Numer. Math. Theor. Meth. Appl.
doi: 10.4208/nmtma.OA-2023-0058

Vol. **xx**, No. x, pp. 1-29
x 2023

Analysis of Deep Ritz Methods for Semilinear Elliptic Equations

Mo Chen[1], Yuling Jiao[1,2], Xiliang Lu[1,2,*], Pengcheng Song[1], Fengru Wang[1] and Jerry Zhijian Yang[1,2]

[1] *School of Mathematics and Statistics, Wuhan University,
299 Ba Yi Road, Wuhan 430072, P.R. China*
[2] *Hubei Key Laboratory of Computational Science, Wuhan
University, 299 Ba Yi Road, Wuhan 430072, P.R. China*

Received 9 May 2023; Accepted (in revised version) 8 October 2023

Abstract. In this paper, we propose a method for solving semilinear elliptical equations using a ResNet with ReLU^2 activations. Firstly, we present a comprehensive formulation based on the penalized variational form of the elliptical equations. We then apply the Deep Ritz Method, which works for a wide range of equations. We obtain an upper bound on the errors between the acquired solutions and the true solutions in terms of the depth $\mathcal{D}$, width $\mathcal{W}$ of the ReLU^2 ResNet, and the number of training samples n. Our simulation results demonstrate that our method can effectively overcome the curse of dimensionality and validate the theoretical results.

AMS subject classifications: 35J61, 68T07, 65N12, 65N15

Key words: Semilinear elliptic equations, Deep Ritz method, ReLU^2 ResNet, convergence rate.

1. Introduction

Solving semilinear partial differential equations in high dimensional space is a challenging problem in physics and engineering with applications in hydromechanics (Navier-Stokes equations, Burgers equations) [5,11,23], quantum mechanics (Gross Pitaevskii equations) [3], variational geometry (Plateaus equations) [13], and more. Traditional numerical methods such as finite element, finite difference, and finite volume encounter the curse of dimensionality, where the number of parameters exponentially increases as the dimension grows, rendering these mesh-based methods im-

[*] Corresponding author. *Email addresses:* `cm.math@whu.edu.cn` (M. Chen), `yulingjiaomath@whu.edu.cn` (Y. Jiao), `xllv.math@whu.edu.cn` (X. Lu), `2017300030056@whu.edu.cn` (P. Song), `wangfr@whu.edu.cn` (F. Wang), `zjyang.math@whu.edu.cn` (J. Z. Yang)

practical. Recent attempts have been made to overcome this challenge, with one of the most promising tools being deep neural networks (DNN). The approximability of DNNs has been shown to overcome the curse of dimensionality, leading to the development of related methods [16, 38, 39], such as physics-informed neural networks (PINNs) [10, 20, 21, 31–33], Deep Galerkin method (DGM) [9, 22, 26, 36], and weak adversarial networks (WAN) [2, 7, 40].

The Deep Ritz method (DRM) is one of the most renowned approaches in the field of elliptic equations, capable of solving both the equations and the eigenvalue problems [9, 12, 14, 17, 19, 25, 27, 28, 30]. In this article, we present its application in nonlinear elliptic equations and provide a convergent analysis. To apply the method, we identify the functional variation that corresponds to the PDEs, and then replace the trial function with a deep neural network (DNN). We subsequently discretize it using the Monte Carlo algorithm [18, 37] and solve the discretized variation to approximate the solution. By following these steps, we can divide the error into two components: the approximation error and the statistical error. To bound the statistical error, we need to calculate the infinity norm of both the solution and its derivative [14, 28]. However, this requirement narrows down the method's applicability. To address this issue, we can use one of two methods. The first is to restrict the feasible parameter region of DNN [27]. In this case, the statistical error can be easily estimated, as the Rademacher complexity can be computed in the parameter space. However, the approximation error can be challenging to compute, especially for DNN with large depth. The second method is the one we propose in this article. By directly bounding the $W^{1,\infty}$ norm of the neural networks, we estimate the Rademacher complexity in the function space, and we can obtain the approximation error through the traditional mollifier technique.

The outline of this paper is as follows. In Section 2, we establish the primary problem of our article and introduce the notation we use. In Section 3, we present the variational loss of the problem and construct a simple error decomposition. The main theorem of the article is presented in Section 4 before its proof, for ease of reading. In Section 5, we provide numerical results to verify the effectiveness of the proposed method. Finally, we conclude the main body of our article with a discussion in Section 6. In Appendix A, we provide some lengthy proofs of the lemma in Section 4.3.

2. Preliminaries and notations

In this article, we consider the semilinear elliptic equation

$$\begin{cases} -\Delta u + f(u) = g & \text{in } \Omega, \\ u + \dfrac{1}{\varepsilon}\dfrac{\partial u}{\partial n} = h & \text{on } \partial\Omega, \end{cases} \tag{2.1}$$

where $\varepsilon \in (0, +\infty]$. The interval for ε includes the cases of Dirichlet boundary condition ($\varepsilon = +\infty$) and Robin boundary condition ($\varepsilon \in (0, +\infty)$). We limit our equation to the following assumption.

Assumption 2.1.

- $\Omega \subset \mathbb{R}^d$ is bounded and smooth.
- f is Lipschitz continuous increasing function and

$$|f(x)| \leq C\left(|x|+1\right)^{\frac{d}{d-2}}$$

for some constant C.

- $g \in L^p$ for some $p \geq 2$.
- There exists a harmonic function w on $\Omega^{'} \supset \bar{\Omega}$ such that

$$w + \frac{1}{\varepsilon}\frac{\partial w}{\partial n} = h \quad \text{on } \partial\Omega.$$

We denote F being the primitive of f, therefore F is a convex function.

Schauder's theory gives.

Proposition 2.1. *Under Assumption* 2.1*, there exists a unique solution* $u^\varepsilon \in W^{2,2}(\Omega)$ *of Eq.* (2.1)*, and* $\|u^\varepsilon\|_{W^{2,2}}$ *can be dominated by* g *and* h.

Proof. See [34, Proposition 2.104]. □

Next, the neural network we used is introduced together with its notation. A deep neural network $u_\phi : \mathbb{R}^d \to \mathbb{R}$ is defined by

$$\begin{aligned}
&u_0(\boldsymbol{x}) = \boldsymbol{x}, \\
&u_\ell(\boldsymbol{x}) = \sigma_\ell\left(A_\ell u_{\ell-1} + b_\ell\right), \quad \ell = 1,2,\ldots,L-1, \\
&u = u_L(\boldsymbol{x}) = A_L u_{L-1} + b_L,
\end{aligned}$$

where $A_\ell \in \mathbb{R}^{N_\ell \times N_{\ell-1}}$, $b_\ell \in \mathbb{R}^{N_\ell}$ and σ_ℓ denotes the activations of layer ℓ. The depth $\mathcal{D}$ and the width $\mathcal{W}$ of neural networks u_ϕ are defined as

$$\mathcal{D} = L, \quad \mathcal{W} = \max\{N_\ell : \ell = 1,2,\ldots,L\}.$$

$\sum_{\ell=1}^L N_\ell$ is called the number of units of u_ϕ, and $\phi = \{A_\ell, b_\ell\}_{\ell=1}^N$ is called the free parameters of the networks.

Definition 2.1. *The class* $\mathcal{N}^\alpha(\mathcal{D},\mathcal{W},\mathcal{B})$ *is the collection of neural networks* u_ϕ *such that:*

(i) Depth and width are $\mathcal{D}$ *and* $\mathcal{W}$*, respectively.*

(ii) The function values of $u_\phi(\boldsymbol{x})$ *and its derivative* $\nabla u_\phi(\boldsymbol{x})$ *are bounded by* $\mathcal{B}$*.*

(iii) Activation functions are given by ReLU^α*, where* α *is a multi-index.*

For example, $\mathcal{N}^2(\mathcal{D},\mathcal{W},\mathcal{B})$ is the class of networks with activation functions as ReLU^2, and $\mathcal{N}^{1,2}(\mathcal{D},\mathcal{W},\mathcal{B})$ is that with activation functions as ReLU^1 or ReLU^2. We may simply use $\mathcal{N}^\alpha$ if there is no confusion.

3. Model construction

Theorem 3.1. *For $v \in H^1(\Omega)$, set*

$$\mathcal{L}^\varepsilon(v) = \int_\Omega \|\nabla v\|_2^2 + 2F(v) - 2vg\,dx + \varepsilon \int_{\partial\Omega} v^2 - 2vh\,ds. \tag{3.1}$$

Then

- *For $\varepsilon \in (0, +\infty)$, there exists one and only one minimizer u^ε of $\mathcal{L}^\varepsilon$. It is the solution to Eq.* (2.1).
- *For $\varepsilon \to +\infty$, let u^∞ be the solution of Eq.* (2.1) *with the Dirichlet boundary condition. Then we have*
$$\|u^\varepsilon - u^\infty\|_{H^1} \le C(\Omega, f, g, h)\frac{1}{\varepsilon}.$$

Proof. If $\varepsilon < +\infty$, the Euler-Lagrange equation of Eq. (3.1) writes

$$\begin{aligned}\delta\mathcal{L}^\varepsilon(v) &= 2\int_\Omega \nabla v^T \nabla \delta v + f(v)\delta v - g\delta v\,dx + 2\varepsilon \int_{\partial\Omega} (v-h)\delta v\,ds \\ &= 2\int_\Omega \big(-\Delta v + f(v) - g\big)\delta v\,dx + 2\varepsilon \int_{\partial\Omega} \left(v + \frac{1}{\varepsilon}\frac{\partial v}{\partial n} - h\right)\delta v\,ds.\end{aligned}$$

All critical points of Eq. (3.1) must be the solutions to Eq. (2.1) and vice versa. Consequently, there exists a unique critical point. Moreover, the convexity of Eq. (3.1) guarantees that this critical point is the minimum.

Considering the Dirichlet boundary condition, we set w^ε as the solution satisfying the following equations:

$$\begin{cases} -\Delta w^\varepsilon + f(u^\infty) = g & \text{in } \Omega, \\ w^\varepsilon + \dfrac{1}{\varepsilon}\dfrac{\partial w^\varepsilon}{\partial n} = h & \text{on } \partial\Omega. \end{cases}$$

It is known that

$$\|w^\varepsilon - u^\infty\|_{H^1} \le C\big(\Omega, f(u^\infty), g, h\big)\frac{1}{\varepsilon},$$

cf. [29, Proposition 2.3]. The classical linear elliptic theory provides us with

$$\int_\Omega \|\nabla(w^\varepsilon - u^\varepsilon)\|_2^2\,dx + \varepsilon\int_{\partial\Omega} (w^\varepsilon - u^\varepsilon)^2\,ds \le \|f(u^\infty) - f(u^\varepsilon)\|_{H^{-1}}^2 \le C(\Omega, f, u^\infty),$$

given $f(u^\varepsilon), f(u^\infty) \in L^2$. Then $\{u^\varepsilon\}$ is a bounded series in H^1. But

$$\begin{aligned}\int_\Omega \|\nabla(u^\infty - u^\varepsilon)\|_2^2\,dx &= \int_\Omega \nabla(u^\infty - u^\varepsilon)^T \nabla u^\infty\,dx - \int_\Omega \nabla(u^\infty - u^\varepsilon)^T \nabla u^\varepsilon\,dx \\ &= \int_\Omega -\left(u^\infty - u^\varepsilon\right)\Delta u^\infty\,dx - \int -\left(u^\infty - u^\varepsilon\right)\Delta u^\varepsilon\,dx\end{aligned}$$

$$
\begin{aligned}
&+\int_{\partial\Omega} \frac{\partial u^\infty}{\partial n}(u^\infty - u^\varepsilon)\, ds - \int_{\partial\Omega} \frac{\partial u^\varepsilon}{\partial n}(u^\infty - u^\varepsilon)\, ds \\
&= -\int_\Omega \left[f(u^\infty) - f(u^\varepsilon)\right](u^\infty - u^\varepsilon)\, dx \\
&+ \frac{1}{\varepsilon}\int_{\partial\Omega} \frac{\partial u^\infty}{\partial n}\frac{\partial u^\varepsilon}{\partial n}\, ds - \frac{1}{\varepsilon}\int_{\partial\Omega}\left(\frac{\partial u^\varepsilon}{\partial n}\right)^2 ds \\
&\le \frac{1}{2\varepsilon}\left[\int_{\partial\Omega}\left(\frac{\partial u^\infty}{\partial n}\right)^2 - \left(\frac{\partial u^\varepsilon}{\partial n}\right)^2 ds\right] \le C(\Omega, f, g, h)\frac{1}{\varepsilon},
\end{aligned} \tag{3.2}
$$

noticing that f is non-decreasing. Next, we proceed with the substitution of u^∞ into $\mathcal{L}^\varepsilon$, yielding

$$
\mathcal{L}^\varepsilon(u^\varepsilon) \le \mathcal{L}^\varepsilon(u^\infty) \le C(\Omega, f, g, h) - \varepsilon\int_{\partial\Omega} h^2.
$$

Therefore,

$$
\varepsilon\int_{\partial\Omega}(u^\varepsilon - h)^2 \le C(\Omega, f, g, h).
$$

In conjunction with Eq. (3.2), we can deduce the desired conclusion. □

Moving on to the second step of the DRM, we substitute the trial function with neural networks

$$
u_\phi^\varepsilon \in \operatorname*{argmin}_{v\in\mathcal{N}^2} \mathcal{L}^\varepsilon(v).
$$

The discrete version of the loss function is expressed as follows [8]:

$$
\begin{aligned}
\widehat{\mathcal{L}}_K^\varepsilon(v) = \frac{|\Omega|}{N}\sum_{i=1}^N &\left[\|\nabla v(X_i)\|_2^2 + 2F\left(v(X_i)\right) - 2g_K(X_i)v(X_i)\right] \\
&+ \varepsilon\frac{|\partial\Omega|}{M}\sum_{j=1}^M \left[v^2(Y_j) - 2v(Y_j)h(Y_j)\right],
\end{aligned} \tag{3.3}
$$

where $\{X_k\}_{k=1}^N \sim U(\Omega)$ i.i.d., $\{Y_j\}_{j=1}^M \sim U(\partial\Omega)$ i.i.d. and $g_K = \min\{g, K\}$ with some constant K. The truncation bound K is introduced to avoid the discontinuity of the discrete $\mathcal{L}^\varepsilon$. Let $\widehat{u}_\phi^\varepsilon$ be the minimizer of the loss over $\mathcal{N}^2$

$$
\widehat{u}_\phi^\varepsilon \in \operatorname*{argmin}_{v\in\mathcal{N}^2} \widehat{\mathcal{L}}_K^\varepsilon(v). \tag{3.4}
$$

In the subsequent discussions, we denote n being the sample number N or M.

