A Deep Learning approach to Reduced Order Modelling of parameter dependent Partial Differential Equations

Franco, N.; Manzoni, A.; Zunino, P.

A DEEP LEARNING APPROACH TO REDUCED ORDER MODELLING OF PARAMETER DEPENDENT PARTIAL DIFFERENTIAL EQUATIONS

NICOLA R. FRANCO, ANDREA MANZONI, AND PAOLO ZUNINO

ABSTRACT. Within the framework of parameter dependent PDEs, we develop a constructive approach based on Deep Neural Networks for the efficient approximation of the parameter-to-solution map. The research is motivated by the limitations and drawbacks of state-of-the-art algorithms, such as the Reduced Basis method, when addressing problems that show a slow decay in the Kolmogorov n-width. Our work is based on the use of deep autoencoders, which we employ for encoding and decoding a high fidelity approximation of the solution manifold. In order to fully exploit the approximation capabilities of neural networks, we consider a nonlinear version of the Kolmogorov n-width over which we base the concept of a minimal latent dimension. We show that this minimal dimension is intimately related to the topological properties of the solution manifold, and we provide some theoretical results with particular emphasis on second order elliptic PDEs. Finally, we report numerical experiments where we compare the proposed approach with classical POD-Galerkin reduced order models. In particular, we consider parametrized advection-diffusion PDEs, and we test the methodology in the presence of strong transport fields, singular terms and stochastic coefficients.

INTRODUCTION

In many areas of science, such as physics, biology and engineering, phenomena are often modeled in terms of Partial Differential Equations (PDEs) that exhibit dependence on one or multiple parameters. As an example, consider the (stationary) advection-diffusion equation below,

$$\begin{cases} -\mathrm{div}(\boldsymbol{\sigma_\mu}\nabla u) + \boldsymbol{b_\mu} \cdot \nabla u = f_{\boldsymbol{\mu}} & \text{in } \Omega \\ u = g_{\boldsymbol{\mu}} & \text{on } \partial\Omega \end{cases},$$

where $\Omega \subset \mathbb{R}^d$ is a bounded domain and $\boldsymbol{\mu}$ a vector parameter taking values in a suitable parameter space $\Theta \subset \mathbb{R}^p$. For each $\boldsymbol{\mu} \in \Theta$, we assume the above to admit a unique solution $u_{\boldsymbol{\mu}}$, to be sought within a given Hilbert space $(V, ||\cdot||)$.

In some cases, one is not interested in computing the PDE solution for a single fixed $\boldsymbol{\mu} \in \Theta$, but rather for an ensemble of parameter values. In general, this corresponds to exploring the so-called *solution manifold* [23, 41], that is $\mathcal{S} := \{u_{\boldsymbol{\mu}}\}_{\boldsymbol{\mu}\in\Theta}$. The map $\boldsymbol{\mu} \to u_{\boldsymbol{\mu}}$ is known under many equivalent names such as the *parametric map* [55], the parameter-to-state map [34] or the solution map [50]. Approximating the parametric map in a highly-efficient way is a challenging task that can be encountered in several contexts, from optimal control problems with parametric PDEs constraints [8] to multiscale fluid mechanics [37], or Bayesian inversion and uncertainty quantification [11]. In all these cases, the main drawback is represented by

1

the computational cost entailed by traditional PDE solvers. In fact, despite their accuracy, each query of a numerical scheme such as the Finite Element Method (FEM) implies a computational cost that easily becomes unsustainable in many query applications, where computations are supposed to be carried out within a very short amount of time.

For this reason, Full Order Models (FOMs) can often be replaced with cheaper surrogate models, namely Reduced Order Models (ROMs). ROMs originate from the need of boosting the computational burden of FOMs at the price of a negligible compromise in terms of accuracy. During the last decades, several successful model reduction techniques have been developed, such as the Reduced Basis method [53] and others. However, the majority of these ROMs heavily relies on linear projections as tools for model reduction, which significantly limits the spectrum of possible applications, leaving out many families of parametrized PDEs. Indeed, ROMs based on linear reduction techniques encounter substantial drawbacks whenever the solution manifold has a so-called Kolmogorov n-width [38] that decays slowly with n. The Kolmogorov n-width is a quantity that measures the degree of accuracy by which a set can be approximated using linear subspaces of dimension n, namely

$$(1) \qquad d_n(\mathcal{S}) := \inf_{\substack{V_n \subset V, \\ \dim(V_n) = n}} \sup_{u \in \mathcal{S}} \inf_{\mathbf{v} \in V_n} ||u - v||.$$

If $d_n(\mathcal{S})$ decays slowly with n, then projection-based ROMs can reach meaningful accuracies only for large values of n, which in turn leads to expensive models. We point out that this phenomenon is far from being uncommon. As a matter of fact, the slow decay on $d_n(\mathcal{S})$ is typical of transport-dominated problems, even under fairly simple circumstances [27, 50]. The same is also true for purely diffusive problems, provided that the parameters enter in a highly nonlinear and possibly spatially localized way; we report a toy example of this situation at the end of the current Section, Example 1.

In order to tackle these drawbacks, we propose a novel approach based on Deep Neural Networks (DNNs) [56] that naturally accounts for possible nonlinearities in the solution manifold. Our construction is mostly inspired by the recent advancements in nonlinear approximation theory, e.g. [16, 17, 55], and the increasing use of deep-learning techniques for parametrized PDEs, as in [14, 24, 39, 44].

Our contribution. The purpose of the present work is to provide alternative ROM techniques for parametrized PDEs that are able to overcome the drawbacks implied by the slow decay of the Kolmogorov n-width. In particular, we focus on nonintrusive ROMs where the solution map is approximated by a deep neural network Φ. This idea has been recently investigated both theoretically, as in [39, 55], and practically, e.g. [24, 26]. By now, the drawbacks posed by this approach are mainly practical: it is often unclear how the network architecture should be designed and which optimization strategies are better suited for the purpose. Also, we lack the understanding of the possible ways the nonlinearities in the DNN should be exploited in order to make the most out of it. Here, we wish to partially answer these questions and provide a constructive way of designing such Φ.

The key idea is to break the problem into two parts. First, we seek for a low-dimensional representation of the solution manifold, which we obtain by training a deep AutoEncoder [32], $\Psi \circ \Psi'$. The encoder, Ψ', is used to map the solution

manifold into a reduced feature space $\mathbb{R}^n$, while the decoder serves for the reconstruction task. Here we see a clear analogy with the Nonlinear Kolmogorov n-width as defined in DeVore et al. [18]. There, the authors define

$$\delta_n(\mathcal{S}) := \inf_{\substack{\Psi' \in \mathcal{C}(\mathcal{S},\, \mathbb{R}^n) \\ \Psi \in \mathcal{C}(\mathbb{R}^n,\, V)}} \sup_{u \in \mathcal{S}} \|u - \Psi(\Psi'(u))\|,$$

as a nonlinear counterpart of $d_n(\mathcal{S})$. In light of this, we introduce the concept of *minimal latent dimension*, denoted as $n_{\min}(\mathcal{S})$, which we define as the smallest n for which $\delta_n(\mathcal{S}) = 0$. By choosing this particular $n_{\min}(\mathcal{S})$ as latent dimension for the autoencoder, we are then able to perform a significant model reduction.

Once the autoencoder has been trained, we exploit the encoder Ψ' in order to represent each solution $u_{\boldsymbol{\mu}}$ through a low-dimensional vector $\mathbf{u}^n_{\boldsymbol{\mu}} \in \mathbb{R}^n$. We then train a third network $\phi : \Theta \to \mathbb{R}^n$ to learn the *reduced map* $\boldsymbol{\mu} \to \mathbf{u}^n_{\boldsymbol{\mu}}$. In this way, by connecting the architectures of ϕ and Ψ we obtain the complete model, $\Phi := \Psi \circ \phi$, which we later term as DL-ROM (Deep Learning based Reduced Order Model, in the same spirit of previous works [24, 25]).

The novelty in our contribution is twofold. On one hand, we develop a new constructive way of using neural networks to approximate the solution map and we test it on some numerical examples. On the other hand, we prove theoretical results that motivate the choice of the ROM dimension. Indeed, despite the popularity of autoencoders, e.g. [24, 42, 47, 64], the choice of the latent dimension n is often handled by trial and error. In contrast, we establish precise bounds on n thanks to a rigorous theoretical analysis.

First, in Theorems 1 and 2, we investigate the link between the minimal latent dimension $n_{\min}(\mathcal{S})$ and the topological properties of $\mathcal{S}$. Then, in Theorem 3 we explicitly bound $n_{\min}(\mathcal{S})$ in terms of the dimensionality of the parameter space. In particular, we show that $n_{\min}(\mathcal{S}) \le 2p + 1$ as soon as the parametric map is Lipschitz continuous. Finally, in Theorem 4 we apply the theory to the case of second order elliptic PDEs, where we demonstrate how the parameters directly affect the value of the minimal latent dimension.

The paper is organized as follows. In Section 1 we introduce the general framework and briefly recall the driving ideas of linear reduction. In Section 2 we move to the nonlinear case, where we establish a solid theoretical background for the construction of the DL-ROM, with particular emphasis on minimal representations and parametrized PDEs. In Section 3 we dive into the details of our deep-learning approach, thereby discussing the general construction and a few possible variants. In Section 4 we present some numerical results and assess the proposed methodology. Finally, to make the paper self-contained, auxiliary mathematical results are reported in the Appendix.

Example 1. For any $x_0 \in \mathbb{R}$, let δ_{x_0} be the Dirac delta distribution centered at x_0. Consider the 2-dimensional parameter space $\Theta := \{\boldsymbol{\mu} = (\mu_1, \mu_2) \in [-1, 1] \times [0, 1] \mid -1 \le \mu_1 - \mu_2 \le \mu_1 + \mu_2 \le 1\}$, together with the differential problem below

$$\begin{cases} -u'' = 2\delta_{\mu_1} - \delta_{\mu_1 - \mu_2} - \delta_{\mu_1 + \mu_2} & x \in (-2, 2) \\ u(-2) = u(2) = 0 \end{cases}$$

For each $\boldsymbol{\mu} \in \Theta$, the corresponding solution $u_{\boldsymbol{\mu}}$ is a piece-wise linear function with support given by $[\mu_1 - \mu_2, \mu_1 + \mu_2]$. In particular, $u_{\boldsymbol{\mu}}$ is a hat function with a peak of height μ_2 at exactly $x = \mu_1$. Also, by direct calculation, its L^2 norm is $||u_{\boldsymbol{\mu}}||_2 = \sqrt{\frac{2}{3}\mu_2^3}$. Let now $\mathcal{S} := \{u_{\boldsymbol{\mu}}\}_{\boldsymbol{\mu} \in \Theta} \subset V := L^2[-2, 2]$ and fix any positive $n \in \mathbb{N}$. It is then easy to see that the functions

$$v_{i,n} := u_{\left(-1+\frac{i}{n}-\frac{1}{2n}, \frac{1}{2n}\right)}, \quad i = 1, ..., 2n$$

are mutually orthogonal in $L^2[-2, 2]$. As a consequence, the Kolmogorov n-width of $\mathcal{S}$ satisfies

$$d_n(\mathcal{S}) \geq d_n(\{v_{i,n}\}_{i=1}^{2n}) =$$

$$= d_n\left(\{||v_{i,n}||_2^{-1}v_{i,n}\}_{i=1}^{2n}\right)||v_{i,n}||_2 = \frac{1}{\sqrt{2}}||v_{i,n}||_2 = \frac{1}{2\sqrt{6}}n^{-3/2},$$

where the second last equality follows by noticing that the set $\{||v_{i,n}||_2^{-1}v_{i,n}\}_{i=1}^{2n}$ is isometric to the canonical basis of $\mathbb{R}^{2n}$ (see [50]).

Therefore, $d_n(\mathcal{S})$ decays with a rate of at most $n^{-3/2}$, which is relatively slow when compared to the ideal case where the parametric map is analytic and the Kolmogorov n-width is known to decay exponentially, $d_n(\mathcal{S}) \sim \mathrm{e}^{-\gamma n}$.

1. General background

Within the present Section we formally introduce the problem of reduced order modelling for parametrized PDEs. For later comparison, we also take the chance to recall the linear reduction technique known as Principal Orthogonal Decomposition [45, 53]. In the remainder of the paper, we make use of elementary notions coming from the areas of Functional Analysis, Numerical Analysis and Topology. We respectively refer to [1, 22], [54] and [33].

1.1. **Reduced Order Models for parametrized PDEs.** We are given a parameter space $\Theta \subset \mathbb{R}^p$, a Hilbert state space $(V, ||\cdot||)$ and a parameter dependent operator (including data) $A_{\boldsymbol{\mu}} : V \times V \to \mathbb{R}$. For each $\boldsymbol{\mu} \in \Theta$ we consider the problem

$$(2) \qquad u \in V : \quad A_{\boldsymbol{\mu}}(u, v) = 0 \quad \forall v \in V.$$

We assume the problem to be well-posed, so that for each $\boldsymbol{\mu} \in \Theta$ there exists a unique solution $u = u_{\boldsymbol{\mu}} \in V$. Our interest is to define a ROM that is able to approximate the parametric map $\boldsymbol{\mu} \to u_{\boldsymbol{\mu}}$ efficiently. In general, the workflow goes as follows. First, one chooses a FOM, which we here assume to be based on Galerkin projections. This corresponds to fixing a so-called *high-fidelity discretization*, that is a finite dimensional subspace $V_h \subset V$, $\dim(V_h) = N_h$, to be used as substitute for the original trial space. Having chosen an (orthonormal) basis for V_h, say $\{v_i\}_{i=1}^{N_h}$, for each $\boldsymbol{\mu} \in \Theta$ one replaces equation (2) with

$$\mathbf{u} \in \mathbb{R}^{N_h} : \quad A_{\boldsymbol{\mu}} \left(\sum_{i=1}^{n} \mathbf{u}_i v_i, \ v \right) = 0 \quad \forall v \in V_h.$$

In order to emphasize the presence of the discretization and the dependence on $\boldsymbol{\mu}$, from now on we will write $\mathbf{u}_{\boldsymbol{\mu}}^h$ to intend the solution to the above. The main purpose of the high-fidelity discretization is to reframe the original problem within a finite dimensional setting, without particular care on the computational cost (for now). Regarding the choice of V_h, we make the following assumption.

Assumption 1. *For any $\varepsilon > 0$ there exists $V_h := \mathrm{span}\{v_i\}_i^{N_h} \subset V$ such that*

$$\sup_{\boldsymbol{\mu} \in \Theta} \left\| u_{\boldsymbol{\mu}} - \sum_{i=1}^{N_h} \mathbf{u}_{\boldsymbol{\mu},i}^h v_i \right\| < \varepsilon$$

that is, the FOM accuracy can be bounded independently on the value of $\boldsymbol{\mu} \in \Theta$.

The above is a very common assumption in the literature, see e.g. [39], that allows us to formally replace V with V_h. The objective now becomes that of learning the map $\boldsymbol{\mu} \to \mathbf{u}_{\boldsymbol{\mu}}^h$ in a way that reduces the FOM cost. In particular, the construction of the ROM consists in finding a suitable map $\Phi : \mathbb{R}^p \to \mathbb{R}^{N_h}$ for which $\Phi(\boldsymbol{\mu}) \approx \mathbf{u}_{\boldsymbol{\mu}}^h$. To do so, the common practice is to make extensive use of the FOM during a preliminary offline stage, which results in the collection of the so-called *snapshots*, $\{\boldsymbol{\mu}_i, \mathbf{u}_{\boldsymbol{\mu}_i}^h\}_i \subset \mathbb{R}^p \times \mathbb{R}^{N_h}$. These snapshots are then analyzed in order to build the ROM. In this sense, the identification of Φ can be seen as a problem of Statistical Learning, as argued in [39]. The way Φ is defined from the data is what characterizes each ROM, its efficiency and accuracy.

1.2. **Methods based on linear projections.** Many state-of-the-art ROMs are built upon the use of linear reduction techniques, which are known to work particularly well for second order elliptic PDEs with affine coefficients [2]. The idea is the following. Having fixed a high-fidelity discretization, one considers the (discretized) solution manifold $\mathcal{S}^h := \{\mathbf{u}_{\boldsymbol{\mu}}^h\}_{\boldsymbol{\mu} \in \Theta}$ and tries to approximate it using linear subspaces. This translates into fixing a reduced dimension $n \in \mathbb{N}$ and searching for the orthonormal matrix $\mathbf{V} \in \mathbb{R}^{N_h \times n}$ that minimizes the errors $\|\mathbf{u}_{\boldsymbol{\mu}}^h - \mathbf{V}\mathbf{V}^T\mathbf{u}_{\boldsymbol{\mu}}^h\|$. In practice, the identification of such $\mathbf{V}$ is done empirically by exploiting the aforementioned snapshots $\{\boldsymbol{\mu}_i, \mathbf{u}_{\boldsymbol{\mu}_i}^h\}_{i=1}^N$, which is often achieved through the so-called Principal Orthogonal Decomposition (POD). In short, this latter approach considers all the FOM snapshots as columns of a matrix, $\mathbf{U} := [\mathbf{u}_{\boldsymbol{\mu}_1}^h, , \mathbf{u}_{\boldsymbol{\mu}_N}^h]$ and computes its singular value decomposition

$$\mathbf{U} = \tilde{\mathbf{U}} \Sigma \mathbf{W}^T$$

where $\Sigma = \mathrm{diag}(\sigma_1, .., \sigma_N)$ with $\sigma_1 \geq ... \geq \sigma_N \geq 0$. Then, in the POD approach, $\mathbf{V}$ is defined by extracting the first n columns of $\tilde{\mathbf{U}}$. It is well-known that this choice of $\mathbf{V}$ is optimal -in some sense- over the training sample $\{\mathbf{u}_{\boldsymbol{\mu}_i}^h\}_{i=1}^N$. We also mention that, while this version of the POD considers $\mathbb{R}^{N_h}$ as a Euclidean space, slight modifications allow to account for different norms.

Once $\mathbf{V}$ has been identified, the solution manifold is projected onto the reduced space $\mathbb{R}^n$, and each FOM solution is associated with the corresponding low-dimensional representation, $\mathbf{u}^n_{\boldsymbol{\mu}} := \mathbf{V}^T \mathbf{u}^h_{\boldsymbol{\mu}}$. The final purpose is to approximate the reduced map $\boldsymbol{\mu} \to \mathbf{u}^n_{\boldsymbol{\mu}}$ through a suitable function $\phi : \Theta \to \mathbb{R}^n$, so that the final ROM reads $\Phi(\boldsymbol{\mu}) := \mathbf{V}\phi(\boldsymbol{\mu}) \approx \mathbf{u}^h_{\boldsymbol{\mu}}$. Within the literature, this step has been handled in multiple ways. In the Reduced Basis method [41, 53], particularly in the so-called POD-Galerkin method, the reduced map is computed exactly by projecting the original PDE onto $\mathbb{R}^n$. Depending on the parametrization and on the type of PDE, this procedure may turn to be too expensive, which is why several variants have been proposed, e.g. [4, 49, 57]. Other nonintrusive alternatives for defining ϕ have also been suggested, such as: Gaussian process regression [30], polynomial chaos expansions [35] and neural networks [14, 61].

