Deep Ritz Methods for Laplace Equations with Dirichlet Boundary Condition

Chenguang Duan[1], Yuling Jiao[2], Yanming Lai[1], Xiliang Lu[1], Qimeng Quan[1] and Jerry Zhijian Yang[2,*]

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

Abstract. Deep Ritz methods (DRM) have been proven numerically to be efficient in solving partial differential equations. In this paper, we present a convergence rate in H^1 norm for deep Ritz methods for Laplace equations with Dirichlet boundary condition, where the error depends on the depth and width in the deep neural networks and the number of samples explicitly. Further we can properly choose the depth and width in the deep neural networks in terms of the number of training samples. The main idea of the proof is to decompose the total error of DRM into three parts, that is approximation error, statistical error and the error caused by the boundary penalty. We bound the approximation error in H^1 norm with ReLU^2 networks and control the statistical error via Rademacher complexity. In particular, we derive the bound on the Rademacher complexity of the non-Lipschitz composition of gradient norm with ReLU^2 network, which is of immense independent interest. We also analyze the error inducing by the boundary penalty method and give a prior rule for tuning the penalty parameter.

AMS subject classifications: 62G05, 65N12, 65N15, 68T07

Key words: Deep Ritz methods, convergence rate, Dirichlet boundary condition, approximation error, Rademacher complexity.

1 Introduction

Partial differential equations (PDEs) are one of the fundamental mathematical models in studying a variety of phenomenons arising in science and engineering. There have been

---

761

established many conventional numerical methods successfully for solving PDEs in the case of low dimension ($d \leq 3$), particularly the finite element method [8,9,25,37,44]. However, one will encounter some difficulties in both of theoretical analysis and numerical implementation when extending conventional numerical schemes to high-dimensional PDEs. The classic analysis of convergence, stability and any other properties will be trapped into troublesome situation due to the complex construction of finite element space [8, 9]. Moreover, in the term of practical computation, the scale of the discrete problem will increase exponentially with respect to the dimension.

Motivated by the well-known fact that deep learning method for high-dimensional data analysis has been achieved great successful applications in discriminative, generative and reinforcement learning [18, 22, 42], solving high dimensional PDEs with deep neural networks becomes an extremely potential approach and has attracted much attentions [2, 6, 21, 31, 38, 43, 48, 50]. Roughly speaking, these works can be divided into three categories. The first category is using deep neural network to improve classical numerical methods, see for example [19, 24, 45, 47]. In the second category, the neural operator is introduced to learn mappings between infinite-dimensional spaces with neural networks [1, 28, 29]. For the last category, one utilizes deep neural networks to approximate the solutions of PDEs directly including physics-informed neural networks (PINNs) [38], deep Ritz method (DRM) [48] and weak adversarial networks (WAN) [50]. PINNs is based on residual minimization for solving PDEs [2, 31, 38, 43]. Proceed from the variational form, [48–50] propose neural-network based methods related to classical Ritz and Galerkin method. In [50], WAN are proposed inspired by Galerkin method. Based on Ritz method, [48] proposes the DRM to solve variational problems corresponding to a class of PDEs.

1.1   Related works and contributions

The idea using neural networks to solve PDEs goes back to 1990's [12,26]. Although there are great empirical achievements in recent several years, a challenging and interesting question is to provide a rigorous error analysis such as finite element method. Several recent efforts have been devoted to making processes along this line, see for example [14, 15, 23, 30, 32, 34, 36, 41, 46]. In [32], least squares minimization method with two-layer neural networks is studied, the optimization error under the assumption of over-parametrization and generalization error without the over-parametrization assumption are analyzed. In [30, 49], the generalization error bounds of two-layer neural networks are derived via assuming that the exact solutions lie in spectral Barron space.

Dirichlet boundary condition corresponds to a constrained minimization problem, which may cause some difficulties in computation. The penalty method has been applied in finite element methods and finite volume method [4, 33]. It is also been used in deep PDEs solvers [38, 48, 49] since it is not easy to construct a network with given values on the boundary. We also apply penalty method to DRM with ReLU^2 activation functions, and obtain the error estimation in this work. The main contribution are listed as follows:

- We derive a bound on the approximation error of deep ReLU^2 network in H^1 norm, which is of independent interest, see Theorem 3.2. That is, for any $u_\lambda^* \in H^2(\Omega)$, there exist a ReLU^2 network $\bar{u}_{\tilde{\phi}}$ with depth $\mathcal{D} \leq \lceil \log_2 d \rceil + 3$, width $\mathcal{W} \leq \mathcal{O}\left(4d \lceil \frac{1}{\epsilon} - 4 \rceil^d \right)$ (where d is the dimension), such that

$$\left\| u_\lambda^* - \bar{u}_{\tilde{\phi}} \right\|_{H^1(\Omega)}^2 \leq \epsilon^2 \quad \text{and} \quad \left\| Tu_\lambda^* - T\bar{u}_{\tilde{\phi}} \right\|_{L^2(\partial\Omega)}^2 \leq C_d \epsilon^2.$$

- We establish a bound on the statistical error in DRM with the tools of pseudo-dimension, especially we give a bound on

$$\mathbb{E}_{Z_i, \sigma_i, i=1,\dots,n} \left[\sup_{u_\phi \in \mathcal{N}^2} \frac{1}{n} \left| \sum_i \sigma_i \left\| \nabla u_\phi(Z_i) \right\|^2 \right| \right],$$

i.e., the Rademacher complexity of the non-Lipschitz composition of gradient norm and ReLU^2 network, via calculating the Pseudo dimension of networks with both ReLU and ReLU^2 activation functions, see Theorem 3.3. The technique we used here is also helpful for bounding the statistical errors to other deep PDEs solvers.

- We give an upper bound of the error caused by the Robin approximation without additional assumptions, i.e., bound the error between the minimizer of the penalized form u_λ^* and the weak solutions of the Laplace equation u^*, see Theorem 3.4,

$$\|u_\lambda^* - u^*\|_{H^1(\Omega)} \leq \mathcal{O}(\lambda^{-1}).$$

This result improves the one established in [23, 35, 36].

- Based on the above two error bounds we establish a nonasymptotic convergence rate of deep Ritz method for Laplace equation with Dirichlet boundary condition. We prove that if we set

$$\mathcal{D} \leq \lceil \log_2 d \rceil + 3, \quad \mathcal{W} \leq \mathcal{O}\left(4d \left\lceil \left(\frac{n}{\log n} \right)^{\frac{1}{2(d+2)}} - 4 \right\rceil^d \right),$$

and

$$\lambda \sim n^{\frac{1}{3(d+2)}} (\log n)^{-\frac{d+3}{3(d+2)}},$$

it holds that

$$\mathbb{E}_{X,Y} \left[\|\widehat{u}_\phi - u^*\|_{H^1(\Omega)}^2 \right] \leq \mathcal{O}\left(n^{-\frac{2}{3(d+2)}} \log n \right),$$

where n is the number of training samples on both the domain and the boundary. Our theory shed lights on how to choose the topological structure of the employed networks and tune the penalty parameters to achieve the desired convergence rate in terms of number of training samples.

Recently, [35, 36] also study the convergence of DRM with Dirichlet boundary condition via penalty method. However, the results of derive in [35, 36] are quite different from ours. Firstly, the approximation results in [35, 36] is based on the approximation error of ReLU networks in Sobolev norms established in [20]. However, the ReLU network may not be suitable for solving PDEs. In this work, we derive an upper bound on the approximation error of ReLU^2 networks in H^1 norm, which is of independent interest. Secondly, to analyze the error caused by the penalty term, [35, 36] assumed some additional conditions, and we do not need these conditions to obtain the error inducing by the penalty. Lastly, we provide the convergence rate analysis involving the statistical error caused by finite samples used in the SGD training, while in [35, 36] they do not consider the statistical error at all. Moreover, to bound the statistical error we need to control the Rademacher complexity of the non-Lipschitz composition of gradient norm and ReLU^2 network, such technique can be useful for bounding the statistical errors to other deep PDEs solvers.

The rest of this paper is organized as follows. In Section 2 we describe briefly the model problem and recall some standard properties of PDEs and variational problems. We also introduce some notations in deep Ritz methods as preliminaries. We devote Section 3 to the detail analysis on the convergence rate of the deep Ritz method with penalty, where various error estimations are analyzed rigorously one by one and the main results on the convergence rate are presented. Some concluding remarks and discussions are given in Section 4.

2　Preliminaries

Consider the following elliptic equation with zero-boundary condition

$$\begin{cases} -\Delta u + wu = f, & \text{in } \Omega, \\ u = 0, & \text{on } \partial\Omega, \end{cases} \tag{2.1}$$

where Ω is a bounded open subset of $\mathbb{R}^d$, $d > 1$, $f \in L^2(\Omega)$ and $w \in L^\infty(\Omega)$. Moreover, we suppose the coefficient w satisfies $w \geq c_1 > 0$ a.e.. Without loss of generality, we assume $\Omega = [0,1]^d$. Define the bilinear form

$$a: H^1(\Omega) \times H^1(\Omega) \to \mathbb{R}, \quad (u,v) \mapsto \int_\Omega \nabla u \cdot \nabla v + wuv \, dx, \tag{2.2}$$

and the corresponding quadratic energy functional by

$$\mathcal{L}(u) = \frac{1}{2} a(u,u) - \langle f, u \rangle_{L^2(\Omega)}. \tag{2.3}$$

Lemma 2.1. [16] *The unique weak solution $u^* \in H_0^1(\Omega)$ of (2.1) is the unique minimizer of $\mathcal{L}(u)$ over $H_0^1(\Omega)$. Moreover, $u^* \in H^2(\Omega)$.*

Now we introduce the Robin approximation of (2.1) with $\lambda > 0$ as below

$$
\begin{cases}
-\Delta u + wu = f, & \text{in } \Omega, \\
\dfrac{1}{\lambda}\dfrac{\partial u}{\partial n} + u = 0, & \text{on } \partial\Omega.
\end{cases}
\tag{2.4}
$$

Similarly, we define the bilinear form

$$
a_\lambda : H^1(\Omega) \times H^1(\Omega) \to \mathbb{R}, \quad (u,v) \mapsto a(u,v) + \lambda \int_{\partial\Omega} uv \, ds,
$$

and the corresponding quadratic energy functional with boundary penalty

$$
\mathcal{L}_\lambda(u) = \frac{1}{2}a_\lambda(u,u) - \langle f,u \rangle_{L^2(\Omega)} = \mathcal{L}(u) + \frac{\lambda}{2}\|Tu\|^2_{L^2(\partial\Omega)},
\tag{2.5}
$$

where T means the trace operator.

Lemma 2.2. *The unique weak solution $u_\lambda^* \in H^1(\Omega)$ of (2.4) is the unique minimizer of $\mathcal{L}_\lambda(u)$ over $H^1(\Omega)$. Moreover, $u_\lambda^* \in H^2(\Omega)$.*

Proof. See Appendix A.1. □

From the perspective of infinite dimensional optimization, $\mathcal{L}_\lambda$ can be seen as the penalized version of $\mathcal{L}$. The following lemma provides the relationship between the minimizers of them.

