SOLVING ELLIPTIC PROBLEMS WITH SINGULAR SOURCES USING SINGULARITY SPLITTING DEEP RITZ METHOD*

TIANHAO HU†, BANGTI JIN‡, AND ZHI ZHOU§

Abstract. In this work, we develop an efficient solver based on neural networks for second-order elliptic equations with variable coefficients and a singular source. This class of problems covers general point sources, line sources, and the combination of point-line sources and has a broad range of practical applications. The proposed approach is based on decomposing the true solution into a singular part that is known analytically using the fundamental solution of the Laplace equation and a regular part that satisfies a suitable modified elliptic PDE with a smoother source and then solving for the regular part using the deep Ritz method. A path-following strategy is suggested to select the penalty parameter for enforcing the Dirichlet boundary condition. Extensive numerical experiments in two- and multi-dimensional spaces with point sources, line sources, or their combinations are presented to illustrate the efficiency of the proposed approach, and a comparative study with several existing approaches based on neural networks is also given, which shows clearly its competitiveness for the specific class of problems. In addition, we briefly discuss the error analysis of the approach.

Key words. variable coefficient Poisson equation, singular source, deep Ritz method, penalty method, neural networks

MSC code. 65N12

DOI. 10.1137/22M1520840

1. Introduction. Partial differential equations (PDEs) represent a very important class of mathematical models that plays a vital role in physics, science, and engineering. Traditional numerical methods for solving PDEs include the finite difference method (FDM), the finite element method (FEM), the finite volume method (FVM), the boundary element method (BEM), etc. These methods have been maturely developed over the past few decades, and efficient implementations and rigorous theoretical guarantees, e.g., error estimates, are also readily available. In recent years, neural networks (NNs) have emerged as a promising way to solve PDEs [43], motivated by their great successes in computer vision, speech recognition, natural language processing, etc. A large number of neural PDE solvers have been developed, including physics informed neural networks (PINNs) [55], the deep Ritz method (DRM) [71], the deep Galerkin method (DGM) [61], the weak adversarial network (WAN) [72], and the deep least-squares method [10], to name just a few. Compared with more traditional methods, deep learning solvers have demonstrated very encouraging results in several direct and inverse problems [39, 21, 38, 14].

†School of Mathematics, Jilin University, Changchun 130012, People's Republic of China (huth1019@mails.jlu.edu.cn).

‡Department of Mathematics, The Chinese University of Hong Kong, Shatin, New Territories, Hong Kong, People's Republic of China (b.jin@cuhk.edu.hk).

§Department of Applied Mathematics, The Hong Kong Polytechnic University, Kowloon, Hong Kong, People's Republic of China (zhi.zhou@polyu.edu.hk).

Note that in all these approaches, one uses NNs as ansatz functions to approximate the solutions to the PDEs. According to the existing approximation theory of NNs [18], the convergence rates of these methods depend heavily on the regularity of the solution (as well as the stability of the mathematical formulation). A direct application of these methods might be ineffective or even fail spectacularly when dealing with challenging scenarios [68, 42], e.g., convection-dominated problems, transport problems, high-frequency wave propagation, problems with geometric singularities (cracks/corner singularity), and singular sources. Note that all these settings lead to either strong directional behavior or weak solution singularities or highly oscillatory behavior, which are challenging for physics-agnostic NNs to capture effectively.

1.1. Mathematical formulation. This work deals with one challenging class of boundary value problems, i.e., second-order elliptic problems of divergence form with singular sources. Let Ω be an open bounded domain in $\mathbb{R}^d$ ($d \geq 2$) with a smooth boundary $\partial\Omega$. Without loss of generality, consider the following variable coefficient second-order elliptic equation in the divergence form with a singular source:

$$(1.1) \qquad -\nabla \cdot (\kappa \nabla u) = S + g \quad \text{in } \Omega,$$

where the scalar-valued diffusion coefficient $\kappa \in C^2(\overline{\Omega})$ satisfies uniform ellipticity, i.e., $\kappa(\mathbf{x}) \geq m$, for some fixed constant $m > 0$, and S and g denote a singular and smooth source, respectively. Here the singular source S can be expressed in terms of the Dirac delta function and covers both point and line sources. Specifically, let $\delta(\mathbf{x})$ denote the Dirac delta function concentrated at the origin, defined in the sense of distribution, i.e., $\langle \delta, v \rangle = v(\mathbf{0})$ for all $v \in C(\overline{\Omega})$. Likewise, with Λ being a line/line segment in the domain Ω, let δ_Λ be a Dirac measure concentrated on Λ, such that $\langle \delta_\Lambda, v \rangle = \int_\Lambda v \, \mathrm{d}s$ for all $v \in C(\overline{\Omega})$. Then the singular source S can take the form

$$S(\mathbf{x}) = \sum_{i=1}^{N_p} c_i \delta(\mathbf{x} - \mathbf{x}_i) + \sum_{j=1}^{N_L} f_j \delta_{\Lambda_j}(\mathbf{x}),$$

where $\{(c_i, \mathbf{x}_i) : c_i \in \mathbb{R}, \mathbf{x}_i \in \Omega\}_{i=1}^{N_p}$ denotes a collection of N_p Dirac point sources in the domain, concentrated at $\mathbf{x}_i$ with a strength c_i, and $\{(f_j, \Lambda_j) : f_j \in C(\overline{\Omega}), \Lambda_j \subset \Omega\}_{j=1}^{N_L}$ denotes a collection of N_L line sources, supported on the line segment Λ_j with the corresponding density f_j. The proposed approach can be adapted to surface sources, etc. Singular sources supported on high-dimensional sets induce a weaker singularity in the solution than point sources.

This class of problems has a broad range of practical applications. For example, a point source can describe a pulse exciting the electric field in electromagnetic simulation [24, 63] or volume injection in an acoustic wave equation [50]. In practice, the source can concentrate not only on points but also on lines, curves, or surfaces, and the latter arises naturally in biological modeling. For example, the works [30, 57, 49] discuss blood flow in the vascularized tissue of the brain, and the works [12, 17] studied drug delivery through microcirculation and tissue perfusion. In these applications, the resulting mathematical models are PDEs with line source(s). Hence, it is also of great interest to develop efficient numerical solvers for these problems.

Elliptic problems with point sources have been extensively studied in the context of classical numerical PDE solvers, and there are several different approaches to handle the singularity. In some situations, traditional solvers, mostly FEMs, can be applied directly but require careful interpretation, and several (weighted) $L^2(\Omega)$

error estimates have been obtained (see, e.g., [7, 5, 16, 41]). Due to the limited solution regularity, there is no error estimate in the H^1 norm. Alternatively, one may regularize the Dirac function with a smooth function (e.g., Gaussians or Taylor expansion technique) or approximate the solution in weighted Sobolev spaces (with a weight vanishing at the singularity) so that the singularity of the solution plays a less prominent role (see [34, 22, 45] and references therein). Once the source singularity is regularized, the problem can then be solved using any stand-alone techniques, e.g., FDM or FEM. There are also several alternative approaches taking into account the problem structures, e.g., mesh grading [5], generalized FEM [26], and singularity reconstruction [11]. In particular, these approaches build the analytical insights into the construction of the numerical procedures directly in order to achieve better computational efficiency. So far most studies are concerned with point sources. The case of line sources is far less studied. The few exceptions include D'Angelo and Quarteroni (see [17, 16]), which gave the existence of a solution to the variational formulation (in weighted Sobolev spaces) and studied its finite element approximation, and Gjerde et al. (see [28, 27]), where a novel singularity splitting technique was suggested and investigated within the context of the Galerkin and mixed formulations. In particular, the singularity splitting technique [28, 27] enjoys improved approximation properties: it yields optimal convergence rates for lowest-order elements and removes the pollution around the line source [28].

NN-based solvers have not been extensively studied for this class of problems. Due to the low regularity of the source term, the standard PINN or deep Ritz formulations cannot be applied directly. Indeed, the corresponding mathematical formulations are not always well-defined, and this is also clearly manifested in the numerical experiments. See section 2 for detailed discussions on this aspect of the challenges. It is worth pointing out that due to the persistent optimization error, none of existing neural PDE solvers can exhibit a steady and consistent convergence rate in practical computation, which is in stark contrast to classical PDE solvers, e.g., FEM and FDM. See the recent works [2, 15] for detailed discussions on this important issue.

1.2. The contributions of this work and organization. In this work, we propose a modification of the DRM [71], called singularity splitting DRM (SSDRM), that can be applied effectively for the numerical solution of second-order elliptic PDEs with singular sources. Our approach is inspired by [28, 53]: we split the solution into a singular and a regular part and then approximate only the regular part using NNs. The singular part is explicitly expressed in terms of the fundamental solution Φ to the Laplace equation, and the regular part is smoother and satisfies a modified elliptic PDE that can be effectively solved using the DRM. Due to the improved regularity of the regular part, NNs are effective for the numerical approximation, as indicated by the approximation theory of NNs [70, 32]. The overall approach is as easy to implement as the original DRM, but significantly improves the accuracy of the approximation by carefully exploiting the known analytic structure of the underlying exact solution, which follows closely the current paradigm of building physical insights into machine learning techniques for problems in computational science and engineering.

The extensive numerical experiments in two- and multi-dimensions in section 4 show clearly the efficiency and accuracy of the proposed SSDRM; especially, extracting the solution singularity can greatly enhance the approximation accuracy. The comparative study shows that it significantly outperforms existing NN-based approaches in terms of accuracy. In addition, we also briefly discuss the associated theoretical

issues of the approach, which provide guidelines on the choice of the penalty parameter and the NN architecture. More precisely, our contributions can be summarized as follows:

 (i) We develop a novel singularity splitting based technique for problem (1.1) to enable the use of DRM.
 (ii) We provide relevant theoretical underpinning of SSDRM, including both penalization and generalization errors.
 (iii) We carry out extensive numerical examples in two- and multi-dimensions, with one or multiple singularities to illustrate the flexibility and accuracy of SSDRM.
 (iv) We conduct a comparative study with several state-of-the-art NN-based PDE solvers.

The rest of the paper is organized as follows. In section 2, we review existing NN-based PDE solvers and discuss challenges associated with their applications to problem (1.1). In section 3, we propose the numerical method, based on the idea of singularity splitting. Then in section 4 we present extensive numerical experiments to illustrate the accuracy and efficiency of the approach. Finally, we discuss relevant theoretical issues in section 5, and conclude the paper with a short conclusion. Throughout, we denote by $(\cdot, \cdot)$ and $(\cdot, \cdot)_{L^2(\partial\Omega)}$ the $L^2(\Omega)$ and $L^2(\partial\Omega)$ inner products, respectively. The notation $|\cdot|$ and $\cdot$ denote, respectively, the usual Euclidean norm and inner product on $\mathbb{R}^d$.

2. Challenges for NN-based solvers. The presence of a singular source in the model (1.1) poses big challenges to its efficient numerical approximation. This is particularly severe for NN-based approaches, which tend to approximate smooth solutions well due to their implicit structural bias [62, 67, 68] but not weakly singular solutions as arising in problem (1.1). Below we illustrate the inherent computational challenges for three existing NN solvers, i.e., PINN, DRM, and WAN, for solving a model version of problem (1.1) with a Dirichlet boundary condition

$$(2.1) \qquad \begin{cases} -\nabla \cdot (\kappa\nabla u) = \delta(\mathbf{x} - \mathbf{x}_0) + g & \text{in } \Omega, \\ u = h & \text{on } \partial\Omega \end{cases}$$

(i.e., with a point source $\delta(\mathbf{x} - \mathbf{x}_0)$, $\mathbf{x}_0 \in \Omega$) in their continuous formulations. In practice, all these solvers approximate the PDE solution u directly using NNs, which are then discretized properly.

A physics informed neural network (PINN) is based on PDE residual minimization. When applying PINN to solve problem (2.1) directly, the loss function $L(u)$ is commonly taken to be

$$(2.2) \qquad \min_{u \in H^1(\Omega)} L(u) = L_r(u) + \lambda L_b(u),$$

where $\lambda > 0$ is a penalty parameter to approximately enforce the Dirichlet boundary condition in (2.1), and the physical loss $L_r(u)$ and boundary loss $L_b(u)$ are given, respectively, by

$$L_r(u) = \| - \nabla \cdot (\kappa\nabla u) - \delta(\mathbf{x} - \mathbf{x}_0) - g\|^2_{L^2(\Omega)} \quad \text{and} \quad L_b(u) = \|u - h\|^2_{L^2(\partial\Omega)}.$$

Note that the point source $\delta(\mathbf{x} - \mathbf{x}_0)$ does not belong to the space $L^2(\Omega)$ but only the space of Radon measures, and thus the standard PINN loss $L_r(u)$ with the $L^2(\Omega)$ norm of the PDE residual is not well defined. Hence, one cannot apply PINN directly to problem (2.1), and this is also confirmed by numerical experiments. Indeed,

when applying PINN loss to solve (2.1) naively, there are pronounced errors near the singularity [56, 50]. This calls for proper modification of the original PINN in order to obtain good approximations, e.g., self-adaptive PINN described below. There are alternative formulations of PINNs, in the spirit of the weak formulation for PDEs, e.g., variational PINNs [40], that can partly alleviate the regularity requirement on the solution.

Next, we describe the deep Ritz method (DRM) [71], which is based on the Ritz formulation. For problem (2.1), it is given by

$$(2.3) \qquad \min_{u \in H^1(\Omega)} \tfrac{1}{2}(\kappa \nabla u, \nabla u) - u(\mathbf{x}_0) + \tfrac{\lambda}{2}\|u - h\|^2_{L^2(\partial\Omega)}.$$

Here, using the definition of $\delta(\mathbf{x} - \mathbf{x}_0)$, we eliminate the singularity in the source formally. By the Sobolev embedding theorem [1], the loss is well-posed in the one-dimensional case, but it is ill-defined when $d \geq 2$, since the point evaluation $u(\mathbf{x}_0)$ is not defined for a function $u \in H^1(\Omega)$ when $d \geq 2$. Numerically, it does not perform well for problem (1.1), which is also confirmed by the numerical experiments below.

The weak adversarial network (WAN) [72] is based on the Galerkin weak formulation. For problem (2.1), it is derived as follows. Let $v \in H_0^1(\Omega)$ be any test function. Then, by integration by parts,

$$(2.4) \qquad \int_\Omega (\kappa \nabla u \cdot \nabla v - \delta(\mathbf{x} - \mathbf{x}_0)v - gv)\, \mathrm{d}\mathbf{x} = 0.$$

Thus, the weak solution u solves the following min-max problem:

$$(2.5) \qquad \min_{u \in H^1(\Omega):Tu=h}\ \max_{v \in H_0^1(\Omega)} \frac{\int_\Omega (\kappa \nabla u \cdot \nabla v - \delta(\mathbf{x} - \mathbf{x}_0)v - gv)\, \mathrm{d}\mathbf{x}}{\left(\int_\Omega |\nabla v|^2 \mathrm{d}\mathbf{x}\right)^{1/2}}.$$

Zang et al. [72] proposed to employ two NNs for the trial function u and test function v. Due to its reliance on the weak formulation on $H^1(\Omega) \times H_0^1(\Omega)$, WAN suffers from the same issue as DRM, since it is not well-defined on the space $H^1(\Omega) \times H_0^1(\Omega)$. Additionally, since the resulting min-max problem involves two optimization steps, the training is often more delicate.

To address these challenges, several strategies have been proposed. One recent proposal is self-adaptive NNs [48]. In this approach, each term in the loss function is given a weight, and these weights are then adjusted to balance the effect of each term on the objective. This idea was recently employed to treat the influence brought by the singularity in PINN [35], and the resulting method is called SAPINN. Specifically, one modifies the loss function into

$$L(u) = \lambda_1 L_{r_1}(u) + \lambda_2 L_{r_2}(u) + \lambda_3 L_b(u),$$

where the three terms are given, respectively, by

$$L_{r_1}(u) = \| - \nabla \cdot (\kappa \nabla u) - \delta_H(\mathbf{x} - \mathbf{x}_0) - g\|^2_{L^2(B(\mathbf{x}_0, H))},$$
$$L_{r_2}(u) = \| - \nabla \cdot (\kappa \nabla u) - g\|^2_{L^2(\Omega \backslash B(\mathbf{x}_0, H))}, \quad \text{and} \quad L_b(u) = \|u - h\|^2_{L^2(\partial\Omega)},$$

and λ_i, $i = 1, 2, 3$, are positive weights to be adjusted, and $\delta_H(\mathbf{x} - \mathbf{x}_0)$ is a smooth approximation to the Dirac delta function $\delta(\mathbf{x} - \mathbf{x}_0)$ in a small ball $B(\mathbf{x}_0, H)$ (a ball of radius H, centered at $\mathbf{x}_0$). In this work, we follow [34] and choose the approximation δ_H to be

$$\delta_H(\mathbf{x}) = \begin{cases} \dfrac{12}{\pi H^d}\left(5\left(\dfrac{|\mathbf{x}|}{H}\right)^2 - 8\dfrac{|\mathbf{x}|}{H} + 3\right), & |\mathbf{x}| \le H, \\ \\ \qquad\qquad\qquad\qquad 0, & |\mathbf{x}| > H. \end{cases}$$

The partition of the domain Ω into two disjoint parts with different penalty parameters allows better compensation for the singular behavior, in a manner similar to weighted Sobolev spaces. The experimental results in [35] indicate that this method can improve the accuracy of the solution by PINN to some extent, but the increase in the number of hyperparameters (i.e., λ_is) also complicates the optimization process, due to the need for self-adaptation during training, which in turn makes the overall training lengthier; see section 4 for a detailed comparison.

