

PINNs and GaLS: A Priori Error Estimates for Shallow Physics Informed Neural Networks Applied to Elliptic Problems

U. Zerbinati*

* King Abdullah University Of Science and Technology, Thuwal, SA
(e-mail: umberto.zerbinati@kaust.edu.sa).

Abstract: Physics Informed Neural Networks (PINNs) have recently gained popularity for solving partial differential equations, given the fact they escape the curse of dimensionality. In this paper, we present Physics Informed Neural Networks as an underdetermined point matching collocation method then expose the connection between Galerkin Least Squares (GaLS) and PINNs, to develop an a priori error estimate, in the context of elliptic problems. In particular, techniques that belong to the realm of least squares finite elements and Rademacher complexity analysis are used to obtain the error estimate.

Copyright © 2022 The Authors. This is an open access article under the CC BY-NC-ND license (<https://creativecommons.org/licenses/by-nc-nd/4.0/>)

Keywords: Machine Learning, Least-squares method, Finite Element Analysis, Physics Informed Neural Networks, A Priori Error Estimate

1. INTRODUCTION

A number of algorithms have been recently developed to solve partial differential equations by using deep and shallow neural networks, in order to escape the curse of dimensionality, e.g. Mishra and Molinaro (2021); Karniadakis et al. (2021); Weinan et al. (2017); Gu et al. (2021); Gao and Duan (2017); Weinan et al. (2021); Wojtowysch and Weinan (2020). In particular, three methods are dominating the landscape of numerical solutions of partial differential equations (PDEs) by means of deep learning: the deep Ritz method more information in Yu et al. (2017) and Weinan et al. (2021); the finite neuron method in Xu (2020) and He et al. (2018); and PINNs in Lu et al. (2021) and Karniadakis et al. (2021). Key differences among the three methods are that the first two are based on the variational formulation of the PDE, while the third one is based on the strong formulation of the problem. Physics Informed Neural Networks (PINNs) can be seen as an underdetermined point matching collocation method, as described in Eason (1976) and Bochev and Gunzburger (2009), Section 12.4. In fact the idea underlying PINNs is to minimize the least square residual evaluated at a fixed number of collocation points. Obtaining a discrete least square principle by first discretizing the integral in the energy functional using a quadrature rule is not a new concept. It can be found for example in Jiang (1998). The key difference between PINNs and a standard least square collocation method is that with the latter, one tends to choose a greater number of collocation points compared to the number of basis function in the discrete space, which is not the case for PINNs. Usually a priori error estimates are developed using Rademacher complexity analysis, as shown in Hong et al. (2021) and Shin et al. (2020), however this technique results in underestimating the convergence rate of the method. In order to deal with this problem another approach has been developed, which is more similar to an a posteriori error estimate. This idea is presented in Mishra and Molinaro (2020) and Lye et al. (2021). The aim of this article is to expose the connection between Galerkin

Least Squares (GaLS) and PINNs, and to develop an a priori error estimate using techniques that belong to the realm of the least square finite elements for PINNs. To achieve this, we give a brief overview on GaLS in section 2, essentially following Bochev and Gunzburger (2009). In section 3, we present the notion of PINNs and present some results that will be used in section 4. These concepts can be found in Lu et al. (2021) and Hong et al. (2021). In section 4, we develop an a priori error estimate for PINNs when dealing with an elliptic operator, building on the idea that were introduced by Cai et al. (2020). The novelty of this paper is that it applies ideas used to analyse the convergence of GaLS least squares methods to PINNs.

2. GALERKIN LEAST SQUARES

Let X and Y be two Hilbert spaces and consider a Fredholm operator $Q \in \mathcal{L}(X, Y)$, we will now focus our attention on the following problem,

$$\text{find } u \in X \text{ such that } Qu = F. \quad (1)$$

We will work under the following assumption to simplify the discussion, $\dim(N(Q)) = 0$, where $N(Q)$ is the null space of Q . We will focus our interest on the **residual energy functional** and we consider the following minimization principle,

$$J(u; F) = \|Qu - F\|_Y^2, \quad u = \arg \min_{v \in X} J(v; F). \quad (2)$$

The above minimization principle admits a unique minimizer. To show this we use the following well known result.

Theorem 1. Given a reflexive Banach space X and a continuous and strongly convex functional, $J : X \rightarrow \mathbb{R}$, if the following conditions are satisfied,

- (1) $\lim_{\|x\| \rightarrow \infty} J(x) = \infty$,
- (2) K is a closed convex subset of X ,

than it exists a unique element $x^* \in K$ such that,

$$J(x^*) = \inf_{x \in K} J(x).$$

Theorem 2. The minimization principle (2) has a unique minimizer $u^* \in X$ for any $F \in Y$.

Proof. First we want to prove the ellipticity of the operator Q . To do this we notice that by assumption the range of Q , denoted $R(Q)$, is closed in Y and therefore $(R(Q), (\cdot, \cdot)_Y)$ is a Hilbert space. Now we consider the restricted mapping,

$$Q : X \rightarrow R(Q),$$

such mapping is a bijection because of the assumption $\dim(N(Q)) = 0$ and the fact we have taken as codomain of the operator $R(Q)$. Since the mapping is a bijection one can use the bounded inverse theorem to state that $Q^{-1} : R(Q) \rightarrow X$ is a linear bounded operator and therefore,

$$\|Qu\|_Y \geq C_1 \|u\|_X. \quad (3)$$

Now we proceed to prove the coercivity for $J(u; F)$, which is property 1 in *Theorem 1* statement.