Lemma 2.3. *The minimizer u_λ^* of the penalized problem (2.5) converges to u^* in $H^1(\Omega)$ as $\lambda \to \infty$.*

Proof. This result follows from Proposition 2.1 in [33] directly. □

The deep Ritz method can be divided into three steps. First, one use deep neural network to approximate the trial function. A deep neural network $u_\phi : \mathbb{R}^d \to \mathbb{R}^{N_L}$ is defined by

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

where $A_\ell \in \mathbb{R}^{N_\ell \times N_{\ell-1}}$, $b_\ell \in \mathbb{R}^{N_\ell}$ and the activation functions σ_ℓ may be different for different ℓ. The depth $\mathcal{D}$ and the width $\mathcal{W}$ of neural networks u_ϕ are defined as

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

$\sum_{\ell=1}^{L} N_\ell$ 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 network.

Definition 2.1. *The class $\mathcal{N}_{\mathcal{D},\mathcal{W},\mathcal{B}}^\alpha$ is the collection of neural networks u_ϕ which satisfies that*

(i) depth and width are $\mathcal{D}$ and $\mathcal{W}$, respectively;

(ii) the function values $u_\phi(x)$ and the squared norm of $\nabla u_\phi(x)$ are bounded by $\mathcal{B}$;

(iii) activation functions are given by ReLU^α, where α is the (multi-)index.

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

Second, one use Monte Carlo method to discretize the energy functional. We rewrite (2.5) as

$$
\mathcal{L}_\lambda(u) = |\Omega| \underset{X \sim U(\Omega)}{\mathbb{E}} \left[\frac{\|\nabla u(X)\|_2^2}{2} + \frac{w(X)u^2(X)}{2} - u(X)f(X) \right]
$$
$$
+ \frac{\lambda}{2} |\partial\Omega| \underset{Y \sim U(\partial\Omega)}{\mathbb{E}} \left[Tu^2(Y) \right], \tag{2.6}
$$

where $U(\Omega)$, $U(\partial\Omega)$ are the uniform distribution on Ω and $\partial\Omega$. We now introduce the discrete version of (2.5) and replace u by neural network u_ϕ, as follows

$$
\widehat{\mathcal{L}}_\lambda(u_\phi) = \frac{|\Omega|}{N} \sum_{i=1}^N \left[\frac{\|\nabla u_\phi(X_i)\|_2^2}{2} + \frac{w(X_i)u_\phi^2(X_i)}{2} - u_\phi(X_i)f(X_i) \right]
$$
$$
+ \frac{\lambda}{2} \frac{|\partial\Omega|}{M} \sum_{j=1}^M \left[Tu_\phi^2(Y_j) \right]. \tag{2.7}
$$

We denote the minimizer of (2.7) over $\mathcal{N}^2$ as $\widehat{u}_\phi$, that is

$$
\widehat{u}_\phi = \underset{u_\phi \in \mathcal{N}^2}{\arg\min} \widehat{\mathcal{L}}_\lambda(u_\phi), \tag{2.8}
$$

where $\{X_i\}_{i=1}^N \sim U(\Omega)$ i.i.d. and $\{Y_j\}_{j=1}^M \sim U(\partial\Omega)$ i.i.d..

Finally, we choose an algorithm for solving the optimization problem, and denote $u_{\phi_\mathcal{A}}$ as the solution by optimizer $\mathcal{A}$.

3   Error analysis

In this section we prove the convergence rate analysis for DRM with deep ReLU^2 networks. The following Theorem plays an important role by decoupling the total errors into four types of errors.

Theorem 3.1.

$$\|u_{\phi_A} - u^*\|^2_{H^1(\Omega)}$$

$$\leq \frac{4}{c_1 \wedge 1} \left\{ \underbrace{\inf_{\bar{u} \in \mathcal{N}^2} \left[\frac{\|w\|_{L^\infty(\Omega)} \vee 1}{2} \|\bar{u} - u^*_\lambda\|^2_{H^1(\Omega)} + \frac{\lambda}{2} \|T\bar{u} - Tu^*_\lambda\|^2_{L^2(\partial\Omega)} \right]}_{\mathcal{E}_{app}} \right.$$

$$\left. + \underbrace{2 \sup_{u \in \mathcal{N}^2} \left| \mathcal{L}_\lambda(u) - \widehat{\mathcal{L}}_\lambda(u) \right|}_{\mathcal{E}_{sta}} + \underbrace{\left[\widehat{\mathcal{L}}_\lambda(u_{\phi_A}) - \widehat{\mathcal{L}}_\lambda(\hat{u}_\phi) \right]}_{\mathcal{E}_{opt}} \right\} + \underbrace{2 \|u^*_\lambda - u^*\|^2_{H^1(\Omega)}}_{\mathcal{E}_{pen}}.$$

Proof. Given $u_{\phi_A} \in H^1(\Omega)$, we can decompose its distance to the weak solution of (2.1) using triangle inequality

$$\|u_{\phi_A} - u^*\|_{H^1(\Omega)} \leq \|u_{\phi_A} - u^*_\lambda\|_{H^1(\Omega)} + \|u^*_\lambda - u^*\|_{H^1(\Omega)}. \tag{3.1}$$

First, we decouple the first term into three parts. For any $\bar{u} \in \mathcal{N}^2$, we have

$$\mathcal{L}_\lambda(u_{\phi_A}) - \mathcal{L}_\lambda(u^*_\lambda)$$
$$= \mathcal{L}_\lambda(u_{\phi_A}) - \widehat{\mathcal{L}}_\lambda(u_{\phi_A}) + \widehat{\mathcal{L}}_\lambda(u_{\phi_A}) - \widehat{\mathcal{L}}_\lambda(\hat{u}_\phi) + \widehat{\mathcal{L}}_\lambda(\hat{u}_\phi) - \widehat{\mathcal{L}}_\lambda(\bar{u})$$
$$\quad + \widehat{\mathcal{L}}_\lambda(\bar{u}) - \mathcal{L}_\lambda(\bar{u}) + \mathcal{L}_\lambda(\bar{u}) - \mathcal{L}_\lambda(u^*_\lambda)$$
$$\leq [\mathcal{L}_\lambda(\bar{u}) - \mathcal{L}_\lambda(u^*_\lambda)] + 2 \sup_{u \in \mathcal{N}^2} \left| \mathcal{L}_\lambda(u) - \widehat{\mathcal{L}}_\lambda(u) \right| + \left[\widehat{\mathcal{L}}_\lambda(u_{\phi_A}) - \widehat{\mathcal{L}}_\lambda(\hat{u}_\phi) \right].$$

Since $\bar{u}$ can be any element in $\mathcal{N}^2$, we take the infimum of $\bar{u}$

$$\mathcal{L}_\lambda(u_{\phi_A}) - \mathcal{L}_\lambda(u^*_\lambda) \leq \inf_{\bar{u} \in \mathcal{N}^2} [\mathcal{L}_\lambda(\bar{u}) - \mathcal{L}_\lambda(u^*)] + 2 \sup_{u \in \mathcal{N}^2} \left| \mathcal{L}_\lambda(u) - \widehat{\mathcal{L}}_\lambda(u) \right|$$
$$+ \left[\widehat{\mathcal{L}}_\lambda(u_{\phi_A}) - \widehat{\mathcal{L}}_\lambda(\hat{u}_\phi) \right]. \tag{3.2}$$

For any $u \in \mathcal{N}^2$, set $v = u - u^*_\lambda$, then

$$\mathcal{L}_\lambda(u) = \mathcal{L}_\lambda(u^*_\lambda + v)$$
$$= \frac{1}{2} \langle \nabla(u^*_\lambda + v), \nabla(u^*_\lambda + v) \rangle_{L^2(\Omega)} + \frac{1}{2} \langle u^*_\lambda + v, u^*_\lambda + v \rangle_{L^2(\Omega;w)} - \langle u^*_\lambda + v, f \rangle_{L^2(\Omega)}$$
$$\quad + \frac{\lambda}{2} \langle T(u^*_\lambda + v), T(u^*_\lambda + v) \rangle_{L^2(\partial\Omega)}$$
$$= \mathcal{L}_\lambda(u^*_\lambda) + \langle \nabla u^*_\lambda, \nabla v \rangle_{L^2(\Omega)} + \langle u^*_\lambda, v \rangle_{L^2(\Omega;w)} - \langle v, f \rangle_{L^2(\Omega)} + \lambda \langle Tu^*_\lambda, Tv \rangle_{L^2(\partial\Omega)}$$
$$\quad + \frac{1}{2} \langle \nabla v, \nabla v \rangle_{L^2(\Omega)} + \frac{1}{2} \langle v, v \rangle_{L^2(\Omega;w)} + \frac{\lambda}{2} \langle Tv, Tv \rangle_{L^2(\partial\Omega)}$$
$$= \mathcal{L}_\lambda(u^*_\lambda) + \frac{1}{2} \langle \nabla v, \nabla v \rangle_{L^2(\Omega)} + \frac{1}{2} \langle v, v \rangle_{L^2(\Omega;w)} + \frac{\lambda}{2} \langle Tv, Tv \rangle_{L^2(\partial\Omega)},$$

where the last equality comes from the fact that u_λ^* is the minimizer of (2.5). Therefore

$$\mathcal{L}_\lambda(u) - \mathcal{L}_\lambda(u_\lambda^*) = \frac{1}{2}\langle \nabla v, \nabla v \rangle_{L^2(\Omega)} + \frac{1}{2}\langle v, v \rangle_{L^2(\Omega;w)} + \frac{\lambda}{2}\langle Tv, Tv \rangle_{L^2(\partial\Omega)},$$

that is

$$\frac{c_1 \wedge 1}{2}\|u - u_\lambda^*\|_{H^1(\Omega)}^2 \leq \mathcal{L}_\lambda(u) - \mathcal{L}_\lambda(u_\lambda^*) - \frac{\lambda}{2}\|Tu - Tu_\lambda^*\|_{L^2(\partial\Omega)}^2$$

$$\leq \frac{\|w\|_{L^\infty(\Omega)} \vee 1}{2}\|u - u_\lambda^*\|_{H^1(\Omega)}^2. \tag{3.3}$$

Combining (3.2) and (3.3), we obtain

$$\|u_{\phi_\mathcal{A}} - u_\lambda^*\|_{H^1(\Omega)}^2$$

$$\leq \frac{2}{c_1 \wedge 1}\left\{ \mathcal{L}_\lambda(u_{\phi_\mathcal{A}}) - \mathcal{L}_\lambda(u_\lambda^*) - \frac{\lambda}{2}\|Tu_{\phi_\mathcal{A}} - Tu_\lambda^*\|_{L^2(\partial\Omega)}^2 \right\}$$

$$\leq \frac{2}{c_1 \wedge 1}\left\{ \inf_{\bar{u} \in \mathcal{N}^2}[\mathcal{L}_\lambda(\bar{u}) - \mathcal{L}_\lambda(u_\lambda^*)] + 2\sup_{u \in \mathcal{N}^2}\left| \mathcal{L}_\lambda(u) - \widehat{\mathcal{L}}_\lambda(u) \right| \right.$$

$$\left. + \left[\widehat{\mathcal{L}}_\lambda(u_{\phi_\mathcal{A}}) - \widehat{\mathcal{L}}_\lambda(\widehat{u}_\phi) \right] \right\}$$

$$\leq \frac{2}{c_1 \wedge 1}\left\{ \inf_{\bar{u} \in \mathcal{N}^2}\left[\frac{\|w\|_{L^\infty(\Omega)} \vee 1}{2}\|\bar{u} - u_\lambda^*\|_{H^1(\Omega)}^2 + \frac{\lambda}{2}\|T\bar{u} - Tu_\lambda^*\|_{L^2(\partial\Omega)}^2 \right] \right.$$

$$\left. + 2\sup_{u \in \mathcal{N}^2}\left| \mathcal{L}_\lambda(u) - \widehat{\mathcal{L}}_\lambda(u) \right| + \left[\widehat{\mathcal{L}}_\lambda(u_{\phi_\mathcal{A}}) - \widehat{\mathcal{L}}_\lambda(\widehat{u}_\phi) \right] \right\}. \tag{3.4}$$

Substituting (3.4) into (3.1), it is evident to see that the theorem holds. □

The approximation error $\mathcal{E}_{app}$ describes the expressive power of the ReLU^2 networks $\mathcal{N}^2$ in H^1 norm, which corresponds to the approximation error in FEM known as the Céa's lemma [9]. The statistical error $\mathcal{E}_{sta}$ is caused by the Monte Carlo discretization of $L_\lambda(\cdot)$ defined in (2.5) with $\widehat{L}_\lambda(\cdot)$ in (2.7). While, the optimization error $\mathcal{E}_{opt}$ indicates the performance of the solver $\mathcal{A}$ we utilized. In contrast, this error is corresponding to the error of solving linear systems in FEM. In this paper we consider the scenario of perfect training with $\mathcal{E}_{opt} = 0$. The error $\mathcal{E}_{pen}$ caused by the boundary penalty is the distance between the minimizer of the energy with zero boundary condition and the minimizer of the energy with penalty.

