Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Seminar for Applied Mathematics SAM

ReLU Neural Network Galerkin BEM

R. Aylwin and F. Henriquez and Ch. Schwab

Research Report No. 2022-01

January 2022
Latest revision: February 2023

Seminar für Angewandte Mathematik
Eidgenössische Technische Hochschule
CH-8092 Zürich
Switzerland

RELU NEURAL NETWORK GALERKIN BEM

RUBÉN AYLWIN[1], FERNANDO HENRÍQUEZ[2], AND CHRISTOPH SCHWAB[3]

Dedicated to Professor Wolfgang L. Wendland on the Occasion of his 85th Birthday

Abstract. We introduce Neural Network (NN for short) approximation architectures for the numerical solution of Boundary Integral Equations (BIEs for short). We exemplify the proposed NN approach for the boundary reduction of the potential problem in two spatial dimensions. We adopt a Galerkin formulation-based method, in polygonal domains with a finite number of straight sides. Trial spaces used in the Galerkin discretization of the BIEs are built by using NNs that, in turn, employ the so-called Rectified Linear Units (ReLU) as the underlying *activation function*. The ReLU-NNs used to approximate the solutions to the BIEs depend nonlinearly on the parameters characterizing the NNs themselves. Consequently, the computation of a numerical solution to a BIE by means of ReLU-NNs boils down to a fine tuning of these parameters, in *network training*.

We argue that ReLU-NNs of fixed depth and with a variable width allow us to recover well-known approximation rate results for the standard Galerkin Boundary Element Method (BEM). This observation hinges on existing well-known properties concerning the regularity of the solution of the BIEs on Lipschitz, polygonal boundaries, i.e. accounting for the effect of corner singularities, and the *expressive* power of ReLU-NNs over different classes of functions. We prove that *shallow* ReLU-NNs, i.e. networks having a fixed, moderate depth but with increasing width, can achieve optimal order algebraic convergence rates.

We propose novel loss functions for NN training which are obtained using *computable, local residual a posteriori error estimators* with ReLU-NNs for the numerical approximation of BIEs. We find that weighted residual estimators, which are reliable without further assumptions on the quasi-uniformity of the underlying mesh, can be employed for the construction of computationally efficient loss functions for ReLU-NN training. The proposed framework allows us to leverage on state-of-the-art computational deep learning technologies such as TENSORFLOW and TPUs for the numerical solution of BIEs using ReLU-NNs. Exploratory numerical experiments validate our theoretical findings and indicate the viability of the proposed ReLU-NN Galerkin BEM approach.

[1]Faculty of Engineering and Sciences, Universidad Adolfo Ibañez, Santiago, Chile.

[2]Chair of Computational Mathematics and Simulation Science (MCSS), École Polytechnique Fédéral de Lausanne, Lausanne, Switzerland.

[3]Seminar for Applied Mathematics, ETH Zu rich, 8092 Zürich, Switzerland.

E-mail addresses: fernando.henriquez@epfl.ch, ruben.aylwin@uai.cl, schwab@math.ethz.ch.

Contents

1. Introduction

Following fundamental mathematical developments in the 90's that established density (resp. "universal approximation capacity") of shallow NNs (see, e.g., the surveys [28] and the references therein), *Deep Neural Networks* (DNNs for short) based computations have seen, during the past five to ten years, an increasing development driven by the advent of data science and deep learning techniques. In these developments, in particular the quantitative advantages in expressive power furnished by deep NNs has moved into the focus of interest.

Mounting computational and theoretical evidence points to significant advantages of the expressive power of DNNs when furnished by possibly large NN depth. We mention only [36, 29] for the use of DNNs in the numerical solution of PDEs, and the recent works [27, 32] that indicate a scope for the efficacy of DNNs in the approximation of maps between high-dimensional spaces.

In the present work, we propose the use of NNs for the approximation of *variational* BIEs, in particular of the first kind. We focus on NNs with ReLU activation, of fixed depth. As we shall show, mathematically and computationally, these ReLU-NNs allow for optimal convergence rates of the Galerkin BEM in polygons. We also comment in passing on advantages afforded by NNs with large depth. Here, approximation theory indicates that exponential convergence is possible, in principle.

Contributions. We consider the Laplace equation with appropriate boundary conditions on $\mathbb{R}^2 \setminus \Gamma$, where Γ corresponds to a one dimensional open arc, and on a polygonal Lipschitz domain D, and consider their boundary reduction by means of BIOs as described, for example, in [31]. We prove that by using *shallow* ReLU-NNs as trial spaces in the Galerkin discretization of the resulting BIEs one can recover well-known algebraic convergence rate bounds of the BEM, which are in turn traditionally obtained by means of other methods, such as graded meshes or the adaptive BEM [15, 9] (ABEM). Even though not thoroughly studied in the present work, we remark that by using Deep ReLU-NNs (and working under the assumption of analytic regularity hypotheses on the data) one can recover exponential convergence rates usually obtained using the so-called *hp* Galerkin BEM. This can be obtained by recalling recently obtained ReLU-NN emulation results from [27] together with analytic regularity results of the solution to BIEs in polygonal Lipschitz domain D, with a finite number of straight sides, in weighted function spaces [3]. However, we hasten to add that the main goal of the present work is to study the approximation of the solution to BIEs by means of shallow ReLU-NNs.

The insight behind the results present herein is the interpretation that shallow ReLU-NNs realize approximations in *linear subspaces*, whose basis elements are realized by the so-called "hidden layers" of the ReLU-NN and are, therefore, *subject to optimization during ReLU-NN training*. In the present paper, we *partially* leverage this flexibility and the ability of the hidden layers of ReLU-NNs to express: (a) low order boundary element spaces with "free-knots" (in the terminology of spline approximation), i.e., to adapt the partitions of the boundary to the structure of the unknown solution (which is reminiscent of adaptive meshing strategies in Galerkin BEM), and (b) to leverage adaptive mesh refinement methods in BEM for a rational procedure to "enlarge" the ReLU-NN through the insertion of nodes.

We propose two different algorithms to, computationally, perform the construction of the ReLU-NNs (thus, two different paths to train the network.) The first one is based in the observation that the solution of symmetric, coercive problems, such as the ones arising from the boundary reduction of the Laplace problem, can be written as the minimization of a suitable energy functional.

The second algorithm makes use of the well-known a posteriori error estimates, which are commonly used in the ABEM. The definition of an efficiently computable loss function which is based on computable, reliable a-posteriori error estimators for Galerkin discretizations differs from other widely established methods, and is not limited to Galerkin BEM. Numerical experiments show the computational feasibility of these algorithms and a detailed convergence analysis shows that shallow ReLU-NN Galerkin BEM can attain the optimal algebraic convergence rates. NN training can, in particular, compensate for reduced convergence rates due to, e.g., corner singularities of the physical domain.

Outline. This work is structured as follows. In Section 2 we briefly review the boundary reduction of potential problems in two space dimensions. We also present a short recap of the direct method of boundary reduction and the strong ellipticity of boundary integral operators of the first-kind in fractional order Sobolev spaces. Additionally, we recall results concerning the regularity of the solution to BIEs both in Lipschitz polygons and on open arcs, which will be used ahead to obtain convergence rates for the ReLU-NN approximation of BIEs.

We begin Section 3 by introducing rigorous definitions of DNNs and describing, in detail, the connection existing between $\mathbb{P}_1$-spline boundary element spaces and ReLU-NNs. A key observation in our

analysis is that the low-order BEM spaces with adaptive mesh refinement is shown to admit a representation through ReLU-NNs architectures. This property, together with the regularity results of the solutions to BIEs, allows us to establish convergence results with optimal rates in the approximation of the described BIEs by means of ReLU-NNs. These theoretical results provide a benchmark to asses the performance of the different training algorithms proposed ahead in Section 4.

In Section 4, we propose two concrete training algorithms for the construction of ReLU-NNs approximating the solution of the analyzed BIEs. The first one leverages on the coercivity properties of the BIEs for the potential problem and the interpretation of its solution as the minimizer of a suitable energy functional. By performing a combination of the optimization of hidden parameters of the ReLU-NN and computation of the output layer by solving the corresponding Galerkin linear system, the proposed algorithm attains the (theoretically proven) optimal convergence rates for different numerical test cases. The second algorithm proposed in Section 4 hinges on *a posteriori* error estimates which, in turn, provide a computable upper bound for the error between the exact solution to the BIE and the ReLU-NN. Consequently, we can use this property to guide the training of the ReLU-NNs at each step of the iterative process. We mention that the proposed numerical methodology does not rely on automatic differentiation of the loss function with respect to the DNN parameters. Section 5 then presents concrete numerical experiments and is followed by Section 6, which recapitulates our principal findings and indicates several lines of investigation for their extension and possible further mathematical results.

2. Preliminaries

2.1. **Sobolev Spaces in Bounded Domains.** Let $\mathrm{D} \subset \mathbb{R}^2$ be a bounded, connected Lipschitz domain with boundary $\partial\mathrm{D}$. For $s \in \mathbb{R}$, we denote by $H^s(\mathrm{D})$ the standard Sobolev spaces of order s defined in D equipped with the norm $\|\cdot\|_{H^s(\mathrm{D})}$. As it is customary, we identify $H^0(\mathrm{D})$ with $L^2(\mathrm{D})$. Sobolev spaces on the boundary $\Gamma := \partial\mathrm{D}$ are denoted by $H^s(\Gamma)$, where the range of $s \in \mathbb{R}$ is restricted in accordance to the regularity of the domain D (*cf.* [26, Sections 3.8 and 3.11]). For Lipschitz domains, for example, the space $H^s(\Gamma)$ is well-defined for $s \in [-1, 1]$. Furthermore, we shall denote by $H^s(\mathrm{D})/\mathbb{R}$ (resp. $H^s(\Gamma)/\mathbb{R}$) the quotient space of $H^s(\mathrm{D})$ (resp. $H^s(\Gamma)$) by the subspace $\mathrm{span}\{1\}$. We denote by $\gamma_0 : H^1(\mathrm{D}) \to H^{\frac{1}{2}}(\Gamma)$ and $\gamma_1 : H^1(\Delta, \mathrm{D}) \to H^{-\frac{1}{2}}(\Gamma)$ the Dirichlet and Neumann trace operators on Γ, respectively, with $H^1(\Delta, \mathrm{D}) := \{u \in H^1(\mathrm{D}) : \Delta u \in L^2(\mathrm{D})\}$. The duality between $H^s(\Gamma)$ and $H^{-s}(\Gamma)$ is denoted by $\langle \psi, \phi \rangle_\Gamma$ for $\psi \in H^{-s}(\Gamma)$ and $\phi \in H^s(\Gamma)$.

