

DPG LOSS FUNCTIONS FOR LEARNING PARAMETER-TO-SOLUTION MAPS BY NEURAL NETWORKS

PABLO CORTÉS CASTILLO, WOLFGANG DAHMEN, AND JAY GOPALAKRISHNAN

ABSTRACT. We develop, analyze, and experimentally explore residual-based loss functions for machine learning of parameter-to-solution maps in the context of parameter-dependent families of partial differential equations (PDEs). Our primary concern is on rigorous accuracy certification to enhance prediction capability of resulting deep neural network reduced models. This is achieved by the use of variationally correct loss functions. Through one specific example of an elliptic PDE, details for establishing the variational correctness of a loss function from an ultraweak Discontinuous Petrov Galerkin (DPG) discretization are worked out. Despite the focus on the example, the proposed concepts apply to a much wider scope of problems, namely problems for which stable DPG formulations are available. The issue of high-contrast diffusion fields and ensuing difficulties with degrading ellipticity are discussed. Both numerical results and theoretical arguments illustrate that for high-contrast diffusion parameters the proposed DPG loss functions deliver much more robust performance than simpler least-squares losses.

1. INTRODUCTION

The need to recover information on a physical state of interest from a limited amount of observational data is ubiquitous in scientific and technological applications. To make such tasks feasible, it is crucial to leverage the physical laws that the states obey. However, these are often only partially known. A common way to deal with such uncertainties is to formulate the mathematical model as a system of partial differential equations (PDEs) depending on unknown model data, (such as coefficient fields, initial or boundary conditions, source terms, or constitutive laws), collected into *parameters* α from finite- or infinite-dimensional spaces. We have in mind PDE solutions $u(x, \alpha)$ depending on a spatial variable x in a low-dimensional domain $\Omega \subset \mathbb{R}^d$ and a parameter variable α in a possibly high-dimensional domain \mathcal{Q} . For each parameter instance $\alpha \in \mathcal{Q}$, the solution $u(x, \alpha)$, also written as $u(\alpha)$ or simply u , is the unique solution of

$$\mathcal{R}(u, \alpha) = 0, \tag{1}$$

for some (generally nonlinear) PDE \mathcal{R} written in residual form. The structure and range of \mathcal{Q} determine the “design space” or “solution manifold” \mathcal{M} comprised of the states $u(\alpha)$ that are obtained as solutions of (1) when α traverses \mathcal{Q} . Equation (1) implicitly defines a *parameter-to-solution* map $\mathcal{F} : \mathcal{Q} \rightarrow \mathcal{M}$. Generally vector spaces in which u is to be found may change from point to point on the manifold. At each parameter instance $\alpha \in \mathcal{Q}$, the problem (1) for u at the corresponding point on the manifold is referred to as a “fiber problem.” Exploration of \mathcal{M} or solving inverse tasks like state-estimation or parameter-estimation when using \mathcal{M} as a prior, require a large—sometimes prohibitive—number of high-fidelity forward simulations. This calls for generating reduced models providing efficiently computable surrogates for the parameter-to-solution map \mathcal{F} . The underlying rationale is that evaluating such a surrogate at any given parameter instance is much more efficient than computing a corresponding high-fidelity approximate solution, see e.g. [7, 18, 27]. Generating a surrogate typically takes

a substantial computational effort, which though has to be performed only once, and is considered an “offline-cost”. It amortizes quickly when many parameter queries are needed.

Generating such surrogates is a central instance of *Operator Learning* which has lately been attracting considerable attention, see e.g. [8, 29, 30, 31]. In full generality, it aims at approximating mappings between Banach space, hence permitting infinite-dimensional inputs and outputs. In this latter regard, we are more modest in this paper and focus on a scenario where \mathcal{Q} is a finite-dimensional diffusion field in a second-order elliptic boundary value problem. However, we address different challenges which are, in our opinion, of paramount importance when striving for *prediction capability*, namely challenges in controlling prediction error in model-compliant norms even in singularly perturbed regimes of parameters. The second-order elliptic boundary value problem has numerous applications, a typical one being where u represents the pressure in a porous medium flow and \mathcal{R} represents the simplest version of Darcy’s model. In the construction of surrogates, we use finite elements to discretize the low-dimensional Ω but use deep neural networks (NN) to discretize \mathcal{F} from the high-dimensional \mathcal{Q} . This means that our solution predictions take the form

$$u_\theta(x, \alpha) = \sum_k v_{k,\theta}(\alpha) \psi_k(x) \quad (2)$$

where $\psi_k(x)$ are standard finite element basis functions and the coefficients $v_{k,\theta}(\alpha)$ in the basis expansion are provided by the NN whose settings are determined by a collection θ . While many works discretize spatial dependence also using NN (exceptions include [3, 24, 26]) we opt for (2) to leverage existing finite element techniques and avoid issues in imposing essential boundary conditions on unstructured spatial domains.

To put our focus on prediction into perspective, most operator learning approaches are purely data-driven: one first computes a sufficiently large number of high-fidelity solutions that are then used as training samples for performing regression over suitable hypothesis classes for operators—see, e.g., [8, 24, 26, 30, 29, 31]. The accuracy can then be assessed via estimating the “generalization error” incurred. To mitigate the potentially high cost of computing the high-fidelity solutions, other approaches use *residual-based loss functions* and avoid solving large systems during training. The best known version is PINN (Physics-Informed Neural Network) [22, 28, 34] which employs a Monte-Carlo loss

$$\theta^* \in \operatorname{argmin}_{\theta \in \Theta} \frac{1}{\#\widehat{\mathcal{Q}} \#\widehat{\Omega}} \sum_{\alpha \in \widehat{\mathcal{Q}}, x \in \widehat{\Omega}} |\mathcal{R}(u_\theta(x, \alpha), \alpha)|^2, \quad (3)$$

to learn an approximation u_{θ^*} from a hypothesis class determined by a finite budget Θ of trainable weights θ . Here $\widehat{\Omega}$ is a finite set of spatial samples (such as quadrature points in Ω as well as on $\partial\Omega$) and $\widehat{\mathcal{Q}} \subset \mathcal{Q}$ is a finite collection of model parameters. It is not possible to summarize the burgeoning PINN literature here, but we note that error control and error analysis have received the attention of others: see [1, 10, 36, 38, 40] and references therein. Most of such literature is however concerned with parameter-independent problems and aim to solve just a single fixed PDE, while our focus is on the parameter-to-solution map. While (3) offers great computational convenience (as it only requires evaluation of residuals \mathcal{R} at spatial and parametric samples), it has the following serious drawbacks. Depending on the type of PDE, the residual \mathcal{R} need not be a function in $L_2(\Omega)$ (a problem noted in several works [3, 38, 40]). Moreover, when pursuing an ansatz such as (3), the only possible accuracy

assessment would be based on the loss itself, and a loss of the form (3) need neither be an efficient nor a reliable indicator of the generalization errors.

In this paper, we also opt for employing residual-based loss functions but insist on these losses being *variationally correct*. A precise definition for our situation is provided in Definition 2.4, but to convey the idea right away, when a loss function is variationally correct, its size, at any stage of the optimization process, provides, up to uniform constant factors, a lower and upper bound for the approximation error with respect to the respective model compliant norm (see [3]). As explained later in detail, this is intimately related to a *stable variational formulation* of the PDE (see also [39]). Again, related ideas in prior works have been confined to solving a single PDE in low spatial dimensions because corresponding residual loss functions are only then practically feasible [5, 6, 15, 23, 32].

Even though the presented concepts apply to a much wider scope of PDE models, namely whenever we have a stable Discontinuous Petrov Galerkin (DPG) formulation [20, 21] at our disposal, in favor of quantitative accuracy studies, we focus here on the homogeneous Dirichlet problem for second-order elliptic equations with *parameter-dependent* diffusion coefficients. Even when the range \mathcal{Q} of parameters guarantees uniform ellipticity, PINN in the form (3) need not be variationally correct. As explained in Section 2 (see also [3]), one key step to realizing variational correctness is to reformulate the PDE first as a well-posed *first-order system* for the flux and the solution, i.e., for $d + 1$ scalar solution components. For strictly positive and bounded diffusion coefficients, there exist actually several distinct stable variational formulations for the first-order system, differing by the choice of trial and test space. In this paper, we discuss two such formulations that mark, in some sense, extreme cases. In the first one, a closed subspace \mathcal{X} of a graph space, determined by homogeneous boundary conditions, serves as the trial space. It can then be shown that the induced operator maps the trial space *onto* a test space \mathcal{Y} which equals the $(d + 1)$ -fold Cartesian product of $L_2(\Omega)$, thus giving a well-posed variational formulation in $\mathcal{X} \times \mathcal{Y}$. This is the *First-Order System Least Squares* (FOSLS) formulation introduced in [12]. Its residual is well-defined in L_2 and the L_2 -residual is uniformly proportional to the corresponding error measured in the graph norm. Incidentally, although PINN is not variationally correct for the original second-order formulation, it is for the FOSLS formulation.

How sharp the L_2 -loss is as a lower and upper bound for the graph norm errors, depends, however, on the size-range of the diffusion coefficients. This raises the main issue addressed in this paper, namely, permitting arbitrarily large size variations in the diffusion coefficient. This is sometimes referred to as *high-contrast* problems. To indicate the issue, it is not hard to see that for piecewise constant diffusion (and constant right hand side) the solution tends to zero in regions where the diffusion tends to infinity while the gradient tends to infinity where the diffusion tends to zero—see [16] and further literature cited there. The approach in [16] developed particular reduced bases incorporating these limit features. In contrast, here we address this issue by considering the so-called *ultraweak* formulation where the trial space is just the $(d + 1)$ -fold Cartesian product of $L_2(\Omega)$, while the test space \mathcal{Y} is now a boundary condition-induced closed subspace of the graph space of the *dual* operator (see Example 2.3). This results in a well-posed formulation in which parameter robustness issues are manifested in a hard-to-compute dual norm of \mathcal{Y} . To overcome these issues, we tap into DPG methodology [20, 21]. In quantitative terms this approach is slightly more expensive than the FOSLS formulation. However, we will show that it is much more robust for “degenerating” parameters α than FOSLS. While we have a complete theory establishing

the variational correctness of DPG if the parameter α is confined to a bounded range, we do not have a full theory establishing the observed robustness of the practically implemented DPG loss. We do show however, that a closely related ideal DPG loss function is variationally correct with explicit constants showing α -robustness.

The layout of the paper is as follows. Section 2 is devoted to the problem description. Starting with three simple examples in § 2.1, we outline the issues that motivate the remainder of the developments. In § 2.5, we define concisely the central notion of variationally correct loss functions based on a functional setting that properly respects the mapping properties of parameter-to-solution maps. In Section 3 we introduce the FOSLS loss function and establish its variational correctness. In essence, this follows [3], except that we highlight the dependence of the constants involved on the parameter α . An alternate DPG formulation is then introduced and analyzed in Section 4 along analogous lines. The performance of both formulations with regard to prediction accuracy is studied in Section 5. These findings indicate some advantages of the DPG formulation when the contrast in the diffusion coefficient increases and the conditioning of the FOSLS operator exhibits a stronger degradation. This motivates a more in-depth robustness study in Section 6.

2. A PARAMETRIC PDE MODEL

2.1. The model problem and first-order reformulations. Let Ω be a bounded Lipschitz domain in \mathbb{R}^d , let $\alpha : \Omega \rightarrow \mathbb{R}$ be a positive function, and let $a(x) = \alpha(x)^{-1}$. Consider the stationary diffusion equation

$$\begin{aligned} -\operatorname{div}(a(x)\nabla u) &= f & \text{in } \Omega, \\ u &= 0 & \text{on } \partial\Omega. \end{aligned} \tag{4}$$

Here we are not interested in solving (4) for a fixed $a(x)$, but for a parameterized range \mathcal{Q} of diffusion coefficients which will be specified later below. When a varies, or equivalently α varies, so does the solution u , which we shall write as $u(\alpha)$ or $u(x, \alpha)$ when we need to highlight the α -dependence. As mentioned in Section 1, a central objective in operator learning is to construct surrogates for the mapping $\alpha \mapsto u(\alpha)$ so as to facilitate an efficient exploration of the corresponding design space \mathcal{M} of viable states $u(\alpha)$, $\alpha \in \mathcal{Q}$.

The boundary value problem (4) admits many well-posed variational formulations (three of which are given below). Suppose \mathbf{u} is a group variable which contains u as one of the components and suppose that there is a well-posed variational formulation that uniquely fixes $\mathbf{u} \in \mathcal{X}$ as the solution of a weak formulation

$$b_\alpha(\mathbf{u}, \mathbf{v}) = \ell(\mathbf{v}) \quad \text{for all } \mathbf{v} \in \mathcal{Y}, \tag{5}$$

where $b_\alpha : \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}$ is a continuous bilinear form on the product of some Hilbert spaces \mathcal{X} and \mathcal{Y} and $\ell \in \mathcal{Y}'$ is the functional capturing the source f . Let the continuous operator generated by b_α be denoted by $B_\alpha : \mathcal{X} \rightarrow \mathcal{Y}'$, i.e., $(B_\alpha \mathbf{w})(\mathbf{v}) = b_\alpha(\mathbf{w}, \mathbf{v})$ for all $\mathbf{w} \in \mathcal{X}$ and $\mathbf{v} \in \mathcal{Y}$. The wellposedness of (5), by Babuška's theorem, implies that

$$\|\mathbf{w}\|_{b_\alpha} = \sup_{0 \neq \mathbf{v} \in \mathcal{Y}} \frac{b_\alpha(\mathbf{w}, \mathbf{v})}{\|\mathbf{v}\|_{\mathcal{Y}}} \tag{6}$$

is an equivalent norm on \mathcal{X} . Suppose we are given a candidate approximation $\mathbf{w} \in \mathcal{X}$ to the exact solution \mathbf{u} of (5). The error in such an approximation satisfies

$$\|\mathbf{u} - \mathbf{w}\|_{b_\alpha} = \sup_{0 \neq \mathbf{v} \in \mathcal{Y}} \frac{b_\alpha(\mathbf{u} - \mathbf{w}, \mathbf{v})}{\|\mathbf{v}\|_{\mathcal{Y}}} = \sup_{0 \neq \mathbf{v} \in \mathcal{Y}} \frac{\ell(\mathbf{v}) - b_\alpha(\mathbf{w}, \mathbf{v})}{\|\mathbf{v}\|_{\mathcal{Y}}} = \|\ell - B_\alpha \mathbf{w}\|_{\mathcal{Y}'}, \quad (7)$$

which we refer to as the *error-residual equation*. We may now set \mathcal{R} in (1) by $\mathcal{R}(\mathbf{w}, \alpha) = \|\ell - B_\alpha \mathbf{w}\|_{\mathcal{Y}'}$ and observe that the solution \mathbf{u} indeed satisfies (1) due to (7).

The central issues that occupy us in this paper can be described in simple terms using (7). We would like to control the error $\mathbf{u} - \mathbf{w}$ in some desired α -independent norm as robustly as possible while α varies in \mathcal{Q} . To do so, we may use the residual $\|\ell - B_\alpha \mathbf{w}\|_{\mathcal{Y}'}$ in (7) containing only known quantities ℓ and \mathbf{w} , provided we know (i) how to compute (the residual in) the dual norm $\|\cdot\|_{\mathcal{Y}'}$, and (ii) how to obtain α -robust equivalences between the desired α -independent norm and $\|\cdot\|_{b_\alpha}$ (which involves dealing with degenerating ellipticity when α attains very large or very small values while ranging over \mathcal{Q}). The following examples show to what extent these *two issues can be influenced by the choice of the variational formulation*.

Example 2.1 (Primal formulation). The canonical weak formulation of (4) seeks the solution $\mathbf{u} = u$ in the Sobolev space $\dot{H}^1 = \{v \in L_2(\Omega) : \partial_i v \in L_2(\Omega), \text{ for every } i = 1, \dots, d \text{ and } v|_{\partial\Omega} = 0\}$, setting

$$\mathcal{X} = \mathcal{Y} = \dot{H}^1, \quad b_\alpha(u, v) = \int_{\Omega} \alpha^{-1} \nabla u \cdot \nabla v \, dx, \quad \ell(v) = \int_{\Omega} f v \, dx$$

and solving (5). Issue (i) manifests as the difficulty to compute the $H^{-1} := (\dot{H}^1)'$ norm. Issue (ii) is evident from the strong α -dependence of the $\|\cdot\|_{b_\alpha}$ -norm in this case, e.g.,

$$\|\nabla v\|_{L_2}^2 \min_{x \in \Omega} \alpha^{-1} \leq \|v\|_{b_\alpha}^2 \leq \|\nabla v\|_{L_2}^2 \max_{x \in \Omega} \alpha^{-1}$$

shows that attempts to control the error in an α -independent norm, such as $\|u\|_{\mathcal{X}} = \|u\|_{\mathcal{Y}} = \|\nabla u\|_{L_2}$, or the standard H^1 -norm, are plagued by difficulties. Here and throughout, $\|\cdot\|_{L_2}$ denotes the norm of L_2 or its Cartesian products.

Example 2.2 (FOSLS formulation). Given a square-integrable $f \in L_2(\Omega)$, (4) can equivalently be written in the first-order form using α as

$$\alpha(x) q + \nabla u = 0 \quad \text{in } \Omega, \quad (8a)$$

$$\operatorname{div} q = f \quad \text{in } \Omega, \quad (8b)$$

$$u = 0 \quad \text{on } \partial\Omega. \quad (8c)$$

It is useful to introduce a notation for the entire first-order operator in (8).

$$A_\alpha(q, u) = (\alpha q + \nabla u, \operatorname{div} q). \quad (9)$$

Then (8) takes the form $A_\alpha \mathbf{u} = (0, f)$ in Ω for the group variable $\mathbf{u} = (q, u)$.

Since the weak formulation of (4) allows for $f \in H^{-1}(\Omega)$ the above assumption requiring f in $L_2(\Omega)$ seems to entail a restriction. However, recall that every $f \in H^{-1}(\Omega)$ can be written as $f = f_\circ + \operatorname{div} g$ where $f_\circ \in L_2(\Omega)$ and $g \in L_2(\Omega)^d$. Then the first-order system takes the form $A_\alpha(q, u) = (g, f_\circ)$, thus covering the same range of source terms as (4). It is only for the sake of convenience that we continue to work with $f_\circ = f, g = 0$.

The choice of a pair of trial and test spaces \mathcal{X}, \mathcal{Y} , that gives rise to a stable weak formulation of (9), is now less canonical. Equation (8a) defines the “flux variable” q in terms of derivatives of the “primal variable” u . The former lies in $H(\operatorname{div})$, the space of vector fields whose

components and divergence are in $L_2(\Omega)$. The latter lies in \dot{H}^1 in accordance with the weak formulation of (4). Thus, we view A_α as a mapping

$$A_\alpha : H(\operatorname{div}) \times \dot{H}^1 \longmapsto [L_2(\Omega)]^d \times L_2(\Omega), \quad (10)$$

and construct a weak formulation of $A_\alpha \mathbf{u} = (0, f)$ by multiplying both sides by $\mathbf{v} = (r, w)$ in $[L_2(\Omega)]^d \times L_2(\Omega)$ and integrating over Ω to arrive at the setting (5) with

$$\begin{aligned} \mathcal{X} &= H(\operatorname{div}) \times \dot{H}^1, & \mathcal{Y} &= L_2(\Omega) \times [L_2(\Omega)]^d, \\ b_\alpha(\mathbf{u}, \mathbf{v}) &= \int_\Omega A_\alpha \mathbf{u} \cdot \mathbf{v} \, dx, & \ell(\mathbf{v}) &= \int_\Omega F \cdot \mathbf{v} \, dx, \end{aligned} \quad (11)$$

where $F = (0, f)$. Equation (5) in this case can equivalently be written in component form, with $\mathbf{u} = (q, u)$ and $\mathbf{v} = (r, w)$ as

$$\int_\Omega (\alpha q + \nabla u) \cdot r + (\operatorname{div} q - f)w \, dx = 0, \quad w \in L_2(\Omega), r \in [L_2(\Omega)]^d. \quad (12)$$

This is the weak formulation underlying the FOSLS method. It is well known that (10) is a continuous bijection and that (12) is uniquely solvable for every α that is uniformly positive and bounded in $\bar{\Omega}$. Moreover, it can be shown to yield the same solution as the canonical weak formulation of (4) in the sense that u is the weak solution of (4) if and only if $q = -a(x)\nabla u$ and u solves (12)—see [3, 12].