Nevertheless, because of the linear approximation, these ROMs encounter substantial difficulties as soon as $\mathcal{S}$ has a Kolmogorov n-width, see equation (1), that decays slowly. In fact, within this framework we have

$$\sup_{\boldsymbol{\mu} \in \Theta} ||\mathbf{u}^h_{\boldsymbol{\mu}} - \mathbf{V}\phi(\boldsymbol{\mu})|| \; \geq \; \sup_{\boldsymbol{\mu} \in \Theta} ||\mathbf{u}^h_{\boldsymbol{\mu}} - \mathbf{V}\mathbf{V}^T \mathbf{u}^h_{\boldsymbol{\mu}}|| \; \geq \; d_n(\mathcal{S}^h) \; \geq \; d_n(\mathcal{S}) - \varepsilon$$

where $\varepsilon > 0$ is the accuracy of the high-fidelity discretization. Therefore, if $d_n(\mathcal{S})$ decays slowly, one may be forced to consider large values of n, which in turn makes ϕ more expensive and harder to identify. As we argue in the next Section, one possible solution to this problem is given by nonlinear reduction techniques. However, despite this being a promising direction, only a few steps have been made so far, e.g. [6, 24, 42].

2. Nonlinear Dimensionality Reduction

In the present Section we formalize the idea of using nonlinear reduction techniques for the compression of the solution manifold. We start by introducing all concepts and results in an abstract fashion. Only at the end, Section 2.2, we rephrase the content in terms of parametrized PDEs.

2.1. **Nonlinear Kolmogorov n-width and minimal latent dimension.** Within this Section, we consider an abstract setting where $(V, || \cdot ||)$ is a Hilbert space and $\mathcal{S} \subset V$ a generic subset. In particular, $\mathcal{S}$ needs not to be the solution manifold of a parametrized PDE and the theory is presented regardless of a possible discretization. We address the problem of finding a low-dimensional representation of $\mathcal{S}$ while minimizing the reconstruction error.

When V is finite-dimensional, the linear reduction described in Section 1.2 performs an encoding of $\mathcal{S}$ via the map $\mathbf{u} \to \mathbf{V}^T\mathbf{u} =: \mathbf{u}^n \in \mathbb{R}^n$, where n is the reduced dimension; the set is then recovered through $\mathbf{u}^n \to \mathbf{V}\mathbf{u}^n \approx \mathbf{u}$. Therefore, a possible generalization to the nonlinear case is to substitute $\mathbf{V}^T$ with some *encoder* $\Psi' : \mathcal{S} \to \mathbb{R}^n$ and $\mathbf{V}$ with a *decoder* $\Psi : \mathbb{R}^n \to V$. Of note, this is also an approach that easily extends to infinite-dimensional settings. Depending on the restrictions that we impose on Ψ' and Ψ, different reconstruction accuracies can be obtained.

Here, we only require Ψ' and Ψ to be continuous. This, naturally gives rise to the optimization problem below,

$$(3) \qquad \delta_n(\mathcal{S}) := \inf_{\substack{\Psi' \in \mathcal{C}(\mathcal{S}, \, \mathbb{R}^n) \\ \Psi \in \mathcal{C}(\mathbb{R}^n, \, V)}} \sup_{u \in \mathcal{S}} ||u - \Psi(\Psi'(u))||,$$

where $\mathcal{C}(X, Y)$ denotes the collection of all continuous maps from X to Y. As we mentioned in the Introduction, the above corresponds to the (continuous) Nonlinear Kolmogorov n-width as defined in [18]. It is clear that $d_n(\mathcal{S}) \geq \delta_n(\mathcal{S})$. Also, $\delta_n(\mathcal{S})$ is nonincreasing in n, which reflects the fact that better approximations can be achieved in higher dimensional spaces. However, in the context of reduced order modelling, smaller values of n are often preferable, as they allow for less expensive models. In this sense, whenever there exists a smallest dimension $n_{\min}$ that allows for an arbitrarily accurate reduction, i.e. $\delta_{n_{\min}}(\mathcal{S}) = 0$, we may want to focus on that one. For this reason, we introduce the notation

$$n_{\min}(\mathcal{S}) := \min\{n \in \mathbb{N} \mid \delta_n(\mathcal{S}) = 0\},$$

were we adopt the convention $\min \emptyset = +\infty$. We refer to $n_{\min}(\mathcal{S})$ as to the minimal latent dimension of $\mathcal{S}$. Clearly, when $V \cong \mathbb{R}^{N_h}$ is finite-dimensional, the above definition is of interest only if $n_{\min}(\mathcal{S}) \ll N_h$. Nevertheless, as we will see below, this is always the case as soon as $\mathcal{S}$ has an intrinsic low-dimensional structure. Indeed, the value of $n_{\min}(\mathcal{S})$ is strongly related to the topological properties of $\mathcal{S}$. For instance, in the case of compact sets, it is invariant under bicontinuous transformations. More precisely, we have the following.

Theorem 1. *Let V and W be two Hilbert spaces. Let $\mathcal{S} \subset V$ and $\mathcal{M} \subset W$ be two compact subsets. If $\mathcal{S}$ and $\mathcal{M}$ are homeomorphic, then $n_{\min}(\mathcal{S}) = n_{\min}(\mathcal{M})$.*

Proof. Since the situation is symmetric in $\mathcal{S}$ and $\mathcal{M}$, it is sufficient to prove that $n_{\min}(\mathcal{S}) \geq n_{\min}(\mathcal{M})$. If $n_{\min}(\mathcal{S}) = +\infty$, the inequality is obvious. Hence, we assume there exists some $n \in \mathbb{N}$ for which $\delta_n(\mathcal{S}) = 0$. By definition of infimum, there exists a sequence of encoding-decoding pairs $\{(\Psi'_j, \Psi_j)\}_{j \geq 0}$ in $\mathcal{C}(\mathcal{S}, \mathbb{R}^n) \times \mathcal{C}(\mathbb{R}^n, V)$ such that $\sup_{u \in \mathcal{S}} ||u - \Psi_j(\Psi'_j(u))||_V \to 0$ as $j \to +\infty$. Let now $\phi : \mathcal{S} \to \mathcal{M}$ be bicontinuous (recall that the sets are homeomorphic). Since $\mathcal{S}$ is compact, ϕ admits a uniformly continuous extension $\tilde{\phi} : V \to W$ (cf. Theorem 1.12 in [5]). We are then allowed to consider the continuous maps $\tilde{\Psi}_j := \tilde{\phi} \circ \Psi_j : \mathbb{R}^n \to W$ and $\tilde{\Psi}'_j := \Psi'_j \circ \phi^{-1} : \mathcal{M} \to \mathbb{R}^n$. For ω a monotone modulus of continuity of $\tilde{\phi}$, we have

$$\delta_n(\mathcal{M}) \leq \lim_{j \to +\infty} \sup_{m \in \mathcal{M}} ||m - \tilde{\Psi}_j(\tilde{\Psi}'_j(m))||_W =$$

$$= \lim_{j \to +\infty} \sup_{m \in \mathcal{M}} ||m - \tilde{\phi}(\Psi_j(\Psi'_j(\phi^{-1}(m))))||_W =$$

$$= \lim_{j \to +\infty} \sup_{s \in \mathcal{S}} ||\tilde{\phi}(s) - \tilde{\phi}(\Psi_j(\Psi'_j(s)))||_W \leq$$

$$\leq \lim_{j \to +\infty} \sup_{s \in \mathcal{S}} \omega\left(||s - \Psi_j(\Psi'_j(s))||_V\right) \leq$$

$$\leq \lim_{j \to +\infty} \omega\left(\sup_{s \in \mathcal{S}} ||s - \Psi_j(\Psi'_j(s))||_V\right) = 0,$$

as $\omega(h) \downarrow 0$ whenever $h \downarrow 0$. This proves that $\delta_n(\mathcal{S}) = 0 \implies \delta_n(\mathcal{M}) = 0$ and

hence $n_{\min}(\mathcal{S}) \geq n_{\min}(\mathcal{M})$. $\hfill\square$

The minimal latent dimension is also related to the so-called topological dimension, or Lebesgue covering dimension. For a formal definition of the latter we refer to [19, 21]. In particular, if $\mathcal{S}$ has an intrinsic p-dimensional structure, then we are able to bound $n_{\min}(\mathcal{S})$ explicitly. Indeed, by classical results of Dimension Theory, the following theorem holds true.

Theorem 2. *Let V be a Hilbert space and $\mathcal{S} \subset V$ a compact subset. If $\mathcal{S}$ has topological dimension p, then $n_{\min}(\mathcal{S}) \leq 2p + 1$, and the infimum appearing in (3) is attained at all reduced dimensions $n \geq 2p + 1$. Additionally, if $\mathcal{S}$ is a topological p-manifold, then the lower bound $n_{\min}(\mathcal{S}) \geq p$ also holds.*

Proof. We shall prove that $\delta_{2p+1}(\mathcal{S}) = 0$, and that the infimum is attained. To this end, we notice that $\mathcal{S}$ is compact and thus separable. Therefore, by the Menger–Nöbeling embedding theorem (see Theorem 1.11.4 in [21]), there exists a subset $A \subset \mathbb{R}^{2p+1}$ and a bicontinuous map $\phi : \mathcal{S} \to A$. By continuity, the set A is compact. In particular, ϕ^{-1} admits a continuous extension $\Psi : A \to V$. The existence of such an extension can be argued as in the proof Theorem 1, or using other results such as Dugundji extension theorem [20], with the advantage of generalizing Theorem 2 to the case of normed spaces. Next, we define $\Psi' := \phi$. Then, the pair (Ψ', Ψ) agrees with the definition of Nonlinear Kolmogorov n-width and it also yields a perfect reconstruction of $\mathcal{S}$. The first claim in the theorem follows.

Assume now that $\mathcal{S}$ is a p-manifold and let $n < p$. By definition, there exists a bicontinuous map ϕ from the closed unit ball $B := \{x \in \mathbb{R}^p, \ |x| \leq 1\}$ to a certain compact subset $U \subseteq \mathcal{S}$. Let

$$m := \min_{|x|=1} ||\phi(x) - \phi(-x)||.$$

Due compactness, the minimum is attained and thus $m > 0$ (recall that ϕ is bijective). We now prove that $\delta_n(\mathcal{S}) \geq m/2$, and therefore $n_{\min}(\mathcal{S}) \geq p$. Let $\Psi' : \mathcal{S} \to \mathbb{R}^n$ and $\Psi : \mathbb{R}^n \to V$ be continuous. We note that the composition $\Psi' \circ \phi$ is continuous from $B \subset \mathbb{R}^p \to \mathbb{R}^n$. Therefore, as $n < p$, by the Borsuk-Ulam theorem [10] we are granted the existence of a point $x^* \in B$, $|x^*| = 1$, for which $\Psi'(\phi(x^*)) = \Psi'(\phi(-x^*)) =: z$. It follows that,

$$\sup_{v \in \mathcal{S}} ||v - \Psi(\Psi'(v))|| \geq \sup_{v \in U} ||v - \Psi(\Psi'(v))|| = \sup_{x \in B} ||\phi(x) - \Psi(\Psi'(\phi(x)))|| \geq$$

$$\geq \max \left\{ ||\phi(x^*) - \Psi(\Psi'(\phi(x^*)))||, \ ||\phi(-x^*) - \Psi(\Psi'(\phi(-x^*)))|| \right\} \geq$$

$$\geq \frac{1}{2}||\phi(x^*) - \Psi(z)|| + ||\phi(-x^*) - \Psi(z)|| \geq \frac{1}{2}||\phi(x) - \phi(-x)|| \geq \frac{m}{2},$$

and the proof is complete. $\hfill\square$

We mention that, in the particular case of p-manifolds and under suitable smoothness assumptions, the bounds in Theorem 2 can be sharpened to $n_{\min}(\mathcal{S}) \leq 2p$ or even $n_{\min}(\mathcal{S}) \leq 2p - 1$ in case p is not a power of 2. These are all consequences of the so-called Whitney embedding theorem and a few of its variants. We do not dive deeper into the matter but leave [60] as a reference for the interested reader.

Notice that the intrinsic dimension of $\mathcal{S}$ does not uniquely determine the value of $n_{\min}(\mathcal{S})$. In particular, $\mathcal{S}$ may have topological dimension p but $n_{\min}(\mathcal{S}) > p$, coherently with Theorem 2. In this respect, we report below two simple examples.

Example 2. Let $V = \mathbb{R}^2$ and $\mathcal{S} = \{\mathbf{x} \in V : |\mathbf{x}| = 1\}$ be the unit circle. Then, $\mathcal{S}$ is a one-dimensional manifold but $\delta_1(\mathcal{S}) = 1$ and $n_{\min}(\mathcal{S}) = 2$. To see this, consider any pair of continuous maps $\Psi' : \mathcal{S} \to \mathbb{R}$ and $\Psi : \mathbb{R} \to \mathbb{R}^2$. By the Borsuk-Ulam theorem, there exists a point $\mathbf{x} \in \mathcal{S}$ such that $\Psi'(\mathbf{x}) = \Psi'(-\mathbf{x})$. Therefore, being $||\cdot|| = |\cdot|$ the Euclidean norm,

$$\sup_{\mathbf{v} \in \mathcal{S}} ||\mathbf{v} - \Psi(\Psi'(\mathbf{v}))|| \geq \max\{|\mathbf{x} - \Psi(\Psi'(\mathbf{x}))|, \ |-\mathbf{x} - \Psi(\Psi'(-\mathbf{x}))|\} \geq$$

$$\geq \frac{1}{2}\left(|x - \Psi(\Psi'(\mathbf{x}))| + |-\mathbf{x} - \Psi(\Psi'(-\mathbf{x}))|\right) \geq \frac{1}{2}|\mathbf{x} - (-\mathbf{x})| = 1.$$

As Ψ' and Ψ are arbitrary, we conclude that $\delta_1(\mathcal{S}) \geq 1$. The equality is then obtained by considering the case in which both Ψ' and Ψ are identically zero.

Example 3. On the spatial domain $\Omega = (0, \pi/2)$, consider the boundary value problem

$$\begin{cases} u'' = -u & x \in \Omega \\ u(0) = 10(2\mu^3 - 3\mu^2 + \mu) & \\ u(\pi/2) = 2|1 - 2\mu| - 1 \end{cases},$$

depending on a parameter $\mu \in [0, 1]$. Let us then consider the solution manifold $\mathcal{S} = \{u_\mu\}_{\mu \in [0,1]}$ as a subset of $V = L^2(\Omega)$. Then, $\mathcal{S}$ is a 1-dimensional manifold but its minimal latent dimension equals $n_{\min}(\mathcal{S}) = 2$. Indeed, explicitly expanding the solutions reads

$$u_\mu(x) = 10(2\mu^3 - 3\mu^2 + \mu)\cos x + (2|1 - 2\mu| - 1)\sin x.$$

It is then clear that, up to scaling of the L^2-norm, $\mathcal{S}$ can be isometrically identified with the curve $\mu \to (20\mu^3 - 30\mu^2 + 10\mu, \ 2|1 - 2\mu| - 1)$ in $\mathbb{R}^2$. But the latter curve is a compact manifold with positive nonlinear Kolmogorov 1-width, as it is homeomorphic to the unit circle (see Figure 1 and Theorem 1).

Remark. Here we only considered the case of Hilbert spaces, which is the typical framework used for elliptic PDEs. However, as mentioned in the proof of Theorem 2, many of the above ideas and results can be adapted to the more general context of normed and Banach spaces.

2.2. **Application to Parametrized PDEs.** Let us now consider the case of a PDE that depends on a vector of p parameters. We fix a parameter space $\Theta \subset \mathbb{R}^p$ and a Hilbert state space V. As before, for each $\boldsymbol{\mu} \in \Theta$ we denote the corresponding PDE solution with $u_{\boldsymbol{\mu}}$. Similarly, we define $\mathcal{S} = \{u_{\boldsymbol{\mu}}\}_{\boldsymbol{\mu} \in \Theta}$. Notice that we refer to $\mathcal{S}$
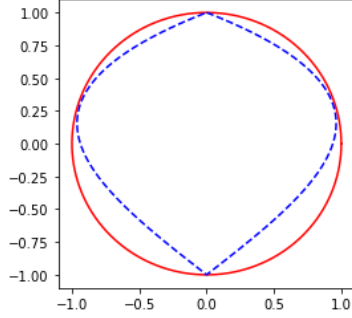
FIGURE 1. Reference picture for Example 3, Section 2.1. In blue (dashed line), the curve $\mu \to (20\mu^3 - 30\mu^2 + 10\mu, \ 2|1 - 2\mu| - 1)$, for $0 \le \mu \le 1$; in red (straight line), the unit circle. The two curves are clearly homeomorphic.

as the solution manifold even though, in fact, it is not granted that $\mathcal{S}$ is a manifold in the topological sense. This latter property can be recovered under additional hypotheses on the parameter space and the parametric map.

We consider the problem of finding a low-dimensional representation of $\mathcal{S}$ by means of nonlinear reduction. In particular, we wish to compress $\mathcal{S}$ as much as possible without paying in terms of accuracy, which corresponds to working with the minimal dimension $n_{\min}(\mathcal{S})$. To this end, we must take into account the fact that the dimension of the parameter space Θ influences the low-dimensional structure of $\mathcal{S}$, in fact, the latter is ultimately defined in terms of p scalar parameters.

Parallel to this, one may also exploit the parameters as additional tools during the dimensionality reduction process. This corresponds to replacing the solution manifold with the augmented set $\mathcal{S}_\Theta := \{(\boldsymbol{\mu}, u_{\boldsymbol{\mu}})\}_{\boldsymbol{\mu} \in \Theta} \subset \mathbb{R}^p \times V$, where $\boldsymbol{\mu}$ appears explicitly. The following Theorem provides some insights about both alternatives.