$$\begin{aligned} J(u; F) &= (Qu - F, Qu - F)_Y = \\ &= (Qu, Qu)_Y - 2(Qu, F)_Y + (F, F)_Y = \\ &= \|Qu\|_Y^2 + \|F\|_Y^2 - 2(Qu, F)_Y \end{aligned}$$

using Young inequality with $\varepsilon = 2$ and (3) we get the following inequality,

$$J(u; F) \geq \frac{1}{2} \|Qu\|_Y^2 - \|F\|_Y^2 \geq \frac{C_1^2}{2} \|u\|_X^2 - \|F\|_Y^2,$$

which implies $\lim_{\|u\| \rightarrow \infty} J(x) = \infty$. The only thing left to prove is that the functional $J(u; F)$ is strictly convex, to do this we consider $u, v \in X$ and $t \in [0, 1]$ and we evaluate,

$$\begin{aligned} j(t) &= J(tu + (1-t)v; F) = \\ &= t^2(Qu - F, Qu - F)_Y + (1-t)^2(Qv - F, Qv - F)_Y + \\ &+ t(1-t)(Qu - F, Qv - F)_Y. \end{aligned}$$

If the coefficient of the quadratic term is greater than zero then the functional $J(u; F)$ is convex and this is the case because,

$$\begin{aligned} &(Qu - F, Qu - F)_Y - 2(Qu - F, Qv - F)_Y \\ &+ (Qv - F, Qv - F)_Y \geq 0 \end{aligned}$$

where we have used Young's inequality to obtain the last bound. In particular we notice that if

$$(Qu - F, Qu - F)_Y - 2(Qu - F, Qv - F)_Y + (Qv - F, Qv - F)_Y$$

is null then,

$$(Qu - Qv, Qu - Qv) = 0 \Leftrightarrow Q(u - v) = 0 \Leftrightarrow u = v,$$

where the last implication comes from the fact that we assume $\dim(N(Q)) = 0$. To conclude one needs to apply the *Theorem 1*.

In particular one can characterize the minimizer of (2) using the following proposition.

Proposition 3. The minimizer u^* of (2) solves the following variational equation,

$$\text{find } u \in X \text{ such that } a(u, v) = G(v) \quad \forall v \in X, \quad (4)$$

where $a(u, v)$ is the scalar product $(Qu, Qv)_Y$ and $G(v)$ is the following linear functional

$$\begin{aligned} G : X &\rightarrow Y \\ v &\mapsto (F, Qv)_Y. \end{aligned}$$

Proof. We define the following function of a real variable $j(t) = J(u^* + tv; F)$ and we notice that since u^* is a minimum for $J(u; F)$ then $j'(0) = 0$. Expanding $j'(t)$ we get,

$$j'(t) = 2t(Qv, Qv)_Y + 2(Qv, Qu - F)_Y.$$

Imposing $j'(0) = 0$ for all $v \in X$ one finds (4).

Now that we have shown that the continuous minimization principle (2) has a unique minimizer and such a minimizer is characterized by solving the variational equation (4) we focus our attention on the discrete problem associated with (2). Usually in the context of GaLS one chooses a conforming finite dimensional subspace $X_h \subset X$ and consider as discrete least square principle (2) the energy functional evaluated on X_h , i.e.

$$J(u; F) = \|Qu - F\|_Y^2, \quad u_h = \arg \min_{v \in X_h} J(v; F).$$

The advantage of the above mentioned approach is that to produce an a priori error estimate one can follow the standard practice of using Cea's lemma and interpolation error estimate in X_h . Another approach would be to use directly a discrete minimization principle, i.e.

$$J^h(u; F) = \|Qu - F\|_h^2, \quad u_h = \arg \min_{v \in X_h} J^h(v; F). \quad (5)$$

where $\|\cdot\|_h$ is the discrete energy norm associated with a discrete inner product, $(\cdot, \cdot)_h : X_h \times X_h \rightarrow \mathbb{R}$. A useful assumption to develop a priori error estimate for the above described discrete energy functional is that the discrete energy functional can be extended to $u \in X$. We will further assume that two positive semi-definite bilinear form $e(\cdot, \cdot)$ and $\epsilon(\cdot, \cdot)$ exist, such that,

$$\begin{aligned} J^h(u^h, F) &= \frac{1}{2} \left((u_h, u_h)_h + (u, u)_h + \epsilon(u, u) \right) - (u, u_h)_h \\ &- e(u, u_h) \quad \forall u \in X, \quad \forall u_h \in X_h, \end{aligned} \quad (6)$$

under this assumption and proceed to prove an a priori error estimate for our discrete least square principle (5). It is important to notice that we can proceed as we have done at the beginning of this section in order to prove the existence of a minimizer for (5).

Theorem 4. Under the above assumption and letting $u \in X$ be the solution of (2), $u_h \in X_h$ be the unique solution of (5), then the following error estimate holds,

$$\|u - u_h\|_h \leq \inf_{v \in X_h} \|u - v\|_h + \sup_{v \in X_h} \frac{e(u, v)}{\|v\|_h}. \quad (7)$$

In particular if the least square functional J^h is r -consistent, i.e. it exist $r > 0$ such that,

$$\sup_{v \in X_h} \frac{e(u, v)}{\|v\|_h} \leq C(u)h^r,$$

then the above error estimate becomes,

$$\|u - u_h\|_h \leq \inf_{v \in X_h} \|u - v\|_h + C(u)h^r.$$

Proof. Let u_h^\perp be the orthogonal projection of u onto X_h^\perp with respect to the discrete scalar product $(\cdot, \cdot)_h$. We know that the following chain of inequalities holds,

$$\begin{aligned} \|u_h - u\|_h &\leq \|u_h^\perp - u\|_h + \|u_h - u_h^\perp\|_h \\ &\leq \inf_{v \in X_h} \|u - v\|_h + \|u_h - u_h^\perp\|_h. \end{aligned}$$