The error-residual equation (7) now takes the form

$$\|\mathbf{u} - \mathbf{w}\|_{b_\alpha} = \|F - A_\alpha \mathbf{w}\|_{L_2}, \quad \mathbf{w} \in H(\operatorname{div}) \times \dot{H}^1. \quad (13)$$

The interest in the FOSLS formulation (11) is now evident: all difficulties with the issue (i) disappears since the norm of $\mathcal{Y} = \mathcal{Y}'$ is just the L_2 -norm which is simple to evaluate. In regards to issue (ii), note that (11) implies $\|\mathbf{w}\|_{b_\alpha} = \|A_\alpha \mathbf{w}\|_{L_2}$. Unfortunately, this is an α -dependent norm, which cannot be easily controlled by α -independent norms with α -robust equivalence constants—see (32) below and Section 3. We will see practical manifestations of this difficulty in our experimental studies in Section 4.

Example 2.3 (Ultraweak formulation). This formulation is also based on the first-order reformulation (8). Integrating (12) by parts,

$$\int_\Omega \alpha q \cdot r - u \operatorname{div} r - q \nabla w - f w \, dx + \int_{\partial\Omega} (n \cdot r)w + (n \cdot q)w \, ds = 0.$$

Taking $w \in \dot{H}^1$ and $r \in H(\operatorname{div})$ as test functions, we arrive at the problem of finding $(q, u) \in [L_2(\Omega)]^d \times L_2(\Omega)$ such that

$$\int_\Omega q \cdot (\alpha r - \nabla w) - u \operatorname{div} r - f w \, dx = 0. \quad (14)$$

What were previously essential boundary conditions are now natural ones. Introducing the adjoint operator

$$A_\alpha^*(q, u) = (\alpha q - \nabla u, -\operatorname{div} q), \quad (15)$$

we may rewrite (14) as (5) with the following settings:

$$\begin{aligned} \mathcal{X} &= [L_2(\Omega)]^d \times L_2(\Omega), & \mathcal{Y} &= H(\operatorname{div}) \times \dot{H}^1, \\ b_\alpha(\mathbf{u}, \mathbf{v}) &= \int_\Omega \mathbf{u} \cdot A_\alpha^* \mathbf{v} \, dx, & \ell(\mathbf{v}) &= \int_\Omega F \cdot \mathbf{v} \, dx, \end{aligned} \quad (16)$$

with $\mathbf{u} = (q, u)$ and $\mathbf{v} = (r, w)$. This is often called an “ultraweak formulation” since all the solution components need only be regular enough to be in $L_2(\Omega)$. It is well posed [13]. If the ultraweak solution belongs to $H(\text{div}) \times \dot{H}^1$ then it agrees with the solution to (12).

Consider the error-residual equation (7) for the formulation (16). It is well known that $A_\alpha^* : H(\text{div}) \times \dot{H}^1 \rightarrow [L_2(\Omega)]^d \times L_2(\Omega)$ is a continuous bijection (just as (10) is). Hence \mathcal{Y} may equivalently be normed by

$$\|\mathbf{v}\|_{\mathcal{Y}_\alpha} := \|A_\alpha^* \mathbf{v}\|_{L_2}, \quad \mathbf{v} \in H(\text{div}) \times \dot{H}^1. \quad (17)$$

Such norms have been used under the name “optimal test norms” previously [17, 21, 41]. Let \mathcal{Y}_α denote $H(\text{div}) \times \dot{H}^1$ endowed with the above norm $\|\cdot\|_{\mathcal{Y}_\alpha}$. Revising \mathcal{Y} in (16) to \mathcal{Y}_α , the norm in (6) becomes

$$\|\mathbf{w}\|_{b_\alpha} = \sup_{0 \neq \mathbf{v} \in \mathcal{Y}_\alpha} \frac{b_\alpha(\mathbf{w}, \mathbf{v})}{\|A_\alpha^* \mathbf{v}\|_{L_2}} = \|\mathbf{w}\|_{L_2},$$

so (7) implies

$$\|\mathbf{u} - \mathbf{v}\|_{L_2} = \|\ell - B_\alpha \mathbf{v}\|_{\mathcal{Y}_\alpha}. \quad (18)$$

An attractive feature of (18) is that it has no α -dependent constants and the norm on the left hand side is the L_2 -norm, which is completely independent of α . Thus the previously mentioned issue (ii) disappears. However, the norm on the right hand side of (18) is a non-trivial dual norm whose numerical evaluation is an issue, i.e., issue (i) remains. This can be fixed to some extent using the so-called ultraweak DPG method, as we shall see in Section 6.

2.2. Parameter-to-solution map. Since handling arbitrary variations in the function parameter α is beyond the scope of this paper, we first restrict ourselves to a finite-dimensional parameter set for α by subdividing Ω into finitely many subdomains Ω_i , $i = 1, \dots, m_\alpha$, and requiring α to be a constant in each Ω_i . Namely, using $\chi_i(x)$ to denote the indicator function of Ω_i , we define the parameter domain to be

$$\mathcal{Q} = \left\{ \sum_{i=1}^{m_\alpha} \alpha_i \chi_i(x) : \alpha_i \geq 0 \right\}. \quad (19)$$

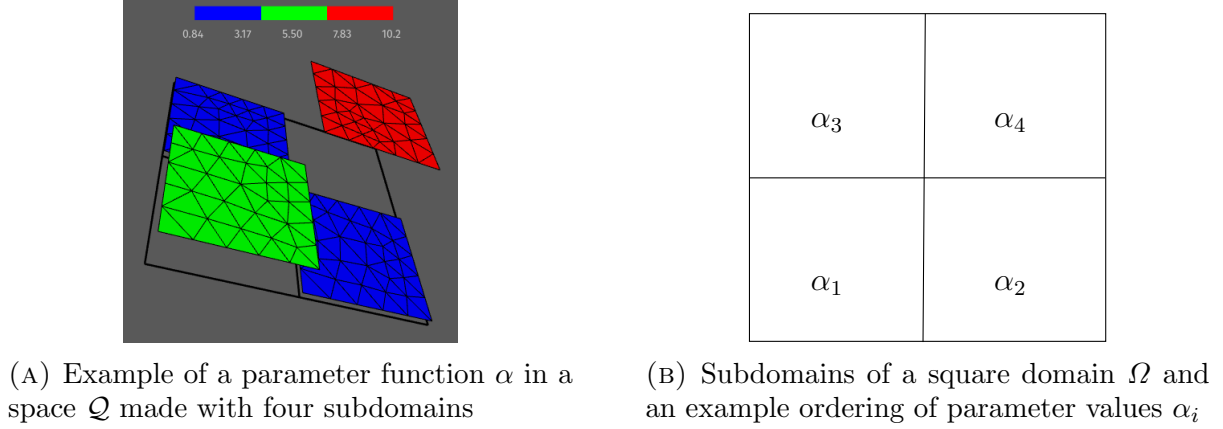
We are interested in approximating the parameter-to-solution map \mathcal{F} defined by

$$\alpha \xrightarrow{\mathcal{F}} \text{the } (q, u) \text{ solving (8)}, \quad (20)$$

i.e., $\mathcal{F}(\alpha) = (q(x, \alpha), u(x, \alpha))$ is a function of x and α defined on the spatio-parametric domain $\Omega \times \mathcal{Q}$. In a learning context such approximations result from minimizing appropriate loss functions. In constructing such approximations, even though \mathcal{Q} is a subset of a vector space of dimension $\dim \mathcal{Q} = m_\alpha$, the vector space structure is not used. In fact, in many practical problems, the parameter domain is a bounded subset of \mathcal{Q} . We keep \mathcal{Q} unbounded in our studies here to make the learning task more challenging.

The approximation to \mathcal{F} depends on the particular weak formulation of (8) we choose to use. Without specifying the formulation at the moment, suppose that we are given a basis ψ_k , $k = 1, \dots, m_h$, for a finite element space, built using a mesh of maximal element diameter h and that an approximation of $\mathcal{F}(\alpha) = (q(x, \alpha), u(x, \alpha))$ is found, for each given α in \mathcal{Q} , in their span

$$\mathcal{X}_h = \left\{ \sum_{k=1}^{m_h} v_k \psi_k(x) : v_k \in \mathbb{R} \right\}$$

FIGURE 1. Illustration of the functions in parameter space \mathcal{Q} .

by some method. Then a computable approximation of the exact \mathcal{F} in (20), which we denote by \mathcal{F}_h , can be implemented as a map between Euclidean vector spaces, $\mathbb{F} : \mathbb{R}^{m_\alpha} \rightarrow \mathbb{R}^{m_h}$. The mapping of functions \mathcal{F}_h is generated from the mapping \mathbb{F} of vectors by

$$\mathcal{F}_h \left(\sum_{i=1}^{m_\alpha} \alpha_i \chi_i(x) \right) = \sum_{k=1}^{m_h} [\mathbb{F}(\mathfrak{w})]_k \psi_k(x), \quad x \in \Omega. \quad (21)$$

where \mathfrak{w} is the vector of coefficients of $\alpha \in \mathcal{Q}$, i.e., $\mathfrak{w} = (\alpha_1, \alpha_2, \dots)$. We shall construct such approximate mappings \mathcal{F}_h using neural networks. Two different choices of the finite element space \mathcal{X}_h (and accompanying loss functions for training the neural networks) are discussed in Sections 3 and 4.

2.3. Neural network (NN) architecture. To construct the map \mathbb{F} using an NN, we use an architecture proposed in [3], described next. Let ρ denote a leaky ReLU function defined component wise by

$$\rho : \mathbb{R}^N \rightarrow \mathbb{R}^N, \quad \rho(y)_i = \max(y_i, 10^{-3}y_i), \quad y \in \mathbb{R}^N,$$

for some integer N . Next, we select another integer $r \ll N$, to promote sparsity as suggested in [3] and consider matrices $A_{\text{out}} \in \mathbb{R}^{m_h \times N}$, $A_m \in \mathbb{R}^{N \times r}$, $W_m \in \mathbb{R}^{r \times N}$, $A_{\text{in}} \in \mathbb{R}^{N \times m_\alpha}$, and vectors $b_m \in \mathbb{R}^r$, $b_{\text{out}} \in \mathbb{R}^{m_h}$, $b_{\text{in}} \in \mathbb{R}^{m_\alpha}$. For $m = 1, \dots, l$, let

$$\begin{aligned} \Phi_m : \mathbb{R}^N &\rightarrow \mathbb{R}^N, & \Phi_m(z) &= z + A_m \rho(W_m z + b_m), \\ L_{\text{out}} : \mathbb{R}^N &\rightarrow \mathbb{R}^{m_h}, & L_{\text{out}}(z) &= A_{\text{out}} z + b_{\text{out}}, \\ L_{\text{in}} : \mathbb{R}^{m_\alpha} &\rightarrow \mathbb{R}^N, & L_{\text{in}}(p) &= A_{\text{in}} p + b_{\text{in}}, \end{aligned}$$

for any $z \in \mathbb{R}^N$ and $p \in \mathbb{R}^{m_\alpha}$. Then we approximate \mathbb{F} by expressions of the form

$$\mathbb{F}_\theta : \mathbb{R}^{m_\alpha} \rightarrow \mathbb{R}^{m_h}, \quad \mathbb{F}_\theta = L_{\text{out}} \circ \Phi_l \circ \dots \circ \Phi_1 \circ L_{\text{in}} \quad (22)$$

(as illustrated in Figure 2) where the subscript θ represents all the “trainable” degrees of freedom (weights and biases) above, i.e., $\theta = \{A_{\text{in}}, b_{\text{in}}, A_m, W_m, b_m, b_{\text{out}}, A_{\text{out}} : m \in 1, \dots, l\}$. Let Θ denote the set of all such values that θ can take. An appropriate value of θ in Θ is found by a training process, as commonly done in the context of all neural networks. Let $\mathcal{F}_{h,\theta} : \mathcal{Q} \rightarrow \mathcal{X}_h$ denote the mapping of functions obtained by replacing \mathbb{F} by \mathbb{F}_θ in (21).

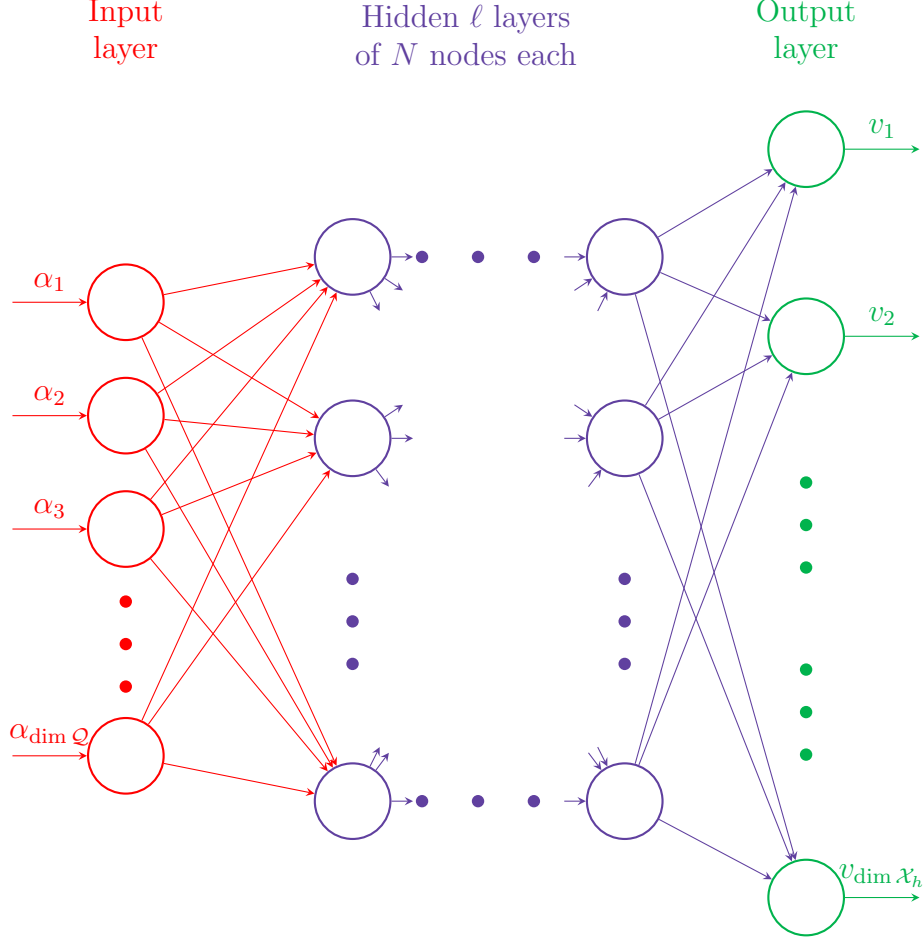


FIGURE 2. Schematic of the neural network architecture of (22). The input layer (L_{in}) receives parameter values α_i , the output layer (L_{out}) produces coefficients v_k of finite element basis expansion of a solution prediction, and the hidden layers involve the transformations Φ_m .

The training process requires a loss function $\mathcal{L} \equiv \mathcal{L}_\alpha$ that estimates the error in any approximation generated by the NN at a given parameter value α ,

$$\mathcal{L} : \mathcal{X}_h \rightarrow \mathbb{R}.$$

The calculation of $\mathcal{L}(v)$, for any $v \in \mathcal{X}_h$, at a given parameter value α , can depend on α even when v is independent of α . Although a proper notation indicating so would be \mathcal{L}_α , we dispense with the subscript to lighten the notation. The training is carried out through a stochastic minimization process for

$$\theta^* = \operatorname{argmin}_{\theta \in \Theta} \left(\frac{1}{M} \sum_{\alpha \in \mathcal{Q}_M} \mathcal{L}(\mathcal{F}_{h,\theta}(\alpha)) \right) \quad (23)$$

where \mathcal{Q}_M is a collection of M training samples, drawn from a random distribution of functions in \mathcal{Q} after endowing it with a probability measure μ . Here, in terms of computational

vectors \mathfrak{w} , the summand above is

$$\mathcal{L}(\mathcal{F}_{h,\theta}(\alpha)) = \mathcal{L}\left(\sum_{k=1}^{m_h} [\mathbb{F}_\theta(\mathfrak{w})]_k \psi_k(x)\right)$$

where \mathbb{F}_θ is as defined by the NN in (22).

In machine learning jargon, the minimization in (23) is referred to as *empirical risk minimization*. When a scheme produces candidate solution approximations $\mathfrak{w}(\alpha) \in \mathcal{X}_h$ for each α in the probability space \mathcal{Q} with measure μ , its “ideal risk” is defined as the expectation of the random variable $\mathcal{L} \circ \mathfrak{w}$, i.e.,

$$\mathbb{L}(\mathfrak{w}; \mathcal{Q}) := \int_{\mathcal{Q}} \mathcal{L}(\mathfrak{w}(\alpha)) d\mu(\alpha) = \mathbb{E}_\mu[\mathcal{L}(\mathfrak{w}(\alpha))]. \quad (24)$$

The empirical risk is the analogue computed from the M random samples in \mathcal{Q}_M drawn from \mathcal{Q} , defined by

$$\mathbb{L}(\mathfrak{w}, \mathcal{Q}_M) := \frac{1}{M} \sum_{\alpha \in \mathcal{Q}_M} \mathcal{L}(\mathfrak{w}(\alpha)). \quad (25)$$

It can be viewed as a Monte Carlo approximation to the ideal risk. Viewing the NN parameter-to-solution map $\mathcal{F}_{h,\theta}$ as a function on the probability space \mathcal{Q} , the ideal risk of the NN model is $\mathbb{L}(\mathcal{F}_{h,\theta}; \mathcal{Q})$. The objective function being minimized in (23), equaling $\mathbb{L}(\mathcal{F}_{h,\theta}, \mathcal{Q}_M)$, is then the empirical risk, a Monte Carlo approximation to the ideal risk. Desirable criteria while choosing \mathcal{L} are described in §2.5.

2.4. The “lifted” problem. To quantify the accuracy of outcomes of the trained model \mathcal{F}_{h,θ^*} from (23) in relation to the exact \mathcal{F} , as a function of $x \in \Omega$ and $\alpha \in \mathcal{Q}$, one needs an appropriate model-compliant “continuous reference metric”. As suggested in [3] and in agreement with (24), we consider the “lifted” variational problem for a group variable $\mathbf{u}(x, \alpha)$ containing the flux and solution components. as functions on the spatio-parametric domain $\Omega \times \mathcal{Q}$. For each $\alpha \in \mathcal{Q}$ the solution $\mathbf{u}(\cdot, \alpha)$ is in some trial Hilbert space \mathcal{X} and satisfies a well-posed formulation (5) with some trial space \mathcal{Y} . Generalizing over all parameters using $\mathbb{X} := L_2(\mathcal{Q}; \mathcal{X})$, the lifted formulation seeks $\mathbf{u} \in \mathbb{X}$ such that for all $\mathbf{v} \in \mathbb{Y} := L_2(\mathcal{Q}; (\mathcal{Y}_\alpha)_{\alpha \in \mathcal{Q}})$,

$$b(\mathbf{u}, \mathbf{v}) := \int_{\mathcal{Q}} b_\alpha(\mathbf{u}, \mathbf{v}) d\mu(\alpha) = \int_{\mathcal{Q}} \ell(\mathbf{v}) d\mu(\alpha) \quad (26)$$

where μ is a given probability measure on \mathcal{Q} . When the test-spaces \mathcal{Y}_α depend on α , the notation $L_2(\mathcal{Q}; (\mathcal{Y}_\alpha)_{\alpha \in \mathcal{Q}})$ indicates that, under mild measurability assumptions on the bilinear forms $b_\alpha(\cdot, \cdot)$, the space \mathbb{Y} is indeed a well-defined Hilbert space, a so-called direct integral: see [3]. Moreover, it is shown there that (26) is well-posed if and only if the individual fiber problems (5) are well-posed (in the sense of the Babuška’s theorem) *uniformly* in $\alpha \in \mathcal{Q}$. The reference norm $\|\cdot\|_{\mathbb{X}}$ for the continuous problem is reasonable because it measures errors

$$\|\mathbf{u} - \mathfrak{w}\|_{\mathbb{X}}^2 = \int_{\mathcal{Q}} \|\mathbf{u}(\cdot, \alpha) - \mathfrak{w}(\cdot, \alpha)\|_{\mathcal{X}}^2 d\mu(\alpha) = \mathbb{E}_\mu[\|\mathbf{u}(\cdot, \alpha) - \mathfrak{w}(\cdot, \alpha)\|_{\mathcal{X}}^2] \quad (27)$$

as an *expectation* over \mathcal{Q} . This underlines the stochastic nature of the learning problem.

