρ the density of the fluid, by $u = [u_1, u_2]$ the velocity field, by E the total energy and by p the pressure; we further define the vector of conserved variables $U = [\rho, \rho u, E] : \Omega \to \mathbb{R}^{D=4}$. In this work, we consider the case of ideal gases for which we have the following relationship between pressure and conserved variables U:

$$p = (\gamma - 1)\left(E - \frac{1}{2}\rho \|u\|_2^2\right), \tag{7a}$$

where γ is the ratio of specific heats, which is here set equal to $\gamma = 1.4$. We further introduce the speed of sound a and the Mach number Ma with respect to the channel axis such that

$$a = \sqrt{\gamma \frac{p}{\rho}}, \quad \text{Ma} = \frac{u_1}{a}. \tag{7b}$$

Finally, we introduce the Euler physical flux and source term:

$$F(U) = \begin{bmatrix} \rho u^T \\ \rho u\, u^T + p\mathbb{1} \\ u^T(E+p) \end{bmatrix}, \quad S(U) = 0. \tag{7c}$$

We consider a parametric channel flow past a circular bump: the parameters are the free-stream Mach number Ma_∞ and the central angle α associated with the bump — cf. Figures 1(a),

$$\mu = [\alpha, \mathrm{Ma}_\infty] \in \mathcal{P} = [0.75, 0.8] \times [1.7, 1.8]. \tag{8}$$

The horizontal length of the bump and the height of the channel are set to one. We impose wall conditions at the lower and upper boundaries, transmissive boundary conditions at the outflow and we set $U = U_\infty$ at the inflow with

$$\rho_\infty = \frac{p_\infty}{T_\infty}, \qquad u_\infty = \sqrt{\gamma T_\infty} \begin{bmatrix} \mathrm{Ma}_\infty \\ 0 \end{bmatrix},$$
$$p_\infty = \frac{1}{(1 + \frac{\gamma-1}{2}\mathrm{Ma}_\infty^2)^{\frac{\gamma}{\gamma-1}}}, \quad T_\infty = \frac{1}{1 + \frac{\gamma-1}{2}\mathrm{Ma}_\infty^2}.$$

Figure 1(b) shows an horizontal slice of the Mach number at $x_2 = 0.6$ for three parameters $\mu_{\min} = [0.75, 1.7]$, $\bar{\mu} = [0.775, 1.75]$ $\mu_{\max} = [0.8, 1.8]$; Figures 1(c) and (d) show the contour lines of the Mach number for $\mu_{\min}$ and $\mu_{\max}$: the red dots in the Figures denote salient points of the flow for $\mu = \mu_{\min}$ and are intended to simplify the comparisons between the two flows.

We resort to a DG discretization based on artificial viscosity. We use the local Lax-Friedrichs flux for the advection term, and the BR2 flux (cf. [6]) for the diffusion term. We consider the piecewise-constant viscosity

$$(\nu(U))_k \;=\; c_{\mathrm{visc}} \left(\frac{h_k}{\mathtt{p}}\right)^2 \frac{1}{|\mathtt{D}_k|} \int_{\mathtt{D}_k} |\nabla \cdot u| dx \tag{9}$$

where $h_k = \sqrt{|\mathtt{D}_k|}$ is the characteristic size of the k-th element of the mesh and $c_{\mathrm{visc}} > 0$ is a constant set equal to $c_{\mathrm{visc}} = 10$ in the numerical simulations. Note that (9) is an example of dilation-based model for the viscosity: we refer to the recent review [72] for alternative viscosity models and for extensive comparisons.

To estimate the `hf` solution $U_\mu^{\mathrm{hf}} \in \mathcal{X}_{\mathrm{hf}}$, we resort to the pseudo-time continuation strategy proposed in [5]. More in detail, if we denote by $\mathbf{R}_\mu : \mathcal{X}_{\mathrm{hf}} \to \mathbb{R}^{N_{\mathrm{hf}}}$ and by $\mathbf{J}_\mu : \mathcal{X}_{\mathrm{hf}} \to \mathbb{R}^{N_{\mathrm{hf}}, N_{\mathrm{hf}}}$ the `hf` residual and the `hf` Jacobian and by $\mathbf{M} \in \mathbb{R}^{N_{\mathrm{hf}}, N_{\mathrm{hf}}}$ the mass matrix, we consider the iterative scheme:

$$\begin{cases} \left(\mathbf{M} + \Delta t_k \mathbf{J}_\mu(U_\mu^{\mathrm{hf,k}})\right) \delta U_\mu^{\mathrm{hf,k+1}} = -\mathbf{R}_\mu(U_\mu^{\mathrm{hf,k}}) \quad k = 1, 2, \ldots, \\ U_\mu^{\mathrm{hf,k+1}} = U_\mu^{\mathrm{hf}} + \Delta t_k \delta U_\mu^{\mathrm{hf,k+1}}, \end{cases} \tag{10}$$

where Δt_k is chosen adaptively based on the strategy detailed in [16, Chapter 4]. Note that (10) can be interpreted as a Newton solver with an adaptive relaxation factor.

We conclude this section by introducing the purely-geometric map used to deform the mesh in absence of a priori information about the solution: in section 3.1, we introduce a generalization of this map that takes into account the

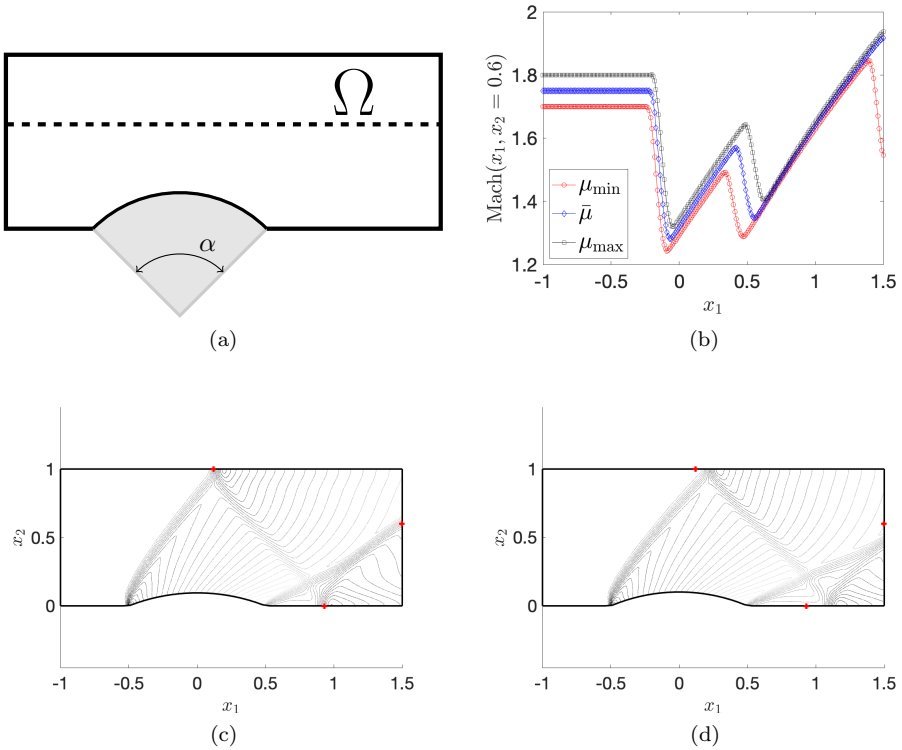


Figure 1: flow past a circular bump. (a) geometric configuration. (b) horizontal slices of the Mach number at $x_2 = 0.6$ for $\mu_{\min} = [0.75, 1.7]$, $\bar{\mu} = [0.775, 1.75]$ $\mu_{\max} = [0.8, 1.8]$. (c)-(d) contour lines of the Mach number for $\mu_{\min}$ and $\mu_{\max}$.

parametric field of interest. Towards this end, we define $\widehat{\Omega} = (0,1)^2$ and we introduce the parameterized Gordon-Hall map (cf. [26]) as

$$\begin{aligned}\Psi_\mu(x) = \quad & (1-x_2)c_{\text{btm},\mu}(x_1) + x_2 c_{\text{top}}(x_1) + (1-x_1)c_{\text{left}}(x_2) + x_1 c_{\text{right}}(x_2) \\ & - ((1-x_1)(1-x_2)c_{\text{btm},\mu}(0) + x_1 x_2 c_{\text{top}}(1) + x_1(1-x_2)c_{\text{btm},\mu}(1) \\ & + (1-x_1)x_2 c_{\text{top}}(0))\,,\end{aligned} \tag{11}$$

where $c_{\text{btm}}, c_{\text{top}}, c_{\text{left}}, c_{\text{right}}$ are parameterizations of the bottom, top, left and right boundaries of the domain, respectively. Note that c_{btm} depends on the parameter μ through the angle α (cf. Figure 1(a)): we build c_{btm} so that the jump discontinuities of its derivative c'_{btm} — which correspond to the extrema of the bump — are located at $x_1 = 0.2$ and $x_1 = 0.6$ for all parameters. We further define the inverse map $\Lambda_\mu = \Psi_\mu^{-1} : \Omega_\mu \to \widehat{\Omega}$. We have now the elements to introduce the parametric mapping Φ^{geo} such that

$$\Phi^{\text{geo}}_\mu \;=\; \Psi_\mu \circ \Lambda_{\bar{\mu}}, \tag{12}$$

where $\bar{\mu}$ is the centroid of $\mathcal{P}$. Given the mesh $\mathcal{T}_{\text{hf}}$, we compute the reference points $\{x_j^{\text{hf,ref}} = \Lambda_{\bar{\mu}}(x_j^{\text{hf}})\}_{j=1}^{N_{\text{hf,v}}}$; then, for any new value of the parameter,

we compute the deformed points of the mesh using the identity $\Phi_\mu^{\text{geo}}(x_j^{\text{hf}}) = \Psi_\mu(x_j^{\text{hf,ref}})$ for $j = 1, \dots, N_{\text{hf,v}}$.

3 Methodology

In this section, we present the methodology through the vehicle of the model problem introduced in section 2. In section 3.1, we present the registration procedure, while in section 3.2, we discuss in detail the projection-based MOR scheme. Finally, in section 3.3, we illustrate the multifidelity approach to reduce offline costs. We state upfront that the two building blocks of our formulation, registration and LSPG formulation in parameterized geometries, have been extensively discussed in [61] and [60].

3.1 Registration

The registration procedure takes as input a mesh $\mathcal{T}_{\text{hf}}$ of Ω, a set of snapshots $\{(\mu^k, U^k = U^{\text{hf}}_{\mu^k})\}_{k=1}^{n_{\text{train}}}$, and returns a parameterized mapping $\Phi : \Omega \times \mathcal{P} \to \mathbb{R}^2$,

$$\Phi = \texttt{param\_registration}\left(\mathcal{T}_{\text{hf}},\ \{(\mu^k, U^k = U^{\text{hf}}_{\mu^k})\}_{k=1}^{n_{\text{train}}}\right).$$

In the remainder of this section, we illustrate the key features of the procedure and we provide several comments.

3.1.1 Spectral maps

The first step of our registration procedure consists in introducing a class of approximation maps. Following [61], we consider mappings of the form

$$\texttt{N}(\mathbf{a};\mu) \ = \ \Psi_\mu \circ \widetilde{\Phi} \circ \Lambda_{\bar{\mu}}, \quad \widetilde{\Phi} = \texttt{id} + \varphi, \ \ \varphi = \sum_{m=1}^{M} (\mathbf{a})_m \varphi_m. \tag{13a}$$

Note that $\texttt{N}$ generalizes the map (12) in the sense that $\texttt{N}(\mathbf{0};\mu) = \Phi_\mu^{\text{geo}}$. Here, $\bar{\mu}$ is the centroid of $\mathcal{P}$ and $\varphi_1, \dots, \varphi_M$ belong to the polynomial space

$$\mathcal{W}_{\text{hf}} = \left\{\varphi \in [\mathbb{Q}_J]^2 : \ \varphi \cdot \widehat{n}|_{\partial\widehat{\Omega}} = 0, \ \ \varphi(s,0) = 0, \ s \in \{0.2, 0.6\}\right\}, \tag{13b}$$

where $\mathbb{Q}_J$ denotes the space of tensorized polynomials of degree at most J in each variable, $\widehat{n}$ is the outward normal to $\widehat{\Omega}$. In the numerical tests, we consider $J = 15$. Note that the second condition in (13b) ensures that jump discontinuities of $\nabla\texttt{N}(\mathbf{a};\mu)$ are located in $[-0.5, 0], [0.5, 0]$ for all $\mathbf{a} \in \mathbb{R}^M$ and $\mu \in \mathcal{P}$. We equip the mapping space $\mathcal{W}_{\text{hf}}$ with the H^2 norm,

$$\|\varphi\|^2_{H^2(\widehat{\Omega})} := \int_{\widehat{\Omega}} \left(\sum_{i,j,k=1}^{2} (\partial_{j,k}\varphi_i)^2 \ + \ \sum_{i=1}^{2} \varphi_i^2 \right) dx. \tag{14}$$