Lemma 3.1. *Let $1 \le q < p < \infty$. If $u \in L^p$ and $u_N = u1_{|u|<N}$, then we have*

$$
\|u - u_N\|_{L^q} \le C(u)\frac{1}{N^{p/q-1}}.
$$

Proof. Let $v = |u - u_N|$, $f_v(t) = m(\{x : v(x) < t\})$, where m is the Lebesgue measure

$$\begin{aligned}\|v\|_{L^q}^q &= q\int_0^\infty t^{q-1} f_v(t)\,dt\\ &= q\int_N^\infty t^{p-1} f_v(t) t^{q-p}\,dt\\ &= qN^{q-p}\int_N^\infty t^{p-1} f_v(t)\,dt\\ &= qN^{q-p}\|v\|_{L^p}^p,\end{aligned}$$

which leads to the conclusion. □

Lemma 3.2. *Let $\widehat{u}_\phi^\varepsilon$ be the function defined above, we have*

$$\begin{aligned}\mathcal{L}^\varepsilon\left(\widehat{u}_\phi^\varepsilon\right) - \mathcal{L}^\varepsilon(u^\varepsilon) \le & \underbrace{\inf_{u_\phi \in \mathcal{N}^2}\left[\mathcal{L}^\varepsilon(u_\phi) - \mathcal{L}^\varepsilon(u^\varepsilon)\right]}_{\mathcal{E}_{app}} + \underbrace{2\sup_{u_\phi \in \mathcal{N}^2}\left|\mathcal{L}_K^\varepsilon(u_\phi) - \widehat{\mathcal{L}}_K^\varepsilon(u_\phi)\right|}_{\mathcal{E}_{sta}}\\ & + \underbrace{C(g)\mathcal{B}K^{1-p}}_{\mathcal{E}_{tru}},\end{aligned}$$

which are called the approximation error, the statistical error, and the truncation error respectively.

Proof. Set

$$\mathcal{L}_K^\varepsilon(v) = \int_\Omega \|\nabla v\|_2^2 + 2F(v) - 2vg_K\,dx + \varepsilon\int_{\partial\Omega} v^2 - 2vhds.$$

With the help of Lemma 3.1, we have

$$|\mathcal{L}_K^\varepsilon(u_\phi) - \mathcal{L}^\varepsilon(u_\phi)| \le \int_\Omega |u_\phi(g - g_K)|\,dx \le C(g)\mathcal{B}K^{1-p}$$

for any $u_\phi \in \mathcal{N}^2(\mathcal{D}, \mathcal{W}, \mathcal{B})$. Therefore for any $u \in \mathcal{N}^2$

$$\begin{aligned}&\mathcal{L}^\varepsilon\left(\widehat{u}_\phi^\varepsilon\right) - \mathcal{L}^\varepsilon(u^\varepsilon)\\ =&\left[\mathcal{L}^\varepsilon\left(\widehat{u}_\phi^\varepsilon\right) - \mathcal{L}_K^\varepsilon\left(\widehat{u}_\phi^\varepsilon\right)\right] + \left[\mathcal{L}_K^\varepsilon\left(\widehat{u}_\phi^\varepsilon\right) - \widehat{\mathcal{L}}_K^\varepsilon\left(\widehat{u}_\phi^\varepsilon\right)\right]\\ &+\left[\widehat{\mathcal{L}}_K^\varepsilon\left(\widehat{u}_\phi^\varepsilon\right) - \widehat{\mathcal{L}}_K^\varepsilon\left(u_\phi\right)\right] + \left[\widehat{\mathcal{L}}_K^\varepsilon\left(u_\phi\right) - \mathcal{L}_K^\varepsilon\left(u_\phi\right)\right]\\ &+\left[\mathcal{L}_K^\varepsilon\left(u_\phi\right) - \mathcal{L}^\varepsilon\left(u_\phi\right)\right] + \left[\mathcal{L}^\varepsilon\left(u_\phi\right) - \mathcal{L}^\varepsilon(u^\varepsilon)\right].\end{aligned}$$

The given expression consists of six terms, where the first and fifth terms represent the true error denoted by $\mathcal{E}_{\text{tru}}$, the second and fourth terms correspond to the statistical error denoted by $\mathcal{E}_{\text{sta}}$, the third term is negative, and the infimum of the last term is precisely equal to the approximate error denoted by $\mathcal{E}_{\text{app}}$. □

4. Main results

4.1. Main theorem

For Robin boundary condition, we have

Theorem 4.1. *Let Assumption* 2.1 *holds and* $\widehat{u}_{\phi}^{\varepsilon}$ *be the functions defined in Eq.* (3.4). *We have*

$$\|\widehat{u}_{\phi}^{\varepsilon} - u^{\varepsilon}\|_{H^1} \leq C(\Omega, u^{\varepsilon}, f, g, h)\sqrt{\delta}$$

with

$$\delta \leq C\left(\Omega, u^{\varepsilon}, f, g, h\right) \varepsilon\big(\mathcal{B}^{-2/(d-2)} + \mathcal{W}^{-1/d}\big) + C(g)\mathcal{B}K^{1-p} \\ + C(\Omega, f, h)(K + \varepsilon\mathcal{B})\mathcal{B}\mathcal{D}^2\mathcal{W}\sqrt{\mathcal{D} + \log\mathcal{W}}\sqrt{\frac{\log n}{n}}.$$

Proof. By Lemma 3.2 and the estimation of both the approximation error in Section 4.2 and the statistical error and Section 4.3, we have

$$\mathcal{L}^{\varepsilon}\left(\widehat{u}_{\phi}^{\varepsilon}\right) - \mathcal{L}^{\varepsilon}\left(u^{\varepsilon}\right) \leq C\left(\Omega, u^{\varepsilon}, f, g, h\right)\delta.$$

Let

$$\mathcal{L}_0^{\varepsilon}\left(u\right) = \mathcal{L}^{\varepsilon}\left(u\right) - 2\int_{\Omega} F(u)\, dx.$$

We can deduce that

$$\begin{aligned}
&\mathcal{L}^{\varepsilon}\left(u^{\varepsilon}+v\right) - \mathcal{L}^{\varepsilon}\left(u^{\varepsilon}\right) \\
&= \mathcal{L}_0^{\varepsilon}\left(u^{\varepsilon}+v\right) - \mathcal{L}_0^{\varepsilon}\left(u^{\varepsilon}\right) + 2\int_{\Omega} F\left(u^{\varepsilon}+v\right) - F\left(u^{\varepsilon}\right)\, dx \\
&= \int_{\Omega} \|\nabla\left(u^{\varepsilon}+v\right)\|_2^2 - 2\left(u^{\varepsilon}+v\right) g\, dx + \varepsilon\int_{\partial\Omega} \left(u^{\varepsilon}+v\right)^2 \\
&\quad - 2\left(u^{\varepsilon}+v\right) h\, ds - \mathcal{L}_0^{\varepsilon}\left(u^{\varepsilon}\right) + 2\int_{\Omega} f\left(u^{\varepsilon}+\theta v\right) v\, dx \\
&= \int_{\Omega} \|\nabla v\|_2^2 - 2vg\, dx + \varepsilon\int_{\partial\Omega} v^2 - 2v\left(h - u^{\varepsilon}\right)\, ds \\
&\quad + 2\int_{\Omega} \nabla v^T \nabla u^{\varepsilon}\, dx + 2\int_{\Omega} f\left(u^{\varepsilon}+\theta v\right) v\, dx \\
&= \int_{\Omega} \|\nabla v\|_2^2 dx + \varepsilon\int_{\partial\Omega} v^2\, ds + 2\int_{\Omega} \left[f\left(u^{\varepsilon}+\theta v\right) - f\left(u^{\varepsilon}\right)\right] v\, dx \\
&\geq \int_{\Omega} \|\nabla v\|_2^2\, dx + \varepsilon\int_{\partial\Omega} v^2\, ds \gtrsim C(\Omega, \varepsilon)\|v\|_{H^1}^2,
\end{aligned}$$

where $\theta \in (0,1)$ is the parameter of Lagrange remainder. The second-to-last inequality holds due to the following reasoning: Given that f is a monotonic function, it follows that for any $\theta > 0$, both $f(u^{\varepsilon}+v) - f(u^{\varepsilon})$ and v exhibit the same sign, leading to $[f(u^{\varepsilon}+v) - f(u^{\varepsilon})]v \geq 0$. □

Corollary 4.1. *Under the same condition of the Theorem* 4.1*, if we take*

$$\mathcal{D} = O(1), \quad \mathcal{W} = O(n^{t_1}), \quad \mathcal{B} = O(n^{t_2}), \quad K = O(n^{t_3}),$$

we can acquire that

$$\|\widehat{u}_\phi^\varepsilon - u^\varepsilon\|_{H^1} \leq C\,(\Omega, u^\varepsilon, f, g, h)\, n^{-t_4/2},$$

where

$$t_1 = \frac{d(p-1)}{3dp - 2d - p + 1}, \qquad t_2 = \frac{(d-2)(p-1)}{2(3dp - 2d - p + 1)},$$
$$t_3 = \frac{d}{2(3dp - 2d - p + 1)}, \quad t_4 = \frac{p-1}{3dp - 2d - p + 1}$$

are all positive numbers.

For the Dirichlet boundary condition, we have

Theorem 4.2. *Let Assumption* 2.1 *holds and* $\widehat{u}_\phi^\varepsilon$ *be the functions defined in Eq.* (3.4)*. We have*

$$\|\widehat{u}_\phi^\varepsilon - u^\infty\|_{H^1} \leq C(\Omega, u^\varepsilon, f, g, h)\sqrt{\delta}$$

with

$$\delta \leq C\,(\Omega, u^\varepsilon, f, g, h)\, \varepsilon\big(\mathcal{B}^{-2/(d-2)} + \mathcal{W}^{-1/d}\big) + C(g)\mathcal{B}K^{1-p}$$
$$+ C(\Omega, f, h)(K + \varepsilon\mathcal{B})\mathcal{B}\mathcal{D}^2\mathcal{W}\sqrt{\mathcal{D} + \log\mathcal{W}}\sqrt{\frac{\log n}{n}} + C(\Omega, f, g, h)\frac{1}{\varepsilon}.$$

Proof. It is a direct deduction of Theorem 4.1 and the conclusion in Theorem 3.1. □

Corollary 4.2. *Under the same condition of the Theorem* 4.2*, if we take*

$$\varepsilon = O(n^{t_4/2}), \quad \mathcal{D} = O(1), \quad \mathcal{W} = O(n^{t_1}), \quad \mathcal{B} = O(n^{t_2}), \quad K = O(n^{t_3}).$$

We can acquire that

$$\|\widehat{u}_\phi^\varepsilon - u^\infty\|_{H^1} \leq C(p) n^{-t_4/4},$$

where $\{t_i\}_1^4$ *are the numbers in Corollary* 4.1.