2.2. **Sobolev Spaces on Open Arcs.** Let $\Lambda \subset \mathbb{R}^2$ be a Jordan arc (in the sense of [30, Definition 2.4.2]). For $s \in (-1, 1)$, we denote by $H^s(\Lambda)$ the standard Sobolev space on Λ and, furthermore, introduce the spaces $\widetilde{H}^s(\Lambda)$ as in [26, Section 3.6]. Moreover, the following relations hold (*cf.* [26, Section 3.11])

$$\widetilde{H}^{-s}(\Lambda) \equiv (H^s(\Lambda))' \quad \text{and} \quad H^{-s}(\Lambda) \equiv \left(\widetilde{H}^s(\Lambda)\right)',$$

where, for a general Banach space X, X' denotes its dual space. The duality between $H^s(\Lambda)$ and $\widetilde{H}^{-s}(\Lambda)$ is denoted by $\langle \phi, \psi \rangle_\Lambda$ for $\psi \in \widetilde{H}^{-s}(\Lambda)$ and $\phi \in H^s(\Lambda)$.

2.3. **Boundary Integral Operators in Lipschitz Domains.** Again, let $\mathrm{D} \subset \mathbb{R}^2$ be a bounded Lipschitz domain with boundary $\Gamma := \partial\mathrm{D}$. In the following, we introduce the main concepts concerning BIOs and BIEs to be used throughout this manuscript. As we will only discuss boundary value problems in $\mathbb{R}^2$, we limit our presentation of the aforementioned tools to the two dimensional case. Let $\mathrm{G}(\mathbf{x}, \mathbf{y})$ be the fundamental solution of the Laplacian in $\mathbb{R}^2$ (*cf.* [34, Chapter 5] or [31, Section 3.1]), given by

$$\mathrm{G}(\mathbf{x}, \mathbf{y}) = -\frac{1}{2\pi} \log \|\mathbf{x} - \mathbf{y}\|, \quad \mathbf{x}, \mathbf{y} \in \mathbb{R}^2, \quad \mathbf{x} \neq \mathbf{y}.$$

Let $\mathcal{N} : \widetilde{H}^{-1}(\mathrm{D}) \to H^1(\mathrm{D})$ denote the *Newton* potential, defined for $\varphi \in \mathscr{C}_0^\infty(\mathrm{D})$ as

$$(\mathcal{N}\varphi)(\mathbf{x}) := \int_{\mathrm{D}} \mathrm{G}(\mathbf{x}, \mathbf{y}) \varphi(\mathbf{y}) d\mathbf{y}, \quad \mathbf{x} \in \mathrm{D}.$$

We define the *single* and *double* layer potentials, respectively, as follows:

$$\mathcal{S} := \mathcal{N} \circ \gamma_0' \quad \text{and} \quad \mathcal{D} := \mathcal{N} \circ \gamma_1'. \tag{2.1}$$

The operators $\mathcal{S} : H^{-\frac{1}{2}}(\Gamma) \to H^1(\mathrm{D})$ and $\mathcal{D} : H^{\frac{1}{2}}(\Gamma) \to H^1(\mathrm{D})$ define continuous mappings (*cf.* [31, Theorem 3.1.12 & Theorem 3.1.16]). Equipped with these definitions, we introduce the *boundary integral*

operators (BIOs) on Γ as:

$$\mathsf{V} \coloneqq \gamma_0 \mathcal{S}, \qquad \mathsf{K} \coloneqq \frac{1}{2}\mathsf{Id} + \gamma_0 \mathcal{D}, \qquad \mathsf{K}' \coloneqq -\frac{1}{2}\mathsf{Id} + \gamma_1 \mathcal{S}, \qquad \mathsf{W} \coloneqq -\gamma_1 \mathcal{D}, \tag{2.2}$$

referred to as the *single layer*, *double layer*, *adjoint double layer* and *hypersingular* BIOs, respectively, where Id signifies the identity operator. The mapping properties of the BIOs in (2.2) are stated in the following result.

Proposition 2.1 ([10, Theorem 1])**.** *Let* $\mathrm{D} \subset \mathbb{R}^2$ *be a bounded Lipschitz domain with boundary* $\Gamma \coloneqq \partial\mathrm{D}$*. For each* $\sigma \in (-\frac{1}{2}, \frac{1}{2})$*, the BIOs introduced in* (2.2) *define continuous maps according to:*

$$\mathsf{V} : H^{-\frac{1}{2}+\sigma}(\Gamma) \to H^{\frac{1}{2}+\sigma}(\Gamma), \qquad \mathsf{K} : H^{\frac{1}{2}+\sigma}(\Gamma) \to H^{\frac{1}{2}+\sigma}(\Gamma),$$
$$\mathsf{K}' : H^{-\frac{1}{2}+\sigma}(\Gamma) \to H^{-\frac{1}{2}+\sigma}(\Gamma), \qquad \mathsf{W} : H^{\frac{1}{2}+\sigma}(\Gamma) \to H^{-\frac{1}{2}+\sigma}(\Gamma).$$

Moreover, the single layer and hypersingular BIOs are coercive.

Proposition 2.2 ([34, Theorems 6.23 & Corollary 6.25])**.** *Let* $\mathrm{D} \subset \mathbb{R}^2$ *be a bounded Lipschitz domain with boundary* $\Gamma \coloneqq \partial\mathrm{D}$*. There exists* $c_{\mathsf{W}} > 0$ *such that*

$$\langle \mathsf{W}\phi, \phi \rangle_\Gamma \geq c_{\mathsf{W}} \left\| \phi \right\|^2_{H^{\frac{1}{2}}(\Gamma)}, \quad \forall \phi \in H^{\frac{1}{2}}(\Gamma)/\mathbb{R}.$$

Assume that $\operatorname{diam}(\mathrm{D}) < 1$*, then there exists* $c_{\mathsf{V}} > 0$ *such that*

$$\langle \mathsf{V}\psi, \psi \rangle_\Gamma \geq c_{\mathsf{V}} \left\| \psi \right\|^2_{H^{-\frac{1}{2}}(\Gamma)}, \quad \forall \psi \in H^{-\frac{1}{2}}(\Gamma).$$

Remark 1. Let us define the bilinear form $\check{\mathsf{a}} : H^{\frac{1}{2}}(\Gamma) \times H^{\frac{1}{2}}(\Gamma) \to \mathbb{R}$ as

$$\check{\mathsf{a}}(\phi, \psi) \coloneqq \langle \mathsf{W}\phi, \psi \rangle_\Gamma + \langle \phi, 1 \rangle_\Gamma \langle \psi, 1 \rangle_\Gamma, \quad \forall \phi, \psi \in H^{\frac{1}{2}}(\Gamma). \tag{2.3}$$

One can also prove (see, e.g., [37, Section 2] and the references therein) that there exists a constant $\alpha > 0$ such that

$$\check{\mathsf{a}}(\phi, \phi) \geq \alpha \left\| \phi \right\|^2_{H^{\frac{1}{2}}(\Gamma)}, \quad \forall\, \phi \in H^{\frac{1}{2}}(\Gamma).$$

Lemma 2.3 (Maue's formula, [20, Lemma 1.2.2] & [34, Theorem 6.15])**.** *Let* $\mathrm{D} \subset \mathbb{R}^2$ *be a bounded Lipschitz polygon with boundary* $\Gamma \coloneqq \partial\mathrm{D}$*. Then, for all* $\varphi \in \mathscr{C}^0(\Gamma)$ *with continuous derivative on each smooth segment of* Γ *it holds that*

$$\mathsf{W}\varphi = -\frac{d}{ds}\mathsf{V}\frac{d}{ds}\varphi \quad \textit{in} \quad H^{-\frac{1}{2}}(\Gamma), \tag{2.4}$$

where $\frac{d}{ds}$ *denotes the arc-length derivative. The relation* (2.4) *remains valid for* $\varphi \in H^{\frac{1}{2}}(\Gamma)$*.*

2.3.1. *Direct Boundary Integral Formulation of the interior Dirichlet BVP.* We consider the interior Laplace problem equipped with Dirichlet boundary conditions in a bounded Lipschitz domain $\mathrm{D} \subset \mathbb{R}^2$ with boundary $\Gamma \coloneqq \partial\mathrm{D}$.

Problem 2.4 (Dirichlet Boundary Value Problem)**.** *Let* $f \in H^{\frac{1}{2}}(\Gamma)$ *be given. We seek* $u \in H^1(\mathrm{D})$ *satisfying*

$$-\Delta u = 0 \quad \textit{in } \mathrm{D} \qquad \textit{and} \qquad \gamma_0 u = f \quad \text{on } \Gamma.$$

As is customary, one may recast Problem 2.4 as an equivalent BIE using the BIOs introduced in (2.2). The starting point is the so-called *integral representation formula*: we express the weak solution $u \in H^1(\mathrm{D})$ to Problem 2.4, using the layer potentials introduced in (2.1) as follows:

$$u = \mathcal{S}(\gamma_1 u) - \mathcal{D}(\gamma_0 u) \quad \text{in } \mathrm{D}. \tag{2.6}$$

By applying the Dirichlet trace operator $\gamma_0 : H^1(\mathrm{D}) \to H^{\frac{1}{2}}(\Gamma)$ to (2.6) and using the boundary condition on Γ stated in Problem 2.4, one obtains the following BIE for the unknown datum $\gamma_1 u \in H^{-\frac{1}{2}}(\Gamma)$.