A key consideration in our approach is the requirement that (27) be uniformly proportional to the loss (24) when $\mathfrak{w} = \mathcal{F}_{h,\theta}$ ranges over the hypothesis class. Since we will not be able to assess exact values of (24) over the entire parameter range, we use a large (random) test set $\widehat{\mathcal{Q}}_{\text{test}} \subset \mathcal{Q}$ to gauge accuracy. The errors in the ideal reference metric $\|\cdot\|_{\mathbb{X}}$ of (27)

have two sources. The first is the deviation between the ideal loss and its empirical counterpart $\mathbb{L}(\mathcal{F}_{h,\theta}, \hat{\mathcal{Q}}_{\text{test}})$. The second is the error in the *discrete Bochner norm* $\|\mathbf{u} - \mathcal{F}_{h,\theta}\|_{\mathbb{X}(\hat{\mathcal{Q}}_{\text{test}})}$ where $\mathbb{X}(\hat{\mathcal{Q}}_{\text{test}}) := \ell_2(\hat{\mathcal{Q}}_{\text{test}}; \mathcal{X})$. The former is often referred to as the *generalization error* and depends (among other things) on the size of sufficiently large test set $\hat{\mathcal{Q}}_{\text{test}} \subset \mathcal{Q}$. We do not address the associated stochastic estimation concepts (see e.g. [4]) but concentrate on estimating the empirical risk $\|\mathbf{u} - \mathcal{F}_{h,\theta}\|_{\mathbb{X}(\hat{\mathcal{Q}}_{\text{test}})}$ from corresponding computable losses $\mathbb{L}(\mathcal{F}_{h,\theta}, \hat{\mathcal{Q}}_{\text{test}})$.

2.5. Variational correctness, reliability, and efficiency. As we have seen, classical boundary value problems like (8) admit many different variational formulations of the form (5). The effectiveness of an approximation $\mathbf{w} \in \mathcal{X}$ to the exact solution \mathbf{u} in \mathcal{X} can be measured by the norm $\|\mathbf{u} - \mathbf{w}\|_{\mathcal{X}}$. Conforming finite element methods are based on a variational formulation and use a finite element subspace $\mathcal{X}_h \subset \mathcal{X}$. Following [3], we pursue loss functions that track the error, i.e., $\mathcal{L}(\mathbf{w})$ is bounded above and below by the true error $\|\mathbf{u} - \mathbf{w}\|_{\mathcal{X}}$, scaled by some \mathbf{w} -independent constants, for all candidate approximations \mathbf{w} . In the finite element literature, such estimates are well known for *a posteriori* “error estimators” which are typically based on residuals. Then they do not require knowing the exact solution \mathbf{u} but involve problem data such as source terms whose evaluation may also require some care. We now borrow the standard terminology for finite element error estimators into NN loss functions and specify what we mean by their variational correctness.

Definition 2.4. For some class of source functions f representable in computations, the loss function $\mathcal{L} : \mathcal{X}_h \rightarrow \mathbb{R}$ is said to be *reliable* in \mathcal{X}_h if there is a $C_1 > 0$ such that

$$C_1 \|\mathbf{u} - \mathbf{w}\|_{\mathcal{X}}^2 \leq \mathcal{L}(\mathbf{w})$$

for all $\mathbf{w} \in \mathcal{X}_h$. Here \mathbf{u} is the exact solution of (5). If there is a $C_2 > 0$ such that

$$\mathcal{L}(\mathbf{w}) \leq C_2 \|\mathbf{u} - \mathbf{w}\|_{\mathcal{X}}^2$$

for all $\mathbf{w} \in \mathcal{X}_h$, then the loss function \mathcal{L} is said to be *efficient*. A loss function that is both reliable and efficient is said to be *variationally correct* in \mathcal{X}_h .

When a prediction $\mathbf{w} \in \mathcal{X}_h$ is of the form (21), i.e., the expansion coefficients are NNs, the notorious uncertainty of optimization success renders *a priori* estimates based on the expressive power of NNs useless. More useful information on the fidelity of an NN-based prediction of the exact solution \mathbf{u} is drawn from the value of the loss $\mathcal{L}(\mathbf{w})$. Then the two-sided bounds of a variationally correct loss function,

$$C_1 \|\mathbf{u} - \mathbf{w}\|_{\mathcal{X}}^2 \leq \mathcal{L}(\mathbf{w}) \leq C_2 \|\mathbf{u} - \mathbf{w}\|_{\mathcal{X}}^2, \quad \mathbf{w} \in \mathcal{X}_h, \quad (28)$$

show that $\mathcal{L}(\mathbf{w})$ provides an “error certificate,” up to the values of the constants C_1 and C_2 .

In Definition 2.4, the solution $\mathbf{u} \equiv \mathbf{u}(x, \alpha)$, the candidate approximation $\mathbf{w} \equiv \mathbf{w}(x, \alpha)$, as well as the constants $C_1 \equiv C_1(\alpha)$, $C_2 \equiv C_2(\alpha)$, all depend on α . To indicate this, we often call \mathcal{L} the “fiber loss” (of the fiber problem) at α . Let α vary in some selected parameter domain $\tilde{\mathcal{Q}}$ (which is typically a strict compact subset of the \mathcal{Q} in (19)). We say that a variationally correct loss function \mathcal{L} is *robust over* $\tilde{\mathcal{Q}}$ if the constants C_1 and C_2 do not depend on the parameter α as it ranges over $\tilde{\mathcal{Q}}$. Arguably a useful concept of parameter robustness should also constrain $\|\cdot\|_{\mathcal{X}}$ to include a parameter-independent part, but this is hard to prescribe abstractly.

When \mathcal{L} is robust over $\tilde{\mathcal{Q}}$, it is shown in [3] that (28) holds uniformly for $\alpha \in \tilde{\mathcal{Q}}$ if and only if

$$C_1 \|\mathbf{u} - \mathbf{w}\|_{\mathbb{X}}^2 \leq \mathbb{L}(\mathbf{w}; \tilde{\mathcal{Q}}) \leq C_2 \|\mathbf{u} - \mathbf{w}\|_{\mathbb{X}}^2, \quad \mathbf{w} \in \mathbb{X}, \quad (29)$$

where \mathbb{L} is as in (24) and $\|\cdot\|_{\mathbb{X}}$ denotes the Bochner norm of $\mathbb{X} = L_2(\tilde{\mathcal{Q}}; \mathcal{X}_h)$, i.e., if and only if the “lifted loss” \mathbb{L} is variationally correct in \mathbb{X} . Moreover, uniform validity of (28) over $\tilde{\mathcal{Q}}$ implies the two-sided bounds

$$C_1 \|\mathbf{u} - \mathbf{w}\|_{\mathbb{X}(\tilde{\mathcal{Q}}_M)}^2 \leq \mathbb{L}(\mathbf{w}; \tilde{\mathcal{Q}}_M) \leq C_2 \|\mathbf{u} - \mathbf{w}\|_{\mathbb{X}(\tilde{\mathcal{Q}}_M)}^2, \quad \mathbf{w} \in \mathbb{X}(\tilde{\mathcal{Q}}_M), \quad (30)$$

for any $\tilde{\mathcal{Q}}_M \subset \tilde{\mathcal{Q}}$ of M samples, and for the corresponding computable quantities, involving the discrete Bochner norm of $\mathbb{X}(\tilde{\mathcal{Q}}_M) = \ell_2(\tilde{\mathcal{Q}}_M; \mathcal{X}_h)$ and the empirical risk defined in (25). Thus, variational correctness of \mathcal{L} over \mathcal{X}_h and robustness of \mathcal{L} over $\tilde{\mathcal{Q}}$ ensures variational correctness of the lifted loss \mathbb{L} over $\mathbb{X} = L_2(\tilde{\mathcal{Q}}; \mathcal{X}_h)$ and ensures that the computable $\mathbb{L}(\mathbf{w}; \tilde{\mathcal{Q}}_M)$ provides rigorous error certificates over *any* test set of random samples $\tilde{\mathcal{Q}}_M \subset \tilde{\mathcal{Q}}$. Inference on the continuous counterpart $\mathbb{L}(\mathbf{w}; \tilde{\mathcal{Q}})$ is then a matter of statistical estimation when M increases.

Two-sided inequalities like (28) have been developed extensively for error estimators in the finite element literature [2]. When the source f is not represented exactly in the finite element calculations, one can only expect a modified version of (28) to hold with additional data representation errors (often also called data oscillation terms). In our numerical studies we fix f to a simple function (without the need for such terms) to focus exclusively on α dependence. While some finite element error estimators admit bounds like (28) only when \mathbf{w} is a Galerkin approximation, note that we are requiring (28) to hold for any $\mathbf{w} \in \mathcal{X}_h$.

In the next two sections, we proceed to discuss two variationally correct loss functions \mathcal{L} compared in this paper. They are based on the variational formulations (12) and (14). In both cases, we will need the following inequality to establish variational correctness: there exists (an α -dependent) $c_\alpha > 0$ such that

$$\|(r, w)\|_{L_2} \leq c_\alpha \|A_\alpha(r, w)\|_{L_2}, \quad (r, w) \in H(\text{div}) \times \dot{H}^1. \quad (31)$$

Clearly, this inequality is independent of the finite element discretization and the specifics (including the norms and the spaces) of different variational formulations used to treat (8). The inequality (31) can be proved using the theory of mixed systems [11, Ch. II, Prop. 1.3]. A proof can be found in many places, such as [9], [12, Theorem 3.1], or [19, Lemma 4.4], and it holds provided that α satisfies the so called ellipticity condition, i.e., there exists two finite positive constants $\alpha_{\min}, \alpha_{\max} > 0$ such that

$$\alpha_{\min} \leq \alpha(x) \leq \alpha_{\max} \quad \text{a.e. } x \in \Omega. \quad (32)$$

Then one can make explicit the α -dependence of c_α , namely, there exists [19, Lemma 4.4] some α -independent constant $c_0 > 0$ such that

$$c_\alpha = c_0 \left(1 + \frac{\max(1, \alpha_{\max})}{\alpha_{\min}} \right). \quad (33)$$

One often also uses the graph norm generated by the operator A_α , namely $(\|\mathbf{w}\|_{L_2}^2 + \|A_\alpha \mathbf{w}\|_{L_2}^2)^{1/2}$, for any $\mathbf{w} = (r, w) \in H(\text{div}) \times H^1$. By triangle inequality,

$$\begin{aligned} \|\nabla w\|_{L_2} + \|\text{div } r\|_{L_2} &\leq \|\alpha r + \nabla w\|_{L_2} + \|\alpha r\|_{L_2} + \|\text{div } r\|_{L_2} \\ &\leq \|A_\alpha(r, w)\|_{L_2} + \bar{\alpha} \|r\|_{L_2}, \end{aligned}$$

showing that

$$\|\mathfrak{w}\|_{H(\text{div}) \times H^1}^2 \lesssim \|\mathfrak{w}\|_{L_2}^2 + \|A_\alpha \mathfrak{w}\|_{L_2}^2. \quad (34)$$

Here and throughout, when two quantities A and B admit a bound $A \leq CB$ for some constant $C > 0$ which may depend on α but is independent of h , we write $A \lesssim B$, and we write $A \sim B$ to mean that both $A \lesssim B$ and $B \lesssim A$ hold. It is easy to see that

$$\|\mathfrak{w}\|_{L_2}^2 + \|A_\alpha \mathfrak{w}\|_{L_2}^2 \sim \|\mathfrak{w}\|_{H(\text{div}) \times H^1}^2, \quad \mathfrak{w} \in H(\text{div}) \times H^1, \quad (35)$$

since the reverse bound of (34) can also be established using the triangle inequality.

3. THE FOSLS LOSS FUNCTION

3.1. The FOSLS solution. We consider the FOSLS method [12], based on (11) or (12), in its lowest order form. Since A_α as an operator in (10) is onto, we can symmetrize the variational formulation in (11) by writing every test function \mathfrak{v} there as $A_\alpha \mathfrak{w}$ for some trial function \mathfrak{w} to get the following equivalent FOSLS formulation: find $\mathbf{u} \in \mathcal{X}^{\text{fos}}$ satisfying

$$b_\alpha^{\text{fos}}(\mathbf{u}, \mathfrak{w}) = \ell^{\text{fos}}(\mathfrak{w}), \quad \text{for all } \mathfrak{w} \in \mathcal{X}^{\text{fos}}, \quad (36a)$$

where

$$\mathcal{X}^{\text{fos}} = H(\text{div}) \times \mathring{H}^1, \quad \|(r, w)\|_{\mathcal{X}^{\text{fos}}}^2 = \|r\|_{H(\text{div})}^2 + \|w\|_{\mathring{H}^1}^2. \quad (36b)$$

$$b_\alpha^{\text{fos}}(\mathbf{u}, \mathfrak{w}) := (A_\alpha \mathbf{u}, A_\alpha \mathfrak{w})_{L_2}, \quad \ell^{\text{fos}}(\mathfrak{w}) := (F, A_\alpha \mathfrak{w})_{L_2}. \quad (36c)$$

Here $F = (0, f)$ and $(\cdot, \cdot)_{L_2}$ denotes the inner product in L_2 or its Cartesian products.

The lowest order FOSLS scheme sets the finite element space \mathcal{X}_h to a subspace of \mathcal{X}^{fos} described now. Let Ω_h be a conforming simplicial finite element mesh of Ω , where h denotes the maximal diameter of elements in the mesh. Let \mathcal{P}_k denote the space of piecewise polynomials of degree at most k on each element of Ω_h (without any inter-element continuity conditions). Then the lowest order Raviart-Thomas subspace [35] of $H(\text{div})$ is

$$\mathcal{RT}_0 = [\mathcal{P}_0^d + x\mathcal{P}_0] \cap H(\text{div}). \quad (37a)$$

Using also the lowest order Lagrange space on the same mesh,

$$\mathcal{U}_1 = \mathcal{P}_1 \cap \mathring{H}^1 \quad (37b)$$

the FOSLS method sets the finite element space \mathcal{X}_h to

$$\mathcal{X}_h^{\text{fos}} = \mathcal{RT}_0 \times \mathcal{U}_1.$$

Given a diffusion parameter α , the FOSLS approximation to the exact solution $\mathbf{u} = \mathbf{u}(\cdot, \alpha) = (q(\cdot, \alpha), u(\cdot, \alpha))$ in $H(\text{div}) \times \mathring{H}^1$ satisfying (8) is the unique $\mathbf{u}_h \in \mathcal{X}_h^{\text{fos}}$ satisfying

$$(A_\alpha \mathbf{u}_h^{\text{fos}}, A_\alpha \mathfrak{w})_{L_2} = (F, A_\alpha \mathfrak{w})_{L_2}, \quad \text{for all } \mathfrak{w} \in \mathcal{X}_h^{\text{fos}}, \quad (38)$$

where $F = (0, f)$ and A_α is as in (10). The inner product in L_2 or its Cartesian products is denoted above and throughout by $(\cdot, \cdot)_{L_2}$ and the corresponding norm by $\|\cdot\|_{L_2}$. The evaluation of the FOSLS parameter-to-solution map for each given α

$$\alpha \xrightarrow{\mathcal{F}_h^{\text{fos}}} \mathbf{u}_h^{\text{fos}}(\alpha), \quad \mathcal{F}_h^{\text{fos}}(\alpha) = \mathbf{u}_h^{\text{fos}}$$

can be obtained by solving (38). Note that the set of equations contained in (38) for each test function is the same as the set of normal equations for the minimization problem

$$\min_{\mathfrak{w} \in \mathcal{X}_h^{\text{fos}}} \|A_\alpha \mathfrak{w} - F\|_{L_2}^2.$$

Thus, for each $\alpha \in \mathcal{Q}$, the FOSLS solution can be characterized as the following unique minimizer:

$$\mathcal{F}_h^{\text{fos}}(\alpha) = \mathbf{u}_h^{\text{fos}} = \operatorname{argmin}_{\mathbf{w} \in \mathcal{X}_h^{\text{fos}}} \|A_\alpha \mathbf{w} - F\|_{L_2}^2.$$

3.2. Variational correctness of the FOSLS loss function. For the fiber problem at any $\alpha \in \mathcal{Q}$, the “FOSLS loss function” is defined by

$$\mathcal{L}^{\text{fos}}(\mathbf{w}) := \|A_\alpha \mathbf{w} - F\|_{L_2}^2, \quad \mathbf{w} \in \mathcal{X}_h^{\text{fos}}. \quad (39)$$

The FOSLS parameter-to-solution map is found by performing the minimization in (23) with \mathcal{L} set to \mathcal{L}^{fos} and \mathcal{X}_h set to $\mathcal{X}_h^{\text{fos}}$. As previously mentioned, we refer to this residual in (39) as “fiber loss.” Note that it explicitly depends on α through the operator A_α (even though to avoid notational clutter we have chosen to write \mathcal{L}^{fos} instead of $\mathcal{L}_\alpha^{\text{fos}}$). The next proposition is essentially contained in [12] but we include a short proof for completeness.

Proposition 3.1. *For each $\alpha \in \mathcal{Q}$, \mathcal{L}^{fos} is variationally correct in $\mathcal{X}_h^{\text{fos}}$, i.e., there are α -dependent constants $C_1, C_2 > 0$ such that*

$$C_1 \|\mathbf{u}(\alpha) - \mathbf{w}\|_{\mathcal{X}^{\text{fos}}}^2 \leq \mathcal{L}^{\text{fos}}(\mathbf{w}) \leq C_2 \|\mathbf{u}(\alpha) - \mathbf{w}\|_{\mathcal{X}^{\text{fos}}}^2, \quad (40)$$

for all $\mathbf{w} \in \mathcal{X}^{\text{fos}}$. Here $\mathbf{u}(\alpha) = (q(\alpha), u(\alpha)) \in \mathcal{X}^{\text{fos}}$ is the exact solution of (8) found using (12) or equivalently (36).