3.1 Approximation error

Theorem 3.2. *Assume* $\|u_\lambda^*\|_{H^2(\Omega)} \leq c_2$, *then there exist a* ReLU^2 *network* $\bar{u}_{\bar{\phi}} \in \mathcal{N}^2$ *with depth and width satisfying*

$$\mathcal{D} \leq \lceil \log_2 d \rceil + 3, \quad \mathcal{W} \leq 4d\left\lceil \frac{Cc_2}{\varepsilon} - 4 \right\rceil^d$$

such that

$$\left\| u_\lambda^* - \bar{u}_{\bar{\phi}} \right\|_{H^1(\Omega)}^2 \leq \epsilon^2 \quad and \quad \left\| T u_\lambda^* - T \bar{u}_{\bar{\phi}} \right\|_{L^2(\partial\Omega)}^2 \leq C_d \epsilon^2$$

and

$$\mathcal{E}_{app} \leq \left(\frac{\|w\|_{L^\infty(\Omega)} \vee 1}{2} + \frac{\lambda C_d}{2} \right) \varepsilon^2,$$

where C is a genetic constant and $C_d > 0$ is a constant depending only on Ω.

Proof. Our proof is based on some classical approximation results of B-splines [11, 40]. Let us recall some notation and useful results. We denote by π_l the dyadic partition of $[0,1]$, i.e.,

$$\pi_l : t_0^{(l)} = 0 < t_1^{(l)} < \cdots < t_{2^l-1}^{(l)} < t_{2^l}^{(l)} = 1,$$

where $t_i^{(l)} = i \cdot 2^{-l} (0 \leq i \leq 2^l)$. The cardinal B-spline of order 3 with respect to partition π_l is defined by

$$N_{l,i}^{(3)}(x) = (-1)^k \left[t_i^{(l)}, \cdots, t_{i+3}^{(l)}, (x-t)_+^2 \right] \cdot \left(t_{i+3}^{(l)} - t_i^{(l)} \right), \quad i = -2, \cdots, 2^l - 1,$$

which can be rewritten in the following equivalent form,

$$N_{l,i}^{(3)}(x) = 2^{2l-1} \sum_{j=0}^{3} (-1)^j \binom{3}{j} (x - i 2^{-l} - j 2^{-l})_+^2, \quad i = -2, \cdots, 2^l - 1. \tag{3.5}$$

The multivariate cardinal B-spline of order 3 is defined by the product of univariate cardinal B-splines of order 3, i.e.,

$$N_{l,\boldsymbol{i}}^{(3)}(\boldsymbol{x}) = \prod_{j=1}^{d} N_{l,i_j}^{(3)}(x_j), \quad \boldsymbol{i} = (i_1, \cdots, i_d), \quad -3 < i_j < 2^l.$$

Denote

$$S_l^{(3)}([0,1]^d) = \text{span}\{ N_{l,\boldsymbol{i}}^{(3)}, -3 < i_j < 2^l, j = 1, 2, \cdots, d \}.$$

Then, the element f in $S_l^{(3)}([0,1]^d)$ are piecewise polynomial functions according to partition π_l^d with each piece being degree 2 and in $C^1([0,1]^d)$. Since

$$S_1^{(3)} \subset S_2^{(3)} \subset S_3^{(3)} \subset \cdots,$$

we can further denote

$$S^{(3)}([0,1]^d) = \bigcup_{l=1}^{\infty} S_l^{(3)}([0,1]^d).$$

The following approximation result of cardinal B-splines in Sobolev spaces is a direct consequence of theorem 3.4 in [39], which plays an important role in this proof.

Lemma 3.1. *Assume $u^* \in H^2([0,1]^d)$, there exists $\{e_j\}_{j=1}^{(2^l-4)^d} \subset \mathbb{R}$ with $l > 2$ such that*

$$\left\| u^* - \sum_{j=1}^{(2^l-4)^d} e_j N_{l,\boldsymbol{i}_j}^{(3)} \right\|_{H^1(\Omega)} \leq \frac{C}{2^l} \|u^*\|_{H^1(\Omega)},$$

where C is a constant only depend on d.

Lemma 3.2. *The multivariate B-spline $N_{l,\boldsymbol{i}}^{(3)}(\boldsymbol{x})$ can be implemented exactly by a ReLU^2 network with depth $\lceil \log_2 d \rceil + 2$ and width $4d$.*

Proof. Denote

$$\sigma(x) = \begin{cases} x^2, & x \geq 0, \\ 0, & \text{else,} \end{cases}$$

as the activation function in ReLU^2 network. By definition of $N_{l,i}^{(3)}(x)$ in (3.5), it's clear that $N_{l,i}^{(3)}(x)$ can be implemented by ReLU^2 network without any error with depth 2 and width 4. On the other hand ReLU^2 network can also realize multiplication without any error. In fact, for any $x, y \in \mathbb{R}$,

$$xy = \frac{1}{4}[(x+y)^2 - (x-y)^2] = \frac{1}{4}[\sigma(x+y) + \sigma(-x-y) - \sigma(x-y) - \sigma(y-x)].$$

Hence multivariate B-spline of order 3 can be implemented by ReLU^2 network exactly with depth $\lceil \log_2 d \rceil + 2$ and width $4d$. □

For any $\epsilon > 0$, by Lemma 3.1 and 3.2 with $\frac{1}{2^l} \leq \left\lceil \frac{C\|u^*\|_{H^2}}{\epsilon} \right\rceil$, there exists $\bar{u}_{\bar{\phi}} \in \mathcal{N}^2$, such that

$$\left\| u_\lambda^* - \bar{u}_{\bar{\phi}} \right\|_{H^1(\Omega)} \leq \epsilon. \tag{3.6}$$

By the trace theorem, we have

$$\left\| T u_\lambda^* - T \bar{u}_{\bar{\phi}} \right\|_{L^2(\partial\Omega)} \leq C_d^{1/2} \left\| u_\lambda^* - \bar{u}_{\bar{\phi}} \right\|_{H^1(\Omega)} \leq C_d^{1/2} \epsilon, \tag{3.7}$$

where $C_d > 0$ is a constant depending only on Ω. The depth $\mathcal{D}$ and width $\mathcal{W}$ of $\bar{u}_{\bar{\phi}}$ are satisfying $\mathcal{D} \leq \lceil \log_2 d \rceil + 3$ and $\mathcal{W} \leq 4dn = 4d \left\lceil \frac{C\|u^*\|_{H^2}}{\epsilon} - 4 \right\rceil^d$, respectively. Combining (3.6) and (3.7), we arrive at the result. □

3.2 Statistical error

In this section, we bound the statistical error

$$\mathcal{E}_{sta} = 2 \sup_{u \in \mathcal{N}^2} \left| \mathcal{L}_\lambda(u) - \widehat{\mathcal{L}}_\lambda(u) \right|.$$

For simplicity of presentation, we use c_3 to denote the upper bound of f, w and suppose $c_3 \geq \mathcal{B}$, that is

$$\|f\|_{L^\infty(\Omega)} \vee \|w\|_{L^\infty(\Omega)} \vee \mathcal{B} \leq c_3 < \infty.$$

First, we need to decompose the statistical error into four parts, and estimate each one.

Lemma 3.3.

$$\sup_{u \in \mathcal{N}^2} \left| \mathcal{L}_\lambda(u) - \widehat{\mathcal{L}}_\lambda(u) \right| \leq \sum_{j=1}^{3} \sup_{u \in \mathcal{N}^2} \left| \mathcal{L}_{\lambda,j}(u) - \widehat{\mathcal{L}}_{\lambda,j}(u) \right| + \frac{\lambda}{2} \sup_{u \in \mathcal{N}^2} \left| \mathcal{L}_{\lambda,4}(u) - \widehat{\mathcal{L}}_{\lambda,4}(u) \right|,$$

where

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

$$\mathcal{L}_{\lambda,2}(u) = |\Omega| \mathop{\mathbb{E}}_{X \sim U(\Omega)} \left[\frac{w(X)u^2(X)}{2} \right], \quad \widehat{\mathcal{L}}_{\lambda,2}(u) = \frac{|\Omega|}{N} \sum_{i=1}^{N} \left[\frac{w(X_i)u^2(X_i)}{2} \right],$$

$$\mathcal{L}_{\lambda,3}(u) = |\Omega| \mathop{\mathbb{E}}_{X \sim U(\Omega)} [u(X)f(X)], \quad \widehat{\mathcal{L}}_{\lambda,3}(u) = \frac{|\Omega|}{N} \sum_{i=1}^{N} [u(X_i)f(X_i)],$$

$$\mathcal{L}_{\lambda,4}(u) = |\partial\Omega| \mathop{\mathbb{E}}_{Y \sim U(\partial\Omega)} [Tu^2(Y)], \quad \widehat{\mathcal{L}}_{\lambda,4}(u) = \frac{|\partial\Omega|}{M} \sum_{j=1}^{M} [Tu^2(Y_j)].$$

Proof. It is easy to verified by triangle inequality. □

We use μ to denote $U(\Omega)(U(\partial\Omega))$. Given $n = N(M)$ i.i.d samples $\mathbf{Z}_n = \{Z_i\}_{i=1}^n$ from μ, with $Z_i = X_i(Y_i) \sim \mu$, we need the following Rademacher complexity to measure the capacity of the given function class $\mathcal{N}$ restricted on n random samples $\mathbf{Z}_n$.

Definition 3.1. *The Rademacher complexity of a set $A \subseteq \mathrm{R}^n$ is defined as*

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

where $\Sigma_n = \{\sigma_i\}_{i=1}^n$ 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

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

For the sake of simplicity, we deal with last three terms first.