3. Singularity splitting deep Ritz method. In this section, we propose a novel numerical method for solving the elliptic problem (1.1). In practice, in addition to the Dirichlet boundary condition (2.1), equation (1.1) may also be equipped with a Neumann or Robin one (possibly different conditions on different parts of the boundary $\partial\Omega$):

(3.1a) $$\kappa\frac{\partial u}{\partial n} = h \quad \text{on } \partial\Omega,$$

(3.1b) $$\alpha u + \kappa\frac{\partial u}{\partial n} = h \quad \text{on } \partial\Omega,$$

with the boundary data $h \in L^2(\partial\Omega)$ or its subspace, and $\alpha > 0$. Below we focus the discussion on the pure Dirichlet and Neumann boundary conditions, since the Robin case can be treated similarly. In the Neumann case, one additionally needs to impose the compatibility condition $\int_\Omega (f\delta + g)\mathrm{d}\mathbf{x} + \int_{\partial\Omega} h\mathrm{d}S = 0$, so that a solution does exist. Then the solution is only unique up to an arbitrary constant, and to have uniqueness, one can normalize the solution, e.g., $\int_\Omega u\mathrm{d}\mathbf{x} = 0$ or $\int_{\partial\Omega} u\mathrm{d}S = 0$, or just specify the value of the solution at one point in the domain/on the boundary. All these considerations have to be incorporated into the loss function.

3.1. Solution splitting. First we develop the singularity splitting technique for point and line sources. One key challenge in the numerical solution of problem (1.1) lies in the low solution regularity, and thus a direct approximation of the solution u via NNs can be inefficient. In order to overcome the challenge, we exploit analytic properties of the solution u, or more specifically extract the solution singularity directly and employ it in the solution procedure. Note that incorporating analytic insights into numerical procedures is a well-established computational paradigm, especially prominent in many analytic/semi-analytic methods. Notable examples include locally refined meshes [5], singular function enrichment [25], singularity reconstruction [11], and generalized FEM [26]. This work continues along this long established tradition and applies the idea in the context of neural PDE solvers for solving elliptic PDEs with singular sources.

First we describe the key analytic insights into the problem. Recall that the fundamental solution $\Phi(\mathbf{x})$ of the Laplace equation is given by [24, p. 22]

$$\Phi(\mathbf{x}) = \begin{cases} -\dfrac{1}{2\pi}\ln|\mathbf{x}|, & d = 2, \\ \dfrac{1}{d(d-2)\alpha(d)|\mathbf{x}|^{d-2}}, & d \ge 3, \end{cases}$$

where $\alpha(d) = \frac{\pi^{\frac{d}{2}}}{\Gamma(\frac{d}{2}+1)}$ denotes the volume of the unit ball in $\mathbb{R}^d$. Note that Φ is smooth away from the singularity at the origin. By definition, $\Phi(\mathbf{x})$ satisfies

$$-\Delta\Phi(\mathbf{x} - \mathbf{x}_0) = \delta(\mathbf{x} - \mathbf{x}_0) \quad \text{in } \mathbb{R}^d \setminus \{\mathbf{x}_0\},$$

where $\delta(\mathbf{x} - \mathbf{x}_0)$ denotes the Dirac delta function concentrated at the point $\mathbf{x}_0 \in \mathbb{R}^d$.

3.1.1. Singularity reconstruction for point sources. Using the fundamental solution $\Phi(\mathbf{x})$, we can split the solution u to problem (1.1) with one point source $S(\mathbf{x}) = c_0\delta(\mathbf{x} - \mathbf{x}_0)$ into

(3.2) $$u = c_0\kappa^{-1}\Phi + v,$$

where Φ is an abbreviation of $\Phi(\mathbf{x} - \mathbf{x}_0)$. The extension of the approach and the discussion below to the case of multiple Dirac functions is straightforward. In the splitting (3.2), the term $c_0\kappa^{-1}\Phi$ captures the leading singularity of the solution u near the point source $\mathbf{x}_0$ and v is the regular part. Since the fundamental solution $\Phi(\mathbf{x} - \mathbf{x}_0) \neq 0$ when $\mathbf{x} \in \partial\Omega$, the boundary condition and the source term have to be modified accordingly. For example, for the Dirichlet problem, by substituting the splitting (3.2) into (1.1), we get the following governing equation for the regular part v:

(3.3) $$\begin{cases} -\nabla \cdot (\kappa\nabla v) = F & \text{in } \Omega, \\ v = h - c_0\kappa^{-1}\Phi & \text{on } \partial\Omega, \end{cases}$$

with the modified source F given by

(3.4) $$F = g - c_0\kappa^{-1}[\Delta\kappa\Phi + \nabla\kappa \cdot \nabla\Phi] + c_0\kappa^{-2}|\nabla\kappa|^2\Phi.$$

Next, we state the Ritz variational formulation for the regular part v, on which DRM is based. Let $\tilde{h} = h - c_0\kappa^{-1}\Phi$ in the Dirichlet case, and let $\tilde{h} = h - c_0\kappa\partial_n(\kappa^{-1}\Phi)$ in the Neumann case be the modified boundary data. Then the Ritz variational formulation is posed on $H^1(\Omega)$ and given by

(3.5) $$L(v) = \begin{cases} \frac{1}{2}(\kappa\nabla v, \nabla v) - (F, v), & \text{Dirichlet}, \\ \frac{1}{2}(\kappa\nabla v, \nabla v) - (F, v) - (\tilde{h}, v)_{L^2(\partial\Omega)}, & \text{Neumann}. \end{cases}$$

Let T be the trace operator. Then we have the following well-known result [24].

PROPOSITION 3.1. *If $F \in (H^1(\Omega))'$ and $\tilde{h} \in H^{\frac{1}{2}}(\partial\Omega)$ and $\tilde{h} \in (H^{\frac{1}{2}}(\partial\Omega))'$ in the Dirichlet and Neumann cases, respectively, then the regular part v is a minimizer of the functional $L(v)$ defined in (3.5) over the set $\{v \in H^1(\Omega) : Tv = \tilde{h}\}$ and $H^1(\Omega)$ in the Dirichlet and Neumann cases, respectively.*

The regularity assumptions on F and $\tilde{h}$ will be analyzed in section 3.1.2 below. Note that in the Neumann case, a solution u exists only under suitable compatibility conditions and to achieve the uniqueness, one may specify the value of the solution u at a selected point $x^* \in \overline{\Omega}$, e.g., $u(x^*) = a$. Then it has to be supplied with the condition $v(x^*) = \tilde{a} \equiv a - \kappa(x^*)^{-1}c_0\Phi(x^*)$, which is then used as a constraint in minimizing the functional $L(v)$.

3.1.2. Well-posedness of the modified problem (3.3). Note that in the presence of a point source, the solution u contains the singularity of the fundamental solution Φ, which belongs only to $W^{1,p}(\Omega)$ for $p < \frac{d}{d-1}$ [31, Theorem 1.1, p. 3], and as the dimension d grows, the Sobolev regularity of Φ deteriorates. In particular, it does not have the $H^1(\Omega)$ regularity when $d \geq 2$, and thus the standard Ritz-type formulation fails to make sense. Next, we discuss the smoothness of the regular part v. This depends on the regularity of κ and g and that of the coefficient κ. Below we discuss the regularity of the modified source F and the regular part v for one single point source. The next result summarizes the regularity of F. The special case of κ being locally constant in the neighborhood of the singularities is common in practice, e.g., the standard Poisson equation, which is arguably the most widely studied case for elliptic PDEs with point sources [7, 59, 41]. The analysis also sheds interesting insights into regularity restriction and the impact of the dimension d.

PROPOSITION 3.2. *Fix $d \geq 2$. Let $\kappa \in W^{2,r}(\Omega)$ with $r \geq 2$, and let $g \in L^2(\Omega)$. Then there holds that*

$$
F \in L^s(\Omega), \quad with \quad
\begin{cases}
s < \dfrac{d}{d-1}, & r \geq d,\, d \geq 2, \\[2mm]
s < \min\left(\dfrac{dr}{(d-2)r+d}, \dfrac{dr}{(d-2)r+2(d-r)} \right) & r < d,\, d \geq 3.
\end{cases}
$$

Further, if κ is locally constant in the neighborhood of the singularity at $\mathbf{x}_0$, then $F \in L^2(\Omega)$.

Proof. The assertion follows from direct computation with the standard Sobolev embedding theorem [1, Theorem 4.12, p. 85]. It is known that $\Phi \in W^{1,p}(\Omega)$ for any $p < \frac{d}{d-1}$ [31, Theorem 1.1, p. 3]. Meanwhile, by the Sobolev embedding theorem,

$$
(3.6) \qquad W^{1,r}(\Omega) \hookrightarrow L^{r^*}(\Omega), \quad with \quad
\begin{cases}
r^* = \infty, & r > d, \\[1mm]
r^* < \infty, & r = d, \\[1mm]
r^* = \dfrac{dr}{d-r}, & r < d.
\end{cases}
$$

Using the embedding (3.6) and the regularity $\Phi \in W^{1,p}(\Omega)$ with $p < \frac{d}{d-1}$ [31, Theorem 1.1, p. 3], we have

$$
(3.7) \qquad \Phi \in L^{p^*}(\Omega), \quad with \quad
\begin{cases}
p^* < \infty, & d = 2, \\[1mm]
p^* < \dfrac{d}{d-2}, & d \geq 3.
\end{cases}
$$

In view of the expression (3.4) of F, it suffices to analyze the three terms $\Phi\Delta\kappa$, $\nabla\kappa \cdot \nabla\Phi$, and $|\nabla\kappa|^2\Phi$. By Hölder's inequality, we deduce

$$
\Phi\Delta\kappa \in L^s(\Omega), \quad with \quad
\begin{cases}
s < r, & d = 2, \\[1mm]
s < \dfrac{dr}{(d-2)r+d}, & d \geq 3.
\end{cases}
$$

Similarly, for the term $\nabla\kappa \cdot \nabla\Phi$ (noting the condition $r \geq 2$), we have

$$
\nabla\kappa \cdot \nabla\Phi \in L^s(\Omega), \quad with \quad
\begin{cases}
s < \dfrac{d}{d-1}, & r \geq d,\, d \geq 2, \\[2mm]
s < \dfrac{dr}{(d-2)r+d}, & r < d,\, d \geq 3.
\end{cases}
$$

Last, it follows from the embeddings (3.6) and (3.7) and the generalized Hölder inequality that

$$|\nabla \kappa|^2 \Phi \in L^q(\Omega), \quad \text{with} \quad \begin{cases} q < \infty, & d = 2, \\ q < \dfrac{d}{d-2}, & r \geq d, \, d \geq 3, \\ q < \dfrac{dr}{(d-2)r + 2(d-r)}, & r < d, \, d \geq 3. \end{cases}$$

Now combining the preceding three results yields the first assertion. The remaining assertion follows from direct computation. □

Based on the regularity result of the modified source F in Proposition 3.2, we can discuss the well-posedness of the Ritz variational formulation of problem (3.3), which lays the foundation for applying DRM. For a general variable coefficient κ, when $d = 2, 3$, the formulation is indeed well-posed. Then, by the standard elliptic regularity theory, we have $v \in H^1(\Omega)$, whereas the regularity of the solution u to problem (1.1) with a point source is inherently limited to $u \in W^{1,p}(\Omega)$ for any $p < \frac{d}{d-1}$. One can also deduce the precise Sobolev regularity of the regular part v: in the two-dimensional case, v belongs to $W^{2,s}(\Omega)$ for any $s < 2$, just falling short of $H^2(\Omega)$, whereas in the three-dimensional case, v can at most have a regularity $W^{2,\frac{3}{2}}(\Omega)$ (when $r = \infty$). This shows the benefit of singularity splitting for problem (1.1). When $d \geq 4$, it may be ill-defined, except the special case of κ being locally constant. In the latter case, v does belong to $H^2(\Omega)$ for any $d \geq 2$, and thus the regular part v indeed enjoys much better regularity than the solution u.

COROLLARY 3.3. *Let $\kappa \in W^{2,r}(\Omega)$, with $r \geq d$, $h \in H^{\frac{1}{2}}(\partial\Omega)$, and $g \in L^2(\Omega)$. Then problem (3.3) has a weak solution $v \in H^1(\Omega)$ for $d = 2, 3$. Further, if κ is locally constant in a neighborhood of $\mathbf{x}_0$, then it has a weak solution $v \in H^1(\Omega)$ for any $d \geq 2$.*

Proof. First, since the singularity point $\mathbf{x}_0 \in \Omega$, we have $\kappa^{-1}\phi \in H^{\frac{1}{2}}(\partial\Omega)$, and hence $\tilde{h} \in H^{\frac{1}{2}}(\partial\Omega)$. By Sobolev embedding (3.6), we have

$$H^1(\Omega) \hookrightarrow L^s(\Omega), \quad \text{with} \quad \begin{cases} s < \infty, & d = 2, \\ s = \dfrac{2d}{d-2}, & d \geq 3. \end{cases}$$

Therefore, by duality, we have

$$L^{s^*}(\Omega) \hookrightarrow (H^1(\Omega))', \quad \text{with} \quad \begin{cases} s^* > 1, & d = 2, \\ s^* = \dfrac{2d}{d+2}, & d \geq 3. \end{cases}$$

Then, for $d = 2, r \geq 2$, clearly $F \in L^s(\Omega)$ for any $s < 2$. Similarly, for $d = 3, r > 2$, we have $F \in L^s(\Omega)$, with $s < \frac{3r}{r+3}$ and $\frac{3r}{r+3} > \frac{2d}{d+2}$. Meanwhile, for $d = 4, r > 4$, we have $F \in L^s(\Omega)$ for any $s < \frac{4}{3} = \frac{2d}{d+2}$. Thus, for sufficiently smooth g, we have $F \in (H^1(\Omega))'$ for $d = 2, 3$ only. Meanwhile, if κ is locally constant in a neighborhood of the singularity at $\mathbf{x}_0$, then $F \in L^2(\Omega) \hookrightarrow (H^1(\Omega))'$ for any $d \geq 2$. □

3.1.3. Singularity splitting for line sources. Next, we discuss the case of line sources. We describe the procedure for a line source supported on a line segment in $\mathbb{R}^d$, $d \geq 3$. The construction below follows the procedure in [28], which also contains a detailed construction of the singularity Φ_L in $\mathbb{R}^3$. Nonetheless, to the best of our knowledge, the resulting expressions appear to be new, since the numerical approximation of high-dimensional PDEs has not received the due attention. Specifically, we

parameterize the line segment $\Lambda \subset \Omega$ connecting two points $\mathbf{a}, \mathbf{b} \in \Omega$ by $\mathbf{y} = \mathbf{a} + t\boldsymbol{\tau}$ for $t \in (0, L)$, with $L = |\mathbf{b} - \mathbf{a}|$ and $\boldsymbol{\tau} = (\mathbf{b} - \mathbf{a})/L$ the unit tangent vector of Λ. Consider the elliptic problem (1.1) with δ being replaced with δ_Λ. Then we seek a function Φ_L so that

$$-\int_\Omega \Delta \Phi_L v \, d\mathbf{x} = \int_\Lambda v \, ds \quad \forall v \in C(\overline{\Omega}).$$

A natural candidate for Φ_L is the convolution of δ_Λ and Φ, i.e.,

$$\Phi_L(\mathbf{x}) = \int_\Omega \delta_\Lambda(\mathbf{y}) \Phi(\mathbf{x} - \mathbf{y}) \, d\mathbf{y} = \int_0^L \frac{1}{d(d-2)\alpha(d)} |\mathbf{x} - (\mathbf{a} + \boldsymbol{\tau}t)|^{-(d-2)} \, dt.$$

So it suffices to evaluate the integral $I := \int_0^L |\mathbf{x} - (\mathbf{a} + \boldsymbol{\tau}t)|^{2-d} dt$. Note that in $\mathbb{R}^3$, with $r_a = |\mathbf{x} - \mathbf{a}|$ and $r_b = |\mathbf{x} - \mathbf{b}|$, we can compute directly [28, p. 1722]

$$I = \ln \frac{r_b + L + \boldsymbol{\tau} \cdot (\mathbf{a} - \mathbf{x})}{r_a + \boldsymbol{\tau} \cdot (\mathbf{a} - \mathbf{x})}.$$