Therefore we are left to estimate the term $\|u_h - u_h^\perp\|_h$. In order to achieve such estimate we observe that,

$$(u_h - u, v)_h = e(u, v) \quad \forall v \in X_h$$

is the variational equation associated with minimizing (6) and given the fact that u_h^\perp is the orthogonal projection of u one has,

$$(u_h^\perp - u, v)_h = 0 \quad \forall v \in X_h.$$

Subtracting the last two equations and using the definition of norm induced by the linear operator we get,

$$(u_h^\perp - u_h, v)_h = e(u, v) \quad v \in X_h$$

$$\|u_h^\perp - u_h\|_h = \sup_{v \in X_h} \frac{(u_h^\perp - u_h, v)_h}{\|v\|_h} = \sup_{v \in X_h} \frac{e(u, v)_h}{\|v\|_h}.$$

We will split the contribution of the differential operator on the inside of the domain Ω and on its boundary $\partial\Omega$ obtaining the following version of (1),

$$Q : X \rightarrow Y := A(\Omega) \times B(\partial\Omega) \quad (8)$$

$$u \mapsto (\mathcal{L}u, \mathcal{B}u),$$

where $A(\Omega)$ and $B(\partial\Omega)$ are Hilbert spaces representing respectively component of Y on Ω and $\partial\Omega$.

3. PHYSICS INFORMED NEURAL NETWORK

In this section we will introduce the notion of Physics Informed Neural Networks (PINNs) and the correct functional setting where to study PINNs.

Definition 5. (Forward Neural Network). We say $u_h^L(\mathbf{x}) : \mathbb{R}^{d_{in}} \rightarrow \mathbb{R}^{d_{out}}$ is a $(L - 1)$ hidden layer forward neural network (FNN) with N_ℓ neurons in the $\ell - th$ layer and activation function $\sigma : \mathbb{R} \rightarrow \mathbb{R}$, if the action $u_h^L(\mathbf{x})$ is recursively defined as,

- (1) $u_h^0(\mathbf{x}) = \mathbf{x} \in \mathbb{R}^{d_{in}}$,
- (2) $\forall 1 \leq \ell \leq L - 1 \quad u_h^\ell(\mathbf{x}) = \sigma(W^\ell N^{\ell-1}(\mathbf{x}) + \mathbf{b}_\ell) \in \mathbb{R}^{N_\ell}$,
- (3) $u_h^L(\mathbf{x}) = W^L u_h^{L-1}(\mathbf{x}) + \mathbf{b}^L \in \mathbb{R}^{d_{out}}$,

where $W^\ell \in \mathbb{R}^{N_\ell \times N_{\ell-1}}$ and $\mathbf{b}_\ell \in \mathbb{R}^{N_\ell}$. In particular the value of the matrix W^ℓ will be called kernel parameters and the value of the vector \mathbf{b}^ℓ will be called bias parameters for the layer ℓ . The parameters of the network will be denoted as $\boldsymbol{\theta} = (\{W_\ell\}_{\ell=1}^L, \{\mathbf{b}_\ell\}_{\ell=1}^L)$ and we indicate the dependence of the FNN on a particular choice of parameters $\hat{\boldsymbol{\theta}}$ writing $u_{h, \hat{\boldsymbol{\theta}}}^L$. We will often omit both the dimension of the network and the dependence on the parameters writing simply u_h .

We will focus our attention on the logistic sigmoid, i.e. $\sigma(x) = \frac{1}{1+e^{-x}}$, and the hyperbolic tangent as activation functions. When we speak about Physics Informed Neural Networks (PINNs), we mean a FNN trained to solve (1) using the following algorithm:

- (1) Given a probability distribution of points on a set A , called $\mathcal{D}(A)$, we consider two vectors of realisations, i.e.

$$\omega_i \sim \mathcal{D}(\Omega) \quad \forall 1 \leq i \leq N_\Omega$$

$$\beta_i \sim \mathcal{D}(\partial\Omega) \quad \forall 1 \leq i \leq N_{\partial\Omega}.$$

- (2) We consider the following loss function, i.e.

$$\mathcal{L}(\boldsymbol{\theta}; \omega_i, \beta_i) = \frac{1}{N_\Omega \gamma_1} \sum_{i=1}^{N_\Omega} \|\mathcal{L}u_h^L(\omega_i) - f(\omega_i)\|_a^2 + \frac{1}{N_{\partial\Omega} \gamma_2} \sum_{i=1}^{N_{\partial\Omega}} \|\mathcal{B}u_h^L(\beta_i) - g(\beta_i)\|_b^2, \quad (9)$$

where γ_i are parameters depending on the dimension of Ω , and $F = (f, g) \in A(\Omega) \times B(\partial\Omega)$.

- (3) We train the FNN in order to find $\boldsymbol{\theta}^*$ that minimize the above loss function.

It is worth noticing that the optimization problem in step 3 of the PINNs algorithm is highly non convex with respect to $\boldsymbol{\theta}$, Blum and Rivest (1993), and the state of the art for training PINNs is to use a first-order descent method such as Adam or L-BFGS.