Lemma 3.4. *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 classes of functions on* Ω *and* $\psi \circ \mathcal{N} = \{\psi \circ u : x \mapsto \psi(x,u(x)), u \in \mathcal{N}\}$. *For any* $n \in \mathbb{N}_{\geq 1}$,

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

Proof. Corollary 3.17 in [27]. $\qquad\square$

Lemma 3.5.

$$\mathbb{E}_{\mathbf{Z}_N}\left[\sup_{u \in \mathcal{N}^2} \left| \mathcal{L}_{\lambda,2}(u) - \widehat{\mathcal{L}}_{\lambda,2}(u) \right|\right] \leq c_3^2 \mathfrak{R}_N(\mathcal{N}^2),$$

$$\mathbb{E}_{\mathbf{Z}_N}\left[\sup_{u \in \mathcal{N}^2} \left| \mathcal{L}_{\lambda,3}(u) - \widehat{\mathcal{L}}_{\lambda,3}(u) \right|\right] \leq c_3 \mathfrak{R}_N(\mathcal{N}^2),$$

$$\mathbb{E}_{\mathbf{Z}_M}\left[\sup_{u \in \mathcal{N}^2} \left| \mathcal{L}_{\lambda,4}(u) - \widehat{\mathcal{L}}_{\lambda,4}(u) \right|\right] \leq 2c_3 \mathfrak{R}_M(\mathcal{N}^2).$$

Proof. Suppose $|y| < c_3$. Define

$$\psi_2(x,y) = \frac{w(x)y^2}{2}, \quad \psi_3(x,y) = f(x)y, \quad \psi_4(x,y) = y^2.$$

According to the symmetrization method, we have

$$\mathbb{E}_{\mathbf{Z}_N}\left[\sup_{u \in \mathcal{N}^2} \left| \mathcal{L}_{\lambda,2}(u) - \widehat{\mathcal{L}}_{\lambda,2}(u) \right|\right] \leq \mathfrak{R}_N(\psi_2 \circ \mathcal{N}^2),$$

$$\mathbb{E}_{\mathbf{Z}_N}\left[\sup_{u \in \mathcal{N}^2} \left| \mathcal{L}_{\lambda,3}(u) - \widehat{\mathcal{L}}_{\lambda,3}(u) \right|\right] \leq \mathfrak{R}_N(\psi_3 \circ \mathcal{N}^2), \tag{3.8}$$

$$\mathbb{E}_{\mathbf{Z}_M}\left[\sup_{u \in \mathcal{N}^2} \left| \mathcal{L}_{\lambda,4}(u) - \widehat{\mathcal{L}}_{\lambda,4}(u) \right|\right] \leq \mathfrak{R}_M(\psi_4 \circ \mathcal{N}^2).$$

The result will follow from Lemma 3.4 and (3.8) directly, if we can show that $\psi_i(x,y)$, $i=2,3,4$ are c_3^2, c_3, $2c_3$-Lipschitz continuous on y for all x, respectively. For arbitrary y_1, y_2 with $|y_i| \leq c_3$, $i=1,2$,

$$|\psi_2(x,y_1) - \psi_2(x,y_2)| = \left| \frac{w(x)y_1^2}{2} - \frac{w(x)y_2^2}{2} \right| = \frac{|w(x)(y_1+y_2)|}{2}|y_1 - y_2| \leq c_3^2|y_1 - y_2|,$$

$$|\psi_3(x,y_1) - \psi_3(x,y_2)| = |f(x)y_1 - f(x)y_2| = |f(x)||y_1 - y_2| \leq c_3|y_1 - y_2|,$$

$$|\psi_4(x,y_1) - \psi_4(x,y_2)| = |y_1^2 - y_2^2| = |y_1 + y_2||y_1 - y_2| \leq 2c_3|y_1 - y_2|.$$

$\qquad\square$

We now turn to the most difficult term in Lemma 3.3. Since gradient is not a Lipschitz operator, Lemma 3.4 does not work and we can not bound the Rademacher complexity in the same way.

Lemma 3.6.

$$\mathbb{E}_{\mathbf{Z}_N}\left[\sup_{u\in\mathcal{N}^2}\left|\mathcal{L}_{\lambda,1}(u)-\widehat{\mathcal{L}}_{\lambda,1}(u)\right|\right]\leq\mathfrak{R}_N(\mathcal{N}^{1,2}).$$

Proof. Based on the symmetrization method, we have

$$\mathbb{E}_{\mathbf{Z}_N}\left[\sup_{u\in\mathcal{N}^2}\left|\mathcal{L}_{\lambda,1}(u)-\widehat{\mathcal{L}}_{\lambda,1}(u)\right|\right]\leq\mathbb{E}_{\mathbf{Z}_N,\Sigma_N}\left[\sup_{u\in\mathcal{N}^2}\frac{1}{N}\left|\sum_i\sigma_i\|\nabla u(Z_i)\|^2\right|\right]. \tag{3.9}$$

The proof of (3.9) is a direct consequence of the following claim.

Claim: 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 $D_iu(i=1,2,\cdots,d)$ can be implemented by a ReLU-ReLU^2 network respectively, we can easily obtain the network we desire, since, $\|\nabla u\|_2^2=\sum_{i=1}^d|D_iu|^2$ and the square function can be implemented by $x^2=\sigma_2(x)+\sigma_2(-x)$.

Now we show that for any $i=1,2,\cdots,d$, D_iu can be implemented by a ReLU-ReLU^2 network. We deal with the first two layers in details since there are a little bit difference for the first two layer and apply induction for layers $k\geq3$. For the first layer, since $\sigma_2'(x)=2\sigma_1(x)$, we have for any $q=1,2\cdots,n_1$,

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

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

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

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

$$x\cdot y=\frac{1}{4}\left[(x+y)^2-(x-y)^2\right]$$
$$=\frac{1}{4}\left[\sigma_2(x+y)+\sigma_2(-x-y)-\sigma_2(x-y)-\sigma_2(-x+y)\right],$$

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

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

We assume that $D_i u_q^{(k)}$ $(q = 1, 2, \cdots, n_k)$ can be implemented by a ReLU-ReLU^2 network and $\mathcal{D}\left(D_i u_q^{(k)}\right) = k + 2$, $\mathcal{W}\left(D_i u_q^{(3)}\right) \leq (k+2)\mathcal{W}$. For the $(k+1)-$th layer,

$$D_i u_q^{(k+1)} = D_i \sigma_2\left(\sum_{j=1}^{n_k} a_{qj}^{(k+1)} u_j^{(k)} + b_q^{(k+1)}\right)$$

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

Since

$$\mathcal{D}\left(\sigma_1\left(\sum_{j=1}^{n_k} a_{qj}^{(k+1)} u_j^{(k)} + b_q^{(k+1)}\right)\right) = k + 2,$$

$$\mathcal{W}\left(\sigma_1\left(\sum_{j=1}^{n_k} a_{qj}^{(k+1)} u_j^{(k)} + b_q^{(k+1)}\right)\right) \leq \mathcal{W},$$

and

$$\mathcal{D}\left(\sum_{j=1}^{n_k} a_{qj}^{(k+1)} D_i u_j^{(k)}\right) = k+2,$$

$$\mathcal{W}\left(\sum_{j=1}^{n_k} a_{qj}^{(k+1)} D_i u_j^{(k)}\right) \leq \max\{(k+2)\mathcal{W}, 4\mathcal{W}\} = (k+2)\mathcal{W},$$

we conclude that $D_i u_q^{(k+1)}$ can be implemented by a ReLU-ReLU^2 network and $\mathcal{D}(D_i u_q^{(k+1)}) = k+3$, $\mathcal{W}(D_i u_q^{(k+1)}) \leq \max\{(k+3)\mathcal{W}, 4\} = (k+3)\mathcal{W}$.

Hence we derive that $D_i u = D_i u_1^{\mathcal{D}}$ can be implemented by a ReLU-ReLU^2 network and $\mathcal{D}(D_i u) = \mathcal{D}+2$, $\mathcal{W}(D_i u) \leq (\mathcal{D}+2)\mathcal{W}$. Finally we obtain that $\mathcal{D}(\|\nabla u\|^2) = \mathcal{D}+3$, $\mathcal{W}(\|\nabla u\|^2) \leq d(\mathcal{D}+2)\mathcal{W}$. $\qquad\square$

We are now in a position to bound the Rademacher complexity of $\mathcal{N}^2$ and $\mathcal{N}^{1,2}$. To obtain the estimation, we need to introduce covering number, VC-dimension, pseudo-dimension and recall several properties of them.

Definition 3.2. *Suppose that $W \subset \mathbb{R}^n$. For any $\epsilon > 0$, let $V \subset \mathbb{R}^n$ be a ϵ-cover of W with respect to the distance d_∞, that is, for any $w \in W$, there exists a $v \in V$ such that $d_\infty(u, v) < \epsilon$, where d_∞ is defined by*

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

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

Definition 3.3. *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*

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

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

$$\mathcal{C}_\infty(\epsilon, \mathcal{N}, n) = \max_{\mathbf{Z}_n \in \Omega^n} \mathcal{C}(\epsilon, \mathcal{N}|_{\mathbf{Z}_n}, d_\infty).$$

Next we give a upper bound of $\mathfrak{R}(\mathcal{N})$ in terms of the covering number of $\mathcal{N}$ by using the Dudley's entropy formula [13].

Lemma 3.7 (Massart's finite class lemma [7])**.** *For any finite set $V \in \mathbb{R}^n$ with diameter $D = \sum_{v \in V} \|v\|_2$, then*

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

We give an upper bound of $\mathfrak{R}(\mathcal{N})$ in terms of the covering number by using the Dudley's entropy formula [13].

Lemma 3.8 (Dudley's entropy formula [13]). *Assume $0 \in \mathcal{N}$ 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 any $n \in \mathbb{N}_{\ge 1}$,*

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

Proof. By definition

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

Thus, it suffice to show

$$\mathbb{E}_\Sigma \left[\sup_{u \in \mathcal{N}} \frac{1}{n} \left| \sum_i \sigma_i u(Z_i) \right| \right] \le \inf_{0 < \delta < \mathcal{B}} \left(4\delta + \frac{12}{\sqrt{n}} \int_\delta^{\mathcal{B}} \sqrt{\log \mathcal{C}(\varepsilon, \mathcal{N}^2, n)} d\varepsilon \right)$$

by conditioning on $\mathbf{Z}_n$. Given an positive integer K, let $\varepsilon_k = 2^{-k+1}\mathcal{B}$, $k = 1, \cdots, K$. Let C_k be a cover of $\mathcal{N}|_{\mathbf{Z}_n} \subseteq \mathbb{R}^n$ whose covering number is denoted as $\mathcal{C}(\varepsilon_k, \mathcal{N}|_{\mathbf{Z}_n}, d_\infty)$. Then, by definition, $\forall u \in \mathcal{N}$, there $\exists c^k \in C_k$ such that

$$d_\infty(u|_{\mathbf{Z}_n}, c^k) = \max\{|u(Z_i) - c_i^k|, i = 1, \cdots, n\} \le \varepsilon_k, \quad k = 1, \cdots, K.$$

Moreover, we denote the best approximate element of u in C_k with respect to d_∞ as $c^k(u)$. Then,

$$\mathbb{E}_\Sigma \left[\sup_{u \in \mathcal{N}} \frac{1}{n} \left| \sum_{i=1}^n \sigma_i u(Z_i) \right| \right]$$

$$= \mathbb{E}_\Sigma \left[\sup_{u \in \mathcal{N}} \frac{1}{n} \left| \sum_{i=1}^n \sigma_i(u(Z_i) - c_i^K(u)) + \sum_{j=1}^{K-1} \sum_{i=1}^n \sigma_i(c_i^j(u) - c_i^{j+1}(u)) + \sum_{i=1}^n \sigma_i c_i^1(u) \right| \right]$$

$$\le \mathbb{E}_\Sigma \left[\sup_{u \in \mathcal{N}} \frac{1}{n} \left| \sum_{i=1}^n \sigma_i(u(Z_i) - c_i^K(u)) \right| \right] + \sum_{j=1}^{K-1} \mathbb{E}_\Sigma \left[\sup_{u \in \mathcal{N}} \frac{1}{n} \left| \sum_{i=1}^n \sigma_i(c_i^j(u) - c_i^{j+1}(u)) \right| \right]$$

$$+ \mathbb{E}_\Sigma \left[\sup_{u \in \mathcal{N}} \frac{1}{n} \left| \sum_{i=1}^n \sigma_i c_i^1(u) \right| \right].$$