Theorem 3. *Let $\boldsymbol{\mu} \to u_{\boldsymbol{\mu}}$ be a map from a compact set $\Theta \subset \mathbb{R}^p$ to some Hilbert space V. Define the sets $\mathcal{S} := \{u_{\boldsymbol{\mu}}\}_{\boldsymbol{\mu} \in \Theta}$ and $\mathcal{S}_\Theta := \{(\boldsymbol{\mu}, u_{\boldsymbol{\mu}})\}_{\boldsymbol{\mu} \in \Theta}$. We have the following:*

  a1) *if the map $\boldsymbol{\mu} \to u_{\boldsymbol{\mu}}$ is Lipschitz continuous, then $n_{\min}(\mathcal{S}) \le 2p + 1$.*
  a2) *if there exists at least an internal point $\boldsymbol{\mu}_0 \in \Theta$ where the correspondence $\boldsymbol{\mu} \to u_{\boldsymbol{\mu}}$ is locally injective, then $n_{\min}(\mathcal{S}) \ge p$.*
  a3) *if the map $\boldsymbol{\mu} \to u_{\boldsymbol{\mu}}$ is continuous and injective, then $n_{\min}(\mathcal{S}) = n_{\min}(\Theta)$. In particular, $n_{\min}(\mathcal{S}) = p$ whenever Θ has nonempty interior.*
  b1) *if the map $\boldsymbol{\mu} \to u_{\boldsymbol{\mu}}$ is continuous, then $n_{\min}(\mathcal{S}_\Theta) = n_{\min}(\Theta)$. In particular, $n_{\min}(\mathcal{S}_\Theta) = p$ whenever Θ has nonempty interior.*

Proof. For the sake of brevity, let us define the map $u : \Theta \to V$ as $u(\boldsymbol{\mu}) := u_{\boldsymbol{\mu}}$.

  a1) Let $\dim(\mathcal{S})$ and $\dim_H(\mathcal{S})$ be respectively the topological dimension (covering dimension) and the Hausdorff dimension of $\mathcal{S}$. A result due to Sznirelman, see Theorem 2.43 in [19], ensures that $\dim(\mathcal{S}) \le \dim_H(\mathcal{S})$. Since u is Lipschitz, we also have $\dim_H(\mathcal{S}) = \dim_H(u(\Theta)) \le \dim_H(\Theta) \le p$, as $\Theta \subset \mathbb{R}^p$ and $\dim_H$ is known to be Lipschitz subinvariant. Thus $\dim(\mathcal{S}) \le p$ and the conclusion follows by Theorem 2 (notice that, since Θ is compact and u is continuous, $\mathcal{S}$ is also compact).

  a2) Let $B \subseteq \Theta$ be a closed ball centered at $\boldsymbol{\mu}_0$ such that $u_{|B}$, the restriction of u to B, is injective. Then, $u_{|B} : B \to u(B)$ is a continuous bijection between compact metric spaces, which is enough to grant the existence and the continuity of the inverse map $u_{|B}^{-1}$. In particular, the sets B and $u(B)$

are homeomorphic and, by Theorem 1, $p = n_{\min}(B) = n_{\min}(u(B))$. Since $n_{\min}(u(B)) \leq n_{\min}(\mathcal{S})$, this proves (a2).

a3) As in the proof of a2), we notice that u admits a continuous inverse and is thus an homeomorphism between Θ and $\mathcal{S}$ (which are both compact). Then, $n_{\min}(\Theta) = n_{\min}(\mathcal{S})$ by Theorem 1. Finally, if Θ has nonempty interior, we may select a closed ball $B \subseteq \Theta$ and notice that $p = n_{\min}(B) \leq n_{\min}(\Theta) \leq p \implies n_{\min}(\Theta) = p$.

b1) Define the Hilbert space $\tilde{V} := \mathbb{R}^p \times V$ and the map $U : \Theta \to \tilde{V}$ as $U(\boldsymbol{\mu}) := (\boldsymbol{\mu}, u_{\boldsymbol{\mu}})$. Then U is both continuous and injective. Thus, by a3) we have $n_{\min}(\Theta) = n_{\min}(U(\Theta)) = n_{\min}(\mathcal{S}_\Theta)$.

$\square$

Remark. Theorem 3 holds for a generic Hilbert-valued map, meaning that the correspondence $\boldsymbol{\mu} \to u_{\boldsymbol{\mu}}$ needs not to involve the solution of a PDE. Because of this generality, some hypotheses cannot be weakened. For instance, one cannot replace the requirement of Lipschitz continuity in statement (a1) with continuity, mainly because of space-filling curves. As a counterexample, consider the Hilbert space of real square summable sequences, $V = \ell^2$. Then, by a straightforward application of the Hahn–Mazurkiewicz theorem (see Theorem 3-30 in [33]), there exists a continuous map from the unit interval to ℓ^2 whose image $\mathcal{S}$ is the so-called Hilbert cube, informally $\mathcal{S} = \prod_{n=1}^{+\infty}[0, 1/n]$. Therefore, being $\Theta := [0,1]$ the parameter space, we have a case in which $p = 1$ but $n_{\min}(\mathcal{S}) = +\infty$. In fact, for each $n \in \mathbb{N}$, the Hilbert cube contains an homeomorphic copy of the n-dimensional unit cube I^n. Thus, $n = n_{\min}(I^n) \leq n_{\min}(\mathcal{S})$ for all $n \geq 0$ and so $n_{\min}(\mathcal{S}) = +\infty$.

Before moving to the actual description of our Deep Learning approach, we conclude this Section with a practical application of the results we have presented so far. In particular, we focus on the case of second order elliptic PDEs.

2.2.1. *Second Order Elliptic PDEs.* In order to state the main result, we first provide some notation. We denote by Ω a bounded domain in $\mathbb{R}^d$ and by $\cdot$ the scalar product in $\mathbb{R}^d$. For $1 \leq q < +\infty$, we denote by $L^q(\Omega)$ the Lebesgue space of q-integrable real-valued maps; when $q = +\infty$, $L^\infty(\Omega)$ is defined as the Banach space of essentially-bounded maps. Similarly, we define the spaces $L^q(\Omega, \mathbb{R}^d)$ and $L^q(\Omega, \mathbb{R}^{d \times d})$ in the Bochner sense, where $\mathbb{R}^d$ is considered with the Euclidean norm $|\cdot|$ and $\mathbb{R}^{d \times d}$ with the operator norm, $|\boldsymbol{A}|_{\mathbb{R}^{d \times d}} := \sup_{|\boldsymbol{\xi}|=1} |\boldsymbol{A}\boldsymbol{\xi}|$. Given $k \in \mathbb{N}$, $1 \leq q < +\infty$, we write $W^{k,q}(\Omega)$ for the Sobolev space of all $w \in L^q(\Omega)$ that are k-times weakly differentiable with derivatives in $L^q(\Omega)$. We use $W_0^{k,q}(\Omega)$ to denote the subspace of all $w \in W^{k,q}(\Omega)$ that vanish on $\partial\Omega$, and we write $W^{-k,q}(\Omega)$ for the dual space of $W_0^{k,q}(\Omega)$ with respect to the duality product $\langle f, g \rangle \to \int_\Omega fg$. In order to prescribe Dirichlet boundary data, we also make use of the Sobolev-Slobodeckij spaces $W^{s,q}(\partial\Omega)$, where $s > 0$ is typically not an integer. All the aforementioned spaces are considered with their usual norms, see e.g. [22].

We define the sets of all admissible conductivity tensor-fields and transport fields, respectively $\Sigma(\Omega) \subset L^\infty(\Omega, \mathbb{R}^{d \times d})$ and $B(\Omega) \subset L^\infty(\Omega, \mathbb{R}^{d \times d})$, as follows:

- $\boldsymbol{\sigma} \in \Sigma(\Omega)$ if and only if it is uniformly elliptic, that is, there exists $\varepsilon > 0$ such that for almost all $x \in \Omega$ one has $\boldsymbol{\sigma}(x)\boldsymbol{\xi} \cdot \boldsymbol{\xi} \geq \varepsilon |\boldsymbol{\xi}|^2$ for all $\boldsymbol{\xi} \in \mathbb{R}^d$;
- $\boldsymbol{b} \in B(\Omega)$ if and only if it is differentiable and divergence free, that is, $\boldsymbol{b} \in \mathcal{C}^1(\Omega, \mathbb{R}^d)$ and $\mathrm{div}(\boldsymbol{b}) = 0$ in Ω.

We endow both $\Sigma(\Omega)$ and $B(\Omega)$ with the infinity norm $||\cdot||_\infty$. We are now able to state the following.

Theorem 4. *Let $\Omega \subset \mathbb{R}^d$ be a bounded domain with Lipschitz boundary, and let $\Theta \subset \mathbb{R}^p$ be a compact subset with nonempty interior. Let $q \geq 2d/(d+2)$ be finite, and define the conjugate exponent $q' := q/(q-1)$. Moreover, let $\boldsymbol{\mu} \to \boldsymbol{\sigma}_{\boldsymbol{\mu}} \in \Sigma(\Omega)$, $\boldsymbol{\mu} \to \boldsymbol{b}_{\boldsymbol{\mu}} \in B(\Omega)$, $\boldsymbol{\mu} \to f_{\boldsymbol{\mu}} \in W^{-1,q'}(\Omega)$ be parameter dependent coefficients and $\boldsymbol{\mu} \to g_{\boldsymbol{\mu}} \in W^{1/q',q}(\partial\Omega)$ boundary data. For each $\boldsymbol{\mu} \in \Theta$, we define $u_{\boldsymbol{\mu}} \in W^{1,q}(\Omega)$ as the unique solution to the following second order elliptic PDE*

$$u \in W^{1,q}(\Omega):$$

$$u_{|\partial\Omega} = g_{\boldsymbol{\mu}} \ and \ \int_\Omega \boldsymbol{\sigma}_{\boldsymbol{\mu}} \nabla u \cdot \nabla w + \int_\Omega (\boldsymbol{b}_{\boldsymbol{\mu}} \cdot \nabla u)\, w = \int_\Omega f_{\boldsymbol{\mu}} w \quad \forall w \in W_0^{1,q'}(\Omega),$$

Consider the solution manifold $\mathcal{S} := \{u_{\boldsymbol{\mu}}\}_{\boldsymbol{\mu}\in\Theta}$ as a subset of $V := L^2(\Omega)$. The following hold true:

- i) *if the dependence of $\boldsymbol{\sigma}_{\boldsymbol{\mu}}$, $\boldsymbol{b}_{\boldsymbol{\mu}}$, $f_{\boldsymbol{\mu}}$, $g_{\boldsymbol{\mu}}$ on $\boldsymbol{\mu}$ is Lipschitz continuous, then $n_{\min}(\mathcal{S}) \leq 2p + 1$.*
- ii) *if $\boldsymbol{\sigma}_{\boldsymbol{\mu}}$, $\boldsymbol{b}_{\boldsymbol{\mu}}$, $f_{\boldsymbol{\mu}}$, $g_{\boldsymbol{\mu}}$ depend continuously on $\boldsymbol{\mu}$ and the solution map $\boldsymbol{\mu} \to u_{\boldsymbol{\mu}}$ is one-to-one, then $n_{\min}(\mathcal{S}) = p$.*

Additionally, let $\mathcal{S}_\Theta := \{(\boldsymbol{\mu}, u_{\boldsymbol{\mu}})\}_{\boldsymbol{\mu}\in\Theta} \subset \mathbb{R}^d \times V$ be the augmented manifold. Then:

- iii) *if $\boldsymbol{\sigma}_{\boldsymbol{\mu}}$, $\boldsymbol{b}_{\boldsymbol{\mu}}$, $f_{\boldsymbol{\mu}}$, $g_{\boldsymbol{\mu}}$ depend continuously on $\boldsymbol{\mu}$, then $n_{\min}(\mathcal{S}_\Theta) = p$.*

Proof. First of all, we notice that if the data $\boldsymbol{\sigma}_{\boldsymbol{\mu}}, \boldsymbol{b}_{\boldsymbol{\mu}}, f_{\boldsymbol{\mu}}, g_{\boldsymbol{\mu}}$ depend continuously on $\boldsymbol{\mu}$, then so does $u_{\boldsymbol{\mu}} \in W^{1,q}(\Omega)$. This is easily proven by composition (cf. Lemma B.2 in the Appendix). Also, the compactness of Θ implies that of the subsets

$$\{\boldsymbol{\sigma}_{\boldsymbol{\mu}}\}_{\boldsymbol{\mu}\in\Theta} \subset \Sigma(\Omega), \quad \{\boldsymbol{b}_{\boldsymbol{\mu}}\}_{\boldsymbol{\mu}\in\Theta} \subset B(\Omega),$$

$$\{f_{\boldsymbol{\mu}}\}_{\boldsymbol{\mu}\in\Theta} \subset W^{-1,q'}(\Omega), \quad \{g_{\boldsymbol{\mu}}\}_{\boldsymbol{\mu}\in\Theta} \subset W^{1/q',q}(\partial\Omega).$$

Therefore, whenever the coefficients and the boundary data are Lipschitz continuous in $\boldsymbol{\mu}$, so is the solution map $\Theta \to W^{1,q}(\Omega)$ (by composition, cf. Lemma B.2). Finally, since $q \geq 2d/(d+2)$, we have the embedding $W^{1,q}(\Omega) \hookrightarrow L^2(\Omega)$ according to classical Sobolev inequalities (cf. Theorem 5.4 in [1]). In particular, all the aforementioned properties are preserved if we consider the parametric map as taking values in $V := L^2(\Omega)$. Statements (i), (ii) and (iii) now directly follow from Theorem 3. □

Remark. In Theorem 4, the PDE is firstly solved in the Banach space $W^{1,q}(\Omega)$ and the solution manifold is then embedded in the Hilbert space $L^2(\Omega)$. This construction allows for a large spectrum of PDEs where the solution $u_{\boldsymbol{\mu}}$ may exhibit singularities. A remarkable example is found for the dimensions $d = 2, 3$, where singular forces such as Dirac delta functions produce solutions $u_{\boldsymbol{\mu}} \notin H^1(\Omega) :=$

$W^{1,2}(\Omega)$ [12]. In these cases, the above Theorem still applies, e.g. with $q' = 4$ and $q = 4/3$ (cf. Morrey embedding). Nevertheless, we shall point out that in the Hilbert case, $q' = q = 2$, it is possible to restrict the state space to $V = H^1(\Omega) \subset L^2(\Omega)$. Note also that in this case the condition $q \geq 2d/(d+2)$ is redundant as it is trivially satisfied for any $d \geq 1$, coherently with the fact that $H^1(\Omega)$ always embeds in $L^2(\Omega)$.

3. Learning the Solution Manifold by means of Neural Networks

In the current Section, we present our Deep-Learning approach to reduced order modelling. Section 3.1 contains a brief recap on deep feedforward neural networks. The actual method, termed DL-ROM, is instead discussed in detail in Section 3.2. In particular, Section 3.2.1 and Section 3.2.2 respectively describe the design choices for the nonlinear dimensionality reduction and the approximation of the reduced map. Practical matters, such as the training of networks, are discussed in Section 3.2.3.

3.1. **Neural Networks.** Neural networks are nonlinear universal approximators that have recently gained a lot of popularity in several fields such as Machine Learning, Statistics and Approximation Theory. The fundamental building block of a neural network is the *layer*. Given two state spaces $V_1 := \mathbb{R}^{n_1}$, $V_2 := \mathbb{R}^{n_2}$ and a scalar-valued function $\rho : \mathbb{R} \to \mathbb{R}$, a layer L with activation ρ is a map $L : V_1 \to V_2$ of the form

$$L(\mathbf{v}) = \rho\left(\mathbf{W}\mathbf{v} + \mathbf{b}\right)$$

where $\mathbf{W} \in \mathbb{R}^{n_2 \times n_1}$ and $\mathbf{b} \in \mathbb{R}^{n_2}$ are respectively the weights and biases of the layer. Notice that, as $\mathbf{W}\mathbf{v} + \mathbf{b}$ is n_2-dimensional, the operation $\mathbf{W}\mathbf{v} + \mathbf{b} \to \rho\left(\mathbf{W}\mathbf{v} + \mathbf{b}\right)$ is intended componentwise.

Deep neural networks are defined through the composition of multiple layers. More precisely, a DNN with $l \geq 0$ *hidden layers* is a map of the form $\Phi := L_{l+1} \circ L_l... \circ L_1$, where each L_i is a layer. The layers $L_i : \mathbb{R}^{n_i} \to \mathbb{R}^{n_{i+1}}$, for $i = 1,..,l$, are called hidden layers, while L_{l+1} is the output layer. The latter is sometimes assumed to have no activation, as in [39, 55].

We refer to l as to the depth of the network. When $l = 1$, the adjective *deep* is usually dropped and Φ is said to be *shallow*. Note that we also allow for the degenerate case $l = 0$, where the NN actually reduces to the output layer. This is somewhat unusual, but it will help us in making the notation lighter. Finally, we say that Φ has activation function ρ, or equivalently that Φ is a ρ-DNN, if all of its (hidden) layers share that same activation.

The practical implementation of a neural networks is usually done as follows. At first, one designs the NN architecture. This corresponds to choosing (i) the depth l, (ii) the number of neurons within each layer L_i, i.e. the output dimension n_{i+1}, and (iii) suitable constraints on the weights and biases. Among other things, the latter is what makes the difference between dense layers (no constraints on $\mathbf{W}$) and convolutional layers ($\mathbf{W}$ is sparse and multiple values are shared), which heavily impacts on the network *complexity* [28].

The second step regards the network training. There, the weights and biases are tuned and optimized according to a suitable loss function. The optimization is typically performed using gradient descent algorithms [56].

To measure the network complexity, we count its degrees of freedom, that is the number of scalar parameters that are actually optimized during the training. Therefore, the contribution of each layer L_i depends on the corresponding type. For dense layers it is $(n_i + 1)n_{i+1}$, while it can be substantially lower in sparse and convolutional layers.

3.2. **Deep-Learning based Reduced Order Model.** We are given a parameter space $\Theta \subset \mathbb{R}^p$, a parameter dependent PDE and a high-fidelity FOM $\boldsymbol{\mu} \to \mathbf{u}^h_{\boldsymbol{\mu}} \in \mathbb{R}^{N_h}$. Our purpose is to approximate the solution map by means of a suitable neural network $\Phi : \mathbb{R}^p \to \mathbb{R}^{N_h}$. Through the whole section, we make the following assumption.

Assumption 2. *The parameter space Θ is compact and the solution map $\boldsymbol{\mu} \to \mathbf{u}^h_{\boldsymbol{\mu}}$ is continuous. Also, $\mathbf{u}^h_{\boldsymbol{\mu}} \neq 0$ for all $\boldsymbol{\mu} \in \Theta$.*

The last nonvanishing assumption $\mathbf{u}^h_{\boldsymbol{\mu}} \neq 0$ is a technicality related to the fact that in our experiments we quantify the network accuracy in terms of the mean relative error (MRE),

$$\mathrm{MRE}(\Phi) := \frac{1}{|\Theta|} \int_\Theta \frac{||\mathbf{u}^h_{\boldsymbol{\mu}} - \Phi(\boldsymbol{\mu})||_2}{||\mathbf{u}^h_{\boldsymbol{\mu}}||_2} d\boldsymbol{\mu},$$

where $|\Theta| > 0$ is the Lebesgue measure of Θ, while $|| \cdot ||_2$ is the Euclidean norm. Then, Assumption 2 ensures that $\mathrm{MRE}(\Phi)$ is finite. Clearly, other metrics may be used as well. Note that, up to the high-fidelity discretization and a rescaling (which does not impact the MRE), the errors are measured according to the L^2 distance.

In order to build Φ, we mimic the two steps paradigm of the Reduced Basis method, yielding the workflow depicted in Figure 2. First, we perform a nonlinear reduction that results in the identification of two networks, an encoder Ψ' and a decoder Ψ. We then exploit the encoder to construct and train a third network, ϕ, whose purpose is that of approximating the reduced map. The final model is then obtained by composition as $\Phi := \Psi \circ \phi$.

From now on, we shall fix once for all an activation $\rho : \mathbb{R} \to \mathbb{R}$ to be used for all networks and layers (except, possibly, for output layers). We assume ρ to be Lipschitz continuous and not a polynomial. A typical choice satisfying these requirements is the so-called α-leaky ReLU, i.e. $\rho(x) = x\mathbf{1}_{[0,+\infty)}(x) + \alpha x \mathbf{1}_{(-\infty,0)}(x)$ where $\alpha > 0$ is fixed.