Problem 2.5 (Boundary Integral Formulation of Problem 2.4)**.** *Let* $f \in H^{\frac{1}{2}}(\Gamma)$ *be given. We seek* $\psi \coloneqq \gamma_1 u \in H^{-\frac{1}{2}}(\Gamma)$ *such that*

$$\mathsf{V}\psi = \left(\frac{1}{2}\mathsf{Id} + \mathsf{K}\right) f.$$

The well-posedness of the BIE in Problem 2.5 follows from the mapping properties of the BIOs in Proposition 2.1, the ellipticity of the single layer BIO $\mathsf{V} : H^{-\frac{1}{2}}(\Gamma) \to H^{\frac{1}{2}}(\Gamma)$ in Proposition 2.2, and the Lax-Milgram lemma.

2.3.2. *Direct Boundary Integral Formulation of the Interior Neumann BVP.* We consider the interior Laplace problem equipped with nonhomogeneous Neumann boundary conditions in a bounded, simply connected Lipschitz domain $\mathrm{D} \subset \mathbb{R}^2$ with boundary $\Gamma := \partial\mathrm{D}$, thus rendering the boundary Γ connected itself.

Problem 2.6 (Neumann BVP in D)**.** *Let $g \in H^{-\frac{1}{2}}(\Gamma)$ be given. We seek $u \in H^1(\mathrm{D})$ satisfying*

$$\Delta u = 0 \quad \text{in } \mathrm{D} \qquad \text{and} \qquad \gamma_1 u = g \quad \text{on } \Gamma.$$

It is well-established that there exists a unique $u \in H^1(\mathrm{D})/\mathbb{R}$ solution to Problem 2.6 provided that one further imposes $g \in H^{-\frac{1}{2}}(\Gamma)/\mathbb{R}$ (see, e.g., [34, Theorem 4.9]). We reduce the boundary value problem stated in Problem 2.6 to an equivalent BIE using the BIOs introduced in (2.2) and the integral representation formula (2.6). The application of the Neumann trace operator $\gamma_1 : H^1(\Delta, \mathrm{D}) \to H^{-\frac{1}{2}}(\Gamma)$ to (2.6) together with boundary condition stated in Problem 2.6 yields the following boundary integral formulation of Problem 2.6.

Problem 2.7 (Boundary Integral Formulation of Problem 2.6)**.** *Let $g \in H^{-\frac{1}{2}}(\Gamma)/\mathbb{R}$ be given. We seek $\phi := \gamma_0 u \in H^{\frac{1}{2}}(\Gamma)/\mathbb{R}$ such that*

$$\mathsf{W}\phi = \left(\frac{1}{2}\mathsf{Id} - \mathsf{K}'\right) g.$$

The well-posedness of Problem 2.7 follows from the mapping properties of the BIOs stated in Proposition 2.1, Proposition 2.2, the fact that K' preserves integral mean value zero for densities defined over Γ and from the Lax-Milgram lemma, as it is thoroughly explained in [34, Section 7.2].

2.4. **Regularity of the solution to the BIEs in Lipschitz Polygons.** We recapitulate results concerning the regularity of solutions to the BIEs stated in Problems 2.5 and 2.7. In the following, we assume that $\mathrm{D} \subset \mathbb{R}^2$ is a bounded Lipschitz polygon with boundary $\Gamma := \partial\mathrm{D}$ characterized by a finite number $J \geq 3$ of vertices $\{\mathbf{x}_j\}_{j=1}^J \subset \mathbb{R}^2$. We enumerate cyclically mod J, i.e., $\mathbf{x}_{J+1} = \mathbf{x}_1$. We denote $\Gamma_j = \mathrm{conv}(\mathbf{x}_j, \mathbf{x}_{j+1})$, for $j = 1, \ldots, J$, (with the convention $\Gamma_0 = \Gamma_J$) and let $\omega_j \in (0, 2\pi)$ be the internal angle at $\mathbf{x}_j$, i.e., that of the wedge formed by the edges Γ_{j-1} and Γ_j, $j = 1 \ldots, J$.

Let $\{\chi_j\}_{j=1}^J$ be a partition of unity of Γ, where each function χ_j, $j = 1, \ldots, J$, is constructed by considering the restriction of a function in $\mathscr{C}_0^2(\mathbb{R}^2)$ to the boundary Γ such that $\chi_j = 1$ in a neighborhood of $\mathbf{x}_j$ and $\mathrm{supp}\{\chi_j\} \subset \Gamma_{j-1} \cup \{\mathbf{x}_j\} \cup \Gamma_j$. Let $\varphi : \Gamma \to \mathbb{R}$ be a function defined on the boundary Γ. Using the previously described partition of unity, we may write

$$\varphi = \sum_{j=1}^{J} (\varphi_-, \varphi_+)\, \chi_j \quad \text{on } \Gamma,$$

where by (φ_-, φ_+) we denote the restriction of $\varphi : \Gamma \to \mathbb{R}$ to $\Gamma_{j-1} \cup \{\mathbf{x}_j\} \cup \Gamma_j$ with $\varphi_- := \varphi|_{\Gamma_{j-1}}$ and $\varphi_+ := \varphi|_{\Gamma_j}$, for $j = 1, \ldots, J$.

The following result (from [16]) describes the regularity of the solution to Problems 2.5 and 2.7.

Proposition 2.8 ([16, Proposition 2.1])**.** *Set $\alpha_{jk} := k\frac{\pi}{\omega_j}$, for $k \in \mathbb{N}$ and $j = 1, \ldots, J$. Let $t \geq 1/2$ and $n \in \mathbb{N}$ be such that $n + 1 \geq \frac{\omega_j}{\pi}\left(t - \frac{1}{2}\right) \geq n$ for all $j = 1, \ldots, J$.*

(i) Assume that $f \in H^{\frac{1}{2}}(\Gamma)$ in Problem 2.5 is additionally piecewise analytic. Then, there exists ψ_0 satisfying $\psi_0|_{\Gamma_j} \in H^{t-1}(\Gamma_j)$ for all $j = 1, \ldots, J$, such that the solution $\psi \in H^{-\frac{1}{2}}(\Gamma)$ to Problem 2.5 admits the following representation:

$$\psi = \sum_{j=1}^{J} \sum_{k=1}^{n} ((\psi_{jk})_-, (\psi_{jk})_+)\, \chi_j + \psi_0. \tag{2.7}$$

In (2.7), if $\alpha_{j,k} \notin \mathbb{Z}$

$$(\psi_{jk})_\pm = c_{jk}^\pm \left\|\mathbf{x} - \mathbf{x}_j\right\|^{\alpha_{jk}-1},$$

and if $\alpha_{j,k} \in \mathbb{Z}$

$$(\psi_{jk})_\pm = c_{jk}^\pm \left\|\mathbf{x} - \mathbf{x}_j\right\|^{\alpha_{jk}-1} + d_{jk}^\pm \left\|\mathbf{x} - \mathbf{x}_j\right\|^{\alpha_{jk}-1} \log\left\|\mathbf{x} - \mathbf{x}_j\right\|,$$

where $c_{jk}^\pm$, $d_{jk}^\pm \in \mathbb{R}$, for $j = 1, \ldots, J$ and $k = 1, \ldots, n$.

(ii) Assume that $g \in H^{-\frac{1}{2}}(\Gamma)/\mathbb{R}$ in Problem 2.7 is additionally piecewise analytic. Then, there exists ϕ_0 such that $\phi_0|_{\Gamma_j} \in H^t(\Gamma_j)$, for all $j = 1, \dots, J$, such that the solution $\phi \in H^{\frac{1}{2}}(\Gamma)$ to Problem 2.7 admits the following representation:

$$\phi = \sum_{j=1}^{J} \sum_{k=1}^{n} \left((\phi_{jk})_-, (\phi_{jk})_+ \right) \chi_j + \phi_0. \tag{2.8}$$

In (2.8), if $\alpha_{j,k} \notin \mathbb{Z}$

$$(\phi_{jk})_\pm = c_{jk}^\pm \left\| \mathbf{x} - \mathbf{x}_j \right\|^{\alpha_{jk}},$$

and if $\alpha_{j,k} \in \mathbb{Z}$

$$(\phi_{jk})_\pm = c_{jk}^\pm \left\| \mathbf{x} - \mathbf{x}_j \right\|^{\alpha_{jk}} + d_{jk}^\pm \left\| \mathbf{x} - \mathbf{x}_j \right\|^{\alpha_{jk}} \log \left\| \mathbf{x} - \mathbf{x}_j \right\|,$$

where $c_{jk}^\pm,\ d_{jk}^\pm \in \mathbb{R}$, for $j = 1, \dots, J$ and $k = 1, \dots, n$.

2.5. **Boundary Integral Operators on Open Arcs.** We proceed to extend the definitions and results introduced in Section 2.3 for BIOs in Lipschitz domains D with closed boundary $\Gamma = \partial \mathrm{D}$, i.e., $\partial\Gamma = \emptyset$, to open arcs Λ in $\mathbb{R}^2$ (in the sense of [30, Definition 2.4.2]), for which $\partial\Lambda \neq \emptyset$. To this end, we consider a bounded Lipschitz $\mathrm{D} \subset \mathbb{R}^2$ with boundary $\Gamma := \partial \mathrm{D}$ and assume that $\Lambda \subset \mathbb{R}^2$ is a connected open arc of Γ with positive measure and endpoints $\mathbf{x}_1$ and $\mathbf{x}_2$ in Γ. By [10, Theorem 1], the potentials in (2.1) are well defined as elements on local Sobolev spaces over the unbounded domain $\Lambda^{\mathrm{c}} := \mathbb{R}^2 \setminus \overline{\Lambda}$, so that the layer potentials

$$\mathcal{S} : \widetilde{H}^{-\frac{1}{2}}(\Lambda) \to H^1_{\mathrm{loc}}(\mathbb{R}^2) \quad \text{and} \quad \mathcal{D} : \widetilde{H}^{\frac{1}{2}}(\Lambda) \to H^1_{\mathrm{loc}}(\Lambda^{\mathrm{c}}),$$

define continuous operators. The continuity properties of the layer potentials and trace operators [26, Theorem 3.38] together with the jump properties of the layer potentials [10, Lemma 4.1] allow us to define the BIOs on Λ as before:

$$\mathsf{V} := \gamma_0 \mathcal{S} \quad \text{and} \quad \mathsf{W} := -\gamma_1 \mathcal{D}.$$

We recall key properties of the single layer and hypersingular BIOs on a open arc Λ.