4.2. Approximation error

Theorem 4.3. *If* $\mathcal{D}, \mathcal{W}$*, and* $\mathcal{B}$ *are large enough, we can establish the following result:*

$$\mathcal{E}_{app} = \inf_{u_\phi \in \mathcal{N}^2(\mathcal{D},\mathcal{W},\mathcal{B})} \big[\mathcal{L}^\varepsilon\,(u_\phi) - \mathcal{L}^\varepsilon\,(u^\varepsilon)\big] \leq C(\Omega, u^\varepsilon, f)\varepsilon\big(\mathcal{B}^{-2/(d-2)} + \mathcal{W}^{-1/d}\big).$$

Proof. Our proof is based on some classical polynomial approximation results [35]. Let $\rho \in C_0^\infty$ be a mollifier, i.e.

- $\displaystyle\int \rho\, dx = 1$,
- $\rho \geq 0$,
- $\rho(x) = 0$ for $\|x\| > 1$,
- $\|\rho(x)\|_\infty = 2V_d$ where V_d is the volume of the unit ball.

Take $q = 2d/(d-2)$ and

$$u^\varepsilon_{\mathcal{B}}(x) = \int_y \mathcal{B}^q \rho\left(\frac{y}{\mathcal{B}^{q/d}}\right) u^\varepsilon(x+y)\, dy.$$

We claim that:

- $\|u^\varepsilon_{\mathcal{B}}\|_{W^{1,\infty}(\Omega)} \leq \big(2V_d \|u^\varepsilon\|^q_{W^{2,2}(\Omega)} + 2\big)\mathcal{B}$,
- $\|u^\varepsilon_{\mathcal{B}} - u^\varepsilon\|_{H^1(\Omega)} \leq 2\|u^\varepsilon\|_{W^{2,2}(\Omega)} \mathcal{B}^{-q/d}$,
- $\|u^\varepsilon_{\mathcal{B}}\|_{W^{2,2}(\Omega)} \leq \|u^\varepsilon\|_{W^{2,2}(\Omega)}$.

For the first claim, let

$$E_{\mathcal{B}} = \{x : |u^\varepsilon(x)| > \mathcal{B}\},$$

then we have

$$\|u^\varepsilon 1_{E_{\mathcal{B}}}\|_{L^1} \leq 2\|u^\varepsilon\|^q_{L^q} \mathcal{B}^{1-q}$$

according to Lemma 3.1. Thus

$$\begin{aligned}
|u^\varepsilon_{\mathcal{B}}(x)| &\leq \int_y \mathcal{B}^q \rho\left(\frac{y}{\mathcal{B}^{q/d}}\right) |u^\varepsilon(x+y)|\, dy \\
&= \int_{E_{\mathcal{B}}} \mathcal{B}^q \rho\left(\frac{y}{\mathcal{B}^{q/d}}\right) |u^\varepsilon(x+y)|\, dy + \int_{E^c_{\mathcal{B}}} \mathcal{B}^q \rho\left(\frac{y}{\mathcal{B}^{q/d}}\right) |u^\varepsilon(x+y)|\, dy \\
&\leq \left\|\mathcal{B}^q \rho\left(\frac{y}{\mathcal{B}^{q/d}}\right)\right\|_{L^\infty} \|u^\varepsilon 1_{E_{\mathcal{B}}}\|_{L^1} + \left\|\mathcal{B}^q \rho\left(\frac{y}{\mathcal{B}^{q/d}}\right)\right\|_{L^1} \|u^\varepsilon 1_{E^c_{\mathcal{B}}}\|_{L^\infty} \\
&\leq \left(2V_d \|u^\varepsilon\|^q_{L^q} + 1\right) \mathcal{B}.
\end{aligned}$$

Since $\nabla u^\varepsilon \in L^q$ and the differential operator commutes with the mollifier, the same argument stands for $\nabla u^\varepsilon_{\mathcal{B}}$ as well.

For the second claim, we have

$$\begin{aligned}
\|u^\varepsilon_{\mathcal{B}} - u^\varepsilon\|_{L^2} &\leq \left[\int_x \left(\int_y \rho(y) \big|u^\varepsilon(x + \mathcal{B}^{-q/d} y) - u^\varepsilon(x)\big|\, dy\right)^2 dx\right]^{1/2} \\
&\leq \left[\int_x \left(\int_y \int_{t=0}^{\mathcal{B}^{-q/d}} \rho(y) \left|y^T \nabla u^\varepsilon(x+ty)\right|\, dtdy\right)^2 dx\right]^{1/2}
\end{aligned}$$

$$
\begin{aligned}
&\leq \left[\int_x \left(\int_y \int_{t=0}^{\mathcal{B}^{-q/d}} \rho(y)\|y\|_2 \|\nabla u^\varepsilon(x+ty)\|_2 \, dtdy\right)^2 dx\right]^{1/2} \\
&\leq \int_y \int_{t=0}^{\mathcal{B}^{-q/d}} \left[\int_x \rho(y)^2 \|y\|_2^2 \|\nabla u^\varepsilon(x+ty)\|_2^2 \, dx\right]^{1/2} dtdy \\
&\leq \int_y \int_{t=0}^{\mathcal{B}^{-q/d}} \rho(y)\|y\|_2 \|\nabla u^\varepsilon(x+ty)\|_{L_x^2} \, dtdy \\
&= \|\nabla u^\varepsilon\|_{L^2} \mathcal{B}^{-q/d},
\end{aligned}
$$

which also stands for ∇u^ε since $\nabla u^\varepsilon \in W^{1,2}$.

The third claim is a direct conclusion of Fubini's Theorem.

Next, we demonstrate that any function in $W^{s,p}(\Omega)$ can be approximated by a neural network $\tilde{f}$ in $W^{1,p}$ norm, where $s > 1, \tilde{f} \in \mathcal{N}^2$. Without loss of generality, we assume that $\Omega \subset [0,1]^d$ and extend the function into it. Let

$$
\psi(x;\delta) = \frac{2}{\delta^2}\left[\mathrm{ReLU}^2(x) + \mathrm{ReLU}^2(x+\delta) - 2\mathrm{ReLU}^2\left(x + \frac{\delta}{2}\right)\right].
$$

Then ψ is a ReLU^2-network with width $\{1,3,1\}$ and

$$
\psi(x;\delta) = \begin{cases}
0, & x \in (-\infty, -\delta], \\
\dfrac{2}{\delta^2}(x+\delta)^2, & x \in (-\delta, -\delta/2], \\
-\dfrac{2}{\delta^2}x^2 + 1, & x \in (-\delta/2, 0], \\
1, & x \in (0, \infty).
\end{cases}
$$

For any multi-index $\mathbf{I} \in \{1,2,3,\dots,N\}^d$, we define

$$
\lambda_{\mathbf{I}}(x) = \prod_{j=1}^d \left[\psi\left(x^{(j)} - \frac{\mathbf{I}(j)}{N}; \frac{1}{N}\right) - \psi\left(x^{(j)} - \frac{\mathbf{I}(j)+1}{N}; \frac{1}{N}\right)\right].
$$

Then $\lambda_{\mathbf{I}}$ forms a positive partition of unity in $[0,1]^d$, and it is supported on $U_{\mathbf{I}} = \prod_j [(\mathbf{I}(j)-1)/N, (\mathbf{I}(j)+1)/N]$. Taking note that $\lambda_{\mathbf{I}}$ is a piecewise polynomial function of at most degree d, and observing

$$
f(x)g(x) = \frac{1}{4}\left[\mathrm{ReLU}^2(f+g) + \mathrm{ReLU}^2(-f-g) - \mathrm{ReLU}^2(f-g) - \mathrm{ReLU}^2(g-f)\right].
$$

It can be easily deduced that $\lambda_{\mathbf{I}}$ can be exactly expressed by a ReLU^2-network with width $4d$ and depth $\lceil \log_2 d \rceil + 3$.

Lemma 4.1. *Let $f \in W^{s,p}(\Omega) \cap W^{1,\infty}(\Omega)$, where $s \in (1,2]$. For any $\mathbf{I}$, let $V_{\mathbf{I}} = \prod_j [(I(j)-2)/N, (I(j)+2)/N]$. Then there exists a ReLU^2-neural network $\Psi_{\mathbf{I}}[f]$ with width $\{d, (d^2+3d+2)/2, 1\}$ such that*

- $\|\Psi_{\mathbf{I}}[f]\|_{W^{1,\infty}(V_{\mathbf{I}})} \leq 2\|f\|_{W^{1,\infty}(\Omega)}$,
- $\|f - \psi_{\mathbf{I}}[f]\|_{W^{1,p}(V_{\mathbf{I}})} \leq C(s,p,d)[f]_{s,p,V_{\mathbf{I}}}N^{1-s}$,
- $\|f - \psi_{\mathbf{I}}[f]\|_{L^p(V_{\mathbf{I}})} \leq C(s,p,d)[f]_{s,p,V_{\mathbf{I}}}N^{-s}$.

Proof. By judiciously choosing the values of $\{a_i, b_i\}$, we can ensure that the set $\{\mathrm{ReLU}^2(a_i^T x + b_i)\}$ forms a complete linear basis of $P(V_{\mathbf{I}})$, where

$$P(V_{\mathbf{I}}) = \{\text{All polynomials on } V_{\mathbf{I}} \text{ with degree less than } 2\}.$$

The dimension of $P(V_{\mathbf{I}})$ is $(d^2+3d+2)/2$. The result of polynomial approximation is a direct consequence of the work of [35]. □

Now for any $f \in W^{s,p}(\Omega)$, define

$$\tilde{f} = \sum_{\mathbf{I}} \lambda_{\mathbf{I}} \Psi_{\mathbf{I}}[f].$$

As $\lambda_{\mathbf{I}}$ and $\Psi_{\mathbf{I}}[f]$ are both ReLU^2-neural networks, it follows that $\tilde{f}$ is a neural network with width at most $N^d(2d^2+6d+2)$ and depth at most $\lceil \log_2 d \rceil + 5$. Further, we have

$$\begin{aligned}
\|\partial_i f - \partial_i \tilde{f}\|_{L^p(\Omega)} &= \left\|\partial_i \sum_{\mathbf{I}} \lambda_{\mathbf{I}} \left(f - \Psi_{\mathbf{I}}[f]\right)\right\|_{L^p(\Omega)} \\
&\leq \sum_{\mathbf{I}} \|\partial_i \lambda_{\mathbf{I}} \left(f - \Psi_{\mathbf{I}}[f]\right)\|_{L^p(\Omega)} + \sum_{\mathbf{I}} \|\lambda_{\mathbf{I}} \partial_i \left(f - \Psi_{\mathbf{I}}[f]\right)\|_{L^p(\Omega)} \\
&\leq \sum_{\mathbf{I}} \|\partial_i \lambda_{\mathbf{I}}\|_{L^\infty(V_{\mathbf{I}})} \|(f - \Psi_{\mathbf{I}}[f])\|_{L^p(V_{\mathbf{I}})} + \sum_{\mathbf{I}} \|\partial_i \left(f - \Psi_{\mathbf{I}}[f]\right)\|_{L^p(V_{\mathbf{I}})} \\
&\leq \sum_{\mathbf{I}} 2NC(s,p,d)[f]_{s,p,V_{\mathbf{I}}}N^{-s} + \sum_{\mathbf{I}} C(s,p,d)[f]_{s,p,V_{\mathbf{I}}}N^{1-s} \\
&\leq C(s,p,d)[f]_{s,p,\Omega}N^{1-s}.
\end{aligned}$$

Summary it all up, now we have proven the Lemma 4.2.

Lemma 4.2. *For any $f \in W^{s,p}(\Omega) \cap W^{1,\infty}(\Omega)$, $s \in (1,2]$, there exists a $f_\phi \in \mathcal{N}^2(\mathcal{W},\mathcal{D},\mathcal{B})$ s.t.*

$$\|f - f_\phi\|_{W^{1,p}} \leq C(s,p,d,\Omega)[f]_{s,p,\Omega}\mathcal{W}^{(1-s)/d}$$

as long as $\mathcal{B} \geq 2\|f\|_{W^{1,\infty}(\Omega)}$, $\mathcal{D} \geq \lceil \log_2 d \rceil + 5$ and $\mathcal{W}$ is large enough.