Exploiting the analysis in [57, 61], we find that $\mathbf{N}(\mathbf{a};\mu)$ is a bijection from Ω to Ω_μ for all $\mathbf{a}$ in the set

$$\mathcal{A}_{\text{bj}} := \left\{\mathbf{a} \in \mathbb{R}^M \ : \ \inf_{x \in \widehat{\Omega}} \widehat{g}(x;\mathbf{a}) > 0\right\}, \quad \widehat{g}(\cdot;\mathbf{a}) := \det\nabla\widetilde{\Phi}(\mathbf{a}). \tag{15a}$$

The set $\mathcal{A}_{\rm bj}$ is difficult to deal with numerically: as a result, we define $\mathcal{A}'_{\rm bj} := \{\mathbf{a} \in \mathbb{R}^M \,:\, \mathfrak{C}(\mathbf{a}) \le 0\}$ such that

$$\mathfrak{C}(\mathbf{a}) \;:=\; \int_{\widehat{\Omega}} \exp\left(\frac{\epsilon - \widehat{g}(x;\mathbf{a})}{C_{\rm exp}}\right) + \exp\left(\frac{\widehat{g}(x;\mathbf{a}) - 1/\epsilon}{C_{\rm exp}}\right) dx - \delta, \tag{15b}$$

where $\epsilon, C_{\rm exp}, \delta$ are positive constants that will be specified in the next section. Provided that $\exp\left(\frac{\epsilon}{C_{\rm exp}}\right)$ is sufficiently large, we find that there exists a constant $C > 0$ such that (see [57, section 2.2]):

$$\mathcal{A}_{\rm bj} \subset \mathcal{A}'_{\rm bj} \cap \{\mathbf{a} \,:\, \sup_{x \in \widehat{\Omega}} \|\nabla \widehat{g}(x;\mathbf{a})\|_2 \le C\}. \tag{16}$$

The discussion above motivates the combination of the constraint $\mathfrak{C}(\mathbf{a}) \le 0$ with a (strong or weak) control of the second-order derivatives of the mapping. We refer to $\mathfrak{C}(\mathbf{a}) \le 0$ as to the bijectivity constraint.

3.1.2 Optimization-based registration

Given $\mu \in \mathcal{P}$, we denote by $s : \widehat{\Omega} \times \mathcal{P} \to \mathbb{R}$ a target sensor that depends on the solution U_μ and such that $s_\mu \in L^2(\widehat{\Omega})$ for all $\mu \in \mathcal{P}$, and we introduce the N-dimensional template space $\mathcal{S}_N \subset L^2(\widehat{\Omega})$. The sensor s should capture relevant features of the solution field and is crucial for the success of the method: we discuss its choice in section 3.1.4. We further denote by $\mathcal{W}_M \subset \mathcal{W}_{\rm hf}$ an M-dimensional mapping space and by $W_M : \mathbb{R}^M \to \mathcal{W}_M$ an isometry such that $\|W_M \mathbf{a}\|_{H^2(\widehat{\Omega})} = \|\mathbf{a}\|_2$ for all $\mathbf{a} \in \mathbb{R}^M$. We discuss the construction of $\mathcal{S}_N, \mathcal{W}_M$ and the sensor s_μ in the next sections.

We can then introduce the optimization statement that is used to identify the mapping coefficients for a given $\mu \in \mathcal{P}$:

$$\begin{aligned} &\min_{\mathbf{a} \in \mathbb{R}^M} \mathfrak{f}(\mathbf{a}; s_\mu, \mathcal{S}_N, W_M) \;+\; \xi |W_M \mathbf{a}|^2_{H^2(\widehat{\Omega})} \;+\; \xi_{\rm msh} \mathfrak{R}_{\rm msh}(\mathbf{a};\mu); \\ &\text{subject to } \; \mathfrak{C}(\mathbf{a}) \le 0, \end{aligned} \tag{17a}$$

where $|\varphi|^2_{H^2(\widehat{\Omega})} = \int_{\widehat{\Omega}} \sum_{i,j,k=1}^2 (\partial_{j,k} \varphi_i)^2 \;+\; \sum_{i=1}^2 \varphi_i^2 \, dx$ is the H^2 seminorm. Here, the proximity measure $\mathfrak{f}$ measures the projection error associated with the mapped target s_μ with respect to the template space $\mathcal{S}_N$,

$$\mathfrak{f}(\mathbf{a}; s_\mu, \mathcal{S}_N, W_M) \;:=\; \min_{\psi \in \mathcal{S}_N} \int_{\widehat{\Omega}} \left(s_\mu \circ \tilde{\Phi}(\cdot;\mathbf{a}) - \psi\right)^2 dx, \quad \tilde{\Phi} = \mathtt{id} + W_M \mathbf{a}. \tag{17b}$$

The contribution $\xi |W_M \mathbf{a}|^2_{H^2(\widehat{\Omega})}$ is a regularization term that is intended to control the norm of the mapping Hessian and, in particular, the gradient of the Jacobian determinant $\nabla \hat{g}(\cdot;\mathbf{a})$: recalling (16), the latter is important to enforce bijectivity. The term $\mathfrak{R}_{\rm msh}$ penalizes excessive distortions of the mesh and ultimately preserves the discrete bijectivity (cf. section 1.5):

$$\mathfrak{R}_{\rm msh}(\mathbf{a};\mu) = \sum_{k=1}^{N_{\rm e}} |\mathtt{D}_k| \exp\left(\mathfrak{f}_{\rm msh,k}\left(\mathtt{N}(\mathbf{a};\mu)\right) \,-\, \mathfrak{f}_{\rm msh,max}\right), \tag{17c}$$

where $\mathfrak{f}_{\text{msh,max}} > 0$ is a given positive constant and

$$\mathfrak{f}_{\text{msh,k}}(\Phi) := \frac{1}{2} \frac{\|\nabla \Psi_{k,\Phi}^{\text{hf},1}\|_{\text{F}}^2}{(\det(\nabla \Psi_{k,\Phi}^{\text{hf},1}))_+}, \; k = 1, \ldots, N_{\text{e}}, \tag{17d}$$

$\|\cdot\|_{\text{F}}$ is the Frobenius norm, $(\cdot)_+ = \max\{0, \cdot\}$, and $\Psi_{k,\Phi}^{\text{hf},1}$ is the elemental mapping associated with the mapped mesh and a $\mathtt{p}$=1 discretization. We observe that the indicator (17d) is widely used for high-order mesh generation, and has also been considered in [74] to prevent mesh degradation, in the DG framework. Finally, $\mathfrak{C}$ is the bijectivity constraint in (15b).

We observe that the choice of the L^2 norm in (17b) allows explicit calculation of the minimizer $\psi^\star \in \mathcal{S}_N$ and thus simplifies evaluation of the proximity measure and its derivative. In [66], Welper proposed to use the L^1 norm for a related task. The L^1 norm is natural for hyperbolic PDEs; however, it increases the cost of evaluating (17b) due to the lack of an explicit formula; furthermore, it might also affect the optimization process due to the lack of smoothness.

We observe that the optimization statement depends on several parameters: here, we set

$$\epsilon = 0.1, \;\; C_{\text{exp}} = 0.025\epsilon, \;\; \delta = 1, \;\; \mathfrak{f}_{\text{msh,max}} = 10, \;\; \xi = 10^{-3}, \;\; \xi_{\text{msh}} = 10^{-3}.$$

The choice of $\epsilon, C_{\text{exp}}, \delta$ is the same considered in [57, 61, 59]; on the other hand, the choice of ξ, ξ_{msh} is more involved and is discussed in Remark 3.1. Since the optimization statement (17) is highly nonlinear and non-convex, the choice of the initial condition is of paramount importance: here, we exploit the strategy described in [57, section 3.1.2] to initialize the optimizer; furthermore, we resort to the Matlab function `fmincon` [38], which relies on an interior penalty algorithm to find local minima of (17). In our implementation, we provide gradients of the objective function and we rely on a structured mesh on $\widehat{\Omega}$ to speed up evaluations of the sensor and its gradient at deformed quadrature points, at each iteration of the optimization algorithm.

Remark 3.1. *In our experience, the choice of ξ is of paramount importance for performance. Small values of ξ lead to lower values of the proximity measure at the price of more irregular mappings (i.e., larger values of $|W_M \mathbf{a}|_{H^2}$). We empirically observe that the latter reduces the generalization properties of the regression algorithm (cf. section 3.1.5) used to define the parameterized mapping; in terms of reconstruction performance, we also find that the mapping process introduces small-amplitude smaller spatial scale distortions that ultimately control convergence of the ROM (cf. [57, Figure 5]) and become more and more noticeable as ξ decreases. We perform thorough investigations of the sensitivity to ξ in A.*

Remark 3.2. *We observe that $\mathfrak{R}_{\text{msh}}(\mathbf{a}; \mu) = \infty$ if the deformed mesh has degenerate elements; as a result, careful implementations of the optimization algorithm — i.e., implementations that are robust to non-double results of the objective function (cf. see Matlab `fmincon` documentation) — are guaranteed to not explore degenerate deformation maps. We further recall (cf. [57, Proposition 2.3]) that if $\det(\nabla \widetilde{\Phi}(\cdot; \mathbf{a}))$ is strictly positive in $\overline{\widetilde{\Omega}}$, then $\mathtt{N}(\mathbf{a}; \mu)$ is guaranteed to map Ω in Ω_μ.*

3.1.3 Parametric registration

Given snapshots of the sensor s, $\{(\mu^k, s_{\mu^k})\}_{k=1}^{n_{\text{train}}}$, we propose to iteratively build the template space $\mathcal{S}_N$, $\mathcal{W}_M$ through the Greedy procedure provided in Algorithm 1. The algorithm takes as input (i) the sensors associated with the snapshot set, (ii) the initial template $\mathcal{S}_{N_0}$, and (iii) the mesh $\mathcal{T}_{\text{hf}}$, and returns (i) the final template space $\mathcal{S}_N$, (ii) the isometry W_M associated with the mapping space, and (iii) the mapping coefficients $\{\mathbf{a}^k\}_k$. To clarify the procedure, we introduce notation

$$\left[\mathbf{a}^\star, \mathfrak{f}^\star_{N,M}\right] = \texttt{registration}\left(s_\mu, \mathcal{S}_N, W_M, \mathcal{T}_{\text{hf}}, \mu\right)$$

to refer to the function that takes as input the target sensor s, the template space $\mathcal{S}_N$, the isometry $W_M : \mathbb{R}^M \to \mathcal{W}_M$ associated with the mapping space, the mesh $\mathcal{T}_{\text{hf}}$ of Ω and the parameter $\mu \in \mathcal{P}$ and returns a solution to (17) and the value of the proximity measure $\mathfrak{f}^\star_{N,M} = \mathfrak{f}(\mathbf{a}^\star, s_\mu, \mathcal{S}_N, W_M)$. Furthermore, we introduce the POD function that takes as input a set of mapping coefficients and returns the reduced isometry and the projected mapping coefficients

$$[W_M,\ \{\mathbf{a}^k\}_k] = \texttt{POD}\left(\{\varphi^k\}_{k=1}^{n_{\text{train}}}, tol_{\text{pod}}, \|\cdot\|_{H^2(\widehat{\Omega})}\right),$$

where $\{\varphi^k\}_{k=1}^{n_{\text{train}}} \subset \mathcal{W}_{\text{hf}}$ is a set of admissible displacements (cf. (13b)), and M is chosen according to the eigenvalues $\{\lambda_m\}_m$ of the Gramian matrix $\mathbf{C} \in \mathbb{R}^{n_{\text{train}}, n_{\text{train}}}$ such that $\mathbf{C}_{k,k'} = (\varphi^k, \varphi^{k'})_{H^2(\widehat{\Omega})}$ and,

$$M := \min\left\{ M' :\ \sum_{m=1}^{M'} \lambda_m \geq (1 - tol_{\text{pod}}) \sum_{i=1}^{n_{\text{train}}} \lambda_i \right\}. \tag{18}$$

Algorithm 1 Registration algorithm

Inputs: $\{(\mu^k, s^k = s_{\mu^k})\}_{k=1}^{n_{\text{train}}} \subset \mathcal{P} \times L^2(\widehat{\Omega})$ snapshot set, $\mathcal{S}_{N_0} = \text{span}\{\psi_n\}_{n=1}^{N_0}$ initial template space; $\mathcal{T}_{\text{hf}}$ mesh.

Outputs: $\mathcal{S}_N = \text{span}\{\psi_n\}_{n=1}^{N}$ template space, $W_M : \mathbb{R}^M \to \mathcal{W}_M$ mapping isometry, $\{\mathbf{a}^k\}_k$ mapping coefficients.