In order to develop an a priori error estimate for PINNs we need a class of sufficient regularity for the solution of our PDE, thus we will here give a very brief brush on concepts that are treated in more detail in Siegel and Xu (2021b), Weinan et al. (2019), Wojtowysch and Weinan (2020) and Siegel and Xu (2021a). Let X be the Hilbert space that we have introduced in the previous section and $\mathbb{D} \subset X$ a dictionary of elements in X such that, $K_{\mathbb{D}} := \sup_{d \in \mathbb{D}} \|d\|_X < \infty$.

In the above setting one can define the functional space of interest for our work, as:

Definition 6. (Barron Space). Let us consider the closure of the convex symmetric hull of \mathbb{D} ,

$$B_1(\mathbb{D}) = \left\{ \sum_{j=1}^n a_j d_j : n \in \mathbb{N}, d_j \in \mathbb{D}, \|\{a_j\}_{j=1}^n\|_{\ell_1} \leq 1 \right\}.$$

we define the Barron space and the associated norm as follow:

$$\|\cdot\|_{\mathcal{K}(\mathbb{D})} = \inf \left\{ c > 0 : cf \in B_1(\mathbb{D}) \right\},$$

$$\mathcal{K}(\mathbb{D}) = \left\{ f \in X : \|f\|_{\mathcal{K}(\mathbb{D})} < \infty \right\}.$$

In particular given the fact that X is a Hilbert space it is obvious that $\mathcal{K}_{\mathbb{D}}$ is a subset of X .

The importance of the Barron space is that given a function in $\mathcal{K}(\mathbb{D})$ we have an a priori approximation estimates for shallow neural network. To formalize such a concept we notice that the set of shallow FNN, i.e. FNN with one hidden layer and null bias on the output layer, can be rewritten as

$$\Sigma_{N, M}(\mathbb{D}) = \left\{ \sum_{j=1}^N a_j d_j : d_j \in \mathbb{D} \text{ and } \|\{a_j\}_{j=1}^N\|_{\ell_1} \leq M \right\},$$

where we imposed the additional constraint on the ℓ_1 norm of the coefficients a_j and chosen a particular dictionary,

$$\mathbb{D}_\sigma = \left\{ \sigma(\mathbf{w}_i \cdot \mathbf{x} + \mathbf{b}_i) : \mathbf{w}_i \in \mathbb{R}^d \text{ and } \mathbf{b}_i \in \mathbb{R} \right\}.$$

Now since X is a Hilbert space it is also a type 2 Banach space, we have the following approximation estimate.

Theorem 7. (Maurey and Pisier (1976), DeVore (1998)). Let X be a Hilbert space and given $f \in \mathcal{K}(\mathbb{D}_\sigma)$ we have the following approximation estimate:

$$\inf_{f_N \in \Sigma_{N, M}(\mathbb{D}_\sigma)} \|f - f_N\|_X \leq C_X K_{\mathbb{D}_\sigma} \|f\|_{\mathcal{K}_1(\mathbb{D}_\sigma)} N^{-\frac{1}{2}},$$

where C_X is the type-2 constant for the space X . Furthermore when σ is a bounded activation function then \mathbb{D}_σ is uniformly bounded in $\mathcal{K}(\mathbb{D}_\sigma)$.

The above error estimate can be improved for certain types of activation functions. More details can be found in Siegel and Xu (2021b), Makovoz (1996) and Pinkus (1999). In the next section we will use the concept developed in the previous section together with *Theorem 7* to prove an a priori error estimate, similarly to what is done using Céa's lemma in the context of classical conforming finite element methods. To conclude this section we notice that a regularity theory for elliptic problems in Barron space has yet to be fully developed, in particular the sufficient condition to impose on the source function in order for the solution to live in $\mathcal{K}(\mathbb{D}_\sigma)$ are not known, more information on this problem can be found in Chen et al. (2021) and Weinan and Wojtowytsch (2022).

4. A PRIORI ERROR ESTIMATE

In this section the a priori error estimate that is the aim of this paper will be presented. In particular we will consider the following energy functional associated with the strong formulation (8),

$$J(u; F) = \|Qu - F\|_Y^2 = \int_Y \|Qu - F\|_y \, d\mathbf{x} = \int_\Omega \|\mathcal{L}u - f\|_a \, d\mathbf{x} + \int_{\partial\Omega} \|\mathcal{B}u - g\|_b \, d\mathbf{x} \quad (10)$$

where $Q = (\mathcal{L}, \mathcal{B})$, the space Y is the product space $A(\Omega) \times B(\partial\Omega)$ and the space A and B are Hilbert spaces equipped with the following scalar product and induced norm,

$$\begin{aligned} (u, v)_{A(\Omega)} &= \int_\Omega \left(u(\mathbf{x}), v(\mathbf{x}) \right)_a \, d\mathbf{x} \\ \|u\|_{A(\Omega)}^2 &= \int_\Omega \left(u(\mathbf{x}), u(\mathbf{x}) \right)_a \, d\mathbf{x} = \int_\Omega |u(\mathbf{x})|_a^2 \, d\mathbf{x}, \\ (u, v)_{B(\partial\Omega)} &= \int_{\partial\Omega} \left(u(\mathbf{x}), v(\mathbf{x}) \right)_b \, d\mathbf{x} \\ \|u\|_{B(\partial\Omega)}^2 &= \int_{\partial\Omega} \left(u(\mathbf{x}), u(\mathbf{x}) \right)_b \, d\mathbf{x} = \int_{\partial\Omega} |u(\mathbf{x})|_b^2 \, d\mathbf{x} \end{aligned}$$