Since $0 \in \mathcal{N}$, and the diameter of $\mathcal{N}$ is smaller than $\mathcal{B}$, we can choose $C_1 = \{0\}$ such that the third term in the above display vanishes. By Hölder's inequality, we deduce that the first term can be bounded by ε_K as follows.

$$\mathbb{E}_\Sigma \left[\sup_{u \in \mathcal{N}} \frac{1}{n} \left| \sum_{i=1}^n \sigma_i(u(Z_i) - c_i^K(u)) \right| \right]$$

$$\le \mathbb{E}_\Sigma \left[\sup_{u \in \mathcal{N}} \frac{1}{n} \left(\sum_{i=1}^n |\sigma_i| \right) \left(\sum_{i=1}^n \max_{i=1,\dots,n} \left\{ \left| u(Z_i) - c_i^K(u) \right| \right\} \right) \right]$$

$$\le \varepsilon_K.$$

Let $V_j = \{c^j(u) - c^{j+1}(u) : u \in \mathcal{N}\}$. Then by definition, the number of elements in V_j and C_j satisfying

$$|V_j| \leq |C_j||C_{j+1}| \leq |C_{j+1}|^2.$$

And the diameter of V_j denoted as D_j can be bounded as

$$
\begin{aligned}
D_j = \sup_{v \in V_j} \|v\|_2 &\leq \sqrt{n} \sup_{u \in \mathcal{N}} \|c^j(u) - c^{j+1}(u)\|_\infty \\
&\leq \sqrt{n} \sup_{u \in \mathcal{N}} \|c^j(u) - u\|_\infty + \|u - c^{j+1}(u)\|_\infty \\
&\leq \sqrt{n}(\varepsilon_j + \varepsilon_{j+1}) \\
&\leq 3\sqrt{n}\varepsilon_{j+1}.
\end{aligned}
$$

Then,

$$
\begin{aligned}
\mathbb{E}_\Sigma \left[\sup_{u \in \mathcal{N}} \frac{1}{n} \left| \sum_{j=1}^{K-1} \sum_{i=1}^n \sigma_i (c_i^j(u) - c_i^{j+1}(u)) \right| \right] &\leq \sum_{j=1}^{K-1} \mathbb{E}_\Sigma \left[\sup_{v \in V_j} \frac{1}{n} \left| \sum_{i=1}^n \sigma_i v_j \right| \right] \\
&\leq \sum_{j=1}^{K-1} \frac{D_j}{n} \sqrt{2\log(2|V_j|)} \\
&\leq \sum_{j=1}^{K-1} \frac{6\varepsilon_{j+1}}{\sqrt{n}} \sqrt{\log(2|C_{j+1}|)},
\end{aligned}
$$

where we use triangle inequality in the first inequality, and use Lemma 3.7 in the second inequality. Putting all the above estimates together, we get

$$
\begin{aligned}
\mathbb{E}_\Sigma \left[\sup_{u \in \mathcal{N}} \frac{1}{n} \left| \sum_i \sigma_i u(Z_i) \right| \right] &\leq \varepsilon_K + \sum_{j=1}^{K-1} \frac{6\varepsilon_{j+1}}{\sqrt{n}} \sqrt{\log(2|C_{j+1}|)} \\
&\leq \varepsilon_K + \sum_{j=1}^K \frac{12(\varepsilon_j - \varepsilon_{j+1})}{\sqrt{n}} \sqrt{\log(2\mathcal{C}(\varepsilon_j, \mathcal{N}, n))} \\
&\leq \varepsilon_K + \frac{12}{\sqrt{n}} \int_{\varepsilon_{K+1}}^{\mathcal{B}} \sqrt{\log(2\mathcal{C}(\varepsilon, \mathcal{N}, n))} \, d\varepsilon \\
&\leq \inf_{0 < \delta < \mathcal{B}} \left(4\delta + \frac{12}{\sqrt{n}} \int_\delta^{\mathcal{B}} \sqrt{\log(2\mathcal{C}(\varepsilon, \mathcal{N}, n))} \, d\varepsilon \right),
\end{aligned}
$$

where last inequality holds since for $0 < \delta < \mathcal{B}$, we can choose K to be the largest integer such that $\varepsilon_{K+1} > \delta$, at this time $\varepsilon_K \leq 4\varepsilon_{K+2} \leq 4\delta$. $\qquad \square$

Definition 3.4. *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*

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

Definition 3.5. *The VC-dimension of $\mathcal{N}$, denoted as $\mathrm{VCdim}(\mathcal{N})$, is defined to be the maximum cardinality among all sets shattered by $\mathcal{N}$.*

VC-dimension reflects the capability of a class of functions to perform binary classification of points. The larger VC-dimension is, the stronger the capability to perform binary classification is. For more discussion of VC-dimension, readers are referred to [3].

For real-valued functions, we can generalize the concept of VC-dimension into pseudo-dimension [3].

Definition 3.6. *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 $y_1, y_2, \cdots, y_n$ such that for any $b \in \{0,1\}^n$, there exists a $u \in \mathcal{N}$ satisfying*

$$\mathrm{sign}(u(x_i) - y_i) = b_i, \quad i = 1, 2, \cdots, n,$$

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

Definition 3.7. *The pseudo-dimension of $\mathcal{N}$, denoted as $\mathrm{Pdim}(\mathcal{N})$, is defined to be the maximum cardinality among all sets pseudo-shattered by $\mathcal{N}$.*

The following proposition showing a relationship between uniform covering number and pseudo-dimension.

Lemma 3.9. *Let $\mathcal{N}$ be a set of real functions from a domain X to the bounded interval $[0, \mathcal{B}]$. Let $\varepsilon > 0$. Then*

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

which is less than $\left(\frac{en\mathcal{B}}{\varepsilon \cdot \mathrm{Pdim}(\mathcal{N})} \right)^{\mathrm{Pdim}(\mathcal{N})}$ for $n \geq \mathrm{Pdim}(\mathcal{N})$.

Proof. See Theorem 12.2 in [3]. □

We now present the bound of pseudo-dimension for the $\mathcal{N}^2$ and $\mathcal{N}^{1,2}$.

Lemma 3.10. *Let $p_1, \cdots, p_m$ be polynomials with n variables of degree at most d. If $n \leq m$, then*

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

Proof. See Theorem 8.3 in [3]. □

Lemma 3.11. *Let $\mathcal{N}$ be a set of functions that*

(i) *can be implemented by a neural network with depth no more than $\mathcal{D}$ and width no more than $\mathcal{W}$, and*

(ii) *the activation function in each unit be the ReLU or the ReLU^2.*

Then
$$\mathrm{Pdim}(\mathcal{N}) = \mathcal{O}(\mathcal{D}^2 \mathcal{W}^2 (\mathcal{D} + \log \mathcal{W})).$$

Proof. The argument is follows from the proof of Theorem 6 in [5]. The result stated here is somewhat stronger than Theorem 6 in [5] since $\mathrm{VCdim}(\mathrm{sign}(\mathcal{N})) \leq \mathrm{Pdim}(\mathcal{N})$.

We consider a new set of functions:
$$\widetilde{\mathcal{N}} = \{\tilde{u}(x,y) = \mathrm{sign}(u(x) - y) \colon u \in \mathcal{H}\}.$$

It is clear that $\mathrm{Pdim}(\mathcal{N}) \leq \mathrm{VCdim}(\widetilde{\mathcal{N}})$. 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 network implementing functions in $\mathcal{N}$, in our case we want to derive the uniform bound for
$$K_{\{x_i\},\{y_i\}}(m) := |\{(\mathrm{sign}(f(x_1,a) - y_1), \cdots, \mathrm{sign}(u(x_m,a) - y_m)) \colon a \in \mathbb{R}^{\mathcal{M}}\}|$$

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 3.10, 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 exactly the same as the partition in [5]. Denote the partition as $\{P_1, P_2, \cdots, P_N\}$ with some integer N satisfying

$$N \leq \prod_{i=1}^{\mathcal{D}-1} 2 \left(\frac{2emk_i(1 + (i-1)2^{i-1})}{\mathcal{M}_i} \right)^{\mathcal{M}_i}, \tag{3.10}$$

where k_i and $\mathcal{M}_i$ denotes the number of units at the ith layer and the total number of parameters at the inputs to units in all the layers up to layer i of the neural network implementing functions in $\mathcal{N}$, respectively. See [5] for the construction of the partition. Obviously we have

$$K_{\{x_i\},\{y_i\}}(m) \leq \sum_{i=1}^{N} |\{(\mathrm{sign}(u(x_1,a) - y_1), \cdots, \mathrm{sign}(u(x_m,a) - y_m)) \colon a \in P_i\}|. \tag{3.11}$$

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 [5]. Hence by Lemma 3.10, we have

$$|\{(\mathrm{sign}(u(x_1,a) - y_1), \cdots, \mathrm{sign}(u(x_m,a) - y_m)) \colon a \in P_i\}|$$
$$\leq 2 \left(\frac{2em(1 + (\mathcal{D}-1)2^{\mathcal{D}-1})}{\mathcal{M}_{\mathcal{D}}} \right)^{\mathcal{M}_{\mathcal{D}}}. \tag{3.12}$$

Combining $(3.10),(3.11),(3.12)$ 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 $\mathcal{G}_{\widetilde{\mathcal{N}}}(m)$. Some algebras as that of the proof of Theorem 6 in [5], we obtain

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

where $\mathcal{U}$ refers to the number of units of the neural network implementing functions in $\mathcal{N}$. $\qquad\square$

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

Theorem 3.3. *Let $\mathcal{D}$ and $\mathcal{W}$ be the depth and width of the network respectively. Set the size of training samples $N = M = n$, then*

$$
\mathbb{E}_{X,Y}\mathcal{E}_{sta} \leq C_{c_3} d(\mathcal{D}+3)(\mathcal{D}+2)\mathcal{W}\sqrt{\mathcal{D}+3+\log(d(\mathcal{D}+2)\mathcal{W})}\left(\frac{\log n}{n}\right)^{1/2}
$$
$$
+ C_{c_3} d\mathcal{D}\mathcal{W}\sqrt{\mathcal{D}+\log\mathcal{W}}\left(\frac{\log n}{n}\right)^{1/2}\lambda.
$$