Initialization: $\mathcal{S}_{N=N_0} = \mathcal{S}_{N_0}$, $\mathcal{W}_M = \mathcal{W}_{\text{hf}}$.
for $N = N_0, \ldots, N_{\max} - 1$ **do**
  $\left[\mathbf{a}^{\star,k}, \mathfrak{f}^{\star,k}_{N,M}\right] = \texttt{registration}\left(s^k, \mathcal{S}_N, W_M, \mathcal{T}_{\text{hf}}, \mu\right)$ for $k = 1, \ldots, n_{\text{train}}$.
  $[W_M,\ \{\mathbf{a}^k\}_k] = \texttt{POD}\left(\{W_M \mathbf{a}^{\star,k}\}_{k=1}^{n_{\text{train}}}, tol_{\text{pod}}, \|\cdot\|_{H^2(\widehat{\Omega})}\right)$
  if $\max_k \mathfrak{f}^{\star,k}_{N,M} < \texttt{tol}$ **then**, `break`
  else
    $\mathcal{S}_{N+1} = \mathcal{S}_N \cup \text{span}\{s^{k^\star} \circ (\texttt{id} + W_M \mathbf{a}^{k^\star})\}$ with $k^\star = \arg\max_k \mathfrak{f}^{\star,k}_{N,M}$.
  end if
end for

We observe that our approach depends on several hyper-parameters. In our tests, we set $\mathcal{S}_{N_0=1} = \text{span}\{s_{\bar{\mu}}\}$, where $\bar{\mu}$ is the centroid of $\mathcal{P}$; furthermore, we set $N_{\max} = 5$, $tol_{\text{pod}} = 10^{-3}$ and $\texttt{tol} = 10^{-4}$.

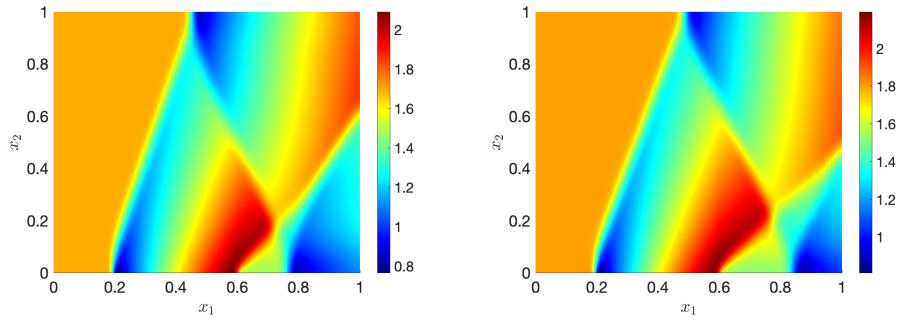
3.1.4 Choice of the registration sensor

The sensor $s : \widehat{\Omega} \times \mathcal{P} \to \mathbb{R}$ should be designed to capture relevant features of the solution field that are important to track through registration; furthermore, it should be sufficiently smooth to allow efficient applications of gradient-based optimization methods. Given the FE field in the deformed mesh $(\Phi_\mu^{\text{geo}}(\mathcal{T}_{\text{hf}}), \mathbf{U}_\mu^{\text{hf}})$, we compute the Mach number $\text{Ma}_\mu^{\text{hf}}$ (see (7b)) in the nodes of the mesh; then, we define the sensor as the solution to the following smoothing problem:

$$s_\mu := \arg \min_{s \in H^1(\widehat{\Omega})} \xi_{\text{s}} \|\nabla s\|^2_{L^2(\widehat{\Omega})} + \sum_{j=1}^{N_{\text{hf,v}}} \left(s(x_j^{\text{hf,ref}}) - \left(\mathbf{Ma}_\mu^{\text{hf}} \right)_j \right)^2 . \tag{19}$$

The regularization term associated with the hyper-parameter $\xi_{\text{s}} > 0$ is needed due to the fact that s_μ is defined over a structured[1] mesh over $\widehat{\Omega}$ that is not related to the mesh $\mathcal{T}_{\text{hf}}$ used for FE calculations. In all our tests, we consider $\xi_{\text{s}} = 10^{-4}$. We refer to [61, section 3.3] for an alternative strategy for the construction of the sensor. We remark that the sensor is exclusively used to construct the mapping: as a result, the smoothing of the sensor in (19) does not affect the numerical dissipation of the `hf` solver (and thus of the reduced order model).

We observe that the choice of the Mach number to define the registration sensor is coherent with the choice made in [46] to define the highest-modal decay artificial viscosity. Other choices are possible: in particular, we obtain similar results using the fluid density ρ in (19) as opposed to Ma. Figure 2 shows the behavior of the registration sensors for the two values of the parameter in Figure 1. We recall that the solution to the Euler equations might exhibit shocks that correspond to jump discontinuities of density, velocity, and internal energy, and contact discontinuities that correspond to jump discontinuities of density and internal energy (cf. [62, Chapter 4]). For this reason, the use of density (or Mach number) is sufficient to detect discontinuities of the whole vector-valued solution field.



(a) $\mu = \mu_{\min}$ (b) $\mu = \mu_{\max}$

Figure 2: registration sensor for two values of the parameter, $\mu_{\min} = [0.75, 1.7]$, $\mu_{\max} = [0.8, 1.8]$.

[1] As explained in [61], the use of structured meshes for the sensor is crucial to speed up the evaluation of the objective function of (17).

3.1.5 Generalization

Given the dataset $\{(\mu^k, \mathbf{a}^k)\}_{k=1}^{n_{\text{train}}}$ as provided by Algorithm 1, we resort to a multi-target regression algorithm to learn a regressor $\mu \mapsto \widehat{\mathbf{a}}_\mu$ for the mapping coefficients, and ultimately the parametric mapping

$$\Phi : \Omega \times \mathcal{P} \to \mathbb{R}^2, \quad \Phi_\mu := \mathtt{N}(W_M \widehat{\mathbf{a}}_\mu; \mu). \tag{20}$$

We here resort to radial basis function (RBF, [67]) approximation: other regression algorithms could also be considered. Similarly to [57, 59], to avoid overfitting, we verify the statistical significance of the RBF estimators. We randomly split the dataset $\{(\mu^k, \mathbf{a}^k)\}_{k=1}^{n_{\text{train}}}$ into the learning and test sets $\{(\mu^k, \mathbf{a}^k)\}_{k=1}^{n_{\text{learn}}}$ and $\{(\mu^j, \mathbf{a}^j)\}_{j=1}^{n_{\text{test}}}$ (we here consider a 80%-20% learning/test split); we compute the RBF approximation $\widehat{\mathbf{a}} : \mathcal{P} \to \mathbb{R}^M$ based on the learning set and we compute the out-of-sample R-squared coefficient for each component:

$$\mathtt{R}_m^2 = 1 - \frac{\sum_{j=1}^{n_{\text{test}}} \left(a_m^j - \left(\widehat{\mathbf{a}}_{\mu^j}\right)_m\right)^2}{\sum_{j=1}^{n_{\text{test}}} \left(a_m^j - \bar{a}_m^{\text{learn}}\right)^2}, \quad \bar{a}_m^{\text{learn}} = \frac{1}{n_{\text{learn}}} \sum_{k=1}^{n_{\text{learn}}} a_m^k. \tag{21}$$

Then, we retain exclusively modes for which $\mathtt{R}_m^2$ is above the threshold $R_{\min} = 0.75$.

Since the maps $\{\mathtt{N}(W_M \mathbf{a}^k; \mu^k)\}_k$ are bijective by construction, we expect and numerically verify that Φ_μ is also bijective for all $\mu \in \{\mu^k\}_{k=1}^{n_{\text{train}}}$; however, since we do not enforce bijectivity for out-of-sample parameters, we cannot in general expect bijectivity of Φ_μ over the whole parameter domain. In practice, we should thus consider sufficiently large training sets in Algorithm 1. This is a major limitation of the present approach that motivates the multifidelity proposal discussed in section 3.3.

3.2 Projection-based reduced-order model

To clarify the formulation and also provide insights into the implementation, we introduce a number of definitions and further notation. Given the FE vector $\mathbf{w} \in \mathbb{R}^{N_{\text{hf}}}$, we define the elemental restriction operators $\mathbf{E}_k : \mathbb{R}^{N_{\text{hf}}} \to \mathbb{R}^{n_{\text{lp}} \cdot D}$ such that $\mathbf{E}_k \mathbf{w}$ contains the values of the FE field in the nodes of the k-th element for $k = 1, \ldots, N_{\text{e}}$; the elemental restriction operators $\mathbf{E}_k^{\text{ext}} : \mathbb{R}^{N_{\text{hf}}} \to \mathbb{R}^{n_{\text{lp}} \cdot D,3}$ such that $\mathbf{E}_k^{\text{ext}} \mathbf{w}$ contains the values of the FE field in the nodes of the neighbors of the k-th element, for $k = 1, \ldots, N_{\text{e}}$. We further introduce the set of mesh nodes associated with the k-th element and its neighbors: $\mathtt{X}_k^{\text{hf}} = \{x_{\mathtt{T}_{i,k}}^{\text{hf}}\}_{i=1}^{n_{\text{lp}}}$ and $\mathtt{X}_{k,\text{ext}}^{\text{hf}} = \{x_{\mathtt{T}_{i,k'}}^{\text{hf}} : i = 1, \ldots, n_{\text{lp}},\ \overline{\mathtt{D}}_k \cap \overline{\mathtt{D}}_{k'} \neq \emptyset\}$; given the mapping Φ, we define $\Phi_\mu(\mathtt{X}_k^{\text{hf}}) = \{\Phi_\mu(x_{\mathtt{T}_{i,k}}^{\text{hf}})\}_{i=1}^{n_{\text{lp}}}$ and $\Phi_\mu(\mathtt{X}_{k,\text{ext}}^{\text{hf}}) = \{\Phi_\mu(x_{\mathtt{T}_{i,k'}}^{\text{hf}}) : i = 1, \ldots, n_{\text{lp}},\ \overline{\mathtt{D}}_k \cap \overline{\mathtt{D}}_{k'} \neq \emptyset\}$.

We have now the elements to introduce the DG residual associated with (1):

$$R_\mu^{\text{hf}}(\mathbf{U}, \mathbf{V}) \ = \ \sum_{k=1}^{N_{\text{e}}} r_\mu^k(\mathbf{U}, \mathbf{V}), \quad \forall\, \mathbf{U}, \mathbf{V} \in \mathbb{R}^{N_{\text{hf}}}, \tag{22a}$$

where the local residual r_μ^k corresponds to the contribution to the global residual associated with the k-th element of the mesh and depends on the value of the

FE fields U, V in the k-th element and in its neighbors,

$$r_\mu^k(\mathbf{U}, \mathbf{V}) \;=\; r_\mu\left(\mathbf{E}_k\mathbf{U},\, \mathbf{E}_k\mathbf{V},\, \mathbf{E}_k^{\text{ext}}\mathbf{U},\, \mathbf{E}_k^{\text{ext}}\mathbf{V},\, \Phi_\mu(\mathtt{X}_k^{\text{hf}}),\, \Phi_\mu(\mathtt{X}_{k,\text{ext}}^{\text{hf}})\right), \tag{22b}$$

for $k = 1, \dots, N_\text{e}$. In the DG literature, schemes in which the primal unknown is only coupled with the unknowns of the adjacent elements are referred to as "compact": the BR2 flux considered in this work is an example of compact treatment of second-order terms for DG formulations (cf. [4]). Decomposition of the residual as the sum of local elemental contributions is at the foundation of the `hf` assembling and also of the hyper-reduction procedure discussed below. We emphasize that the decomposition of the facets' contributions is not unique: in order to ensure certain stability and conservation properties for the hyper-reduced ROM, we here consider the energy-stable element-wise decomposition in [69, section 3.1].

Given the ROBs $\mathbf{Z} \in \mathbb{R}^{N_\text{hf}, N}$ and $\mathbf{Y} \in \mathbb{R}^{N_\text{hf}, J_\text{es}}$, where N, J_es denote the size of the trial and test ROB respectively, with $N \leq J_\text{es}$, and the trial and test norms $\|\cdot\|$ and $|||\cdot|||$, the EQ-LSPG ROM considered in this work reads as follows: find $\widehat{\mathbf{U}}_\mu = \mathbf{Z}\widehat{\boldsymbol{\alpha}}_\mu$ to minimize

$$\min_{\boldsymbol{\zeta} \in \text{col}(\mathbf{Z})} \sup_{\boldsymbol{\eta} \in \text{col}(\mathbf{Y})} \frac{R_\mu^\text{eq}(\boldsymbol{\zeta}, \boldsymbol{\eta})}{|||\boldsymbol{\eta}|||}. \tag{23a}$$

Here, R_μ^eq is the empirical residual defined as

$$R_\mu^\text{eq}(\mathbf{U}, \mathbf{V}) \;=\; \sum_{k \in \mathtt{I}_\text{eq}} \rho_k^\text{eq} r_\mu^k(\mathbf{U}, \mathbf{V}), \quad \forall\, \mathbf{U}, \mathbf{V} \in \mathbb{R}^{N_\text{hf}}, \tag{23b}$$

where $\mathtt{I}_\text{eq} \subset \{1, \dots, N_\text{e}\}$ are the indices of the sampled elements and $\boldsymbol{\rho}^\text{eq} = [\rho_1^\text{eq}, \dots, \rho_{N_\text{e}}^\text{eq}]$ are positive empirical weights to be determined, $\rho_k^\text{eq} > 0 \Leftrightarrow k \in \mathtt{I}_\text{eq}$. Provided that the columns $[\boldsymbol{\eta}_1, \dots, \boldsymbol{\eta}_{J_\text{es}}]$ of $\mathbf{Y}$ are orthonormal with respect to the $|||\cdot|||$ norm, we can rewrite (23a) as

$$\widehat{\boldsymbol{\alpha}}_\mu \in \arg\min_{\boldsymbol{\alpha} \in \mathbb{R}^N} \|\mathbf{R}_\mu^\text{eq}(\boldsymbol{\alpha})\|_2, \quad \mathbf{R}_\mu^\text{eq}(\boldsymbol{\alpha}) = \left[R_\mu^\text{eq}(\mathbf{Z}\boldsymbol{\alpha}, \boldsymbol{\eta}_1), \dots, R_\mu^\text{eq}(\mathbf{Z}\boldsymbol{\alpha}, \boldsymbol{\eta}_{J_\text{es}})\right]. \tag{23c}$$

Note that (23c) is a nonlinear least-squares problem that can be solved using the Gauss-Newton algorithm. We initialize the iterative procedure using a non-intrusive estimate of the solution coefficients: if the number of training points is sufficiently large — such as in the case of POD data compression — we use RBF regression as in [59, 60]; for small training sets — such as in the first steps of the Greedy algorithm — we use nearest-neighbors regression. Similarly to [59], we resort to a discrete L^2 norm for the trial space and to a discrete H^1 norm for the test space: we refer to [9] for a discussion on variational formulations for first-order linear hyperbolic problems.