Proof. First note that Since $\mathbf{u} = \mathbf{u}(\alpha)$ is the exact solution, $F = A_\alpha \mathbf{u}$ and $\mathcal{L}^{\text{fos}}(\mathbf{w}) = \|A_\alpha(\mathbf{w} - \mathbf{u})\|_{L_2}^2$ for any \mathbf{w} in $\mathcal{X}_h^{\text{fos}}$. Then (31) implies that

$$\|\mathbf{u} - \mathbf{w}\|_{L_2}^2 + \|A_\alpha(\mathbf{u} - \mathbf{w})\|_{L_2}^2 \leq (c_\alpha^2 + 1) \|A_\alpha(\mathbf{u} - \mathbf{w})\|_{L_2}^2 = (c_\alpha^2 + 1) \mathcal{L}^{\text{fos}}(\mathbf{w}).$$

Hence by (35),

$$\|\mathbf{u} - \mathbf{w}\|_{\mathcal{X}^{\text{fos}}}^2 \lesssim \mathcal{L}^{\text{fos}}(\mathbf{w})$$

for all $\mathbf{w} \in \mathcal{X}^{\text{fos}}$, showing reliability of the FOSLS loss function (the lower inequality of (40)). Furthermore, a simple application of triangle inequality shows that

$$\|A_\alpha \mathbf{w}\|_{L_2} \leq \max\{1, \alpha_{\max}\} \|\mathbf{w}\|_{\mathcal{X}^{\text{fos}}}, \quad (41)$$

for all $\mathbf{w} \in \mathcal{X}^{\text{fos}}$. Thus, \mathcal{L}^{fos} is also efficient. \square

Although FOSLS loss is variationally correct, it is not robust over the unbounded parameter domain \mathcal{Q} in (19). Let a and b be two positive constants and consider the bounded parameter domain

$$\mathcal{Q}^{[a,b]} = \{\alpha \in \mathcal{Q} : a \leq \min_{x \in \Omega} \alpha(x), \max_{x \in \Omega} \alpha(x) \leq b\}. \quad (42)$$

From the proof of Proposition 3.1 it is clear that the FOSLS loss is robust over the bounded parameter domain $\mathcal{Q}^{[a,b]}$. Indeed the constants C_1 and C_2 produced in the proof depend only on α_{\max} and c_α (which can depend also on α_{\min} per (33)), so C_1 and C_2 can also be expressed using only a and b , making them independent of $\alpha \in \mathcal{Q}^{[a,b]}$. As a consequence, per the discussion in Section 2.5, the lifted loss function

$$\mathbb{L}^{\text{fos}}(\mathbf{w}; \mathcal{Q}_M^{[a,b]}) := \frac{1}{M} \sum_{\alpha \in \mathcal{Q}_M^{[a,b]}} \mathcal{L}^{\text{fos}}(\mathbf{w}(\alpha)) \quad (43)$$

is variationally correct for any set of M samples $\mathcal{Q}_M^{[a,b]} \subset \mathcal{Q}^{[a,b]}$ and admits two-sided bounds as in (30) with efficiency and reliability constants depending solely on a and b .

4. THE DPG LOSS FUNCTION

In this section we introduce the ultraweak DPG method for (8). This method was first presented in [19] and put in a larger context in the recent review [21]. After presenting the discrete method, we present the DPG loss function and discuss its variational correctness.

4.1. The DPG solution. The ultraweak DPG formulation can be described for very general PDEs, in discretized and undiscretized forms, and in arbitrary order spaces (see [21, Section 7]). It is a mesh-dependent variational formulation in so-called “broken graph spaces” whose unbroken counterpart was already given in Example 2.3. For the discretization, we restrict ourselves here to the simplest lowest order case for transparent presentation of the ideas. Let $\text{tr}(w)$ and $\text{tr}_n(r)$ denote element-wise trace operators defined by $\text{tr}(w)|_{\partial K} = w|_{\partial K}$ and $\text{tr}_n(r)|_{\partial K} = (r \cdot n)|_{\partial K}$. Here and throughout n denotes the outward unit normals on element boundaries. Using the spaces defined in (37), define the following spaces on element boundaries by

$$\begin{aligned}\hat{\mathcal{U}}_1 &= \{\hat{w} : \hat{w}|_{\partial K} = w|_{\partial K} \text{ for some } w \in \mathcal{U}_1 \text{ for all } K \in \Omega_h\} \\ \hat{\mathcal{RT}}_0 &= \{\hat{q}_n : \hat{q}_n|_{\partial K} = (r \cdot n)|_{\partial K} \text{ for some } r \in \mathcal{RT}_0 \text{ for all } K \in \Omega_h\}.\end{aligned}$$

Since the lowest order spaces \mathcal{U}_1 and \mathcal{RT}_0 are completely defined by their element-wise traces, the spaces $\hat{\mathcal{U}}_1$ and $\hat{\mathcal{RT}}_0$ are isomorphic to \mathcal{U}_1 and \mathcal{RT}_0 , respectively. We write a generic element of $\hat{\mathcal{RT}}_0$ as $\hat{r} \cdot n$ or \hat{r}_n to indicate its dependence on n .

The lowest order ultraweak DPG method uses distinct trial and test spaces,

$$\mathcal{X}_h^{\text{dpg}} = \mathcal{P}_0^d \times \mathcal{P}_0 \times \hat{\mathcal{U}}_1 \times \hat{\mathcal{RT}}_0, \quad \mathcal{Y}_h^{\text{dpg}} = \mathcal{P}_2^d \times \mathcal{P}_{d+1}.$$

Recall the formal adjoint of A_α , given by (15). As in Example 2.3, it is used to norm the test space by

$$\|(\tau, \nu)\|_{\mathcal{Y}}^2 = \sum_{K \in \Omega_h} \left(\|A_\alpha^*(\tau, \nu)\|_{L_2(K)}^2 + \|(\tau, \nu)\|_{L_2(K)}^2 \right) \quad (44)$$

for all $(\tau, \nu) \in \mathcal{Y}_h^{\text{dpg}}$. Note that the differential operators within A_α^* are applied element by element when computing the norm (44), which is referred to as a “broken graph norm”. The corresponding inner product is denoted by $(\cdot, \cdot)_{\mathcal{Y}}$ and it is an essential ingredient in computations with the DPG method. Different flavors of the method are obtained by changing the \mathcal{Y} inner product, as we shall see later.

Define the DPG bilinear form by

$$b_\alpha^{\text{dpg}}((q, u, \hat{u}, \hat{q} \cdot n), (\tau, \nu)) = \sum_{K \in \Omega_h} \left(\int_K (q, u) \cdot A_\alpha^*(\tau, \nu) + \int_{\partial K} \hat{u} \tau \cdot n + \int_{\partial K} \hat{q} \cdot n \nu \right), \quad (45)$$

for any $(q, u, \hat{u}, \hat{q} \cdot n) \in \mathcal{X}_h^{\text{dpg}}$ and $(\tau, \nu) \in \mathcal{Y}_h^{\text{dpg}}$. We omit the standard Lebesgue measure in integrals such as the above when there can be no confusion. The DPG method can be presented as a Petrov-Galerkin formulation or equivalently as a mixed Galerkin method. We only present the latter (and refer the reader to [21] for other equivalent forms). Given an F in $L_2(\Omega)^d \times L_2(\Omega)$ (which, as before, will be set to $(0, f)$ with f as in (8)), the DPG method produces an approximate solution $\mathbf{u}_h^{\text{dpg}} = (q_h^{\text{dpg}}, u_h^{\text{dpg}}, \hat{u}_h^{\text{dpg}}, \hat{q}_h^{\text{dpg}})$ in $\mathcal{X}_h^{\text{dpg}}$ as well as a built-in

approximate error representation e_h in $\mathcal{Y}_h^{\text{dpg}}$, that together satisfy the equations

$$(e_h, y)_{\mathcal{Y}} + b_{\alpha}^{\text{dpg}}(\mathbf{u}_h^{\text{dpg}}, y) = (F, y)_{L_2}, \quad y \in \mathcal{Y}_h^{\text{dpg}}, \quad (46a)$$

$$b_{\alpha}^{\text{dpg}}(\mathbf{w}, e_h) = 0, \quad \mathbf{w} \in \mathcal{X}_h^{\text{dpg}}. \quad (46b)$$

This generates the DPG parameter-to-solution map

$$\alpha \xrightarrow{\mathcal{F}_h^{\text{dpg}}} \mathbf{u}_h^{\text{dpg}}, \quad \mathcal{F}_h^{\text{dpg}}(\alpha) = \mathbf{u}_h^{\text{dpg}},$$

for any $\alpha \in \mathcal{Q}$, i.e., the application of $\mathcal{F}_h^{\text{dpg}}$ to any α is obtained by solving (46). We want to approximate $\mathcal{F}_h^{\text{dpg}}$ by a NN.

4.2. The DPG loss function. The DPG loss function is computed in two steps. First, given any $\mathbf{w} \in \mathcal{X}_h^{\text{dpg}}$, we compute a residual representation $\varepsilon_h^{\mathbf{w}} \in \mathcal{Y}_h^{\text{dpg}}$ by solving

$$(\varepsilon_h^{\mathbf{w}}, \mathbf{v})_{\mathcal{Y}} = (F, \mathbf{v})_{L_2} - b_{\alpha}^{\text{dpg}}(\mathbf{w}, \mathbf{v}), \quad \text{for all } \mathbf{v} \in \mathcal{Y}_h^{\text{dpg}}. \quad (47a)$$

Note that this is the same equation as (46a) with e_h and $\mathbf{u}_h^{\text{dpg}}$ replaced by $\varepsilon_h^{\mathbf{w}}$ and \mathbf{w} , respectively. As before we focus first on parameter-dependent DPG fiber loss of \mathbf{w} at α , which, using $\varepsilon_h^{\mathbf{w}}$, we define by

$$\mathcal{L}^{\text{dpg}}(\mathbf{w}) := \|\varepsilon_h^{\mathbf{w}}\|_{\mathcal{Y}^{\text{dpg}}}^2, \quad \mathbf{w} \in \mathcal{X}_h^{\text{dpg}}, \quad (47b)$$

where the α -dependent \mathcal{Y}^{dpg} norm is as defined in (44). We can also write this loss in operator form. One way to do so quickly is to introduce the mappings $R_{\alpha, h} : \mathcal{Y}_h^{\text{dpg}} \rightarrow (\mathcal{Y}_h^{\text{dpg}})'$ and $B_{\alpha, h}^{\text{dpg}} : \mathcal{X}_h^{\text{dpg}} \rightarrow (\mathcal{Y}_h^{\text{dpg}})'$, defined by $(R_{\alpha, h}y)(z) = (y, z)_{\mathcal{Y}}$ and $(B_{\alpha, h}^{\text{dpg}}\mathbf{w})(y) = B_{\alpha, h}^{\text{dpg}}(\mathbf{w}, y)$ for all $\mathbf{w} \in \mathcal{X}_h^{\text{dpg}}$ and $y, z \in \mathcal{Y}_h^{\text{dpg}}$. Then (47a) can be expressed as

$$R_{\alpha, h}\varepsilon_h^{\mathbf{w}} + B_{\alpha, h}^{\text{dpg}}\mathbf{w} = F$$

and (47b) becomes

$$\mathcal{L}^{\text{dpg}}(\mathbf{w}) = \|R_{\alpha, h}^{-1}(F - B_{\alpha, h}^{\text{dpg}}\mathbf{w})\|_{\mathcal{Y}}^2, \quad \mathbf{w} \in \mathcal{X}_h^{\text{dpg}}. \quad (48)$$

The key feature that makes this fiber loss function quickly computable is the block diagonal nature of the matrix of $R_{\alpha, h}$ (representing the Riesz map), with one block per element (so its inverse appearing above can be computed locally element by element). This is a consequence of the fact that functions in $\mathcal{Y}_h^{\text{dpg}}$ have no inter-element continuity constraints.

4.3. Variational correctness of the DPG loss function. The variational setting which the ultraweak DPG method is based upon is more complicated than that of FOSLS. First consider the trial space. To describe the norm in an infinite-dimensional Hilbert space \mathcal{X}^{dpg} enclosing the discrete computational trial space $\mathcal{X}_h^{\text{dpg}}$, let

$$\text{tr} : \dot{H}^1 \longrightarrow \prod_{K \in \Omega_h} H^{1/2}(\partial K), \quad (\text{tr } u)|_{\partial K} = u|_{\partial K},$$

and

$$\text{tr}_n : H(\text{div}) \longrightarrow \prod_{K \in \Omega_h} H^{-1/2}(\partial K), \quad (\text{tr}_n r)|_{\partial K} = r \cdot n|_{\partial K},$$

denote element-wise boundary trace operators extended to the indicated global Sobolev spaces. We define norms of trace and flux functions on element boundaries using their preimages of the appropriate trace map by

$$\|\hat{u}\|_{\dot{H}^{1/2}(\partial\Omega_h)} := \inf_{u \in \text{tr}^{-1}\{\hat{u}\}} \|u\|_{\dot{H}^1}, \quad \|\hat{r}_n\|_{H^{-1/2}(\partial\Omega_h)} := \inf_{q \in \text{tr}_n^{-1}\{\hat{r}_n\}} \|q\|_{H(\text{div})}$$

and set $\dot{H}^{1/2}(\partial\Omega_h)$ and $H^{-1/2}(\partial\Omega_h)$ to be the ranges of $\text{tr}(\cdot)$ and $\text{tr}_n(\cdot)$, respectively. The well-posedness of ultraweak DPG formulation can be proven in the trial norm defined by

$$\|(r, w, \hat{w}, \hat{r})\|_{\mathcal{X}^{\text{dpg}}}^2 = \|r\|_{L_2}^2 + \|w\|_{L_2}^2 + \|\hat{w}\|_{\dot{H}^{1/2}(\partial\Omega_h)}^2 + \|n \cdot \hat{r}\|_{H^{-1/2}(\partial\Omega_h)}^2. \quad (49)$$

The DPG test space is set to the product of the following spaces

$$H(\text{div}, \Omega_h) = \prod_{K \in \Omega_h} H(\text{div}, K), \quad H^1(\Omega_h) = \prod_{K \in \Omega_h} H^1(K),$$

representing piecewise $H(\text{div})$ and H^1 spaces (containing functions without any interelement continuity constraints). With these definitions, we can now give the undiscretized DPG formulation, representing the fiber problem at some given parameter α , for the exact solution with four components

$$\mathbf{u}^{\text{dpg}} = (q, u, \hat{u}, \hat{q}_n). \quad (50a)$$

Here (q, u) is the exact solution of (8), $\hat{u} = \text{tr}(u)$ and $\hat{q}_n = \text{tr}_n(q)$. It is well known that \mathbf{u}^{dpg} is the unique function in \mathcal{X}^{dpg} satisfying

$$b_\alpha^{\text{dpg}}(\mathbf{u}^{\text{dpg}}, \mathbf{v}) = \ell^{\text{dpg}}(\mathbf{v}), \quad \text{for all } \mathbf{v} \in \mathcal{Y}^{\text{dpg}}, \quad (50b)$$

where

$$\mathcal{X}^{\text{dpg}} = L_2(\Omega)^d \times L_2(\Omega) \times \dot{H}^{1/2}(\partial\Omega_h) \times H^{-1/2}(\partial\Omega_h), \quad \text{normed by (49),} \quad (50c)$$

$$\mathcal{Y}^{\text{dpg}} = H(\text{div}, \Omega_h) \times H^1(\Omega_h), \quad \text{normed by broken graph norm (44),} \quad (50d)$$

$$b_\alpha^{\text{dpg}}((q, u, \hat{u}, \hat{q} \cdot n), (\tau, \nu)) = \sum_{K \in \Omega_h} \left(\langle \tau \cdot n, \hat{u} \rangle_{\partial K} + \langle \hat{q} \cdot n, \nu \rangle_{\partial K} + \int_K (q, u) \cdot A_\alpha^*(\tau, \nu) \right) \quad (50e)$$

$$\ell^{\text{dpg}}((\tau, \nu)) = (f, \nu)_{L_2}, \quad (50f)$$

where $\langle \cdot, \cdot \rangle_{\partial K}$ denotes the duality pairing between $H^{-1/2}(\partial K)$ and $H^{1/2}(\partial K)$. Note that $\mathcal{Y}_h^{\text{dpg}} \subset \mathcal{Y}$ and $\mathcal{X}_h^{\text{dpg}} \subset \mathcal{X}^{\text{dpg}}$. Since the exact problem has zero error, with a zero error representation $e = 0$, we can expand (50b) into a mixed method of the form (46) to see that (46) is indeed a conforming discretization of the exact formulation (50). The following result is obtained by putting together known results from DPG analysis [14, 21].

Proposition 4.1. *The DPG loss function \mathcal{L}^{dpg} is variationally correct in $\mathcal{X}_h^{\text{dpg}}$ when f is in \mathcal{P}_0 , i.e., there are α -dependent constants $C_1, C_2 > 0$ (that depend on c_α in (31)) such that*

$$C_1 \|\mathbf{u}^{\text{dpg}} - \mathbf{w}\|_{\mathcal{X}^{\text{dpg}}}^2 \leq \mathcal{L}^{\text{dpg}}(z) \leq C_2 \|\mathbf{u}^{\text{dpg}} - \mathbf{w}\|_{\mathcal{X}^{\text{dpg}}}^2,$$

for all $\mathbf{w} \in \mathcal{X}^{\text{dpg}}$. Here $\mathbf{u}^{\text{dpg}} = (q, u, \hat{u}, \hat{q}_n)$ is the exact solution of (50).

Proof. We apply [21, Theorem 6.4] (see also [14]) which yields

$$\begin{aligned} \gamma \|\mathbf{u}^{\text{dpg}} - \mathbf{w}\|_{\mathcal{X}^{\text{dpg}}} &\leq \|I_h\| \|\varepsilon_h^{\mathbf{w}}\|_{\mathcal{Y}^{\text{dpg}}} + \text{osc}(F), \\ \|\varepsilon_h^{\mathbf{w}}\|_{\mathcal{Y}^{\text{dpg}}} &\leq \|b_\alpha^{\text{dpg}}\| \|\mathbf{u}^{\text{dpg}} - \mathbf{w}\|_{\mathcal{X}^{\text{dpg}}}, \end{aligned} \quad (51)$$

provided there is a continuous operator $\Pi_h : \mathcal{Y}^{\text{dpg}} \rightarrow \mathcal{Y}_h^{\text{dpg}}$, called a Fortin operator, satisfying

$$b_\alpha^{\text{dpg}}(\mathbf{w}_h, y - \Pi_h y) = 0 \quad \text{for all } \mathbf{w}_h \in \mathcal{X}_h^{\text{dpg}}, y \in \mathcal{Y}^{\text{dpg}}, \quad (52)$$

and provided the norm $\|b_\alpha^{\text{dpg}}\|$ and the inf-sup constant γ of the DPG form over $\mathcal{X}^{\text{dpg}} \times \mathcal{Y}^{\text{dpg}}$ are finite and positive. In (51), $\|\Pi_h\|$ is the operator norm of Π_h and

$$\text{osc}(F) = \|F \circ (I - \Pi_h)\|_{(\mathcal{Y}^{\text{dpg}})'}.$$

Since the DPG loss function is defined by (47b), the estimates of (51) prove the result once we verify these conditions and show that $\text{osc}(F)$ vanishes.

To obtain the required Fortin operator Π_h , recall that for any $r \in H(\text{div})$ and $w \in \mathring{H}^1$, by [21, Theorems 4.1 and 5.8], there are functions $\Pi_h^g w \in \mathcal{P}_{1+d}$ and $\Pi_h^d r \in \mathcal{P}_2^d$ such that

$$\begin{aligned} \int_K \Pi_h^g w - w &= 0, & \int_F \Pi_h^g w - w &= 0, \\ \int_K \Pi_h^d r - r &= 0, & \int_{\partial K} (\Pi_h^d r - r) \cdot n &= 0, \end{aligned} \quad (53)$$

for all mesh elements K and all facets F of ∂K . In the last equality the integral of $r \cdot n$ over ∂K must in general be interpreted as a functional action of the $H^{-1/2}(\partial K)$ trace $(r \cdot n)|_{\partial K}$ on ∂K . Setting $\Pi_h : \mathcal{Y}^{\text{dpg}} \rightarrow \mathcal{Y}_h^{\text{dpg}}$ by $\Pi_h(r, w) = (\Pi_h^d r, \Pi_h^g w)$ the same theorems yield the bound

$$\|\Pi_h(r, w)\|_{H(\text{div}) \times \mathring{H}^1} \lesssim \|(r, w)\|_{H(\text{div}) \times \mathring{H}^1}.$$

Now exactly the same argument that proved (35) shows that the above inequality implies

$$\|\Pi_h(r, w)\|_{\mathcal{Y}^{\text{dpg}}} \lesssim \|(r, w)\|_{\mathcal{Y}^{\text{dpg}}}, \quad (r, w) \in \mathcal{Y}^{\text{dpg}},$$

so $\|\Pi_h\| \lesssim 1$. Also, (53) implies $b_\alpha^{\text{dpg}}(\mathbf{w}_h, (r, w) - \Pi_h(r, w)) = 0$ for all $\mathbf{w}_h \in \mathcal{X}_h^{\text{dpg}}$ and $(r, w) \in \mathcal{Y}^{\text{dpg}}$. Furthermore,

$$\text{osc}(F) = \sup_{y \in \mathcal{Y}^{\text{dpg}}} \frac{(F, y - \Pi_h y)_{L_2}}{\|y\|_{\mathcal{Y}^{\text{dpg}}}}$$

vanishes since (53) implies

$$(F, (r, w) - \Pi_h(r, w))_{L_2} = (f, w - \Pi_h^g w)_{L_2} = 0$$

for all $(r, w) \in \mathcal{Y}^{\text{dpg}}$ since $f \in \mathcal{P}_0$. It only remains to verify that

$$\|b_\alpha^{\text{dpg}}\| \lesssim 1, \quad 1 \lesssim \gamma. \quad (54)$$

These are shown in [21, Theorem 7.6 and Example 7.8] (see also [19]) using (31). \square

The DPG parameter-to-solution map is found by performing the minimization in (23) with \mathcal{L} set to \mathcal{L}^{dpg} and \mathcal{X}_h set to $\mathcal{X}_h^{\text{dpg}}$. Precisely as in the preceding section, we employ the lifted loss function

$$\mathbb{L}^{\text{dpg}}(\mathbf{w}; \mathcal{Q}_M) := \frac{1}{M} \sum_{\alpha \in \mathcal{Q}_M} \mathcal{L}^{\text{dpg}}(\mathbf{w}(\alpha)), \quad (55)$$

for the empirical risk minimization that trains the NN. As in the FOSLS case, one can track constants to argue robustness for bounded parameter domains such as in (42). But we have instead chosen to devote an entire section (Section 6) to the robustness of the DPG loss.