Now, we turn our attention to the approximation error. Let L_f be the Lipschitz constant of f. Apparently, we have

$$\begin{aligned}
\mathcal{L}^\varepsilon\left(u_\varepsilon + v\right) &= \int_\Omega \|\nabla\left(u_\varepsilon + v\right)\|_2^2 + 2F(u_\varepsilon + v) - 2\left(u_\varepsilon + v\right) g \, dx \\
&\quad + \varepsilon \int_{\partial\Omega} \left(u_\varepsilon + v\right)^2 - 2\left(u_\varepsilon + v\right) h \, ds
\end{aligned}$$

$$
\begin{aligned}
&= \int_\Omega \|\nabla u_\varepsilon\|_2^2 + \|\nabla v\|_2^2 - 2\Delta u^\varepsilon v + 2F(u_\varepsilon + v) - 2\left(u_\varepsilon + v\right) g \, dx \\
&\quad + \varepsilon \int_{\partial\Omega} \left(u_\varepsilon + v\right)^2 - 2\left(u_\varepsilon + v\right) h \, ds + \int_{\partial\Omega} 2v \frac{\partial u^\varepsilon}{\partial n} \, ds \\
&= \int_\Omega \|\nabla u_\varepsilon\|_2^2 + \|\nabla v\|_2^2 + 2F(u_\varepsilon + v) - 2f(u^\varepsilon)v - 2u_\varepsilon g \, dx \\
&\quad + \varepsilon \int_{\partial\Omega} \left(u_\varepsilon + v\right)^2 - 2\left(u_\varepsilon + v\right) h + 2v \frac{1}{\varepsilon} \frac{\partial u^\varepsilon}{\partial n} \, ds \\
&= \int_\Omega \|\nabla v\|_2^2 + 2\left[F(u_\varepsilon + v) - F(u_\varepsilon) - f(u^\varepsilon)v\right] dx \\
&\quad + \varepsilon \int_{\partial\Omega} v^2 + 2\left(u_\varepsilon + \frac{1}{\varepsilon}\frac{\partial u^\varepsilon}{\partial n} - h\right) v ds + \mathcal{L}^\varepsilon\left(u^\varepsilon\right) \\
&\le \int_\Omega \|\nabla v\|_2^2 \, dx + 2L_f \int_\Omega v^2 \, dx + \varepsilon \int_{\partial\Omega} v^2 \, ds + \mathcal{L}^\varepsilon\left(u^\varepsilon\right).
\end{aligned}
$$

It leads to

$$
\begin{aligned}
\mathcal{E}_{app} &= \inf_{u_\phi \in \mathcal{N}^2} \left[\mathcal{L}^\varepsilon\left(u_\phi\right) - \mathcal{L}^\varepsilon\left(u^\varepsilon\right)\right] \\
&\le C(\Omega, f)\varepsilon \inf_{u_\phi \in \mathcal{N}^2} \|u_\phi - u^\varepsilon\|_{H^1}^2 \\
&\le C(\Omega, f)\varepsilon \Big(\inf_{u_\phi \in \mathcal{N}^2} \|u_\phi - u_{\mathcal{B}}^\varepsilon\|_{H^1}^2 + \|u_{\mathcal{B}}^\varepsilon - u^\varepsilon\|_{H^1}^2 \Big) \\
&\le C(\Omega, f, u^\varepsilon)\varepsilon \left(\mathcal{W}^{-1/d} + \mathcal{B}^{-q/d}\right).
\end{aligned}
$$

The first inequality arises from the continuity of $\mathcal{L}^\varepsilon$. The second inequality is obtained by applying the triangle inequality. By incorporating Lemma 4.2 with the definition of $u_{\mathcal{B}}^\varepsilon$, we deduce the final inequality. □

4.3. Statistical error

This section focuses on bounding the statistical error

$$
\mathcal{E}_{sta} = 2 \sup_{u_\phi \in \mathcal{N}^2} \left|\mathcal{L}_K^\varepsilon(u_\phi) - \widehat{\mathcal{L}}_K^\varepsilon(u_\phi)\right|.
$$

We begin by decomposing the statistical error into five parts to estimate them separately. Thus we set

$$
\sup_{u_\phi \in \mathcal{N}^2} \left|\mathcal{L}_K^\varepsilon(u_\phi) - \widehat{\mathcal{L}}_K^\varepsilon(u_\phi)\right| \le \sum_{j=1}^{5} \sup_{u_\phi \in \mathcal{N}^2} \left|\mathcal{L}_j^\varepsilon(u_\phi) - \widehat{\mathcal{L}}_j^\varepsilon(u_\phi)\right|, \tag{4.1}
$$

where

$$
\mathcal{L}_1^\varepsilon(v) = |\Omega| \mathop{\mathbb{E}}_{X \sim U(\Omega)} \left[\|\nabla v(X)\|_2^2\right], \qquad \widehat{\mathcal{L}}_1^\varepsilon(v) = \frac{|\Omega|}{N} \sum_{i=1}^{N} \left[\|\nabla v(X_i)\|_2^2\right],
$$

$$\mathcal{L}_2^\varepsilon(v) = 2|\Omega| \mathop{\mathbb{E}}_{X\sim U(\Omega)} [F(v(X))] , \qquad \widehat{\mathcal{L}}_2^\varepsilon(v) = 2\frac{|\Omega|}{N}\sum_{i=1}^{N} [F(v(X_i))] ,$$

$$\mathcal{L}_3^\varepsilon(v) = -2|\Omega| \mathop{\mathbb{E}}_{X\sim U(\Omega)} [v(X)g_K(X)] , \qquad \widehat{\mathcal{L}}_3^\varepsilon(v) = -2\frac{|\Omega|}{N}\sum_{i=1}^{N} [v(X_i)g_K(X_i)] ,$$

$$\mathcal{L}_4^\varepsilon(v) = \varepsilon|\partial\Omega| \mathop{\mathbb{E}}_{Y\sim U(\partial\Omega)} \left[v^2(Y)\right] , \qquad \widehat{\mathcal{L}}_4^\varepsilon(v) = \varepsilon\frac{|\partial\Omega|}{M}\sum_{j=1}^{M} \left[v^2(Y_j)\right] ,$$

$$\mathcal{L}_5^\varepsilon(v) = -2\varepsilon|\partial\Omega| \mathop{\mathbb{E}}_{Y\sim U(\partial\Omega)} [v(Y)h(Y)] , \qquad \widehat{\mathcal{L}}_5^\varepsilon(v) = -2\varepsilon\frac{|\partial\Omega|}{M}\sum_{j=1}^{M} [v(Y_j)h(Y_j)] .$$

Here, $U(\Omega)$ and $U(\partial\Omega)$ represent the uniform distribution on Ω and $\partial\Omega$, respectively. Given n i.i.d samples $\mathbf{Z}_n = \{Z_i\}_{i=1}^n$ from a uniform distribution, we can utilize the Rademacher complexity to assess the capacity of a given function class $\mathcal{N}$ restricted on n random samples $\mathbf{Z}_n$.

Here is the sketch of bounding the statistic error: First we define the Rademacher complexity $\mathfrak{R}$, covering number $\mathcal{C}_\infty$ and pseudo-dimension Pdim from Definitions 4.1 to 4.5. Then Lemma 4.3 bounds every part of the statistical error by the Rademacher complexity. Lemma 4.4 dominates the Rademacher complexity by an integral of the covering number, which is controlled by the pseudo-dimension through the Lemma 4.5. Finally, Lemma 4.6 expresses the order of the pseudo-dimension through the width and depth of the neural networks. It leads to the Theorem 4.4 combining them all together.

For ease of reading, we state the above lemmas and theorems first and leave their proofs in Appendix A.

Definition 4.1. *The Rademacher complexity of a set $A \subseteq \mathrm{R}^n$ is defined as*

$$\mathfrak{R}(A) := \mathbb{E}_{\mathbf{Z}_n,\sigma_i}\left[\sup_{a\in A}\frac{1}{n}\Big|\sum_i \sigma_i a_i\Big|\right] ,$$

where $\{\sigma_i\}_{i=1}^n$ are n i.i.d Rademacher variables with $\mathbb{P}(\sigma_i = 1) = \mathbb{P}(\sigma_i = -1) = 1/2$. The Rademacher complexity of function class $\mathcal{N}$ associated with random sample $\mathbf{Z}_n$ is defined as

$$\mathfrak{R}(\mathcal{N}) := \mathbb{E}_{\mathbf{Z}_n,\sigma_i}\left[\sup_{u\in \mathcal{N}}\frac{1}{n}\Big|\sum_i \sigma_i u(Z_i)\Big|\right] .$$

Definition 4.2. *Suppose that $W \subset \mathbb{R}^n$. For any $\varepsilon > 0$, let $V \subset \mathbb{R}^n$ be an ε-cover of W with respect to the distance d_∞, that is, for any $u \in W$, there exists a $v \in V$ such that $d_\infty(u,v) < \varepsilon$, where d_∞ is defined by*

$$d_\infty(u,v) := \|u-v\|_\infty .$$

The covering number $\mathcal{C}(\varepsilon, W, d_\infty)$ is defined to be the minimum cardinality among all ε-cover of W with respect to the distance d_∞.

Definition 4.3. *Suppose that $\mathcal{N}$ is a class of functions from Ω to $\mathbb{R}$. Given n sample $\mathbf{Z}_n = (Z_1, Z_2, \cdots, Z_n) \in \Omega^n$, $\mathcal{N}|_{\mathbf{Z}_n} \subset \mathbb{R}^n$ is defined by*

$$\mathcal{N} \mid_{\mathbf{Z}_n} = \{(u(Z_1), u(Z_2), \cdots, u(Z_n)) : u \in \mathcal{N}\}.$$

The uniform covering number $\mathcal{C}_\infty(\varepsilon, \mathcal{N}, n)$ is defined by

$$\mathcal{C}_\infty(\varepsilon, \mathcal{N}, n) = \max_{\mathbf{Z}_n \in \Omega^n} \mathcal{C}\left(\varepsilon, \mathcal{N} \mid_{\mathbf{Z}_n}, d_\infty\right).$$

Definition 4.4. *Let $\mathcal{N}$ be a set of functions from X to $\mathbb{R}$. Suppose that $S = \{x_1, x_2, \cdots, x_n\} \subset X$. We say that S is pseudo shattered by $\mathcal{N}$ if there exists $y_1, \cdots, y_n$ such that for any $b \in \{0,1\}^n$, there exists a $u \in \mathcal{N}$ satisfying*

$$\operatorname{sign}\big(u(x_i) - y_i\big) = b_i, \quad i = 1, 2, \ldots, n,$$

and we say that $\{y_i\}_{i=1}^n$ witness the shattering.

Definition 4.5. *The pseudo-dimension of $\mathcal{N}$, denoted as $\operatorname{Pdim}(\mathcal{N})$, is defined to be the maximum cardinality among all the sets pseudo-shattered by $\mathcal{N}$.*

The following lemma elucidates the relationship between statistical errors and the Rademacher complexity of the function class.

Lemma 4.3. *We have*

$$\begin{aligned}
&\mathbb{E}_{\{X_i\}_{i=1}^N} \sup_{u \in \mathcal{N}^2} \left|\mathcal{L}_1^\varepsilon(u) - \widehat{\mathcal{L}}_1^\varepsilon(u)\right| \le |\Omega| \mathfrak{R}\left(\mathcal{N}^{1,2}\right), \\
&\mathbb{E}_{\{X_i\}_{i=1}^N} \sup_{u \in \mathcal{N}^2} \left|\mathcal{L}_2(u) - \widehat{\mathcal{L}}_2(u)\right| \le 2|\Omega| \|f\|_\infty \mathfrak{R}\left(\mathcal{N}^2\right), \\
&\mathbb{E}_{\{X_i\}_{i=1}^N} \sup_{u \in \mathcal{N}^2} \left|\mathcal{L}_3^\varepsilon(u) - \widehat{\mathcal{L}}_3^\varepsilon(u)\right| \le 2|\Omega| K \mathfrak{R}\left(\mathcal{N}^2\right), \\
&\mathbb{E}_{\{Y_i\}_{i=1}^M} \sup_{u \in \mathcal{N}^2} \left|\mathcal{L}_4^\varepsilon(u) - \widehat{\mathcal{L}}_4^\varepsilon(u)\right| \le 2\varepsilon |\partial\Omega| \mathcal{B} \mathfrak{R}\left(\mathcal{N}^2\right), \\
&\mathbb{E}_{\{Y_i\}_{i=1}^M} \sup_{u \in \mathcal{N}^2} \left|\mathcal{L}_5^\varepsilon(u) - \widehat{\mathcal{L}}_5^\varepsilon(u)\right| \le 4\varepsilon |\partial\Omega| \|h\|_\infty \mathfrak{R}\left(\mathcal{N}^2\right).
\end{aligned}$$

The proof is given in the Appendix A.

To bound the Rademacher complexity by using the covering numbers defined in Definition 4.5, we refer to Dudley's classical result.

Lemma 4.4 (Dudley's Entropy Formula [15])**.** *Assume that $0 \in \mathcal{N}$ and the diameter of $\mathcal{N}$ is less than $\mathcal{B}$, i.e., $\|u\|_{L^\infty(\Omega)} \le \mathcal{B}, \forall u \in \mathcal{N}$. Then*

$$\mathfrak{R}(\mathcal{N}) \le \inf_{0 < \delta < \mathcal{B}} \left(4\delta + \frac{12}{\sqrt{n}} \int_\delta^{\mathcal{B}} \sqrt{\log(2\mathcal{C}(\varepsilon, \mathcal{N}, n))} \mathrm{d}\varepsilon\right).$$

The proof is given in Appendix A.

The subsequent lemma uncovers the interrelation between covering numbers and pseudo-dimension. In pursuit of an upper bound on the Pdim of piecewise polynomial functions, we refer to the conclusions presented in [4], which are adaptable to the function class defined in our formulation.