The MOR formulation (23) depends on the choice of the trial and test ROBs $\mathbf{Z}$ and $\mathbf{Y}$ and on the sparse vector of empirical weights $\boldsymbol{\rho}^\text{eq}$: we discuss their construction in the remainder of section 3.2. Before proceeding with the discussion, we remark that we can exploit (23b) to assemble the reduced residual $\mathbf{R}_\mu^\text{eq}$: first, we evaluate $\Phi_\mu(\mathtt{X}_k^\text{hf})$ and $\Phi_\mu(\mathtt{X}_{k,\text{ext}}^\text{hf})$ for all $k \in \mathtt{I}_\text{eq}$; then, we compute the local residuals $\{r_\mu^k(\mathbf{Z}\boldsymbol{\alpha}, \boldsymbol{\eta}_j)\}_k$ using (22b); finally, we compute $\mathbf{R}_\mu^\text{eq}(\boldsymbol{\alpha})$ by summing over the sampled elements, cf. (23b). Note that, since the residuals

$\{r_\mu^k\}_k$ are linear with respect to the test function, we can use standard element-wise residual evaluation routines to compute local contributions to the residual. Furthermore, we observe that computation of the residual $\mathbf{R}_\mu^{\mathrm{eq}}(\boldsymbol{\alpha})$ requires the storage of trial and test ROBs in the sampled elements and in their neighbors, and is thus independent of the total number of mesh elements. We refer to [60] for further details.

Remark 3.3. ***Treatment of geometry parameterization.*** *We here resort to a discretize-then-map (DtM, [17, 60, 65]) treatment of parameterized geometries. As discussed in [60], the DtM approach — as opposed to the more standard map-then-discretize (MtD, [33, 53, 2, 3, 52])) approach — in combination with EQ allows to reuse* `hf` *local integration routines and is thus considerably easier to implement, particularly for nonlinear PDEs. In this approach, the mapping Φ_μ is evaluated at the beginning of the online stage to deform the nodes of the sampled elements of the mesh and its neighbors (cf.* (22b)*). As a result, online costs are nearly independent of the size M of the mapping space. On the other hand, we should store the mapping ROB W_M in the reference nodes of all sampled elements and its neighbors: as a result, memory costs scale with $(dM + D(N + J_{\mathrm{es}}))n_{\mathrm{lp}}Q$ — where $d = 2$ is the space dimension, $D = 4$ is the number of equations, Q is the number of sampled elements, and n_{lp} is the number of degrees of freedom in each element.*

3.2.1 Construction of trial and test spaces

We resort to the standard data compression algorithms POD and weak-Greedy to build the trial ROB $\mathbf{Z}$. For stability reasons, we ensure that the columns $\boldsymbol{\zeta}_1, \ldots, \boldsymbol{\zeta}_N$ of $\mathbf{Z}$ are orthonormal with respect to the $\|\cdot\|$ norm. We anticipate that, for the problem considered in this paper, POD leads to superior performance (cf. section 4) in terms of online accuracy; however, POD requires more extensive explorations of the parameter domain and is thus more onerous during the offline stage. For this reason, in section 3.3, we resort to the weak-Greedy method in combination with multi-fidelity training to reduce offline costs. We refer to the monographies [28, 48] for extensive discussions on POD and weak-Greedy data compression.

For completeness, we report in Algorithm 2 the weak-greedy algorithm as implemented in our code. Note that the algorithm takes as input the mesh $\mathcal{T}_{\mathrm{hf}}$ and the mapping Φ which define the FE mesh for all parameters, and returns the ROB $\mathbf{Z}$ and the ROM for the solution coefficients. The residual indicator is presented in section 3.2.3. The function `Gram-Schmidt` at Line 4 performs one step of the Gram Schmidt process to ensure that the trial ROB is orthonormal with respect to the $\|\cdot\|$ norm. Construction of the ROM at Line 5 involves the construction of the test ROB $\mathbf{Y}$ and the computation of the empirical quadrature rule: these procedures are described below.

Algorithm 2 Weak-greedy algorithm.

Inputs: $\mathcal{P}_{\text{train}} := \{\mu^k\}_{k=1}^{n_{\text{train}}}$ training parameter set, $\Phi : \Omega \times \mathcal{P} \to \mathbb{R}^2$ mapping; $\mathcal{T}_{\text{hf}}$ mesh.

Outputs: $\mathbf{Z}$ trial ROB; $\mu \in \mathcal{P} \mapsto \widehat{\boldsymbol{\alpha}}_\mu$ ROM for the solution coefficients.

Offline stage

Choose $\mu^{\star,1} = \bar{\mu}$.
for $N = 1, \dots, N_{\max}$ **do**
  Solve the `hf` problem for $\mu = \mu^{\star,N}$ to obtain $\mathbf{U}^\star = \mathbf{U}_{\mu^{\star,N}}$.
  Update the ROB $\mathbf{Z} = \texttt{Gram-Schmidt}(\mathbf{Z}, \mathbf{U}^\star, \|\cdot\|)$.
  Build the ROM $\mu \in \mathcal{P} \mapsto \widehat{\boldsymbol{\alpha}}_\mu$.
  for $k = 1, \dots, n_{\text{train}}$ **do**
    Estimate the solution using the ROM for $\mu = \mu^k$.
    Compute the error indicator $\Delta_{\mu^k} := \mathfrak{R}_{\mu^k}(\widehat{\boldsymbol{\alpha}}_{\mu^k})$ (cf. section 3.2.3).
  end for
  Set $\mu^{\star,N+1} = \arg\max_{\mu \in \mathcal{P}_{\text{train}}} \Delta_\mu$.
end for

As rigorously proven in [59, Appendix C] for linear inf-sup stable problems, the test ROB $\mathbf{Y}$ should approximate the Riesz representers of the Fréchet derivative of the residual at $\mathbf{U}_\mu^{\text{hf}}$ applied to the elements of the trial ROB for all $\mu \in \mathcal{P}$. Similarly to [60], we here resort to the sampling strategy based on POD proposed in [59]: first, given the $\mathcal{Y}$ inner product $((\cdot,\cdot))$ such that $|||\cdot||| = \sqrt{((\cdot,\cdot))}$, we compute the Riesz representers of the Fréchet derivative of the residual at $\mathbf{U}_\mu^{\text{hf}}$, evaluated for the elements of the n-th trial bases $\boldsymbol{\zeta}_n$ and for the k-th parameter μ^k in the training set,

$$((\boldsymbol{\psi}_{k,n}, \mathbf{v})) \;=\; DR_\mu^{\text{hf}}[\mathbf{U}_\mu^{\text{hf}}](\boldsymbol{\zeta}_n, \mathbf{v}), \quad \forall\, \mathbf{v} \in \mathbb{R}^{N_{\text{hf}}},$$

for $n = 1, \dots, N$, $k = 1, \dots, n_{\text{train}}$; then, we apply POD for a given tolerance $tol_{\text{test}} > 0$ to find the test ROB $\mathbf{Y}$,

$$[\mathbf{Y}, \cdot] = \texttt{POD}\left(\{\boldsymbol{\psi}_{k,n}\}_{k,n}, tol_{\text{test}}, |||\cdot|||\right).$$

The POD tolerance should be sufficiently tight to ensure the well-posedness of the reduced problem: in the numerical tests of section 4, we set $tol_{\text{test}} = 10^{-3}$.

3.2.2 Empirical quadrature

As in [60], we seek $\boldsymbol{\rho}^{\text{eq}} \in \mathbb{R}_+^{N_{\text{e}}}$ such that (i) the number of nonzero entries in $\boldsymbol{\rho}^{\text{eq}}$, $\|\boldsymbol{\rho}^{\text{eq}}\|_{\ell^0}$, is as small as possible; (ii, *constant function constraint*) the constant function is approximated correctly in Ω (i.e., $\Phi = \texttt{id}$),

$$\left| \sum_{k=1}^{N_{\text{e}}} \rho_k^{\text{eq}} |\mathtt{D}_k| \;-\; |\Omega| \right| \ll 1; \tag{24}$$

(iii, *manifold accuracy constraint*) for all $\mu \in \mathcal{P}_{\text{train,eq}} = \{\mu^k\}_{k=1}^{n_{\text{train}}+n_{\text{train,eq}}}$, the empirical residual satisfies

$$\left\| \mathbf{R}_\mu^{\text{hf}}(\boldsymbol{\alpha}_\mu^{\text{train}}) \; - \; \mathbf{R}_\mu^{\text{eq}}(\boldsymbol{\alpha}_\mu^{\text{train}}) \right\|_2 \ll 1. \tag{25a}$$

where $\mathbf{R}_\mu^{\text{hf}}$ corresponds to substitute $\rho_1^{\text{eq}} = \ldots = \rho_{N_{\text{e}}}^{\text{eq}} = 1$ in (23b) and $\boldsymbol{\alpha}_\mu^{\text{train}}$ satisfies

$$\boldsymbol{\alpha}_\mu^{\text{train}} = \begin{cases} \mathbf{Z}^T \mathbf{X}_{\text{hf}} \mathbf{U}_\mu^{\text{hf}} & \text{if } \mu \in \mathcal{P}_{\text{train}}; \\ \arg\min\limits_{\boldsymbol{\alpha} \in \mathbb{R}^N} \|\mathbf{R}_\mu^{\text{hf}}(\boldsymbol{\alpha})\|_2, & \text{if } \mu \notin \mathcal{P}_{\text{train}}. \end{cases} \tag{25b}$$

Here, $\mathbf{X}_{\text{hf}}$ is the matrix associated with the $(\cdot,\cdot)$ inner product and $\mathcal{P}_{\text{train}} = \{\mu^k\}_{k=1}^{n_{\text{train}}}$ is the set of parameters for which the `hf` solution is available. When we apply POD to generate the ROM, we set $\mathcal{P}_{\text{train}} = \mathcal{P}_{\text{train,eq}}$; when we apply the weak-Greedy algorithm, we augment $\mathcal{P}_{\text{train}}$ with $n_{\text{train,eq}} = 10$ randomly-selected parameters (see [69, Algorithm 1]): we empirically observe that this choice improves performance of the hyper-reduced ROM, particularly for small values of n_{train}. We refer to the above-mentioned literature for a thorough motivation of the previous constraints; in particular, we refer to [14, 69] for a discussion on the conservation properties of the ROM for conservation laws.

It is possible to show (see, e.g., [59]) that (i)-(ii)-(iii) lead to a sparse representation problem of the form

$$\min_{\boldsymbol{\rho} \in \mathbb{R}^{N_{\text{e}}}} \|\boldsymbol{\rho}\|_{\ell^0}, \quad \text{s.t} \quad \begin{cases} \|\mathbf{G}\boldsymbol{\rho} - \mathbf{b}\|_2 \leq \delta; \\ \boldsymbol{\rho} \geq \mathbf{0}; \end{cases} \tag{26}$$

for a suitable threshold $\delta > 0$, and for a suitable choice of $\mathbf{G}, \mathbf{b}$. Following [22], we here resort to the non-negative least-squares method to find approximate solutions to (26). In particular, we use the Matlab function `lssnonneq`, which takes as input the pair $(\mathbf{G}, \mathbf{b})$ and a tolerance $tol_{\text{eq}} > 0$ and returns the sparse vector $\boldsymbol{\rho}^{\text{eq}}$,

$$[\boldsymbol{\rho}^{\text{eq}}] = \texttt{lsqnonneg}\,(\mathbf{G}, \mathbf{b}, tol_{\text{eq}})\,. \tag{27}$$

We refer to [15] for an efficient implementation of the non-negative least-squares method for large-scale problems.