5. PERFORMANCE OF NEURAL NETWORK PREDICTIONS

In this section, we show numerical results for both types of learning methods and compare their performance. In these comparisons we do not take the (substantial, offline) training costs into account but focus solely on the achieved accuracy, reflected by the respective loss functions. Since the evaluation of DPG losses is slightly more expensive, it will be important to understand features that may offset somewhat higher cost. Our numerical results in this section show comparable performance for the parameter-to-solution map approximations obtained using NN of the form (22), trained either using the FOSLS loss or using the DPG loss. This then forms the backdrop for further performance gains described in Section 6.

In all our numerical results we set $m_\alpha = 4$, Ω to the unit square, split into four congruent subdomains, partitioned into a finite element mesh (that aligns with the subdomains) of mesh size $h \approx 0.1$, and set $f \equiv 1$. (Smaller h and higher m_α can certainly be considered without any change to the previous theory provided more computational resources are available.) We use a fixed NN architecture setting in all experiments of this section with the number of hidden layers equal to $L = 13$, $N = 128$, and rank $r = 32$ (see §2.3 for how they determine the NN map). Since $m_\alpha = 4$ and every α in \mathcal{Q} is determined by four numbers. With a small abuse of notation, from now on we do not distinguish between the function α and the vector α and write α as a 4-tuple, i.e., $\alpha = (\alpha_1, \alpha_2, \alpha_3, \alpha_4)$, when reporting experimental results. The components of the tuple correspond to values of α on the bottom left, bottom right, top left and top right subdomains, as illustrated in the example in Figure 1b. We use NGSolve [37], an open-source finite element library, for shape functions, matrix assembly, and efficient implementation of all finite element structures needed for computing the loss functions. We implement the DPG and FOSLS loss functions within the PyTorch [33] framework, interfacing it with NGSolve, so as to leverage the backward differentiation and stochastic gradient descent facilities of PyTorch.

Each NN is trained using a set of $M = 1024$ random samples of α found using a *normal distribution* with a specified mean value $\bar{\alpha} \in \mathbb{R}^4$ and standard deviation σ , as follows. We take the square root of the positive numbers given in $\bar{\alpha}$, then sample from a normal distribution around that mean with standard deviation σ , and then square the result (so α is from a chi-squared distribution). This is done to guarantee that the material coefficients are positive in all experiments (while maintaining the ability to sample difficult extreme values of α). We set the number of epochs to 5000, the batch size to 32 and the learning rate to 0.0001 in all trainings. The NN predictions of solutions are then tested against the actual finite element solution on random samples of α in multiple ways as described below. Note that $M = 1024$ is a somewhat modest number of training samples (even if we had restricted α to a bounded four-dimensional cube). We shall see that good predictions can be obtained even for α in the unbounded four-dimensional domain of (19) with just $M = 1024$ training samples.

5.1. Visualization of some NN predictions. We first present a qualitative visual comparison between the neural network predictions and the exact finite element solution for some α taken from the distribution described before, with mean and standard deviation set by

$$\bar{\alpha} = (10^{-1}, 1, 1, 10^{-1}), \quad \sigma = 0.1. \quad (56)$$

We generate a pair of predictions

$$\mathbf{u}_\theta^{\text{fos}} = (q_\theta^{\text{fos}}, u_\theta^{\text{fos}}) \quad \text{and} \quad \mathbf{u}_\theta^{\text{dpg}} = (q_\theta^{\text{dpg}}, u_\theta^{\text{dpg}}, \hat{u}_\theta^{\text{dpg}}, \hat{q}_\theta^{\text{dpg}}) \quad (57)$$

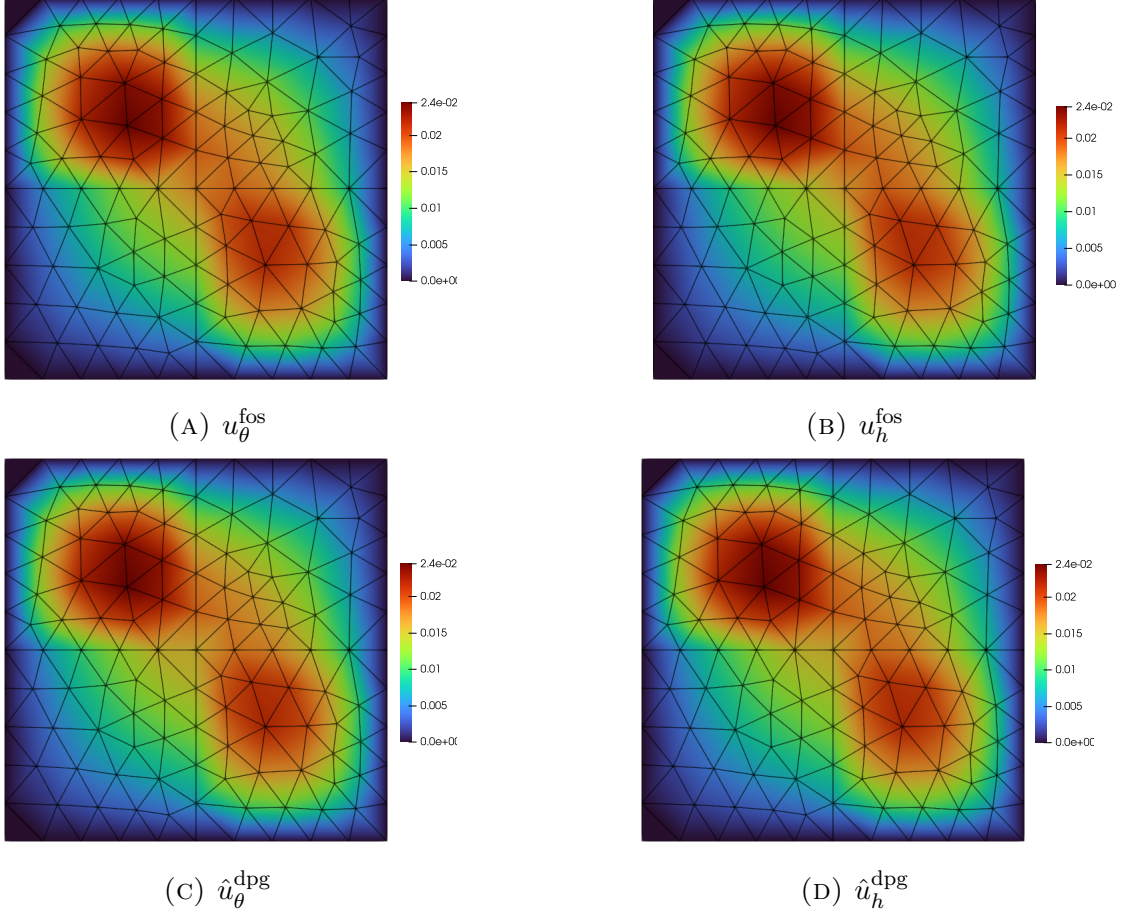


FIGURE 3. Visualization of actual finite element solutions (right) and predictions (left) from an NN trained around the $\bar{\alpha}$ in (56), for the sampled α in (58).

from an NN trained with the FOSLS loss and an NN trained with the DPG loss, respectively. The last two DPG components, namely $(\hat{q}_{\theta}^{\text{dpg}}, \hat{u}_{\theta}^{\text{dpg}})$, can be identified to be in the same space where FOSLS solution belongs since they have a unique extension from the element boundaries to the element interior as functions in $\mathcal{X}_h^{\text{fos}}$.

We compare these predictions with the corresponding finite element solutions $(q_h^{\text{fos}}, u_h^{\text{fos}})$ and $(\hat{q}_h^{\text{dpg}}, \hat{u}_h^{\text{dpg}})$, obtained by solving (38) and (46), respectively, for the same α . The prediction u_{θ}^{fos} and the finite element solution u_h^{fos} are plotted over Ω in Figures 3a and 3b for a randomly chosen

$$\alpha_{\text{sample}} = (0.0904, 0.7255, 0.9192, 0.1948). \quad (58)$$

The analogous quantities from the DPG method of comparable order are $\hat{u}_{\theta}^{\text{dpg}}$ and \hat{u}_h^{dpg} (obtained using the same α_{sample}). They are plotted in Figures 3c and 3d. Clearly, by a visual comparison, we see that both the neural networks perform very well in predicting the finite element solution.

Now, we repeat the exercise, this time for a different α deliberately selected to be away from the mean $\bar{\alpha}$ in (56), namely

$$\alpha_{\text{sample}} = (0.43, 1.0, 1.0, 0.43). \quad (59)$$

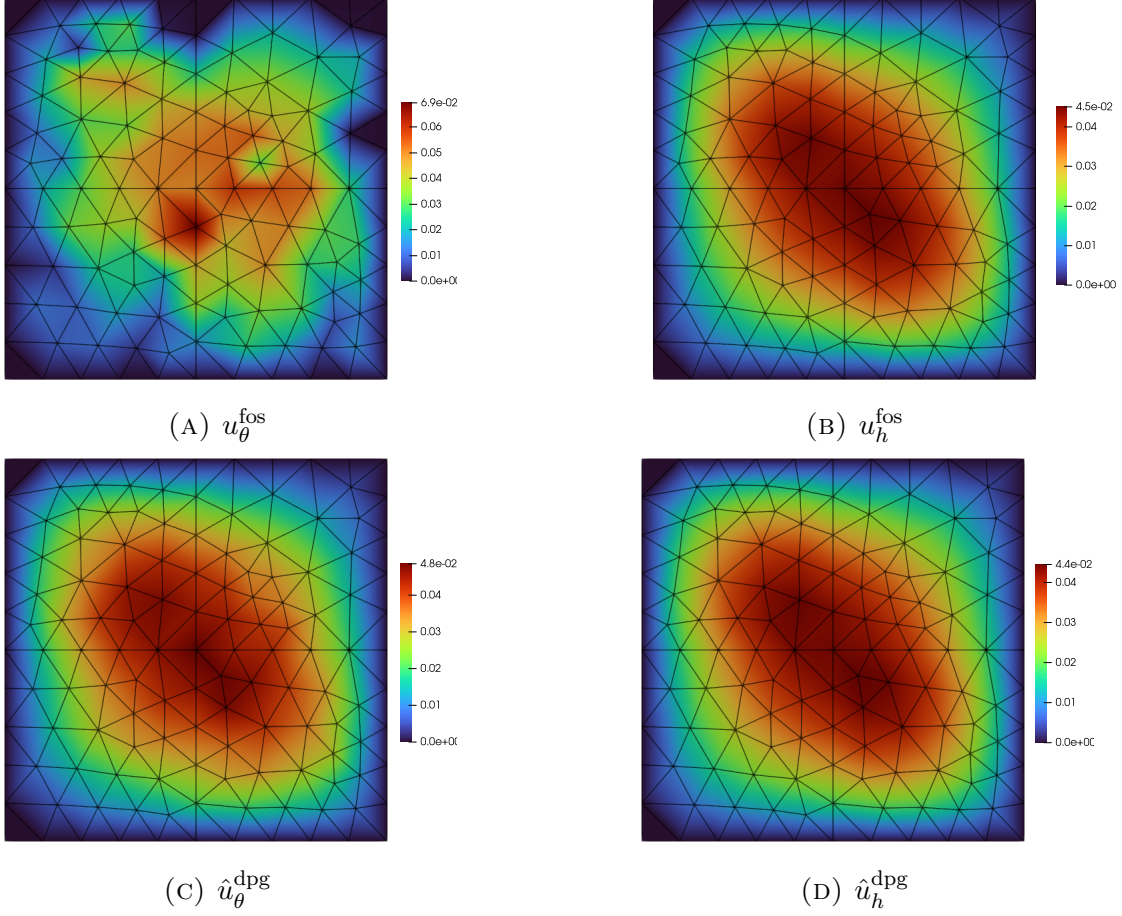


FIGURE 4. Visualization of actual finite element solutions (right) and predictions (left) from an NN trained around the $\bar{\alpha}$ in (56), for the sampled α in (59).

We do so to observe if the predictive capacity of the neural networks decrease when α approaches the tail of the distribution and report this comparison in Figures 4a—4b for FOSLS and Figures 4c—4d for DPG. The NN trained by FOSLS loss produces a prediction which looks poorer than the one produced by the DPG loss. For this α , the finite element solutions produced by both methods (also displayed) are still computed stably, even though the NN predictions vary. (Note that for more difficult α , the finite element solutions from both methods can also visually differ due to stability differences of the methods, but this is not the case in the figure.)

5.2. Performance when varying the mean of α . To obtain a more quantitative measure of performance of NN predictions across the α -distribution, we now tabulate the observed mean of squares of errors from a (relatively large) sample of 10,000 α -values in two experiments.

In our first experiment, we fix α distributions with standard deviation $\sigma = 0.1$ and vary the mean $\bar{\alpha} = (\alpha_1, 1, 1, \alpha_1)$ by choosing α_1 to 100^{-1} , 10^{-1} , 10 , or 100 . With each such choice of mean, an NN was trained with FOSLS loss and another with DPG loss. Predictions from NN, trained using FOSLS loss, are denoted by $(q_{\theta}^{\text{fos}}, u_{\theta}^{\text{fos}})$. They are compared against the actual FOSLS finite element solution $\mathbf{u}_h^{\text{fos}} = (q_h^{\text{fos}}, u_h^{\text{fos}}) \in \mathcal{X}_h^{\text{fos}}$ obtained by solving (38) in the

From α distributions with mean $\bar{\alpha} = (\alpha_1, 1, 1, \alpha_1)$, std.dev. $\sigma = 0.1$					
Mean of error squares	$\alpha_1 = 1/100$	$\alpha_1 = 1/10$	$\alpha_1 = 10$	$\alpha_1 = 100$	
$\ \hat{u}_\theta^{\text{dpg}} - \hat{u}_h^{\text{dpg}}\ _{L_2}^2$ DPG:	2.373e-08	3.136e-08	4.449e-07	5.071e-05	
$\ u_\theta^{\text{fos}} - u_h^{\text{fos}}\ _{L_2}^2$ FOSLS:	1.340e-07	1.671e-08	5.745e-07	4.8261e-02	
$\ \hat{q}_\theta^{\text{dpg}} - \hat{q}_h^{\text{dpg}}\ _{L_2}^2$ DPG:	4.335e-02	1.578e-03	2.146e-04	1.830e-02	
$\ q_\theta^{\text{fos}} - q_h^{\text{fos}}\ _{L_2}^2$ FOSLS:	3.824e-02	4.702e-03	5.265e-05	5.454e-03	

TABLE 1. NN prediction errors as parameter means are varied.

From α distributions with mean $\bar{\alpha} = (0.01, 1, 1, 0.01)$ and std.dev. σ					
Mean of error squares	$\sigma = 0.1$	$\sigma = 0.5$	$\sigma = 1$	$\sigma = 5$	$\sigma = 10$
$\ \hat{u}_\theta^{\text{dpg}} - \hat{u}_h^{\text{dpg}}\ _{L_2}^2$ DPG:	2.373e-08	3.300e-05	2.100e-05	2.335e-03	0.613
$\ u_\theta^{\text{fos}} - u_h^{\text{fos}}\ _{L_2}^2$ FOSLS:	1.340e-07	5.900e-05	1.100e-05	3.284e-02	3.520
$\ \hat{q}_\theta^{\text{dpg}} - \hat{q}_h^{\text{dpg}}\ _{L_2}^2$ DPG:	4.335e-02	3.301e-03	7.866e-03	1.930e-02	0.007
$\ q_\theta^{\text{fos}} - q_h^{\text{fos}}\ _{L_2}^2$ FOSLS:	3.824e-02	6.015e-03	7.209e-03	6.965e-02	0.037

TABLE 2. NN prediction errors as parameter standard deviations are varied

$L_2(\Omega)$ -norm. The results are tabulated in Table 1 together with comparable quantities for the DPG case. Comparable quantities in the DPG solution $\mathbf{u}_h^{\text{dpg}} = (q_h^{\text{dpg}}, u_h^{\text{dpg}}, \hat{u}_h^{\text{dpg}}, \hat{q}_h^{\text{dpg}})$ are the last two components, since $(\hat{q}_h^{\text{dpg}}, \hat{u}_h^{\text{dpg}})$ can be identified to be in the same space where FOSLS solutions belong since they have a unique extension from the element boundaries to the element interior as functions in $\mathcal{X}_h^{\text{fos}}$. Accordingly, we compare, in the $L_2(\Omega)$ -norm, the predictions $(\hat{q}_\theta^{\text{dpg}}, \hat{u}_\theta^{\text{dpg}})$ from the NN (trained using the DPG loss) with the DPG solution $(\hat{q}_h^{\text{dpg}}, \hat{u}_h^{\text{dpg}})$ obtained by solving (46) at each sample point. We reiterate that the error numbers in Table 1 are errors in $\mathbb{X}(\mathcal{Q}_M) = \ell_2(\mathcal{Q}_M; \mathcal{X}^{\text{fos}}), \ell_2(\mathcal{Q}_M; \mathcal{X}^{\text{dpg}})$, respectively (see Section 2.5), i.e., they are the mean of $L_2(\Omega)$ -norms of errors of ten thousand random α samples taken from the above-mentioned distribution for α , even though the training was done only with 1024 samples. Clearly the mean of the errors are small and comparable for both methods.

5.3. Performance when varying the standard deviation of α . In the next experiment, we fix the mean of α and train the NN with various standard deviation choices of σ in $\{0.1, 0.5, 1, 5, 10\}$ using either the DPG loss function or the FOSLS loss function. We compute errors as in the previous table. They are reported in Table 2. Satisfactory performance is again observed for both methods as measured by the mean over ten thousand test samples.

To conclude this section, we highlight an observation from both Tables 1 and 2. We see a slight improvement in the NN prediction error for the the DPG case for extreme values of α_1 in Table 1, namely $\alpha_1 = 100^{-1}$ and 100. Similarly, in Table 2, we see that the DPG error squared means are slightly better for larger standard deviations. This sets the stage for further investigations of parameter robustness of DPG loss functions and further gains that can be extracted from tailored DPG loss functions.