For $d \geq 4$, by first setting $\alpha^2 = r_a^2 - ((\mathbf{a} - \mathbf{x}) \cdot \boldsymbol{\tau})^2$, and then substituting $t = \alpha \tan s$, we have

$$I = \int_0^L [t^2 + 2(\mathbf{a} - \mathbf{x}) \cdot \boldsymbol{\tau}t + r_a^2]^{1 - \frac{d}{2}} dt = \int_0^L (t + (\mathbf{a} - \mathbf{x}) \cdot \tau)^2 - ((\mathbf{a} - \mathbf{x}) \cdot \tau)^2 + r_a^2 dt$$

$$= \int_{(\mathbf{a} - \mathbf{x}) \cdot \boldsymbol{\tau}}^{(\mathbf{a} - \mathbf{x}) \cdot \boldsymbol{\tau} + L} (t^2 + \alpha^2)^{1 - \frac{d}{2}} dt = \frac{1}{\alpha^{d-3}} \int_{\arctan \frac{(\mathbf{a} - \mathbf{x}) \cdot \boldsymbol{\tau}}{\alpha}}^{\arctan \frac{(\mathbf{a} - \mathbf{x}) \cdot \boldsymbol{\tau} + L}{\alpha}} \cos^{d-4} s \, ds.$$

Then using the integral identities [29, 2.513, p. 153]

$$\int \cos^{2n} x \, dx = \frac{1}{2^{2n}} \binom{2n}{n} x + \frac{1}{2^{2n-1}} \sum_{k=0}^{n-1} \binom{2n}{k} \frac{\sin(2n - 2k)x}{2n - 2k},$$

$$\int \cos^{2n+1} x \, dx = \frac{1}{2^{2n}} \sum_{k=0}^{n} \binom{2n+1}{k} \frac{\sin(2n - 2k + 1)x}{2n - 2k + 1},$$

we obtain

$$I = \frac{1}{\alpha^{d-3}} \begin{cases} \arctan \dfrac{(\mathbf{a} - \mathbf{x}) \cdot \boldsymbol{\tau} + L}{\alpha} - \arctan \dfrac{(\mathbf{a} - \mathbf{x}) \cdot \boldsymbol{\tau}}{\alpha}, \quad d = 4, \\[2ex] \dfrac{1}{2^{d-5}} \sum_{i=0}^{m-3} \binom{d-4}{i} \\ \quad \cdot \dfrac{\sin[(d - 4 - 2i) \arctan \frac{(\mathbf{a} - \mathbf{x}) \cdot \boldsymbol{\tau} + L}{\alpha}] - \sin[(d - 4 - 2i) \arctan \frac{(\mathbf{a} - \mathbf{x}) \cdot \boldsymbol{\tau}}{\alpha}]}{d - 4 - 2i} \\[2ex] \quad + \dfrac{1}{2^{d-4}} \binom{d-4}{m-2} \left[\arctan \dfrac{(\mathbf{a} - \mathbf{x}) \cdot \boldsymbol{\tau} + L}{\alpha} - \arctan \dfrac{(\mathbf{a} - \mathbf{x}) \cdot \boldsymbol{\tau}}{\alpha} \right], \\ d = 2m \geq 6, \\[2ex] \dfrac{1}{2^{d-5}} \sum_{i=1}^{m-2} \binom{d-4}{i} \\ \quad \cdot \dfrac{\sin[(d - 4 - 2i) \arctan \frac{(\mathbf{a} - \mathbf{x}) \cdot \boldsymbol{\tau} + L}{\alpha}] - \sin[(d - 4 - 2i) \arctan \frac{(\mathbf{a} - \mathbf{x}) \cdot \boldsymbol{\tau}}{\alpha}]}{d - 4 - 2i}, \\ d = 2m + 1 \geq 5. \end{cases}$$

Combining these identities yields an explicit expression for $\Phi_L(\mathbf{x})$. Using the function $\Phi_L(\mathbf{x})$, we can split the solution u to problem (1.1) into

$$(3.8) \qquad u = \kappa^{-1} f \Phi_L + v.$$

In the splitting (3.8), the term $\kappa^{-1} f \Phi_L$ captures the leading singularity of the solution u near the line sources and v is the regular part. Since the fundamental solution $\Phi_L(\mathbf{x}) \neq 0$ when $\mathbf{x} \in \partial\Omega$, the boundary condition and the source term have to be modified accordingly. For example, for the Dirichlet problem, by substituting the splitting (3.8) into (1.1) we can get the following governing equation for the regular part v:

$$(3.9) \qquad \begin{cases} -\nabla \cdot (\kappa \nabla v) = F & \text{in } \Omega, \\ \qquad\qquad v = h - \kappa^{-1} f \Phi_L & \text{on } \partial\Omega, \end{cases}$$

with the modified source F given by

(3.10)
$$F = g + \Delta f \Phi_L + 2\nabla f \cdot \nabla \Phi_L - \kappa^{-1}[\Delta \kappa f \Phi_L + f \nabla \kappa \cdot \nabla \Phi_L + \Phi_L \nabla \kappa \cdot \nabla f] + \kappa^{-2}|\nabla \kappa|^2 f \Phi.$$

Note that the regularity of the regular part v to problem (3.9) can also be analyzed similarly as in section 3.1.2, under suitable regularity assumption on the density f. Indeed, line sources can be obtained by taking the Laplacian of a lower-dimensional fundamental solution in the high-dimensional ambient space. Similar to Corollary 3.3, generally there is still a restriction on the dimension d (i.e., $d = 2, 3, 4$ only) in the presence of line sources, in order for the variational problem to be well-posed, but again the restriction disappears when f and κ are locally constant in the neighborhood of singularity support, i.e., the line segment. See also the work [28] for some insightful discussions on weighted Sobolev regularity in the three-dimensional case. We leave a precise regularity analysis, including in weighted Sobolev spaces, to a future work. Using the convolution strategy, one can also derive the expression for more general singularities, e.g., singularities on squares / cubes in high-dimensional space. Due to the complexity of the resulting expressions, we do not further pursue the issue below.

3.2. Deep Ritz method. Now we describe the deep Ritz method (DRM) [71] for approximating the regular part v and the details about the practical implementation. First we discuss the Dirichlet case, and will comment on the Neumann case at the end, which only requires minor changes. The energy functional of problem (3.3) requires minimizing the functional $L(v)$ in (3.5) over the set $\{v \in H^1(\Omega) : Tv = \tilde{h}\}$. However, due to the global nature of NNs, it is highly nontrivial to construct an NN function that satisfies the Dirichlet boundary condition $Tv = \tilde{h}$ exactly, although theoretically this is highly desirable for improving the error estimates of the method [52]. In order to enforce the Dirichlet boundary condition $Tv = \tilde{h}$, we adopt the standard penalty formulation

$$(3.11) \qquad L_\sigma(v) = \tfrac{1}{2}(\kappa \nabla v, \nabla v) - (F, v) + \tfrac{\sigma}{2}\|v - \tilde{h}\|_{L^2(\partial\Omega)}^2, \quad v \in H^1(\Omega),$$

where $\sigma > 0$ is the penalty parameter. This choice is standard in the current practice of NN- based PDE solvers [55, 71]. Intuitively, as the penalty parameter σ increases to infinity, the equality constraint representing the Dirichlet boundary condition is better satisfied; see Theorem 5.3 and the remark thereafter for precise statements.

Now we describe NNs for approximating the regular part v, which is the central component of DRM. We employ standard fully connected feed forward NNs, i.e.,

functions $f_\theta : \mathbb{R}^d \to \mathbb{R}$, with the network parameter θ. Specifically, it is defined recursively by

$$\begin{aligned} \mathbf{x}_0 &= \mathbf{x}, \\ \mathbf{x}_\ell &= \rho(\mathbf{A}_\ell \mathbf{x}_{\ell-1} + \mathbf{b}_\ell), \quad \ell = 1, 2, \dots, L-1, \\ f_\theta(\mathbf{x}) &= \mathbf{A}_L \mathbf{x}_{L-1} + \mathbf{b}_L, \end{aligned}$$

where $\mathbf{A}_\ell \in \mathbb{R}^{n_\ell \times n_{\ell-1}}, \mathbf{b}_\ell \in \mathbb{R}^{n_\ell}, \ell = 1, 2, \dots, L$. Clearly, we fix $n_0 = d$ and $n_L = 1$. In the construction, the nonlinear function $\rho : \mathbb{R} \to \mathbb{R}$ is called the activation function, and it is applied componentwise to a vector. L is called the depth, and $W := \max\{n_\ell, \ell = 0, 1, \dots, L\}$ is called the width of the network. The set of parameters of the NN, i.e., $\mathbf{A}_\ell, \mathbf{b}_\ell$ ($\ell = 1, 2, \dots, L$), are trainable and are often stacked into a large vector θ. The training is carried out by minimizing a suitable loss function, which will be specified below. In the construction, $\mathbf{x}_0$ is called the input layer, $\mathbf{x}_\ell$, $\ell = 1, 2, \dots, L-1$, are called the hidden layer, and $f_\theta(\mathbf{x})$ is the output layer, where the subscript θ explicitly indicates the dependence of the NN on the parameter vector θ. The total number of parameters in the NN f_θ is $\sum_{\ell=1}^{L} n_\ell(n_{\ell-1} + 1)$.

For the activation function ρ, there are many possible choices. The most frequently used one in computer vision is the rectified linear unit (ReLU), defined by $\rho(x) = \max(x, 0)$ [44]. However, it is not smooth enough for the use in (SS)DRM, since the formulation requires twice differentiability of the activation function ρ: one spatial derivative, and the optimizer requires another derivative in the NN parameter θ. In the context of neural PDE solvers, hyperbolic tangent $\rho(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$ and logistic $\rho(x) = \frac{1}{1 + e^{-x}}$ are frequently used [55, 14]. In the numerical experiments, we employ the hyperbolic tangent as the activation function. Note that under fairly generous conditions on ρ, the set of NN functions can approximate any $L^p(\Omega)$ function arbitrarily well, as the width W and/ or depth L tend to infinity, a property commonly known as universal approximation property. Furthermore, quantitative estimates on the approximation error in various norms have also been derived in recent years [70, 32]. Below we denote the collection of NN functions of depth L, with the total number of nonzero parameters N_θ, and each parameter being bound by R, with the activation function ρ by $\mathcal{N}_\rho(L, N_\theta, R)$, i.e.,

$$(3.12) \qquad \mathcal{N}_\rho(L, N_\theta, R) = \{v_\theta : v_\theta \text{ has a depth } L, |\theta|_{\ell^0} \le N_\theta, |\theta|_{\ell^\infty} \le R\},$$

where $|\cdot|_{\ell^0}$ and $|\cdot|_{\ell^\infty}$ denote the number of nonzero entries in and the maximum norm of a vector, respectively. Below we also use the shorthand notation $\mathcal{A}$ to denote this collection of functions (with a fixed architecture).

In a neural PDE solver, we use an element v_θ from the set $\mathcal{A}$ to approximate the minimizer $v_\sigma \in H^1(\Omega)$ of the functional $L_\sigma(v)$, i.e.,

$$\theta^* = \arg\min_\theta L_\sigma(v_\theta),$$

and then set $v_{\theta^*} \in \mathcal{A}$ as an approximation to v_σ. Since the parameter θ is finite-dimensional and lives on a compact set (due to the ℓ^∞ bound $|\theta|_{\ell^\infty} \le R$), for a smooth activation function ρ, the loss $L_\sigma(v_\theta)$ is continuous in θ, which directly implies the existence of a global minimizer θ^*. Note that the loss $L_\sigma(v_\theta)$ involves potentially high dimensional integrals (i.e., in Ω and on $\partial\Omega$), and thus direct computation is likely still intractable. Instead, in practical computation, we approximate the integrals using quadrature, often by means of Monte Carlo, especially in the high-dimensional case. Specifically, we rewrite the loss L_σ as

$$(3.13) \qquad L_\sigma(v_\theta) = |\Omega|\mathbb{E}_{X \sim U(\Omega)}\left[\frac{\kappa(X)|\nabla v_\theta(X)|^2}{2} - F(X)v_\theta(X)\right]$$
$$+ \frac{\sigma|\partial\Omega|}{2}\mathbb{E}_{Y \sim U(\partial\Omega)}\left[|v_\theta(Y) - \tilde{h}(Y)|^2\right],$$

where $U(\Omega)$ and $U(\partial\Omega)$ denote the uniform distributions on the domain Ω and the boundary $\partial\Omega$, respectively, and $\mathbb{E}_{U(\Omega)}$ denotes taking expectation with respect to the uniform distribution $U(\Omega)$. This consideration leads to the following empirical loss:

$$(3.14)$$
$$\widehat{L}_\sigma(v_\theta) = \frac{|\Omega|}{N_r}\sum_{i=1}^{N_r}\left[\frac{\kappa(X_i)|\nabla v_\theta(X_i)|^2}{2} - F(X_i)v_\theta(X_i)\right] + \frac{|\partial\Omega|}{N_b}\frac{\sigma}{2}\sum_{j=1}^{N_b}(v_\theta(Y_j) - \tilde{h}(Y_j))^2,$$

where the sampling points $\{X_i\}_{i=1}^{N_r}$ and $\{Y_j\}_{j=1}^{N_b}$ are identically and independently distributed (i.i.d.) random variables, distributed uniformly on the domain Ω and the boundary $\partial\Omega$, respectively, i.e., $\{X_i\}_{i=1}^{N_r} \sim U(\Omega)$ and $\{Y_j\}_{j=1}^{N_b} \sim U(\partial\Omega)$. Note that the distributions are taken to be uniform according to the continuous loss (3.13). In practice, it is also possible to use nonuniform distributions, which corresponds to solving the PDEs in suitable weighted spaces and can be beneficial for certain problems; see [69, 65] for numerical studies along this direction. Note that the resulting optimization problem in θ reads as

$$(3.15) \qquad\qquad \widehat{\theta}^* = \arg\min_\theta \widehat{L}_\sigma(v_\theta),$$

with $v_{\widehat{\theta}^*} \in \mathcal{A}$ being the NN approximation. The continuous loss $L_\sigma(v_\theta)$ and empirical loss $\widehat{L}_\sigma(v_\theta)$ have different minimizers, due to the introduction of additional quadrature errors. We shall discuss this issue briefly in section 5.

The final optimization problem (3.15) is highly nonconvex, due to the nonlinearity of the NN v_θ in the NN parameter θ. Therefore, it is very challenging to find a global minimizer of the loss $\widehat{L}_\sigma(v_\theta)$. Nonetheless, in practice, simple optimization algorithms, e.g., stochastic gradient descent and its variants (e.g., ADAM and AdaGrad) [8], seem to work fairly well, as testified by the great empirical successes [14]. In the numerical experiments, we will employ the limited memory BFGS algorithm [9] to minimize the empirical loss $\widehat{L}_\sigma(v_\theta)$, which has been a very popular choice for NN-based PDE solvers [55, 14].

In the Neumann case, the empirical loss $\widehat{L}_\sigma(v_\theta)$ takes a similar form:

$$\widehat{L}_\sigma(v_\theta) = \frac{|\Omega|}{N_r}\sum_{i=1}^{N_r}\left[\frac{1}{2}\kappa(X_i)|\nabla v_\theta(X_i)|^2 - F(X_i)v_\theta(X_i)\right] - \frac{|\partial\Omega|}{N_b}\sum_{j=1}^{N_b}\tilde{h}(Y_j)v_\theta(Y_j)$$
$$+ \frac{\sigma}{2}|v_\theta(\mathbf{x}^*) - \tilde{a}|^2,$$

where the last term is to approximately enforce the point value evaluation $v(\mathbf{x}^*) = \tilde{a}$ (with $\mathbf{x}^* \in \overline{\Omega}$).

In practice, the choice of the penalty parameter σ in the objective is important, since it balances several competing effects: the larger is the value of σ, the smaller is the error due to penalization, but the larger is the approximation error on the boundary, as indicated by Theorem 5.6. Thus, a good tradeoff between these different sources of errors is needed in order to deliver high-quality NN approximations to the solution u of the PDE (1.1). We have followed a simple path-following procedure, a well-established procedure in optimization [3, 36]: we start with a small value σ_1, and

---

Algorithm 3.1 The penalty method for SSDRM.

Set $\sigma_1 > 0$, increasing factor $\eta > 1$
while Stopping condition not met **do**
  Minimize the loss function $\widehat{L}_{\sigma_k}(v_\theta)$, initialized to $\widehat{\theta}^*_{k-1}$, and find the optimal $\widehat{\theta}_k$
  $\sigma_{k+1} \leftarrow \eta\sigma_k$, and $k \leftarrow k+1$

---

then after each loop we update σ geometrically: $\sigma_{k+1} = \eta\sigma_k$ for some fixed $\eta > 1$. By updating σ_k, the minimizer $\widehat{\theta}^*_k$ of the loss $\widehat{L}_{\sigma_k}(v_\theta)$ also approaches the solution of problem (3.3), and the path-following strategy allows enforcing this progressively. The overall procedure is shown in Algorithm 3.1. The use of the path-following strategy requires the training for multiple σ values, which can potentially be time consuming if done naively. Fortunately, the parameter θ of the σ_{k+1}-problem (i.e., minimizing $\widehat{L}_{\sigma_{k+1}}(v_\theta)$) can be initialized to the converged parameter $\widehat{\theta}^*_k$ of the σ_k-problem in order to warm start the optimization process. This ensures that for each fixed σ_k (except σ_1), the initial parameter configuration is close to the optimal one, so the training loop only requires relatively few iterations to reach convergence. This is also confirmed by the numerical experiments below.