The discrete scalar product that gives us the discrete energy functional (9) is:

$$\begin{aligned} (\cdot, \cdot)_h : X_h \times X_h &\rightarrow \mathbb{R} \\ (u_h, v_h)_h &\mapsto \frac{1}{N_\Omega^{\gamma_1}} \sum_{i=1}^{N_\Omega} \left(u_h(\omega_i), v_h(\omega_i) \right)_a + \\ &\frac{1}{N_{\partial\Omega}^{\gamma_2}} \sum_{i=1}^{N_{\partial\Omega}} \left(u_h(\beta_i), v_h(\beta_i) \right)_b. \end{aligned}$$

In particular we notice that the above mapping is bilinear thanks to the fact that $(\cdot, \cdot)_a$ and $(\cdot, \cdot)_b$ are scalar products. Furthermore since $(\cdot, \cdot)_a$ and $(\cdot, \cdot)_b$ are scalar products one has that $(u_h, u_h)_h \geq 0$ for all $u_h \in X_h$. Last we have to prove that $(u_h, u_h)_h = 0 \Leftrightarrow u_h \equiv 0$ in order to do this we will work under the following assumption.

Assumption 4.1. Fixed $u_h \in X_h$ it exists a set of points $\{\delta_1, \dots, \delta_{NDOF}\}$ such that, $u_h(\delta_1) = \dots = u_h(\delta_{NDOF}) = 0 \Leftrightarrow u_h \equiv 0$. The above mentioned assumption can also be rephrased as $u_h(\delta_i)$ are unsolvent degrees of freedom for functions in X_h .

It's important to notice that in order for *Assumption 4.1* to grant that $(u_h, u_h)_h = 0 \Leftrightarrow u_h \equiv 0$ one needs to slightly modify the algorithm used to train PINNs, in particular at each training step we need to choose the points where we evaluate the loss function (9) dynamically, in order for *Assumption 4.1* to be verified. Furthermore *Assumption 4.1* can't be verified for all activation functions. For example, when dealing with second order PDE if we consider a sigmoid activation function it is sufficient to choose points close enough to the value of the bias layer for *Assumption 4.1* to be verified. However *Assumption 4.1* can't be verified if we consider a RELU activation function.

We now proceed using Rademacher complexity tools as shown in Hong et al. (2021) in order to bound the right hand side of (7).

Theorem 8. (Hong et al. (2021), Wainwright (2019)). Given a class of functions $\mathcal{F} : \Omega \rightarrow \mathbb{R}$ and a collection of sample points $\{\omega_i\}_{i=1}^{N_\Omega}$,

$$\begin{aligned} \mathbb{E}_{\omega_i \sim \mathcal{D}(\Omega)} \sup_{h \in \mathcal{F}} \left| \sum_{i=1}^{N_\Omega} \frac{h(\omega_i)}{N_\Omega} - \int_\Omega h(x) \, d\mathcal{D}(\Omega) \right| &\leq 2R_{N_\Omega}(\mathcal{F}) \\ &:= \mathbb{E}_{\omega_i \sim \mathcal{D}(\Omega)} \mathbb{E}_{\xi_i} \left[\sup_{h \in \mathcal{F}} \frac{1}{N_\Omega} \sum_{i=1}^{N_\Omega} \xi_i h(\omega_i) \right], \end{aligned}$$

where ξ_i are Rademacher random variables and ω_i are uniformly distributed points that includes the $\delta_i, \dots, \delta_{NDOF}$ from *Assumption 4.1*.

Corollary 9. Given a class of functions $\mathcal{F} : \partial\Omega \rightarrow \mathbb{R}$ and a collection of sample points $\{\beta_i\}_{i=1}^{N_{\partial\Omega}}$,

$$\begin{aligned} \mathbb{E}_{\beta_i \sim \mathcal{D}(\partial\Omega)} \sup_{h \in \mathcal{F}} \left| \frac{1}{N_{\partial\Omega}} \sum_{i=1}^{N_{\partial\Omega}} h(\beta_i) - \int_{\partial\Omega} h(x) \, d\mathcal{D}(\partial\Omega) \right| &\leq \\ 2R_{N_{\partial\Omega}}(\mathcal{F}) &:= \mathbb{E}_{\beta_i \sim \mathcal{D}(\partial\Omega)} \mathbb{E}_{\xi_i} \left[\sup_{h \in \mathcal{F}} \frac{1}{N_{\partial\Omega}} \sum_{i=1}^{N_{\partial\Omega}} \xi_i h(\beta_i) \right], \end{aligned}$$

where ξ_i are Rademacher random variables and ω_i are uniformly distributed points that include the $\delta_i, \dots, \delta_{NDOF}$ from *Assumption 4.1*.

From now on when writing the expected value we will not specify the dependence on the distribution.

Theorem 10. (Hong et al. (2021)). Let \mathcal{F}, \mathcal{S} be classes of functions from Ω into \mathbb{R} , then the following statement holds,

- (1) $R_{N_p}(\text{conv}(\mathcal{F})) = R_{N_p}(\mathcal{F})$.
- (2) $R_{N_p}(\mathcal{F} + \mathcal{S}) = R_{N_p}(\mathcal{F}) + R_{N_p}(\mathcal{S})$, where $\mathcal{F} + \mathcal{S} = \{h(x) + g(x) : h \in \mathcal{F}, g \in \mathcal{S}\}$.
- (3) $R_{N_p}(\Phi \circ \mathcal{F}) \leq LR_{N_p}(\mathcal{F})$, where

$$\Phi \circ \mathcal{F} = \{\Phi \circ h(x) : h \in \mathcal{F}\}$$

and $\Phi : \mathbb{R} \rightarrow \mathbb{R}$ is Lipschitz function with Lipschitz constant L .