Proof. In order to apply Lemma 3.8, we need to handle the term

$$
\frac{1}{\sqrt{n}}\int_{\delta}^{\mathcal{B}}\sqrt{\log(2\mathcal{C}(\epsilon,\mathcal{N},n))}d\epsilon
$$
$$
\leq \frac{\mathcal{B}}{\sqrt{n}} + \frac{1}{\sqrt{n}}\int_{\delta}^{B}\sqrt{\log\left(\frac{en\mathcal{B}}{\epsilon\cdot\mathrm{Pdim}(\mathcal{N})}\right)^{\mathrm{Pdim}(\mathcal{N})}}d\epsilon
$$
$$
\leq \frac{\mathcal{B}}{\sqrt{n}} + \left(\frac{\mathrm{Pdim}(\mathcal{N})}{n}\right)^{1/2}\int_{\delta}^{B}\sqrt{\log\left(\frac{en\mathcal{B}}{\epsilon\cdot\mathrm{Pdim}(\mathcal{N})}\right)}d\epsilon,
$$

where in the first inequality we use Lemma 3.9. Now we calculate the integral. Set

$$
t = \sqrt{\log\left(\frac{en\mathcal{B}}{\epsilon\cdot\mathrm{Pdim}(\mathcal{N})}\right)},
$$

then $\epsilon = \frac{en\mathcal{B}}{\text{Pdim}(\mathcal{N})} \cdot e^{-t^2}$. Denote $t_1 = \sqrt{\log\left(\frac{en\mathcal{B}}{\mathcal{B}\cdot\text{Pdim}(\mathcal{N})}\right)}$, $t_2 = \sqrt{\log\left(\frac{en\mathcal{B}}{\delta\cdot\text{Pdim}(\mathcal{N})}\right)}$. And

$$
\begin{aligned}
\int_\delta^\mathcal{B} \sqrt{\log\left(\frac{en\mathcal{B}}{\epsilon\cdot\text{Pdim}(\mathcal{N})}\right)} d\epsilon &= \frac{2en\mathcal{B}}{\text{Pdim}(\mathcal{N})} \int_{t_1}^{t_2} t^2 e^{-t^2} dt \\
&= \frac{2en\mathcal{B}}{\text{Pdim}(\mathcal{N})} \int_{t_1}^{t_2} t\left(\frac{-e^{-t^2}}{2}\right)' dt \\
&= \frac{en\mathcal{B}}{\text{Pdim}(\mathcal{N})} \left[t_1 e^{-t_1^2} - t_2 e^{-t_2^2} + \int_{t_1}^{t_2} e^{-t^2} dt\right] \\
&\leq \frac{en\mathcal{B}}{\text{Pdim}(\mathcal{N})} \left[t_1 e^{-t_1^2} - t_2 e^{-t_2^2} + (t_2 - t_1)e^{-t_1^2}\right] \\
&\leq \frac{en\mathcal{B}}{\text{Pdim}(\mathcal{N})} \cdot t_2 e^{-t_1^2} = \mathcal{B}\sqrt{\log\left(\frac{en\mathcal{B}}{\delta\cdot\text{Pdim}(\mathcal{N})}\right)}.
\end{aligned}
$$

Choosing $\delta = \mathcal{B}\left(\frac{\text{Pdim}(\mathcal{N})}{n}\right)^{1/2} \leq \mathcal{B}$, by Lemma 3.8 and the above display, we get for both $\mathcal{N} = \mathcal{N}^2$ and $\mathcal{N} = \mathcal{N}^{1,2}$ there holds

$$
\begin{aligned}
\mathfrak{R}_n(\mathcal{N}) &\leq 4\delta + \frac{12}{\sqrt{n}} \int_\delta^\mathcal{B} \sqrt{\log(2\mathcal{C}(\epsilon,\mathcal{N},n))} d\epsilon \\
&\leq 4\delta + \frac{12\mathcal{B}}{\sqrt{n}} + 12\mathcal{B}\left(\frac{\text{Pdim}(\mathcal{N})}{n}\right)^{1/2} \sqrt{\log\left(\frac{en\mathcal{B}}{\delta\cdot\text{Pdim}(\mathcal{N})}\right)} \\
&\leq 28\sqrt{\frac{3}{2}}\mathcal{B}\left(\frac{\text{Pdim}(\mathcal{N})}{n}\right)^{1/2} \sqrt{\log\left(\frac{en}{\text{Pdim}(\mathcal{N})}\right)}.
\end{aligned} \tag{3.13}
$$

Then by Lemmas 3.3, 3.8, 3.5, 3.6 and Eq. (3.13), we have

$$
\begin{aligned}
\mathbb{E}_{X,Y}\mathcal{E}_{sta} &= 2\sup_{u\in\mathcal{N}^2} |\mathcal{L}(u) - \widehat{\mathcal{L}}(u)| \\
&\leq 2\mathfrak{R}_n(\mathcal{N}^{1,2}) + 2(2c_3^2 + 2c_3)\mathfrak{R}_n(\mathcal{N}^2) + 2c_3^2\mathfrak{R}_n(\mathcal{N}^2)\lambda \\
&\leq 56\sqrt{\frac{3}{2}}\mathcal{B}\left(\frac{\text{Pdim}(\mathcal{N}^{1,2})}{n}\right)^{1/2} \sqrt{\log\left(\frac{en}{\text{Pdim}(\mathcal{N}^{1,2})}\right)} \\
&\quad + 56\sqrt{\frac{3}{2}}(2c_3^2 + 2c_3)\mathcal{B}\left(\frac{\text{Pdim}(\mathcal{N}^2)}{n}\right)^{1/2} \sqrt{\log\left(\frac{en}{\text{Pdim}(\mathcal{N}^2)}\right)} \\
&\quad + 56\sqrt{\frac{3}{2}}c_3^2\mathcal{B}\left(\frac{\text{Pdim}(\mathcal{N}^2)}{n}\right)^{1/2} \sqrt{\log\left(\frac{en}{\text{Pdim}(\mathcal{N}^2)}\right)}\lambda.
\end{aligned}
$$

Plugging the upper bound of Pdim derived in Lemma 3.9 into the above display and

using the relationship of depth and width between $\mathcal{N}^2$ and $\mathcal{N}^{1,2}$, we get

$$
\mathbb{E}_{X,Y}\mathcal{E}_{sta} \leq C_{c_3}d(\mathcal{D}+3)(\mathcal{D}+2)\mathcal{W}\sqrt{\mathcal{D}+3+\log(d(\mathcal{D}+2)\mathcal{W})}\left(\frac{\log n}{n}\right)^{1/2}
$$

$$
+C_{c_3}d\mathcal{D}\mathcal{W}\sqrt{\mathcal{D}+\log\mathcal{W}}\left(\frac{\log n}{n}\right)^{1/2}\lambda. \tag{3.14}
$$

This completes the proof. □

3.3 Error from the boundary penalty method

Although the Lemma 2.3 shows the convergence property of Robin problem (2.4) as $\lambda \to \infty$

$$
u_\lambda^* \to u^*,
$$

it says nothing about the convergence rate. In this section, we consider the error from the boundary penalty method. Roughly speaking, we bound the distance between the minimizer u^* and u_λ^* with respect to the penalty parameter λ.

Theorem 3.4. *Suppose u_λ^* is the minimizer of* (2.5) *and u^* is the minimizer of* (2.3). *Then*

$$
\|u_\lambda^* - u^*\|_{H^1(\Omega)} \leq C_{c_3,d}\lambda^{-1}.
$$

Proof. Following the idea which is proposed in [33] (proof of Proposition 2.3), we proceed to prove this theorem. For $v \in H^1(\Omega)$, we introduce

$$
R_\lambda(v) = \frac{1}{2}a(u^*-v,u^*-v) + \frac{\lambda}{2}\int_{\partial\Omega}\left(-\frac{1}{\lambda}\frac{\partial u^*}{\partial n}-v\right)^2 ds. \tag{3.15}
$$

Given $\varphi \in H^1(\Omega)$ such that $T\varphi = -\frac{\partial u^*}{\partial n}$, we set $w = \frac{1}{\lambda}\varphi + u^*$. Due to $u^* \in H_0^1(\Omega)$, it follows that

$$
R_\lambda(w) = \frac{1}{2\lambda^2}a(\varphi,\varphi) + \frac{\lambda}{2}\int_{\partial\Omega}(u^*)^2 ds = \frac{1}{2\lambda^2}a(\varphi,\varphi) \leq C\lambda^{-2}, \tag{3.16}
$$

where C is dependent only on $\mathcal{B}$, w and Ω. Apparently, (3.15) can be written

$$
R_\lambda(v) = \frac{1}{2}a(u^*,u^*) - a(u^*,v) + \frac{1}{2}a(v,v) + \frac{1}{2\lambda}\int_{\partial\Omega}\left(\frac{\partial u^*}{\partial n}\right)^2 ds + \int_{\partial\Omega}\frac{\partial u^*}{\partial n}v ds
$$

$$
+ \frac{\lambda}{2}\int_{\partial\Omega}v^2 ds
$$

$$
= \frac{1}{2}a(u^*,u^*) + \frac{1}{2}a(v,v) + \frac{1}{2\lambda}\int_{\partial\Omega}\left(\frac{\partial u^*}{\partial n}\right)^2 ds + \frac{\lambda}{2}\int_{\partial\Omega}v^2 ds - \int_\Omega fv dx
$$

$$
= \frac{1}{2}a(u^*,u^*) + \frac{1}{2\lambda}\int_{\partial\Omega}\left(\frac{\partial u^*}{\partial n}\right)^2 ds + \mathcal{L}_\lambda(v),
$$

where the second equality comes from that

$$a(u^*,v) - \int_{\partial\Omega} \frac{\partial u^*}{\partial n} v \, ds = \int_{\Omega} f v \, dx, \quad \forall v \in H^1(\Omega). \tag{3.17}$$

Since $R_\lambda(v) = \mathcal{L}_\lambda(v) + const$, u_λ^* is also the minimizer of R_λ over $H^1(\Omega)$. Recall (3.16), we obtain the estimation of $R_\lambda(u_\lambda^*)$

$$0 \le R_\lambda(u_\lambda^*) = \frac{1}{2} a(u^* - u_\lambda^*, u^* - u_\lambda^*) + \frac{\lambda}{2} \int_{\partial\Omega} \left(-\frac{1}{\lambda} \frac{\partial u^*}{\partial n} - u_\lambda^* \right)^2 ds \le R_\lambda(w) \le C\lambda^{-2}.$$

Now that $a(\cdot,\cdot)$ is coercive, we arrive at

$$\|u^* - u_\lambda^*\|_{H^1(\Omega)} \le C_{c_3,d} \lambda^{-1}.$$

This completes the proof. $\qquad\square$

In [23], they proved that the error $\|u_\lambda^* - u^*\|_{H^1(\Omega)} \le \mathcal{O}(\lambda^{-1/2})$, which is suboptimal comparing with the above results derived here. In [35,36], they proved the $\mathcal{O}(\lambda^{-1})$ bound under some unverifiable conditions.

3.4 Convergence rate

Note that for $\lambda \to \infty$ the approximation error $\mathcal{E}_{app}$ and the statistical error $\mathcal{E}_{sta}$ approach ∞ and for $\lambda \to 0$ the error from penalty blows up. Hence, there must be a trade off for choosing proper λ.