Lemma 4.5. *Let $\mathcal{N}$ be a set of real functions from a domain X to the bounded interval $[0,\mathcal{B}]$. Let $\varepsilon > 0$. Then*

$$\mathcal{C}(\varepsilon,\mathcal{N},n) \le \sum_{i=1}^{\mathrm{Pdim}(\mathcal{N})} \binom{n}{i} \left(\frac{\mathcal{B}}{\varepsilon}\right)^i,$$

which is less than $(en\mathcal{B}/(\varepsilon\cdot\mathrm{Pdim}(\mathcal{N})))^{\mathrm{Pdim}(\mathcal{N})}$ for $n \ge \mathrm{Pdim}(\mathcal{N})$.

Proof of the lemma can be found in [1, Theorem 12.2].

Lemma 4.6. *Let $\mathcal{N}$ be a set of functions that can be implemented by a neural network with its depth at most $\mathcal{D}$ and its width at most $\mathcal{W}$, and the activation function in each unit is the ReLU or the ReLU^2. Then*

$$\mathrm{Pdim}(\mathcal{N}) \le C_1\mathcal{D}^2\mathcal{W}^2(\mathcal{D}+\log\mathcal{W}).$$

The proof is given in the Appendix A. Particularly, based on Eq. (A.1), we have

$$\mathrm{Pdim}(\mathcal{N}^{1,2}) \le C_2 d^2\mathcal{D}^4\mathcal{W}^2(\mathcal{D}+\log\mathcal{W}),$$

where C_1, C_2 are constant independent of d, if $\mathcal{W} > d$.

With the help of these preparations above, the statistical error can easily be bounded by a simple calculation.

Theorem 4.4.

$$\mathcal{E}_{sta} \le C(\Omega,f,h)\,(K+\varepsilon\mathcal{B})\,\mathcal{B}\mathcal{D}^2\mathcal{W}\sqrt{\mathcal{D}+\log\mathcal{W}}\sqrt{\frac{\log n}{n}}.$$

Proof. Combining Lemmas 4.3-4.6, it can be obtained that

$$\sup_{u\in\mathcal{N}^2}\left|\mathcal{L}^\varepsilon(u)-\widehat{\mathcal{L}}^\varepsilon(u)\right| \le C(\Omega)\mathcal{B}\mathfrak{R}\left(\mathcal{N}^{1,2}\right)+C(\Omega,f,h)\,(\varepsilon\mathcal{B}+K)\,\mathfrak{R}\left(\mathcal{N}^2\right),$$

where

$$\begin{aligned}\mathfrak{R}(\mathcal{N}) &\le 4\delta+\frac{12}{\sqrt{n}}\int_\delta^{\mathcal{B}}\sqrt{\log(2\mathcal{C}(\varepsilon,\mathcal{N},n))}\mathrm{d}\varepsilon\\ &\le 4\delta+\frac{12}{\sqrt{n}}\mathcal{B}\sqrt{\log(2\mathcal{C}(\delta,\mathcal{N},n))}\\ &\le 4\delta+\frac{12}{\sqrt{n}}\mathcal{B}\sqrt{\log 2+\mathrm{Pdim}(\mathcal{N})\log\left(\frac{en\mathcal{B}}{\delta\,\mathrm{Pdim}(\mathcal{N})}\right)}\end{aligned}$$

for that apparently $\mathcal{C}(\varepsilon, \mathcal{N}, n)$ is a decreasing function of ε. By choosing

$$\delta = \mathcal{B}\sqrt{\frac{\mathrm{Pdim}(\mathcal{N})}{n}},$$

it can be acquired that

$$\begin{aligned}\mathfrak{R}\left(\mathcal{N}\right) &\le 4\delta \left\{4 + 3\sqrt{\frac{\log 2}{\mathrm{Pdim}(\mathcal{N})}} + \frac{3\sqrt{6}}{2}\sqrt{\log\left(\frac{n}{\mathrm{Pdim}(\mathcal{N})}\right)}\right\} \\ &\le C\mathcal{B}\sqrt{\mathrm{Pdim}(\mathcal{N})}\sqrt{\frac{\log n}{n}}.\end{aligned}$$

Then by Lemma 4.6, the proof is finished. □

5. Numerical experiments

As indicated in the introduction, conventional grid-dependent PDE numerical solutions encounter challenges in high dimensions due to the curse of dimensionality. However, through our comprehensive analysis, we have theoretically established a dimension-independent convergence analysis for the Deep Ritz method. To further validate the efficacy of our theory in high dimensions, we conducted a series of numerical experiments.

In this section, we provide examples of approximating solutions to semilinear elliptic equations, including Dirichlet problems with homogeneous and inhomogeneous boundary conditions.

We utilize a neural network consisting of two blocks for solving the equations. Each block comprises two linear transformations, two activation functions, and a residual connection, which can be viewed as a four-layer deep neural network. We use the Adagrad or Adam algorithm with a stepwise decreasing learning rate to minimize Eq. (3.1) during optimization.

All experiments and implementations are conducted in Python 3.9.12 with PyTorch on CentOS, using two Intel(R) Xeon(R) E5-2640 v4 x86_64 Processors clocked at 2.40 GHz and a Nvidia Tesla V100 GPU with 16 GB of graphics memory.

5.1. Dirichlet problem with homogeneous boundary condition

We first consider the following homogeneous boundary condition Dirichlet problem:

$$\begin{cases} -\Delta u + S(u) = g_1(x) & \text{on } \Omega, \\ Tu = 0 & \text{on } \partial\Omega, \end{cases} \tag{5.1}$$

where $\Omega = [0,1]^{10}$ and $S(x)$ is the sigmoid function

$$S(x) = \frac{1}{1+e^{-x}}$$

and

$$g_1(x) = 8 \sum_{\substack{1 \le i \le d \\ i \ne j}} \prod_{j=1}^{d} x_j(1-x_j) + S\left(\prod_{i=1}^{d} 4x_i(1-x_i)\right),$$

$S(x)$ is a nonlinear function in $L^\infty(\Omega)$, and is both Lipschitz continuous and non-decreasing. Consequently, it satisfies Assumption 2.1. The exact solution of Eq. (5.1) is given by $u_1(x) = \prod_{i=1}^{d} 4x_i(1-x_i)$.

We commence by examining the scenario where $d = 2$. A multitude of numerical experiments have been conducted across various network parameter configurations. These empirical findings offer robust evidence supporting the convergence properties of our algorithm.

Initially, we focus on assessing the influence of sampling size n on the solution accuracy. We set the network parameters as $\mathcal{W} = 120$ and $\mathcal{D} = 4$, while employing different sampling sizes to address problem (5.1). The outcomes are illustrated in Fig. 1, and as anticipated in Fig. 2, a larger sampling size leads to higher accuracy. This observation aligns with the conclusion of Theorem 4.2.

It is worth noting that in the aforementioned numerical experiment, we only sampled $\{X_i\}_{i=1}^n$ once at the outset of the calculation. When n is small, the algorithm exhibits poor performance and converges slowly. To enhance both the accuracy and efficiency of the algorithm, we implemented a strategy of resampling $\{X_i\}_{i=1}^n$ after each

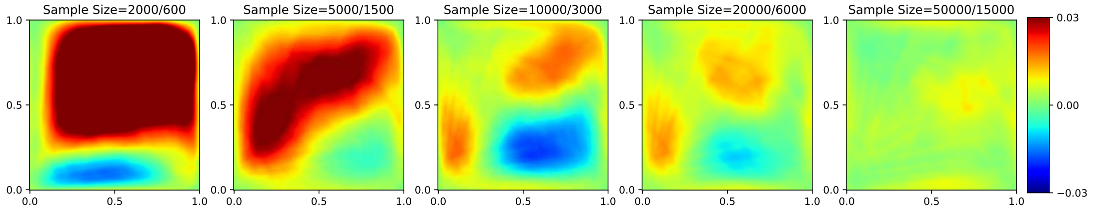


Figure 1: (Dirichlet problem with homogeneous boundary condition, $d = 2$). The pointwise difference between the Deep Ritz solution u_{DRM} and the true solution $u_1(x)$ under different sample sizes n.

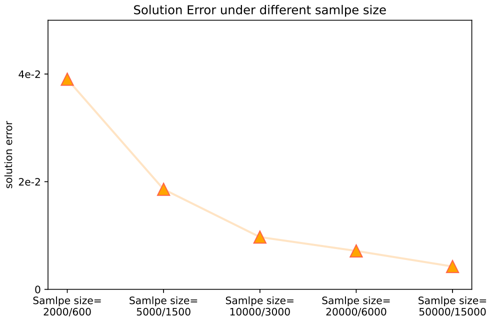


Figure 2: (Dirichlet problem with homogeneous boundary condition, $d = 2$). The L_2 solution error under different sample sizes n.

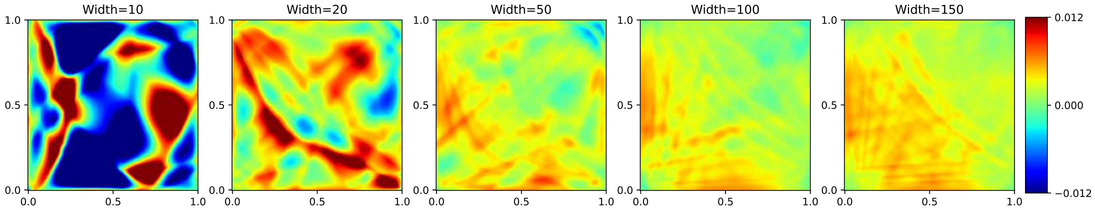
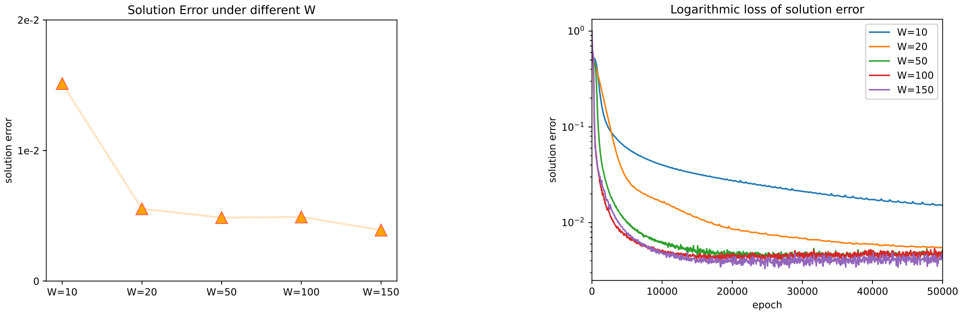


Figure 3: (Dirichlet problem with homogeneous boundary condition, $d = 2$). The pointwise difference between the Deep Ritz solution u_{DRM} and the true solution $u_1(x)$ under different network widths $\mathcal{W}$.



a) L_2 solution error

b) The error during the training process

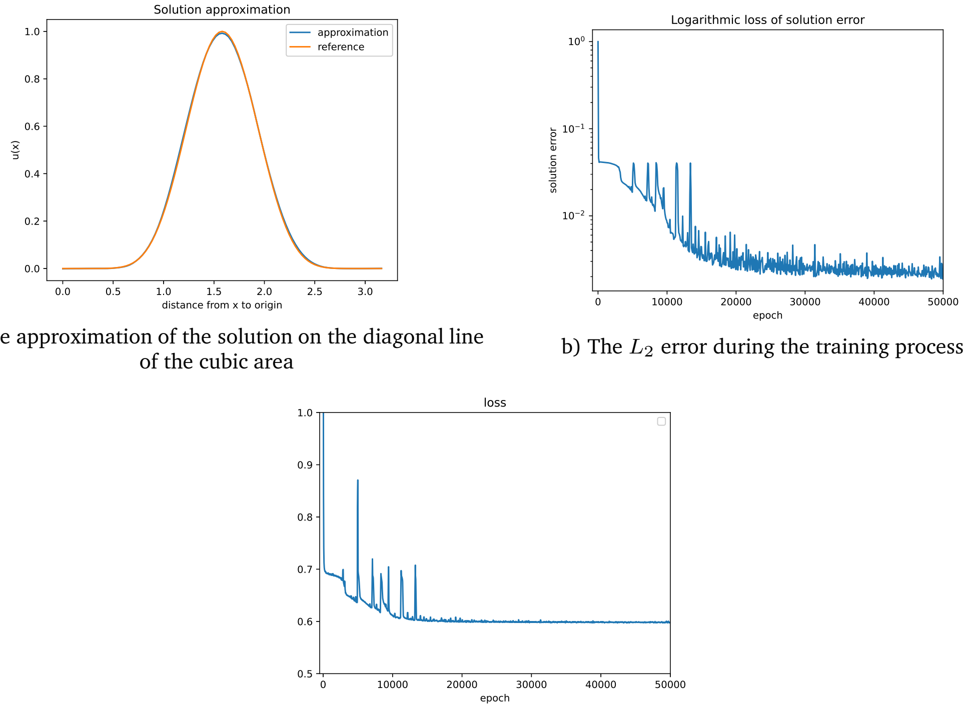
Figure 4: (Dirichlet problem with homogeneous boundary condition, $d = 2$). The convergence performance of Deep Ritz Method under different network widths $\mathcal{W}$.

training step or several steps. This process is analogous to batch training on an infinite sample set. By adopting this approach, we can effectively achieve a larger n with relatively modest computational resources, subsequently improving the algorithm's generalization performance. Our extensive experiments have also substantiated the efficacy of this technique. Unless specified otherwise, the sampling size n mentioned in our subsequent experiments refers to the resampled n.