4. Numerical experiments and discussions. In this section, we present several numerical experiments to illustrate the performance of the proposed SSDRM. Although it works for all three types of boundary conditions, we present results mostly for the Dirichlet problem, and we use f to denote the singularity strength for point sources/density for line sources (and thus omit c_i for singularity strength) and $\mathbf{x} = (x_1, \ldots, x_d) \in \mathbb{R}^d$. In the training, $N_r = 10,000$ points in the domain Ω and $N_b = 400$ points on the boundary $\partial\Omega$ are randomly selected to form the empirical loss $\widehat{L}_\sigma(v_\theta)$, unless otherwise specified. In the path-following strategy for determining the penalty parameter σ, we take the initial penalty parameter $\sigma_1 = 20$ and an increasing factor $\eta = 1.5$. All the numerical experiments were carried out on a personal laptop (operating system: Windows 10, with RAM 8.0 GB, Intel Core i7-10510U CPU, 2.3 GHz), with Python 3.9.7, with the popular software framework PyTorch. The gradient of the NN output with respect to the input $\mathbf{x}$ (i.e., spatial derivatives) and that of the empirical loss with respect to the NN parameter vector θ are computed via automatic differentiation, as implemented by `torch.autograd`. The empirical loss $\widehat{L}_\sigma(v_\theta)$ is minimized using the off-the-shelf optimizer limited memory BFGS [9], as implemented in the SciPy library, with the default setting (tolerance = 1.0e-9, no box constraint) and a maximum of 2500 iterations. To measure the accuracy of an approximation $\hat{v}$ of the regular part v^*, we use the standard relative $L^2(\Omega)$-error e defined by $e = \|v^* - \hat{v}\|_{L^2(\Omega)}/\|v^*\|_{L^2(\Omega)}$. The stopping condition is set to $e < 1.00\text{e-}3$ and $\sigma_k \leq 1200$. The first condition ensures that the approximation can achieve the desired accuracy, and the second one is to terminate the iteration after a finite number of loops. The PyTorch source code for reproducing all the numerical experiments will be made available at the github link https://github.com/hhjc-web/SSDRM.git.

First, we showcase the approach on the case with one point source.

Example 4.1. The domain Ω is $\Omega = (-1,1)^2$, diffusion coefficient $\kappa = |\mathbf{x}|^2 + 1$, a point source at $\mathbf{x}_0 = (0,0)$ with strength $f = 2(|\mathbf{x}|^2 + 1)$, source $g = (|\mathbf{x}|^2 + 1)\left(2x_2 \sin x_1 e^{x_1 x_2} + (1 - x_1^2 + x_2^2)e^{x_1 x_2} \cos x_1\right) + \frac{2}{\pi} - 4x_1 x_2 \cos x_1 e^{x_1 x_2} + 2x_1 \sin x_1 e^{x_1 x_2}$, and the Dirichlet boundary condition $h = -\frac{1}{\pi} \ln |\mathbf{x}| + e^{x_1 x_2} \cos x_1$. The analytic solution u is given by $u = -\frac{1}{\pi} \ln |\mathbf{x}| + e^{x_1 x_2} \cos x_1$.
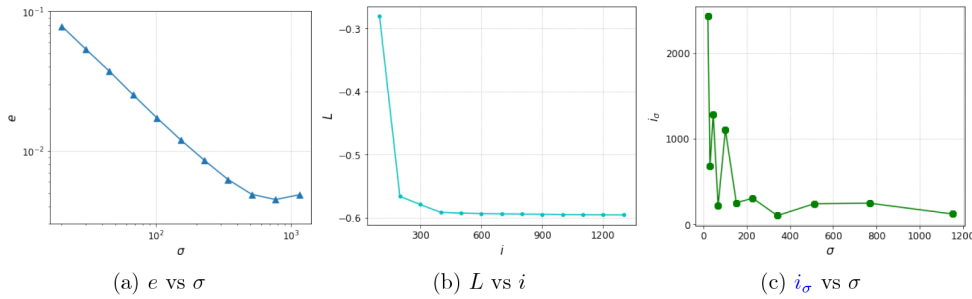
(a) e vs σ (b) L vs i (c) i_σ vs σ

FIG. 1. *The dynamics of the training process:* (a) *the error e versus the penalty parameter σ,* (b) *the decay of the loss function L over iteration index i with $\sigma = 1153.3$, and* (c) *the required iteration number i_σ for each σ_k-problem during the path-following strategy.*

In the splitting, the regular part v is given by $v = \mathrm{e}^{x_1 x_2} \cos x_1$ and has to be learned using NNs. The used NN has three hidden layers, and each layer has 20 neurons. The second condition is satisfied for $\sigma_{11} = 1153.3$. First we examine the convergence of the path-following strategy. We report the variation of the error e with respect to the penalty parameter σ in Figure 1(a). It is observed that as the parameter σ gets larger, the error e drops rapidly during the first few loops (i.e., $\sigma < 400$), after which the decay becomes much slower. The decay plot roughly obeys the exponential law, which is consistent with the theoretical analysis in section 5. Indeed, Theorem 5.6 indicates that as the value of σ increases, the (mean squared) error due to penalization decreases to zero at a rate $O(\sigma^{-2})$, but the approximation error and the statistical error increase linearly with σ, which is also clearly observed in Figure 1. Thus, there is an optimal σ which minimizes the total error. This plot shows also that one can terminate the iteration earlier without compromising much the accuracy of the NN approximation. The empirical loss $\widehat{L}_\sigma$ decreases throughout the path-following procedure (cf. Figure 1(b)), exhibiting a steady convergence behavior of the optimizer. Note that the empirical loss for (SS)DRM is not necessarily positive (but it is bounded below by a constant). Further, the number of iterations required for each σ_k-problem ($k \geq 2$) is indeed much smaller than that for the σ_1-problem, confirming the effectiveness of the path-following strategy for warm starting the optimization for σ_k-problems.

The numerical approximations by SSDRM and three existing NN-based PDE solvers, i.e., SAPINN, DRM, and WAN, are shown in Figure 2, where the employed NN networks, the number of sampling points taken in the domain Ω and on the boundary $\partial\Omega$, and the optimizer are identical for all the methods in order to ensure a fair comparison. The pointwise error of the SSDRM approximation is very small, and the accuracy around the singularity at the origin is excellent, indicating a highly accurate approximation. The results show that a direct application of DRM fails to yield satisfactory results: the error is very large everywhere. This is not surprising, since the corresponding continuous loss function $L_\sigma(v)$ is actually ill-defined in the space $H^1(\Omega)$, as pointed out in section 2. Similar to DRM, WAN also does not perform well, since it suffers from exactly the same issue. In contrast, SAPINN can indeed yield much more accurate approximations than DRM and WAN, but the error around the singularity dominates the overall error. Intuitively, this concurs with the worse approximation property of NNs to singular functions [18]. Table 1 shows the quantitative comparison of these methods for Example 4.1 in terms of computing time
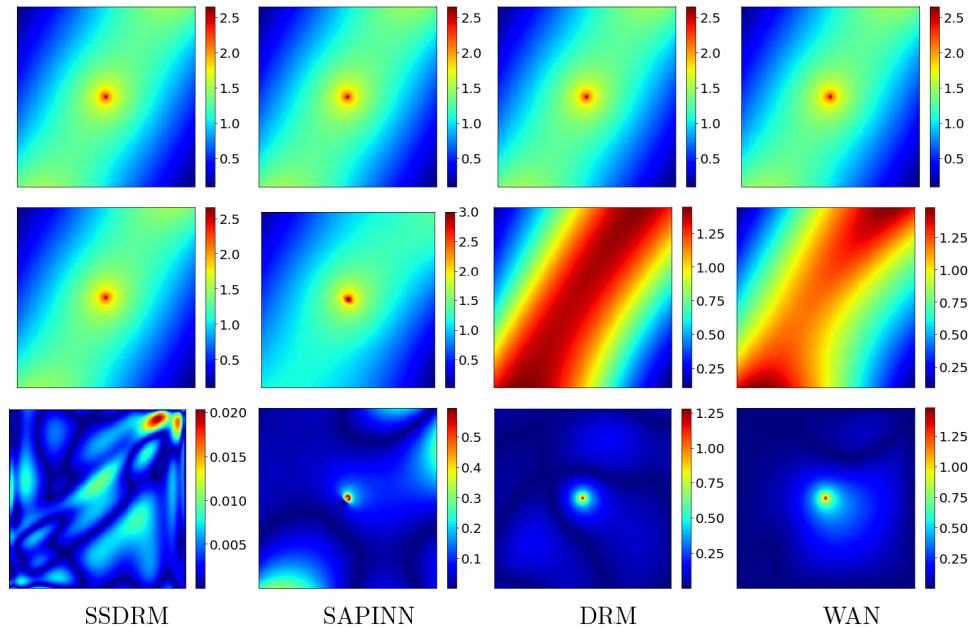
FIG. 2. *The numerical approximations of Example* 4.1 *by the proposed SSDRM, SAPINN, DRM, and WAN. From top to bottom: analytic solution, neural network approximation, and absolute error.*

TABLE 1
The comparison between DRM, WAN, SAPINN, and the proposed SSDRM for Example 4.1.

Method	Time (in secs)	e
SSDRM	252	4.85e-3
SAPINN	3433	7.82e-2
DRM	3	1.02e-1
WAN	32	1.38e-1

(in seconds) and error e. Note that the computing time should be interpreted indicatively only due to its sensitivity with respect to the setting and implementation, etc. The results fully confirm the qualitative results in Figure 2: DRM and WAN perform very poorly on the example, and SAPINN does yield an acceptable approximation to the true solution u, but it is less accurate and computationally more expensive than SSDRM. It is worth mentioning that the accuracy of neural PDE solvers tends to stagnate at a level of $10^{-2} \sim 10^{-3}$, as for both SSDRM and SAPINN, but not much lower. This behavior has been widely observed for neural PDE solvers [55, 71, 72, 14], and it differs markedly from more traditional solvers, e.g., the finite element method, for which one can make the error arbitrarily small by taking a sufficiently refined mesh. Also the computing time is relatively long, given the two-dimensional nature of the problem, indicating the imperative need for an accelerated training process, e.g., pretraining or meta-learning, in order for the method to be competitive with more traditional methods in the low-dimensional setting. We refer interested readers to [54, 58] for detailed discussions on the pros and cons of neural PDE solvers and more classical approaches.
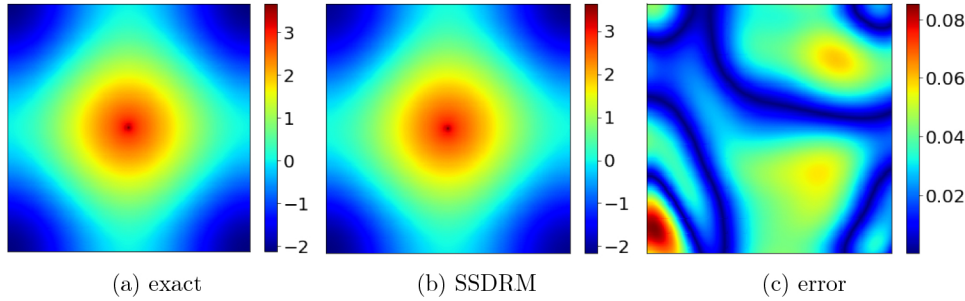
| (a) exact | (b) SSDRM | (c) error |

FIG. 3. *The numerical approximations of Example* 4.2, *with an error* $e = 3.18e - 2$.

The next example is about the Poisson problem with a Neumann boundary condition.

Example 4.2. The domain Ω is the square $\Omega = (-1, 1)^2$, diffusion coefficient $\kappa \equiv 1$, a point source at $\mathbf{x}_0 = (0, 0)$ with strength $f \equiv 1$, source $g = \pi^2(\cos(\pi x_1) + \cos(\pi x_2))$, and the Neumann boundary condition $h = \frac{\partial \Phi}{\partial n}$. The analytic solution u is given by $u = \frac{1}{2\pi} \ln |\mathbf{x}| + \cos(\pi x_1) + \cos(\pi x_2)$.

In the splitting, the regular part $v(\mathbf{x}) = \cos(\pi x_1) + \cos(\pi x_2)$ solves the standard Poisson problem with a source $F(\mathbf{x}) = \pi^2(\cos(\pi x_1) + \cos(\pi x_2))$ and a zero Neumann boundary condition. The value of the solution v at a point $\mathbf{x}^*$ on the boundary $\partial\Omega$ is specified to ensure the uniqueness of the exact solution. We employ an NN that has three layers and each layer has eight neurons. The obtained NN approximation after about 2500 iterations is shown in Figure 3 with an error $e = 3.18$e-2. The observations from Example 4.1 are fully confirmed.

Now we give a three-dimensional problem with a single point source.

Example 4.3. The domain Ω is the cube $\Omega = (-1, 1)^3$, diffusion coefficient $\kappa = |\mathbf{x}|^2 + 1$, a point source at the origin $\mathbf{x}_0 = (0, 0, 0)$ with strength $f = |\mathbf{x}|^2 + 1$, source $g = (|\mathbf{x}|^2 + 1)(|x_1^2 + x_2^2 + 1) \sin(x_1 x_2 + x_3) - (4x_1 x_2 + 2x_3)\cos(x_1 x_2 + x_3) + \frac{1}{2\pi|\mathbf{x}|}$, and the Dirichlet boundary condition $h = \frac{1}{4\pi|\mathbf{x}|} + \sin(x_1 x_2 + x_3)$. The exact solution u is given by $u = \frac{1}{4\pi|\mathbf{x}|} + \sin(x_1 x_2 + x_3)$.

In the splitting, the regular part v is given by $v = \sin(x_1 x_2 + x_3)$. The used NN v_θ has three layers and each layer has five neurons, and the error e of the SSDRM approximation is $e = 8.14$e-3. The numerical approximation is shown in Figure 4, with the two slices at $x_3 = -1$ and $x_3 = 0$. The former illustrates the accuracy of boundary fitting, whereas the latter shows that the approximation is of high quality near the singularity. The plots indicate that the pointwise error near the boundary $\partial\Omega$ tends to be slightly larger than that in the interior of the domain Ω.

The variation of the error e of the NN approximation as the penalty parameter σ increases is shown in Figure 5. In fact, as the penalty parameter σ increases, the statistical and approximation errors of the boundary term increase, whereas the error due to the penalization decreases. Thus, it is necessary to find a suitable value for σ so as to achieve the optimal error of the NN approximation to the PDE (1.1). Figure 5 indicates that one may choose $\sigma = 1200$ to realize a good balance between difference sources of errors. Throughout, the value of the empirical loss $\widehat{L}_\sigma$ decays steadily to a limit. Also, we have tested different numbers of sampling points and different architectures (layers, width) of the NN. The numerical results (i.e., the error
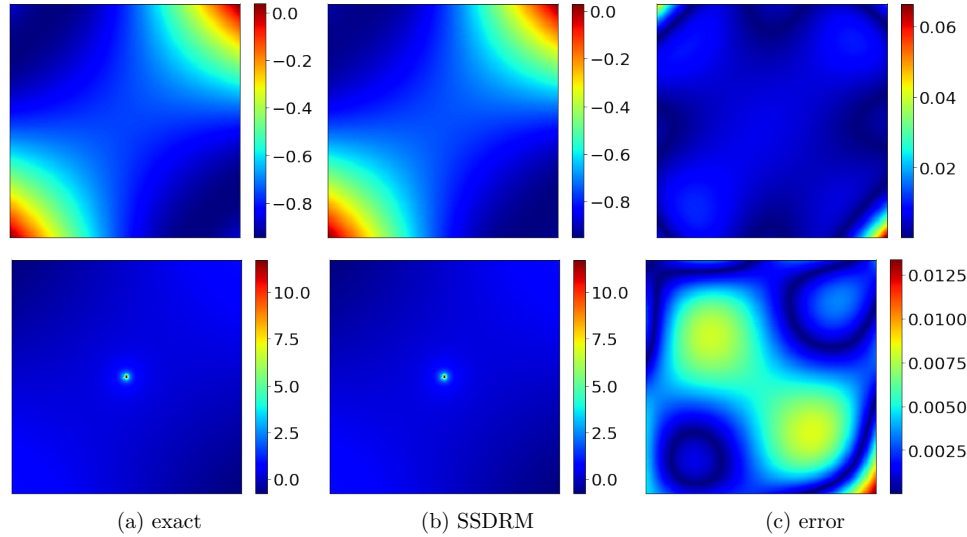
(a) exact   (b) SSDRM   (c) error

FIG. 4. *The neural network approximation for Example 4.3, with two slices at $x_3 = -1$ (top) and $x_3 = 0$ (bottom).*

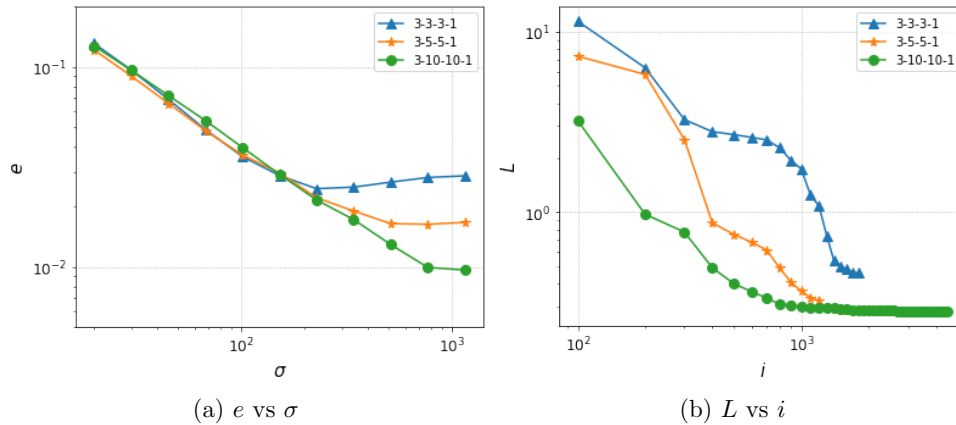

(a) e vs σ (b) L vs i

FIG. 5. *The convergence of the path-following strategy for Example 4.3, with three different NNs: (a) the error e versus σ and (b) the empirical loss $\widehat{L}_\sigma$ versus the iteration index i for $\sigma = 1153.3$.*

e and the training time in seconds) are shown in Table 2. The results indicate that in order to achieve the best approximation with an optimal complexity, one needs a suitable balance between these different parameters. Increasing the size of the NN or the number of sampling points alone does not necessarily lead to a smaller error for the approximation. This is also observed earlier in [38]. Thus, in the experiments, we have chosen the NN of the setting 3-5-5-1 (with $N_b = 600$ and $N_r = 10000$) to examine the training dynamics of the optimization method. Figure 5 also indicates one peculiar but very common behavior of PDE solvers based on NNs: the accuracy of the NN approximation does not exhibit a steady decrease of the error as the NN width and depth increase, unlike the more traditional methods, e.g., finite element approximation, for which the error does decrease steadily to zero as the mesh size tends to zero. In sharp contrast, it appears that none of the existing neural PDE

TABLE 2
The variation of the training time (in seconds) and error with respect to sampling points in the domain N_r and on the boundary N_b.