Proposition 2.9 ([9, Lemmas 1 & 3])**.** *Let $\Lambda \subsetneq \Gamma$ be an open Jordan arc. For $|\sigma| \leq \frac{1}{2}$, the maps*

$$\mathsf{V} : \widetilde{H}^{-\frac{1}{2}+\sigma}(\Lambda) \to H^{\frac{1}{2}+\sigma}(\Lambda) \quad \textit{and} \quad \mathsf{W} : \widetilde{H}^{\frac{1}{2}+\sigma}(\Lambda) \to H^{-\frac{1}{2}+\sigma}(\Lambda)$$

are continuous.

Theorem 2.10 ([9, Section 2] and [35, Theorem 1.5])**.** *Let $\Lambda \subsetneq \Gamma$ be a Jordan arc. Then, there exist positive constants μ, η (depending upon Λ) such that for all $\psi \in \widetilde{H}^{-\frac{1}{2}}(\Lambda)$ and for all $\phi \in \widetilde{H}^{\frac{1}{2}}(\Lambda)$ it holds*

$$\langle \mathsf{V}\psi, \psi \rangle_\Lambda \geq \eta \left\| \psi \right\|^2_{\widetilde{H}^{-\frac{1}{2}}(\Lambda)}, \quad \langle \mathsf{W}\phi, \phi \rangle_\Lambda \geq \mu \left\| \phi \right\|^2_{\widetilde{H}^{\frac{1}{2}}(\Lambda)}.$$

Problem 2.11 (Weakly singular BIE on Λ)**.** *Let $f \in H^{\frac{1}{2}}(\Lambda)$ be given. We seek $u \in \widetilde{H}^{-\frac{1}{2}}(\Lambda)$ satisfying*

$$\langle \mathsf{V}u, v \rangle_\Lambda = \langle f, v \rangle_\Lambda, \quad \forall\, v \in \widetilde{H}^{-\frac{1}{2}}(\Lambda).$$

Problem 2.12 (Hypersingular BIE on Λ)**.** *Let $g \in H^{-\frac{1}{2}}(\Lambda)$ be given. We seek $\phi \in \widetilde{H}^{\frac{1}{2}}(\Lambda)$ satisfying*

$$\langle \mathsf{W}\phi, v \rangle_\Lambda = \langle g, v \rangle_\Lambda, \quad \forall\, v \in \widetilde{H}^{\frac{1}{2}}(\Lambda).$$

As with Problems 2.5 and 2.7, the well-posedness of Problems 2.11 and 2.12 follows from Proposition 2.9, Theorem 2.10 and the Lax-Milgram lemma. Moreover, the respective solutions $\psi \in \widetilde{H}^{-\frac{1}{2}}(\Lambda)$ and $\varphi \in \widetilde{H}^{\frac{1}{2}}(\Lambda)$ satisfy

$$\left\| \psi \right\|_{H^{-\frac{1}{2}}(\Lambda)} \leq \frac{1}{\eta} \left\| f \right\|_{H^{\frac{1}{2}}(\Lambda)} \quad \text{and} \quad \left\| \varphi \right\|_{H^{\frac{1}{2}}(\Lambda)} \leq \frac{1}{\mu} \left\| g \right\|_{H^{-\frac{1}{2}}(\Lambda)},$$

with $\mu > 0$ as in Theorem 2.10.

2.6. **Regularity of the solution to the BIEs on open Arcs.** For the ensuing analysis of lower-order BEM, in particular on open arcs, we shall invoke a decomposition of solutions into regular and singular parts, from [38, 16]. For $i = 1, 2$, let ϱ_i denote the Euclidean distance between $\mathbf{x} \in \Lambda$ and the endpoint $\mathbf{x}_i \in \mathbb{R}^2$ of Λ and let ξ_i be a $\mathscr{C}^\infty$ cut-off function on Λ with $0 \leq \xi_i \leq 1$, $\xi_i = 1$ near $\mathbf{x}_i$ and $\xi = 0$ at the opposite end. For $\alpha_1, \alpha_2 \in \mathbb{R}$ and $\psi_0 \in \widetilde{H}^s(\Lambda)$ for any $s < 2$, we set

$$\psi \coloneqq \sum_{j=1}^{2} \alpha_j \varrho_j^{\frac{1}{2}} \xi_j + \psi_0 \tag{2.9}$$

and define

$$\|\psi\|_{\mathscr{T}^s(\Lambda)} \coloneqq \begin{cases} \sum_{j=1}^2 |\alpha_j| + \|\psi_0\|_{\widetilde{H}^s(\Lambda)} & s \in [1,2) \\ \|\psi\|_{\widetilde{H}^s(\Lambda)} & s < 1, \end{cases}$$
$$\mathscr{T}^s(\Lambda) := \mathbb{R}^2 \times \widetilde{H}^s(\Lambda).$$

Proposition 2.13 ([38, Theorem 1.8])**.** *For $\sigma \in (-\frac{1}{2}, \frac{1}{2})$ the operator $\mathsf{W} : \mathscr{T}^{\frac{3}{2}+\sigma}(\Lambda) \to H^{\frac{1}{2}+\sigma}(\Lambda)$*

$$\{\alpha_1, \alpha_2, \psi_0\} \mapsto \mathsf{W}\left(\sum_{j=1}^{2} \alpha_j \varrho_j^{\frac{1}{2}} \xi_j + \psi_0\right) = g,$$

is bijective and continuous and there exists $C > 0$ such that

$$\|\psi\|_{\mathscr{T}^s(\Lambda)} \leq C \|g\|_{H^{\frac{1}{2}+\sigma}(\Lambda)},$$

where ψ is as in (2.9)*.*

2.7. **Galerkin Boundary Element Discretization.** We proceed to detail the Galerkin discretization of Problems 2.5, 2.7, 2.11 and 2.12. We remark that other forms of numerical discretizations, such as collocation or Petrov-Galerkin formulations, may be considered as well.

In what follows, we assume that $\mathrm{D} \subset \mathbb{R}^2$ is a bounded, Lipschitz polygon with boundary $\Gamma \coloneqq \partial\mathrm{D}$. As is customary in the h-Galerkin BEM, we decompose the boundary Γ into $N \in \mathbb{N}$ straight, disjoint line segments Γ_j, for $j = 1, \ldots, N$, from now onwards referred to as elements. The vertices of the polygon Γ must match the endpoints of some of the elements and the mesh $\mathcal{T}_N \coloneqq \{\Gamma_i\}_{i=1}^N$ covers Γ itself. Equipped with these definitions, we introduce the usual boundary element spaces of piecewise polynomial functions on the mesh $\mathcal{T}_N$ of Γ:

$$S^0(\Gamma, \mathcal{T}_N) \coloneqq \left\{\psi \in L^2(\Gamma) : \ \psi|_{\Gamma_j} \in \mathbb{P}_0(\Gamma_j), \quad j = 1, \ldots, N\right\},$$
$$S^1(\Gamma, \mathcal{T}_N) \coloneqq \left\{\psi \in H^1(\Gamma) : \ \psi|_{\Gamma_j} \in \mathbb{P}_1(\Gamma_j), \quad j = 1, \ldots, N\right\}.$$

In the above, and for each $j = 1, \ldots, N$, $\mathbb{P}_p(\Gamma_j)$ denotes the space of polynomials of degree $p \in \mathbb{N} \cup \{0\}$ on Γ_j.

Problem 2.14 (BEM for Problem 2.5)**.** *Let $\mathcal{T}_N$ be a mesh of the boundary Γ and let $f \in H^{\frac{1}{2}}(\Gamma)$ be given. We seek $\psi_N \in S^0(\Gamma, \mathcal{T}_N)$ satisfying*

$$\langle \mathsf{V}\psi_N, \varphi_N \rangle_\Gamma = \left\langle \left(\frac{1}{2}\mathsf{Id} + \mathsf{K}\right) f, \varphi_N \right\rangle_\Gamma, \quad \forall\, \varphi_N \in S^0(\Gamma, \mathcal{T}_N).$$

Problem 2.15 (BEM for Problem 2.7)**.** *Let $\mathcal{T}_N$ be a partition of the boundary Γ and let $g \in H^{-\frac{1}{2}}(\Gamma)$ be given. We seek $\phi_N \in S^1(\Gamma, \mathcal{T}_N)$ satisfying*

$$\langle \mathsf{W}\phi_N, \varphi_N \rangle_\Gamma + \langle \phi_N, 1 \rangle_\Gamma \langle \varphi_N, 1 \rangle_\Gamma = \left\langle \left(\frac{1}{2}\mathsf{Id} - \mathsf{K}'\right) g, \varphi_N \right\rangle_\Gamma, \quad \forall\, \varphi_N \in S^1(\Gamma, \mathcal{T}_N).$$

Remark 2. The continuity and ellipticity of V and the bilinear form $\breve{\mathsf{a}}(\cdot,\cdot)$ (recall its definition in Remark 1) in $H^{-\frac{1}{2}}(\Gamma)$ and $H^{\frac{1}{2}}(\Gamma)$, respectively, will ensure the existence and uniqueness of solutions to Problems 2.14 and 2.15.