The influence of the network width $\mathcal{W}$ on the method is depicted in the Figs. 3 and 4. We conducted experiments with $\mathcal{D} = 4$ and $n = 150000$. The experimental results demonstrate that a wider network can attain superior accuracy and converge more rapidly. This finding is consistent with our previous conclusion.

We then investigate the case where $d = 10$. Due to the high dimensionality and the non-linearity introduced by $S(x)$, solving the equation becomes notably challenging. Traditional computational techniques such as Finite Difference Method (FDM) and Finite Element Method (FEM) have proven ineffective. Despite these challenges, the Deep Ritz method remains effective, as depicted in Fig. 5. We plot the approximation and exact solutions on the diagonal of Ω as shown in Fig. 5 (a). The trend of L^2 error with epoch is shown in Fig. 5 (b), and the loss is shown in Fig. 5 (c).

The parameters for this numerical experiment are configured as follows: $\varepsilon = 2000$, depth $\mathcal{D} = 4$, width $\mathcal{W} = 80$, sample size as 200000, and boundary sample size as 80000.



a) The approximation of the solution on the diagonal line of the cubic area

b) The L_2 error during the training process

c) The loss during the training process

Figure 5: (Dirichlet problem with homogeneous boundary condition, $d = 10$). Due to the inability to directly visualize functions in high-dimensional spaces, we exclusively present their values along the diagonal of the cubic area and subsequently compare these values with the corresponding values of the true solution.

We employed the Adam algorithm to minimize the objective function, initializing the learning rate at $1.8e-3$. An equidistant learning rate reduction strategy was employed, where the learning rate was reduced by a factor of 0.9 every 5000 step.

5.2. Dirichlet problem with inhomogeneous boundary condition

To illustrate the generality of our theory, we consider the following inhomogeneous Dirichlet problem:

$$\begin{cases} -\Delta u + S(u) = g_2(x) & \text{on } \Omega, \\ Tu = Tg_2(x) & \text{on } \partial\Omega, \end{cases} \tag{5.2}$$

where $\Omega = [-1,1]^{10}$, and

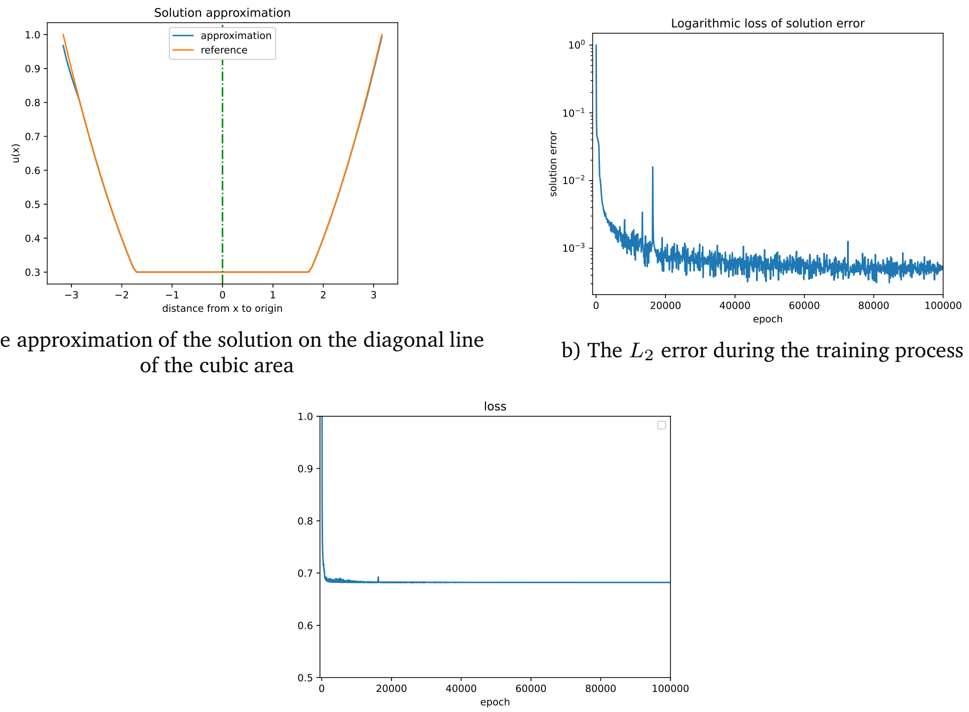
$$g_2(x) = \begin{cases} \dfrac{2}{d} + S\left(\left(\dfrac{1}{d}\sum_{i=1}^{d} x_i\right)^2\right), & \left|\dfrac{1}{d}\sum_{i=1}^{d} x_i\right| > \sqrt{0.3}, \\ S(0.3), & \left|\dfrac{1}{d}\sum_{i=1}^{d} x_i\right| \leq \sqrt{0.3}. \end{cases}$$

The solution of the above inhomogeneous problem is given by

$$u_2(x) = \begin{cases} \left(\dfrac{1}{d}\displaystyle\sum_{i=1}^{d} x_i\right)^2, & \left|\dfrac{1}{d}\displaystyle\sum_{i=1}^{d} x_i\right| > \sqrt{0.3}, \\ 0.3, & \left|\dfrac{1}{d}\displaystyle\sum_{i=1}^{d} x_i\right| \leq \sqrt{0.3}. \end{cases}$$

In contrast to the previous equation, this problem (5.2) is not only a non-homogeneous boundary problem but also involves a non-smooth solution.

The parameters for this numerical experiment are configured as follows: $\varepsilon = 2000$, depth $\mathcal{D} = 4$, width $\mathcal{W} = 150$, sample size as 150000, boundary sample size as 40000, and other parameters are the same as before. As shown in Fig. 6, the proposed method also accurately approximates the solution in this setting.



a) The approximation of the solution on the diagonal line of the cubic area

b) The L_2 error during the training process

c) The loss during the training process

Figure 6: Dirichlet problem with inhomogeneous boundary condition, $d = 10$.

6. Discussion

In this paper, we investigate the use of ResNet with ReLU^2 activations for solving semilinear elliptic problems. We propose a general formulation for computing the solution to semilinear elliptical equations based on a penalized variational form. The penalized variational form is then solved using the Deep Ritz Method. We derive an upper bound on the errors between the estimated solutions and true ones in terms of the depth $\mathcal{D}$ and width $\mathcal{W}$ of the ReLU^2 ResNet, as well as the number of training samples n. Our simulation results demonstrate the effectiveness of the proposed method in circumventing the curse of dimensionality and validate our theoretical results.

Appendix A

In this appendix, we present comprehensive proofs of several lemmas introduced in Section 4.3. The statistical error analysis of the Deep Ritz Method follows a standardized process, and thus, it is omitted from the main body of the text.

Proof of Lemma 4.3. We will present the proof in two parts, each of which can be obtained separately using two distinct facts. The first fact is that the Rademacher complexity can be passed on through a Lipschitz continuous function. This fact enables us to establish the last four inequalities.

Lemma A.1. *Suppose that $\psi : \mathbb{R}^d \times \mathbb{R} \to \mathbb{R}$, $(x, y) \mapsto \psi(x, y)$ is ℓ-Lipschitz continuous on y for all x. Let $\mathcal{N}$ be a class of functions on Ω and $\psi\circ\mathcal{N} = \{\psi\circ u : x \mapsto \psi(x, u(x)), u \in \mathcal{N}\}$. Then*

$$\mathfrak{R}(\psi \circ \mathcal{N}) \leq \ell\, \mathfrak{R}(\mathcal{N}).$$

For the deduction of this statement, we cite [24, Corollary 3.17].

Obviously, the Lipschitz constant for the 2nd, 3rd, 4th, and 5th terms are $2\|f\|_\infty$, $2K$, $2\varepsilon\mathcal{B}$, and $4\varepsilon\|h\|_\infty$, respectively. Therefore their conclusion can be yielded directly in the same manner.

The first term requires special treatment due to the fact that the ∇ operator is not Lipschitz continuous. This consideration follows directly from the following claim.

Claim A.1. Let u be a function implemented by a ReLU^2 network with depth $\mathcal{D}$ and width $\mathcal{W}$. Then $\|\nabla u\|_2^2$ can be implemented by a ReLU-ReLU^2 network with depth $\mathcal{D}+3$ and width $d(\mathcal{D}+2)\mathcal{W}$.

Denote ReLU and ReLU^2 as σ_1 and σ_2, respectively. As long as we show that each partial derivative $D_i u, i = 1, 2, \ldots, d$ can be implemented by a ReLU-ReLU^2 network respectively, we can easily obtain the network desired since $\|\nabla u\|_2^2 = \sum_{i=1}^d |D_i u|^2$ and the square function can be implemented by $x^2 = \sigma_2(x) + \sigma_2(-x)$.

Now we show that for any $i = 1, 2, \ldots, d$, $D_i u$ can be implemented by a ReLU-ReLU^2 network. We will focus on explaining the first two layers in detail, as the process for the layers with $k \geq 3$ is similar and can be derived through induction. For the first layer, since $\sigma_2'(x) = 2\sigma_1(x)$, we have for any $q = 1, 2, \ldots, n_1$

$$D_i u_q^{(1)} = D_i \sigma_2 \left(\sum_{j=1}^d a_{qj}^{(1)} x_j + b_q^{(1)} \right) = 2\sigma_1 \left(\sum_{j=1}^d a_{qj}^{(1)} x_j + b_q^{(1)} \right) \cdot a_{qi}^{(1)}.$$

Hence, $D_i u_q^{(1)}$ can be implemented by a ReLU-ReLU^2 network with depth 2 and width 1. For the second layer,

$$D_i u_q^{(2)} = D_i \sigma_2 \left(\sum_{j=1}^{n_1} a_{qj}^{(2)} u_j^{(1)} + b_q^{(2)} \right) = 2\sigma_1 \left(\sum_{j=1}^{n_1} a_{qj}^{(2)} u_j^{(1)} + b_q^{(2)} \right) \cdot \sum_{j=1}^{n_1} a_{qj}^{(2)} D_i u_j^{(1)}.$$

Since $\sigma_1(\sum_{j=1}^{n_1} a_{qj}^{(2)} u_j^{(1)} + b_q^{(2)})$ and $\sum_{j=1}^{n_1} a_{qj}^{(2)} D_i u_j^{(1)}$ can be implemented by $\mathrm{ReLU}-\mathrm{ReLU}^2$ subnetworks, respectively, and the multiplication can also be implemented by

$$\begin{aligned} x \cdot y &= \frac{1}{4} \left[(x+y)^2 - (x-y)^2 \right] \\ &= \frac{1}{4} \left[\sigma_2(x+y) + \sigma_2(-x-y) - \sigma_2(x-y) - \sigma_2(-x+y) \right]. \end{aligned}$$

We conclude that $D_i u_q^{(2)}$ can be implemented by a ReLU-ReLU^2 network. We have

$$\mathcal{D} \left(\sigma_1 \left(\sum_{j=1}^{n_1} a_{qj}^{(2)} u_j^{(1)} + b_q^{(2)} \right) \right) = 3, \quad \mathcal{W} \left(\sigma_1 \left(\sum_{j=1}^{n_1} a_{qj}^{(2)} u_j^{(1)} + b_q^{(2)} \right) \right) \leq \mathcal{W}$$

and

$$\mathcal{D} \left(\sum_{j=1}^{n_1} a_{qj}^{(2)} D_i u_j^{(1)} \right) = 2, \quad \mathcal{W} \left(\sum_{j=1}^{n_1} a_{qj}^{(2)} D_i u_j^{(1)} \right) \leq \mathcal{W}.$$

Thus $\mathcal{D}(D_i u_q^{(2)}) = 4$, $\mathcal{W}(D_i u_q^{(2)}) \leq \max\{2\mathcal{W}, 4\}$.