- (4) $R_{N_p}(f \cdot \mathcal{F}) \leq \|f\|_{\mathcal{L}^\infty(\Omega)} R_{N_p}(\mathcal{F})$, where

$$f \cdot \mathcal{F} = \{f(x)h(x) : h \in \mathcal{F}\}$$

and $f : \Omega \rightarrow \mathbb{R}$ is a fixed function.

We will from now on assume that the differential operator $Q : X \rightarrow Y$ from (1) is an elliptic operator and we focus on the class of energy functional considered in (10),

$$\mathcal{L}u := \sum_{|\alpha| \leq m} a_\alpha(x) \partial^\alpha u, \quad \mathcal{B}u := \sum_{|\alpha| \leq m} b_\alpha(x) \partial^\alpha u, \quad (11)$$

$$\mathcal{F}_{N,M}^\Omega = \left\{ \left(\mathcal{L}u - f, \mathcal{L}u - f \right)_a : u \in \Sigma_{N,M} \right\},$$

$$\mathcal{F}_{N,M}^{\partial\Omega} = \left\{ \left(\mathcal{B}u - g, \mathcal{B}u - g \right)_b : u \in \Sigma_{N,M} \right\}.$$

Theorem 11. We will assume that $\|a_\alpha\|_{\mathcal{L}^\infty(\Omega)} \leq K_a$, $\|b_\alpha\|_{\mathcal{L}^\infty(\Omega)} \leq K_b$, $f \in \mathcal{L}^\infty(\Omega)$ and $g \in \mathcal{L}^\infty(\partial\Omega)$ then we have the following bound on the Rademacher complexity of $\mathcal{F}_{N,M}$,

$$\begin{aligned} R_{N_\Omega}(\mathcal{F}_{N,M}^\Omega) &\leq MK_a K_{\mathbb{D}} \sum_{|\alpha| \leq m} R_{N_\Omega}(\partial^\alpha \mathbb{D}) + \\ &2MK_a \|f\|_{\mathcal{L}^\infty(\Omega)} \sum_{|\alpha| \leq m} R_{N_\Omega}(\partial^\alpha \mathbb{D}), \\ R_{N_{\partial\Omega}}(\mathcal{F}_{N,M}^{\partial\Omega}) &\leq MK_b K_{\mathbb{D}} \sum_{|\alpha| \leq m} R_{N_{\partial\Omega}}(\partial^\alpha \mathbb{D}) + \\ &2MK_b \|g\|_{\mathcal{L}^\infty(\partial\Omega)} \sum_{|\alpha| \leq m} R_{N_{\partial\Omega}}(\partial^\alpha \mathbb{D}), \end{aligned}$$

provided that $u \mapsto (u, u)_y$ is a locally Lipschitz function.

Proof. We begin considering

$$\mathcal{F}_{N,M} = \left\{ \left(Qu - F, Qu - F \right)_y : u \in \Sigma_{N,M} \right\}$$

and develop the scalar product in order to obtain,

$$\begin{aligned} \left(\sum_{|\alpha| \leq m} a_\alpha(x) \partial^\alpha u - f, \sum_{|\alpha| \leq m} a_\alpha(x) \partial^\alpha u - f \right)_a &= \\ \left(\sum_{|\alpha| \leq m} a_\alpha(x) \partial^\alpha u, \sum_{|\alpha| \leq m} a_\alpha(x) \partial^\alpha u \right)_a &- \\ -2 \left(\sum_{|\alpha| \leq m} a_\alpha(x) \partial^\alpha u, f \right)_a &- (f, f)_a, \\ \left(\sum_{|\alpha| \leq m} b_\alpha(x) \partial^\alpha u - g, \sum_{|\alpha| \leq m} b_\alpha(x) \partial^\alpha u - g \right)_b &= \\ \left(\sum_{|\alpha| \leq m} b_\alpha(x) \partial^\alpha u, \sum_{|\alpha| \leq m} b_\alpha(x) \partial^\alpha u \right)_b &- \\ -2 \left(\sum_{|\alpha| \leq m} b_\alpha(x) \partial^\alpha u, g \right)_b &- (g, g)_b. \end{aligned}$$

Now using *Theorem 10* and the fact that $u \mapsto (u, u)_a$, $u \mapsto (u, u)_b$ are locally Lipschitz then one has the following bound on the Rademacher complexity of $\mathcal{F}_{N,M}$,

$$\begin{aligned} R_{N_\Omega}(\mathcal{F}_{N,M}^\Omega) &\leq MK_a K_{\mathbb{D}} \sum_{|\alpha| \leq m} R_{N_\Omega}(\partial^\alpha \mathbb{D}) + \\ &2MK_a \|f\|_{\mathcal{L}^\infty(\Omega)} \sum_{|\alpha| \leq m} R_{N_\Omega}(\partial^\alpha \mathbb{D}), \\ R_{N_{\partial\Omega}}(\mathcal{F}_{N,M}^{\partial\Omega}) &\leq MK_b K_{\mathbb{D}} \sum_{|\alpha| \leq m} R_{N_{\partial\Omega}}(\partial^\alpha \mathbb{D}) + \\ &2MK_b \|g\|_{\mathcal{L}^\infty(\partial\Omega)} \sum_{|\alpha| \leq m} R_{N_{\partial\Omega}}(\partial^\alpha \mathbb{D}). \end{aligned}$$

In order to obtain the above inequality we have also used the fact that the Rademacher complexity of a constant is null.

Theorem 12. (Hong et al. (2021)). Assuming the activation function σ lives in $\mathcal{W}^{m+1, \infty}$, then for any α such that $|\alpha| \leq m$ one has the following estimate for the Rademacher complexity,

$$R_{N_p}(\partial^\alpha \mathbb{D}_\sigma) \leq CN_p^{-\frac{1}{2}},$$

where the constant C does not depend on N_p .