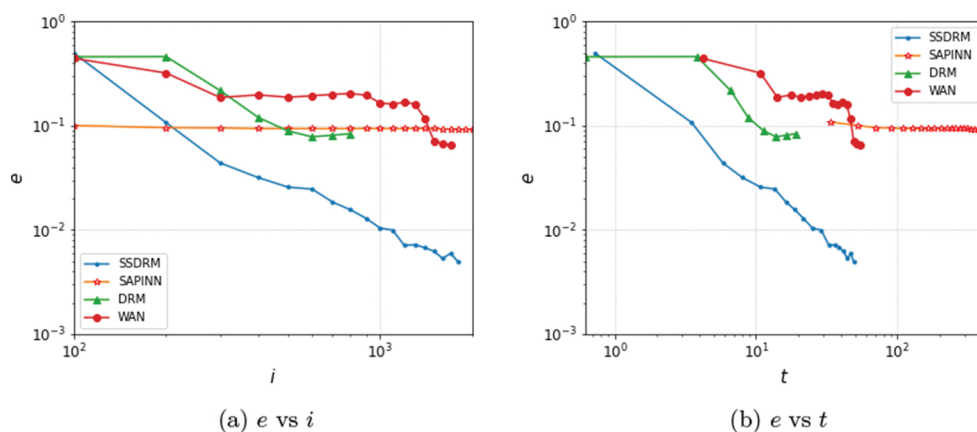| N_b	N_r	Layer	Neurons	Time	e
400	10000	3	3	45	2.86e-2
400	10000	3	5	80	1.70e-2
400	10000	3	10	130	9.66e-3
400	10000	5	3	51	2.25e-2
400	10000	5	5	104	8.63e-3
400	10000	5	10	327	1.79e-2
600	10000	3	3	47	3.45e-2
600	10000	3	5	83	8.14e-3
600	10000	3	10	132	9.60e-3
400	15000	3	3	58	5.95e-2
400	15000	3	5	86	1.10e-2
400	15000	3	10	138	1.63e-2

(a) e vs i (b) e vs t

FIG. 6. *The evolution of the error versus the iteration index i and the time t, respectively, for Example 4.3, with SSDRM, SAPINN, DRM, and WAN.*

solvers can exhibit a steady and consistent convergence rate in practical computation. The figure indicates that the iteration will stop at different functional values (with same penalty parameter σ) for the three NNs. For the 3-3-3-1 NN, the error is larger than the other two NNs, and the loss value cannot stabilize at a lower level. This is generally attributed to the optimization error associated with finding a global minimizer to the empirical loss $\widehat{L}_\sigma(v_\theta)$: instead the (stochastic) optimizer generally may find an approximate minimizer, due to the highly nonconvex landscape of the empirical loss function $\widehat{L}_\sigma$. The optimization error appears to be dominating for many DNN solvers [38], and its precise mathematical characterization is still largely open. We refer interested readers to [15, 2] for detailed discussions on the persistent optimization error.

Now we compare the proposed SSDRM with three existing NN-based PDE solvers. See Figure 6 for the error e versus the iteration index i and the computing time t (in seconds). The proposed SSDRM significantly outperforms all existing approaches, and SAPINN beats both DRM and WAN by a large margin, since the latter two fail spectacularly to yield reasonable approximations. This is attributed to the presence of strong solution singularity, as discussed in section 2. These observations again fully confirm that from Example 4.1.

The next example on a line source is taken from [28, Example 5.2].

Example 4.4. The domain Ω is the unit cube $\Omega = (0,1)^3$, diffusion coefficient $\kappa \equiv 1$, a line source supported on the line segment connecting $\mathbf{x}_1 = (0,0,0.2)$ and $\mathbf{x}_2 = (0,0,0.8)$ with strength $f = x_3$, and $g = 0$, with a Dirichlet boundary condition $h = \frac{x_3}{4\pi} \ln \frac{|\mathbf{x}-\mathbf{x}_2|-(x_3-0.8)}{|\mathbf{x}-\mathbf{x}_1|-(x_3-0.2)} - (|\mathbf{x}-\mathbf{x}_1| - |\mathbf{x}-\mathbf{x}_2|)$. The exact solution u is given by $u = \frac{x_3}{4\pi} \ln \frac{|\mathbf{x}-\mathbf{x}_2|-(x_3-0.8)}{|\mathbf{x}-\mathbf{x}_1|-(x_3-0.2)} - (|\mathbf{x}-\mathbf{x}_1| - |\mathbf{x}-\mathbf{x}_2|)$.

In this example, we have to construct a function Φ_L whose Laplacian is a line source δ_Λ (supported on a line segment Λ). See the construction of Φ_L in section 3.1. Note that the singularities (lines) are actually placed on the boundary $\partial\Omega$ of the domain Ω. Then we split the solution into a singular part and a regular part v and approximate v by an NN v_θ. In the experiment, we have employed an NN v_θ with four layers, each layer having 10 neurons. The stopping criterion is met by the penalty factor $\sigma = 8000$, and the corresponding error is $e = 2.03\text{e-}2$. Two slices at $x_3 = 0$ and $x_3 = 1/2$ of the NN approximation are shown in Figure 7. The slice at $x_3 = 0$ lies on the boundary $\partial\Omega$ of the domain Ω. The error on the boundary $\partial\Omega$ is larger than that in the interior of the domain Ω, but overall the result is still quite satisfactory. From the slice at $x_3 = 1/2$, one can clearly identify the location of the line source (origin), and SSDRM can also yield good results around the singular point.

To examine the influence of the numbers of sample points (i.e., N_r and N_b) and the NN architecture on the convergence behavior of the training process (in the sense of optimization), we have experimented with different settings. The results are presented in Figure 8, where the penalty parameter σ is fixed at $\sigma = 8000$. The training of SSDRM converges robustly for all settings: the loss value decreases very steadily, and for all the settings the approximations exhibit very much comparable accuracy and achieve similar loss values. Although not presented, it is noted that the training may suffer from "overfitting," where the error e starts to increase after
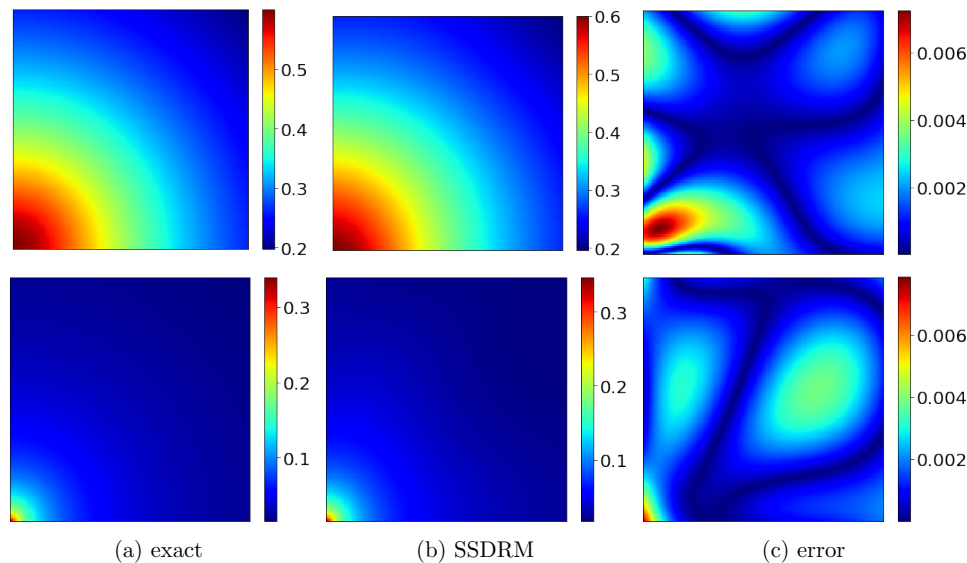

(a) exact   (b) SSDRM   (c) error

FIG. 7. *The numerical approximation for Example 4.4, slice at $x_3 = 0$ (top), and $x_3 = 1/2$ (bottom).*

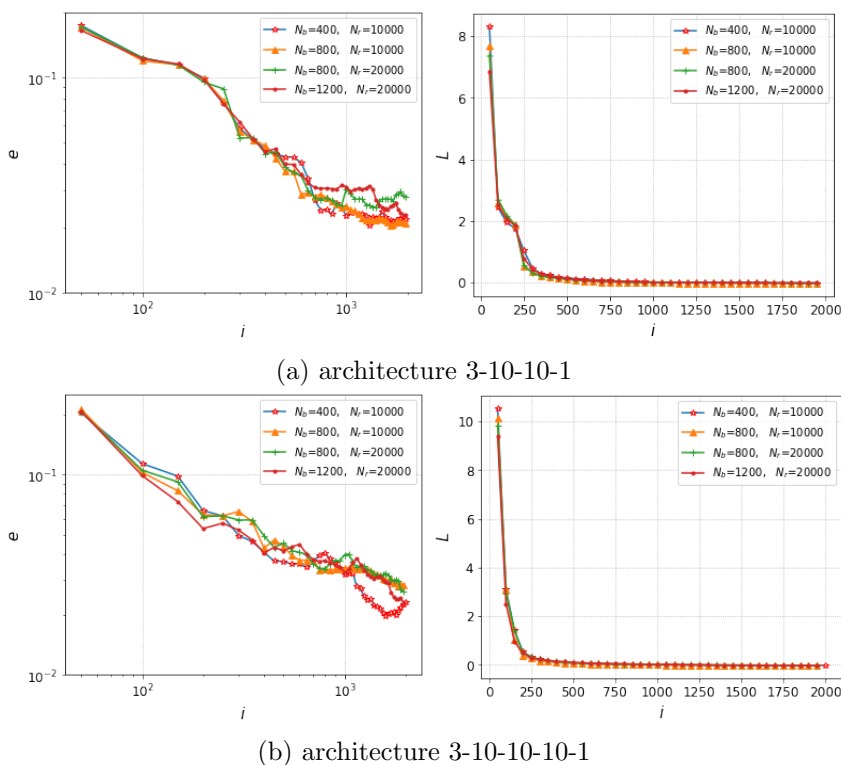(a) architecture 3-10-10-1

(b) architecture 3-10-10-10-1

FIG. 8. *The evolution of the error e and empirical loss $\widehat{L}_\sigma$ during the SSDRM training of Example 4.4 with different numbers (i.e., N_r and N_b) of samples in the domain Ω and on the boundary $\partial\Omega$ for two NN architectures.*

some initial iterations, when the penalty parameter σ is much smaller, e.g., $\sigma = 2000$, $N_b = 400$, and $N_r = 10000$ for the architecture 3-10-10-10-1. This agrees with previous discussions that one needs to choose the penalty parameter σ properly in order to achieve the best possible approximation.

The proposed SSDRM extends straightforwardly to the more complex case involving more than one singularity, or a combination of point and line sources. The next three-dimensional example contains a line source and two point singularities.

Example 4.5. The domain Ω is $\Omega = (-1,1)^3$, diffusion coefficient $\kappa = e^{x_3}$, with a line segment source $\{(0,0,x_3) : -1 \le x_3 \le 1\}$ with two endpoints $\mathbf{x}_1 = (0,0,-1)$, $\mathbf{x}_2 = (0,0,1)$, and two point sources concentrated at $\mathbf{x}_3 = (0,\frac{1}{2},0)$ and $\mathbf{x}_4 = (0,-\frac{1}{2},0)$ and $f = e^{x_3}$ and $g = -(x_1x_2 + x_1^2x_2^2 + x_2^2x_3^2 + x_3^2x_1^2)e^{x_1x_2x_3+x_3} + \frac{x_3e^{x_3}}{4\pi|\mathbf{x}-\mathbf{x}_3|^3} + \frac{x_3e^{x_3}}{4\pi|\mathbf{x}-\mathbf{x}_4|^3} + \frac{e^{x_3}}{4\pi}\left(\frac{1}{|\mathbf{x}-\mathbf{x}_2|} - \frac{1}{|\mathbf{x}-\mathbf{x}_1|}\right)$, and the Dirichlet boundary condition $h = \frac{1}{4\pi}\ln\frac{|\mathbf{x}-\mathbf{x}_2|-(x_3-1)}{|\mathbf{x}-\mathbf{x}_1|-(x_3+1)} + \frac{1}{4\pi}\frac{1}{|\mathbf{x}-\mathbf{x}_3|} + \frac{1}{4\pi}\frac{1}{|\mathbf{x}-\mathbf{x}_4|} + e^{x_1x_2x_3}$. The analytic solution u is given by $u = \frac{1}{4\pi}\ln\frac{|\mathbf{x}-\mathbf{x}_2|-(x_3-1)}{|\mathbf{x}-\mathbf{x}_1|-(x_3+1)} + \frac{1}{4\pi}\frac{1}{|\mathbf{x}-\mathbf{x}_3|} + \frac{1}{4\pi}\frac{1}{|\mathbf{x}-\mathbf{x}_4|} + e^{x_1x_2x_3}$.

The NN architecture for this example is taken to be 3-6-6-1. The training terminates at $e = 4.37\text{e-}3$ and $\sigma = 1730$. The NN approximation is shown in Figure 9, with the two slices at $x_3 = -1$ and $x_3 = 0$, respectively. The results confirm previous observations: the singular part is exactly resolved, and the regular part v is well-approximated so that SSDRM can produce an excellent approximation to the
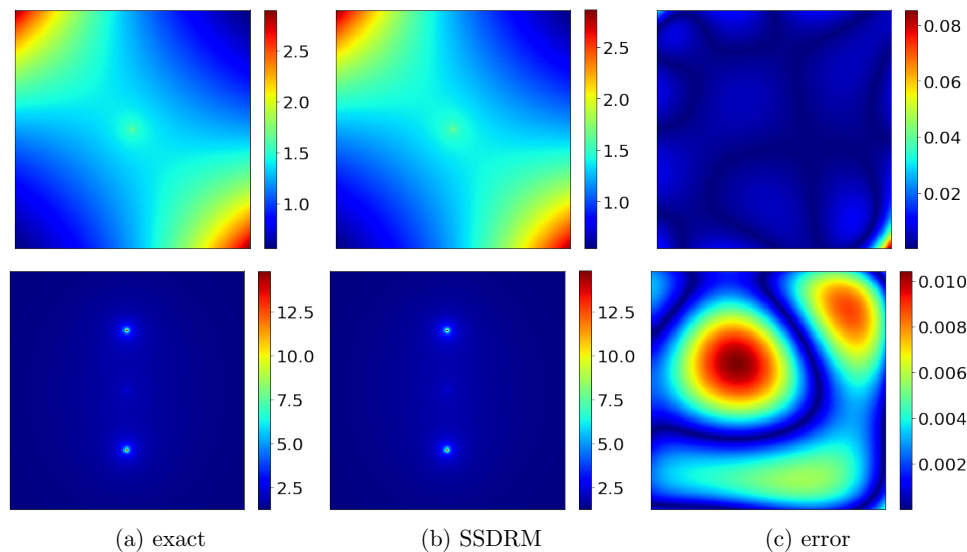
| (a) exact | (b) SSDRM | (c) error |

FIG. 9. *The NN approximation for Example* 4.5, *slices at* $x_3 = -1$ *(top), and* $x_3 = 0$ *(bottom).*

exact solution. This example shows clearly the flexibility of the proposed approach for elliptic problems with multiple singularities.