6. IMPROVED PARAMETER ROBUSTNESS USING DPG IDEAS

The proofs of variational correctness of both the FOSLS loss function and the DPG loss function depended on an inequality independent of discretizations, namely (31). Even if this might suggest that it is impossible to remove the α dependence, we are able to ameliorate

the α dependence in the DPG case, as we shall see in this section. As motivation, we refer again to the antecedent of the ultraweak DPG method given in Example 2.3. There, the optimal test norm of (17), $\|A_\alpha^* \mathbf{v}\|_{L_2}$ for any \mathbf{v} in the (unbroken) graph space yielded an isometry of the operator induced by the corresponding weak formulation, which yielded an attractive characterization of error in the α -independent L_2 -norm in (18) with no α -dependent constants. When moving to the DPG setting, in the broken graph norm of (44), we are forced to add an L_2 -term to make a norm, namely in $(\|A_\alpha^* \mathbf{v}\|_{L_2}^2 + \|\mathbf{v}\|_{L_2}^2)^{1/2}$, for \mathbf{v} in the broken graph space \mathcal{Y}^{dpg} . Consequently, we no longer have (18) to characterize the error in an α -independent norm. In order to retain as much α -robustness as possible, in this section, we weaken the previous DPG norm “towards” the optimal test norm of (17), and then see to what extent we can modify resulting fiber loss functionals so as to obtain sharp bounds for the L_2 -errors of the bulk terms (interior variables).

6.1. Loss functionals built with a weaker test space norm. Instead of (44), we now endow the test space $\mathcal{Y}^{\text{dpg}} = H(\text{div}, \Omega_h) \times H^1(\Omega_h)$ by the following norm with an additional scaling factor $s > 0$,

$$\|(\tau, \nu)\|_{\mathcal{Y},s}^2 = \sum_{K \in \Omega_h} (\|A_\alpha^*(\tau, \nu)\|_{L_2(K)}^2 + s^{-2} \|(\tau, \nu)\|_{L_2(K)}^2).$$

The corresponding inner product is denoted by $(\cdot, \cdot)_{\mathcal{Y},s}$. The DPG variant using this inner product is defined exactly as in (46) but replacing $(\cdot, \cdot)_{\mathcal{Y}}$ with $(\cdot, \cdot)_{\mathcal{Y},s}$. Such a method was first studied in [25]. Analogous to (47), we define $\varepsilon_s^{\mathfrak{w}} \in \mathcal{Y}^{\text{dpg}}$ by

$$(\varepsilon_s^{\mathfrak{w}}, \mathbf{v})_{\mathcal{Y},s} = (F, \mathbf{v})_{L_2} - b_\alpha^{\text{dpg}}(\mathfrak{w}, \mathbf{v}), \quad \text{for all } \mathbf{v} \in \mathcal{Y}^{\text{dpg}}, \quad (60a)$$

and a new s -dependent fiber loss function by

$$\hat{\mathcal{L}}_s^{\text{dpg}}(\mathfrak{w}) := \|\varepsilon_s^{\mathfrak{w}}\|_{\mathcal{Y},s}^2, \quad \mathfrak{w} \in \mathcal{X}_h^{\text{dpg}}, \quad \alpha \in \mathcal{Q}. \quad (60b)$$

As s increases, the test space norm becomes weaker (approaching (17)) and this new loss function gains improved parameter robustness as shown in Theorem 6.1 below.

The drawback of the loss function $\hat{\mathcal{L}}_s^{\text{dpg}}$ is that it requires us to solve the infinite-dimensional problem (60a) for $\varepsilon_s^{\mathfrak{w}}$. As we shall see below, we are able to get good results when using a computable version obtained by replacing \mathcal{Y}^{dpg} with $\mathcal{Y}_h^{\text{dpg}}$ in (60a), i.e, in place of $\varepsilon_s^{\mathfrak{w}}$, we use $\varepsilon_{h,s}^{\mathfrak{w}} \in \mathcal{Y}_h^{\text{dpg}}$ satisfying

$$(\varepsilon_{h,s}^{\mathfrak{w}}, \mathbf{v})_{\mathcal{Y},s} = (F, \mathbf{v})_{L_2} - b_\alpha^{\text{dpg}}(\mathfrak{w}, \mathbf{v}), \quad \text{for all } \mathbf{v} \in \mathcal{Y}_h^{\text{dpg}}, \quad (61a)$$

and

$$\mathcal{L}_s^{\text{dpg}}(\mathfrak{w}) := \|\varepsilon_{h,s}^{\mathfrak{w}}\|_{\mathcal{Y},s}^2, \quad \mathfrak{w} \in \mathcal{X}_h^{\text{dpg}}. \quad (61b)$$

We will also consider the dual-parameter functional

$$\mathcal{L}_{s_1, s_2}^{\text{dpg}}(\mathfrak{w}) := \frac{s_2^2 \mathcal{L}_{s_1}^{\text{dpg}}(\mathfrak{w}) - s_1^2 \mathcal{L}_{s_2}^{\text{dpg}}(\mathfrak{w})}{s_2^2 - s_1^2} \quad (62)$$

for any two positive parameters $s_1 < s_2$.

6.2. Theoretical motivation. We present a few theoretical observations that motivate the new loss function families defined above.

It will prove convenient to express $\mathcal{X}^{\text{dpg}} = L_2(\Omega)^d \times L_2(\Omega) \times \dot{H}^{1/2}(\partial\Omega_h) \times H^{-1/2}(\partial\Omega_h)$ as a Cartesian product of the space of “interior variables” (or bulk variables) in $\mathcal{X}_0^{\text{dpg}} = L_2(\Omega)^d \times L_2(\Omega)$ and the space of “interface variables” $\hat{\mathcal{X}}^{\text{dpg}} = \dot{H}^{1/2}(\partial\Omega_h) \times H^{-1/2}(\partial\Omega_h)$, i.e., any $\mathbf{w}^{\text{dpg}} \in \mathcal{X}^{\text{dpg}}$ will be written as

$$\mathbf{w}^{\text{dpg}} \in \mathcal{X}^{\text{dpg}} = \mathcal{X}_0^{\text{dpg}} \times \hat{\mathcal{X}}^{\text{dpg}}, \quad \mathbf{w}^{\text{dpg}} = (\mathbf{w}, \hat{\mathbf{w}}), \text{ with } \mathbf{w} \in \mathcal{X}_0^{\text{dpg}}, \hat{\mathbf{w}} \in \hat{\mathcal{X}}^{\text{dpg}}. \quad (63)$$

Any $\hat{\mathbf{w}}$ in $\hat{\mathcal{X}}^{\text{dpg}}$ is obtained as the trace of a function in $\dot{H}^1 \times H(\text{div})$, so the following coset definition makes sense:

$$Z(\hat{\mathbf{w}}) = \{(r, w) \in H(\text{div}) \times \dot{H}^1 : (\text{tr } w, \text{tr}_n r) = \hat{\mathbf{w}}\}.$$

By (31), $\|A_\alpha \cdot\|_{L_2}$ is a norm on $H(\text{div}) \times \dot{H}^1$, allowing us to define a minimal extension operator

$$E\hat{\mathbf{w}} = \underset{(r,w) \in Z(\hat{\mathbf{w}})}{\text{argmin}} \|A_\alpha(r, w)\|_{L_2}.$$

Writing any $\mathbf{w}^{\text{dpg}} \in \mathcal{X}^{\text{dpg}}$ as in (63), we use the above extension to define a trial space norm by

$$\|\mathbf{w}^{\text{dpg}}\|_{s,\alpha}^2 \equiv \|(\mathbf{w}, \hat{\mathbf{w}})\|_{s,\alpha}^2 := \|\mathbf{w}\|_{L_2}^2 + s^2 \|A_\alpha E\hat{\mathbf{w}}\|_{L_2}^2. \quad (64)$$

Noting that $Z(0)$ is a closed subspace of $\mathcal{X}_0^{\text{dpg}} = L_2(\Omega)^d \times L_2(\Omega)$, we also define the Peetre K -functional

$$K(s, \mathbf{w}) = \inf_{\mathbf{z} \in Z(0)} \left(\|\mathbf{w} - \mathbf{z}\|_{L_2}^2 + s^2 \|A_\alpha \mathbf{z}\|_{L_2}^2 \right) \quad (65)$$

defined for any $\mathbf{w} \in \mathcal{X}_0^{\text{dpg}}$. In the next theorem, we also use following quantities for positive numbers s, s_1 , and s_2 :

$$k_{s,\alpha} = \frac{1}{2} \left(\frac{c_\alpha^2}{s^2} + \sqrt{\frac{c_\alpha^4}{s^4} + 4 \frac{c_\alpha^2}{s^2}} \right), \quad (66)$$

$$\hat{\mathcal{L}}_{s_1, s_2}^{\text{dpg}}(\mathbf{w}^{\text{dpg}}) = \frac{s_2^2 \hat{\mathcal{L}}_{s_1}^{\text{dpg}}(\mathbf{w}^{\text{dpg}}) - s_1^2 \hat{\mathcal{L}}_{s_2}^{\text{dpg}}(\mathbf{w}^{\text{dpg}})}{s_2^2 - s_1^2}, \quad (67)$$

$$\hat{\mathcal{L}}_{s_1, s_2}^{\text{dpg}, L}(\mathbf{w}^{\text{dpg}}) = \frac{(1 + k_{s_1, \alpha})^{-1} s_2^2 \hat{\mathcal{L}}_{s_1}^{\text{dpg}}(\mathbf{w}^{\text{dpg}}) - (1 + k_{s_2, \alpha}) s_1^2 \hat{\mathcal{L}}_{s_2}^{\text{dpg}}(\mathbf{w}^{\text{dpg}})}{s_2^2 - s_1^2}, \quad (68)$$

$$\hat{\mathcal{L}}_{s_1, s_2}^{\text{dpg}, R}(\mathbf{w}^{\text{dpg}}) = \frac{(1 + k_{s_1, \alpha}) s_2^2 \hat{\mathcal{L}}_{s_1}^{\text{dpg}}(\mathbf{w}^{\text{dpg}}) - (1 + k_{s_2, \alpha})^{-1} s_1^2 \hat{\mathcal{L}}_{s_2}^{\text{dpg}}(\mathbf{w}^{\text{dpg}})}{s_2^2 - s_1^2}, \quad (69)$$

where c_α is as in (31).

Theorem 6.1. *Let $s > 0$, $0 < s_1 < s_2$, $\alpha \in \mathcal{Q}$, $\mathbf{w} \in \mathcal{X}_0^{\text{dpg}}$, $\hat{\mathbf{w}} \in \hat{\mathcal{X}}^{\text{dpg}}$, and $\mathbf{w}^{\text{dpg}} = (\mathbf{w}, \hat{\mathbf{w}})$. Let $\mathbf{u}^{\text{dpg}} = (\mathbf{u}, \hat{\mathbf{u}}) = (q, u, \hat{u}, \hat{q}_n)$ denote the exact solution of (50) for the arbitrarily chosen $\alpha \in \mathcal{Q}$. Then the following results hold.*

- (1) *The fiber loss functionals $\hat{\mathcal{L}}_s^{\text{dpg}}$ are variationally correct and satisfy the two-sided estimates*

$$(1 + k_{s,\alpha})^{-1} \|\mathbf{u}^{\text{dpg}} - \mathbf{w}^{\text{dpg}}\|_{s,\alpha}^2 \leq \hat{\mathcal{L}}_s^{\text{dpg}}(\mathbf{w}^{\text{dpg}}) \leq (1 + k_{s,\alpha}) \|\mathbf{u}^{\text{dpg}} - \mathbf{w}^{\text{dpg}}\|_{s,\alpha}^2 \quad (70)$$

with the explicit constant $k_{s,\alpha}$ defined in (66).

(2) The loss functional $\hat{\mathcal{L}}_s^{\text{dpg}}$ is related to the error through the K -functional in (65):

$$\hat{\mathcal{L}}_s^{\text{dpg}}(\mathbf{w}^{\text{dpg}}) = s^2 \|A_\alpha E(\hat{\mathbf{u}} - \hat{\mathbf{w}})\|_{L_2}^2 + K(s, (\mathbf{u} - \mathbf{w}) - E(\hat{\mathbf{u}} - \hat{\mathbf{w}})). \quad (71)$$

(3) The functional $\hat{\mathcal{L}}_{s_1, s_2}^{\text{dpg}}$ in (67) is non-negative.

(4) The interior variable error $\mathbf{u} - \mathbf{w}$ satisfies

$$\hat{\mathcal{L}}_{s_1, s_2}^{\text{dpg}, L}(\mathbf{w}^{\text{dpg}}) \leq \|\mathbf{u} - \mathbf{w}\|_{L_2}^2 \leq \hat{\mathcal{L}}_{s_1, s_2}^{\text{dpg}, R}(\mathbf{w}^{\text{dpg}}). \quad (72)$$

(5) For all α admitting (31) with a fixed c_α , as $s_1 \rightarrow \infty$, both the upper and lower bounds above converge to the non-negative functional $\hat{\mathcal{L}}_{s_1, s_2}^{\text{dpg}}$,

$$\hat{\mathcal{L}}_{s_1, s_2}^{\text{dpg}, L}(\mathbf{w}^{\text{dpg}}) \rightarrow \hat{\mathcal{L}}_{s_1, s_2}^{\text{dpg}}(\mathbf{w}^{\text{dpg}}), \quad \hat{\mathcal{L}}_{s_1, s_2}^{\text{dpg}, R}(\mathbf{w}^{\text{dpg}}) \rightarrow \hat{\mathcal{L}}_{s_1, s_2}^{\text{dpg}}(\mathbf{w}^{\text{dpg}}).$$

Before turning to the proof of these somewhat technical facts, a few remarks on how to interpret the results in Theorem 6.1 and explain their relevance are in order. The advantage of the new loss function $\hat{\mathcal{L}}_s^{\text{dpg}}$ is evident from (70) of Theorem 6.1(1). Namely, even when c_α is very large, it is possible to choose a scaling factor s such that $c_\alpha/s \lesssim 1$, thus making the constants in (70) approximately independent of α . This motivates the use of the loss function $\mathcal{L}_s^{\text{dpg}}$ defined in (61) (since $\hat{\mathcal{L}}_s^{\text{dpg}}$ is not computable). Yet, a drawback of (70) is that the error is measured in the α -dependent norm $\|\cdot\|_{s, \alpha}$, incorporating the (stronger) graph-norm through the element boundary terms. The estimates of (72) in Theorem 6.1(4) address this drawback by providing upper and lower bounds of an α -independent norm of the error in the interior variable approximation. There are no unknown constants in those two estimates, but the upper and lower bounds use two distinct functionals $\hat{\mathcal{L}}_{s_1, s_2}^{\text{dpg}, L}$ and $\hat{\mathcal{L}}_{s_1, s_2}^{\text{dpg}, R}$. These two functional values are then proved to converge to the value of the functional $\hat{\mathcal{L}}_{s_1, s_2}^{\text{dpg}}$ in Theorem 6.1(5). Together with the non-negativity of $\hat{\mathcal{L}}_{s_1, s_2}^{\text{dpg}}$ proved in Theorem 6.1(3), this provides a strong rationale for using $\hat{\mathcal{L}}_{s_1, s_2}^{\text{dpg}}$ to obtain an error certificate in an α -independent norm for part of the total error. This part of the error consists of the error in q and error in u and is therefore comparable to the H^1 error, the canonical norm in which error is measured for the usual primal formulation (see Example 2.1). Again, since $\hat{\mathcal{L}}_{s_1, s_2}^{\text{dpg}}$ is not computable, we have resorted to the use of $\mathcal{L}_{s_1, s_2}^{\text{dpg}}$ in (62) in some experiments to follow. Multiple tests, reported in the remainder of this section, show the practical effectiveness of these strategies. Note that if we were to restrict ourselves to only collections of α with a bounded c_α (of (31)), then the argument of the proof of Proposition 4.1 using the same Fortin operator would prove the variational correctness of $\mathcal{L}_s^{\text{dpg}}$ with constants that depend on c_α . It is currently an open question to obtain an analogue of (70) with explicit constants for $\mathcal{L}_s^{\text{dpg}}$ instead of $\hat{\mathcal{L}}_s^{\text{dpg}}$. We now prove each item of Theorem 6.1.

Proof of Theorem 6.1: For any $\mathbf{w}^{\text{dpg}} \in \mathcal{X}^{\text{dpg}}$, let

$$\|\mathbf{w}^{\text{dpg}}\|_{b_\alpha^{\text{dpg}}} = \sup_{0 \neq \mathbf{v} \in \mathcal{Y}^{\text{dpg}}} \frac{|b_\alpha^{\text{dpg}}(\mathbf{w}^{\text{dpg}}, \mathbf{v})|}{\|\mathbf{v}\|_{\mathcal{Y}, s}}.$$

Since b_α^{dpg} is bounded and satisfies the inf-sup condition (see (54)), this is a norm on \mathcal{X}^{dpg} .

Proof of items (1)–(2): By [21, Theorem 7.9, Propositions 7.10–7.11], whenever (31) holds, we have

$$\|\mathbf{w}^{\text{dpg}}\|_{b_\alpha^{\text{dpg}}} = s^2 \|A_\alpha E \hat{\mathbf{w}}\|_{L_2}^2 + K(s, \mathbf{w} - E \hat{\mathbf{w}}), \quad (73)$$

and

$$(1 + k_{s, \alpha})^{-1} \|\mathbf{w}^{\text{dpg}}\|_{s, \alpha}^2 \leq \|\mathbf{w}^{\text{dpg}}\|_{b_\alpha^{\text{dpg}}}^2 \leq (1 + k_{s, \alpha}) \|\mathbf{w}^{\text{dpg}}\|_{s, \alpha}^2 \quad (74)$$

for all $\mathbf{w}^{\text{dpg}} \in \mathcal{X}^{\text{dpg}}$. Now observe that (60) implies

$$\begin{aligned} \|\mathbf{u}^{\text{dpg}} - \mathbf{w}^{\text{dpg}}\|_{b_\alpha^{\text{dpg}}} &= \sup_{0 \neq \mathbf{v} \in \mathcal{Y}^{\text{dpg}}} \frac{|b_\alpha^{\text{dpg}}(\mathbf{u}^{\text{dpg}} - \mathbf{w}^{\text{dpg}}, \mathbf{v})|}{\|\mathbf{v}\|_{\mathcal{Y},s}} = \sup_{0 \neq \mathbf{v} \in \mathcal{Y}^{\text{dpg}}} \frac{|\ell^{\text{dpg}}(\mathbf{v}) - b_\alpha^{\text{dpg}}(\mathbf{w}^{\text{dpg}}, \mathbf{v})|}{\|\mathbf{v}\|_{\mathcal{Y},s}} \\ &= \sup_{0 \neq \mathbf{v} \in \mathcal{Y}^{\text{dpg}}} \frac{|(\varepsilon_s^{\mathbf{w}}, \mathbf{v})_{\mathcal{Y},s}|}{\|\mathbf{v}\|_{\mathcal{Y},s}} = \|\varepsilon_s^{\mathbf{w}}\|_{\mathcal{Y},s} = \hat{\mathcal{L}}_s^{\text{dpg}}(\mathbf{w}^{\text{dpg}}). \end{aligned}$$

Hence (70) and (71) follow from (74) and (73), respectively, with $\mathbf{u}^{\text{dpg}} - \mathbf{w}^{\text{dpg}}$ in place of \mathbf{w}^{dpg} .

Proof of item (3): Since $0 < s_1 < s_2$,

$$s_1^{-2} \|\mathbf{w} - \mathbf{z}\|_{L_2}^2 + \|A_\alpha \mathbf{z}\|_{L_2}^2 \geq s_2^{-2} \|\mathbf{w} - \mathbf{z}\|_{L_2}^2 + \|A_\alpha \mathbf{z}\|_{L_2}^2$$

for any $\mathbf{w} \in \mathcal{X}_0^{\text{dpg}}$ and $\mathbf{z} \in Z(0)$. By (65), this implies

$$\frac{K(s_1, \mathbf{w})}{s_1^2} \geq \frac{K(s_2, \mathbf{w})}{s_2^2} \quad (75)$$

for any $\mathbf{w} \in \mathcal{X}_0^{\text{dpg}}$. Let $\mathbf{w}' = \mathbf{u} - \mathbf{w} - E(\hat{\mathbf{u}} - \hat{\mathbf{w}})$ in $\mathcal{X}_0^{\text{dpg}}$. By (71),

$$\frac{\hat{\mathcal{L}}_{s_1}^{\text{dpg}}(\mathbf{w}^{\text{dpg}})}{s_1^2} = \|A_\alpha E(\hat{\mathbf{u}} - \hat{\mathbf{w}})\|_{L_2}^2 + \frac{K(s_1, \mathbf{w}')}{s_1^2}.$$

The same equality holds with s_2 also. Subtracting,

$$\frac{\hat{\mathcal{L}}_{s_1}^{\text{dpg}}(\mathbf{w}^{\text{dpg}})}{s_1^2} - \frac{\hat{\mathcal{L}}_{s_2}^{\text{dpg}}(\mathbf{w}^{\text{dpg}})}{s_2^2} = \frac{K(s_1, \mathbf{w}')}{s_1^2} - \frac{K(s_2, \mathbf{w}')}{s_2^2},$$

which is nonnegative by (75). Hence $\hat{\mathcal{L}}_{s_1, s_2}^{\text{dpg}}(\mathbf{w}^{\text{dpg}}) \geq 0$.