Corollary 13. In the hypothesis of *Theorem 12*, one has the following bound on the Rademacher complexity of $\mathcal{F}_{N,M}^\Omega$ and $\mathcal{F}_{N,M}^{\partial\Omega}$,

$$\begin{aligned} R_{N_\Omega}(\mathcal{F}_{N,M}^\Omega) &\leq MK_a K_{\mathbb{D}} N_\Omega^{-\frac{1}{2}} + 2MK_a \|f\|_{\mathcal{L}^\infty(\Omega)} N_\Omega^{-\frac{1}{2}}, \\ R_{N_{\partial\Omega}}(\mathcal{F}_{N,M}^{\partial\Omega}) &\leq MK_b K_{\mathbb{D}} N_{\partial\Omega}^{-\frac{1}{2}} + 2MK_b \|g\|_{\mathcal{L}^\infty(\partial\Omega)} N_{\partial\Omega}^{-\frac{1}{2}}. \end{aligned}$$

Now we are ready to prove our a priori error estimate, because if we fix $\gamma_i = 1$ for $i = 1, 2$ the above corollary together with *Theorem 8* tell us that our discrete energy functional is $\frac{1}{4}$ -consistent and therefore one has from *Theorem 4* that,

$$\mathbb{E}_{\omega, \beta} \|u - u_h\|_h \leq \mathbb{E}_{\omega, \beta} \inf_{v \in \Sigma_{N,M}} \|u - v\|_h + C_a N_\Omega^{-\frac{1}{4}} + C_b N_{\partial\Omega}^{-\frac{1}{4}},$$

where C_a is $\max \left\{ MK_a K_{\mathbb{D}_\sigma}, 2MK_a \|f\|_{\mathcal{L}^\infty(\Omega)} \right\}$ and C_b is $\max \left\{ MK_b K_{\mathbb{D}_\sigma}, 2MK_b \|g\|_{\mathcal{L}^\infty(\partial\Omega)} \right\}$. Last we notice that once again using *Theorem 8* and *Corollary 13*, one has that $\mathbb{E}_{\omega, \beta} \|u - v\|_h \leq \mathbb{E}_{\omega, \beta} \|u - v\|_X + C_a N_\Omega^{-\frac{1}{4}} + C_b N_{\partial\Omega}^{-\frac{1}{4}}$ and therefore thanks to *Theorem 7* the above equation becomes,

$$\begin{aligned} \mathbb{E}_{\omega, \beta} \|u - u_h\|_h &\leq C_X K_{\mathbb{D}_\sigma} \|u\|_{\mathcal{K}(\mathbb{D}_\sigma)} N^{-\frac{1}{2}} \\ &+ 2C_a N_\Omega^{-\frac{1}{4}} + 2C_b N_{\partial\Omega}^{-\frac{1}{4}}. \end{aligned}$$

where in this case N is the number of neurons in the shallow layer.

Theorem 14. Considering (1) with Q as in (11), assuming $u \in \mathcal{K}(\mathbb{D}_\sigma)$, $F \in \mathcal{L}^\infty(\Omega)$ and $\sigma \in \mathcal{W}^{m+1, \infty}$ then,

$$\begin{aligned} \mathbb{E}_{\omega, \beta} \|u - u_h\|_X &\leq 2C_X K_{\mathbb{D}_\sigma} \|u\|_{\mathcal{K}_1(\mathbb{D}_\sigma)} N^{-\frac{1}{2}} \\ &+ 3C_a N_\Omega^{-\frac{1}{4}} + 3C_b N_{\partial\Omega}^{-\frac{1}{4}}, \end{aligned}$$

where in this case N is the number of neurons in the shallow layer, N_Ω and $N_{\partial\Omega}$ are respectively the number of evaluation points on the boundary and inside of Ω .

5. CONCLUSION

A connection between Galerkin least square finite elements and physics informed neural network has been drawn and standard tools for least square finite elements methods have been used to develop an a priori error estimate for physics informed neural networks when solving (1). A similar idea have been applied when considering the RELU activation function by Cai et al. (2020). It is important to notice that the a priori error estimate presented here is suboptimal with respect to what is observed numerically when using a sigmoid activation function. This suboptimality can not be improved upon when using a sigmoid activation function, given the fact that in Siegel and Xu (2021b) it has been proven that *Theorem 7* is optimal for sigmoid activation functions. It is worth noting that suboptimality problems for a priori error estimates, seem to be common whenever Rademacher complexity is used and can be overcome using ideas presented in Mishra and Molinaro (2020). It is important to notice that the a priori

error estimate presented in *Theorem 14* only works if u_h is the minimizer of the energy functional in $\mathcal{K}(\mathbb{D}_\sigma)$. Given that PINNs training is a highly non convex optimization problem, there are no guarantees that one can find such a minimizer. On the other hand a priori error estimates presented in Hong et al. (2021) depend on the use of a particular greedy algorithm but hold at each step of the minimization process. Last but not least numerical simulation showing the application of PINNs to different elliptic problem is outside the scope of this paper but will be addressed in future papers in collaboration with KAUST Extreme Computing Research Center (ECRC).

ACKNOWLEDGEMENTS

I would like to express my deepest appreciation to Prof. G. Turkiyyah and Dr. S. Zampini without whom this paper would not have been possible.