The next example further shows the flexibility of the proposed SSDRM for resolving singular sources living on high-dimensional subspaces, as mentioned in section 3.1.

Example 4.6. This example is a five-dimensional Poisson equation, with the domain $\Omega = (-1,1)^5$, surface sources at $\mathbf{x}_1' = (\frac{1}{2}, \frac{1}{2})$, $\mathbf{x}_2' = (0, \frac{1}{2})$, $\mathbf{x}_3' = (-\frac{1}{2}, \frac{1}{2})$, $\mathbf{x}_4' = (\frac{1}{2}, 0)$, $\mathbf{x}_5' = (0,0)$, $\mathbf{x}_6' = (-\frac{1}{2}, 0)$, $\mathbf{x}_7' = (\frac{1}{2}, -\frac{1}{2})$, $\mathbf{x}_8' = (0, -\frac{1}{2})$, and $\mathbf{x}_9' = (-\frac{1}{2}, -\frac{1}{2})$ (with $\mathbf{x}' = (x_1, x_2)$), $f \equiv 1$, and $g = -10$. The Dirichlet boundary condition $h = \sum_{i=1}^{9} -\frac{1}{2\pi} \ln |\mathbf{x}' - \mathbf{x}_i'| + |\mathbf{x}|^2$. The analytic solution u is given by $u = \sum_{i=1}^{9} -\frac{1}{2\pi} \ln |\mathbf{x}' - \mathbf{x}_i'| + |\mathbf{x}|^2$.

The singularity is understood to be supported on three-dimensional affine subspace through the bounded region Ω, similar to Gjerde et al. [28]. Then the corresponding singularity function capturing the source term $\delta(x_1', x_2')$ in five-dimensional space is given by $\Phi(\mathbf{x}') = -\frac{1}{2\pi} \ln |\mathbf{x}' - \mathbf{x}_i'|$. The NN for approximating the regular part v has two layers, and each hidden layer has 10 neurons. The loop terminates at the penalty factor $\sigma = 1922$, and the error e of the NN approximation is $e = 8.14\mathrm{e}{-3}$ in the domain. One slice of the approximation at $x_3 = x_4 = x_5 = 0$ is shown in Figure 10. The numerical results show again that the singularities are accurately resolved with a small pointwise error.

5. Error analysis. Now we discuss the error analysis for SSDRM. Due to the use of singularity splitting, the issue boils down to the analysis of the standard DRM for smooth solutions, which has been carried out independently in several recent works under different problem settings and architectures [47, 19, 20, 33, 51, 46]. Below we summarize relevant theoretical results for the Dirichlet case, to provide theoretical underpinnings of the SSDRM and also shed valuable insights into the influence of various algorithmic parameters (e.g., penalty parameter σ and NN width and depth). The Neumann case is simpler (since there is no error associated with the penalization)
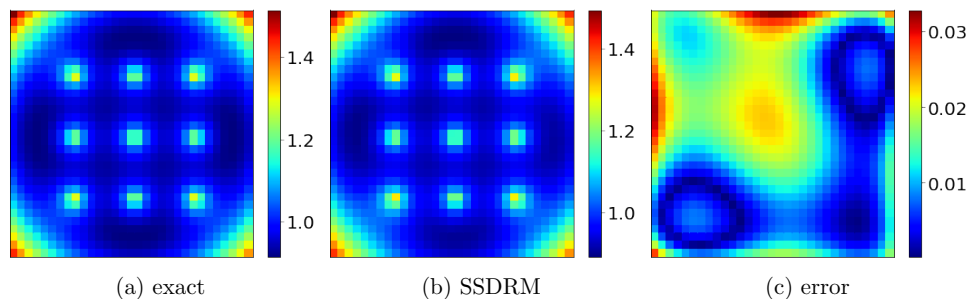
(a) exact (b) SSDRM (c) error

FIG. 10. *The NN approximation for Example* 4.6, *slice at* $x_3 = x_4 = x_5 = 0$.

and can be treated similarly. Throughout the discussion, the domain Ω is assumed to be smooth.

5.1. Error estimate of the penalty method. First, we discuss the error incurred by the penalty method, i.e., how the error decreases as the penalty parameter $\sigma \to \infty$. We denote the minimizer of $L_\sigma(v)$ by v_σ and the solution of problem (3.3) by v^*. The next result shows that v_σ actually can be viewed as a Robin approximation to the Dirichlet problem.

PROPOSITION 5.1. *Let* $v_\sigma \in H^1(\Omega)$ *be the minimizer of* $L_\sigma(v)$. *Then* v_σ *is the weak solution of the following Robin boundary value problem:*

$$(5.1) \qquad \begin{cases} -\nabla \cdot (\kappa \nabla v) = F & in\ \Omega, \\ v + \dfrac{\kappa}{\sigma} \dfrac{\partial v}{\partial n} = \tilde{h} & on\ \partial\Omega. \end{cases}$$

Proof. Since $v_\sigma \in H^1(\Omega)$ is the minimizer of $L_\sigma(v)$, for every $\phi \in H^1(\Omega)$ we have $L_\sigma(v_\sigma) \le L_\sigma(v_\sigma + \lambda\phi)$, $\lambda \in \mathbb{R}$. Let $f(\lambda) = L_\sigma(v_\sigma + \lambda\phi)$. Direct computation gives

$$f'(\lambda) = \lambda \left((\kappa\nabla\phi, \nabla\phi) + \sigma(\phi, \phi)_{L^2(\partial\Omega)} \right) + (\kappa\nabla v_\sigma, \nabla\phi) - (F, \phi) + \sigma(v_\sigma - \tilde{h}, \phi)_{L^2(\partial\Omega)}.$$

The minimizing property of v_σ implies $f'(0) = 0$, and consequently

$$(5.2) \qquad f'(0) = (\kappa\nabla v_\sigma, \nabla\phi) - (F, \phi) + \sigma(v_\sigma - \tilde{h}, \phi)_{L^2(\partial\Omega)} = 0 \quad \forall \phi \in H^1(\Omega).$$

This is the variational formulation of (5.1). So v_σ is the weak solution of problem (5.1). $\qquad\square$

The penalization is a type of singular perturbation. It has been studied under various situations [13, 6]. To derive error bounds, we recall a regularity result for the Robin boundary value problem essentially due to Costabel and Dauge [13].

LEMMA 5.2. *Let* $\kappa \in C^2(\overline{\Omega})$. *Fix* $\epsilon > 0$, *and let* z^ϵ *be the solution to the Robin boundary value problem*

$$(5.3) \qquad \begin{cases} -\nabla \cdot (\kappa\nabla z^\epsilon) = 0 & in\ \Omega, \\ \epsilon\kappa\dfrac{\partial z^\epsilon}{\partial n} + z^\epsilon = g & on\ \partial\Omega. \end{cases}$$

Then, for any $t \in [0, 3/2]$ *and* $t \le s \le t+1$, *there exists some* $C > 0$ *independent of* ϵ *such that*

$$\|z^\epsilon\|_{H^{1+s}(\Omega)} \le C\epsilon^{t-s}\|g\|_{H^{\frac{1}{2}+t}(\partial\Omega)}.$$

Proof. The proof follows closely the argument of [13, Lemma 2.1 and Corollary 2.2] for the Laplacian, but it is adapted to the case of a variable coefficient. The proof utilizes the Dirichlet-to-Neumann map: for a given $g \in H^{\frac{1}{2}}(\partial\Omega)$, let $\Lambda_\kappa g = \kappa \frac{\partial w_g}{\partial n}\big|_{\partial\Omega}$, where $w_g \in H^1(\Omega)$ solves

$$(5.4) \qquad \begin{cases} -\nabla \cdot (\kappa \nabla w_g) = 0 & \text{in } \Omega, \\ \quad\quad\quad\quad\; w_g = g & \text{on } \partial\Omega. \end{cases}$$

Let $T : H^s(\Omega) \to H^{s-\frac{1}{2}}(\partial\Omega)$ be the trace operator. Then Tz^ϵ can be represented as

$$(5.5) \qquad\qquad Tz^\epsilon = (\epsilon\Lambda_\kappa + I)^{-1}g.$$

Then it suffices to show the (uniform in ϵ) boundedness of the operator $(\epsilon\Lambda_\kappa + I)^{-1}$ in $H^s(\partial\Omega)$ with $s \geq 0$. Note that for any $g \in H^{\frac{1}{2}}(\partial\Omega)$

$$((\epsilon\Lambda_\kappa + I)g, g)_{L^2(\partial\Omega)} = \epsilon\|\sqrt{\kappa}\nabla w_g\|^2_{L^2(\Omega)} + \|g\|^2_{L^2(\partial\Omega)} \geq \|g\|^2_{L^2(\partial\Omega)}.$$

Therefore, the operator $\epsilon\Lambda_\kappa + I$ is positive definite in $L^2(\partial\Omega)$ and

$$\|(\epsilon\Lambda_\kappa + I)^{-1}\|_{L^2(\partial\Omega)\to L^2(\partial\Omega)} \leq C,$$

where the constant C is independent of ϵ. To show the uniform bound of the operator in $H^s(\partial\Omega)$, let (x', x_n) be a local coordinate near a point $x_0 \in \partial\Omega$ so that the boundary is given by $x_n = 0$. The function $\lambda_\kappa(x', \xi)$ denotes the symbol of Λ_κ in this coordinate. Then the following asymptotic expansion holds (see, e.g., [66, Theorem 4.1] and [64, Theorem 0.1]):

$$\lambda_\kappa(x', \xi') = \kappa(x', 0)|\xi'| + a_0(x', \xi') + r(x', \xi'),$$

where $\kappa(x', 0)|\xi'|$ denotes a classical pseudodifferential operator of order 1, $a_0(x', \xi')$ is homogeneous of degree 0 in ξ', and the residue $r(x', \xi')$ is a classical symbol of order -1. Hence, for any $s \geq 0$

$$\|(\epsilon\Lambda_\kappa + I)^{-1}g\|_{H^s(\partial\Omega)} \leq C\|(\Lambda_\kappa + I)^s(\epsilon\Lambda_\kappa + I)^{-1}g\|_{L^2(\partial\Omega)}$$
$$= C\|(\epsilon\Lambda_\kappa + I)^{-1}(\Lambda_\kappa + I)^s g\|_{L^2(\partial\Omega)} \leq C\|(\Lambda_\kappa + I)^s g\|_{L^2(\partial\Omega)} \leq C\|g\|_{H^s(\partial\Omega)},$$

since Λ_κ is self-adjoint and semipositive definite, and $(\epsilon\Lambda_\kappa + I)^{-1}$ commutes with $(\Lambda_\kappa + I)^s$. Then, by the defining identity (5.5), we obtain

$$\|Tz^\epsilon\|_{H^s(\partial\Omega)} \leq C\|g\|_{H^s(\partial\Omega)}.$$

Now the smoothing property of the Dirichlet problem implies

$$(5.6) \qquad \|z^\epsilon\|_{H^{1+t}(\Omega)} \leq C\|Tz^\epsilon\|_{H^{\frac{1}{2}+t}(\partial\Omega)} \leq C\|g\|_{H^{\frac{1}{2}+t}(\partial\Omega)} \quad \forall 0 \leq t \leq \tfrac{3}{2}.$$

Meanwhile, the Robin problem (5.3) can be rewritten as

$$(5.7) \qquad \begin{cases} -\nabla \cdot (\kappa \nabla z^\epsilon) = 0 & \text{in } \Omega, \\ \kappa\dfrac{\partial z^\epsilon}{\partial n} + z^\epsilon = \epsilon^{-1}(g - (1-\epsilon)z^\epsilon) & \text{on } \partial\Omega. \end{cases}$$

Then the smoothing property for the Robin boundary value problem leads to [23, Theorem 3.10 (iii)]

$$(5.8) \qquad \begin{aligned} \|z^\epsilon\|_{H^{\frac{3}{2}+t}(\Omega)} &\leq C\epsilon^{-1}\|g - (1-\epsilon)Tz^\epsilon\|_{H^{\frac{1}{2}+t}(\partial\Omega)} \\ &\leq C\epsilon^{-1}(\|g\|_{H^{\frac{1}{2}+t}(\partial\Omega)} + \|Tz^\epsilon\|_{H^{\frac{1}{2}+t}(\partial\Omega)}) \leq C\epsilon^{-1}\|g\|_{H^{\frac{1}{2}+t}(\partial\Omega)}, \end{aligned}$$

where the last step is due to the estimate (5.6). Then interpolating between (5.6) and (5.8) gives the desired result. $\qquad\square$

Now we can state a bound on the error $v_\sigma - v^*$, arising from approximating the Dirichlet problem with a Robin one.

THEOREM 5.3. *Let $F \in H^t(\Omega)$, and let $\tilde{h} \in H^{t+\frac{3}{2}}(\partial\Omega)$, with $t \in [-1, 0]$. Then, for any $t \le s \le t+1$, the following error bound holds:*

$$\|v_\sigma - v^*\|_{H^{1+s}(\Omega)} \le C\sigma^{s-t-1}(\|F\|_{H^t(\Omega)} + \|\tilde{h}\|_{H^{t+\frac{3}{2}}(\partial\Omega)}).$$

Proof. Note that $w_\sigma = v_\sigma - v^*$ satisfies

$$\begin{cases} -\nabla \cdot (\kappa \nabla w_\sigma) = 0 & \text{in } \Omega, \\ \sigma^{-1}\kappa \dfrac{\partial w_\sigma}{\partial n} + w_\sigma = -\sigma^{-1}\kappa \dfrac{\partial v^*}{\partial n} & \text{on } \partial\Omega. \end{cases}$$

Then, by Lemma 5.2 and the trace theorem, there holds for $t \le s \le t+1$ that

$$\|w_\sigma\|_{H^{1+s}(\Omega)} \le C\sigma^{s-t}\Big\|\sigma^{-1}\kappa \frac{\partial v^*}{\partial n}\Big\|_{H^{\frac{1}{2}+t}(\partial\Omega)}$$

$$= C\sigma^{s-t-1}\Big\|\kappa \frac{\partial v^*}{\partial n}\Big\|_{H^{\frac{1}{2}+t}(\partial\Omega)} \le C\sigma^{s-t-1}\|v^*\|_{H^{2+t}(\Omega)}.$$

Meanwhile, by the standard elliptic regularity theory, we have

$$\|v^*\|_{H^{2+t}(\Omega)} \le C\Big(\|F\|_{H^t(\Omega)} + \|\tilde{h}\|_{H^{\frac{3}{2}+t}(\partial\Omega)}\Big).$$

Combining the preceding two estimates completes the proof the theorem. $\qquad\square$

Remark 5.1. Theorem 5.3 indicates that under suitable regularity assumptions on the solution v^*, the error due to penalization decays (sub)linearly. The error bound in Theorem 5.3 is known in various special forms. For example, in both [20] and [51], the following error bound was derived for the case of $\tilde{h} \equiv 0$ and $F \in L^2(\Omega)$:

$$\|v^* - v_\sigma\|_{H^1(\Omega)} \le C\sigma^{-1}\|F\|_{L^2(\Omega)}.$$

It was derived for a strongly coercive bilinear form in the space $H^1(\Omega)$ in [20], using an energy argument. In [51, Theorem 9], it was derived using the theory of the Steklov eigenvalue problem (see [6] and references therein). In contrast to existing results, Theorem 5.3 gives the bound for a broader range of regularity conditions on the problem data F and $\tilde{h}$.

Under the a priori assumption $v^* \in H^1(\Omega)$ only, it can be proved that

$$\|v_\sigma - v^*\|_{L^2(\Omega)} \le C(\kappa)(\|F\|_{(H^{\frac{1}{2}}(\Omega))'} + \|\tilde{h}\|_{H^{\frac{1}{2}}(\Omega)})\sigma^{-\frac{1}{2}}.$$

See [51, Lemma 17] for a detailed proof when $\tilde{h} \equiv 0$, and the case $\tilde{h} \not\equiv 0$ follows similarly. One can also show convergence in $H^1(\Omega)$ but without a rate, using a standard compactness argument from calculus of variations, i.e.,

$$\lim_{\sigma \to \infty} \|v^* - v_\sigma\|_{H^1(\Omega)} = 0.$$

Further discussions on the convergence of the penalization under even weaker regularity assumption for the homogeneous Laplace equation (i.e., $F \equiv 0$), including very weak solutions with Dirichlet boundary data $\tilde{h} \in H^s(\Omega)$ with $0 < s < \frac{1}{2}$, can be found in [6, section 7]

$$\|v_\sigma - v^*\|_{L^2(\Omega)} \le C\sigma^{-s}\|\tilde{h}\|_{H^s(\partial\Omega)}.$$

See [6, Theorem 7.1] for the precise statement.