Proof of item (4): Let $e_0 = \|\mathbf{u} - \mathbf{w}\|_{L_2}$, $\hat{e} = \|A_\alpha(\hat{\mathbf{u}} - \hat{\mathbf{w}})\|_{L_2}$, and $l_i = \hat{\mathcal{L}}_{s_i}^{\text{dpg}}(\mathbf{w}^{\text{dpg}})$. Then by (70),

$$\begin{aligned} \frac{e_0^2}{s_1^2} + \hat{e}^2 &\leq (1 + k_{s_1, \alpha}) \frac{l_1}{s_1^2}, \\ \frac{e_0^2}{s_2^2} + \hat{e}^2 &\geq (1 + k_{s_2, \alpha})^{-1} \frac{l_2}{s_2^2}. \end{aligned}$$

Subtracting,

$$e_0^2 \left(\frac{1}{s_1^2} - \frac{1}{s_2^2} \right) \leq \frac{(1 + k_{s_1, \alpha}) l_1}{s_1^2} - \frac{(1 + k_{s_2, \alpha})^{-1} l_2}{s_2^2}.$$

Rearranging, we obtain

$$e_0^2 \leq \hat{\mathcal{L}}_{s_1, s_2}^{\text{dpg}, R}(\mathbf{w}^{\text{dpg}}),$$

which is the upper inequality of (72). The lower bound of (72) is proved similarly.

Proof of item (5): Holding c_α fixed, we see from the definition of $k_{s, \alpha}$ in (66) that

$$\lim_{s \rightarrow \infty} k_{s, \alpha} = 0.$$

Hence the expressions for $\hat{\mathcal{L}}_{s_1, s_2}^{\text{dpg}, L}(\mathbf{w}^{\text{dpg}})$ and $\hat{\mathcal{L}}_{s_1, s_2}^{\text{dpg}, R}(\mathbf{w}^{\text{dpg}})$ in (68)–(69) converge to the expression for $\hat{\mathcal{L}}_{s_1, s_2}^{\text{dpg}, L}(\mathbf{w}^{\text{dpg}})$ in (67). \square

6.3. Robustness of the s -based DPG loss function. We train an NN (with the same architecture and other parameters mentioned previously) using the new DPG fiber loss function $\mathcal{L}_s^{\text{dpg}}$ (in the correspondingly modified mean squared loss (55)). The NN prediction for any given parameter α now consists of an interior solution prediction and interface solution prediction, which are respectively denoted by $\mathbf{u}_{\theta,s}^{\text{dpg}}(\alpha)$ and $\hat{\mathbf{u}}_{\theta,s}^{\text{dpg}}(\alpha)$. The corresponding exact finite element solutions are denoted by $\mathbf{u}_{h,s}^{\text{dpg}}(\alpha)$ and $\hat{\mathbf{u}}_{h,s}^{\text{dpg}}(\alpha)$, respectively. Notice that when $s = 1$, the loss function coincides with the previous DPG loss \mathcal{L}^{dpg} from (47), a case we have already discussed at length. The $s = 1$ case, the prior FOSLS solution $\mathbf{u}_h^{\text{fos}}(\alpha)$ from (38), and its NN prediction $\mathbf{u}_\theta^{\text{fos}}(\alpha)$ (previously seen in (57)), will offer comparators as we now study variations with s .

We examine the following error measures of interest:

$$\begin{aligned} \hat{e}^{\text{fos}}(\alpha) &:= \|A_\alpha(\mathbf{u}_\theta^{\text{fos}}(\alpha) - \mathbf{u}_h^{\text{fos}}(\alpha))\|_{L_2}^2, & e_0^{\text{fos}}(\alpha) &:= \|\mathbf{u}_\theta^{\text{fos}}(\alpha) - \mathbf{u}_h^{\text{fos}}(\alpha)\|_{L_2}^2, \\ \hat{e}_s^{\text{dpg}}(\alpha) &:= \|A_\alpha(\hat{\mathbf{u}}_{\theta,s}^{\text{dpg}}(\alpha) - \hat{\mathbf{u}}_{h,s}^{\text{dpg}}(\alpha))\|_{L_2}^2, & e_{0,s}^{\text{dpg}}(\alpha) &:= \|\mathbf{u}_{\theta,s}^{\text{dpg}}(\alpha) - \mathbf{u}_{h,s}^{\text{dpg}}(\alpha)\|_{L_2}^2. \end{aligned} \quad (76)$$

We use these quantities (not in any learning algorithm but) to study the effectiveness and α -robustness of the computable loss functionals to estimate the error in NN predictions. We drop the α -argument when it is understood from context. The exact solution \mathbf{u} is unavailable and is absent from the computable errors in (76). Nonetheless, we see, by means of the triangle inequality, that the FOSLS loss $\mathcal{L}^{\text{fos}}(\mathbf{w}) = \|A_\alpha \mathbf{w} - F\|_{L_2}^2 = \|A_\alpha(\mathbf{w} - \mathbf{u})\|_{L_2}^2$ bounds the first quantity \hat{e}^{fos} in (76) as follows:

$$\hat{e}^{\text{fos}} = \|A_\alpha(\mathbf{u}_\theta^{\text{fos}} - \mathbf{u}_h^{\text{fos}})\|_{L_2}^2 \leq 2[\mathcal{L}^{\text{fos}}(\mathbf{u}_\theta^{\text{fos}}) + \mathcal{L}^{\text{fos}}(\mathbf{u}_h^{\text{fos}})]. \quad (77)$$

Since both sides of this inequality are computable once we have computed FOSLS solutions on chosen test sets in \mathcal{Q} , we can use it to check the reliability of the loss function. For an analogous measure for the DPG s -dependent loss, recall the $\|\cdot\|_{s,\alpha}$ -norm introduced in (64) and observe that

$$\|\hat{\mathbf{u}}_{\theta,s}^{\text{dpg}} - \hat{\mathbf{u}}_{h,s}^{\text{dpg}}\|_{s,\alpha}^2 = e_{0,s}^{\text{dpg}} + s^2 \hat{e}_s^{\text{dpg}}.$$

Then, (70) of Theorem 6.1 and a new application of the triangle inequality gives

$$e_{0,s}^{\text{dpg}} + s^2 \hat{e}_s^{\text{dpg}} \leq 2(1 + k_{s,\alpha}) \left[\hat{\mathcal{L}}_s^{\text{dpg}}(\mathbf{u}_{\theta,s}^{\text{dpg}}) + \hat{\mathcal{L}}_s^{\text{dpg}}(\mathbf{u}_{h,s}^{\text{dpg}}) \right]. \quad (78)$$

Thus, just as in the FOSLS case (77), once we have computed DPG solutions for test sets in \mathcal{Q} , both $e_{0,s}^{\text{dpg}} + s^2 \hat{e}_s^{\text{dpg}}$ and the loss functional values in (78) can be computed without knowing the exact solution.

Recall that training on mean-squared losses like (55) approximates the solution $u(x, \alpha)$ as a function of x and α *in expectation*, hence in a mean-squared sense. This does not imply beforehand any worst-case accuracy control over \mathcal{Q} . On the other hand, it is well known that under the given assumptions the parameter-to-solution map is even *holomorphic*, despite the fact that for each $\alpha \in \mathcal{Q}$ the solution as a function of x may have very low regularity. This suggests that worst-case errors and the mean-squared errors with respect to α may not be that different. Subsequent experiments are to shed some light on this issue as well.

With these considerations in mind, we propose the following experiment. Let $\mathcal{Q}_M^{\bar{\alpha}, \sigma}$ be a collection of α samples of size M drawn from the same distribution described in Section 5 with mean $\bar{\alpha}$ and standard deviation σ . Let $\mathcal{Q}_m^{\bar{\alpha}, \sigma}$ denote its subset of the first $m \leq M$ samples. For any function $e(\alpha)$ over the samples, we denote its *cumulative maximum* function by

$$(\text{cmax } e)(m) = \max_{\alpha \in \mathcal{Q}_m^{\bar{\alpha}, \sigma}} e(\alpha), \quad m = 1, 2, \dots, M,$$

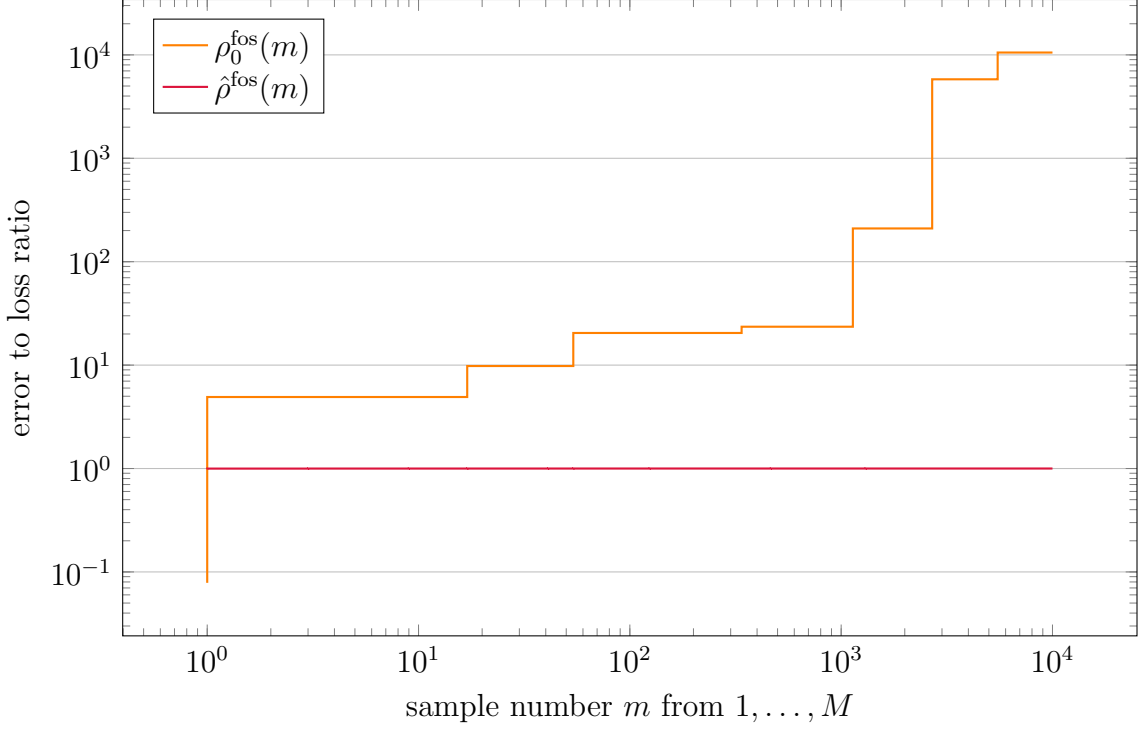


FIGURE 5. Cumulative maxima of FOSLS ratios of error to loss from a random collection of $M = 10,000$ samples of α .

i.e., $(\text{cmax } e)(m) = \|e\|_{\ell_\infty(\mathcal{Q}_m^{\bar{\alpha}, \sigma})}$. Now, turning to (77), we are motivated to define

$$\hat{\rho}^{\text{fos}} := \text{cmax} \left(\frac{\hat{e}^{\text{fos}}}{\mathcal{L}^{\text{fos}}(\mathbf{u}_\theta^{\text{fos}}) + \mathcal{L}^{\text{fos}}(\mathbf{u}_h^{\text{fos}})} \right)$$

a quantity which cannot be greater than two in view of (77). Similarly, (78) motivates the definition of

$$\rho_s^{\text{dpg}} := \text{cmax} \left(\frac{e_{0,s}^{\text{dpg}} + s^2 \hat{e}_s^{\text{dpg}}}{\mathcal{L}_s^{\text{dpg}}(\hat{\mathbf{u}}_{\theta,s}^{\text{dpg}}) + \mathcal{L}_s^{\text{dpg}}(\hat{\mathbf{u}}_{h,s}^{\text{dpg}})} \right). \quad (79)$$

If we had used $\hat{\mathcal{L}}_s^{\text{dpg}}$ instead of $\mathcal{L}_s^{\text{dpg}}$, then in view of (78), the right hand side above would have been bounded by $2 \text{cmax}(1 + k_{s,\alpha})$, which for large s approaches the value of 2 in view of (66). We are interested in observing if this is the case in practice for the computable version $\rho_{M,s}^{\text{dpg}}$. For completeness and to facilitate comparisons, we also introduce the following analogous quantities for FOSLS approximations,

$$\rho_0^{\text{fos}} := \text{cmax} \left(\frac{e_0^{\text{fos}}}{\mathcal{L}^{\text{fos}}(\mathbf{u}_\theta^{\text{fos}}) + \mathcal{L}^{\text{fos}}(\mathbf{u}_h^{\text{fos}})} \right), \quad \rho^{\text{fos}} := \text{cmax} \left(\frac{e_0^{\text{fos}} + \hat{e}^{\text{fos}}}{\mathcal{L}^{\text{fos}}(\mathbf{u}_\theta^{\text{fos}}) + \mathcal{L}^{\text{fos}}(\mathbf{u}_h^{\text{fos}})} \right).$$

We report these quantities observed from our experiment as we draw the random samples of α .

Each experiment drawing random samples of α forms the set $\mathcal{Q}_M^{\bar{\alpha}, \sigma}$ with $M = 10000$, $\bar{\alpha} = (10^{-1}, 1, 1, 10^{-1})$ and $\sigma = 0.5$. To establish baseline cases, we begin with the FOSLS measures ρ^{fos} and $\hat{\rho}^{\text{fos}}$. The latter is bounded by two as seen above, whereas the theory does not provide any α -independent bound for the former. The experimental results in Figure 5

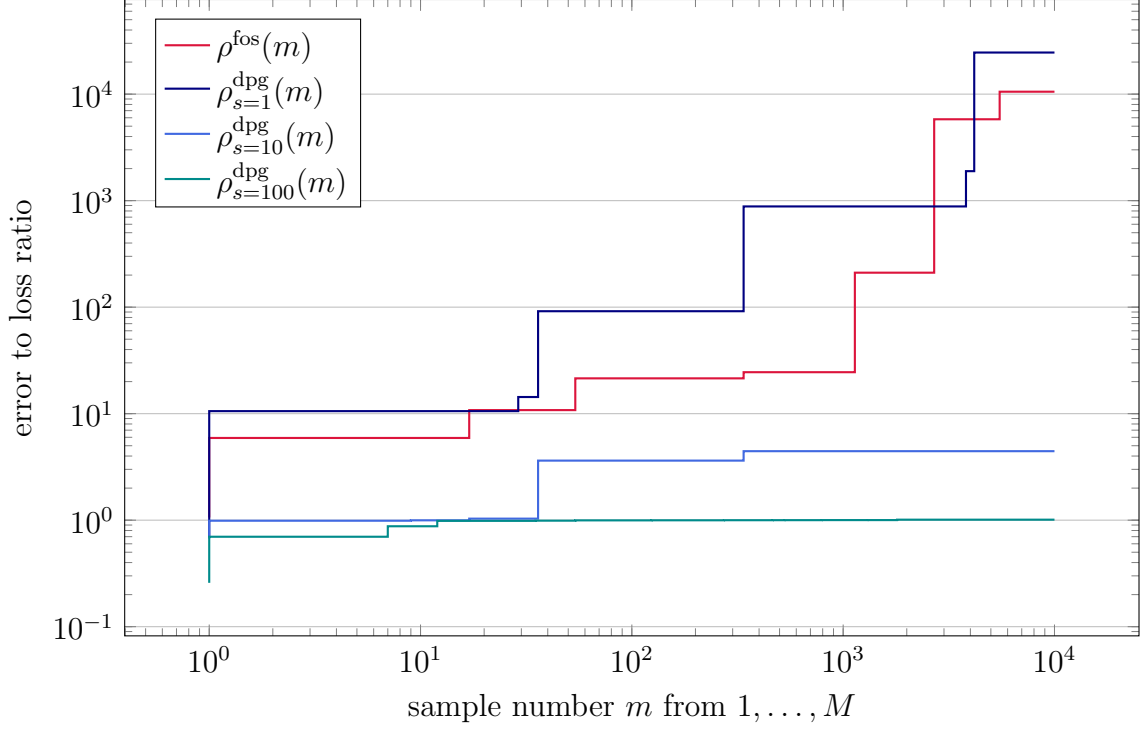


FIGURE 6. Cumulative maxima of DPG ratios of error to loss from a random collection of $M = 10,000$ samples of α .

are in agreement: they show that $\hat{\rho}^{\text{fos}}$ remains bounded across samples while ρ^{fos} grows as the random choices pick up samples of α from the distribution tail. In other words, the ratios of squares of error norms to loss values do not appear to be bounded robustly in α for FOSLS when the norm contains the α -independent L_2 -norm of the error, i.e., L_2 errors appear to suffer most from a worsening conditioning of the FOSLS operator.

In the s -dependent DPG case, we observe better robustness for high values of s . Plots of ρ_s^{dpg} for $s = 1, 10$, and 100 are shown in Figure 6 (together with the baseline plot of ρ^{fos} from Figure 5). As mentioned previously, we are interested in seeing if these ratios would be bounded, in practice, by $2\text{cmax}(1 + k_{s,\alpha})$, a value which approaches two as $s \rightarrow \infty$ for bounded ranges of α . We see in Figure 6 that the graph of ρ_s^{dpg} remains flat with values close to one when $s = 100$, while for lower values of s , it climbs as extreme values of α are picked up on repeated sampling. Of course, we expect there are values of α which will make even $s = 100$ insufficient to obtain a ratio close to one. They may be picked up upon even further sampling. Nonetheless, Figure 6 conveys the practical utility of larger values of s .

Conditioning and numerical stability will limit us from increasing s unboundedly. Indeed, the loss computation requires the calculation of $\varepsilon_{h,s}^{\text{w}} \in \mathcal{Y}_h^{\text{dpg}}$, which involves the inversion of a block diagonal matrix whose condition number can deteriorate as s increases. If a range of α is known, then a practical strategy would be to find an s -value that works for that range and address the numerical stability issues for that s value. Such α -ranges may be available for specific applications like porous media flow, but since this is application-specific, we do not pursue it here. Instead, we proceed to discuss a general strategy that can help us identify appropriate s -values when an α -range of interest is available.

6.4. Training a DPG NN with s and α . The ultraweak DPG setting allows the possibility of tuning the test norm using the scaling factor s . This opens the possibility of letting s be yet another random parameter while training neural networks. Having such an NN allows one to guess s values that may be appropriate for an identified range of α values of interest.

To execute this idea in practice, we consider a new NN trained using the scaled DPG fiber loss function

$$\mathcal{L}_{s,\alpha} = \frac{1}{s^2} \mathcal{L}_s^{\text{dpg}}. \quad (80)$$

The scaling by $1/s^2$ in (80) is motivated by (70) of Theorem 6.1(1), which implies that for any $\mathbf{w}^{\text{dpg}} = (\mathbf{w}, \hat{\mathbf{w}}) \in \mathcal{X}^{\text{dpg}}$,

$$(1 + k_{s,\alpha})^{-1} \frac{1}{s^2} \hat{\mathcal{L}}_s^{\text{dpg}}(\mathbf{w}^{\text{dpg}}) \leq \frac{1}{s^2} \|\mathbf{u} - \mathbf{w}\|_{L_2}^2 + \|A_\alpha E(\hat{\mathbf{u}} - \hat{\mathbf{w}})\|_{L_2}^2 \leq (1 + k_{s,\alpha}) \frac{1}{s^2} \hat{\mathcal{L}}_s^{\text{dpg}}(\mathbf{w}^{\text{dpg}}).$$

Without the $1/s^2$ scaling, the α -dependent part of the error norm, namely $\|A_\alpha E(\hat{\mathbf{u}} - \hat{\mathbf{w}})\|_{L_2}^2$ would get amplified by s^2 , so for large s , the loss $\hat{\mathcal{L}}_s^{\text{dpg}}(\mathbf{w}^{\text{dpg}})$ become extremely large and less sensitive to the α -independent error norm $\|\mathbf{u} - \mathbf{w}\|_{L_2}^2$.