3.2.3 Dual residual estimation

We here resort to the dual residual error indicator

$$\mathfrak{R}_\mu^{\text{hf}}(\boldsymbol{\alpha}) \; := \; \sup_{\mathbf{v} \in \mathbb{R}^{N_{\text{hf}}}} \; \frac{R_\mu^{\text{hf}}(\mathbf{Z}\boldsymbol{\alpha},\, \mathbf{v})}{|||\mathbf{v}|||}, \quad \boldsymbol{\alpha} \in \mathbb{R}^N, \tag{28}$$

to drive the weak-Greedy algorithm. If we denote by $\mathbf{Y}_{\text{hf}}$ the matrix associated with the $|||\cdot|||$ norm, we have that

$$\mathfrak{R}_\mu^{\text{hf}}(\boldsymbol{\alpha}) \; := \; \sqrt{\mathbf{R}_\mu^{\text{hf}}(\mathbf{Z}\boldsymbol{\alpha})^T \, \mathbf{Y}_{\text{hf}}^{-1} \, \mathbf{R}_\mu^{\text{hf}}(\mathbf{Z}\boldsymbol{\alpha})}, \quad \forall\, \mu \in \mathcal{P}, \boldsymbol{\alpha} \in \mathbb{R}^N.$$

Computation of $\mathfrak{R}_\mu^{\text{hf}}(\boldsymbol{\alpha})$ thus requires to assemble the `hf` residual $\mathbf{R}_\mu(\boldsymbol{\alpha}) \in \mathbb{R}^{N_{\text{hf}}}$ and then solve a linear problem of size N_{hf}. Since the matrix $\mathbf{Y}_{\text{hf}}$ is symmetric positive definite and parameter-independent, we use Cholesky factorization to

speed up computations of the inner loop in Algorithm 2 — we further use the Matlab function `symamd` to reduce fill-in.

In the numerical results (cf. A), we show that $\mathfrak{R}^{\mathrm{hf}}_{\mu}(\cdot)$ is highly correlated with the relative error. In order to use $\mathfrak{R}^{\mathrm{hf}}_{\mu}(\cdot)$ during the online stage, we shall perform hyper-reduction: we refer to [60] for the details. In our experience, for the value of n_{train} and for the particular `hf` discretization considered, the cost of the greedy search in Algorithm 2 is negligible compared to the cost of an `hf` solve; as a result, hyper-reduction does not seem needed during the offline stage.

3.3 Offline/online computational decomposition based on two-fidelity sampling

As discussed in section 3.1, the registration procedure relies on a regression algorithm to compute the mapping coefficients $\widehat{\mathbf{a}}_{\mu}$ for out-of-sample parameters. Since the regression algorithm does not explicitly ensure that bijectivity is satisfied for out-of-sample parameters, in practice we should consider sufficiently large training sets $\mathcal{P}_{\mathrm{train}}$. To address this issue, we propose to use a multi-fidelity approach, which relies on $n_{\mathrm{train,c}} \in \mathbb{N}$ `hf` solves on a coarser grid to learn the parametric mapping Φ. Algorithm 3 summarizes the offline/online procedure implemented in our code.

We state below several remarks.

- The snapshots $\{U^{\mathrm{hf,c}}_{\mu^k}\}_{k=1}^{n_{\mathrm{train,c}}}$ — the superscript $(\cdot)^{\mathrm{train,c}}$ highlights the fact that computations are performed on the coarser grid — are exclusively used to compute the sensors $\{s_{\mu^k}\}_{k=1}^{n_{\mathrm{train,c}}}$ that are then fed into the registration algorithm: we might then employ snapshots from third-party solvers and we might also use different grids for different parameters.

- In this work, we propose to build the fine mesh $\mathcal{T}_{\mathrm{hf}}$ based on the coarse snapshot $U^{\mathrm{hf,c}}_{\bar{\mu}}$; we use here the open source mesh generator proposed in [47] based on a suitable relative size function: we provide details concerning the definition of the size function in B. As anticipated in the introduction, we expect that for more challenging problems it might be necessary to adapt the mesh based on multiple snapshots.

- Computation of the ROB $\mathbf{Z}$ and of the ROM for the solution coefficients and the online evaluation can be performed using standard pMOR algorithms for linear approximations in parameterized geometries: we believe that this represents a valuable feature of the proposed approach that allows its immediate application to a broad class of problems.

- Our multi-fidelity procedure does not include any update of the sensors as more accurate simulations become available during Step 5 of the offline stage: as a result, it might lead to poor results if the initial discretization is excessively inaccurate. In this work, we perform a convergence analysis for a single parameter value, the centroid $\bar{\mu} = [0.775, 1.75]$, to identify the hierarchy of two meshes $\mathcal{T}_{\mathrm{hf,c}}$ and $\mathcal{T}_{\mathrm{hf}}$. Development of more sophisticated multi-fidelity techniques is the subject of ongoing research.

Algorithm 3 Offline online algorithm.

Offline stage

1: Generate the snapshots $\{U^{\mathrm{hf,c}}_{\mu^k}\}_{k=1}^{n_{\mathrm{train,c}}}$ based on the grid $\mathcal{T}_{\mathrm{hf,c}}$ and the mapping Φ^{geo}.

2: Use the snapshots $\{U^{\mathrm{hf,c}}_{\mu^k}\}_{k=1}^{n_{\mathrm{train,c}}}$ to compute the sensors $\{s_{\mu^k}\}_{k=1}^{n_{\mathrm{train,c}}}$ using (19).

3: Generate the fine mesh $\mathcal{T}_{\mathrm{hf}}$.

4: Apply registration (cf. Algorithm 1) based on $\{s_{\mu^k}\}_{k=1}^{n_{\mathrm{train,c}}}$ and the mesh $\mathcal{T}_{\mathrm{hf}}$.

5: Generate the ROB $\mathbf{Z}$ and the ROM for the solution coefficients $\mu \in \mathcal{P} \mapsto \widehat{\boldsymbol{\alpha}}_\mu \in \mathbb{R}^N$.

Online stage (for any given $\mu \in \mathcal{P}$)

1: Solve the ROM to compute $\widehat{\boldsymbol{\alpha}}_\mu$.

2: Compute the deformed mesh $\Phi_\mu(\mathcal{T}_{\mathrm{hf}})$ and $\widehat{\mathbf{U}}_\mu = \mathbf{Z}\widehat{\boldsymbol{\alpha}}_\mu$.

4 Numerical results

We present below extensive numerical investigations for the model problem introduced in section 2. Further numerical tests are provided in A.

4.1 Test 1: single-fidelity training

In this first test, we consider performance of our approach without multi-fidelity training. Towards this end, we consider a p=2 DG FE discretization with $N_{\mathrm{hf}} = 197856$ degrees of freedom ($N_{\mathrm{e}} = 8204$): the FE mesh is depicted in Figure 5(a). The mesh is refined for $x_1 > -0.5$ where we expect to have shocks caused by the interaction between the uniform flow and the bump; further refinement is performed at the leading edge of the bump. For completeness, we provide further details about the construction of the mesh in B.

We consider an equispaced grid of 11×11 parameters $\mathcal{P}_{\mathrm{train}} := \{\mu^k\}_{k=1}^{n_{\mathrm{train}}} \subset \mathcal{P}$ ($n_{\mathrm{train}} = 121$); we further consider $n_{\mathrm{test}} = 10$ randomly-selected parameters for testing. We measure performance of the ROM in terms of the average out-of-sample relative prediction error :

$$E_{\mathrm{avg}} := \frac{1}{n_{\mathrm{test}}} \sum_{\mu\in\mathcal{P}_{\mathrm{test}}} \frac{\|U^{\mathrm{hf}}_\mu - \widehat{U}^{\mathrm{hf}}_\mu\|_{L^2(\Omega_\mu)}}{\|U^{\mathrm{hf}}_\mu\|_{L^2(\Omega_\mu)}}. \tag{29}$$

The mapping Φ that is obtained applying the registration procedure in Algorithm 1 consists of three modes ($M = 3$): the R-squared associated with the RBF regressors is above the threshold for all three modes.

Figure 3 shows performance of linear and Lagrangian approaches based on POD data compression. Figure 3(a) shows the projection error, while Figure 3(b) shows the error associated with the EQ-LSPG ROM introduced in section 3.2. We observe that registration significantly improves performance for all val-

ues of N. Figure 4 replicates the results for the ROM based on weak-Greedy[2] compression: note that also in this case registration significantly improves performance for all values of N considered. We further observe that our EQ-LSPG ROM is able to achieve near-optimal performance compared to projection for both linear and Lagrangian approaches and for both POD and Greedy compression.

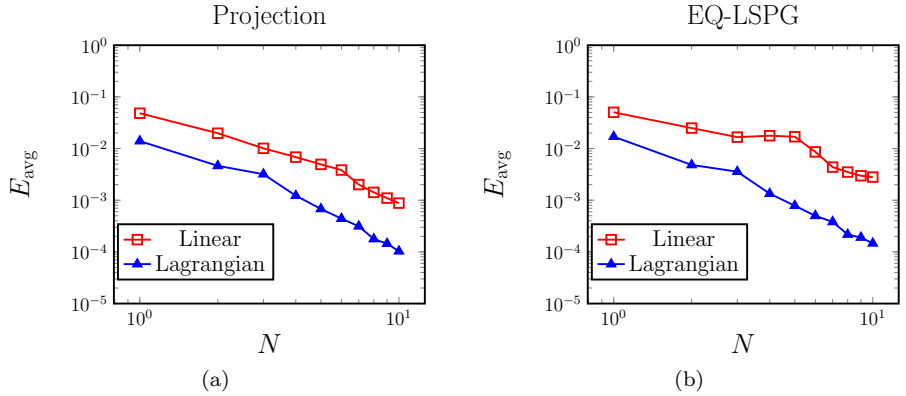


Figure 3: single-fidelity training. Comparison of linear and Lagrangian approaches. Trial ROB $\mathbf{Z}$ is built using POD.

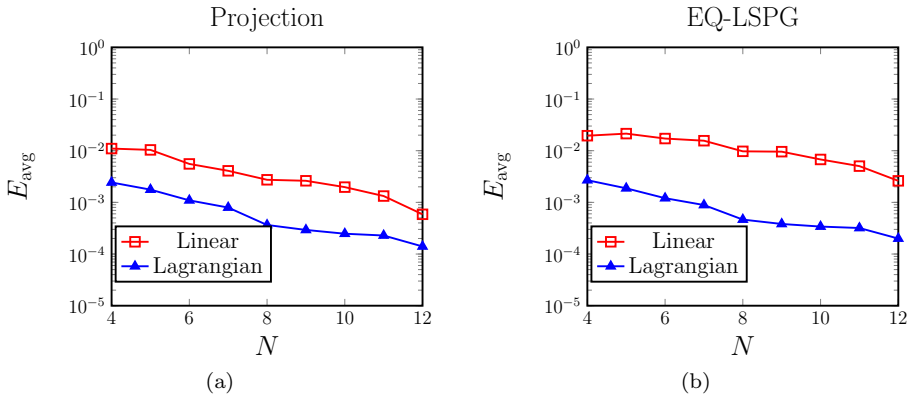


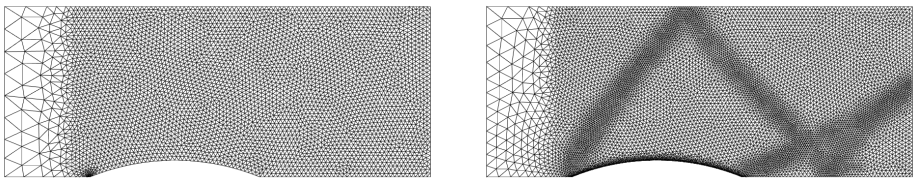
Figure 4: single-fidelity training. Comparison of linear and Lagrangian approaches. Trial ROB $\mathbf{Z}$ is built using weak-Greedy.

4.2 Test 2: multi-fidelity training

We now validate the full offline/online algorithm presented in section 3.3: towards this end, we consider the same `hf` discretization and parameter set $\mathcal{P}_{\text{train}}$ considered in the previous section to compute the mapping Φ (i.e., $N_{\text{hf}} =$

[2]We initialize the Greedy procedure with $N_0 = 4$ equispaced samples. The Greedy search is performed over the training set of $n_{\text{train}} = 121$ parameters.

197856, $N_{\mathrm{e}} = 8204$); on the other hand, we use the refined grid depicted in Figure 5(b) with $N_{\mathrm{hf}} = 402048$ ($N_{\mathrm{e}} = 16752$) to generate the `hf` snapshots. The mesh is refined based on the `hf` solution for $\bar{\mu} = [0.775, 1.75]$ as discussed in B.

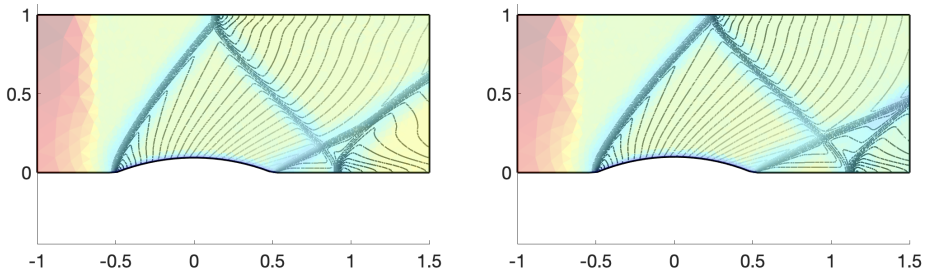


(a) (b)

Figure 5: multi-fidelity training. (a) coarse mesh used for sensor generation. (b) fine mesh used for MOR calculations.