Theorem 3.5. *Let u^* be the weak solution of (2.1) with bounded $f \in L^2(\Omega)$, $w \in L^\infty(\Omega)$. $\widehat{u}_\phi$ is the minimizer of the discrete version of the associated Robin energy with parameter λ. Letting $n \ge 2$ be the number of training samples on the domain and the boundary (i.e., $N = M = n$), there is a ReLU^2 network with depth and width as*

$$\mathcal{D} \le \lceil \log_2 d \rceil + 3, \quad \mathcal{W} \le \mathcal{O}\left(4d \left\lceil \left(\frac{n}{\log n} \right)^{\frac{1}{2(d+2)}} - 4 \right\rceil^d \right),$$

such that

$$\mathbb{E}_{X,Y}\left[\|\widehat{u}_\phi - u^*\|_{H^1(\Omega)}^2 \right] \le C_{c_1,c_2,c_3,d} \mathcal{O}\left(n^{-\frac{1}{d+2}} (\log n)^{\frac{d+3}{d+2}} \right)$$
$$+ C_{c_1,c_2,c_3,d} \mathcal{O}\left(n^{-\frac{1}{d+2}} (\log n)^{\frac{d+3}{d+2}} \right) \lambda$$
$$+ C_{c_3,d} \lambda^{-2}.$$

Furthermore, for

$$\lambda \sim n^{\frac{1}{3(d+2)}} (\log n)^{-\frac{d+3}{3(d+2)}},$$

it holds that

$$\mathbb{E}_{X,Y}\left[\|\widehat{u}_\phi - u^*\|_{H^1(\Omega)}^2 \right] \le C_{c_1,c_2,c_3,d} \mathcal{O}\left(n^{-\frac{2}{3(d+2)}} \log n \right).$$

Proof. Combining Theorem 3.1, Theorem 3.2 and Theorem 3.3, we obtain by taking $\varepsilon^2 = C_{c_1,c_2,c_3}\left(\frac{\log n}{n}\right)^{\frac{1}{d+2}}$

$$\mathbb{E}_{X,Y}[\|\widehat{u}_\phi - u_\lambda^*\|_{H^1(\Omega)}^2]$$

$$\leq \frac{2}{c_1 \wedge 1}\left[C_{c_3}d(\mathcal{D}+3)(\mathcal{D}+2)\mathcal{W}\sqrt{\mathcal{D}+3+\log(d(\mathcal{D}+2)\mathcal{W})}\left(\frac{\log n}{n}\right)^{1/2} + \frac{c_3+1}{2}\varepsilon^2\right]$$

$$+ \frac{2}{c_1 \wedge 1}\left[C_{c_3}d\mathcal{D}\mathcal{W}\sqrt{\mathcal{D}+\log\mathcal{W}}\left(\frac{\log n}{n}\right)^{1/2} + \frac{C_d}{2}\varepsilon^2\right]\lambda$$

$$\leq \frac{2}{c_1 \wedge 1}\left[C_{c_3}4d^2(\lceil\log d\rceil+6)(\lceil\log d\rceil+5)\left\lceil\frac{Cc_2}{\varepsilon}-4\right\rceil^d \cdot\right.$$

$$\sqrt{\lceil\log d\rceil+6+\log\left(4d^2(\lceil\log d\rceil+5)\left\lceil\frac{Cc_2}{\varepsilon}-4\right\rceil^d\right)}\left(\frac{\log n}{n}\right)^{1/2} + \frac{c_3+1}{2}\varepsilon^2\Bigg]$$

$$+ \frac{2}{c_1 \wedge 1}\left[C_{c_3}4d^2(\lceil\log d\rceil+3)\left\lceil\frac{Cc_2}{\varepsilon}-4\right\rceil^d \cdot\right.$$

$$\sqrt{\lceil\log d\rceil+3+\log\left(4d\left\lceil\frac{Cc_2}{\varepsilon}-4\right\rceil^d\right)}\left(\frac{\log n}{n}\right)^{1/2} + \frac{C_d}{2}\varepsilon^2\Bigg]\lambda$$

$$\leq C_{c_1,c_2,c_3,d}\mathcal{O}\left(n^{-\frac{1}{d+2}}(\log n)^{\frac{d+3}{d+2}}\right) + C_{c_1,c_2,c_3,d}\mathcal{O}\left(n^{-\frac{1}{d+2}}(\log n)^{\frac{d+3}{d+2}}\right)\lambda.$$

Using Theorem 3.1 and Theorem 3.4, it holds that for all $\lambda > 0$

$$\mathbb{E}_{X,Y}\left[\|\widehat{u}_\phi - u^*\|_{H^1(\Omega)}^2\right] \leq C_{c_1,c_2,c_3,d}\mathcal{O}\left(n^{-\frac{1}{d+2}}(\log n)^{\frac{d+3}{d+2}}\right)$$

$$+ C_{c_1,c_2,c_3,d}\mathcal{O}\left(n^{-\frac{1}{d+2}}(\log n)^{\frac{d+3}{d+2}}\right)\lambda$$

$$+ C_{c_3,d}\lambda^{-2}. \tag{3.18}$$

We have derive the error estimate for fixed λ, and now we are in a position to find a proper λ and get the convergence rate. Since (3.18) holds for any $\lambda > 0$, we take the infimum of λ:

$$\mathbb{E}_{X,Y}\left[\|\widehat{u}_\phi - u^*\|_{H^1(\Omega)}^2\right] \leq \inf_{\lambda>0}\left\{C_{c_1,c_2,c_3,d}\mathcal{O}\left(n^{-\frac{1}{d+2}}(\log n)^{\frac{d+3}{d+2}}\right)\right.$$

$$+ C_{c_1,c_2,c_3,d}\mathcal{O}\left(n^{-\frac{1}{d+2}}(\log n)^{\frac{d+3}{d+2}}\right)\lambda$$

$$+ C_{c_3,d}\lambda^{-2}\Big\}.$$

By taking

$$\lambda \sim n^{\frac{1}{3(d+2)}}(\log n)^{-\frac{d+3}{3(d+2)}},$$

we can obtain

$$\mathbb{E}_{X,Y}\left[\|\widehat{u}_\phi - u^*\|_{H^1(\Omega)}^2\right] \leq C_{c_1,c_2,c_3,d}\mathcal{O}\left(n^{-\frac{2}{3(d+2)}}\log n\right).$$

This completes the proof.   □

4 Conclusions and extensions

This paper provided an analysis of convergence rate for deep Ritz methods for Laplace equations with Dirichlet boundary condition. More precisely, we provide a rule to choose the depth and width of networks and to choose a proper penalty parameter to achieve the desired convergence rate in terms of number of training samples. The estimation on the approximation error of deep ReLU^2 network is established in H^1. The statistical error can be derived technically by the Rademacher complexity of the non-Lipschitz composition of gradient norm and ReLU^2 network. We also give the error estimation to the boundary penalty method.

There are several interesting further research directions. First, the current analysis can be extended to general second order elliptic equations with other boundary conditions. Second, the approximation and statistical error bounds deriving here can be used for studying the nonasymptotic convergence rate for residual based method, such as PINNs. Finally, the similar result may be applicable to deep Ritz methods for optimal control problems and inverse problems.

Acknowledgments

Appendix

A.1 Proof of Lemma 2.2

We claim that a_λ is coercive on $H^1(\Omega)$. In fact,

$$a_\lambda(u,u) = a(u,u) + \lambda\int_{\partial\Omega} u^2 \, \mathrm{d}s \geq C\|u\|_{H^1(\Omega)}^2, \quad \forall u \in H^1(\Omega),$$

where C is constant from Poincaré inequality [17]. Thus, there exists a unique weak solution $u_\lambda^* \in H^1(\Omega)$ such that

$$a_\lambda(u_\lambda^*,v) = f(v), \quad \forall v \in H^1(\Omega).$$

We can check u_λ^* is the unique minimizer of $\mathcal{L}_\lambda(u)$ by standard technique.

We will study the regularity for weak solutions of (2.4). For the following discussion, we first introduce several useful classic results of second order elliptic equations in [16, 17].

Lemma A.1. *Assume $w \in L^\infty(\Omega)$, $f \in L^2(\Omega)$, $g \in H^{3/2}(\partial\Omega)$ and $\partial\Omega$ is sufficiently smooth. Suppose that $u \in H^1(\Omega)$ is a weak solution of the elliptic boundary-value problem*

$$\begin{cases} -\Delta u + wu = f, & \text{in } \Omega, \\ u = g, & \text{on } \partial\Omega. \end{cases}$$

Then $u \in H^2(\Omega)$ and there exists a positive constant C, depending only on Ω and w, such that

$$\|u\|_{H^2(\Omega)} \le C \left(\|f\|_{L^2(\Omega)} + \|g\|_{H^{3/2}(\partial\Omega)} \right).$$

Proof. See [16]. □

Lemma A.2. *Assume $w \in L^\infty(\Omega)$, $f \in L^2(\Omega)$, $g \in H^{1/2}(\partial\Omega)$ and $\partial\Omega$ is sufficiently smooth. Suppose that $u \in H^1(\Omega)$ is a weak solution of the elliptic boundary-value problem*

$$\begin{cases} -\Delta u + wu = f, & \text{in } \Omega, \\ \dfrac{\partial u}{\partial n} = g, & \text{on } \partial\Omega. \end{cases}$$

Then $u \in H^2(\Omega)$ and there exists a positive constant C, depending only on Ω and w, such that

$$\|u\|_{H^2(\Omega)} \le C \left(\|f\|_{L^2(\Omega)} + \|g\|_{H^{1/2}(\partial\Omega)} \right).$$

Proof. See [17]. □

Lemma A.3. *Assume $w \in L^\infty(\Omega)$, $g \in H^{1/2}(\partial\Omega)$, $\partial\Omega$ is sufficiently smooth and $\lambda > 0$. Let $u \in H^1(\Omega)$ be the weak solution of the following Robin problem*

$$\begin{cases} -\Delta u + wu = 0, & \text{in } \Omega, \\ \dfrac{1}{\lambda} \dfrac{\partial u}{\partial n} + u = g, & \text{on } \partial\Omega. \end{cases} \tag{A.1}$$

Then $u \in H^2(\Omega)$ and there exists a positive constant C independent of λ such that

$$\|u\|_{H^2(\Omega)} \le C\lambda \|g\|_{H^{1/2}(\partial\Omega)}.$$

Proof. Following the idea which is proposed in [10] in a slightly different context. We first estimate the trace $Tu = u|_{\partial\Omega}$. We define the Dirichlet-to-Neumann map

$$\widetilde{T}: u|_{\partial\Omega} \mapsto \left. \frac{\partial u}{\partial n} \right|_{\partial\Omega},$$

where u satisfies $-\Delta u + wu = 0$ in Ω, then

$$Tu = \left(\frac{1}{\lambda}\widetilde{T} + I\right)^{-1} g.$$

Now we are going to show that $\frac{1}{\lambda}\widetilde{T} + I$ is a positive definite operator in $L^2(\partial\Omega)$. We notice that the variational formulation of (A.1) can be read as follow:

$$\int_\Omega \nabla u \cdot \nabla v \, dx + \int_\Omega wuv \, dx + \lambda \int_{\partial\Omega} uv \, ds = \lambda \int_{\partial\Omega} gv \, ds, \quad \forall v \in H^1(\Omega).$$

Taking $v = u$, then we have

$$\|Tu\|^2_{L^2(\partial\Omega)} \leq \left\langle \left(\frac{1}{\lambda}\widetilde{T} + I\right) Tu, Tu \right\rangle.$$

This means that $\lambda^{-1}\widetilde{T} + I$ is a positive definite operator in $L^2(\partial\Omega)$, and further, $(\lambda^{-1}\widetilde{T} + I)^{-1}$ is bounded. We have the estimate

$$\|Tu\|_{H^{1/2}(\partial\Omega)} \leq C\|g\|_{H^{1/2}(\partial\Omega)}. \tag{A.2}$$

We rewrite the Robin problem (A.1) as follows

$$\begin{cases} -\Delta u + wu = 0, & \text{in } \Omega, \\ \dfrac{\partial u}{\partial n} + u = \lambda\left(g - \left(1 - \lambda^{-1}\right)u\right), & \text{on } \partial\Omega. \end{cases}$$

By Lemma A.2 we have

$$\|u\|_{H^2(\Omega)} \leq C\lambda \left\|g - \left(1 - \lambda^{-1}\right) Tu\right\|_{H^{1/2}(\partial\Omega)} \leq C\lambda \left(\|g\|_{H^{1/2}(\partial\Omega)} + \|Tu\|_{H^{1/2}(\partial\Omega)}\right). \tag{A.3}$$

Combining (A.2) and (A.3), we obtain the desired estimation. □

With the help of above lemmas, we now turn to proof the regularity properties of the weak solution.