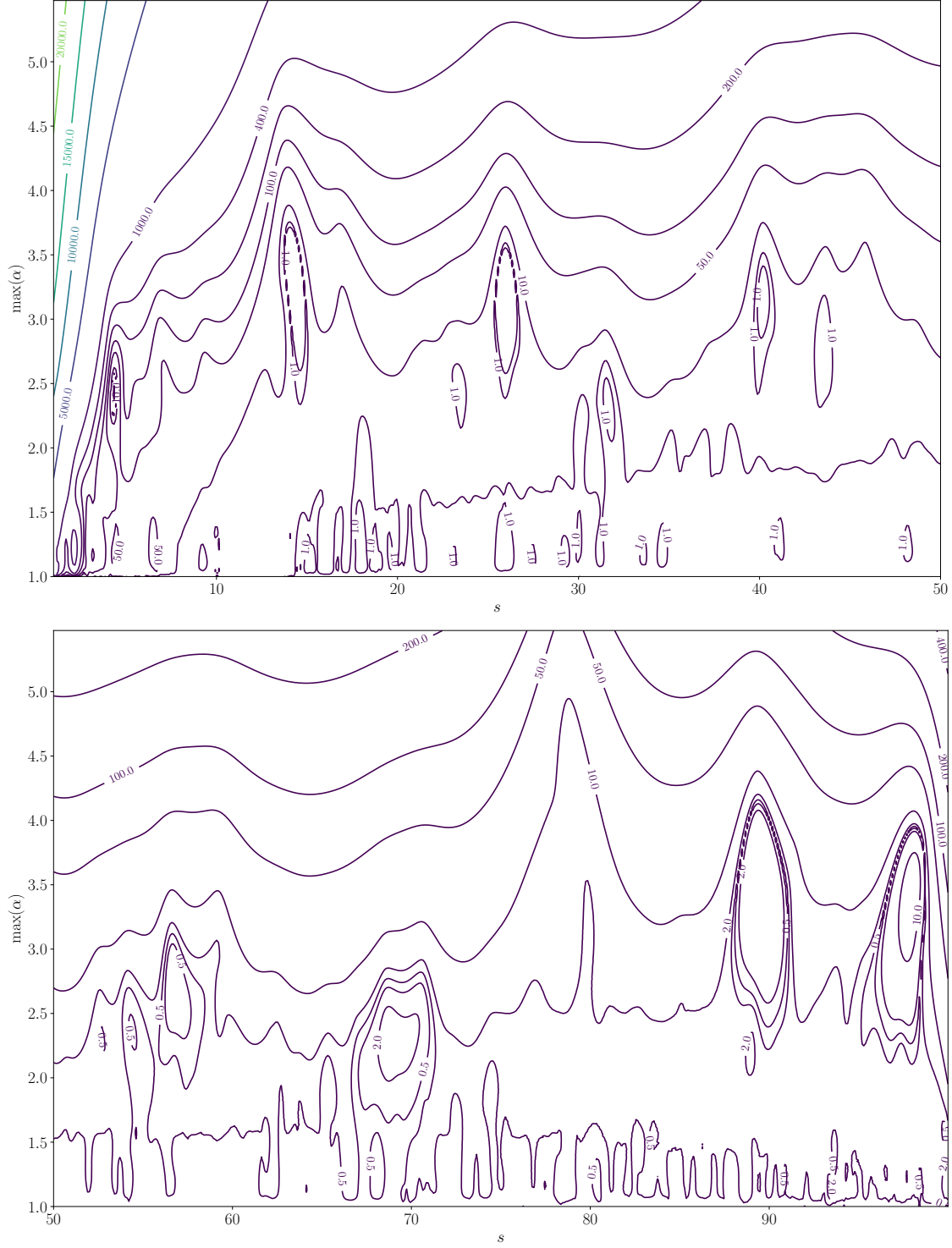
The training process is carried out using samples of (α, s) . We select the distribution for the random variable (α, s) to be the Cartesian product of the α -distribution derived from the normal distribution as described in Section 5, and the uniform distribution of s -values in the interval $[c, d]$. Let $\mathcal{Q}_M^{\bar{\alpha}, \sigma, [c, d]}$ denote a collection of M samples of (α, s) drawn from this distribution. The training process is performed over $M = 1024$ samples and 5000 epochs at the learning rate of 0.001. The parameters of the random distribution are set by

$$\bar{\alpha} = (10^{-1}, 1, 1, 10^{-1}), \quad \sigma = 0.5, \quad c = 1, \quad d = 100, \quad (81)$$

and the NN is trained with the loss (55) where the fiber loss is replaced by $\mathcal{L}_{s,\alpha}$ from (80). We can use the NN to obtain solution predictions as before. But now it is more interesting to use it to see the loss landscape as a function of s and α . By drawing 10,000 samples of (α, s) with α of the form $(\alpha_1, 1, 1, \alpha_1)$ and computing $\mathcal{L}_{s,\alpha}$ of the NN solution prediction for each sample, we obtained the level sets in Figure 7. We observe from the figure that for small s -values, the loss increases steeply with the maximal α -value. However, for larger s -values, smaller losses are obtained for the same range of maximal α .

To conclude this subsection, we have seen that the new loss function in (80) and its level sets in Figure 7 can be used as a guide to determine the smallest s -values that give low losses for selected maximal α -ranges.

6.5. NN trained with two-parameter DPG loss. Recall Example 2.3 which suggested the possibility of obtaining sharp error control in the α -independent L_2 -norm (hence still controlling the solution of (4) in H^1). The extent to which this might be realizable for the DPG interior variables is suggested by Theorem 6.1(4)–(5). In light of these results, we approach this issue by training an NN using the loss (55) with the two-parameter fiber loss functions $\mathcal{L}_{s_1, s_2}^{\text{dpg}}$ introduced in (62). If we have an α data set range, its maximal and minimal ranges can be used to determine an appropriate $s_1 > 0$ and then set $s_2 > s_1$. This can be appreciated in Figure 7, where for a fixed collection of α values we see how the loss function surface flattens as s increases. So, intuitively, one may train $\mathcal{L}_{s_1, s_2}^{\text{dpg}}$ over the box $[\min(\alpha), \max(\alpha)] \times [s_1, s_2]$. However, in this subsection, to keep the tests challenging, we will continue to use the unbounded α distribution as before and fix $s_1 = 50, s_2 = 100$. The

FIGURE 7. $\mathcal{L}_{s,\alpha}$ level sets

associated NN is trained setting

$$\bar{\alpha} = (10^{-1}, 1, 1, 10^{-1}), \quad \sigma = 0.5,$$

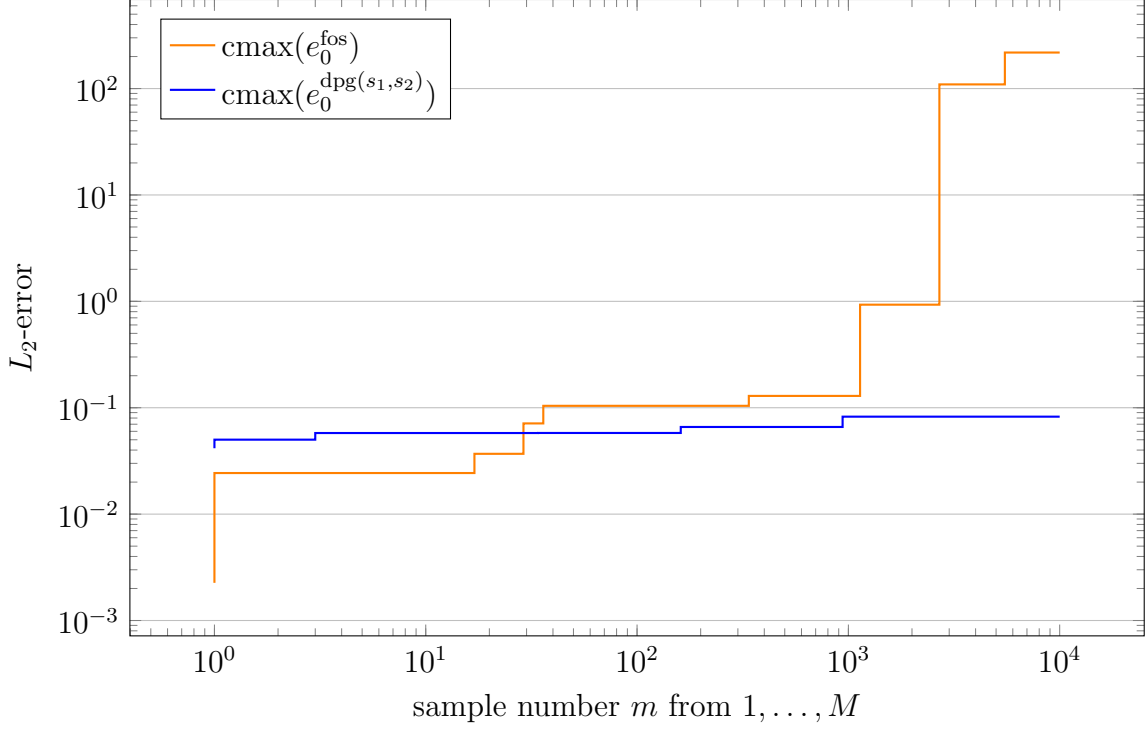


FIGURE 8. Cumulative maxima of α -independent-error-norms obtained using $\mathcal{L}_{s_1, s_2}^{\text{dpg}}$ with $s_1 = 50, s_2 = 100$ and the FOSLS loss over $M = 10,000$ samples of α .

with a learning rate 0.001, 5000 epochs, $M = 1024$ training samples, and a batch size of 32 samples.

To test the performance of the trained NN, we compute for each sampled α the DPG finite element solution for a fixed $s = 1$, take its interior part in $\mathcal{X}_0^{\text{dpg}}$, denoting it by $\mathbf{u}_h^{\text{dpg}(s=1)}$, and compare it with the NN prediction for the same α , denoted by $\mathbf{u}_\theta^{\text{dpg}(s_1, s_2)}$, obtained by substituting the fiber losses $\mathcal{L}_{s_1, s_2}^{\text{dpg}}$ into the lifted loss (55). The comparison is done using the α -independent L_2 error norm

$$e_0^{\text{dpg}(s_1, s_2)} := \|\mathbf{u}_\theta^{\text{dpg}(s_1, s_2)} - \mathbf{u}_h^{\text{dpg}(s=1)}\|_{L_2}^2,$$

which we view as a function of α . This error is compared against the analogous FOSLS error e_0^{fos} introduced before in (76). Results from an experiment carried out over 10000 random samples of α , taken from the previous distribution, are displayed in Figure 8. Clearly, the DPG error in the α -independent norm shows α -robust performance, while the other does not.

ACKNOWLEDGMENTS

This work was supported in part by the NSF under an FRG grant (DMS-2245077, DMS-2245097) and DMS-2012469. The work also benefited from the activities organized under the NSF RTG program (grants DMS-2136228 and DMS-2038080). The authors also acknowledge

funding by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation)—project number 442047500—through the Collaborative Research Center “Sparsity and Singular Structures” (SFB 1481).

REFERENCES

- [1] Mark Ainsworth and Justin Dong. Extended Galerkin neural network approximation of singular variational problems with error control. *SIAM Journal on Scientific Computing*, 47(3):C738–C768, 2025.
- [2] Mark Ainsworth and J. Tinsley Oden. *A Posteriori Error Estimation in Finite Element Analysis*. Wiley, 2000.
- [3] Markus Bachmayr, Wolfgang Dahmen, and Mathias Oster. Variationally correct neural residual regression for parametric pdes: On the viability of controlled accuracy, 2025 (to appear).
- [4] Andrew Barron, Albert Cohen, Wolfgang Dahmen, and Ronald DeVore. Approximation and learning by greedy algorithms. *Annals of Statistics*, 3(1):64–94, 2008.
- [5] S. Berrone, C. Canuto, and M. Pintore. Variational physics informed neural networks: the role of quadratures and test functions. *Journal of Scientific Computing*, 92(3):100, 2022.
- [6] S. Bertoluzza, E. Burman, and C. He. WAN discretization of PDEs: Best approximation, stabilization and essential boundary conditions, 2024.
- [7] K. Bhattacharya, B. Hosseini, N. Kovachki, and A. Stuart. Model reduction and neural networks for parametric pdes. *SMAI-JCM: SMAI Journal of Computational Mathematics*, 7:121–157, 2021.
- [8] K. Bhattacharya, B. Liu, A. Stuart, and M. Trautner. Learning Markovian homogenized models in viscoelasticity. *Multiscale Model. Simul.*, 21(2), 2023.
- [9] Pavel B. Bochev and Max D. Gunzburger. *Least-squares finite element methods*, volume 166 of *Applied Mathematical Sciences*. Springer, New York, 2009.
- [10] Andrea Bonito, Ronald DeVore, Guergana Petrova, and Jonathan W Siegel. Convergence and error control of consistent PINNs for elliptic PDEs. *IMA Journal of Numerical Analysis*, page draf008, 03 2025.
- [11] Franco Brezzi and Michel Fortin. *Mixed and Hybrid Finite Element Methods*. Number 15 in Springer Series in Computational Mathematics. Springer-Verlag, 1991.
- [12] Z. Cai, R. Lazarov, T. A. Manteuffel, and S. F. McCormick. First-order system least squares for second-order partial differential equations. I. *SIAM J. Numer. Anal.*, 31(6):1785–1799, 1994.
- [13] C. Carstensen, L. Demkowicz, and J. Gopalakrishnan. Breaking spaces and forms for the DPG method and applications including Maxwell equations. *Computers and Mathematics with Applications*, 72(3):494–522, August 2016.
- [14] Carsten Carstensen, Leszek Demkowicz, and Jay Gopalakrishnan. A posteriori error control for DPG methods. *SIAM J Numer. Anal.*, 52(3):1335–1353, 2014.
- [15] F. Chen, J. Huang, C. Wang, and H. Yang. Friedrichs learning: Weak solutions of partial differential equations via deep learning. *SIAM Journal on Scientific Computing*, 45(3):A1271–A1299, 2023.
- [16] Albert Cohen, Wolfgang Dahmen, Matthieu Dolbeault, and Augustin Somacal. Reduced order modeling for elliptic problems with high contrast diffusion coefficients. *ESAIM: M2AN*, 57:2775–2802, 2023.
- [17] W. Dahmen, C. Huang, C. Schwab, and G. Welper. Adaptive Petrov–Galerkin methods for first order transport equations. *SIAM Journal on Numerical Analysis*, 50(5):2420–2445, 2012.
- [18] N. Dal Santo, S. Deparis, and L. Pegolotti. Data driven approximation of parametrized PDEs by reduced basis and neural networks. *Journal of Computational Physics*, 416:109550, 2020.
- [19] L. Demkowicz and J. Gopalakrishnan. Analysis of the DPG method for the Poisson equation. *SIAM J Numer. Anal.*, 49(5):1788–1809, 2011.
- [20] L. Demkowicz and J. Gopalakrishnan. A class of discontinuous Petrov–Galerkin methods. Part II: Optimal test functions. *Numerical Methods for Partial Differential Equations*, 27(1):70–105, 2011.
- [21] L. Demkowicz and J. Gopalakrishnan. The discontinuous Petrov–Galerkin method. *Acta Numerica*, 2025 (in press).
- [22] Z. Dongkun, L. Lu, L. Guo, and G. Karniadakis. Quantifying total uncertainty in physics-informed neural networks for solving forward and inverse stochastic problems. *Journal of Computational Physics*, 397:108850, 2019.

- [23] L. Ernst and K. Urban. A certified wavelet-based physics-informed neural network for the solution of parameterized partial differential equations. *ArXiv preprint:arXiv:2212.08389v1 [math.NA]* Dec. 16, 2022, 2022.
- [24] Moritz Geist, Philipp Petersen, Mones Raslan, Reinhold Schneider, and Gitta Kutyniok. Numerical solution of the parametric diffusion equation by deep neural networks. *Journal of Scientific Computing*, 88(1):22, 2021.
- [25] Jay Gopalakrishnan, Ignacio Muga, and Nicole Olivares. Dispersive and dissipative errors in the DPG method with scaled norms for the Helmholtz equation. *SIAM J. Sci. Comput.*, 36(1):A20–A39, 2014.
- [26] Cosmas Hei, Ingo Ghring, and Martin Eigel. Multilevel cnns for parametric pdes. *Journal of Machine Learning Research*, 24(373):1–42, 2023.
- [27] B. Kaushik, B. Hosseini, N. Kovachki, and A. Stuart. Model reduction and neural networks for parametric PDEs. *The SMAI Journal of computational mathematics*, 7:121–157, 2021.
- [28] E. Kharazmi, Z. Zhang, and G. Karniadakis. Variational physics-informed neural networks for solving partial differential equations, 2019.
- [29] N. Kovachki, Z. Li, B. Liu, K. Azizzadenesheli, K. Bhattacharya, A. Stuart, and A. Anandkumar. Neural operator: Learning maps between function spaces with applications to pdes. *Journal of Machine Learning Research*, 24:1–97, 2023.
- [30] Z. Li, N. Kovachki, K. Azizzadenesheli, B. Liu, K. Bhattacharya, A. Stuart, and A. Anandkumar. Fourier neural operator for parametric partial differential equations. In *International Conference on Learning Representations*, 2021.
- [31] Z. Li, N. Kovachki, K. Azizzadenesheli, B. Liu, K. Bhattacharya, A. Stuart, and A. Anandkumar. Fourier neural operator for parametric partial differential equations. 2021.
- [32] A. V. Muzalevsky and S. I. Repin. A posteriori error control of approximate solutions to boundary value problems constructed by neural networks. *Zap. Nauchn. Sem. S.-Peterburg. Otdel. Mat. Inst. Steklov. (POMI)*, 499:77–104, 2021.
- [33] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. *PyTorch: an imperative style, high-performance deep learning library*. Curran Associates Inc., Red Hook, NY, USA, 2019.
- [34] M. Raissi, P. Perdikaris, and G.E. Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378:686–707, 2019.
- [35] P.-A. Raviart and J. M. Thomas. A mixed finite element method for 2nd order elliptic problems. In *Mathematical aspects of finite element methods (Proc. Conf., Consiglio Naz. delle Ricerche (C.N.R.), Rome, 1975)*, pages 292–315. Lecture Notes in Math., Vol. 606. Springer, Berlin, 1977.
- [36] T. De Ryck and S. Mishra. Error analysis for physics-informed neural networks (PINNs) approximating Kolmogorov PDEs. *Advances in Computational Mathematics*, 48(6):79, 2022.
- [37] Joachim Schberl et al. NGSolve. <http://ngsolve.org>.
- [38] J. Taylor, D. Pardo, and I. Muga. A deep Fourier residual method for solving PDEs using neural networks. *Computer Methods in Applied Mechanics and Engineering*, 405:115850, 2023.
- [39] Y. Zang, G. Bao, X. Ye, and H. Zhou. Weak adversarial networks for high-dimensional partial differential equations. *Journal of Computational Physics*, 411:109409, 2020.
- [40] Marius Zeinhofer, Rami Masri, and Kent-Andr Mardal. A unified framework for the error analysis of physics-informed neural networks. *IMA Journal of Numerical Analysis*, page drae081, 11 2024.
- [41] J. Zitelli, I. Muga, L. Demkowicz, J. Gopalakrishnan, D. Pardo, and V. Calo. A class of discontinuous Petrov-Galerkin methods. Part IV: Wave propagation. *Journal of Computational Physics*, 230:2406–2432, 2011.