As in the previous case, the mapping Φ that is obtained applying the registration procedure in Algorithm 1 consists of three modes ($M = 3$); all three mapping coefficients are well-approximated through RBF regression. Note that the mapping considered in this test differs from the one in the previous test due to the fact that Algorithm 1 is fed with a different mesh. Nevertheless, we find that the differences between the two mappings are moderate.

In Figure 6, we investigate the ability of the parametric mesh $\Phi_\mu(\mathcal{T}_{\mathrm{hf}})$ to track the sharp gradient regions. More in detail, in the background we show the mesh density $\log_{10}(h_\mu)$; in the foreground we show the contour lines of the Mach number, for $\mu_{\min} = [0.75, 1.7]$ and $\mu_{\max} = [0.8, 1.8]$. Here, the mesh density is defined as $h_\mu(x) := \sqrt{|\mathtt{D}_{k,\Phi_\mu}|}$ if $x \in \mathtt{D}_{k,\Phi_\mu}$, where $\mathtt{D}_{k,\Phi_\mu}$ is the k-th element of the mesh $\Phi_\mu(\mathcal{T}_{\mathrm{hf}})$. We observe that the mesh "follows" the shocks of the solution field: registration is thus able to correctly deform the mesh to track relevant features of the parametric field.



(a) $\mu_{\min}$ (b) $\mu_{\max}$

Figure 6: multi-fidelity training. Comparison of contour lines of Mach number and mesh density $\log_{10}(h)$ for two values of the parameter.

In Figure 7, we show performance of EQ-LSPG for POD (based on $n_{\mathrm{train}} = 121$ snapshots) and weak-Greedy data compression; to facilitate interpretation, we further report the average error of the coarse solver. We observe that also in this case the ROM is able to provide accurate predictions for extremely moderate

values of the ROB size N. In particular, EQ-LSPG with weak-Greedy sampling is able to achieve average out-of-sample errors below 10^{-3} with only $N = 12$ `hf` solves.

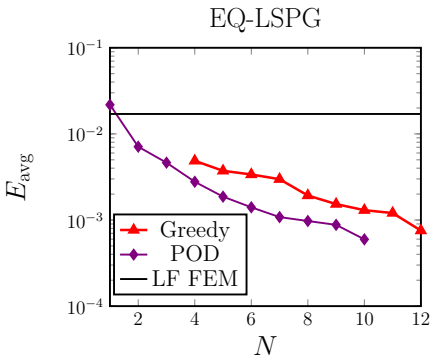


Figure 7: multi-fidelity training. Performance of EQ-LSPG for POD and weak-Greedy data compression.

4.3 Computational costs

In Table 1, we provide an overview of the offline computational costs; all simulations are performed using Matlab 2020b on a commodity desktop computer (Intel i7-8700 @3.20 GHz x 12, RAM 32 Gb). We state upfront that all computations are performed in serial; we further observe that assembly of `hf` structures is fully vectorized. To compute the solution on the fine mesh, we initialize the implicit solver using the coarse solution: the computational cost in Table 1 comprises both computation of the initial condition — ≈ 650 [s] — and computation of the `hf` solution — ≈ 220 [s]. We decompose the costs of the Greedy algorithm in costs of building the ROM (construction of trial and test ROBs, construction of the reduced quadrature rule) and costs of looping over the training set (i.e., greedy search). As explained in section 4, we initialize the Greedy procedure with $N_0 = 4$ equispaced parameters: we thus perform 8 Greedy searches and 9 ROM constructions; the ROM construction for $N = 12$ costs 580 [s]. In conclusion, we can estimate the cost of single and two-fidelity approaches as follows:

$$\text{single fidelity} = 30 + 110 + (650 + 220) \cdot 121 + 190 + 4 \cdot 10^3 + 580 \approx 1.10 \cdot 10^5$$

$$\text{two fidelity} = 30 + 110 + (650) * 121 + 12 * 220 + 190 + 4000 + 1500 + 3800 \approx 0.91 \cdot 10^5.$$

which corresponds a cost reduction of approximately 17.3%.

In Figure 8, we compare the average online costs of linear and Lagrangian ROMs obtained using the weak-greedy algorithm for the single-fidelity case. We observe that the linear ROM is slightly less efficient than the Lagrangian ROM. The difference in cost is attributable to (i) the different size of the test ROB (cf. Figure 9) and (ii) the different number of Gauss-Newton iterations required to meet the convergence criterion. We recall that residual and Jacobian

	unit cost [s]	total cost [s]
Definition of the coarse mesh		30
Coarse solutions to Euler equations	650	$7.8 \cdot 10^4$
Definition of the fine mesh	110	110
Computation of the registration sensors	1.6	190
Registration algorithm		$4.0 \cdot 10^3$
Greedy algorithm (greedy search)		$1.5 \cdot 10^3$
Greedy algorithm (ROM construction)		$3.8 \cdot 10^3$
Greedy algorithm (`hf` solves)	$650 + 220$	$(7.8 + 2.6) \cdot 10^3$

Table 1: offline computational cost; Greedy algorithm is run from $N_0 = 4$ to $N_{\max} = 12$.

assembly costs scale linearly with J_{es}; we further observe that the increased number of Gauss-Newton iterations is due to the drop in accuracy of the non-intrusive estimate based on nearest neighbor regression that is used to initialize the iterative procedure.

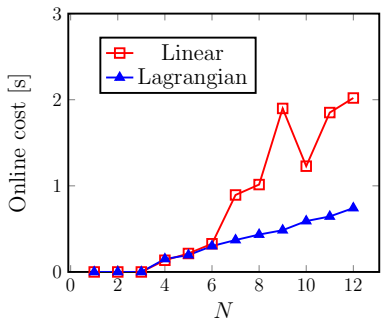


Figure 8: average online computational cost over $n_{\text{test}} = 10$ out-of-sample parameters.

4.4 Discussion

Numerical results clearly indicate the superiority of Lagrangian approximations based on registration to standard linear methods for problems with shock (and contact) discontinuities. By tracking the position of shocks, registration improves the linear compressibility of the solution manifold and thus facilitates the task of model reduction. The combination of multifidelity training and registration provides a framework to generate parsimonious `hf` discretizations that are valid for all parameters (cf. Figure 6) and is beneficial in terms of overall offline costs (cf. section 4.3).

However, the approach suffers from several limitations that need to be addressed. First, registration relies on the fact that shocks are smooth functions of the parameter: in particular, shock topology should be approximately the same for all parameters in $\mathcal{P}$. Second, as discussed in section 3.3, our multi-fidelity approach does not include a feedback control on the accuracy of the coarse simulations: it thus implicitly relies on the fact that the coarse discretization is

sufficiently accurate to approximately locate shocks.

5 Conclusions

In this work, we developed and numerically assessed a multi-fidelity projection- and registration-based MOR procedure for two-dimensional hyperbolic PDEs in presence of shocks. The key features of our approach are (i) a general (i.e., independent of the underlying PDE) registration procedure for the computation of the mapping Φ that tracks moving features of the solution field; (ii) an hyper-reduced LSPG ROM for the computation of the solution coefficients; and (iii) a multi-fidelity approach based on coarse simulations to train the mapping Φ and Greedy sampling in parameter, to reduce offline costs. We illustrate the many pieces of our formulation through the vehicle of a supersonic inviscid flow past a bump.

We wish to extend the present work in several directions. First, we wish to to deal with parametric variations of the shock topology: towards this end, we wish to extend the approach in [19] to registration-based approximations. Second, we wish to devise robust multi-fidelity strategies that are able to correct the inaccuracies of the coarse simulations. Third, we wish to relax the bijectivity-in-Ω constraint in the registration algorithm by suitably extending the field outside the domain of interest: this would allow to increase the flexibility of our approach — particularly, in the presence of fictitious boundaries in the computational domains — and ultimately improve performance. Fourth, as stated in the introduction, we wish to combine our r-type, registration-based, parametric mesh adaptivity technique with h-type adaptivity.

Acknowledgements

The authors thank Professor Angelo Iollo (Inria Bordeaux), Dr. Cédric Goeury and Dr. Angélique Ponçot (EDF) for fruitful discussions. The authors acknowledge the support by European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie Actions, grant agreement 872442 (ARIA). Tommaso Taddei also acknowledges the support of IdEx Bordeaux (projet EMERGENCE 2019).

A Further numerical investigations

We present here further numerical results to better illustrate the performance of our method. We state upfront that in the results of Figures 9, 10, 11, we show results for POD data compression.

In Figure 9 we show the size of the test ROB $\mathbf{Y}$ as obtained using the Algorithm described in section 3.2 for both linear and Lagrangian ROMs. We observe that J_{es} is considerably larger for the linear ROM: registration thus also helps reduce the size of the test space required for stability.

Figure 10 investigates performance of the hyper-reduction procedure: we show the behavior of the out-of-sample error E_{avg} for different EQ tolerances in (27); we further show the percentage of sampled elements $Q/N_{\text{e}} \cdot 100$ selected by the EQ procedure. We remark that EQ ensures accurate performance for

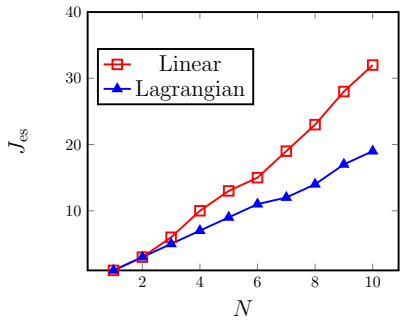


Figure 9: single-fidelity training; size of the empirical test space for $tol_{es} = 10^{-3}$ for linear and Lagrangian ROMs.

$tol_{eq} \leq 10^{-10}$ for all values of N considered and for both linear and Lagrangian ROMs. Interestingly, the linear ROM requires slightly more sampled elements: we conjecture that this is due to the larger size of the test space.

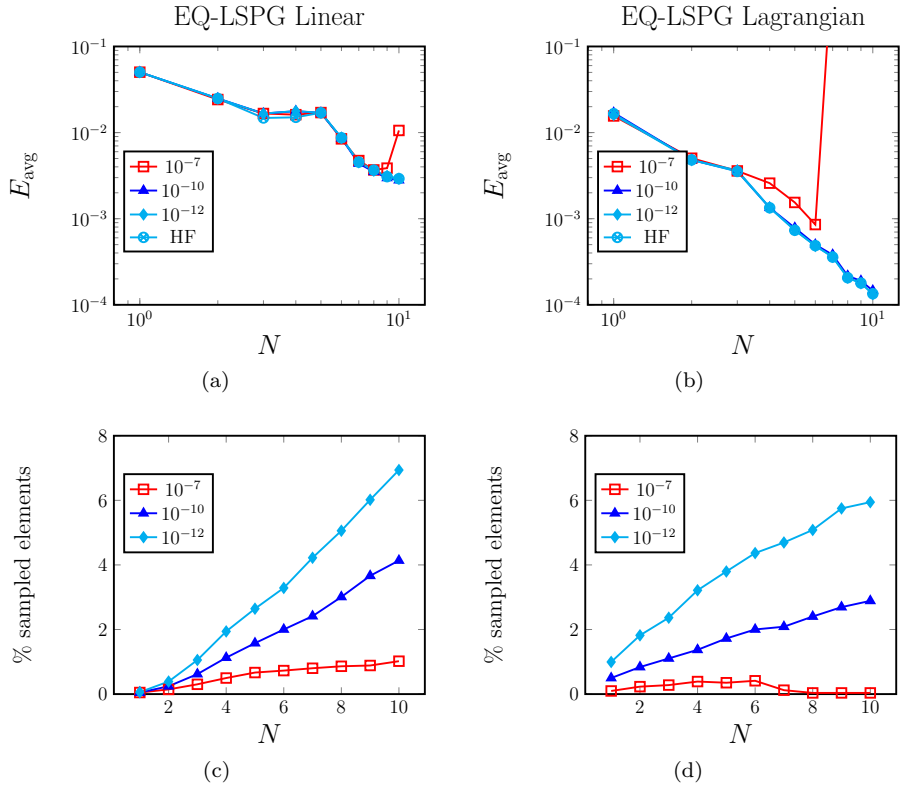


Figure 10: single-fidelity training; hyper-reduction for linear and Lagrangian ROMs. (a)-(b) behavior of relative error E_{avg} for various tolerances tol_{eq} (cf. (27)). (c)-(d) percentage of sampled elements $Q/N_e \cdot 100$ for the same tolerances tol_{eq}.

In Figure 11, we illustrate the effect of discretization on hyper-reduction: we show the percentage of sampled elements $Q/N_e \cdot 100$ selected by the EQ procedure for two tolerances, several values of the trial ROB size N, and for the two meshes considered in this work (cf. Figure 5). We find that the absolute value of sampled elements weakly depends on the underlying FE mesh; as a result, hyper-reduction becomes more and more effective as N_e increases.