Remark 3 (Weakly Singular Operator V). For $\Gamma := \partial\mathrm{D}$ a closed curve of arclength $L = |\Gamma|$, let $\boldsymbol{r} : [0, L) \to \Gamma$ be its arclength parametrization and consider the map I given for $\phi \in \mathscr{C}^0(\Gamma)$ by

$$(I\phi)(\mathbf{x}) := \int_0^{\boldsymbol{r}^{-1}(\mathbf{x})} \phi \circ \boldsymbol{r}(\sigma)\, \mathrm{d}s(\sigma) - \langle \phi, 1 \rangle_\Gamma,$$

for all $\mathbf{x} \in \Gamma$. One verifies that the map I may be extended to $I \in \mathcal{L}_{iso}(H^{-\frac{1}{2}}(\Gamma)/\mathbb{R}, H^{\frac{1}{2}}(\Gamma))$. For $\psi \in H^{-1/2}(\Gamma)$ define $\breve{\psi} := \psi - \frac{1}{L}\langle \psi, 1\rangle_\Gamma \in H^{-1/2}(\Gamma)/\mathbb{R}$. Then Maue's formula in Lemma 2.3 implies, for every $\psi_N, \varphi_N \in S^0(\Gamma, \mathcal{T}_N)$, that

$$\begin{aligned}&\langle \mathsf{V}\psi_N, \varphi_N\rangle_\Gamma \\ &= \left\langle \mathsf{W}(I\breve{\psi}_N), I\breve{\varphi}_N \right\rangle_\Gamma + \frac{1}{L}\left\langle \mathsf{V}\langle \psi_N, 1\rangle_\Gamma, \breve{\phi}_N \right\rangle_\Gamma + \frac{1}{L}\left\langle \mathsf{V}\breve{\psi}_N, \langle \phi_N, 1\rangle_\Gamma \right\rangle_\Gamma + \frac{1}{L^2}\langle \mathsf{V}\langle \psi_N, 1\rangle_\Gamma, \langle \phi_N, 1\rangle_\Gamma\rangle_\Gamma \ .\end{aligned}$$

We continue with the Galerkin discretization of Problems 2.11 and 2.12. Recall $\Lambda \subsetneq \Gamma$ to be a Jordan arc and let $\widetilde{\mathcal{T}}_N := \{\Lambda_i\}_{i=1}^N$ be a mesh on Λ consisting, as before, of straight, disjoint line segments Λ_j, for $j = 1, \ldots, N$. We introduce the following spaces of piecewise polynomials:

$$\begin{aligned}S^0(\Lambda, \widetilde{\mathcal{T}}_N) &:= \left\{\psi \in L^2(\Lambda) \ : \ \psi|_{\Lambda_j} \in \mathbb{P}_0(\Lambda_j), \quad j = 1, \ldots, N\right\}, \\ S^1(\Lambda, \widetilde{\mathcal{T}}_N) &:= \left\{\psi \in \widetilde{H}^1(\Lambda) \ : \ \psi|_{\Lambda_j} \in \mathbb{P}_1(\Lambda_j), \quad j = 1, \ldots, N\right\}.\end{aligned}$$

Problem 2.16 (Boundary element discretization of Problem 2.11)**.** *Let $\widetilde{\mathcal{T}}_N$ be a mesh over the Jordan arc $\Lambda \subsetneq \Gamma$ and let $f \in H^{\frac{1}{2}}(\Lambda)$ be given. We seek $\psi_N \in S^0(\Lambda, \widetilde{\mathcal{T}}_N)$ satisfying*

$$\langle \mathsf{V}\psi_N, \xi_N\rangle_\Lambda = \langle f, \xi_N\rangle_\Lambda \quad \forall\, \xi_N \in S^0(\Lambda, \widetilde{\mathcal{T}}_N).$$

Problem 2.17 (Boundary element discretization of Problem 2.12)**.** *Let $\widetilde{\mathcal{T}}_N$ be a mesh over the Jordan arc $\Lambda \subsetneq \Gamma$ and let $g \in H^{-\frac{1}{2}}(\Lambda)$ be given. We seek $\varphi_N \in S^1(\Lambda, \widetilde{\mathcal{T}}_N)$ satisfying*

$$\langle \mathsf{W}\phi_N, \varphi_N\rangle_\Lambda = \langle g, \varphi_N\rangle_\Lambda \quad \forall\, \varphi_N \in S^1(\Lambda, \widetilde{\mathcal{T}}_N).$$

3. ReLU Neural Networks

We introduce a key ingredient in the present paper, namely the so-called DNNs. Even though in this work we use only the ReLU activation function, other (continuous) activation functions could be considered in the analysis and numerical construction of DNNs, such as sigmoidal, or tanh activations. However, the approximation results presented herein hold with the simpler, and computationally more efficient, ReLU activation function.

Definition 3.1 (Deep Neural Network)**.** *Let $L \geq 2$, $N_0, N_1, \ldots, N_L \in \mathbb{N}$. A map $\Phi : \mathbb{R}^{N_0} \to \mathbb{R}^{N_L}$ given by*

$$\Phi(\boldsymbol{x}) = \boldsymbol{W}_L(\varrho(\boldsymbol{W}_{L-1}(\varrho(\cdots \varrho(\boldsymbol{W}_1((\boldsymbol{x})))))))), \quad \boldsymbol{x} \in \mathbb{R}^{N_0} \tag{3.1}$$

with affine maps $\boldsymbol{W}_\ell : \mathbb{R}^{N_{\ell-1}} \to \mathbb{R}^{N_\ell}$, $\boldsymbol{W}_\ell(\boldsymbol{x}) = \boldsymbol{A}_\ell \boldsymbol{x} + \boldsymbol{b}_\ell$, $1 \leq \ell \leq L$, where $\boldsymbol{A}_\ell \in \mathbb{R}^{N_\ell \times N_{\ell-1}}$, $\boldsymbol{b}_\ell \in \mathbb{R}^{N_\ell}$, and with activation function $\varrho : \mathbb{R} \to \mathbb{R}$ (acting component-wise on vector inputs) is called Deep Neural Network*. In the above definition, N_0 is the dimension of the* input layer*, N_L denotes the dimension of the* output layer*, $L = \mathcal{L}(\Phi)$ denotes the number of layers (excluding the input layer), $N_1, \ldots, N_L$ correspond to the widths of each of the $L-1$ hidden layers, and $M = \mathcal{M}(\Phi) := \max_\ell N_\ell$ corresponds to the width of the network. In addition, we denote by $\mathcal{NN}^\varrho_{L,M,N_0,N_L}$ the set of all DNNs $\Phi : \mathbb{R}^{N_0} \to \mathbb{R}^{N_L}$ with input dimension N_0, output dimension N_L, a depth of at most L layers, maximum width M, and activation function ϱ.*

For any $\Phi \in \mathcal{NN}^\varrho_{L,M,N_0,N_L}$ with $L \geq 2$ as in (3.1)*, we introduce $\Phi^{\mathrm{hid}} : \mathbb{R}^{N_0} \to \mathbb{R}^{N_{L-1}}$ given by*

$$\Phi^{\mathrm{hid}}(\boldsymbol{x}) := \varrho(\boldsymbol{W}_{L-1}(\varrho(\cdots \varrho(\boldsymbol{W}_1(\boldsymbol{x})))))),$$

ie., the subnetwork comprising all "responses from the hidden layers" of a DNN $\Phi \in \mathcal{NN}^\varrho_{L,M,N_0,N_L}$. Moreover, we denote the space of all the responses from the hidden layers of DNN's in $\mathcal{NN}^\varrho_{L,M,N_0,N_L}$ as

$$\mathcal{NN}^{\mathrm{hid},\varrho}_{L,M,N_0,N_L} := \left\{\Phi^{\mathrm{hid}} : \Phi \in \mathcal{NN}^\varrho_{L,M,N_0,N_L}\right\}.$$

3.1. Structure of Galerkin Approximation Spaces generated by ReLU-NNs.

It is an immediate consequence from Definition 3.1 that functions generated by DNNs (understood in the sense that $L \geq 2$ as opposed to shallow NNs, where $L = 1$) depend in a nonlinear fashion on the DNN parameters characterizing the hidden layers, i.e., the weights $\boldsymbol{A}_\ell \in \mathbb{R}^{N_\ell \times N_{\ell-1}}$ and biases $\boldsymbol{b}_\ell \in \mathbb{R}^{N_\ell}$. However, by setting the bias in the output layer of the DNN in Definition 3.1 to $\mathbf{0}$, i.e., $\boldsymbol{b}_L = \mathbf{0}$, we have that DNN functions belong to the *linear span* of the space of functions generated by the "hidden layers" of the corresponding DNN.

Proposition 3.2. *Assume given a DNN $\Phi \in \mathcal{NN}^{\varrho}_{L,M,N_0,N_L}$ with $L \geq 2$ and such that $N_L = 1$, $\boldsymbol{b}_L = \mathbf{0}$. Then, for any activation function $\varrho : \mathbb{R} \to \mathbb{R}$, it holds that*

$$\mathcal{NN}^{\varrho}_{L,M,N_0,N_L} = \operatorname{span}\left\{ \Phi^{\mathrm{hid}}_i : \ \Phi^{\mathrm{hid}} \in \mathcal{NN}^{\mathrm{hid},\varrho}_{L,M,N_0,N_L}, \quad i \in \{1,\dots,N_{L-1}\} \right\} ,$$

where Φ^{hid}_i denotes the i-th component of any $\Phi^{\mathrm{hid}} \in \mathcal{NN}^{\mathrm{hid},\varrho}_{L,M,N_0,N_L}$.

We see from this proposition that DNNs: (i) span particular linear subspaces of dimension N_{L-1} and (ii) span subspaces with basis elements that, in turn, can be chosen in a problem-adapted fashion by adjusting the parameters in the hidden layers of the DNN.

It is clear from this (trivial) observation that shallow ReLU-NNs, which we will consider exclusively in the remainder of this paper, can exactly reproduce spaces of continuous and piecewise linear functions on Γ with a DNN of depth $L = 2$. *Mesh refinement on* Γ can be accounted for by adjusting the widths of the hidden layers during training.

As was shown in [27, Sections 4 and 5], for $L > 2$, ReLU-DNNs can represent *hp*-boundary element spaces on geometric partitions, which are known to afford *exponential convergence rates* for piecewise analytic data. We hasten to add, however, that Proposition 3.2 has wider implications, e.g., training $\mathcal{NN}^{\mathrm{hid},\varrho}$ to emulate reduced bases by a greedy search, for example, will imply that the corresponding Galerkin BEM (with subspaces corresponding to the DNN $\mathcal{NN}^{\varrho} = \operatorname{span}\{\Phi^{\mathrm{hid}} \in \mathcal{NN}^{\mathrm{hid},\varrho}\}$) will deliver performance corresponding to reduced bases BEM. This could be employed to accommodate, for example, Galerkin BEM with basis sets that feature additional properties which are tailored to particular problem classes.