5.2. Generalization error analysis. In practical computation, we use an empirical loss $\widehat{L}_\sigma(v_\theta)$ instead of the continuous loss $L_\sigma(v_\theta)$, due to the use of the Monte Carlo method to approximate the integrals. Let $\widehat{v_\sigma}$ be a (global) minimizer of the empirical loss $\widehat{L}_\sigma(v_\theta)$ over the set $\mathcal{A}$ of NN functions (with a fixed architecture, e.g., depth, number of nonzero parameters, and maximum bound on the parameter vectors). Now we bound the error between $\widehat{v_\sigma}$ and v_σ. This kind of analysis is commonly known as the generalization error in statistical learning theory [4, 60]. For the discussion below, we assume the modified source $F \in L^\infty(\Omega)$, $\tilde{h} \in L^\infty(\partial\Omega)$, which holds for $\kappa, f \in W^{2,\infty}(\Omega)$ and $h \in L^\infty(\partial\Omega)$ and being locally constant in a neighborhood of the singularity δ. This assumption is due to the limitation of the proof technique using Rademacher complexity. The next lemma gives an error decomposition into the quadrature error and approximation error.

LEMMA 5.4. *Let $\mathcal{A}$ be a set of NN functions of fixed depth, width, and upper bound on the parameters. Let $\widehat{v}_\sigma$ be the minimizer of $\widehat{L}_\sigma(v)$ over $\mathcal{A}$. Then*

$$(5.9) \qquad \|\nabla(\widehat{v_\sigma} - v_\sigma)\|^2_{L^2(\Omega)} + \sigma\|\widehat{v_\sigma} - v_\sigma\|^2_{L^2(\partial\Omega)} \leq C(\kappa)(\sup_{\widehat{v}\in\mathcal{A}} |L_\sigma(\widehat{v}) - \widehat{L}_\sigma(\widehat{v})|$$
$$+ \inf_{\widehat{v}\in\mathcal{A}} \sigma\|\widehat{v} - v_\sigma\|^2_{H^1(\Omega)}).$$

Proof. This result follows from direct computation. For every $\widehat{v} \in \mathcal{A}$, by the minimizing property $\widehat{L}_\sigma(\widehat{v_\sigma}) \leq \widehat{L}_\sigma(\widehat{v})$, we have

$$L_\sigma(\widehat{v_\sigma}) - L_\sigma(v_\sigma) = [L_\sigma(\widehat{v_\sigma}) - \widehat{L}_\sigma(\widehat{v_\sigma})] + [\widehat{L}_\sigma(\widehat{v_\sigma}) - \widehat{L}_\sigma(\widehat{v})]$$
$$+ [\widehat{L}_\sigma(\widehat{v}) - L_\sigma(\widehat{v})] + [L_\sigma(\widehat{v}) - L_\sigma(v_\sigma)]$$
$$\leq 2\sup_{\widehat{v}\in\mathcal{A}} |L(\widehat{v}) - \widehat{L}_\sigma(\widehat{v})| + L_\sigma(\widehat{v}) - L_\sigma(v_\sigma).$$

Taking the infimum over $\widehat{v} \in \mathcal{A}$ gives

$$(5.10) \qquad L_\sigma(\widehat{v_\sigma}) - L_\sigma(v_\sigma) \leq 2\sup_{\widehat{v}\in\mathcal{A}} |L_\sigma(\widehat{v}) - \widehat{L}_\sigma(\widehat{v})| + \inf_{\widehat{v}\in\mathcal{A}} [L_\sigma(\widehat{v}) - L_\sigma(v_\sigma)].$$

For any $\widehat{v} \in \mathcal{A}$, and $w = \widehat{v} - v_\sigma$, we deduce

$$L_\sigma(\widehat{v}) - L_\sigma(v_\sigma) = \frac{1}{2}(\kappa\nabla w, \nabla w) + (\kappa\nabla v_\sigma, \nabla w) + \frac{\sigma}{2}\|w\|^2_{L^2(\partial\Omega)}$$
$$+ \sigma(v_\sigma - \tilde{h}, w)_{L^2(\partial\Omega)} - (F, w)$$
$$= \frac{1}{2}(\kappa\nabla w, \nabla w) + \frac{\sigma}{2}\|w\|^2_{L^2(\partial\Omega)},$$

where the last step is due to the weak formulation of v_σ, i.e.,

$$(\kappa\nabla v_\sigma, \nabla w) + \sigma(v_\sigma - \tilde{h}, w)_{L^2(\partial\Omega)} = (F, w).$$

Upon substituting the identity into (5.10) and applying the trace theorem, we obtain the desired bound. □

The error decomposition in Lemma 5.4 provides interesting insights into the total error of the NN approximation $\widehat{v_\theta}$: The two terms $\sup_{\widehat{v}\in\mathcal{A}} |L_\sigma(\widehat{v}) - \widehat{L}_\sigma(\widehat{v})|$ and $\inf_{\widehat{v}\in\mathcal{A}} \sigma\|\widehat{v} - v_\sigma\|_{H^1(\Omega)}$ respectively represent the statistical error, due to the use of the Monte Carlo method to approximate the integral, and the approximation error, due to the use of NNs to approximate the function v_σ.

Note that in the analysis, we have assumed that a global optimizer $\widehat{v_\sigma}$ of the empirical loss $\widehat{L}_\sigma(v_\theta)$ can be numerically realized, which is unfortunately generally

not the case in practice. Indeed, the landscape of the empirical loss $\widehat{L}_\sigma(v_\theta)$ is highly nonconvex (see [42] for graphical illustrations of PINN), due to the nonlinearity of the activation function ρ, and there is no guarantee that an optimizer will find a global minimizer $\widehat{\theta}$. Instead only an approximate minimizer $\tilde{\theta}$ can be expected in practice. This leads to an additional source of error, known as optimization error in the literature. It is often observed to be dominating in many neural PDE solvers [68, 42, 38]. Up to now, it is still completely open to rigorously analyze the optimization error and thus is often ignored from the analysis.

Below we bound the approximation error and the statistical error. This kind of analysis has been pursued in several recent works [47, 19, 37, 51, 46] under different assumptions on the activation function ρ and NN architecture. Under the assumption $F \in L^\infty(\Omega)$, the work [37] provides relevant estimates on the quadrature error by means of the Monte Carlo method using Rademacher complexity and the approximation error of the penalized solution with NNs using the approximation theory from [32].

LEMMA 5.5. *For $\tilde{h} \in H^{\frac{3}{2}}(\partial\Omega)$ and $F \in L^2(\Omega)$, the solution v_σ (of the penalized problem* (5.1)) *satisfies*

$$\|v_\sigma\|_{H^2(\Omega)} \leq C(\|\tilde{h}\|_{H^{\frac{3}{2}}(\partial\Omega)} + \|F\|_{L^2(\Omega)}).$$

Proof. The given conditions on $\tilde{h}$ and F and the standard elliptic regularity theory imply

$$\|v^*\|_{H^2(\Omega)} \leq C(\|\tilde{h}\|_{H^{\frac{3}{2}}(\partial\Omega)} + \|F\|_{L^2(\Omega)}).$$

Meanwhile, $z = v_\sigma - v^*$ satisfies problem (5.3) with $\epsilon = \sigma^{-1}$ and $g = -\epsilon\kappa\frac{\partial v^*}{\partial n}$. Then applying Lemma 5.2 and the trace theorem yields

$$\|v^* - v_\sigma\|_{H^2(\Omega)} \leq C\epsilon^{-1}\|\epsilon\kappa\frac{\partial v^*}{\partial n}\|_{H^{\frac{1}{2}}(\partial\Omega)} \leq C\|v^*\|_{H^2(\Omega)}.$$

Then the triangle inequality shows the desired assertion. □

The next result gives an error bound on the NN approximation $\widehat{v_\sigma}$, by suitably adapting the estimates given in [37, Theorems 4.1 and 5.13], and the a priori regularity on v_σ in Lemma 5.5. The expectation $\mathbb{E}_{\{X_i\}_{i=1}^N, \{Y_j\}_{j=1}^N}[\cdot]$ is taken with respect to the random sampling points $\{X_i\}_{i=1}^N$ and $\{Y_j\}_{j=1}^N$ in the domain Ω and on the boundary $\partial\Omega$. We recall also the set $\mathcal{N}_\rho(D, N_\theta, R)$ of NN functions (of a fixed architecture) defined in (3.12).

THEOREM 5.6. *Let $F \in L^\infty(\Omega)$ and $\tilde{h} \in H^{\frac{3}{2}}(\partial\Omega) \cap L^\infty(\partial\Omega)$. Let N be the number of training samples on the domain Ω and the boundary $\partial\Omega$, and let ρ be logistic function $\frac{1}{1+e^{-x}}$ or tanh function $\frac{e^x - e^{-x}}{e^x + e^{-x}}$. Then, for any tolerance $\epsilon > 0$ and $\mu \in (0,1)$, there exists a parameterized NN function class*

$$\mathcal{A} = \mathcal{N}_\rho\left(C\ln(d+1), C(d)\epsilon^{-\frac{d}{1-\mu}}, C(d)\epsilon^{-\frac{9d+8}{2-2\mu}}\right)$$

such that with the number of samples $N = O(\epsilon^{-C\frac{d\ln(d+1)}{1-\mu}})$ both in the domain Ω and on the boundary $\partial\Omega$, the NN approximation $\widehat{v_\sigma} \in \mathcal{A}$ satisfies

$$(5.11) \quad \mathbb{E}_{\{X\}_{i=1}^N, \{Y_j\}_{j=1}^N}\left[\|\nabla(\widehat{v_\sigma} - v_\sigma)\|_{L^2(\Omega)}^2 + \sigma\|\widehat{v_\sigma} - v_\sigma\|_{L^2(\partial\Omega)}^2\right] \leq C(\Omega, \kappa, F, \tilde{h})\sigma\epsilon^2.$$

Thus, for the NN approximation $\widehat{u_\sigma} = \kappa^{-1}f\Phi + \widehat{v_\sigma}$ by SSDRM, there holds that

$$(5.12) \quad \mathbb{E}_{\{X_i\}_{i=1}^N, \{Y_j\}_{j=1}^N}\left[\|\widehat{u_\sigma} - u^*\|_{H^1(\Omega)}^2\right] \leq C(\Omega, \kappa, F, \tilde{h})\sigma\epsilon^2 + c(F, \tilde{h})\sigma^{-2}.$$

Proof. The stated result follows from several known estimates from [37] and relies on estimating the approximation error and statistical error separately. We sketch the proof for the convenience of readers. First, we bound the approximation error $\inf_{\widehat{v} \in \mathcal{A}} \|\widehat{v} - v_\sigma\|_{H^1(\Omega)}$ by Lemma 5.5:

$$
\inf_{\widehat{v} \in \mathcal{A}} \|\widehat{v} - v_\sigma\|_{H^1(\Omega)} = \|v_\sigma\|_{H^2(\Omega)} \inf_{\widehat{v} \in \mathcal{A}} \left\| \frac{\widehat{v}}{\|v_\sigma\|_{H^2(\Omega)}} - \frac{v_\sigma}{\|v_\sigma\|_{H^2(\Omega)}} \right\|_{H^1(\Omega)}
$$

$$
= \|v_\sigma\|_{H^2(\Omega)} \inf_{\widehat{v} \in \mathcal{A}} \left\| \widehat{v} - \frac{v_\sigma}{\|v_\sigma\|_{H^2(\Omega)}} \right\|_{H^1(\Omega)}
$$

$$
\leq C \big(\|\tilde{h}\|_{H^{\frac{3}{2}}(\partial\Omega)} + \|F\|_{L^2(\Omega)} \big) \inf_{\widehat{v} \in \mathcal{A}} \left\| \widehat{v} - \frac{v_\sigma}{\|v_\sigma\|_{H^2(\Omega)}} \right\|_{H^1(\Omega)}.
$$

By [37, Theorem 4.1] (or [32, Proposition 4.8]), there exists an NN function $v_\rho \in \mathcal{A} = \mathcal{N}_\rho(C\ln(d+1), C(d)\epsilon^{-\frac{d}{1-\mu}}, C(d)\epsilon^{-\frac{9d+8}{2-2\mu}})$ such that

$$
\left\| v_\rho - \frac{v_\sigma}{\|v_\sigma\|_{H^2(\Omega)}} \right\|_{H^1(\Omega)} \leq \epsilon.
$$

Consequently, we get

(5.13)
$$
\inf_{\widehat{v} \in \mathcal{A}} \|\widehat{v} - v_\sigma\|_{H^1(\Omega)}^2 \leq C(\Omega, F, \tilde{h})\epsilon^2.
$$

To bound the statistical error $\sup_{\widehat{v} \in \mathcal{A}} |L_\sigma(\widehat{v}) - \widehat{L_\sigma}(\widehat{v})|$, we customarily split it into three terms:

$$
\sup_{\widehat{v} \in \mathcal{A}} |L_\sigma(\widehat{v}) - \widehat{L_\sigma}(\widehat{v})|
$$

$$
\leq \sup_{\widehat{v} \in \mathcal{A}} \left| |\Omega| \mathbb{E}_{X \sim U(\Omega)} \frac{\kappa(X)|\nabla\widehat{v}(X)|^2}{2} - \frac{|\Omega|}{N} \sum_{i=1}^{N} \frac{\kappa(X_i)|\nabla\widehat{v}(X_i)|^2}{2} \right|
$$

$$
+ \sup_{\widehat{v} \in \mathcal{A}} \left| |\Omega| \mathbb{E}_{X \sim U(\Omega)} \widehat{v}(X)F(X) - \frac{|\Omega|}{N} \sum_{i=1}^{N} \widehat{v}(X_i)F(X_i) \right|
$$

$$
+ \frac{\sigma}{2} \sup_{\widehat{v} \in \mathcal{A}} \left| |\partial\Omega| \mathbb{E}_{Y \sim U(\partial\Omega)} [T\widehat{v}(Y) - \tilde{h}(Y)]^2 - \frac{|\partial\Omega|}{N} \sum_{j=1}^{N} [T\widehat{v}(Y_j) - \tilde{h}(Y_j)]^2 \right|.
$$

Then, by applying [37, Theorem 5.13] (with the following setting of the theorem statement: the depth $\mathcal{D} = C\ln(d+1)$, the number of nonzero parameters $\mathbf{n}_{\mathcal{D}} = C(d)\epsilon^{-\frac{d}{1-\mu}}$, and the upper bound $B_\theta = C(d)\epsilon^{-\frac{9d+8}{2-2\mu}}$ on each parameter) to the above three terms, we obtain

(5.14)
$$
\mathbb{E}_{\{X_i\}_{i=1}^N, \{Y_j\}_{j=1}^N} \left[\sup_{\widehat{v} \in \mathcal{A}} |L_\sigma(\widehat{v}) - \widehat{L_\sigma}(\widehat{v})| \right] \leq C(\Omega, \kappa)\sigma\epsilon^2,
$$

when the number N of sampling points in the domain Ω and on the boundary $\partial\Omega$ is set to $N = C(d, \Omega)\epsilon^{-C\frac{d\ln(d+1)}{1-\mu}}$. This estimate was proved using Rademacher complexity and Lipschitz continuity of the NN output with respect to the NN parameter vector θ. Last, substituting the estimates (5.13) and (5.14) into Lemma 5.4 yields the desired estimate (5.11). The other estimate (5.12) is direct from the construction of the approximation $\widehat{u_\sigma}$, the Poincaré inequality, and Theorem 5.3. □

Note that, generally, the approximation $\widehat{u_\sigma}$ and the true solution u^* do not necessarily belong to the space $H^1(\Omega)$, and thus an error estimate in the $H^1(\Omega)$ norm cannot be expected. Nonetheless, the error $\widehat{u_\sigma} - u^*$ does converge in $H^1(\Omega)$, since the singularity splitting technique extracts the leading singularity directly. The penalty parameter σ is updated using the path-following strategy, and it is increased geometrically by a factor $\eta > 1$ in every outer loop. Thus, as k increases, σ_k grows exponentially so the error due to the penalization method decays exponentially fast to zero; cf. Theorem 5.3. However, this comes at the price of increasing the generalization error, as evidenced by the factor σ in the first term of the error bound. This error can be partially offset by ϵ, which is in turn controlled by the number of sample points (in the domain and on the boundary), and the complexity of the NN (width, depth, and weights). This short analysis clearly indicates that there might be an optimal value for the penalty parameter σ to balance the trade-off between the different sources of the errors. Of course this analysis has left out one very important point in the analysis: it assumes that a global minimizer $\widehat{v_\sigma}$ to the empirical loss $\widehat{L}_\sigma(v_\theta)$ can be found, which is generally not true, due to the highly complex landscape of the empirical loss $\widehat{L}_\sigma(v_\theta)$.

6. Conclusions. In this work, we have investigated the efficient NN solution of the variable coefficient second-order elliptic problems with singular sources, which include point sources, line sources, and their combinations. The presence of the singular source prevents a direct application of the standard NN-based solvers, e.g., PINNs, the deep Ritz method, and weak adversarial networks. We have proposed a simple modification of the standard deep Ritz method (which is based on the Ritz variational formulation) by splitting the solution into a regular part and a singular part, where the singular part is expressed analytically using the fundamental solution to the Laplace equation. Extensive numerical experiments clearly confirm its efficiency and accuracy, when compared with existing NN-based approaches. Furthermore, we have discussed relevant theoretical issues.

Although the idea of singularity splitting has only been illustrated on the deep Ritz method, it applies equally well to other NN-based approaches, e.g., PINNs and weak adversarial networks, and potentially can also be very promising for solving elliptic problems with singular sources, due to the much improved regularity of the regular part of the splitting. These generalizations and related issues, e.g., evolution equations, will be explored in future works.