3.2.1. *Dimensionality reduction: autoencoders and variants.* In order to obtain a low-dimensional representation of the solution manifold $\mathcal{S}^h := \{\mathbf{u}^h_{\boldsymbol{\mu}}\}_{\boldsymbol{\mu} \in \Theta} \subset \mathbb{R}^{N_h}$, we propose two approaches. The first one is based on the so-called autoencoder architecture, which involves two networks: an encoder, $\Psi' : \mathbb{R}^{N_h} \to \mathbb{R}^n$, and a decoder, $\Psi : \mathbb{R}^n \to \mathbb{R}^{N_h}$. The dimension n is often referred to as the *latent dimension*, and it is assumed to be the smallest one among all the layers in $\Psi \circ \Psi'$. Let $\mathcal{E}_n$ and $\mathcal{D}_n$ be the respectively the collections of all such encoders and decoders. Due to our
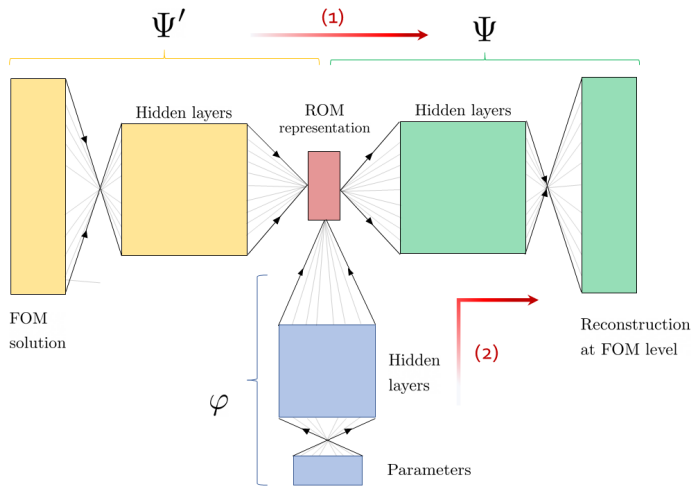
FIGURE 2. Workflow in the DL-ROM approach. The whole process consists of three neural networks, Ψ', Ψ and ϕ. First, the autoencoder $\Psi \circ \Psi'$ is trained over several high-fidelity snapshots in order to learn an approximation of the identity operator over the solution manifold (step 1). In this way, the encoder provides a low-dimensional representation of the FOM solutions, from $\mathbf{u}_{\boldsymbol{\mu}}^h \in \mathbb{R}^{N_h}$ to $\mathbf{u}_{\boldsymbol{\mu}}^n := \Psi'(\mathbf{u}_{\boldsymbol{\mu}}^h) \in \mathbb{R}^n$. At this point, the third network, ϕ, is trained to learn the map $\boldsymbol{\mu} \to \mathbf{u}_{\boldsymbol{\mu}}^n$ (step 2). Finally, the composition $\Phi := \Psi \circ \phi$ defines the DL-ROM approximation of the parameter-to-state map.

assumptions on ρ, these NNs are universal approximators (see e.g. [51]). In particular, under Assumption 2, we have a direct link with the nonlinear Kolmogorov n-width,

$$\delta_n(\mathcal{S}^h) = \inf_{\substack{\Psi' \in \mathcal{E}_n \\ \Psi \in \mathcal{D}_n}} \sup_{\boldsymbol{\mu} \in \Theta} ||\mathbf{u}_{\boldsymbol{\mu}}^h - \Psi(\Psi'(\mathbf{u}_{\boldsymbol{\mu}}^h))||_2.$$

Motivated by this fact, we choose $n := n_{\min}(\mathcal{S}^h)$ as latent dimension (or $n := n_{\min}(\mathcal{S})$, in case the analysis is conducted prior to the discretization). As discussed in Theorem 3, this often translates to $n \leq 2p + 1$, resulting in a massive reduction whenever $p \ll N_h$.

Note that, even after fixing n, the architectures for Ψ' and Ψ are still very flexible in both width and depth. Our experiments suggest that most of the autoencoder complexity (and depth) should be devoted to the decoder. To mitigate the fact that Ψ has very high-dimensional outputs, the decoder may take advantage from the use of sparse or convolutional layers.

A second possibility, as suggested by Theorem 3, is to perform the dimensionality reduction on the augmented manifold $\mathcal{S}_\Theta^h := \{(\boldsymbol{\mu}, \mathbf{u}_{\boldsymbol{\mu}}^h)\}_{\boldsymbol{\mu} \in \Theta} \subset \mathbb{R}^{N_h + p}$. Clearly, one may achieve this task using autoencoders as before. However, as our final objective is to approximate the map $\boldsymbol{\mu} \to \mathbf{u}_{\boldsymbol{\mu}}^h$, we are not actually interested in reconstructing the whole pair $(\boldsymbol{\mu}, \mathbf{u}_{\boldsymbol{\mu}}^h)$. For this reason, we consider a slight modification of the autoencoder, where the encoder is replaced with a *transcoder* $\Psi'_{\boldsymbol{\mu}} : \mathbb{R}^{N_h + p} \to \mathbb{R}^p$, while the decoder $\Psi : \mathbb{R}^p \to \mathbb{R}^{N_h}$ remains as before. Here the

subscript μ is meant to emphasize the presence of the parameters in the transcoder inputs. Notice that, in this case, we fix a priori the latent dimension to be equal to p. This is coherent with the fact that, if $\mathcal{T}_p$ is the collection of all transcoders, $\mathcal{T}_p := \{\Psi'_\mu : \mathbb{R}^{N_h+p} \to \mathbb{R}^p, \text{ s.t. } \Psi'_\mu \text{ is } \rho\text{-DNN}\}$, then under Assumption 2 we have

$$\inf_{\substack{\Psi'_\mu \in \mathcal{T}_p \\ \Psi \in \mathcal{D}_p}} \sup_{\mu \in \Theta} ||\mathbf{u}^h_\mu - \Psi(\Psi'_\mu(\mu, \mathbf{u}^h_\mu))||_2 = 0,$$

thanks to the approximation properties of networks. This is related to statement (b1) in Theorem 3, but it is also a straightforward consequence of the continuity of the solution map. Indeed, the above infimum remains zero even if we fix the transcoder to be $\Psi'_\mu(\mu, \mathbf{u}) = \mu$ (when possible). However, this is only an idealization: doing so would require Ψ to approximate the map $\mu \to \mathbf{u}^h_\mu$, which was our objective to begin with.

For what concerns the actual implementation, we follow the same rule of thumb as before. In general, we give more weight to the decoding part where we employ deep convolutional networks, while we use lighter architectures for the transcoding. For example, in our experiments we consider a limit case where $\Psi'_\mu(\mu, \mathbf{u}) = \rho(\mathbf{W}'\mu + \mathbf{W}\mathbf{u} + \mathbf{b})$. This corresponds to a 0-depth network where the output layer has p neurons and uses ρ as activation.

The two approaches adopt different perspectives and provide different advantages. The first one is completely based on the solution manifold $\mathcal{S}^h$, so it is likely to reflect intrinsic properties of the manifold itself. On the other hand, the second architecture ensures a maximal compression, the latent dimension being always equal to p. In particular, the low-dimensional representation $\mathbf{u}^n_\mu := \Psi'_\mu(\mu, \mathbf{u}^h_\mu) \in \mathbb{R}^p$ can be seen as an *alternative set of coordinates* for the solution manifold. In this sense, the map Ψ'_μ performs a transcoding.

3.2.2. *Approximation of the reduced map.* Once the dimensionality reduction is performed, we may move to the approximation of the reduced map $\mathbb{R}^p \ni \mu \to \mathbf{u}^n_\mu \in \mathbb{R}^n$, where either $\mathbf{u}^n_\mu := \Psi'(\mathbf{u}^h_\mu)$ or $\mathbf{u}^n_\mu := \Psi'_\mu(\mu, \mathbf{u}^h_\mu)$, depending on the adopted approach. We note that, by composition of $\mu \to \mathbf{u}^h_\mu$ and Ψ' (resp. Ψ'_μ), the reduced map is continuous, hence it can be approximated uniformly by some ρ-DNN $\phi : \mathbb{R}^p \to \mathbb{R}^n$. In general, we do not impose a particular structure on ϕ, rather we use a generic fully connected network with dense layers. To design the NN architecture in terms of number of layers and neurons, several theoretical results can be considered, especially in the case of ReLU and leaky-ReLU activations. In this respect, [62] provides error bounds in the case of uniform approximation, while [52] and [29] respectively discuss the approximation rates in terms of Lebesgue and Sobolev norms. Further insights can be found in [9, 17, 51, 59].

Once a suitable ϕ has been defined and trained (see Section 3.2.3), the DL-ROM is complete and the neural network $\Phi := \Psi \circ \phi$ yields the desired approximation of the parametric map $\mu \to \mathbf{u}^h_\mu$. For what concerns the accuracy, let us denote by

$$\mathrm{MRE}_1 := \frac{1}{|\Theta|} \int_\Theta \frac{||\mathbf{u}_{\boldsymbol{\mu}}^h - \Psi(\Psi'(\mathbf{u}_{\boldsymbol{\mu}}^h))||_2}{||\mathbf{u}_{\boldsymbol{\mu}}^h||_2} d\boldsymbol{\mu},$$

$$\mathrm{MRE}_2 := \frac{1}{|\Theta|} \int_\Theta \frac{||\mathbf{u}_{\boldsymbol{\mu}}^n - \phi(\boldsymbol{\mu})||_2}{||\mathbf{u}_{\boldsymbol{\mu}}^n||_2} d\boldsymbol{\mu},$$

where $\Psi'(\mathbf{u}_{\boldsymbol{\mu}}^h)$ is replaced with $\Psi'_\mu(\boldsymbol{\mu}, \mathbf{u}_{\boldsymbol{\mu}}^h)$ in the transcoder case. Then, recalling that ρ is Lipschitz, we have

$$\mathrm{MRE}(\Phi) \leq \mathrm{MRE}_1 + C\mathrm{MRE}_2$$

for some constant $C > 0$ that depends on the minimum $\min_{\mu \in \Theta} ||\mathbf{u}_{\boldsymbol{\mu}}^h||_2$ and on the Lipschitz constants of Ψ' and Ψ. In particular, in order to obtain a low error for Φ, it is sufficient that $\Psi \circ \Psi'$ and ϕ are accurate in terms of the corresponding MRE.

Note that the same ideas also apply to the case of *maximum errors*, that is when the integral over the parameter space is replaced with a supremum. This is in fact coherent with the worst-case approach that we adopted in Section 2.

Remark. We mention that, in the case $n = p$, an interesting alternative for ϕ could be provided by the so-called ODE-nets [13]. In fact, if the reduced map happens to be injective, then Θ and $\{\mathbf{u}_{\boldsymbol{\mu}}^n\}_{\mu \in \Theta}$ define two homeomorphic sets of coordinates. Even though homeomorphisms can be approximated by classical DNNs, we note that fully connected unconstrained networks can easily result in noninvertible models. In this sense, an alternative architecture which ensures the existence and continuity of ϕ^{-1} would be appealing. ODE-nets enjoys such property and have been proven to be universal approximators for homeomorphisms [63]. However, the development and implementation of ODE-nets is still in its infancy so we did not investigate this further.

3.2.3. *Implementation and training of the model.* As in the case of reduced order models for parametrized PDEs, the DL-ROM implementation consists of an offline and an online phase. During the offline phase, one first designs the architectures for Ψ', Ψ and ϕ. This concerns: (i) choosing the nature of Ψ' (encoder or transcoder) and consequently fixing the latent dimension n, (ii) defining the number of hidden layers in each network together with their types and corresponding number of neurons.

At this point, the networks are not operational yet, as their weights and biases need tuning first. To do so, the NNs are trained over a collection of high-fidelity snapshots obtained by extensive use of the FOM (see Algorithm 1). More precisely, having fixed a sample size N_{train}, we exploit the FOM in order to construct a training set $\{(\boldsymbol{\mu}_i, \mathbf{u}_{\boldsymbol{\mu}_i}^h)\}_{i=1}^{N_{\mathrm{train}}}$. The parameter values $\boldsymbol{\mu}_i$ can be chosen at will and they are usually sampled randomly from the parameter space. Note that this step can be computationally expensive, but it only impacts the offline stage. Once the snapshots have been computed, we train the autoencoder $\Psi \circ \Psi'$ by minimizing the empirical MRE

$$\mathcal{L}_1(\Psi', \Psi) = \frac{1}{N_{\text{train}}} \sum_{i=1}^{N_{\text{train}}} \left(\frac{||\mathbf{u}_{\boldsymbol{\mu}_i}^h - \Psi(\Psi'(\mathbf{u}_{\boldsymbol{\mu}_i}^h))||_2}{||\mathbf{u}_{\boldsymbol{\mu}_i}^h||_2} \right),$$

with obvious changes in transcoder case. The optimization of the loss function $\mathcal{L}_1$ is done using stochastic gradient descent algorithms, and results in the determination of the weights and biases for both Ψ' and Ψ. Then, we use the encoder (resp. transcoder) in order to obtain a low-dimensional version of the training set, that is $\{\boldsymbol{\mu}_i, \mathbf{u}_{\boldsymbol{\mu}_i}^n\}_{i=1}^{N_{\text{train}}}$ where $\mathbf{u}_{\boldsymbol{\mu}_i}^n := \Psi'(\mathbf{u}_{\boldsymbol{\mu}_i}^h)$. Subsequently, we train ϕ according to the loss function below

$$\mathcal{L}_2(\phi) = \frac{1}{N_{\text{train}}} \sum_{i=1}^{N_{\text{train}}} \left(\frac{||\mathbf{u}_{\boldsymbol{\mu}_i}^n - \phi(\boldsymbol{\mu}_i)||_2}{||\mathbf{u}_{\boldsymbol{\mu}_i}^n||_2} \right).$$

Notice that the optimization of $\mathcal{L}_2$ only involves ϕ, as the weights and biases of Ψ' are frozen.

At the end of the whole process, we let $\Phi := \Psi \circ \phi$ and the DL-ROM is fully operational. For each new $\boldsymbol{\mu} \in \Theta$, the corresponding solution $\Phi(\boldsymbol{\mu}) \approx \mathbf{u}_{\boldsymbol{\mu}}^h$ can be approximated online almost effortlessly, with very little computational cost.

To evaluate the model performance, we consider the empirical mean relative error

$$(4) \qquad\qquad \mathcal{L}(\Phi) = \frac{1}{N_{\text{test}}} \sum_{i=1}^{N_{\text{test}}} \left(\frac{||\mathbf{u}_{\boldsymbol{\mu}_i^{\text{test}}}^h - \Phi(\boldsymbol{\mu}_i^{\text{test}}))||_2}{||\mathbf{u}_{\boldsymbol{\mu}_i^{\text{test}}}^h||_2} \right).$$

The above is computed on a test set $\{(\boldsymbol{\mu}_i^{\text{test}}, \mathbf{u}_{\boldsymbol{\mu}_i^{\text{test}}}^h)\}_{i=1}^{N_{\text{test}}}$ that is drawn independently on the training set and uniformly over the parameter space. The quantity $\mathcal{L}(\Phi)$ measures the extent to which the model can generalize on unseen data, and it also provides an unbiased Monte Carlo estimate of $\text{MRE}(\Phi)$.

Remark. Once Ψ', Ψ and ϕ have been trained, the network Φ is ready for use. However, it is possible to further tune Φ by running a third training session. This can be done by unfreezing both ϕ and Ψ and then minimizing the empirical MRE on the training set. However, while providing a possible benefit for Φ, this might also decouple Ψ' and Ψ, making the autoencoder unusable.

Algorithm 1: DL-ROM training. Within the pseudo-code below, the map
$\mathcal{L}(\mathbf{y}, \mathbf{y}') := ||\mathbf{y} - \mathbf{y}'||_2 / ||\mathbf{y}||_2$ is defined as the relative error of $\mathbf{y}'$ w.r.t. $\mathbf{y}$.

Input : Training snapshots $\{\boldsymbol{\mu}_i, \mathbf{u}^h_{\boldsymbol{\mu}_i}\}_{i=1}^N$, reduced dimension n, optimizers
$\texttt{Optimizer}_1$, $\texttt{Optimizer}_2$, number(s) of epochs E_1, E_2, batch
size(s) S_1, S_2, encoding type $\underline{\texttt{useparameters}}$ (boolean).

Output: Neural network Φ approximating the parametric map.

$\Psi'_0, \Psi_0 \leftarrow$ Initialize encoder/transcoder and decoder with latent dimension n
$e \leftarrow 0$ Initialize epochs counter
$B_1 \leftarrow N/S_1 \quad$ // *number of batches*
if $\underline{\texttt{useparameters}}$ **then**
$\quad \big|\quad \mathbf{v}_i \leftarrow [\boldsymbol{\mu}_i, \mathbf{u}^h_{\boldsymbol{\mu}_i}] \quad$ // *transcoder case*
else
$\quad \big|\quad \mathbf{v}_i \leftarrow \mathbf{u}^h_{\boldsymbol{\mu}_i} \quad$ // *encoder case*
end
while $\underline{e < E_1}$ **do**
$\quad\big|\quad$ shuffle training data $\{\mathbf{v}_i, \mathbf{u}^h_{\boldsymbol{\mu}_i}\}_{i=1}^N$;
$\quad\big|\quad$ **for** $\underline{m = 1 : B_1}$ **do**
$\quad\big|\quad\big|\quad \mathbf{v}^{\text{batch}} \leftarrow \big[\mathbf{v}_{(m-1)S_1+1}, .., \mathbf{v}_{mS_1}\big]$
$\quad\big|\quad\big|\quad \mathbf{u}^{\text{batch}} \leftarrow \big[\mathbf{u}^h_{\boldsymbol{\mu}_{(m-1)S_1+1}}, .., \mathbf{u}^h_{\boldsymbol{\mu}_{mS_1}}\big]$
$\quad\big|\quad\big|\quad \text{loss} \leftarrow \frac{1}{S_1} \sum_{i=1}^{S_1} \mathcal{L}\big(\mathbf{u}^{\text{batch}}_i, \Psi_{eB_1+m-1}(\Psi'_{eB_1+m-1}(\mathbf{v}^{\text{batch}}_i))\big)$
$\quad\big|\quad\big|\quad \Psi'_{eB_1+m}, \Psi_{eB_1+m} \leftarrow \texttt{Optimizer}_1(\text{loss}, \Psi'_{eB_1+m-1}, \Psi_{eB_1+m-1})$
$\quad\big|\quad$ **end**
$\quad\big|\quad e \leftarrow e + 1$
end

$\phi_0 \leftarrow$ Initialize reduced map DNN
$e \leftarrow 0$ Reset epochs counter
$B_2 \leftarrow N/S_2 \quad$ // *number of batches*
$\mathbf{u}^n_{\boldsymbol{\mu}_i} \leftarrow \Psi'(\mathbf{v}_i) \quad$ // *define training data for ϕ*
while $\underline{e < E_2}$ **do**
$\quad\big|\quad$ shuffle training data $\{\boldsymbol{\mu}_i, \mathbf{u}^n_{\boldsymbol{\mu}_i}\}_{i=1}^N$;
$\quad\big|\quad$ **for** $\underline{m = 1 : B_2}$ **do**
$\quad\big|\quad\big|\quad \boldsymbol{\mu}^{\text{batch}} \leftarrow \big[\boldsymbol{\mu}_{(m-1)S_2+1}, .., \boldsymbol{\mu}_{mS_2}\big]$
$\quad\big|\quad\big|\quad \mathbf{u}^{n,\text{batch}} \leftarrow \big[\mathbf{u}^n_{\boldsymbol{\mu}_{(m-1)S_2+1}}, .., \mathbf{u}^n_{\boldsymbol{\mu}_{mS_2}}\big]$
$\quad\big|\quad\big|\quad \text{loss} \leftarrow \frac{1}{S_2} \sum_{i=1}^{S_2} \mathcal{L}\big(\mathbf{u}^{n,\text{batch}}_i, \phi_{eB_2+m-1}(\boldsymbol{\mu}^{\text{batch}}_i)\big)$
$\quad\big|\quad\big|\quad \phi_{eB_2+m} \leftarrow \texttt{Optimizer}_2(\text{loss}, \phi_{eB_2+m-1})$
$\quad\big|\quad$ **end**
$\quad\big|\quad e \leftarrow e + 1$
end

$\Phi \leftarrow \Psi_{E_1 B_1} \circ \phi_{E_2 B_2}$
return $\underline{\Phi}$

---

4. Numerical experiments

We now present some numerical results obtained with our DL-ROM approach. So far, neural networks have shown remarkable performances in the approximation of the parametric map at least in those contexts where classical POD-based methods succeed, e.g. [7, 26]. There is now an increasing interest in understanding how and if NNs can be of help in more challenging situations. In the case of transport problems, some theoretical and numerical results are now appearing in the literature, see respectively [40] and [24].