3.2. **$\mathbb{P}_1$-Spline Boundary Element Spaces as ReLU-NNs.** Recall that $N \in \mathbb{N}$ corresponds to the number of elements in the meshes on Γ and Λ ($\mathcal{T}_N$ and $\widetilde{\mathcal{T}}_N$, respectively), and let $p \in \mathbb{N}$ denote the polynomial degree of the boundary element spaces $S^p(\Gamma, \mathcal{T}_N)$ and $S^p(\Lambda, \widetilde{\mathcal{T}}_N)$ of piecewise polynomials of degree p. Moreover, define $\mathrm{I} := (-1, 1)$ and let $\boldsymbol{r} : \bar{\mathrm{I}} \to \mathbb{R}^2$ be a Lipschitz continuous and piecewise affine parametrization of the closed curve Γ (where Γ is Lipschitz) satisfying $\boldsymbol{r}(-1) = \boldsymbol{r}(1)$ and $\boldsymbol{r}'(t) \neq \mathbf{0}$ for almost every $t \in \mathrm{I}$. Define, for $\phi \in \mathscr{C}^0(\Gamma)$, the so-called pullback operator as

$$\tau_{\boldsymbol{r}} \phi := \phi \circ \boldsymbol{r} \in \mathscr{C}^0_{\mathrm{per}}(\bar{\mathrm{I}}),$$

where $\mathscr{C}^0_{\mathrm{per}}(\bar{\mathrm{I}})$ denotes the subspace of continuous, 2-periodic functions. This operator can be uniquely extended in such a way that for $s \in [-1, 1]$ the map $\tau_{\boldsymbol{r}} : H^s(\Gamma) \to H^s_{\mathrm{per}}(\mathrm{I})$ defines a linear, continuous operators that admits a bounded inverse, thus inducing an isomorphism between $H^s(\Gamma)$ and $H^s(\mathrm{I})$, where, for $s \geq 0$, $H^s_{\mathrm{per}}(\mathrm{I})$ denotes the Sobolev space of 2-periodic functions of order s and $H^{-s}_{\mathrm{per}}(\mathrm{I})$ signifies its dual in the $L^2(\mathrm{I})$ duality pairing. The following result addresses the representability of the space $S^1(\Gamma, \mathcal{T}_N)$ by means of ReLU-NNs.

Proposition 3.3. *Let $N \in \mathbb{N}$. For each $\phi_N \in S^1(\Gamma, \mathcal{T}_N)$ there is a ReLU neural network $\Phi_N \in \mathcal{NN}_{2,N+1,1,1}$ such that $(\tau_{\boldsymbol{r}} \phi_N)(t) \equiv \Phi_N(t)$ for all $t \in \mathrm{I}$.*

Proof. Recall the setting described in Section 2.7: Given $N \in \mathbb{N}$ we consider a mesh $\mathcal{T}_N$ of Γ consisting of $N+1$ points $\{\mathbf{x}_n\}_{n=0}^N \subset \Gamma$ and where $\mathbf{x}_0 = \mathbf{x}_N$ (i.e. N distinct points). The set of nodes $\{\mathbf{x}_n\}_{n=0}^N \subset \Gamma$ may be uniquely identified with $\{t_j\}_{j=0}^N \subset \bar{\mathrm{I}}$ through $\boldsymbol{r}(t_j) = \mathbf{x}_j$, for $j = 0, \dots, N$, and we assume that $-1 = t_0 < t_1 < \cdots < t_{N-1} < t_N = 1$ set $K_j = (t_j, t_{j+1})$ for $j = 0, \dots, N-1$, and define

$$S^1(\mathrm{I}, \hat{\mathcal{T}}_N) := \left\{ \phi \in H^1(\mathrm{I}) : \ \phi|_{K_j} \in \mathbb{P}_1, \ j = 0, \dots, N-1 \right\},$$

where $\hat{\mathcal{T}}_N = \cup_{j=0}^{N-1} \overline{K}_j$ is a partition of the interval $\bar{\mathrm{I}}$. Set $h_j = t_{j+1} - t_j$. Let us define the so-called "hat functions"

$$\zeta_j(t) := \begin{cases} \frac{t - t_{j-1}}{h_{j-1}}, & t \in \overline{K_{j-1}}, \\ 1 - \frac{t - t_j}{h_j}, & t \in \overline{K_j}, \\ 0, & t \notin \overline{K_{j-1} \cup K_j}, \end{cases} \quad , \quad t \in \bar{\mathrm{I}},$$

for $j = 1, \dots, N-1$ and $t \in \mathrm{I}$, together with

$$\zeta_0(t) := \begin{cases} 1 - \frac{t - t_0}{h_0}, & t \in \overline{K_0}, \\ 0, & t \notin \overline{K_0}, \end{cases} \quad \text{and} \quad \zeta_N(t) := \begin{cases} \frac{t - t_{N-1}}{h_{N-1}}, & t \in \overline{K_{N-1}}, \\ 0, & t \notin \overline{K_{N-1}}, \end{cases}$$

Recall that $\operatorname{span}\{\zeta_0,\dots,\zeta_N\} = S^1(\mathrm{I},\hat{\mathcal{T}}_N)$, whence for all $\Phi_N \in S^1(\mathrm{I},\hat{\mathcal{T}}_N)$ there are unique coefficients $c_0(\Phi_N),\dots,c_N(\Phi_N)\in\mathbb{R}$, such that

$$\Phi_N(t) = \sum_{j=0}^{N} c_j(\Phi_N)\zeta_j(t), \quad t\in\bar{\mathrm{I}}. \tag{3.2}$$

Each ζ_j can be represented (non-uniquely) using ReLU-NNs as follows

$$\zeta_j(t) = \frac{1}{h_{j-1}}\varrho(t-t_{j-1}) - \left(\frac{1}{h_{j-1}}+\frac{1}{h_j}\right)\varrho(t-t_j) + \frac{1}{h_j}\varrho(t-t_{j+1}), \quad t\in\bar{\mathrm{I}}, \tag{3.3}$$

for $j=1,\dots,N-1$, together with

$$\zeta_0(t) = 1 - \frac{1}{h_0}\varrho(t-t_0) + \frac{1}{h_0}\varrho(t-t_1) \quad\text{and}\quad \zeta_N(t) = \frac{1}{h_{N-1}}\varrho(t-t_{N-1}), \quad t\in\bar{\mathrm{I}}$$

Using (3.3) we obtain

$$\begin{pmatrix}\zeta_0(t)\\ \vdots \\ \zeta_N(t)\end{pmatrix} = \boldsymbol{A}_2\varrho\left(\boldsymbol{A}_1 t + \boldsymbol{b}_1\right) + \boldsymbol{b}_2,$$

where $\boldsymbol{A}_1 \coloneqq (1,\dots,1)^\top \in \mathbb{R}^{(N+1)\times 1}$, $\boldsymbol{b}_1 \coloneqq (-t_0,\dots,-t_N)^\top \in \mathbb{R}^{N+1}$, $\boldsymbol{b}_2 \coloneqq (1,0,\dots,0)^\top \in \mathbb{R}^{N+1}$ and $\boldsymbol{A}_2 \in \mathbb{R}^{(N+1)\times(N+1)}$ is defined as follows

$$\boldsymbol{A}_2 \coloneqq \begin{pmatrix}\boldsymbol{v}_1\\ \boldsymbol{H}\\ \boldsymbol{v}_2\end{pmatrix}$$

where $\boldsymbol{v}_1 \coloneqq (-\frac{1}{h_0},\frac{1}{h_0},0,\dots,0)\in\mathbb{R}^{1\times(N+1)}$, $\boldsymbol{v}_2 \coloneqq (0,\dots,0,\frac{1}{h_{N-1}},0)\in\mathbb{R}^{1\times(N+1)}$, and $\boldsymbol{H}\in\mathbb{R}^{(N-1)\times(N+1)}$

$$\boldsymbol{H}[i,j] \coloneqq \begin{cases} \frac{1}{h_{i-1}}, & \text{if } i=j,\\ -\left(\frac{1}{h_{i-1}}+\frac{1}{h_i}\right), & \text{if } i+1=j,\\ \frac{1}{h_i}, & \text{if } i+2=j,\\ 0, & \text{otherwise,}\end{cases}$$

for $i=1,\dots,N+1$, $j=1,\dots,N+1$. Finally, we construct the output layer by using (3.2). Let us define $\boldsymbol{C}\coloneqq(c_0(\Phi_N),\dots,c_N(\Phi_N))\in\mathbb{R}^{1\times(N+1)}$. Then we have that

$$\Phi_N(t) = \boldsymbol{C}\begin{pmatrix}\zeta_1(t)\\ \vdots\\ \zeta_N(t)\end{pmatrix} = \boldsymbol{C}\left(\boldsymbol{A}_2\varrho\left(\boldsymbol{A}_1 t+\boldsymbol{b}_1\right)+\boldsymbol{b}_2\right), \quad t\in\bar{\mathrm{I}}. \tag{3.4}$$

Observe that $\Phi_N\in\mathcal{NN}_{2,N+1,1,1}$ and that, due to the construction (3.4), the weights of Φ_N are bounded in absolute value by $\max\{1+h_1, \|\phi_N\|_{L^\infty(\Gamma)}, 2\max_{j=0,\dots,N-1}\frac{1}{h_j}\}$.

Now, set $\nu_j = \tau_{\boldsymbol{r}}^{-1}\zeta_j$, for $j=0,\dots,N$. Therefore, for each $\phi_N\in S^1(\Gamma,\mathcal{T}_N)$ we have that

$$\phi_N(\mathbf{x}) = \sum_{j=0}^{N}\phi_N(\mathbf{x}_j)\nu_j(\mathbf{x}), \quad \mathbf{x}\in\Gamma. \tag{3.5}$$

Observe that since $\phi_N(\mathbf{x}_0)=\phi_N(\mathbf{x}_N)$ we have $\phi_N\in S^1(\Gamma,\mathcal{T}_N)$. The application of $\tau_{\boldsymbol{r}}$ to (3.5) yields

$$\left(\tau_{\boldsymbol{r}}\phi_N\right)(t) = \sum_{j=0}^{N}\phi_N(\mathbf{x}_j)\zeta_j(t), \quad t\in\bar{\mathrm{I}}. \tag{3.6}$$

Observe that the right-hand side of (3.6) defines an element of $\mathcal{NN}_{2,N+1,1,1}$, thus concluding the proof. □

Remark 4. As pointed out in the proof of Proposition 3.3, the representation of the "hat functions" $\zeta_j\in\phi_N\in S^1(\mathrm{I},\mathcal{T}_N)$ is not unique. One may also write

$$\zeta_j(t) = \varrho\left(1-\varrho\left(\frac{t-t_j}{h_j}\right)-\varrho\left(\frac{t_j-t}{h_{j+1}}\right)\right), \quad t\in\bar{\mathrm{I}}$$

for $j=0,\dots,N$ and $t\in\bar{\mathrm{I}}$. Then, there exists a neural network $\widetilde{\Psi}_j\in\mathcal{NN}_{4,2,1,1}$ such that $\Psi_j(t)=\zeta_j(t)$, for all $t\in\bar{\mathrm{I}}$ and $j=1,\dots,N$. This representation leads to ReLU-NNs of width 2 and depth 3.