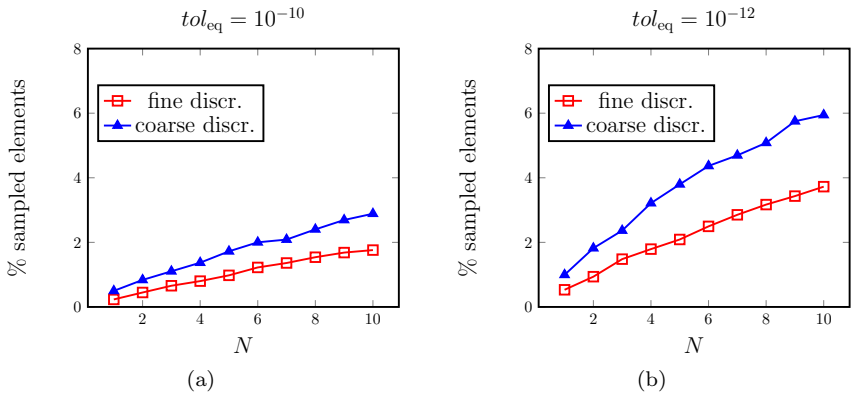


Figure 11: effect of discretization on hyper-reduction. Percentage of sampled elements $Q/N_e \cdot 100$ for two tolerances tol_{eq} and for fine and coarse discretizations (cf. Figure 5).

In Figure 12, we investigate the relationship between dual residual (28) and relative L^2 error for linear and Lagrangian ROMs. More precisely, during each step of the weak-greedy algorithm, we compute both dual residual and relative L^2 error for all training points; then, we show the results for all $N = 4, \ldots, 12$. We observe that there is a strong correlation between error and dual residual: this motivates the use of dual residual norm to drive the Greedy algorithm and also as error indicator during the online stage. We remark that the points associated with the relative error below 10^{-5} correspond to parameters that are sampled by the greedy procedure (see Algorithm 2).

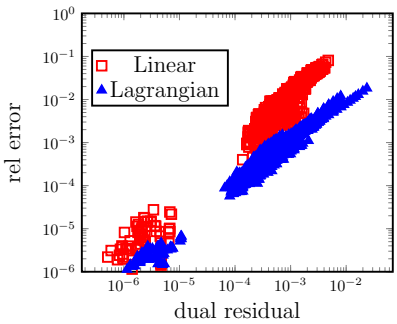


Figure 12: single-fidelity training; dual residual norm estimation. Comparison between dual residual norm and exact relative error for various ROMs and $\mu \in \mathcal{P}_{\text{train}}$.

In Figure 13, we study the sensitivity of the registration procedure to the choice of ξ and ξ_{msh} in Algorithm 1. First, we solve (17) with $\mathcal{S}_{N=1} = \text{span}\{s_{\bar{\mu}}\}$ with $\bar{\mu} = [0.775, 1.75]$, $\mathcal{W}_M = \mathcal{W}_{\text{hf}}$ (i.e., $M = 388$), and $\mu = [0.75, 1.8]$ for several values of ξ and $\xi = \xi_{\text{msh}}$; in Figure 13(a), we show the values of the proximity measure, H^2 seminorm, and $\mathfrak{R}_{\text{msh}}$ for the optimal solutions $\{\widehat{\mathbf{a}}_{\mu,\xi_k}\}_k$. The parameter μ maximizes the proximity measure at the first iteration of the greedy procedure (cf. Line 7). More in detail, given $10^{-8} = \xi_1 < \ldots < \xi_{N_\xi} = 10^{-1}$ with $N_\xi = 15$, we first solve (17a) for $\xi = \xi_{\text{msh}} = \xi_{N_\xi}$ and initial condition $\mathbf{a}_0 = \mathbf{0}$; then, for $\xi = \xi_{\text{msh}} = \xi_k$, we prescribe $\mathbf{a}_0 = \widehat{\mathbf{a}}_{\mu,\xi_{k+1}}$: this choice ensures that the proximity measure is monotonic increasing in ξ. As expected, $\mathfrak{R}_{\text{msh}}(\widehat{\mathbf{a}}_{\mu,\xi};\mu)$ and $\|W_M \widehat{\mathbf{a}}_{\mu,\xi}\|_{H^2(\widehat{\Omega})}$ are monotonic decreasing in ξ.

Second, in Figure 13(b), we compare performance of Algorithm 1 for $\xi = \xi_{\text{msh}} = 10^{-3}$ and $\xi = \xi_{\text{msh}} = 10^{-5}$. As discussed in the main body of the paper, for $\xi = \xi_{\text{msh}} = 10^{-3}$, the registration algorithm terminates after three iterations and returns $M = 3$ modes whose coefficients are all significant (i.e., $\mathtt{R}^2_m > 0.75$ for $m = 1, \ldots, M$); for $\xi = \xi_{\text{msh}} = 10^{-5}$, the registration algorithm terminates after two iterations and returns $M = 8$ modes whose coefficients are all significant. In Figure 13(b), we show the behavior of the out-of-sample relative L^2 error (cf. (29)) for projection associated with the POD spaces: interestingly, the choice $\xi = \xi_{\text{msh}} = 10^{-5}$ is beneficial for $N = 1, 2$, but it is significantly suboptimal for larger values of N. This empirical result is coherent with the results in [57, Figure 5] and with the discussion in Remark 3.1.

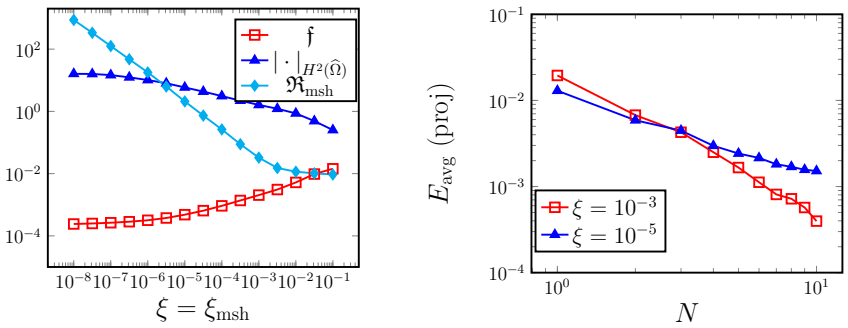


Figure 13: sensitivity to choice of hyper-parameters in Algorithm 1. (a) behavior of proximity measure $\mathfrak{f}(\widehat{\mathbf{a}}_{\mu,\xi};\mu)$, H^2 seminorm $\|W_M \widehat{\mathbf{a}}_{\mu,\xi}\|_{H^2(\widehat{\Omega})}$, and $\mathfrak{R}_{\text{msh}}(\widehat{\mathbf{a}}_{\mu,\xi};\mu)$ for the optimal solutions $\widehat{\mathbf{a}}_{\mu,\xi}$ to (17a) for $\mathcal{S}_{N=1} = \text{span}\{s_{\bar{\mu}}\}$ with $\bar{\mu} = [0.775, 1.75]$, $\mathcal{W}_M = \mathcal{W}_{\text{hf}}$ (i.e., $M = 388$), and $\mu = [0.75, 1.8]$, for several values of ξ and $\xi = \xi_{\text{msh}}$. (b) behavior of the projection relative error of Lagrangian approximation for two choices of ξ and $\xi = \xi_{\text{msh}}$ in Algorithm 1.

B Mesh generation

For completeness, we provide the definition of the mesh size function employed to generate the meshes in Figures 5(a) and 5(b). We here use the Matlab suite `distmesh` that relies on the definitions of a distance function to identify

the domain and a size function for the definition of the grid: we refer to the documentation available at `persson.berkeley.edu/distmesh/` for further details. We envision that the present approach might be greatly improved both in terms of accuracy and in terms of offline computational costs; the use of state-of-the-art adaptive FE techniques might also be important to automatize the refinement procedure.

In order to define the coarse grid, we define the size function

$$h(x) = \min\left\{h_0 + 0.5\text{dist}(x, \Omega_{\text{lead}}),\ 6h_0 + 0.6\text{dist}(x, \Omega_{\text{wake}})\right\},$$

where $h_0 = 0.008$, $\Omega_{\text{wake}} = \{x \in \Omega : x_1 > -0.6\}$ and $\Omega_{\text{lead}} = \{x \in \Omega : \|x - x_{\text{lead}}\|_2 \leq 0.5h_0\}$. Then, we define the coarse mesh $\mathcal{T}_{\text{hf,c}}$ in Figure 5(a) using `distmesh` followed by one iteration of uniform refinement (see the `distmesh` routine `uniref`).

Given the coarse simulation $(\mathcal{T}_{\text{hf,c}}, \mathbf{U}_{\bar{\mu}}^{\text{hf,c}})$, we define the Mach number $\text{Ma}^{\text{hf,c}}$ and we compute the local averages $\mathfrak{s}_1^{\text{c}}, \ldots, \mathfrak{s}_{N_{\text{e}}^{\text{c}}}^{\text{c}}$ such that

$$\mathfrak{s}_k^{\text{c}} := \int_{\mathtt{D}_k^{\text{c}}} \|\nabla \text{Ma}^{\text{hf,c}}\|_2^2 \, dx, \quad k = 1, \ldots, N_{\text{e}}^{\text{c}}.$$

We then reorder the elements so that $\mathfrak{s}_1^{\text{c}} \geq \mathfrak{s}_2^{\text{c}} \geq \ldots$; given $n_1 = n_2 = 0.1 \cdot N_{\text{e}}^{\text{c}}$, we define the barycenters $\{x_j^{\text{c}}\}_j$ and the size function

$$h^{\text{tmp}}(x) = \min\left\{3h_0 + \frac{1}{4}\min\left\{\text{dist}\left(x, \{x_j^{\text{c}}\}_{j=1}^{n_1}\right),\ 2h_0 + \text{dist}\left(x, \{x_j^{\text{c}}\}_{j=n_1+1}^{n_1+n_2}\right)\right\},\ \bar{h}(x)\right\}$$

where $h_0 = 0.007$,

$$\bar{h}(x) = \min\left\{2h_0 + \text{dist}_{\text{bump}}(x),\ 6h_0 + (-0.6 - x_1)_+\right\},$$

and $\text{dist}_{\text{bump}}(x)$ is the distance of the point x from the semicircular bump. The size function h^{tmp} measures the proximity to the regions where the gradient of the Mach number is large: it thus leads to mesh refinement in the proximity of the shocks.

The size function h^{tmp} is excessively irregular for mesh generation purposes: for this reason, we project h^{tmp} over a 100×100 $\mathtt{p} = 2$ structured uniform grid over $\Omega_{\text{box}} = (-1, 1.5) \times (0, 1)$ and we compute a moving average with respect to both coordinates; the resulting FE field $h^\star$ is passed to the mesh generation routine `distmesh2d` to generate the $\mathtt{p} = 1$ FE grid; finally, we perform an iteration of uniform refinement to obtain the mesh in Figure 5(b).

References

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

[2] Francesco Ballarin, Elena Faggiano, Sonia Ippolito, Andrea Manzoni, Alfio Quarteroni, Gianluigi Rozza, and Roberto Scrofani. Fast simulations of patient-specific haemodynamics of coronary artery bypass grafts based on a POD–Galerkin method and a vascular shape parametrization. *Journal of Computational Physics*, 315:609–628, 2016.

[3] Francesco Ballarin, Elena Faggiano, Andrea Manzoni, Alfio Quarteroni, Gianluigi Rozza, Sonia Ippolito, Carlo Antona, and Roberto Scrofani. Numerical modeling of hemodynamics scenarios of patient-specific coronary artery bypass grafts. *Biomechanics and modeling in mechanobiology*, 16(4):1373–1399, 2017.

[4] F Bassi, A Crivellini, S Rebay, and M Savini. Discontinuous Galerkin solution of the Reynolds-averaged Navier–Stokes and k–ω turbulence model equations. *Computers & Fluids*, 34(4-5):507–540, 2005.

[5] Francesco Bassi, L Botti, Alessandro Colombo, Andrea Crivellini, Nicoletta Franchina, Antonio Ghidoni, and Stefano Rebay. Very high-order accurate discontinuous galerkin computation of transonic turbulent flows on aeronautical configurations. In *ADIGMA-A European Initiative on the Development of Adaptive Higher-Order Variational Methods for Aerospace Applications*, pages 25–38. Springer, 2010.

[6] Francesco Bassi and Stefano Rebay. A high-order accurate discontinuous finite element method for the numerical solution of the compressible Navier–Stokes equations. *Journal of computational physics*, 131(2):267–279, 1997.

[7] G Berkooz, P Holmes, and JL Lumley. The proper orthogonal decomposition in the analysis of turbulent flows. *Annual review of fluid mechanics*, 25(1):539–575, 1993.

[8] Patrick J Blonigan, Francesco Rizzi, Micah Howard, Jeffrey A Fike, and Kevin T Carlberg. Model reduction for steady hypersonic aerodynamics via conservative manifold least-squares Petrov–Galerkin projection. *AIAA Journal*, pages 1–17, 2021.

[9] J Brunken, K Smetana, and K Urban. (Parametrized) first order transport equations: realization of optimally stable Petrov–Galerkin methods. *SIAM Journal on Scientific Computing*, 41(1):A592–A621, 2019.