Here, we focus on parameter dependent second order elliptic PDEs. The first test case concerns an advection-diffusion problem with a singular source term. The PDE depends on 7 scalar parameters which affect the equation both in a linear and nonlinear fashion. We consider two variants of the same problem, one of which is transport-dominated.

As second test case, we consider a stochastic Poisson equation. The main difference with respect to the previous case is that the equation is parametrized by a stochastic process, and the PDE formally depends on an infinite-dimensional parameter. In order to apply the DL-ROM approach, we consider a suitable truncation of the Karhunen–Loève expansion of the stochastic process.

All our experiments were implemented in Python 3 and ran over GPUs. Specifically, we used the FEniCS library[1] to run the FOM and obtain the high-fidelity snapshots, while the construction and the training of the DL-ROM was handled in Pytorch[2].

4.1. **Stationary advection-diffusion with singular source.**

4.1.1. *Problem definition.* On the spatial domain $\Omega = (0,1)^2$, we define the sub-domains $\{\Omega_i\}_{i=0}^4$ as in Figure 3. We consider the following parameter dependent PDE in weak form

$$\frac{1}{10}\int_{\Omega_0}\nabla u \cdot \nabla w + \sum_{i=1}^4 \int_{\Omega_i}\mu_i\nabla u \cdot \nabla w + C\int_\Omega\left(\cos\mu_5\frac{\partial u}{\partial x_1}w + \sin\mu_5\frac{\partial u}{\partial x_2}w\right) =$$

$$= w(\mu_6,\mu_7)\quad \forall w \in \mathcal{C}_0^\infty(\Omega)$$

with homogeneous Dirichlet boundary conditions, $u_{|\Omega} = 0$. The above corresponds to a stationary advection-diffusion equation where: the conductivity field $\sigma_{\boldsymbol{\mu}} := 0.1 + \sum_{i=1}^4 \mu_i \mathbf{1}_{\Omega_i}$ is piecewise constant with values that change parametrically within the circular subdomains; the transport field $\boldsymbol{b}_{\boldsymbol{\mu}} := (C\cos\mu_5, C\sin\mu_5)$ has a parametrized direction while it is uniform in space and has a fixed intensity $C > 0$; finally, the source term $f_{\boldsymbol{\mu}}$ is a Dirac delta distribution located at the parameter dependent coordinates (μ_6, μ_7). Globally, the PDE depends on 7 parameters that we consider to be varying in the parameter space $\Theta = [0,1]^4 \times [0,2\pi] \times [0.1,0.9]^2$.

We note that the PDE does not admit solutions in $H_0^1(\Omega)$ because of the singularity introduced by the Dirac delta functions. Nevertheless, the variational problem

---

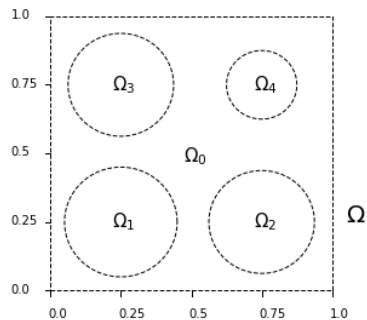[1]https://fenicsproject.org/
[2]https://pytorch.org/

FIGURE 3. Decomposition of the unit square $\Omega = (0,1)^2$ according to the conventions adopted in the first numerical experiment, Section 4.1.

is well-posed in the Banach space $W_0^{1,4/3}(\Omega) \hookrightarrow L^2(\Omega)$, see e.g. [12]. We are hence allowed to consider the solution manifold $\mathcal{S} := \{u_{\boldsymbol{\mu}}\}_{\boldsymbol{\mu} \in \Theta}$ as a subset of the Hilbert space $L^2(\Omega)$.

We analyze two different settings. In the first case we fix the transport field intensity to be $C = 0.5$, so that the diffusion and the advection act over the same scale. Then, we consider a transport-dominated case where $C = 40$.

4.1.2. *Discretization and Full Order Model.* As FOM, we employ Lagrange piecewise linear finite elements over a triangular mesh. Prior to the discretization, we provide a gaussian approximation of the Dirac delta as

$$f_{\boldsymbol{\mu}}^{\epsilon}(x_1, x_2) := \frac{1}{2\pi\epsilon^2} \exp\left(\frac{-(x_1 - \mu_6)^2 - (x_2 - \mu_7)^2}{2\epsilon^2} \right).$$

We shall write $u_{\boldsymbol{\mu}}^{\epsilon}$ for the solutions of this smoothed problem and $\mathcal{S}^{\epsilon}$ for the corresponding solution manifold. We see that the following claim holds (for the details see the Appendix, Section A)

Claim 1. We have $\sup_{\boldsymbol{\mu} \in \Theta} ||u_{\boldsymbol{\mu}} - u_{\boldsymbol{\mu}}^{\epsilon}||_{L^2(\Omega)} \to 0$ as $\epsilon \to 0$.

In particular, $\mathcal{S}^{\epsilon}$ approximates $\mathcal{S}$ uniformly. From here on, we shall fix $\epsilon = \frac{1}{420}$ and formally replace $\mathcal{S}$ with $\mathcal{S}^{\epsilon}$. Next, we discretize the variational problem through P1-Finite Elements over a triangular mesh. Using the classical estimates from the FEM theory, e.g. [54], it is not hard to see that Assumption 1 is satisfied within the state space $L^2(\Omega)$. Here, we fix the mesh size to be $h = 1/210$, which results in a high-fidelity space $V_h \cong \mathbb{R}^{N_h}$ of dimension $N_h = 44521$.

We exploit the FOM to generate respectively $N_{train} = 9000$ and $N_{test} = 1000$ random snapshots. In order to boost the training stability, we rescale all the snapshots by a factor $\alpha > 0$. Note that this operation has not impact on the MRE. We define the value of α either as

$$\max_{\substack{i=1,..,N_{train} \\ j=1,..,N_{test}}} \{\mathbf{u}_{\boldsymbol{\mu}_i}^h, \mathbf{u}_{\boldsymbol{\mu}_j^{test}}^h\} \qquad \text{or} \qquad \max_{\substack{i=1,..,N_{train} \\ j=1,..,N_{test}}} \{\boldsymbol{\mu}_i, \mathbf{u}_{\boldsymbol{\mu}_i}^h, \boldsymbol{\mu}_j^{test}, \mathbf{u}_{\boldsymbol{\mu}_j^{test}}^h\}$$

depending on whether the dimensionality reduction is performed using the autoencoder or the transcoder-decoder (in this latter case, we also scale the parameters).

The purpose is to deal with normalized inputs in [0,1]. Here rescaling by α is enough as all the solutions to the PDE are known to be nonnegative, see e.g. Lemma B.3.

4.1.3. *DL-ROM design and training.* In the construction of the DL-ROM, we do not make a distinction between the case of mild and strong advection, respectively $C = 0.5$ and $C = 40$. In this way, we can see more clearly how the intensity of the transport field affects the ROM performance.

For the dimensionality reduction, we explore both the two alternatives presented in Section 3.2.1. In the autoencoder case, we first need to identify the latent dimension n. We choose to consider the original solution manifold as a reference so that $n := n_{\min}(\mathcal{S})$. Thanks to Theorem 3, the claim below holds true.

Claim 2. $n_{\min}(\mathcal{S}) = p = 7$.

The proof is straightforward and we leave the details to the Appendix, Section A. Note that in this way, regardless of the encoding strategy, we are fixing the reduced dimension to be $n = 7$. Since $N_h = 44521$, this corresponds to a compression of almost 99.98%.

The networks architectures are reported in detail in Tables 1.a, 1.b (encoding step) and Table 1.c (decoding step). The encoder and the transcoder are particularly light, as they actually consist of a single dense layer. In contrast, Ψ is far more complex, with an architecture of depth $l = 4$. The decoder makes use of transposed convolutional layers, choice that is mainly motivated by two reasons: (i) convolutional layers correspond to sparse operators, and are more easy to deal with in the case of high-dimensional data, (ii) 2D convolutions best describe spatially localized behaviors so they are a natural choice when the data itself is defined on a spatial domain. We shall also remark that the decoder architecture is given in terms of a hyperparameter $m \in \mathbb{N}$, $m > 0$. This was done in order to investigate how the network complexity impacts the reconstruction error, which allows for a direct comparison with linear methods such as the POD. We analyze the performance of the networks for different values of m separately, namely $m = 4, 8, 16, 32$.

Prior to training, the networks are initialized differently depending on the encoding type. In the autoencoder case, we initialize both Ψ' and Ψ accordingly to the (gaussian) He initialization [31]. In the transcoder case, instead, we force the weights and biases of Ψ'_μ in such a way that $\Psi'_\mu(\boldsymbol{\mu}, u_{\boldsymbol{\mu}}) = \boldsymbol{\mu}$. This is equivalent to using the parameters as first guess for the intrinsic coordinates: then, during the training, and depending on the decoder needs, Ψ'_μ will have the possibility of finding other representations.

We train the autoencoder (resp. transcoder-decoder) $\Psi \circ \Psi'$ over the 9000 training snapshots using the AdamW optimizer [43], with learning-rate 10^{-4}, weight-decay 10^{-2}, moments coefficients $\beta_1 = 0.99$, $\beta_2 = 0.999$ and adjustment $\varepsilon = 10^{-8}$. We perform the gradient descent using batches of 50 and for a total of 1000 epochs.

At the end of this first training session, we pick the best performing architecture and continue the construction of the DL-ROM from there.

Table 1.d reports the architecture for our third network, ϕ. We initialize ϕ using the He initialization and proceed with its training accordingly to Section 3.2. For the gradient descent we employ the same optimizer as before, only changing the learning rate to 10^{-3}.

TABLE 1. Architectures for the Advection-Diffusion problem (Section 4.1). Tables (a) and (b) respectively concern the encoder and the transcoder, which can be thought as 0-depth network. Table (c) reports the decoder structure, which was chosen regardless of the encoding type. Here Transp. Conv. = Transposed Convolutional, denotes those sparse layers whose linear part is described in terms of 2D transposed convolutions. Therein, the input and output dimensions are written in the form $channels \times height \times width$, as each vector is reshaped in a 3D tensor. The hyperparameters kernel (which stands for *kernel size*) and *stride* determine the characteristics of the transposed convolution. For a more detailed explanation we refer to the Pytorch library documentation. Note that, up to reshaping, the output has dimension $1 \cdot 211 \cdot 211 = N_h$. The network architecture is given in terms of a hyperparameter $m \in \mathbb{N}$, through which we tune the NN complexity. Finally, Table (d) describes the change of coordinates DNN, ϕ. All networks and layers are considered with the 0.1-leaky ReLU activation.

a) Ψ'

Layer	Input	Output	dof
Dense	44521	7	311654

b) Ψ'_μ

Layer	Input	Output	dof
Dense	44528	7	311703

c) Ψ

Layer	Input	Output	Kernel	Stride	dof
Dense	7	$288m$	-	-	$2304m$
Transp. Conv.	$8m \times 6 \times 6$	$4m \times 20 \times 20$	10	2	$3200m^2 + 4m$
Transp. Conv.	$4m \times 20 \times 20$	$2m \times 48 \times 48$	10	2	$800m^2 + 2m$
Transp. Conv.	$2m \times 48 \times 48$	$m \times 102 \times 102$	9	2	$800m^2 + m$
Transp. Conv.	$m \times 102 \times 102$	$1 \times 211 \times 211$	8	2	$162m^2 + 1$

d) ϕ

Layer	Input	Output	dof
Dense	7	1024	7168
Dense	1024	512	524288
Dense	512	256	131072
Dense	256	7	1792

4.1.4. *Numerical results.* Let us first discuss the case $C = 0.5$. The results concerning the dimensionality reduction are reported in Figure 4 and Table 2. Figure 4 compares the performance of the autoencoder, the transcoder-decoder and the POD in terms of model complexity and MRE. For what concerns the POD, the degrees of freedom are defined as the number of entries in the projection matrix $\mathbf{V}$, while the errors are computed as $\|\mathbf{u} - \mathbf{V}\mathbf{V}^T\mathbf{u}\|_2 / \|\mathbf{u}\|_2$ (relative projection error). In particular, the MREs reported in Figure 4 provide a lower-bound for all POD-based ROMs.

Table 2 shows the MREs for the autoencoder and the transcoder-decoder obtained with $m = 32$. We note that, over the training set, POD would require around 800 modal bases in order to achieve the same accuracy of our nonlinear reduction. Similarly, more than 200 bases are needed to surpass the networks performance over the test set. This is caused by the slow decay of the Kolmogorov n-width $d_n(\mathcal{S}^h)$, which forces all POD-based ROMs to work with reduced dimensions $n \geq 200$. Conversely, the DL-ROM approach has the advantage of always providing the maximal compression, i.e. $n = 7$.

TABLE 2. Numerical results for the dimensionality reduction. Case $C = 0.5$ on top, $C = 40$ below. n_{POD} = minimum number of POD-bases (up to an error of 10) that are needed to outperform the DL-ROM. Complexity reduction $= (N_h n_{\mathrm{POD}} - \mathrm{dof})/(N_h n_{\mathrm{POD}})$, where *dof* are the degrees of freedom of the autoencoder (resp. transcoder-decoder). This quantity describes the reduction in complexity obtained by employing the nonlinear reduction in place of the POD.

$C = 0.5$

Data	Encoding	MRE	n_{POD}	Complexity reduction
Train	encoder	1.85%	840	88%
Train	transcoder	1.79%	880	90%
Test	encoder	4.53%	230	55%
Test	transcoder	3.85%	290	65%

$C = 40$

Data	Encoding	MRE	n_{POD}	Complexity reduction
Train	encoder	5.20%	≥ 1000	$\geq 90\%$
Train	transcoder	4.35%	≥ 1000	$\geq 90\%$
Test	encoder	23.40%	420	76%
Test	transcoder	17.70%	700	86%

For the final model, we selected the transcoder-decoder with $m = 32$. The numerical results, obtained after training ϕ and having set $\Phi := \Psi_{\boldsymbol{\mu}} \circ \phi$, are reported in Figure 5.a. The DL-ROM approximates well the parametric map, with an average test error of 5.81%. Also, the approximations appear to be qualitatively accurate even when the relative error is high (for instance in terms of the location of the solution peak).

Within the same Figures and Tables, but different panels, we have reported the corresponding results for the transport-dominated case, $C = 40$. For what concerns the training errors, we cannot remark a big difference with respect to the previous case. This indicates that, differently from POD, our method is able to exploit nonlinearities in order to capture the advective nature of the problem. However, we also see a slight overfitting trend. In fact, the test errors are now higher and the DL-ROM generalizes with more difficulty. Of note, this effect appears to be mitigated by the use of the transcoder (Figure 4.b).

Nevertheless, the DL-ROM reaches accuracies that are substantially higher with respect to the ones achievable using POD-based methods (cf. Table 2). Finally, we observe that test instances with high-relative errors are still qualitatively accurate in terms of peak location and flow direction (Figure 5).

We conclude with a final remark on the computational cost. In general, the offline stage required less than 4 hours, starting from the generation of the snapshots and the networks training (for $m = 32$, nearly an hour less for $m = 4$). The models are instead extremely fast during the online phase: solving the PDE for 1000 different values of the parameters requires less than 3 milliseconds on GPU.
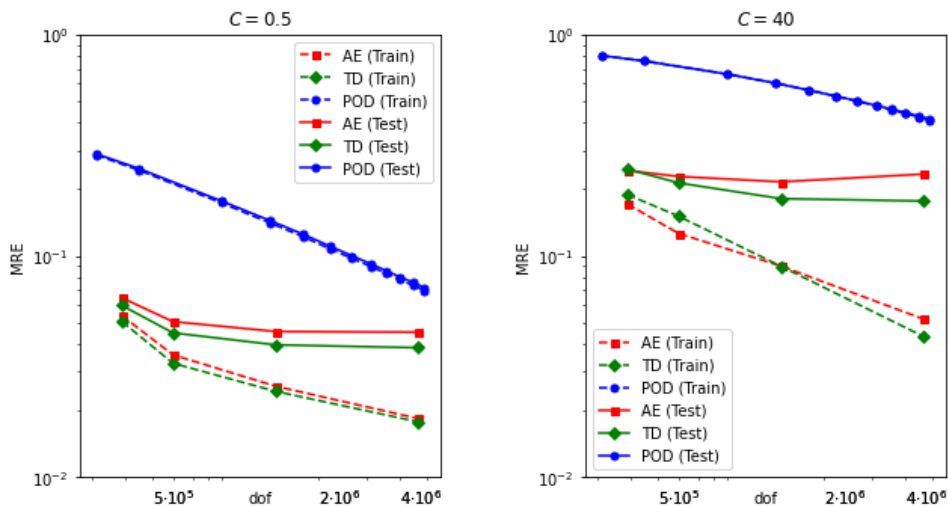
FIGURE 4. Error decay in terms of network complexity (first experiment, Section 4.1). The plots are reported in loglog scale. Case $C = 0.5$ on the left, $C = 40$ on the right. The red-dashed line corresponds to the training MREs of the autoencoders (AE), obtained respectively with $m = 4, 8, 16, 32$. The red straight line, instead, refers to the test errors. The green lines read similarly but concern the transcoder-decoder networks (TD). Finally, the linear benchmark provided by the POD is reported in blue. In this latter case, the degrees of freedom (dof) are computed as nN_h and correspond to the number of entries in the projection matrix $\mathbf{V}$ (assuming n is the reduced dimension). At this regard, we remark that, for the POD, the reduced dimension increases as the complexity grows. Conversely, the DL-ROM approach considers heavier and heavier networks, but the reduced dimension is always fixed to $n = p = 7$.

4.2. Stochastic Poisson equation.

4.2.1. *Problem definition.* On the spatial domain $\Omega = (0, 1)^2$, we consider a centered Gaussian process W with covariance kernel $\mathrm{Cov}(\mathbf{x}, \mathbf{y}) = 10\exp(-4|\mathbf{x} - \mathbf{y}|^2)$. The latter is used to model the stochastic Poisson equation below.

$$\begin{cases} -\mathrm{div}\left(\left(10^{-1} + \mathrm{e}^{W(\omega)}\right)\nabla u\right) = |\mathbf{x}|^2 & \text{in } \Omega \\ u = 0 & \text{on } \partial\Omega \end{cases}$$

Here, for each event ω, the map $W(\omega) : \Omega \to \mathbb{R}$ denotes the corresponding path of the stochastic process W. The problem above can be seen as a parameter dependent PDE that depends on (countably) infinite many parameters. To see this, we recall that there exist positive real numbers $\{\sqrt{\lambda_i}\}_{i=1}^{+\infty}$, orthonormal functions $\{\zeta_i\}_{i=1}^{+\infty} \subset L^2(\Omega)$ and independent standard gaussians $\{X_i\}_{i=0}^{+\infty}$ such that $W = \sum_{i=1}^{+\infty} \sqrt{\lambda_i}X_i\zeta_i$ almost surely. The latter is the so-called Karhunen-Loève expansion of W. We assume the λ_i coefficients to be nonincreasing in i.