3.3. **Approximation Properties of ReLU-NNs: h-Galerkin BEM.** Based on the result stated in Proposition 3.3 (concerning the exact emulation of the standard $\mathbb{P}_1$-BEM spaces) one may conclude that existing results on the convergence rates of Galerkin BEM are straightforwardly "transferred" to the ReLU Galerkin-BEM framework. In this section, we provide a clear result establishing this connection. Firstly, we recapitulate known approximation results of singular functions on graded partitions.

Proposition 3.4. *Set* $\mathrm{I} = (0,1)$ *and consider a function* $u \in \widetilde{H}^{\frac{1}{2}}(\mathrm{I})$ *of the form*

$$u(x) = \sum_{j=1}^{J} \alpha_j x^{\lambda_j} + u_0(x)\,, \quad \text{where } \mathrm{Re}\{\lambda_j\} > 0,\ \ u_0 \in H^2(\mathrm{I}),\ \ \alpha_j \in \mathbb{C},\ \ x \in \mathrm{I}.$$

Assume $\lambda_0 := \min\{\mathrm{Re}\{\lambda_j\} : j = 1, ..., J\} \geq s$ *for some* $0 < s \leq 1$. *For a grading parameter* $\beta \geq 1$, *denote by* $S^1(\mathrm{I}, \mathcal{T}_{N,\beta})$ *the space of continuous, piecewise linear functions in* $\mathrm{I} = (0,1)$ *on the* graded mesh $\mathcal{T}_{N,\beta}$ *characterized by the nodes* $\{x_k^{N,\beta} := (k/N)^\beta,\ k = 0, 1, ..., N\}$ *in* I. *Then, for every* $\varepsilon > 0$ *and* $s \geq 0$ *such that* $s < \lambda_0 + 1/2$ *and* $s \leq 1$, *and with* $I_N^\beta : \mathscr{C}^0(\bar{\mathrm{I}}) \to S^1(\mathrm{I}, \mathcal{T}_{N,\beta})$ *as the nodal interpolant, there holds that*

$$\|u - I_N^\beta u\|_{\widetilde{H}^s(\mathrm{I})} \lesssim \begin{cases} N^{-(\lambda_0 - (s-1/2))\beta + \varepsilon} & \text{if } 1 \leq \beta \leq \frac{2-s}{\lambda_0 - (s-1/2)}, \\ N^{-(2-s)} & \text{if } \beta > \frac{2-s}{\lambda_0 - (s-1/2)}, \end{cases} \tag{3.7}$$

as N *grows to infinity. In* (3.7), *the implied constant depends on* β, $\{\lambda_j\}_{j=1,...,J}$, α_j, u_0 *and* ε.

Proof. A proof of this result is provided, for the convenience of the reader, in Appendix A. □

We observe that Proposition 3.3 allows for *arbitrary locations of the nodes characterizing the mesh* $\mathcal{T}_N$ *on* Γ, while ensuring exact ReLU-NN emulation of the spaces $S^1(\Gamma, \mathcal{T}_N)$. The preceding result, therefore, also implies approximation rate bounds for ReLU-NNs. We describe these now.

Theorem 3.5 (Approximation of the Solution to BIEs by ReLU–NNs)**.**

(i) Let $\phi \in H^{\frac{1}{2}}(\Gamma)$ *be the solution of Problem 2.7 on a bounded Lipschitz polygon with boundary* Γ *characterized by a finite number* $J \geq 3$ *of vertices. Assume that* $g \in H^{-\frac{1}{2}}(\Gamma)/\mathbb{R}$ *in Problem 2.7 is additionally piecewise analytic and let* $\boldsymbol{r} : \mathrm{I} \to \mathbb{R}^2$ *be a Lipschitz continuous and piecewise linear parametrization of* Γ *satisfying* $\boldsymbol{r}(-1) = \boldsymbol{r}(1)$ *and* $\boldsymbol{r}'(t) \neq \mathbf{0}$ *for almost every* $t \in \bar{\mathrm{I}}$.

Then, there exists $C > 0$ *such that for each* $N \in \mathbb{N}$ *there exists* $\Phi_N \in \mathcal{NN}_{2,M,1,1}$ *satisfying*

$$\|\tau_{\boldsymbol{r}}\phi - \Phi_N\|_{H^{\frac{1}{2}}(\mathrm{I})} \leq C N^{-\frac{3}{2}}, \tag{3.8}$$

where $\tau_{\boldsymbol{r}} : H^{\frac{1}{2}}(\Gamma) \to H^{\frac{1}{2}}(\mathrm{I})$ *denotes the pullback operator introduced in Section 3.2 and* $M = \mathcal{O}(N)$.

(ii) Let $\varphi \in \widetilde{H}^{\frac{1}{2}}(\Lambda)$ *be the solution of Problem 2.12 for* $g \in H^1(\Lambda)$, *and let* $\boldsymbol{r} : \mathrm{I} \to \mathbb{R}^2$ *be a regular parametrization, i.e.* $\boldsymbol{r}'(t) \neq \mathbf{0}$ *for* $t \in \mathrm{I}$, *of the open arc* $\Lambda \subset \mathbb{R}^2$.

Then, for every $\varepsilon > 0$ *there exists* $C(\varepsilon) > 0$ *(depending on* $\varepsilon > 0$*) such that for each* $N \in \mathbb{N}$ *there exists* $\varphi_N \in \mathcal{NN}_{2,N,1,1}$ *satisfying*

$$\|\tau_{\boldsymbol{r}}\varphi - \varphi_N\|_{\widetilde{H}^{\frac{1}{2}}(\mathrm{I})} \leq C(\varepsilon) N^{-\frac{3}{2}+\varepsilon}, \tag{3.9}$$

where $\tau_{\boldsymbol{r}} : \widetilde{H}^{\frac{1}{2}}(\Lambda) \to \widetilde{H}^{\frac{1}{2}}(\mathrm{I})$ *denotes the pullback operator introduced in Section 3.2.*

Proof.

(i) Let $\boldsymbol{r} : \mathrm{I} \to \Gamma$ be a Lipschitz continuous and piecewise linear parametrization of Γ. Being Γ a bounded Lipschitz polygon with straight sides and defined by $J \geq 3$ vertices, there exist points $-1 = t_0 < t_1 < \cdots < t_J := 1$ satisfying $\boldsymbol{r}(t_j) = \mathbf{x}_j$, for $j = 0, \cdots, J$, where we set $\mathbf{x}_0 = \mathbf{x}_J$. Throughout this proof, we use $\mathrm{I}_j := [t_j, t_{j+1}] \subset \mathrm{I}$. A parametrization of this polygon is, for instance, given by

$$\boldsymbol{r}(t) = \frac{t_j - t}{t_j - t_{j+1}}\mathbf{x}_j + \frac{t - t_{j+1}}{t_j - t_{j+1}}\mathbf{x}_{j+1}, \quad t \in \mathrm{I}_j, \quad j = 0, ..., J-1.$$

Additionally, for $j = 0, \ldots, J-1$, we define the *extension by zero* and *restriction* operators $\mathcal{E}_j : \mathscr{C}^0(\mathrm{I}_j) \to \mathscr{C}^0(\mathrm{I})$ and $\mathcal{R}_j : \mathscr{C}^0(\mathrm{I}) \to \mathscr{C}^0(\mathrm{I}_j)$ as

$$\mathcal{E}_j(u)(t) = \begin{cases} u & t \in \mathrm{I}_j \\ 0 & t \notin \mathrm{I}_j \end{cases}, \quad \text{a.e. } t \in \mathrm{I}, \quad \text{and} \quad \mathcal{R}_j(v)(t) = v(t), \quad \text{a.e. } t \in \mathrm{I}_j$$

Equipped with this, and according to Proposition 2.8 item (ii), we have that

$$(\tau_{\boldsymbol{r}}\phi)(t) = \eta_{j,1}(t) + \eta_{j,2}(t) + (\tau_{\boldsymbol{r}}\phi_0)(t), \quad t \in \mathrm{I}_j.$$

with

$$\eta_{j,1}(t) \coloneqq \chi_{j,1}(t) \sum_{k=1}^{n} \beta_{j,k} \frac{\|\mathbf{x}_{j+1} - \mathbf{x}_j\|^{\alpha_{jk}}}{|t_{j+1} - t_j|^{\alpha_{j,k}}} |t - t_j|^{\alpha_{j,k}}, \tag{3.10}$$

and

$$\eta_{j,2}(t) \coloneqq \chi_{j,2}(t) \sum_{k=1}^{n} \beta_{j+1,k} \frac{\|\mathbf{x}_{j+1} - \mathbf{x}_j\|^{\alpha_{j+1,k}}}{|t_{j+1} - t_j|^{\alpha_{j+1,k}}} |t - t_{j+1}|^{\alpha_{j+1,k}} \ , \tag{3.11}$$