REFERENCES

- Blum, A.L. and Rivest, R.L. (1993). Training a 3-node neural network is NP-complete. In *Machine learning: from theory to applications*, volume 661 of *Lecture Notes in Comput. Sci.*, 9–28. Springer, Berlin.
- Bochev, P.B. and Gunzburger, M.D. (2009). *Least-squares finite element methods*, volume 166 of *Applied Mathematical Sciences*. Springer, New York.
- Cai, Z., Chen, J., Liu, M., and Liu, X. (2020). Deep least-squares methods: An unsupervised learning-based numerical method for solving elliptic pdes. *Journal of Computational Physics*, 420, 109707.
- Chen, Z., Lu, J., and Lu, Y. (2021). On the representation of solutions to elliptic pdes in barron spaces. *arXiv preprint arXiv:2106.07539*.
- DeVore, R.A. (1998). Nonlinear approximation. In *Acta numerica, 1998*, volume 7 of *Acta Numer.*, 51–150. Cambridge Univ. Press, Cambridge.
- Eason, E.D. (1976). A review of least-squares methods for solving partial differential equations. *International journal for numerical methods in engineering*, 10(5), 1021–1046.
- Gao, X. and Duan, L.M. (2017). Efficient representation of quantum many-body states with deep neural networks. *Nature communications*, 8(1), 1–6.
- Gu, Y., Yang, H., and Zhou, C. (2021). Selectnet: Self-paced learning for high-dimensional partial differential equations. *Journal of Computational Physics*, 441, 110444.
- He, J., Li, L., Xu, J., and Zheng, C. (2018). Relu deep neural networks and linear finite elements. *arXiv preprint arXiv:1807.03973*.
- Hong, Q., Siegel, J.W., and Xu, J. (2021). A priori analysis of stable neural network solutions to numerical pdes. *arXiv preprint arXiv:2104.02903*.
- Jiang, B.n. (1998). *The least-squares finite element method: theory and applications in computational fluid dynamics and electromagnetics*. Springer Science & Business Media.
- Karniadakis, G.E., Kevrekidis, I.G., Lu, L., Perdikaris, P., Wang, S., and Yang, L. (2021). Physics-informed machine learning. *Nature Reviews Physics*, 3(6), 422–440.
- Lu, L., Meng, X., Mao, Z., and Karniadakis, G.E. (2021). DeepXDE: a deep learning library for solving differential equations. *SIAM Rev.*, 63(1), 208–228.
- Lye, K.O., Mishra, S., and Molinaro, R. (2021). A multi-level procedure for enhancing accuracy of machine learning algorithms. *European Journal of Applied Mathematics*, 32(3), 436–469.
- Makovoz, Y. (1996). Random approximants and neural networks. *Journal of Approximation Theory*, 85(1), 98–109. doi:<https://doi.org/10.1006/jrath.1996.0031>.
- Maurey, B. and Pisier, G. (1976). Séries de variables aléatoires vectorielles indépendantes et propriétés géométriques des espaces de banach. *Studia Mathematica*, 58(1), 45–90.
- Mishra, S. and Molinaro, R. (2020). Estimates on the generalization error of physics informed neural networks (pinns) for approximating pdes. *arXiv preprint arXiv:2006.16144*.
- Mishra, S. and Molinaro, R. (2021). Physics informed neural networks for simulating radiative transfer. *Journal of Quantitative Spectroscopy and Radiative Transfer*, 270, 107705.
- Pinkus, A. (1999). Approximation theory of the mlp model in neural networks. *Acta Numerica*, 8, 143–195.
- Shin, Y., Darbon, J., and Karniadakis, G.E. (2020). On the convergence and generalization of physics informed neural networks. *arXiv e-prints*, arXiv–2004.
- Siegel, J.W. and Xu, J. (2021a). Characterization of the variation spaces corresponding to shallow neural networks. *arXiv preprint arXiv:2106.15002*.
- Siegel, J.W. and Xu, J. (2021b). Sharp lower bounds on the approximation rate of shallow neural networks. *arXiv preprint arXiv:2106.14997*.
- Wainwright, M.J. (2019). *High-Dimensional Statistics: A Non-Asymptotic Viewpoint*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press. doi:10.1017/9781108627771.
- Weinan, E., Han, J., and Jentzen, A. (2017). Deep learning-based numerical methods for high-dimensional parabolic partial differential equations and backward stochastic differential equations. *Communications in Mathematics and Statistics*, 5(4), 349–380.
- Weinan, E., Han, J., and Jentzen, A. (2021). Algorithms for solving high dimensional pdes: From nonlinear monte carlo to machine learning. *Nonlinearity*, 35(1), 278.
- Weinan, E., Ma, C., and Wu, L. (2019). Barron spaces and the compositional function spaces for neural network models. *arXiv preprint arXiv:1906.08039*.
- Weinan, E. and Wojtowytsch, S. (2022). Some observations on high-dimensional partial differential equations with barron data. In J. Bruna, J. Hesthaven, and L. Zdeborova (eds.), *Proceedings of the 2nd Mathematical and Scientific Machine Learning Conference*, volume 145 of *Proceedings of Machine Learning Research*, 253–269. PMLR. URL <https://proceedings.mlr.press/v145/e22a.html>.
- Wojtowytsch, S. and Weinan, E. (2020). Can shallow neural networks beat the curse of dimensionality? a mean field training perspective. *IEEE Transactions on Artificial Intelligence*, 1(2), 121–129.
- Xu, J. (2020). The finite neuron method and convergence analysis. *arXiv preprint arXiv:2010.01458*.
- Yu, B. et al. (2017). The deep ritz method: a deep learning-based numerical algorithm for solving variational problems. *arXiv preprint arXiv:1710.00211*.