In order to cast the problem into our framework, we approximate W by truncating the aforementioned expansion at some index k. More precisely, we define $\Theta := \mathbb{R}^k$ and $W_{\boldsymbol{\mu}}^k$ as

a) Case $C = 0.5$,   b) Case $C = 40$

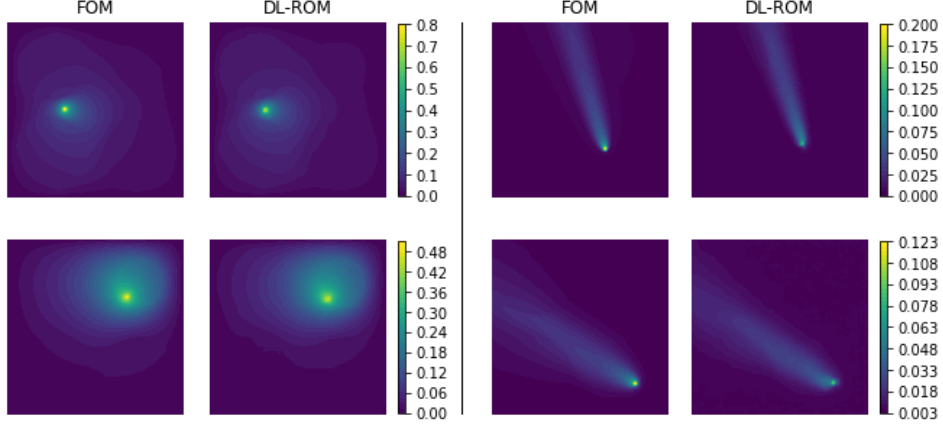Train MRE = 3.16%, Test MRE = 5.81%.   Train MRE = 5.19%, Test MRE = 20.38%.

FIGURE 5. DL-ROM results for the Advection-Diffusion problem (Section 4.1). Panels (a) and (b) respectively refer to the case of mild and strong advection. In both cases, the picture shows two examples extracted from the test set (one for each row) and compares the high-fidelity solution (first column) with the DL-ROM approximation (second column). Relative errors of the approximations in panel (a) are: 8.88% (top) and 4.42% (bottom). Relative errors for panel (b) are: 25.48% (top) and 12.66% (bottom).

$$W_{\boldsymbol{\mu}}^k(x) := \sum_{i=1}^k \sqrt{\lambda_i} \mu_i \zeta_i(x).$$

Thanks to the usual continuity results and the convergence ensured by the Karhunen-Loève expansion, the impact of this substitution on the PDE can be made arbitrarily small with k. Note that on Θ we do not consider the Lebesgue measure, but the measure induced by the k-variate standard Gaussian. This will make a difference in the definition of the MRE.

4.2.2. *Discretization and Full Order Model.* On Ω we define a triangular mesh of size $h = 10^{-2}$, over which we construct the high-fidelity space of piecewise linear Finite Elements V_h. The corresponding FOM dimension is $N_h = 10121$.

As starting point, we compute an approximation of the Karhunen-Loève expansion of W. To do so, we project and solve over V_h the following eigenvalue problem.

$$\int_\Omega \mathrm{Cov}(\mathbf{x}, \mathbf{y}) \zeta_i(\mathbf{y}) d\mathbf{y} = \lambda_i \zeta_i(\mathbf{x}).$$

In particular, we compute the first k eigenvalues λ_i and corresponding eigenfunctions $\zeta_i \in V_h$ for which

$$0.9 \leq \frac{\sum_{i=1}^k \lambda_i}{\sum_{i=1}^{+\infty} \lambda_i} = \frac{1}{10} \sum_{i=1}^k \lambda_i,$$

where the last equality is easily deduced by the covariance kernel, as $\sum_{i=1}^{+\infty} \lambda_i = \int_\Omega \mathrm{Cov}(\mathbf{x}, \mathbf{x}) d\mathbf{x} = \int_{[0,1]^2} 10 d\mathbf{x} = 10$. This procedure results in the choice of the truncation index $k = 38$. From a statistical point of view, we say that $W_{\boldsymbol{\mu}}^k$ explains at least 90% of the variability in W.

After having defined $W_{\boldsymbol{\mu}}^k$, we run the FOM and generate respectively $N_{\text{train}} = 9000$ and $N_{\text{test}} = 1000$ snapshots. The parameter values are sampled from Θ independently and accordingly to a k-variate standard Gaussian distribution. In this way, the empirical MRE as defined in Section 3.2, equation (4), is automatically a Monte Carlo estimate of

$$(5) \qquad \mathbb{E}_{\boldsymbol{\mu} \sim \mathcal{N}_k(0, I_k)} \left[\frac{||\mathbf{u}_{\boldsymbol{\mu}}^h - \Phi(\boldsymbol{\mu})||_2}{||\mathbf{u}_{\boldsymbol{\mu}}^h||_2} \right]$$

where $\mathbb{E}[\,\cdot\,]$ is the expectation operator and $\mathcal{N}_k(0, I_k)$ denotes the k-variate standard gaussian distribution. The above is a natural adaptation of the MRE in Section 3.2 to the case of stochastic parameters.

We note that, in this case, the parameter space Θ is not compact, as it is unbounded. Nevertheless, since Θ has finite measure with respect to the gaussian distribution, it is straightforward to adapt the reasoning in Section 3.2 to this context. For instance, the error defined in (5) can be made arbitrarily small provided that Φ is sufficiently accurate within some compact subdomain $\Theta_M := \{\boldsymbol{\mu} \in \mathbb{R}^p \text{ s.t. } |\boldsymbol{\mu}| < M\}$. A more in depth discussion on the regularity of the parametric map in the case of stochastic coefficients can be found in [3].

4.2.3. *DL-ROM design.* For the dimensionality reduction, we employ a transcoder-decoder. This is to ensure a maximal compression, as the number of parameters is already mildly large. The network topology is reported in Tables 3.a, 3.b. Coherently with the chosen approach, we fix the reduced dimension to be $n := k = p = 38$. In general, the architecture is very similar to those considered in Section 4.1, the only difference being in the specifics of the convolutional layers.

As before, we normalize the snapshots and adopt the He initialization for the decoder while we force the initial state of the transcoder to behave as $\Psi'_\mu(\boldsymbol{\mu}, u_{\boldsymbol{\mu}}) = \boldsymbol{\mu}$. We train $\Psi \circ \Psi'_\mu$ using stochastic gradient descent with minibatches of size 10 and for a total of 1200 epochs. For the optimization we employ the Adamax optimizer [36], with default parameters and learning rate of 10^{-3}. Here, the choice of Adamax over AdamW is motivated by the fact that the former is known to be more stable. Indeed, we observed high oscillations of the loss values when employing AdamW, possibly due to the fact that the snapshots come in rather different scales when compared one another.

Table 3.c reports the architecture for reduced map network, ϕ. We train ϕ using the Adamax optimizer with batch size 50 and learning rate 5e-3, for a total of 5000 epochs. The whole offline stage of the DL-ROM took around 4 hours.

4.2.4. *Numerical results.* The dimensionality reduction is satisfactory, with mean relative errors of 1.10% and 2.57% respectively on the training and test sets. For benchmark, we also computed the MREs obtained using POD for the same reduced dimension, that is $n = 38$. In this case, the MREs rise respectively to 4.84% on the training test, and to 4.75% on the test set.

TABLE 3. Architectures for the Stochastic Poisson equation (Section 4.2). Tables (a) and (b) together describe the transcoder-decoder, while (c) concerns the change of coordinates map ϕ. All layers are considered with the 0.1-leaky ReLU activation.

a) Ψ'_μ

Layer	Input	Output	dof
Dense	10239	38	389120

b) Ψ

Layer	Input	Output	Kernel	Stride	dof
Dense	38	18432	-	-	718848
Transp. Conv.	$512 \times 6 \times 6$	$256 \times 12 \times 12$	2	2	524544
Transp. Conv.	$256 \times 12 \times 12$	$128 \times 24 \times 24$	2	2	131200
Transp. Conv.	$128 \times 24 \times 24$	$64 \times 48 \times 48$	2	2	32832
Transp. Conv.	$64 \times 48 \times 48$	$1 \times 101 \times 101$	7	2	3137

c) ϕ

Layer	Input	Output	dof
Dense	7	1024	7168
Dense	1024	512	524288
Dense	512	256	131072
Dense	256	7	1792

Relative error: 5.90%   Relative error: 23.30%   Relative error: 4.19%
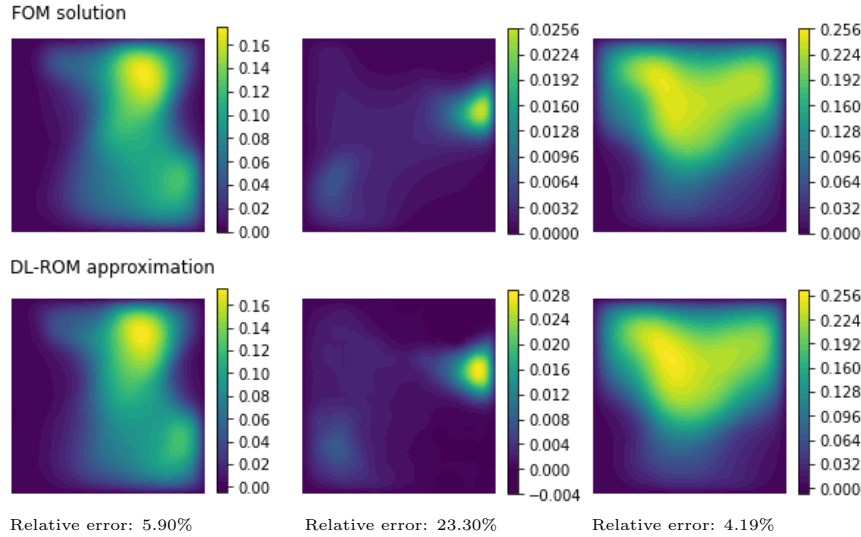
FIGURE 6. DL-ROM results for the Stochastic Poisson equation (Section 4.2). The picture shows three examples coming from the test set. The first row reports the high-fidelity solutions, while the second row displays the corresponding DL-ROM approximations. Relative errors are also reported.

The approximation of the reduced map, instead, appears to be more challenging. While the final model is able to approximate the parameter-to-state map with an error of 4.69% over the training set, the inaccuracy increases to 12.50% on the test set. Figure 6 reports some examples extracted from the test set.

4.3. **A final remark on the choice of the latent dimension.** In both experiments, Section 4.1 and Section 4.2, the reduced dimension was equal to the number of parameters, that is $n = p$. For the advection-diffusion problem, this was motivated by the fact that $n_{\min}(\mathcal{S}) = p$. Conversely, for the stochastic Poisson equation, we fixed $n = p$ due to our design choice of using a transcoder-decoder.

However, this is not always the case. In fact, as we argued in Section 2, it is possible that $n_{\min}(\mathcal{S}) > p$, in which case the autoencoder latent dimension should exceed the number of parameters. As an example, consider the following boundary value problem,

$$(6) \qquad \begin{cases} -\Delta u + 10 \left(\cos \mu, \sin \mu\right) \cdot \nabla u = 10 \mathrm{e}^{-100|\mathbf{x}-\mathbf{x}_0|} & \text{in } \Omega \\ u = 0 & \text{on } \partial\Omega, \end{cases}$$

where $\Omega := (0,1)^2$, $\mathbf{x}_0 := (0.5, 0.5)$ and $\mu \in \Theta := [0, 2\pi]$. In this case, $p = 1$ but $n_{\min}(\mathcal{S}) = 2$. In fact, by considering $\mathcal{S}$ either as a subset of $L^2(\Omega)$ or $H^1(\Omega)$, and applying the same ideas as in the proof of Claim 1, it is not hard to see that the solution manifold $\mathcal{S}$ is homeomorphic to the unit circle. Therefore, the DL-ROM approach requires an autoencoder with latent dimension $n = 2$. In Figure 7 we have summarized the results obtained with this design choice for this particular problem. We do not report the network architectures, as that is not the focus of our discussion here.

We note that the low-dimensional representation of $\mathcal{S}$ is given by a curve in $\mathbb{R}^2$, coherently with the fact that $\phi : \mathbb{R} \to \mathbb{R}^2$. It is interesting to see that the DL-ROM representation of $\mathcal{S}$ actually resembles a circle. We also note that the curve $\phi(\Theta) \subset \mathbb{R}^2$ is not smooth. This is not caused by the PDE itself, which is very regular, by rather by the use of the ReLU activation for ϕ.

For the sake of completeness, we mention that for this example: we considered the same FOM as in Section 4.2; we trained and tested the DL-ROM respectively over 900 and 100 snapshots; the model reported average relative errors below 2% both on the training and test set; the offline stage took less than 10 minutes (on GPU).

5. Conclusions

We developed a novel deep learning approach for reduced order modelling of parameter dependent PDEs, here termed DL-ROM, where the solution map is approximated by a deep neural network Φ. Our construction is based on the use of autoencoders, which we employ as nonlinear alternatives to other reduction techniques such as the POD. In the DL-ROM approach, the latent dimension is chosen to be the smallest one that grants arbitrary accuracy. The value of such dimension was investigated in detail in Section 2. There, we proved theoretical results (Theorems 3 and 4) that can be used as guidelines in practical applications.

The results obtained in our experiments are promising. The DL-ROM appears to be a captivating alternative to traditional ROMs, especially in challenging situations
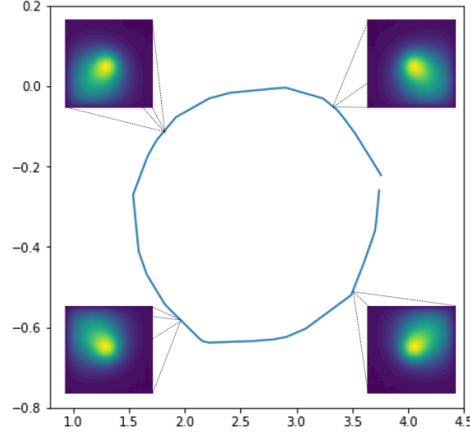
FIGURE 7. Low-dimensional representation of the solution manifold (boundary value problem (6), Section 4.3). Each point on the curve $\phi(\Theta) \subset \mathbb{R}^2$ corresponds to a given solution of the parametrized PDE. The picture also shows the reconstruction $\Psi(\phi(\mu))$ obtained for four different values of $\mu \in \Theta$.

where linear models fail. Our first test case, Section 4.1, shows that the method is able to handle transport-dominated problems and that it behaves well in presence of singularities. Strong transport fields seem to impact on the DL-ROM generalizability, which may be improved by increasing the number of training samples or by including physical terms in the loss function. While we wish to investigate this further in future works, we acknowledge that multiple researchers are now working on this topic, e.g. [46, 58]. Good results are also obtained for high-dimensional parameter spaces, Section 4.2. However, differently from [7, 44], our approach does not directly apply to the case of infinite-dimensional parameters.

We conclude with a few comments on the computational cost. While the offline stage is clearly expensive, our design choices allow for a significant reduction in the model complexity, which results in architectures that are easier to train (cf. e.g. [24, 26]). Nevertheless, the DL-ROM is extremely fast when used online. This makes the method suited for demanding tasks with multiple queries, as the ones typical of Uncertainty Quantification.

APPENDIX A. PROOF OF THE CLAIMS IN SECTION 4

Proof of Claim 1. Let $\boldsymbol{\mu} \in \Theta$. For the sake of brevity, define $\mathbf{x}_{\boldsymbol{\mu}} := (\mu_6, \mu_7) \in \Omega$. We shall recall that, by Morrey's embedding theorem [22], we have $W_0^{1,4}(\Omega) \hookrightarrow \mathcal{C}^{0,1/2}(\Omega)$, the latter being the space of 1/2-Hölder maps. As a consequence, for any $w \in W^{1,4}(\Omega)$, we have

$$\left| w(\mathbf{x}_{\boldsymbol{\mu}}) - \int_{\Omega} f_{\boldsymbol{\mu}}(\mathbf{z})^{\epsilon} w(\mathbf{z}) d\mathbf{z} \right| = \left| \int_{\Omega} (w(\mathbf{x}_{\boldsymbol{\mu}}) - w(\mathbf{z})) f_{\boldsymbol{\mu}}^{\epsilon}(\mathbf{z}) d\mathbf{z} \right| \leq$$

$$\leq \int_{\Omega} |w(\mathbf{x}_{\boldsymbol{\mu}}) - w(\mathbf{z})| f_{\boldsymbol{\mu}}^{\epsilon}(\mathbf{z}) d\mathbf{z} \leq C' \|w\|_{W_0^{1,4}(\Omega)} \int_{\mathbb{R}^2} |\mathbf{x}_{\boldsymbol{\mu}} - \mathbf{z}|^{1/2} f_{\boldsymbol{\mu}}^{\epsilon}(\mathbf{z}) d\mathbf{z}$$

for a constant $C' > 0$ independent on both w and $\boldsymbol{\mu}$. The change of variables $\mathbf{y} := (\mathbf{z} - \mathbf{x}_{\boldsymbol{\mu}})/\epsilon$ then yields

$$\left| w(\mathbf{x}_{\boldsymbol{\mu}}) - \int_\Omega f_{\boldsymbol{\mu}}(\mathbf{z})^\epsilon w(\mathbf{z}) d\mathbf{z} \right| \leq .. \leq C' ||w||_{W_0^{1,4}(\Omega)} \epsilon^{1/2} \int_{\mathbb{R}^2} |\mathbf{y}|^{1/2} G(\mathbf{y}) d\mathbf{y}$$

where G is the probability density of the standard normal distribution in $\mathbb{R}^2$. By passing at the supremum over w with $||w||_{W_0^{1,4}(\Omega)} = 1$ and $\boldsymbol{\mu} \in \Theta$ we get

$$\sup_{\boldsymbol{\mu} \in \Theta} ||f_{\boldsymbol{\mu}} - f_{\boldsymbol{\mu}}^\epsilon||_{W^{-1,4}(\Omega)} \leq C'' \epsilon^{1/2}$$

for some constant $C'' > 0$. By classical stability estimates for elliptic PDEs, see e.g. Lemma B.1, we then have $\sup_{\boldsymbol{\mu} \in \Theta} ||u_{\boldsymbol{\mu}} - u_{\boldsymbol{\mu}}^\epsilon||_{W_0^{1,4/3}(\Omega)} \leq 10 C'' \epsilon^{1/2}$, as $\boldsymbol{\sigma}_{\boldsymbol{\mu}}(\mathbf{x}) \geq 10^{-1}$ for all $\boldsymbol{\mu} \in \Theta$. Up to the embedding the solution manifold in $L^2(\Omega)$, the claim now follows. $\qquad\square$

Proof of Claim 2. The idea is to re-parametrize the solution manifold, as the given parametrization suffers from the lack of injectivity. In fact, both $\mu_5 = 0$ and $\mu_5 = 2\pi$ return the same advective field (and we cannot exclude one extreme, or Θ would lose its compactness). To do so, let S^1 be the unit circle in $\mathbb{R}^2$. We define the hypercylinder $\Theta' := [0,1]^4 \times S^1 \times [0.1, 0.9]^2$. We will adopt a seven component notation as before, even though $\Theta' \subset \mathbb{R}^8$, as $\boldsymbol{\mu}_5 \in S^1$ is now 2-dimensional. We re-parametrize the coefficients of the PDE in terms of this new coordinates in the obvious way, especially for $\boldsymbol{\sigma}_{\boldsymbol{\mu}'}$ and $f_{\boldsymbol{\mu}'}$. For the advective field we let $\boldsymbol{b}_{\boldsymbol{\mu}'} := \boldsymbol{\mu}_5$. We shall now prove that: (i) the new parameter space satisfies $n_{\min}(\Theta') = 7$, (ii) the new parametric map $\boldsymbol{\mu}' \to u_{\boldsymbol{\mu}'}$ is continuous and (iii) injective. Claim 2 then follows by Theorem 3.