REFERENCES

[1] R. A. ADAMS AND J. J. F. FOURNIER, *Sobolev Spaces*, 2nd ed., Elsevier/Academic Press, Amsterdam, 2003.
[2] B. ADCOCK AND N. DEXTER, *The gap between theory and practice in function approximation with deep neural networks*, SIAM J. Math. Data Sci., 3 (2021), pp. 624–655, https://doi.org/10.1137/20M131309X.
[3] E. L. ALLGOWER AND K. GEORG, *Introduction to Numerical Continuation Methods*, SIAM, Philadelphia, 2003, https://doi.org/10.1137/1.9780898719154.
[4] M. ANTHONY AND P. L. BARTLETT, *Neural Network Learning: Theoretical Foundations*, Cambridge University Press, Cambridge, UK, 1999, https://doi.org/10.1017/CBO9780511624216.

[5] T. APEL, O. BENEDIX, D. SIRCH, AND B. VEXLER, *A priori mesh grading for an elliptic problem with Dirac right-hand side*, SIAM J. Numer. Anal., 49 (2011), pp. 992–1005, https://doi.org/10.1137/090778018.

[6] G. AUCHMUTY, *Robin approximation of Dirichlet boundary value problems*, Numer. Funct. Anal. Optim., 39 (2018), pp. 999–1010, https://doi.org/10.1080/01630563.2018.1453520.

[7] I. BABUŠKA, *Error-bounds for finite element method*, Numer. Math., 71 (1970), pp. 322–333, https://doi.org/10.1007/BF02165003.

[8] L. BOTTOU, F. E. CURTIS, AND J. NOCEDAL, *Optimization methods for large-scale machine learning*, SIAM Rev., 60 (2018), pp. 223–311, https://doi.org/10.1137/16M1080173.

[9] R. H. BYRD, P. LU, J. NOCEDAL, AND C. ZHU, *A limited memory algorithm for bound constrained optimization*, SIAM J. Sci. Comput., 16 (1995), pp. 1190–1208, https://doi.org/10.1137/0916069.

[10] Z. CAI, J. CHEN, AND M. LIU, *Least-squares ReLU neural network (LSNN) method for linear advection-reaction equation*, J. Comput. Phys., 443 (2021), 110514, https://doi.org/10.1016/j.jcp.2021.110514.

[11] Z. CAI AND S. KIM, *A finite element method using singular functions for the Poisson equation: Corner singularities*, SIAM J. Numer. Anal., 39 (2001), pp. 286–299, https://doi.org/10.1137/S0036142999355945.

[12] L. CATTANEO AND P. ZUNINO, *A computational model of drug delivery through microcirculation to compare different tumor treatments*, Int. J. Numer. Methods Biomed. Eng., 30 (2014), pp. 1347–1371.

[13] M. COSTABEL AND M. DAUGE, *A singularly perturbed mixed boundary value problem*, Comm. Partial Differential Equations, 21 (1996), pp. 1919–1949, https://doi.org/10.1080/03605309608821249.

[14] S. CUOMO, V. SCHIANO DI COLA, F. GIAMPAOLO, G. ROZZA, M. RAISSI, AND F. PICCIALLI, *Scientific machine learning through physics-informed neural networks: Where we are and what's next*, J. Sci. Comput., 92 (2022), 88, https://doi.org/10.1007/s10915-022-01939-z.

[15] E. C. CYR, M. A. GULIAN, R. G. PATEL, M. PEREGO, AND N. A. TRASK, *Robust training and initialization of deep neural networks: An adaptive basis viewpoint*, in Proceedings of the First Mathematical and Scientific Machine Learning Conference, Proc. Mach. Learn. Res. 107, J. Lu and R. Ward, eds., 2020, pp. 512–536, https://proceedings.mlr.press/v107/cyr20a.html.

[16] C. D'ANGELO, *Finite element approximation of elliptic problems with Dirac measure terms in weighted spaces: Applications to one- and three-dimensional coupled problems*, SIAM J. Numer. Anal., 50 (2012), pp. 194–215, https://doi.org/10.1137/100813853.

[17] C. D'ANGELO AND A. QUARTERONI, *On the coupling of 1d and 3d diffusion-reaction equations: Application to tissue perfusion problems*, Math. Models Methods Appl. Sci., 18 (2008), pp. 1481–1504.

[18] R. DEVORE, B. HANIN, AND G. PETROVA, *Neural network approximation*, Acta Numer., 30 (2021), pp. 327–444, https://doi.org/10.1017/S0962492921000052.

[19] C. DUAN, Y. JIAO, Y. LAI, D. LI, X. LU, AND J. Z. YANG, *Convergence rate analysis for deep Ritz method*, Commun. Comput. Phys., 31 (2022), pp. 1020–1048, https://doi.org/10.4208/cicp.oa-2021-0195.

[20] C. DUAN, Y. JIAO, Y. LAI, X. LU, Q. QUAN, AND J. Z. YANG, *Deep Ritz methods for Laplace equations with Dirichlet boundary condition*, CSIAM Trans. Appl. Math., 3 (2022), pp. 761–791, https://doi.org/10.4208/csiam-am.so-2021-0043.

[21] W. E, J. HAN, AND A. JENTZEN, *Algorithms for solving high dimensional PDEs: From nonlinear Monte Carlo to machine learning*, Nonlinearity, 35 (2022), pp. 278–310, https://doi.org/10.1088/1361-6544/ac337f.

[22] B. ENGQUIST, A.-K. TORNBERG, AND R. TSAI, *Discretization of Dirac delta functions in level set methods*, J. Comput. Phys., 207 (2005), pp. 28–51, https://doi.org/10.1016/j.jcp.2004.09.018.

[23] A. ERN AND J.-L. GUERMOND, *Theory and Practice of Finite Elements*, Springer-Verlag, New York, 2004, https://doi.org/10.1007/978-1-4757-4355-5.

[24] L. C. EVANS, *Partial Differential Equations*, 2nd ed., American Mathematical Society, Providence, RI, 2010, https://doi.org/10.1090/gsm/019.

[25] G. J. FIX, S. GULATI, AND G. I. WAKOFF, *On the use of singular functions with finite element approximations*, J. Comput. Phys., 13 (1973), pp. 209–228, https://doi.org/10.1016/0021-9991(73)90023-5.

[26] T.-P. FRIES AND T. BELYTSCHKO, *The extended/generalized finite element method: An overview of the method and its applications*, Internat. J. Numer. Methods Engrg., 84 (2010), pp. 253–304, https://doi.org/10.1002/nme.2914.

[27] I. G. Gjerde, K. Kumar, and J. M. Nordbotten, *A mixed approach to the Poisson problem with line sources*, SIAM J. Numer. Anal., 59 (2021), pp. 1117–1139, https://doi.org/10.1137/19M1296549.

[28] I. G. Gjerde, K. Kumar, J. M. Nordbotten, and B. Wohlmuth, *Splitting method for elliptic equations with line sources*, ESAIM Math. Model. Numer. Anal., 53 (2019), pp. 1715–1739.

[29] I. S. Gradshteyn and I. M. Ryzhik, *Table of Integrals, Series, and Products*, 8th ed., Elsevier/Academic Press, Amsterdam, 2015.

[30] L. Grinberg, E. Cheever, T. Anor, J. R. Madsen, and G. Karniadakis, *Modeling blood flow circulation in intracranial arterial networks: A comparative 3d/1d simulation study*, Ann. Biomed. Eng., 39 (2011), pp. 297–309.

[31] M. Grüter and K.-O. Widman, *The Green function for uniformly elliptic equations*, Manuscripta Math., 37 (1982), pp. 303–342, https://doi.org/10.1007/BF01166225.

[32] I. Guhring and M. Raslan, *Approximation rates for neural networks with encodable weights in smoothness spaces*, Neural Netw., 134 (2021), pp. 107–130.

[33] Q. Hong, J. W. Siegel, and J. Xu, *A Priori Analysis of Stable Neural Network Solutions to Numerical PDEs*, preprint, arXiv:2104.02903, 2021.

[34] B. Hosseini, N. Nigam, and J. M. Stockie, *On regularizations of the Dirac delta distribution*, J. Comput. Phys., 305 (2016), pp. 423–447.

[35] X. Huang, H. Liu, B. Shi, Z. Wang, K. Yang, Y. Li, B. Weng, M. Wang, H. Chu, J. Zhou, F. Yu, B. Hua, B. Dong, and L. Chen, *Solving partial differential equations with point source based on physics-informed neural networks*, in Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence (IJCAI-22), 2022, pp. 3839–3846.

[36] Y. Jiao, B. Jin, and X. Lu, *A primal dual active set with continuation algorithm for the ℓ^0-regularized optimization problem*, Appl. Comput. Harmon. Anal., 39 (2015), pp. 400–426, https://doi.org/10.1016/j.acha.2014.10.001.

[37] Y. Jiao, Y. Lai, Y. Lo, Y. Wang, and Y. Yang, *Error Analysis of Deep Ritz Methods for Elliptic Equations*, preprint, arXiv:2107.14478, 2021.

[38] B. Jin, X. Li, and X. Lu, *Imaging conductivity from current density magnitude using neural networks*, Inverse Problems, 38 (2022), 075003, https://doi.org/10.1088/1361-6420/ac6d03.

[39] G. E. Karniadakis, I. G. Kevrekidis, L. Lu, P. Perdikaris, S. Wang, and L. Yang, *Physics-informed machine learning*, Nature Rev. Phys., 3 (2021), pp. 422–440.

[40] E. Kharazmi, Z. Zhang, and G. E. M. Karniadakis, *hp-VPINNs: Variational physics-informed neural networks with domain decomposition*, Comput. Methods Appl. Mech. Engrg., 374 (2021), 113547, https://doi.org/10.1016/j.cma.2020.113547.

[41] T. Köppl and B. Wohlmuth, *Optimal a priori error estimates for an elliptic problem with Dirac right-hand side*, SIAM J. Numer. Anal., 52 (2014), pp. 1753–1769, https://doi.org/10.1137/130927619.

[42] A. Krishnapriyan, A. Gholami, S. Zhe, R. Kirby, and M. W. Mahoney, *Characterizing possible failure modes in physics-informed neural networks*, in Proceedings of the Thirty-Fifth Annual Conference on Neural Information Processing Systems (NeurIPS 2021), 2021.

[43] I. E. Lagaris, A. Likas, and D. I. Fotiadis, *Artificial neural networks for solving ordinary and partial differential equations*, IEEE Trans. Neural Netw., 9 (1998), pp. 987–1000.

[44] Y. LeCun, L. Bottou, G. B. Orr, and K.-R. Müller, *Efficient BackProp*, in Neural Networks: Tricks of the Trade, G. Montavon, G. Orr, and K. Müller, eds., Springer, Heidelberg, 1998, pp. 9–48.

[45] Y. Liu and Y. Mori, *Properties of discrete delta functions and local convergence of the immersed boundary method*, SIAM J. Numer. Anal., 50 (2012), pp. 2986–3015, https://doi.org/10.1137/110836699.

[46] Y. Lu, H. Chen, J. Lu, L. Ying, and J. Blanchet, *Machine Learning for Elliptic PDEs: Fast Rate Generalization Bound, Neural Scaling Law and Minimax Optimality*, preprint, arXiv:2110.06897, 2021.

[47] Y. Lu, J. Lu, and M. Wang, *A priori generalization analysis of the deep Ritz method for solving high dimensional elliptic partial differential equations*, in Proceedings of the Conference on Learning Theory, 2021, pp. 3196–3241.

[48] L. McClenny and U. Braga-Neto, *Self-adaptive Physics-Informed Neural Networks Using a Soft Attention Mechanism*, preprint, https://arXiv:2009.04544, 2020.

[49] S. Moore, K. Moorhead, J. Chase, T. David, and J. Fink, *One-dimensional and three-dimensional models of cerebrovascular flow*, J. Biomech. Eng., 127 (2005), pp. 440–449.

[50] B. Moseley, A. Markham, and T. Nissen-Meyer, *Solving the Wave Equation with Physics-Informed Deep Learning*, preprint, https://arXiv:2006.11894, 2020.

[51] J. MÜLLER AND M. ZEINHOFER, *Error estimates for the deep Ritz method with boundary penalty*, in Proceedings of the 3rd Annual Conference on Mathematical and Scientific Machine Learning, 2022.

[52] J. MÜLLER AND M. ZEINHOFER, *Notes on exact boundary values in residual minimisation*, in Proceedings of the 3rd Annual Conference on Mathematical and Scientific Machine Learning, 2022.

[53] D. PEACEMAN, *Interpretation of well-block pressures in numerical reservoir simulation*, SPE J., 18 (1978), pp. 183–194, https://doi.org/10.2118/6893-PA.

[54] G. C. Y. PENG, M. ALBER, A. BUGANZA TEPOLE, W. R. CANNON, S. DE, S. DURA-BERNAL, K. GARIKIPATI, G. KARNIADAKIS, W. W. LYTTON, P. PERDIKARIS, L. PETZOLD, AND E. KUHL, *Multiscale modeling meets machine learning: What can we learn?*, Arch. Comput. Methods Eng., 28 (2021), pp. 1017–1037, https://doi.org/10.1007/s11831-020-09405-5.

[55] 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), pp. 686–707.

[56] M. RASHT-BEHESHT, C. HUBER, K. SHUKLA, AND G. E. KARNIADAKIS, *Physics-informed neural networks (pinns) for wave propagation and full waveform inversions*, J. Geophys. Res. Solid Earth, 127 (2022), e2021JB023120.

[57] J. REICHOLD, M. STAMPANONI, A. L. KELLER, A. BUCK, P. JENNY, AND B. WEBER, *Vascular graph model to simulate the cerebral blood flow in realistic vascular networks*, J. Cereb. Blood Flow Metabol., 29 (2009), pp. 1429–1443.

[58] S. REZAEI, A. HARANDI, A. MOEINEDDIN, B.-X. XU, AND S. REESE, *A mixed formulation for physics-informed neural networks as a potential solver for engineering problems in heterogeneous domains: Comparison with finite element method*, Comput. Methods Appl. Mech. Engrg., 401 (2022), 115616, https://doi.org/10.1016/j.cma.2022.115616.

[59] R. SCOTT, *Finite element convergence for singular data*, Numer. Math., 74 (1973), pp. 317–327, https://doi.org/10.1007/BF01436386.

[60] S. SHALEV-SHWARTZ AND S. BEN-DAVID, *Understanding Machine Learning: From Theory to Algorithms*, Cambridge University Press, New York, 2014.

[61] J. SIRIGNANO AND K. SPILIOPOULOS, *DGM: A deep learning algorithm for solving partial differential equations*, J. Comput. Phys., 375 (2018), pp. 1339–1364.

[62] D. SOUDRY, E. HOFFER, M. S. NACSON, S. GUNASEKAR, AND N. SREBRO, *The implicit bias of gradient descent on separable data*, J. Mach. Learn. Res., 19 (2018), 70.

[63] D. M. SULLIVAN, *Electromagnetic Simulation Using the FDTD Method*, John Wiley & Sons, Hoboken, NJ, 2013.

[64] J. SYLVESTER AND G. UHLMANN, *Inverse boundary value problems at the boundary—continuous dependence*, Comm. Pure Appl. Math., 41 (1988), pp. 197–219, https://doi.org/10.1002/cpa.3160410205.

[65] K. TANG, X. WAN, AND C. YANG, *DAS-PINNs: A deep adaptive sampling method for solving high-dimensional partial differential equations*, J. Comput. Phys., 476 (2022), 111868.

[66] G. UHLMANN, *Electrical impedance tomography and Calderón's problem*, Inverse Problems, 25 (2009), 123011, https://doi.org/10.1088/0266-5611/25/12/123011.

[67] S. WANG, Y. TENG, AND P. PERDIKARIS, *Understanding and mitigating gradient flow pathologies in physics-informed neural networks*, SIAM J. Sci. Comput., 43 (2021), pp. A3055–A3081, https://doi.org/10.1137/20M1318043.

[68] S. WANG, X. YU, AND P. PERDIKARIS, *When and why PINNs fail to train: A neural tangent kernel perspective*, J. Comput. Phys., 449 (2022), 110768, https://doi.org/10.1016/j.jcp.2021.110768.

[69] C. WU, M. ZHU, Q. TAN, Y. KARTHA, AND L. LU, *A comprehensive study of non-adaptive and residual-based adaptive sampling for physics-informed neural networks*, Comput. Methods Appl. Mech. Engrg., 403 (2023), 115671, https://doi.org/10.1016/j.cma.2022.115671.

[70] D. YAROTSKY, *Error bounds for approximations with deep ReLU networks*, Neural Netw., 94 (2017), pp. 103–114.

[71] B. YU AND W. E., *The deep Ritz method: A deep learning-based numerical algorithm for solving variational problems*, Commun. Math. Stat., 6 (2018), pp. 1–12.

[72] Y. ZANG, G. BAO, X. YE, AND H. ZHOU, *Weak adversarial networks for high-dimensional partial differential equations*, J. Comput. Phys., 411 (2020), 109409, https://doi.org/10.1016/j.jcp.2020.109409.