Theorem A.1. *Assume $w \in L^\infty(\Omega)$, $f \in L^2(\Omega)$. Suppose that $u \in H^1(\Omega)$ is a weak solution of the boundary-value problem (2.4). If $\partial\Omega$ is sufficiently smooth, then $u \in H^2(\Omega)$, and we have the estimate*

$$\|u\|_{H^2(\Omega)} \leq C\|f\|_{L^2(\Omega)},$$

where the constant C depending only on Ω and w.

Proof. We decompose (2.4) into two equations

$$\begin{cases} -\Delta u_0 + w u_0 = f, & \text{in } \Omega, \\ u_0 = 0, & \text{on } \partial\Omega, \end{cases} \tag{A.4}$$

$$\begin{cases} -\Delta u_1 + w u_1 = 0, & \text{in } \Omega, \\ \dfrac{1}{\lambda}\dfrac{\partial u_1}{\partial n} + u_1 = -\dfrac{\partial u_0}{\partial n}, & \text{on } \partial\Omega, \end{cases} \tag{A.5}$$

and obtain the solution of (2.4)

$$u = u_0 + \frac{1}{\lambda} u_1.$$

Applying Lemma A.1 to (A.4), we have

$$\|u_0\|_{H^2(\Omega)} \le C \|f\|_{L^2(\Omega)}, \tag{A.6}$$

where C depends on Ω and w. Using Lemma A.3, it is easy to obtain

$$\|u_1\|_{H^2(\Omega)} \le C\lambda \left\| \frac{\partial u_0}{\partial n} \right\|_{H^{1/2}(\partial\Omega)} \le C\lambda \|u_0\|_{H^2(\Omega)}, \tag{A.7}$$

where the last inequality follows from the trace theorem. Combining (A.6) and (A.7), the desired estimation can be derived by triangle inequality. $\qquad\square$

References

[1] Anandkumar, A., Azizzadenesheli, K., Bhattacharya, K., Kovachki, N., Li, Z., Liu, B., Stuart, A.: Neural operator: Graph kernel network for partial differential equations. In: ICLR 2020 Workshop on Integration of Deep Neural Models and Differential Equations (2020)

[2] Anitescu, C., Atroshchenko, E., Alajlan, N., Rabczuk, T.: Artificial neural network methods for the solution of second order boundary value problems. Cmc-computers Materials & Continua **59**(1), 345–359 (2019)

[3] Anthony, M., Bartlett, P.L.: Neural Network Learning: Theoretical Foundations. Cambridge University Press (2009)

[4] Babuska, I.: The finite element method with penalty. Mathematics of Computation (1973)

[5] Bartlett, P.L., Harvey, N., Liaw, C., Mehrabian, A.: Nearly-tight vc-dimension and pseudodimension bounds for piecewise linear neural networks. J. Mach. Learn. Res. **20**(63), 1–17 (2019)

[6] Berner, J., Dablander, M., Grohs, P.: Numerically solving parametric families of high-dimensional kolmogorov partial differential equations via deep learning. In: H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, H. Lin (eds.) Advances in Neural Information Processing Systems, vol. 33, pp. 16615–16627. Curran Associates, Inc. (2020)

[7] Boucheron, S., Lugosi, G., Massart, P.: Concentration Inequalities: A Nonasymptotic Theory of Independence. Oxford University Press (2013)

[8] Brenner, S., Scott, R.: The Mathematical Theory of Finite Element Methods, vol. 15. Springer Science & Business Media (2007)

[9] Ciarlet, P.G.: The Finite Element Method for Elliptic Problems. SIAM (2002)

[10] Costabel, M., Dauge, M.: A singularly perturbed mixed boundary value problem. Communications in Partial Differential Equations **21** (1996)

[11] De Boor, C., De Boor, C.: A Practical Guide to Splines, vol. 27. Springer-Verlag New York (1978)

[12] Dissanayake, M., Phan-Thien, N.: Neural-network-based approximations for solving partial differential equations. Communications in Numerical Methods in Engineering **10**(3), 195–201 (1994)

[13] Dudley, R.: The sizes of compact subsets of hilbert space and continuity of Gaussian processes. Journal of Functional Analysis **1**(3), 290–330 (1967). 10.4208/csiam-am.SO-2021-0043https://doi.org/10.1016/0022-1236(67)90017-1. URL https://www.sciencedirect.com/science/article/pii/0022123667900171

[14] E, W., Ma, C., Wu, L.: The Barron space and the flow-induced function spaces for neural network models (2021)

[15] E, W., Wojtowytsch, S.: Some observations on partial differential equations in Barron and multi-layer spaces (2020)

[16] Evans, L.C.: Partial Differential Equations, second edition (2010)

[17] Gilbarg, D., Trudinger, N.: Elliptic Partial Differential Equations of Second Order, 2nd ed. Springer (1998)

[18] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial networks. Advances in Neural Information Processing Systems **3** (2014). 10.4208/csiam-am.SO-2021-004310.1145/3422622

[19] Greenfeld, D., Galun, M., Basri, R., Yavneh, I., Kimmel, R.: Learning to optimize multigrid PDE solvers. In: K. Chaudhuri, R. Salakhutdinov (eds.) Proceedings of the 36th International Conference on Machine Learning, *Proceedings of Machine Learning Research*, vol. 97, pp. 2415–2423. PMLR (2019)

[20] Gühring, I., Kutyniok, G., Petersen, P.: Error bounds for approximations with deep ReLU neural networks in $w^{s,p}$ norms (2019)

[21] Han, J., Jentzen, A., E, W.: Solving high-dimensional partial differential equations using deep learning. Proceedings of the National Academy of Sciences **115**(34), 8505–8510 (2018). 10.4208/csiam-am.SO-2021-004310.1073/pnas.1718942115. URL https://www.pnas.org/content/115/34/8505

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

[23] Hong, Q., Siegel, J.W., Xu, J.: Rademacher complexity and numerical quadrature analysis of stable neural networks with applications to numerical pdes (2021)

[24] Hsieh, J.T., Zhao, S., Eismann, S., Mirabella, L., Ermon, S.: Learning neural PDE solvers with convergence guarantees. In: International Conference on Learning Representations (2018)

[25] Hughes, T.J.: The Finite Element Method: Linear Static and Dynamic Finite Element Analysis. Courier Corporation (2012)

[26] Lagaris, I.E., Likas, A., Fotiadis, D.I.: Artificial neural networks for solving ordinary and partial differential equations. IEEE Trans. Neural Networks **9**(5), 987–1000 (1998). URL https://doi.org/10.1109/72.712178

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

[28] Li, Z., Kovachki, N., Azizzadenesheli, K., Liu, B., Stuart, A., Bhattacharya, K., Anand-

kumar, A.: Multipole graph neural operator for parametric partial differential equations. In: H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, H. Lin (eds.) Advances in Neural Information Processing Systems, vol. 33, pp. 6755–6766. Curran Associates, Inc. (2020). URL https://proceedings.neurips.cc/paper/2020/file/4b21cf96d4cf612f239a6c322b10c8fe-Paper.pdf

[29] Li, Z., Kovachki, N.B., Azizzadenesheli, K., liu, B., Bhattacharya, K., Stuart, A., Anandkumar, A.: Fourier neural operator for parametric partial differential equations. In: International Conference on Learning Representations (2021)

[30] Lu, J., Lu, Y., Wang, M.: A priori generalization analysis of the deep Ritz method for solving high dimensional elliptic equations (2021)

[31] Lu, L., Meng, X., Mao, Z., Karniadakis, G.E.: Deepxde: A deep learning library for solving differential equations. CoRR **abs/1907.04502** (2019). URL http://arxiv.org/abs/1907.04502

[32] Luo, T., Yang, H.: Two-layer neural networks for partial differential equations: Optimization and generalization theory. ArXiv **abs/2006.15733** (2020)

[33] Maury, B.: Numerical analysis of a finite element/volume penalty method. Siam Journal on Numerical Analysis **47**(2), 1126–1148 (2009)

[34] Mishra, S., Molinaro, R.: Estimates on the generalization error of physics informed neural networks (PINNs) for approximating pdes. ArXiv **abs/2007.01138** (2020)

[35] Müller, J., Zeinhofer, M.: Deep Ritz revisited. In: ICLR 2020 Workshop on Integration of Deep Neural Models and Differential Equations (2020)

[36] Müller, J., Zeinhofer, M.: Error estimates for the variational training of neural networks with boundary penalty. ArXiv **abs/2103.01007** (2021)

[37] Quarteroni, A., Valli, A.: Numerical Approximation of Partial Differential Equations, vol. 23. Springer Science & Business Media (2008)

[38] Raissi, M., Perdikaris, P., Karniadakis, G.E.: Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. Journal of Computational Physics **378**, 686–707 (2019)

[39] Schultz, M.H.: Approximation theory of multivariate spline functions in sobolev spaces. SIAM Journal on Numerical Analysis **6**(4), 570–582 (1969)

[40] Schumaker, L.: Spline Functions: Basic Theory. Cambridge University Press (2007)

[41] Shin, Y., Zhang, Z., Karniadakis, G.: Error estimates of residual minimization using neural networks for linear pdes. ArXiv **abs/2010.08019** (2020)

[42] Silver, D., Huang, A., Maddison, C.J., Guez, A., Sifre, L., Van Den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., et al.: Mastering the game of go with deep neural networks and tree search. nature **529**(7587), 484–489 (2016)

[43] Sirignano, J.A., Spiliopoulos, K.: Dgm: A deep learning algorithm for solving partial differential equations. Journal of Computational Physics **375**, 1339–1364 (2018)

[44] Thomas, J.: Numerical Partial Differential Equations: Finite Difference Methods, vol. 22. Springer Science & Business Media (2013)

[45] Um, K., Brand, R., Fei, Y.R., Holl, P., Thuerey, N.: Solver-in-the-loop: Learning from differentiable physics to interact with iterative pde-solvers. In: H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, H. Lin (eds.) Advances in Neural Information Processing Systems, vol. 33, pp. 6111–6122. Curran Associates, Inc. (2020)

[46] Wang, S., Yu, X., Perdikaris, P.: When and why PINNs fail to train: A neural tangent kernel perspective. ArXiv **abs/2007.14527** (2020)

[47] Wang, Y., Shen, Z., Long, Z., Dong, B.: Learning to discretize: Solving 1d scalar conservation laws via deep reinforcement learning. Communications in Computational Physics **28**(5),

2158–2179 (2020)

[48] Weinan, E., Yu, B.: The deep Ritz method: A deep learning-based numerical algorithm for solving variational problems. Communications in Mathematics and Statistics **6**(1), 1–12 (2017)

[49] Xu, J.: Finite neuron method and convergence analysis. Communications in Computational Physics **28**(5), 1707–1745 (2020)

[50] Zang, Y., Bao, G., Ye, X., Zhou, H.: Weak adversarial networks for high-dimensional partial differential equations. Journal of Computational Physics **411**, 109409 (2020)