Proof that $n_{\min}(\Theta') = 7$. Consider the map $\phi : \Theta' \to \mathbb{R}^7$ given by

$$\phi(\boldsymbol{\mu}') = (\boldsymbol{\mu}_5'(1 + \mu_1'), \ \mu_2', \ \mu_3', \ \mu_4', \ \mu_6', \ \mu_7').$$

Then the image $\phi(\Theta') = \{\mathbf{z} \in \mathbb{R}^2 : 1 \leq |\mathbf{z}| \leq 2\} \times [0,1]^3 \times [0.1, 0.9]^2 \subset \mathbb{R}^7$ has nonempty interior. In particular, $n_{\min}(\phi(\Theta')) = 7$. Since ϕ clearly admits a continuous inverse, $\phi^{-1} : \phi(\Theta') \to \Theta'$, we conclude that $n_{\min}(\Theta') = 7$.

Proof that the parametric map $\boldsymbol{\mu}' \to u_{\boldsymbol{\mu}'}$ is continuous. Clearly $\boldsymbol{\sigma}_{\boldsymbol{\mu}'}$ and $\boldsymbol{b}_{\boldsymbol{\mu}'}$ depend continuously on $\boldsymbol{\mu}'$. Using again the embedding $W_0^{1,4}(\Omega) \hookrightarrow \mathcal{C}^{0,1/2}(\Omega)$ as in the proof of Claim 1, it is also easy to see that the map $\boldsymbol{\mu}' \to f_{\boldsymbol{\mu}'}$ is $\Theta' \to W^{-1,4}(\Omega)$ Hölder continuous. By composition (see Lemma B.2), we then obtain the continuity of the parametric map.

Proof that the parametric map $\boldsymbol{\mu}' \to u_{\boldsymbol{\mu}'}$ is injective. Let $\boldsymbol{\mu}', \boldsymbol{\mu}'' \in \Theta'$ and assume that $u \in W_0^{1,4/3}(\Omega)$ is a solution for both parameters, that is $u = u_{\boldsymbol{\mu}'} = u_{\boldsymbol{\mu}''}$. Classical results on inner regularity of solutions to elliptic PDEs ensure that $u_{\boldsymbol{\mu}'}$ is locally H^1 at all points except at the location of the Dirac delta $f_{\mu'}$. The analogue holds for $u_{\boldsymbol{\mu}''}$, so clearly it must be $\mu_6' = \mu_6''$ and $\mu_7' = \mu_7''$ in order for the solutions

to coincide. Next, let $w \in \mathcal{C}_0^\infty(\Omega_0)$ and extend it to zero on $\Omega \setminus \Omega_0$. Using w as test function for the equations of both $\boldsymbol{\mu}'$ and $\boldsymbol{\mu}''$ and then subtracting term by term yields

$$C \int_{\Omega_0} (\boldsymbol{b}_{\boldsymbol{\mu}'} - \boldsymbol{b}_{\boldsymbol{\mu}''}) \cdot \nabla u w = 0.$$

As w is arbitrary, it follows that ∇u is orthogonal to $(\boldsymbol{b}_{\boldsymbol{\mu}'} - \boldsymbol{b}_{\boldsymbol{\mu}''})$ on Ω_0. In particular, if $\boldsymbol{b}_{\boldsymbol{\mu}'} \neq \boldsymbol{b}_{\boldsymbol{\mu}''}$, then u must be constant along the direction $(\boldsymbol{b}_{\boldsymbol{\mu}'} - \boldsymbol{b}_{\boldsymbol{\mu}''})$ within Ω_0. But, because of the boundary conditions, this would make u identically zero near at least one edge of $\partial\Omega$. However, this is a contradiction, as classical maximum principles ensure that $u > 0$ a.e. in Ω (see e.g. Lemma B.3). It follows that $\boldsymbol{b}_{\boldsymbol{\mu}'} = \boldsymbol{b}_{\boldsymbol{\mu}''}$ and so $\boldsymbol{\mu}'_5 = \boldsymbol{\mu}''_5$. We now notice that, by subtracting the equations for $\boldsymbol{\mu}'$ and $\boldsymbol{\mu}''$, we have

$$\sum_{i=1}^{4} (\mu'_i - \mu''_i) \int_{\Omega_i} \nabla u \cdot \nabla w = 0 \quad \forall w \in \mathcal{C}_0^\infty(\Omega).$$

Fix any $i \in \{1, 2, 3, 4\}$ and let $v \in \mathcal{C}^\infty(\Omega_i)$. Define $w \in \mathcal{C}^\infty(\Omega_i)$ to be any of the strong solutions to the PDE $-\Delta w = v$ with homogeneous Neumann boundary condition on $\partial\Omega_i$. Since the subdomains are clearly separated, it is possible to extend w on the whole domain Ω so that w is still smooth but also vanishes on $\partial\Omega$ and on Ω_j for all $j \neq i$. Using such w in the last identity above and integrating by parts yields

$$0 = (\mu'_i - \mu''_i) \int_{\Omega_i} \nabla u \cdot \nabla w = (\mu'_i - \mu''_i) \int_{\Omega_i} u(-\Delta w) = (\mu'_i - \mu''_i) \int_{\Omega_i} uv.$$

Now assume that $\mu'_i \neq \mu''_i$. Then $\int_{\Omega_i} uv = 0$ for all $v \in \mathcal{C}^\infty(\Omega_i) \implies u_{|\Omega_i} \equiv 0$, contradiction. Then $\mu'_i = \mu''_i$ and thus $\boldsymbol{\mu}' = \boldsymbol{\mu}''$, as claimed. $\qquad\square$

Appendix B. Auxiliary results on Partial Differential Equations

Lemma B.1. *Let $(V, ||\cdot||_V)$ and $(W, ||\cdot||_W)$ be two Banach spaces, with W reflexive. Let $(W^*, ||\cdot||_*)$ be the dual space of W and define $(\mathfrak{B}(V,W), |||\cdot|||)$ as the normed space of bounded bilinear forms $V \times W \to \mathbb{R}$, where*

$$|||a||| := \sup_{\substack{||v||_V=1 \\ ||w||_W=1}} |a(v,w)|.$$

Let $\mathfrak{B}_c(V,W) \subset \mathfrak{B}(V,W)$ be the subset of coercive bounded bilinear forms, i.e. $a \in \mathfrak{B}(V,W)$ for which

$$\lambda(a) := \inf_{||v||_V=1} \sup_{||w||_W=1} |a(v,w)| > 0, \qquad \text{and} \qquad \inf_{||w||_W=1} \sup_{||v||_V=1} |a(v,w)| > 0.$$

Then,

    i) *λ is $\mathfrak{B}(V,W) \to \mathbb{R}$ continuous*

ii) *For each $a \in \mathfrak{B}_c(V, W)$ and $F \in W^*$ there exists a unique $u = u_{a,F} \in V$ such that $a(v, w) = F(w)$ for all $w \in W$. Furthermore, u depends continuously on both a and F. In particular:*

(7) $$\|u_{a,F} - u_{a',F'}\|_V \leq \frac{1}{\lambda(a)} \left(\|F - F'\|_* + \frac{1}{\lambda(a')}\||a - a'\|| \cdot \|F'\|_* \right)$$

for all $a, a' \in \mathfrak{B}_c(V, W)$ and $F, F' \in W^*$.

Proof. i) Let $a, a' \in \mathfrak{B}(V, W)$. For every $v \in V$ and $w \in W$ with $\|v\|_V = \|w\|_W = 1$ we have

$$a(v, w) = a'(v, w) + (a - a')(v, w) \leq |a'(v, w)| + \||a - a'\||.$$

Since the above holds for both w and $-w$, we actually have $|a(v, w)| \leq |a'(v, w)| + \||a - a'\||$, and thus $\lambda(a) \leq \lambda(a') + \||a - a'\||$. As the situation is symmetric in a and a', it follows that $|\lambda(a) - \lambda(a')| \leq \||a - a'\||$. In particular, λ is Lipschitz-continuous.

ii) Given $a \in \mathfrak{B}_c(V, W)$ and $F \in W^*$, the existence and uniqueness of $u_{a,F}$ follow from a Banach space version of the Lions-Lax-Milgram theorem (see Lemma 3.1. in [34]). Furthermore, one also has the stability estimate $\|u_{a,F}\|_V \leq (\lambda(a))^{-1}\|F\|_*$.

To get the inequality in (7), let $a, a' \in \mathfrak{B}(V, W)$, $F, F' \in W^*$ and $u := u_{a,F}$, $u' := u_{a',F'}$. Then $a(u, w) = F(w)$ and $a'(u', w) = F'(w)$ for all $w \in V$. We subtract these two identities to get

$$a(u, w) - a'(u', w) = F(w) - F'(w)$$
$$\implies a(u - u', w) = (F - F')(w) + (a' - a)(u', w).$$

It follows that, for all $w \in W$, one has $a(u - u', w) \leq \|F - F'\|_*\|w\|_W + \||a' - a\|| \cdot \|u'\|_V\|w\|_W$. By linearity, using both w and $-w$, we conclude that

$$|a(u - u', w)| \leq \|F - F'\|_*\|w\|_W + \||a - a'\|| \cdot \|u'\|_V\|w\|_W \qquad \forall w \in W.$$

In particular, passing at the supremum over $\|w\|_W = 1$ yields

$$\lambda(a)\|u - u'\|_V \leq \|F - F'\|_* + \||a - a'\|| \cdot \|u'\|_V.$$

Now, we may apply the stability estimate for $\|u'\|_V$ and divide by $\lambda(a)$ to get (7). Finally the latter, together with (i), shows that $u' \to u$ as soon as $a' \to a$ and $F' \to F$. $\square$

For the next Lemma, we consider the notation introduced in Section 2.2.1.

Lemma B.2. *Let $\Omega \subset \mathbb{R}^d$ be a bounded domain. Let $1 < q < +\infty$ and define the conjugate exponent $q' := (q-1)^{-1}q$. For each $\sigma \in \Sigma(\Omega)$, $b \in B(\Omega)$, $f \in W^{-1,q'}(\Omega)$ and $g \in W^{1/q',q}(\partial\Omega)$ let $u = u_{\sigma,b,f,g}$ be the unique solution to the following variational problem*

$u \in W^{1,q}(\Omega)$:

$$u_{|\partial\Omega} = g \quad \text{and} \quad \int_\Omega \boldsymbol{\sigma}\nabla u \cdot \nabla w + \int_\Omega (\boldsymbol{b}\cdot\nabla u)\, w = \int_\Omega fw \quad \forall w \in W_0^{1,q'}(\Omega).$$

Then, the solution map $(\boldsymbol{\sigma}, \boldsymbol{b}, f, g) \to u_{\boldsymbol{\sigma},\boldsymbol{b},f,g}$ is: (i) continuous, (ii) Lipschitz continuous on all compact subsets.

Proof. Before moving the actual proof, we shall recall that there exists a bounded linear operator $T : W^{1/q',q}(\partial\Omega) \to W^{1,q}(\Omega)$ for which $Tg_{|\partial\Omega} = g$, namely a right-inverse of the trace operator (see [48]). In particular, there exists a constant $\tilde{C} > 0$ such that $||Tg||_{W^{1,q}(\Omega)} \le \tilde{C}||g||_{W^{1/q',q}(\partial\Omega)}$.

For the sake of brevity, we let $V := W_0^{1,q}(\Omega)$, $W := W_0^{1,q'}$ and define W^* as the dual space of $W_0^{1,q'}$ endowed with the operator norm. As in Lemma B.1, we also let $\mathfrak{B}(V, W)$ be the collection of all bounded bilinear maps $V \times W \to \mathbb{R}$ equipped with the corresponding operator norm. Similarly, we define $\mathfrak{B}_c(V, W)$ to be the subset of coercive bounded bilinear maps. We introduce the following operators:

$$\begin{aligned} \mathcal{A} : \quad & L^\infty(\Omega, \mathbb{R}^{d\times d}) \times L^\infty(\Omega, \mathbb{R}^d) \longrightarrow \mathfrak{B}(V, W) \\ & (\boldsymbol{\sigma}, \boldsymbol{b}) \longrightarrow a_{\boldsymbol{\sigma},\boldsymbol{b}} := \mathcal{A}(\boldsymbol{\sigma}, \boldsymbol{b}), \end{aligned}$$

$$\begin{aligned} \mathcal{F} : \quad & L^\infty(\Omega, \mathbb{R}^{d\times d}) \times L^\infty(\Omega, \mathbb{R}^d) \times W^{-1,q'}(\Omega) \times W^{1/q',q}(\partial\Omega) \longrightarrow W^* \\ & (\boldsymbol{\sigma}, \boldsymbol{b}, f, g) \longrightarrow F_{\boldsymbol{\sigma},\boldsymbol{b},f,g} := \mathcal{F}(\boldsymbol{\sigma}, \boldsymbol{b}, f, g), \end{aligned}$$

where,

$$a_{\boldsymbol{\sigma},\boldsymbol{b}}(v, w) := \int_\Omega \boldsymbol{\sigma}\nabla v \cdot \nabla w + \int_\Omega (\boldsymbol{b}\cdot\nabla v)\, w,$$

$$F_{\boldsymbol{\sigma},\boldsymbol{b},f,g}(w) := \int_\Omega \boldsymbol{\sigma}\nabla Tg \cdot \nabla w + \int_\Omega (\boldsymbol{b}\cdot Tg)w + \int_\Omega fw.$$

We claim that:

1) The operator $\mathcal{A}$ is linear and continuous. Also, $a_{\boldsymbol{\sigma},\boldsymbol{b}} \in \mathfrak{B}_c(V, W)$ for all choices of $\boldsymbol{\sigma} \in \Sigma(\Omega)$ and $\boldsymbol{b} \in B(\Omega)$.

2) The operator $\mathcal{F}$ continuous. Also, it is Lipschitz continuous when restricted to any compact subset of its domain.

We shall now prove these claims. First of all, let $C > 0$ be the Poincàre constant for the domain Ω and the exponent q'. Then, it is straightforward to see that

$$\begin{aligned} |a_{\boldsymbol{\sigma},\boldsymbol{b}}(v, w)| \le\ & ||\boldsymbol{\sigma}||_{L^\infty(\Omega,\mathbb{R}^{d\times d})}||v||_{W_0^{1,q}(\Omega)}||w||_{W_0^{1,q'}(\Omega)} \\ & + C||\boldsymbol{b}||_{L^\infty(\Omega,\mathbb{R}^d)}||v||_{W_0^{1,q}(\Omega)}||w||_{W_0^{1,q'}(\Omega)}, \end{aligned}$$

for all $v \in V$ and $w \in W$. In particular, $\mathcal{A}$ is both linear and bounded, thus continuous. Let now $\boldsymbol{\sigma} \in \Sigma(\Omega)$, $\boldsymbol{b} \in B(\Omega)$ and define $\varepsilon = \varepsilon(\boldsymbol{\sigma}) > 0$ to be the

ellipticity constant of $\boldsymbol{\sigma}$. We notice that if $\varphi \in \mathcal{C}_0^\infty(\Omega)$, then φ is both an element of V and W. Also, integrating by parts yields

$$a_{\boldsymbol{\sigma},\boldsymbol{b}}(\varphi,\varphi) = \int_\Omega \boldsymbol{\sigma}\nabla\varphi \cdot \nabla\varphi + \int_\Omega \boldsymbol{b}\cdot(\varphi\nabla\varphi) \geq$$

$$\geq \varepsilon||\varphi||_{W_0^{1,q}(\Omega)}||\varphi||_{W_0^{1,q'}(\Omega)} + \int_\Omega \boldsymbol{b}\cdot\nabla\left(\frac{1}{2}\varphi^2\right) =$$

$$= \varepsilon||\varphi||_{W_0^{1,q}(\Omega)}||\varphi||_{W_0^{1,q'}(\Omega)} - \frac{1}{2}\int_\Omega \operatorname{div}(\boldsymbol{b})\varphi^2 =$$

$$= \varepsilon||\varphi||_{W_0^{1,q}(\Omega)}||\varphi||_{W_0^{1,q'}(\Omega)},$$

as $\boldsymbol{b}$ is divergence free. It follows that for each $\varphi \in \mathcal{C}_0^\infty(\Omega)$ with $\varphi \neq 0$

$$\sup_{\substack{\psi\in\mathcal{C}_0^\infty(\Omega)\\||\psi||_{W_0^{1,q'}(\Omega)}=1}} |a_{\boldsymbol{\sigma},\boldsymbol{b}}(\varphi,\psi)| \geq a_{\boldsymbol{\sigma},\boldsymbol{b}}\left(\varphi, ||\varphi||_{W_0^{1,q'}(\Omega)}^{-1}\varphi\right) \geq \varepsilon||\varphi||_{W_0^{1,q}(\Omega)}$$

and, similarly,

$$\sup_{\substack{\psi\in\mathcal{C}_0^\infty(\Omega)\\||\psi||_{W_0^{1,q}(\Omega)}=1}} |a_{\boldsymbol{\sigma},\boldsymbol{b}}(\psi,\varphi)| \geq a_{\boldsymbol{\sigma},\boldsymbol{b}}\left(||\varphi||_{W_0^{1,q}(\Omega)}^{-1}\varphi, \varphi\right) \geq \varepsilon||\varphi||_{W_0^{1,q'}(\Omega)}.$$

Since $a_{\boldsymbol{\sigma},\boldsymbol{b}}$ is continuous and $\mathcal{C}_0^\infty(\Omega)$ is both dense in V and W, by the above we conclude that $a_{\boldsymbol{\sigma},\boldsymbol{b}} \in \mathfrak{B}_c(V,W)$. This proves claim (1).

We now move to (2). For each $\boldsymbol{\sigma},\boldsymbol{b},f,g$ and $w \in W$ we have

$$|F_{\boldsymbol{\sigma},\boldsymbol{b},f,g}(w)| \leq$$

$$\leq ||\boldsymbol{\sigma}||_{L^\infty(\Omega,\mathbb{R}^{d\times d})}||Tg||_{W^{1,q}(\Omega)}||w||_{W_0^{1,q'}(\Omega)}$$

$$+ C||\boldsymbol{b}||_{L^\infty(\Omega,\mathbb{R}^d)}||Tg||_{W^{1,q}(\Omega)}||w||_{W_0^{1,q'}(\Omega)}$$

$$+ ||f||_{W^{-1,q'}(\Omega)}||w||_{W_0^{1,q'}(\Omega)}.$$

In particular, for all $w \in W$ with unitary norm,

$$|F_{\boldsymbol{\sigma},\boldsymbol{b},f,g}(w)| \leq \tilde{C}||g||_{W^{1/q',q}(\partial\Omega)}\left(||\boldsymbol{\sigma}||_{L^\infty(\Omega,\mathbb{R}^{d\times d})} + C||\boldsymbol{b}||_{L^\infty(\Omega,\mathbb{R}^d)}\right) + ||f||_{W^{-1,q'}(\Omega)}.$$

From here, arguing by linearity easily yields (2).