[10] N Cagniart, Y Maday, and B Stamm. Model order reduction for problems with large convection effects. In *Contributions to Partial Differential Equations and Applications*, pages 131–150. Springer, 2019.

[11] K Carlberg, M Barone, and H Antil. Galerkin v. least-squares Petrov–Galerkin projection in nonlinear model reduction. *Journal of Computational Physics*, 330:693–734, 2017.

[12] Kevin Carlberg. Adaptive h-refinement for reduced-order models. *International Journal for Numerical Methods in Engineering*, 102(5):1192–1210, 2015.

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

[14] Jesse Chan. Entropy stable reduced order modeling of nonlinear conservation laws. *Journal of Computational Physics*, 423:109789, 2020.

[15] Todd Chapman, Philip Avery, Pat Collins, and Charbel Farhat. Accelerated mesh sampling for the hyper reduction of nonlinear computational models. *International Journal for Numerical Methods in Engineering*, 109(12):1623–1654, 2017.

[16] Alessandro Colombo. *An agglomeration-based discontinuous Galerkin method for compressible flows*. PhD thesis, Università degli studi di Bergamo, 2011.

[17] Niccolò Dal Santo and Andrea Manzoni. Hyper-reduced order models for parametrized unsteady navier-stokes equations on domains with variable shape. *Advances in Computational Mathematics*, 45(5):2463–2501, 2019.

[18] Dinh Dung and Vu Thanh. On nonlinear n-widths. *Proceedings of the American Mathematical Society*, 124(9):2757–2765, 1996.

[19] Jens L Eftang, David J Knezevic, and Anthony T Patera. An "hp" certified reduced basis method for parametrized parabolic partial differential equations. *Mathematical and Computer Modelling of Dynamical Systems*, 17(4):395–422, 2011.

[20] Virginie Ehrlacher, Damiano Lombardi, Olga Mula, and François-Xavier Vialard. Nonlinear model reduction on metric spaces. application to one-dimensional conservative pdes in wasserstein spaces. *ESAIM. Mathematical Modelling and Numerical Analysis*, 54, 2020.

[21] Philip A Etter and Kevin T Carlberg. Online adaptive basis refinement and compression for reduced-order models via vector-space sieving. *Computer Methods in Applied Mechanics and Engineering*, 364:112931, 2020.

[22] C Farhat, T Chapman, and P Avery. Structure-preserving, stability, and accuracy properties of the energy-conserving sampling and weighting method for the hyper reduction of nonlinear finite element dynamic models. *International Journal for Numerical Methods in Engineering*, 102(5):1077–1110, 2015.

[23] Thomas Franz, Ralf Zimmermann, Stefan Görtz, and Niklas Karcher. Interpolation-based reduced-order modelling for steady transonic flows via manifold learning. *International Journal of Computational Fluid Dynamics*, 28(3-4):106–121, 2014.

[24] Stefania Fresca and Andrea Manzoni. A comprehensive deep learning-based approach to reduced order modeling of nonlinear time-dependent parametrized PDEs. *Journal of Scientific Computing*, 87(2):1–36, 2021.

[25] Jean-Frédéric Gerbeau and Damiano Lombardi. Approximated lax pairs for the reduced order integration of nonlinear evolution equations. *Journal of Computational Physics*, 265:246–269, 2014.

[26] W J Gordon and C A Hall. Construction of curvilinear co-ordinate systems and applications to mesh generation. *International Journal for Numerical Methods in Engineering*, 7(4):461–477, 1973.

[27] Ingo Gühring, Mones Raslan, and Gitta Kutyniok. Expressivity of deep neural networks. *arXiv preprint arXiv:2007.04759*, 2020.

[28] J S Hesthaven, G Rozza, and B Stamm. *Certified reduced basis methods for parametrized partial differential equations*. Springer, 2016.

[29] A Iollo and D Lombardi. Advection modes by optimal mass transfer. *Physical Review E*, 89(2):022923, 2014.

[30] K Kashima. Nonlinear model reduction by deep autoencoder of noise response data. In *2016 IEEE 55th Conference on Decision and Control (CDC)*, pages 5750–5755. IEEE, 2016.

[31] Mariella Kast, Mengwu Guo, and Jan S Hesthaven. A non-intrusive multifidelity method for the reduced order modeling of nonlinear problems. *Computer Methods in Applied Mechanics and Engineering*, 364:112947, 2020.

[32] Y Kim, Y Choi, D Widemann, and T Zohdi. Efficient nonlinear manifold reduced order model. *arXiv preprint arXiv:2011.07727*, 2020.

[33] Toni Lassila, Andrea Manzoni, Alfio Quarteroni, and Gianluigi Rozza. Model order reduction in fluid dynamics: challenges and perspectives. In *Reduced Order Methods for modeling and computational reduction*, pages 235–273. Springer, 2014.

[34] K Lee and K T Carlberg. Model reduction of dynamical systems on nonlinear manifolds using deep convolutional autoencoders. *Journal of Computational Physics*, 404:108973, 2020.

[35] Jiayi Ma, Weichao Qiu, Ji Zhao, Yong Ma, Alan L Yuille, and Zhuowen Tu. Robust L_2E estimation of transformation for non-rigid registration. *IEEE Transactions on Signal Processing*, 63(5):1115–1129, 2015.

[36] Jiayi Ma, Jia Wu, Ji Zhao, Junjun Jiang, Huabing Zhou, and Quan Z Sheng. Nonrigid point set registration with robust transformation learning under manifold regularization. *IEEE transactions on neural networks and learning systems*, 30(12):3584–3597, 2018.

[37] Youssef Marzouk, Tarek Moselhy, Matthew Parno, and Alessio Spantini. Sampling via measure transport: An introduction. *Handbook of uncertainty quantification*, pages 1–41, 2016.

[38] MATLAB. version 9.5 (r2020b), 2020.

[39] R Mojgani and M Balajewicz. Arbitrary Lagrangian Eulerian framework for efficient projection-based reduction of convection dominated nonlinear flows. In *APS Division of Fluid Dynamics Meeting Abstracts*, 2017.

[40] Rambod Mojgani and Maciej Balajewicz. Physics-aware registration based auto-encoder for convection dominated pdes. *arXiv preprint arXiv:2006.15655*, 2020.

[41] N J Nair and M Balajewicz. Transported snapshot model order reduction approach for parametric, steady-state fluid flows containing parameter-dependent shocks. *International Journal for Numerical Methods in Engineering*, 2018.

[42] M Ohlberger and S Rave. Nonlinear reduced basis approximation of parameterized evolution equations via the method of freezing. *Comptes Rendus Mathematique*, 351(23-24):901–906, 2013.

[43] M Ohlberger and S Rave. Reduced basis methods: success, limitations and future challenges. In *Proceedings of the Conference Algoritmy*, pages 1–12, 2016.

[44] Benjamin Peherstorfer. Model reduction for transport-dominated problems via online adaptive bases and adaptive sampling. *SIAM Journal on Scientific Computing*, 42(5):A2803–A2836, 2020.

[45] Benjamin Peherstorfer, Karen Willcox, and Max Gunzburger. Survey of multifidelity methods in uncertainty propagation, inference, and optimization. *Siam Review*, 60(3):550–591, 2018.

[46] Per-Olof Persson and Jaime Peraire. Sub-cell shock capturing for discontinuous Galerkin methods. In *44th AIAA Aerospace Sciences Meeting and Exhibit*, page 112, 2006.

[47] Per-Olof Persson and Gilbert Strang. A simple mesh generator in MATLAB. *SIAM review*, 46(2):329–345, 2004.

[48] A Quarteroni, A Manzoni, and F Negri. *Reduced basis methods for partial differential equations: an introduction*, volume 92. Springer, 2015.

[49] J Reiss, P Schulze, J Sesterhenn, and V Mehrmann. The shifted proper orthogonal decomposition: a mode decomposition for multiple transport phenomena. *SIAM Journal on Scientific Computing*, 40(3):A1322–A1344, 2018.

[50] Julius Reiss. Optimization-based modal decomposition for systems with multiple transports. *SIAM Journal on Scientific Computing*, 43(3):A2079–A2101, 2021.

[51] Donsub Rim, Scott Moe, and Randall J LeVeque. Transport reversal for model reduction of hyperbolic partial differential equations. *SIAM/ASA Journal on Uncertainty Quantification*, 6(1):118–150, 2018.

[52] G Rozza, DBP Huynh, and AT Patera. Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations. *Archives of Computational Methods in Engineering*, 15(3):229—-275, 2007.

[53] Gianluigi Rozza, Martin Hess, Giovanni Stabile, Marco Tezzele, and Francesco Ballarin. *Basic ideas and tools for projection-based model reduction of parametric partial differential equations*, pages 1–47. Handbook on Model Order Reduction: Snapshot-Based Methods and Algorithms, De Gruyter, 2021.

[54] Neeraj Sarna, Jan Giesselmann, and Peter Benner. Data-driven snapshot calibration via monotonic feature matching. *arXiv preprint arXiv:2009.08414*, 2020.

[55] Neeraj Sarna and Sara Grundel. Model reduction of time-dependent hyperbolic equations using collocated residual minimisation and shifted snapshots. *arXiv preprint arXiv:2003.06362*, 2020.

[56] Michael K Sleeman and Masayuki Yano. Goal-oriented model reduction for parametrized time-dependent nonlinear partial differential equations. *Computer Methods in Applied Mechanics and Engineering*, 388:114206, 2022.

[57] T Taddei. A registration method for model order reduction: data compression and geometry reduction. *SIAM Journal on Scientific Computing*, 42(2):A997–A1027, 2020.

[58] T Taddei, S Perotto, and A Quarteroni. Reduced basis techniques for nonlinear conservation laws. *ESAIM: Mathematical Modelling and Numerical Analysis*, 49(3):787–814, 2015.

[59] T Taddei and L Zhang. Space-time registration-based model reduction of parameterized one-dimensional hyperbolic pdes. *ESAIM: Mathematical Modelling and Numerical Analysis*, 55(1):99–130, 2021.

[60] Tommaso Taddei and Lei Zhang. A discretize-then-map approach for the treatment of parameterized geometries in model order reduction. *Computer Methods in Applied Mechanics and Engineering*, 384:113956, 2021.

[61] Tommaso Taddei and Lei Zhang. Registration-based model reduction in complex two-dimensional geometries. *Journal of Scientific Computing*, 88(3):79, 2021.

[62] Eleuterio F Toro. *Riemann solvers and numerical methods for fluid dynamics: a practical introduction*. Springer Science & Business Media, 2013.

[63] S Volkwein. Model reduction using proper orthogonal decomposition. *Lecture Notes, Institute of Mathematics and Scientific Computing, University of Graz. see math.uni-konstanz.de/numerik/personen/volkwein/teaching/POD-Vorlesung.pdf*, 1025, 2011.

[64] Christian Walder and Bernhard Schölkopf. Diffeomorphic dimensionality reduction. *Advances in Neural Information Processing Systems*, 21:1713–1720, 2008.

[65] Kyle M Washabaugh, Matthew J Zahr, and Charbel Farhat. On the use of discrete nonlinear reduced-order models for the prediction of steady-state flows past parametrically deformed complex geometries. In *54th AIAA Aerospace Sciences Meeting*, page 1814, 2016.

[66] G Welper. Interpolation of functions with parameter dependent jumps by transformed snapshots. *SIAM Journal on Scientific Computing*, 39(4):A1225–A1250, 2017.

[67] H Wendland. *Scattered data approximation*, volume 17. Cambridge university press, 2004.

[68] Pieter Wesseling. *Principles of computational fluid dynamics*, volume 29. Springer Science & Business Media, 2009.

[69] M Yano. Discontinuous Galerkin reduced basis empirical quadrature procedure for model reduction of parametrized nonlinear conservation laws. *Advances in Computational Mathematics*, pages 1–34, 2019.

[70] Masayuki Yano. A reduced basis method for coercive equations with an exact solution certificate and spatio-parameter adaptivity: energy-norm and output error bounds. *SIAM Journal on Scientific Computing*, 40(1):A388–A420, 2018.

[71] Masayuki Yano. *Model reduction in computational aerodynamics*, pages 201–236. Handbook on Model Order Reduction: applications, De Gruyter, 2021.

[72] Jian Yu and Jan S Hesthaven. A study of several artificial viscosity models within the discontinuous Galerkin framework. *Communications in Computational Physics*, 27(ARTICLE):1309–1343, 2020.

[73] M J Zahr and P-O Persson. An optimization-based approach for high-order accurate discretization of conservation laws with discontinuous solutions. *Journal of Computational Physics*, 365:105–134, 2018.

[74] Matthew J Zahr, Andrew Shi, and P-O Persson. Implicit shock tracking using an optimization-based high-order discontinuous Galerkin method. *Journal of Computational Physics*, 410:109385, 2020.

[75] R Zimmermann, B Peherstorfer, and K Willcox. Geometric subspace updates with applications to online adaptive nonlinear model reduction. *SIAM Journal on Matrix Analysis and Applications*, 39(1):234–261, 2018.

[76] Barbara Zitova and Jan Flusser. Image registration methods: a survey. *Image and vision computing*, 21(11):977–1000, 2003.