Now we apply induction for layers $k \geq 3$. For the third layer,

$$D_i u_q^{(3)} = D_i \sigma_2 \left(\sum_{j=1}^{n_2} a_{qj}^{(3)} u_j^{(2)} + b_q^{(3)} \right) = 2\sigma_1 \left(\sum_{j=1}^{n_2} a_{qj}^{(3)} u_j^{(2)} + b_q^{(3)} \right) \cdot \sum_{j=1}^{n_2} a_{qj}^{(3)} D_i u_j^{(2)}.$$

Since

$$\mathcal{D} \left(\sigma_1 \left(\sum_{j=1}^{n_2} a_{qj}^{(3)} u_j^{(2)} + b_q^{(3)} \right) \right) = 4, \quad \mathcal{W} \left(\sigma_1 \left(\sum_{j=1}^{n_2} a_{qj}^{(3)} u_j^{(2)} + b_q^{(3)} \right) \right) \leq \mathcal{W}$$

and

$$\mathcal{D} \left(\sum_{j=1}^{n_2} a_{qj}^{(3)} D_i u_j^{(2)} \right) = 4, \quad \mathcal{W} \left(\sum_{j=1}^{n_1} a_{qj}^{(3)} D_i u_j^{(2)} \right) \leq \max\{2\mathcal{W}, 4\mathcal{W}\} = 4\mathcal{W},$$

we conclude that $D_i u_q^{(3)}$ can be implemented by a ReLU-ReLU^2 network and

$$\mathcal{D} \left(D_i u_q^{(3)} \right) = 5, \quad \mathcal{W} \left(D_i u_q^{(3)} \right) \leq \max\{5\mathcal{W}, 4\} = 5\mathcal{W}.$$

We assume that $D_i u_q^{(k)}, q = 1, 2, \ldots, n_k$ can be implemented by a ReLU-ReLU^2 network and

$$\mathcal{D}\big(D_i u_q^{(k)}\big) = k+2, \quad \mathcal{W}\big(D_i u_q^{(3)}\big) \le (k+2)\mathcal{W}.$$

For the $(k+1)$-th layer,

$$\begin{aligned} D_i u_q^{(k+1)} &= D_i \sigma_2 \left(\sum_{j=1}^{n_k} a_{qj}^{(k+1)} u_j^{(k)} + b_q^{(k+1)} \right) \\ &= 2\sigma_1 \left(\sum_{j=1}^{n_k} a_{qj}^{(k+1)} u_j^{(k)} + b_q^{(k+1)} \right) \cdot \sum_{j=1}^{n_k} a_{qj}^{(k+1)} D_i u_j^{(k)}. \end{aligned}$$

Since

$$\begin{aligned} &\mathcal{D}\left(\sigma_1 \left(\sum_{j=1}^{n_k} a_{qj}^{(k+1)} u_j^{(k)} + b_q^{(k+1)} \right)\right) = k+2, \\ &\mathcal{W}\left(\sigma_1 \left(\sum_{j=1}^{n_k} a_{qj}^{(k+1)} u_j^{(k)} + b_q^{(k+1)} \right)\right) \le \mathcal{W}, \\ &\mathcal{D}\left(\sum_{j=1}^{n_k} a_{qj}^{(k+1)} D_i u_j^{(k)} \right) = k+2, \\ &\mathcal{W}\left(\sum_{j=1}^{n_k} a_{qj}^{(k+1)} D_i u_j^{(k)} \right) \le \max\{(k+2)\mathcal{W}, 4\mathcal{W}\} = (k+2)\mathcal{W}, \end{aligned}$$

we conclude that $D_i u_q^{(k+1)}$ can be implemented by a ReLU-ReLU^2 network and

$$\begin{aligned} &\mathcal{D}\left(D_i u_q^{(k+1)} \right) = k+3, \\ &\mathcal{W}\left(D_i u_q^{(k+1)} \right) \le \max\{(k+3)\mathcal{W}, 4\} = (k+3)\mathcal{W}. \end{aligned}$$

Hence, we derive that $D_i u = D_i u_1^{\mathcal{D}}$ can be implemented by a ReLU-ReLU^2 network and $\mathcal{D}(D_i u) = \mathcal{D}+2$, $\mathcal{W}(D_i u) \le (\mathcal{D}+2)\mathcal{W}$. Finally, we obtain

$$\mathcal{D}\left(\|\nabla u\|^2\right) = \mathcal{D}+3, \quad \mathcal{W}\left(\|\nabla u\|^2\right) \le d\,(\mathcal{D}+2)\,\mathcal{W}. \tag{A.1}$$

The proof is complete. □

Proof of Lemma 4.4. First, we introduce Massart's finite class lemma whose proof can be found in [6].

Lemma A.2 (Massart's Finite Class Lemma [6]). *For any finite set $V \in \mathbb{R}^n$ with diameter $D = \sum_{v \in V} \|v\|_2$, then*

$$\mathbb{E}_{\sigma_i}\left[\sup_{v \in V} \frac{1}{n}\left|\sum_i \sigma_i v_i\right|\right] \leq \frac{D}{n}\sqrt{2\log(2|V|)},$$

where $\{\sigma_i\}_{i=1}^n$ are the Rademacher variables defined the same as in Definition 4.1.

Set $\varepsilon_j = 2^{-k+1}B$. We denote $\mathcal{F}_k$ as an ε_k-cover of $\mathcal{F}$ and $|\mathcal{F}_k| = \mathcal{C}(\varepsilon_k, \mathcal{F}, \|\cdot\|_\infty)$. Hence, for any $u \in \mathcal{F}$, there exists $u_k \in \mathcal{F}_k$ such that $\|u - u_k\|_\infty \leq \varepsilon_k$. Let K be a positive integer determined later. We have

$$\begin{aligned}
&\mathbb{E}_{\{\sigma_i, Z_i\}_{i=1}^n}\left[\sup_{u \in \mathcal{F}} \frac{1}{n}\left|\sum_{i=1}^n \sigma_i u\left(Z_i\right)\right|\right] \\
&= \mathbb{E}_{\{\sigma_i, Z_i\}_{i=1}^n}\Bigg[\sup_{u \in \mathcal{F}} \frac{1}{n}\Bigg|\sum_{i=1}^n \sigma_i\big(u\left(Z_i\right) - u_K\left(Z_i\right)\big) \\
&\qquad + \sum_{j=1}^{K-1}\sum_{i=1}^n \sigma_i\big(u_{j+1}\left(Z_i\right) - u_j\left(Z_i\right)\big) + \sum_{i=1}^n \sigma_i u_1\left(Z_i\right)\Bigg|\Bigg] \\
&\leq \mathbb{E}_{\{\sigma_i, Z_i\}_{i=1}^n}\left[\sup_{u \in \mathcal{F}} \frac{1}{n}\left|\sum_{i=1}^n \sigma_i\big(u\left(Z_i\right) - u_K\left(Z_i\right)\big)\right|\right] \\
&\quad + \sum_{j=1}^{K-1} \mathbb{E}_{\{\sigma_i, Z_t\}_{i=1}^n}\left[\sup_{u \in \mathcal{F}} \frac{1}{n}\left|\sum_{i=1}^n \sigma_i\big(u_{j+1}\left(Z_i\right) - u_j\left(Z_i\right)\big)\right|\right] \\
&\quad + \mathbb{E}_{\{\sigma_i, Z_t\}_{k=1}^n}\left[\sup_{u \in \mathcal{F}} \frac{1}{n}\left|\sum_{i=1}^n \sigma_i u_1\left(Z_i\right)\right|\right].
\end{aligned}$$

Since $0 \in \mathcal{F}$, we can choose $\mathcal{F}_1 = \{0\}$ to eliminate the third term. For the first term,

$$\begin{aligned}
&\mathbb{E}_{\{\sigma_i, Z_i\}_{t=1}^n}\left[\sup_{u \in \mathcal{F}} \frac{1}{n}\left|\sum_{i=1}^n \sigma_i\big(u\left(Z_i\right) - u_K\left(Z_i\right)\big)\right|\right] \\
&\leq \mathbb{E}_{\{\sigma_i, Z_i\}_{i=1}^n}\left[\sup_{u \in \mathcal{F}} \frac{1}{n}\sum_{i=1}^n |\sigma_i| \left\|u - u_K\right\|_\infty\right] \leq \varepsilon_K.
\end{aligned}$$

For the second term, defining $v_i^j = u_{j+1}(Z_i) - u_j(Z_i)$, and applying Lemma A.2, we have

$$\begin{aligned}
&\sum_{j=1}^{K-1} \mathbb{E}_{\{\sigma\}_{i=1}^n}\left[\sup_{u \in \mathcal{F}} \frac{1}{n}\left|\sum_{i=1}^n \sigma_i\big(u_{j+1}\left(Z_i\right) - u_j\left(Z_i\right)\big)\right|\right] \\
&= \sum_{j=1}^{K-1} \mathbb{E}_{\{\sigma_t\}_{t=1}^n}\left[\sup_{v \in V_j} \frac{1}{n}\left|\sum_{i=1}^n \sigma_i v_i^j\right|\right] \leq \sum_{j=1}^{K-1} \frac{D_j}{n}\sqrt{2\log\left(2\left|V_j\right|\right)}.
\end{aligned}$$

By the definition of V_j, we know that $|V_j| \leq |\mathcal{F}_j||\mathcal{F}_{j+1}| \leq |\mathcal{F}_{j+1}|^2$ and

$$\begin{aligned}\|V\|_2 &= \left(\sum_{i=1}^{n} |u_{j+1}(Z_i) - u_j(Z_i)|^2\right)^{1/2} \\ &\leq \sqrt{n}\,\|u_{j+1} - u_j\|_\infty \\ &\leq \sqrt{n}\,\|u_{j+1} - u\|_\infty + \sqrt{n}\,\|u_j - u\|_\infty \\ &= \sqrt{n}\varepsilon_{j+1} + \sqrt{n}\varepsilon_j = 3\sqrt{n}\varepsilon_{j+1}.\end{aligned}$$

Hence,

$$\begin{aligned}&\sum_{j=1}^{K-1} \mathbb{E}_{\{\sigma_i,Z_t\}_{t=1}^n}\left[\sup_{u\in\mathcal{F}} \frac{1}{n}\left|\sum_{i=1}^{n} \sigma_i\big(u_{j+1}(Z_i) - u_j(Z_i)\big)\right|\right] \\ \leq &\sum_{j=1}^{K-1} \frac{D_j}{n}\sqrt{2\log(2|V_j|)} \leq \sum_{j=1}^{K-1} \frac{3\varepsilon_{j+1}}{\sqrt{n}}\sqrt{2\log\left(2|\mathcal{F}_{j+1}|^2\right)}.\end{aligned}$$

Now we obtain

$$\begin{aligned}&\mathbb{E}_{\{\sigma_i,Z_i\}_{i=1}^n}\left[\sup_{u\in\mathcal{F}} \frac{1}{n}\left|\sum_{i=1}^{n} \sigma_i u(Z_i)\right|\right] \\ \leq\ &\varepsilon_K + \sum_{j=1}^{K-1} \frac{6\varepsilon_{j+2}}{\sqrt{n}}\sqrt{2\log\left(2|\mathcal{F}_{j+1}|^2\right)} \\ =\ &\varepsilon_K + \frac{6}{\sqrt{n}}\sum_{j=1}^{K-1} (\varepsilon_{j+1} - \varepsilon_{j+2})\sqrt{2\log\left(2\mathcal{C}(\varepsilon_{j+1}, \mathcal{F}, \|\cdot\|_\infty)^2\right)} \\ \leq\ &\varepsilon_K + \frac{6}{\sqrt{n}}\int_{\varepsilon_{K+1}}^{B/2} \sqrt{2\log\left(2\mathcal{C}(\varepsilon, \mathcal{F}, \|\cdot\|_\infty)^2\right)}d\varepsilon.\end{aligned}$$

The lemma can be concluded by selecting an appropriate K such that $\varepsilon_{K+2} < \delta \leq \varepsilon_{K+1}$ for any $0 < \delta < B/2$. □

Proof of Lemma 4.6. The sketch of the proof is given as follows: Firstly, the VCdim and pseudo-shattering is introduced as a lower bound of the Pdim. Then the VCdim for a polynomial is estimated through a lemma in [1]. Based on the conclusion above, the proof can be finished by a deduction similar to [4, Theorem 6].

Here are the definitions of pseudo-shattering and VCdim.

Definition A.1. *Let $\mathcal{N}$ be a set of functions from $X = \Omega(\partial\Omega)$ to $\{0,1\}$. Suppose that $S = \{x_1, x_2, \cdots, x_n\} \subset X$. We say that S is shattered by $\mathcal{N}$ if for any $b \in \{0,1\}^n$, there exists a $u \in \mathcal{N}$ satisfying*

$$u(x_i) = b_i, \quad i = 1, 2, \ldots, n.$$

Definition A.2. *The VC-dimension of $\mathcal{N}$, denoted as $\operatorname{VCdim}(\mathcal{N})$, is defined to be the maximum cardinality among all sets shattered by $\mathcal{N}$.*

Lemma A.3 is introduced to estimate the Pdim for polynomials. The proof can be found in [1, Theorem 8.3].

Lemma A.3. *Let $p_1, \cdots, p_m$ be polynomials with n variables of degree at most d. If $n \leq m$, then*

$$|\{(\operatorname{sign}(p_1(x)), \cdots, \operatorname{sign}(p_m(x))) : x \in \mathbb{R}^n\}| \leq 2\left(\frac{2emd}{n}\right)^n.$$

The argument follows from the proof of [4, Theorem 6]. The result stated here is somewhat stronger than [4, Theorem 6] since $\operatorname{VCdim}(\operatorname{sign}(\mathcal{N})) \leq \operatorname{Pdim}(\mathcal{N})$.