Finally, for each $\boldsymbol{\sigma} \in \Sigma(\Omega), \boldsymbol{b} \in B(\Omega), f \in W^{-1,q'}(\Omega), g \in W^{1/q',q}(\partial\Omega)$ let $\tilde{u}_{\boldsymbol{\sigma},\boldsymbol{b},f,g} \in V = W_0^{1,q}(\Omega)$ be the unique solution to the variational problem

$$a_{\boldsymbol{\sigma},\boldsymbol{b},f,g}(\tilde{u},w) = F_{\boldsymbol{\sigma},\boldsymbol{b},f,g}(w) \quad \forall w \in W.$$

At this regard, we notice that $W = W_0^{1,q'}(\Omega)$ is reflexive, in fact $1 < q < +\infty$ implies $1 < q' < +\infty$. Therefore, by Lemma B.1, we know that $\tilde{u}_{\boldsymbol{\sigma},\boldsymbol{b},f,g}$ exists unique and it depends continuously (by composition) on $(\boldsymbol{\sigma},\boldsymbol{b},f,g)$. Furthermore, as clear from inequality (7) in Lemma B.1, the correspondence $(\boldsymbol{\sigma},\boldsymbol{b},f,g) \to \tilde{u}_{\boldsymbol{\sigma},\boldsymbol{b},f,g}$ is Lipschitz continuous on every compact subset of the product space $\Sigma(\Omega) \times B(\Omega) \times$

$W^{-1,q'}(\Omega) \times W^{1/q',q}(\partial\Omega)$. This is easily deduced by the properties of $\mathcal{A}$ and $\mathcal{F}$ as well as by the fact that compactness is preserved under continuous transformations. Finally, we notice that

$$u_{\boldsymbol{\sigma},\boldsymbol{b},f,g} = \tilde{u}_{\boldsymbol{\sigma},\boldsymbol{b},f,g} + Tg.$$

The conclusion follows. □

Lemma B.3. *Consider the context and notation in Lemma B.2. If $g \equiv 0$ and $f > 0$ in the distributional sense, then $u > 0$ a.e. in Ω.*

Proof. This simply derives from maximum principles. Let $\eta \in \mathcal{C}_0^\infty(\Omega)$ be such that $\eta > 0$ everywhere in Ω. Let $w \in H_0^1(\Omega)$ be the solution to the following adjoint variational problem:

$$\int_\Omega \boldsymbol{\sigma}^T \nabla w \cdot \nabla v - \int_\Omega (\boldsymbol{b} \cdot \nabla w)v = \int_\Omega \eta v \quad \forall v \in \mathcal{C}_0^\infty(\Omega).$$

Within this regular case, the classical maximum principle states $w > \max w_{|\Omega} = 0$ in Ω, see e.g. Theorem 2 in [15]. Now we notice that $w \in W_0^{1,q'}(\Omega)$, as the PDE also admits a unique solution in that space. Thus, by density, we are allowed to consider u as test function for w and viceversa. Doing so and subtracting the equations for u and w yields

$$\int_\Omega \eta u = \int_\Omega fw,$$

since $\boldsymbol{\sigma}\nabla u \cdot \nabla w = \boldsymbol{\sigma}^T \nabla w \cdot \nabla u$ and the advective terms cancel out using the integration by parts formula (recall that $\boldsymbol{b}$ is divergence free while both u and w vanish on $\partial\Omega$). The above shows that $\int_\Omega \eta u > 0$, as the right hand side is positive by hypothesis. As η was arbitrary, we conclude that $u > 0$ a.e. in Ω. □

References

1. Adams, R.A., *Sobolev Spaces*, Pure and Applied Mathematics, vol. 65, Academic Press (1975).
2. Bachmayr, M., and Cohen, A., *Kolmogorov widths and low-rank approximations of parametric elliptic PDEs*, Journal: Math. Comp., vol. 86, pp. 701-724 (2017).
3. Babuška, I., Nobile, F., and Tempone, R., *A stochastic collocation method for elliptic partial differential equations with random input data*, SIAM Journal on Numerical Analysis, vol. 45(3), pp. 1005-1034 (2007).
4. Barrault, M., Maday, Y., Nguyen, N.C., and Patera, A.T., *An "empirical interpolation" method: Application to efficient reduced-basis discretization of partial differential equations*, C. R. Acad. Sci. Paris, Serie I 339, pp. 667–672 (2004).
5. Benyamini, Y., and Lindenstrauss, J., *Geometric Nonlinear Functional Analysis*, vol. 1, Amer. Math. Soc. Providence, RI (2000).
6. Bhattacharjee, S., and Matouš, K., *A nonlinear manifold-based reduced order model for multiscale analysis of heterogeneous hyperelastic materials*, Journal of Computational Physics, vol. 313, pp. 635-653 (2016).
7. Bhattacharya, K., Hosseini, B., Kovachki, N.B., and Stuart, A.M., *Model Reduction and Neural Networks for Parametric PDEs*, arXiv preprint, arXiv:2005.03180v1 [math.NA] (2020).
8. Binev, P., Cohen, A., Dahmen, W., DeVore, R., Petrova, G. and Wojtaszczyk P., *Convergence rates for greedy algorithms in reduced basis method*, SIAM J. Math. Anal., 43(3), pp. 1457–1472 (2011).

9. Bolcskei, H., Grohs, P., Kutyniok, G., and Petersen, P., *Optimal Approximation with Sparsely Connected Deep Neural Networks*, SIAM Journal on Mathematics of Data Science, vol. 1(1), pp. 8-45 (2019).

10. Borsuk, K., *Drei Sätze über die n-dimensionale euklidische Sphäre*, Fundamenta Mathematicae, 20: 177–190 (1933).

11. Bui-Thanh, T., Burstedde, C., Ghattas, O., Martin, J., Stadler, G., and Wilcox, L., *Extreme-scale UQ for Bayesian inverse problems governed by PDEs*, International Conference for High Performance Computing, Networking, Storage and Analysis, SC. 1-11. 10.1109/SC.2012.56 (2012).

12. Casas, E., *L2 estimates for the finite element method for the Dirichlet problem with singular data*, Numerische Mathematik, vol. 47(4), pp. 627-632 (1985).

13. Chen, T.Q., Rubanova, Y., Bettencourt, J., and Duvenaud, D., *Neural Ordinary Differential Equations*, arXiv preprint, arXiv:1806.07366 [cs.LG] (2018).

14. Chen, W., Wang, Q., Hesthaven, J.S. and Zhang, C., *Physics-informed machine learning for reduced-order modeling of nonlinear problems*, Journal of Computational Physics preprint (2020).

15. Chicco, M., *Principio di massimo forte per sottosoluzioni di equazioni ellittiche di tipo variazionale*, Bollettino dell'Unione Matematica Italiana, Serie 3, vol. 22(3), pp. 368-372 (1967).

16. Cohen, A., DeVore, R., Petrova, G. and Wojtaszczyk, P., *Optimal Stable Nonlinear Approximation*, arXiv preprint, arXiv:2009.09907 [math.NA] (2020).

17. Daubechies, I., DeVore, R., Foucart, S., Hanin, B. and Petrova, G., *Nonlinear Approximation and (Deep) ReLU Networks*, arXiv preprint, arXiv:1905.02199 [cs.LG] (2019).

18. DeVore, R., Howard, R., and Micchelli, C., *Optimal Nonlinear Approximation*, Manuscripta Mathematica, 63, pp. 469-478 (1989).

19. Druțu, C., and Kapovich, M., *Geometric Group Theory*, American Mathematical Society (2010).

20. Dugundji, J., *An extension of Tietze's theorem*, Pacific J. Math., vol. 1 (3), pp. 353-367 (1951).

21. Engelking, R., *Dimension Theory*, North-Holland Publishing Company, Amsterdam, Oxford, New York (1978).

22. Evans, L.C., *Partial Differential Equations, Second Edition*, American Mathematical Society (2010).

23. Fink, J.P., and Rheinboldt, W.C., *Solution manifolds and submanifolds of parametrized equations and their discretization errors*, Numerische Mathematik, vol. 45, pp. 323–343 (1984).

24. Fresca, S., Manzoni, A. and Dede, L., *A comprehensive deep learning-based approach to reduced order modeling of nonlinear time-dependent parametrized PDEs*, arXiv preprint, arXiv:2001.04001 [math.NA] (2020).

25. Fresca, S., and Manzoni, A., *POD-DL-ROM: enhancing deep learning-based reduced order models for nonlinear parametrized PDEs by proper orthogonal decomposition*, arXiv preprint, arXiv:2101.11845 [math.NA] (2021).

26. Geist, M., Petersen, P., Raslan, M., Schneider, R. and Kutyniok, G., *Numerical Solution of the Parametric Diffusion Equation by Deep Neural Networks*, arXiv preprint, arXiv:2004.12131 [math.NA] (2020).

27. Greif, C., and Urban, K., *Decay of the Kolmogorov N-width for wave problems*, Appl. Math. Lett., vol. 96, pp. 216-222 (2019).

28. Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., Liu, T., Wang, X., Wang, G., Cai, J., and Chen, T., *Recent advances in convolutional neural networks*, Pattern Recognition, vol. 77, pp. 354-377 (2018).

29. Gühring, I., Kutyniok, G., and Petersen, P., *Error bounds for approximations with deep ReLU neural networks in norms*, Analysis and Applications, vol. 18(5), pp. 803-859 (2020).

30. Guo, M., and Hesthaven, J.S., *Reduced order modeling for nonlinear structural analysis using Gaussian process regression*, Computer Methods in Applied Mechanics and Engineering, vol. 341, pp. 807-826 (2018).

31. He, K., Zhang, X., Ren, S., and Sun, J., *Delving deep into rectifiers: Surpassing human-level performance on imagenet classification*, Proceedings of the IEEE International Conference on Computer Vision (ICCV), pp. 1026–1034 (2015).

32. Hinton, G.E., and Salakhutdinov, R.R., *Reducing the dimensionality of data with neural networks*, Science, 313, pp. 504–507 (2006).

33. Hocking, J.G., and Young, G.S., *Topology*, Addison-Wesley Publishing Company, Inc. (1961).
34. Hoffmann, H., and Wald, A., *On parameter identification problems for elliptic boundary value problems in divergence form Part 1: An abstract framework*, arXiv pre-print, arXiv:2002.05009 [math.NA] (2020).
35. Jiang, J., Chen, Y., and Narayan, A., *Goal-Oriented Reduced Basis Methods-Accelerated Generalized Polynomial Chaos Algorithm*, SIAM/ASA J. Uncertainty Quantification, 4(1), pp. 1398–1420 (2016).
36. Kingma, D.P., and Ba, J., *Adam: A Method for Stochastic Optimization*, conference paper at the 3rd International Conference for Learning Representations, San Diego (2015).
37. Knezevic, D., and Patera, A.T., *A Certified Reduced Basis Method for the Fokker–Planck Equation of Dilute Polymeric Fluids: FENE Dumbbells in Extensional Flow*, SIAM Journal on Scientific Computing, 32.2: 793 (2010).
38. Kolmogorov, A.N., *Über die beste Annghergung von Funktionen einer gegebenen Funktionklasse*, Ann. Math., (2) 37, pp. 107-111 (1936).
39. Kutyniok, G., Petersen, P., Raslan, M. and Schneider, R., *A Theoretical Analysis of Deep Neural Networks and Parametric PDEs*, arXiv preprint, arXiv:1904.00377 [math.NA] (2019).
40. Laakmann, F. and Petersen, P., *Efficient Approximation of Solutions of Parametric Linear Transport Equations by ReLU DNNs*, arXiv preprint, arXiv:2001.11441 [math.NA] (2020).
41. Lassila, T., Manzoni, A., Quarteroni, A. and Rozza, G., *Generalized Reduced Basis Methods and n-Width Estimates for the Approximation of the Solution Manifold of Parametric PDEs*, Bollettino della Unione Matematica Italiana, Series IX, (2013).
42. Lee, K. and Carlberg, K.T., *Model reduction of dynamical systems on nonlinear manifolds using deep convolutional autoencoders*, Journal of Computational Physics, vol. 404 (2020).
43. Loshchilov, I., and Hutter, F., *Decoupled weight decay regularization*, conference paper at ICLR 2019, (2019).
44. Lu, L., Jin, P. and Karniadakis, G.E., *DeepONet: Learning nonlinear operators for identifying differential equations based on the universal approximation theorem of operators*, arXiv preprint, arXiv:1910.03193 [cs.LG] (2019).
45. Luo, Z. and Chen, G., *In Mathematics in Science and Engineering, Proper Orthogonal Decomposition Methods for Partial Differential Equations*, Academic Press (2019).
46. Mishra, S., and Molinaro, R., *Estimates on the generalization error of Physics Informed Neural Networks (PINNs) for approximating PDEs II: A class of inverse problems*, arXiv preprint, arXiv:2007.01138 [math.NA] (2020).
47. Mücke, N.T., Bohté, S.M., and Oosterlee, C.W., *Reduced Order Modeling for Parameterized Time-Dependent PDEs using Spatially and Memory Aware Deep Learning*, arXiv preprint, arXiv:2011.11327 [math.NA] (2020).
48. Nečas, J., *Les méthodes directes en théorie des équations elliptiques*, Paris: Masson et Cie, Éditeurs, Prague: Academia, Éditeurs. pp. 90–104 (1967).
49. Negri, F., Manzoni, A., and Amsallem, D., *Efficient model reduction of parametrized systems by matrix discrete empirical interpolation*, Journal of Computational Physics, vol. 303, pp. 431-454 (2015).
50. Ohlberger, M. and Rave, S., *Reduced basis methods: Success, limitations and future challenges*, Proceedings of the Conference Algoritmy, pp. 1–12 (2016).
51. Park, S., Yun, C., Lee, J., and Shin, J., *Minimum Width for Universal Approximation*, arXiv preprint, arXiv:2006.08859 [cs.LG] (2020).
52. Petersen, P., and Voigtlaender, F., *Optimal approximation of piecewise smooth functions using deep ReLU neural networks*, Neural Networks, vol. 108, pp. 296-330 (2018).
53. Quarteroni, A., Manzoni, A. and Negri F., *Reduced basis methods for partial differential equations - An introduction*, vol. 92 of Unitext. Springer, Cham (2016).
54. Quarteroni, A., and Valli, A., *Numerical Approximation of Partial Differential Equations*, Springer Series in Computational Mathematics, (2008).
55. Schwab, C. and Zech, J., *Deep learning in high dimension: Neural network expression rates for generalized polynomial chaos expansions in UQ*, Analysis and Applications, vol. 17(1), pp. 19-55 (2019).
56. Schmidhuber, J., *Deep learning in neural networks: An overview*, Neural Networks, vol. 61, pp. 85-117 (2015).

57. Shah, A.A., Xing, W.W., and Triantafyllidis, V., *Reduced-order modelling of parameter-dependent, linear and nonlinear dynamic partial differential equation models*, Proc. R. Soc. A.473: 20160809 (2017).
58. Shin, Y., Darbon, J., and Karniadakis, G.E., *On the convergence and generalization of physics informed neural networks*, arXiv preprint, arXiv:2004.01806v1 [math.NA] (2020).
59. Siegel, J.W., and Xu, J., *High-Order Approximation Rates for Neural Networks with ReLU^k Activation Functions*, arXiv preprint, arXiv:2012.07205 [math.NA] (2020).
60. Skopenkov, A., *Embedding and knotting of manifolds in Euclidean spaces*, Surveys in Contemporary Mathematics (London Mathematical Society Lecture Note Series, pp. 248-342), Cambridge: Cambridge University Press (2007).
61. Wang, Q., Hesthaven, J.S. and Ray, D., *Non-intrusive reduced order modeling of unsteady flows using artificial neural networks with application to a combustion problem*, Journal of Computational Physics, vol. 384, pp. 289-307 (2019).
62. Yarotski, D., *Error bounds for approximations with deep ReLU networks*, Neural Networks, vol. 94, pp. 103-114 (2017).
63. Zhang, H., Gao, X., Unterman, J., and Arodz, T., *Approximation Capabilities of Neural ODEs and Invertible Residual Networks*, arXiv preprint, arXiv:1907.12998v2 [cs.LG] (2020).
64. Zhu, Y. and Zabaras, N., *Bayesian deep convolutional encoder–decoder networks for surrogate modeling and uncertainty quantification*, Journal of Computational Physics, vol. 366, pp. 415-447 (2018).

MOX Laboratory, Department of Mathematics, Politecnico di Milano, Italy
Email address: `nicolarares.franco@polimi.it`

MOX Laboratory, Department of Mathematics, Politecnico di Milano, Italy
Email address: `andrea1.manzoni@polimi.it`

MOX Laboratory, Department of Mathematics, Politecnico di Milano, Italy
Email address: `paolo.zunino@polimi.it`

MOX Technical Reports, last issues

Dipartimento di Matematica
Politecnico di Milano, Via Bonardi 9 - 20133 Milano (Italy)

70/2021   Beirao da Veiga, L.; Canuto, C.; Nochetto, R.H.; Vacca, G.; Verani, M.
Adaptive VEM: Stabilization-Free A Posteriori Error Analysis

69/2021   Antonietti, P.F.; Caldana, M.; Dede', L.
Accelerating Algebraic Multigrid Methods via Artificial Neural Networks

65/2021   Mazzieri, I.; Muhr, M.; Stupazzini, M.; Wohlmuth, B.
Elasto-acoustic modelling and simulation for the seismic response of structures: The case of the Tahtali dam in the 2020 Izmir earthquake

66/2021   Antonietti, P.F.; Botti, M.; Mazzieri, I.
On mathematical and numerical modelling of multiphysics wave propagation with polygonal Discontinuous Galerkin methods

68/2021   Regazzoni, F.; Salvador, M.; Dede', L.; Quarteroni, A.
A machine learning method for real-time numerical simulations of cardiac electromechanics

67/2021   Salvador, M.; Regazzoni, F.; Pagani, S.; Dede', L.; Trayanova, N.; Quarteroni, A.
The role of mechano-electric feedbacks and hemodynamic coupling in scar-related ventricular tachycardia

64/2021   Clarotto, L; Allard, D.; Menafoglio, A.
A new class of alpha-transformations for the spatial analysis of Compositional Data

63/2021   Rosafalco, L.; Torzoni, M.; Manzoni, A.; Mariani, S.; Corigliano, A.
Online structural health monitoring by model order reduction and deep learning algorithms

62/2021   Lupo Pasini, M.; Burcul, M.; Reeve, S.; Eisenbach, M.; Perotto, S.
Fast and accurate predictions of total energy for solid solution alloys with graph convolutional neural networks

60/2021   Rosafalco, L.; Manzoni, A.; Mariani, S.; Corigliano, A.
Fully convolutional networks for structural health monitoring through multivariate time series classification