We consider a new set of functions

$$\widetilde{\mathcal{N}} = \{\widetilde{u}(x,y) = \operatorname{sign}(u(x) - y) : u \in \mathcal{N}\}.$$

It is clear that $\operatorname{Pdim}(\mathcal{N}) \leq \operatorname{VCdim}(\widetilde{\mathcal{N}})$. We now bound the VC-dimension of $\widetilde{\mathcal{N}}$. Denoting $\mathcal{M}$ as the total number of parameters (weights and biases) in the neural networks implementing functions in $\mathcal{N}$, our objective is to derive a uniform bound for

$$K_{\{x_i\},\{y_i\}}(m) := \left|\{(\operatorname{sign}(u(x_1,a) - y_1), \ldots, \operatorname{sign}(u(x_m,a) - y_m)) : a \in \mathbb{R}^{\mathcal{M}}\}\right|,$$

over all $\{x_i\}_{i=1}^m \subset X$ and $\{y_i\}_{i=1}^m \subset \mathbb{R}$. Actually, the maximum of $K_{\{x_i\},\{y_i\}}(m)$ over all $\{x_i\}_{i=1}^m \subset X$ and $\{y_i\}_{i=1}^m \subset \mathbb{R}$ is the growth function $\mathcal{G}_{\widetilde{\mathcal{N}}}(m)$.

In order to apply Lemma A.3, we partition the parameter space $\mathbb{R}^{\mathcal{M}}$ into several subsets to ensure that in each subset $u(x_i,a) - y_i$ is a polynomial with respect to a without any breakpoints. In fact, our partition is the same as the partition in [4]. Denote the partition as $\{P_1, P_2, \cdots, P_N\}$ with some integer N satisfying

$$N \leq \prod_{i=1}^{\mathcal{D}-1} 2\left(\frac{2emk_i(1+(i-1)2^{i-1})}{\mathcal{M}_i}\right)^{\mathcal{M}_i}, \tag{A.2}$$

where k_i and $\mathcal{M}_i$ denote the number of units at the i-th layer and the total number of parameters at the inputs to units in all the layers up to layer i of the neural networks implementing functions in $\mathcal{N}$, respectively. See [4] for the construction of the partition. Obviously we have

$$K_{\{x_i\},\{y_i\}}(m) \leq \sum_{i=1}^{N} |\{(\operatorname{sign}(u(x_1,a) - y_1), \cdots, \operatorname{sign}(u(x_m,a) - y_m)) : a \in P_i\}|. \tag{A.3}$$

Note that $u(x_i,a) - y_i$ is a polynomial with respect to a with degree the same as the degree of $u(x_i,a)$, which is equal to $1 + (\mathcal{D}-1)2^{\mathcal{D}-1}$ as shown in [4]. Hence, by Lemma A.3, we have

$$\begin{aligned} &|\{(\operatorname{sign}(u(x_1,a) - y_1), \cdots, \operatorname{sign}(u(x_m,a) - y_m)) : a \in P_i\}| \\ &\leq 2\left(\frac{2em(1+(\mathcal{D}-1)2^{\mathcal{D}-1})}{\mathcal{M}_{\mathcal{D}}}\right)^{\mathcal{M}_{\mathcal{D}}}. \end{aligned} \tag{A.4}$$

Combining Eq. (A.2)-Eq. (A.4) yields

$$K_{\{x_i\},\{y_i\}}(m) \leq \prod_{i=1}^{\mathcal{D}} 2\left(\frac{2emk_i(1+(i-1)2^{i-1})}{\mathcal{M}_i}\right)^{\mathcal{M}_i}.$$

We then have

$$\mathcal{G}_{\widetilde{\mathcal{N}}}(m) \leq \prod_{i=1}^{\mathcal{D}} 2\left(\frac{2emk_i(1+(i-1)2^{i-1})}{\mathcal{M}_i}\right)^{\mathcal{M}_i},$$

since the maximum of $K_{\{x_i\},\{y_i\}}(m)$ over all $\{x_i\}_{i=1}^m \subset X$ and $\{y_i\}_{i=1}^m \subset \mathbb{R}$ is the growth function $\mathcal{G}_{\widetilde{\mathcal{N}}}(m)$. Using algebras as that of the proof of [4, Theorem 6], we obtain

$$\mathrm{Pdim}(\mathcal{N}) \leq C\left(\mathcal{D}^2\mathcal{W}^2\log\mathcal{U} + \mathcal{D}^3\mathcal{W}^2\right) \leq C\left(\mathcal{D}^2\mathcal{W}^2\left(\mathcal{D}+\log\mathcal{W}\right)\right),$$

where $\mathcal{U}$ refers to the number of units of the neural networks implementing functions in $\mathcal{N}$. □

Acknowledgments

We would like the thank the anonymous referees and associated editor for their useful comments and suggestions, which have led to considerable improvements in the paper.

This work is supported by the National Key Research and Development Program of China (Grant No. 2020YFA0714200), the National Nature Science Foundation of China (Grant Nos. 12125103,12071362,12371424,12371441) and supported by the Fundamental Research Funds for the Central Universities. The numerical calculations have been done at the Supercomputing Center of Wuhan University.

References

[1] M. Anthony and P. L. Bartlett, *Neural Network Learning: Theoretical Foundations*, Cambridge University Press, 2009.

[2] G. Bao, X. Ye, Y. Zang, and H. Zhou, *Numerical solution of inverse problems by weak adversarial networks*, Inverse Problems 36 (2020), 115003.

[3] W. Bao, D. Jaksch, and P. A. Markowich, *Numerical solution of the Gross-Pitaevskii equation for Bose-Einstein condensation*, J. Comput. Phys. 187 (2003), 318–342.

[4] P. L. Bartlett, N. Harvey, C. Liaw, and A. Mehrabian, *Nearly-tight vc-dimension and pseudodimension bounds for piecewise linear neural networks*, J. Mach. Learn. Res. 20 (2019), 2285–2301.

[5] C. Basdevant, M. Deville, P. Haldenwang, J. Lacroix, J. Ouazzani, R. Peyret, P. Orlandi, and A. Patera, *Spectral and finite difference solutions of the Burgers equation*, Comput. & Fluids 14 (1986), 23–41.

[6] S. Boucheron, G. Lugosi, and P. Massart, *Concentration Inequalities: A Nonasymptotic Theory of Independence*, Oxford University Press, 2013.

[7] F. Chen, J. Huang, C. Wang, and H. Yang, *Friedrichs learning: Weak solutions of partial differential equations via deep learning*, 2021.

[8] J. Chen, R. Du, P. Li, and L. Lyu, *Quasi-Monte Carlo sampling for solving partial differential equations by deep neural networks*, Numer. Math. Theor. Meth. Appl. 14 (2021), 377–404.

[9] J. Chen, R. Du, and K. Wu, *A comparison study of deep Galerkin method and deep Ritz method for elliptic problems with different boundary conditions*, Commun. Math. Res. 36 (2020), 23.

[10] Y. Chen, L. Lu, G. E. Karniadakis, and L. Dal Negro, *Physics-informed neural networks for inverse problems in nano-optics and metamaterials*, Optics Express 28 (2020), 11618–11633.

[11] T. De Ryck, A. D. Jagtap, and S. Mishra, *Error estimates for physics informed neural networks approximating the Navier-Stokes equations*, arXiv:2203.09346, (2022).

[12] P. Dondl, J. Müller, and M. Zeinhofer, *Uniform convergence guarantees for the deep Ritz method for nonlinear problems*, Adv Cont Discr Mod 2022 49 (2022), 1–19.

[13] J. Douglas, *A method of numerical solution of the problem of plateau*, Ann. of Math. (2) 29 (1927), 180–188.

[14] C. Duan et al., *Convergence rate analysis for deep Ritz method*, Commun. Comput. Phys. 31 (2022), 1020–1048.

[15] R. M. Dudley, *The sizes of compact subsets of Hilbert space and continuity of Gaussian processes*, J. Funct. Anal. 1(3) (1967), 290–330.

[16] P. Grohs, F. Hornung, A. Jentzen, and P. Von Wurstemberger, *A proof that artificial neural networks overcome the curse of dimensionality in the numerical approximation of black-scholes partial differential equations*, arXiv:1809.02362, (2018).

[17] J. Han, J. Lu, and M. Zhou, *Solving high-dimensional eigenvalue problems using deep neural networks: A diffusion Monte Carlo like approach*, J. Comput. Phys. 423 (2020), 109792.

[18] P. Henry-Labordere, N. Oudjane, X. Tan, N. Touzi, and X. Warin, *Branching diffusion representation of semilinear PDEs and Monte Carlo approximation*, Annales de l'Institut Henri Poincaré - Probabilités et Statistiques 55 (2019), 184–210.

[19] J. Huang et al., *An augmented Lagrangian deep learning method for variational problems with essential boundary conditions*, Commun. Comput. Phys. 31 (2022), 966–986.

[20] A. D. Jagtap and G. E. Karniadakis, *Extended physics-informed neural networks (XPINNs): A generalized space-time domain decomposition based deep learning framework for nonlinear partial differential equations*, in: AAAI Spring Symposium: MLPS, (2021), 2002–2041.

[21] Y. Jiao et al., *A rate of convergence of physics informed neural networks for the linear second order elliptic PDEs*, Commun. Comput. Phys. 31 (2022), 1272–1295.

[22] Y. Jiao, Y. Lai, Y. Wang, H. Yang, and Y. Yang, *Convergence analysis of the deep Galerkin method for weak solutions*, arXiv:2302.02405, (2023).

[23] X. Jin, S. Cai, H. Li, and G. E. Karniadakis, *Nsfnets (Navier-Stokes flow nets): Physics-informed neural networks for the incompressible Navier-Stokes equations*, J. Comput. Phys. 426 (2021), 109951.

[24] M. Ledoux and M. Talagrand, *Probability in Banach Spaces: Isoperimetry and Processes*, Springer Science & Business Media, 2013.

[25] H. Li and L. Ying, *A semigroup method for high dimensional elliptic PDEs and eigenvalue problems based on neural networks*, J. Comput. Phys. 453 (2022), 110939.

[26] J. Li, W. Zhang, and J. Yue, *A deep learning Galerkin method for the second-order linear*

elliptic equations, Int. J. Numer. Anal. Model. 18 (2021).

[27] J. Lu and Y. Lu, *A priori generalization error analysis of two-layer neural networks for solving high dimensional Schrödinger eigenvalue problems*, Comm. Amer. Math. Soc. 2 (2022), 1–21.

[28] J. Lu, Y. Lu, and M. Wang, *A priori generalization analysis of the deep Ritz method for solving high dimensional elliptic equations*, in: Proceedings of Machine Learning Research 134 (2021), 1–46.

[29] B. Maury, *Numerical analysis of a finite element/volume penalty method*, SIAM J. Numer. Anal. 47 (2009), 1126–1148.

[30] Y. L. Ming et al., *Deep Nitsche method: Deep Ritz method with essential boundary conditions*, Commun. Comput. Phys. 29 (2021), 1365–1384.

[31] S. Mishra and R. Molinaro, *Estimates on the generalization error of physics-informed neural networks for approximating a class of inverse problems for PDEs*, IMA J. Numer. Anal. 42 (2022), 981–1022.

[32] M. Raissi, P. Perdikaris, and G. E. Karniadakis, *Physics informed deep learning (Part i): Data-driven solutions of nonlinear partial differential equations*, arXiv:1711. 10561, (2017).

[33] 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*, J. Comput. Phys. 378 (2019), 686–707.

[34] T. Roubiek, *Nonlinear Partial Differential Equations with Applications*, Birkhauser, 2013.

[35] A. Sanchez and R. Arcangeli, *Estimation of the best polynomial approximation error and the Lagrange interpolation error in fractional-order Sobolev spaces*, Numer. Math. 45 (1984), 301–321.

[36] J. Sirignano and K. Spiliopoulos, *DGM: A deep learning algorithm for solving partial differential equations*, J. Comput. Phys. 375 (2018), 1339–1364.

[37] X. Warin, *Nesting Monte Carlo for high-dimensional non-linear PDEs*, Monte Carlo Methods Appl. 24 (2018), 225–247.

[38] E. Weinan, J. Han, and A. Jentzen, *Deep learning-based numerical methods for high-dimensional parabolic partial differential equations and backward stochastic differential equations*, Commun. Math. Stat. 4 (2017), 349–380.

[39] E. Weinan and B. Yu, *The deep Ritz method: A deep learning-based numerical algorithm for solving variational problems*, Commun. Math. Stat. 1 (2018), 1–12.

[40] Y. Zang, G. Bao, X. Ye, and H. Zhou, *Weak adversarial networks for high-dimensional partial differential equations*, J. Comput. Phys. 411 (2020), 109409.