where $\chi_{j,1}, \chi_{j,2} : \mathrm{I}_j \to \mathbb{R}$ are infinitely differentiable functions, which, furthermore, are identically equal to 1 for $|t-t_j| < L_j/4$ and $|t-t_{j+1}| < L_j/4$ (with $L_j \coloneqq |t_{j+1}-t_j|$), respectively. In addition, $\chi_{j,1}$ and $\chi_{j,2}$ vanish for $|t-t_j| > L_j/2$ and $|t-t_{j+1}| > L_j/2$, respectively. In (3.10) and (3.11), for each $k \in \mathbb{N}$ and $j = 1, \ldots, J$, we have that $\frac{3}{2}\frac{\omega_j}{\pi} - 1 \leq n \leq \frac{3}{2}\frac{\omega_j}{\pi}$, $\alpha_{jk} \coloneqq k\frac{\pi}{\omega_j}$, $\mathcal{R}_j(\tau_{\boldsymbol{r}}\phi_0) \in H^2(\mathrm{I}_j)$ and $\beta_{j,k}, \beta_{j+1,k} \in \mathbb{R}$, as stated in Proposition 2.8. Observe that in (3.10) and (3.11) we have singularities arising at $t = t_j$ and $t = t_{j+1}$. The strength of these singularities is dictated by the inner angles of the polygon at the corresponding vertices. It follows, from Proposition 3.4, with $s = \frac{1}{2}$ and $\beta_j > \max\left\{\frac{3}{2\alpha_{j,1}}, \frac{3}{2\alpha_{j+1,1}}\right\}$, that there exist $\eta_{j,1,N}, \eta_{j,2,N} \in S^1(\mathrm{I}_j, \mathcal{T}_{N,\beta_j})$ and a constant $C > 0$ independent of N such that

$$\|\eta_{1,j} - \eta_{1,j,N}\|_{\widetilde{H}^{\frac{1}{2}}(\mathrm{I}_j)} \leq CN^{-\frac{3}{2}}, \quad \text{and} \quad \|\eta_{2,j} - \eta_{2,j,N}\|_{\widetilde{H}^{\frac{1}{2}}(\mathrm{I}_j)} \leq CN^{-\frac{3}{2}},$$

and, in addition, there exists $\phi_{j,N} \in S^1(\mathrm{I}_j, \mathcal{T}_{N,\beta_j})$ such that

$$\|\mathcal{R}_j(\tau_{\boldsymbol{r}}\phi_0) - \phi_{j,N}\|_{\widetilde{H}^{\frac{1}{2}}(\mathrm{I}_j)} \leq CN^{-\frac{3}{2}}.$$

Set $\Phi_{j,N} \coloneqq \eta_{j,1,N} + \eta_{j,2,N} + \phi_{j,N} \in S^1(\mathrm{I}, \mathcal{T}_{N,\beta_j})$ and we have that

$$\|\mathcal{R}_j(\tau_{\boldsymbol{r}}\phi) - \Phi_{j,N}\|_{\widetilde{H}^{\frac{1}{2}}(\mathrm{I}_j)} \leq CN^{-\frac{3}{2}} \tag{3.12}$$

for some positive constant C independent of N. At this point, we make the following observation: It holds that $\mathcal{R}_j(\tau_{\boldsymbol{r}}\phi) = \Phi_{j,N}$ on $\partial\mathrm{I}_j$, hence $\mathcal{E}_j(\mathcal{R}(\tau_{\boldsymbol{r}}\phi) - \Phi_{j,N}) \in H^{\frac{1}{2}}(\mathrm{I})$ and

$$\|\mathcal{E}_j(\mathcal{R}(\tau_{\boldsymbol{r}}\phi) - \Phi_{j,N})\|_{H^{\frac{1}{2}}(\mathrm{I})} = \|\mathcal{R}(\tau_{\boldsymbol{r}}\phi) - \Phi_{j,N}\|_{\widetilde{H}^{\frac{1}{2}}(\mathrm{I}_j)}.$$

Following Proposition 3.3, on each side of the polygon we have constructed a ReLU-NN belonging to $\mathcal{NN}_{3,N+2,1,1}$ that approximates $\mathcal{R}_j(\tau_{\boldsymbol{r}}\phi)$ according to (3.12). In addition, Proposition 3.4 implies that these ReLU-NNs interpolate exactly the value of the solution to Problem 2.7 at the vertices of the polygon. Hence, by defining $\Phi_N \coloneqq \Phi_{j,N}$ in I_j, for $j = 1, \ldots, J$, namely we define Φ_N to be equal to the previously constructed ReLU on each side of the polygon, we have constructed a ReLU-NN satisfying (3.8). Indeed, we have that

$$\|\tau_{\boldsymbol{r}}\phi - \Phi_N\|_{H^{\frac{1}{2}}(\mathrm{I})} \leq \sum_{j=0}^{J-1} \|(\mathcal{E}_j \circ \mathcal{R}_j)(\tau_{\boldsymbol{r}}\phi - \Phi_N)\|_{H^{\frac{1}{2}}(\mathrm{I})} = \sum_{j=0}^{J-1} \|\mathcal{E}_j(\mathcal{R}_j(\tau_{\boldsymbol{r}}\phi) - \Phi_{j,N})\|_{\widetilde{H}^{\frac{1}{2}}(\mathrm{I}_j)}$$

This ReLU-NN in particular belongs to $\mathcal{NN}_{3,J(N+1),1,1}$, thus concluding the proof of this statement.

(ii) For the sake of simplicity and without loss of generality, we consider $\Lambda \coloneqq (-1,1) \times \{0\} \subset \mathbb{R}^2$. It follows from Proposition 2.13 that

$$(\tau_{\boldsymbol{r}}\phi)(t) \coloneqq \alpha_1 |1+t|^{\frac{1}{2}} \chi_1(t) + \alpha_2 |1-t|^{\frac{1}{2}} \chi_2(t) + \check{v}(t), \quad t \in \mathrm{I},$$

where we have used the parametrization $\boldsymbol{r}(t) = (t,0)^\top$ of $\Lambda \subset \mathbb{R}^2$, $\alpha_1, \alpha_2 \in \mathbb{R}$, $\chi_1, \chi_2 \in \mathscr{C}^\infty(\mathrm{I})$ are fixed cut-off functions with $\chi_1 = 1$, $\chi_2 = 1$ in a neighborhood of $t = -1$ and $t = 1$, respectively, and $\check{v} \in H^{\frac{1}{2}+\sigma}(\mathrm{I})$ for $\sigma \in \left(-\frac{1}{2}, \frac{1}{2}\right)$. Set

$$\eta_1(t) \coloneqq \alpha_1 |1+t|^{\frac{1}{2}} \chi_1(t), \quad \eta_2(t) \coloneqq \alpha_2 |1-t|^{\frac{1}{2}} \chi_2(t), \quad t \in \mathrm{I}. \tag{3.13}$$

Observe that $[0,1] \ni t \mapsto \eta_1(2t-1)$ and $[0,1] \ni t \mapsto \eta_1(-2t+1)$ defined in (3.13) fit the framework of Proposition 3.4 with $J = 1$ and $\lambda_1 = \frac{1}{2}$. It follows from Proposition 3.4 with $s = \frac{1}{2}$ that for $\beta > \frac{2-s}{\lambda_0 - (s-1/2)} = 3$ that there exists $\eta_{1,N}, \eta_{2,N} \in S^1(\mathrm{I}, \mathcal{T}_{N,\beta})$ and a constant $C > 0$ such that

$$\|\eta_1 - \eta_{1,N}\|_{\widetilde{H}^{\frac{1}{2}}(\mathrm{I})} \leq CN^{-\frac{3}{2}}, \quad \text{and} \quad \|\eta_2 - \eta_{2,N}\|_{\widetilde{H}^{\frac{1}{2}}(\mathrm{I})} \leq CN^{-\frac{3}{2}}.$$

In addition, since $\check{v} \in H^{\frac{1}{2}+\sigma}(\mathrm{I})$ for for $\sigma \in \left(-\frac{1}{2}, \frac{1}{2}\right)$, for every $\varepsilon > 0$ there exists $C(\varepsilon) > 0$ (depending on $\varepsilon > 0$) such that for each $N \in \mathbb{N}$ there exists $\check{v}_N \in S^1(\mathrm{I}, \mathcal{T}_{N,\beta})$ satisfying

$$\|\check{v} - \check{v}_N\|_{\widetilde{H}^{\frac{1}{2}}(\mathrm{I})} \leq C(\varepsilon) N^{-\frac{3}{2}+\varepsilon}.$$

Observe that by adding $\eta_{1,N}, \eta_{2,N}, \check{v}_N \in S^1(\mathrm{I}, \mathcal{T}_{N,\beta})$ and recalling that $S^1(\mathrm{I}, \mathcal{T}_{N,\beta})$ is a vector space, we conclude that $\varphi_N := \eta_{1,N} + \eta_{2,N} + \check{v}_N \in S^1(\mathrm{I}, \mathcal{T}_{N,\beta})$ fulfills the estimate in (3.9). According to Proposition 3.3, $\varphi_N \in \mathcal{NN}_{2,N+1,1,1}$. However, recalling that $\varphi_N(-1) = \varphi_N(1) = 0$, we can discard the basis functions associated to the endpoints of the interval and conclude that $\varphi_N \in \mathcal{NN}_{3,N,1,1}$.

□

4. ReLU Neural Network Galerkin BEM

In this section, we propose two algorithms to construct the ReLU-NNs described in Section 3 for the approximation of the solution to BIEs in polygons and arcs, introduced in Sections 2.3 and 2.5, respectively. Following the representation of the lower-order boundary element spaces as ReLU-NNs elaborated in Section 3, we focus only on *shallow* NNs. The first method, described herein in Section 4.1, aims to construct a ReLU-NN by considering as a loss function the *total energy* of the problem. Indeed, it is well-known that the solution to operators equations involving "elliptic" operators in Hilbert spaces may be cast as minimization problems on Hilbert spaces. We rely on this observation to mathematically justify this algorithm for the construction of the corresponding ReLU-NN. This approach has been previously described for example in [19, Section 7.2] in the context of one-dimensional FEM for the Poisson problem with non-homogeneous Dirichlet boundary conditions, and actually is the key ingredient of the so-called *Deep Ritz Method*, described in [13, 24, 22, 14]. The second method proposed in this work consists in the construction of a computable loss function based on *a posteriori* error estimators for BIEs of the first kind. These tools come in different flavors, and we refer to [15] for an extensive review, including their application to the Adaptive BEM. In particular, here we make use of the so-called *weighted residual estimators* [7, 8, 9], which have been proven to be *reliable*, thus providing a computable upper bound for the error of the Galerkin BEM. To date, these are the only ones shown to deliver optimal convergence rates when used in the Adaptive BEM algorithm (ABEM for short). Finally, in Section 4.3, we describe precise algorithms for the computations of the ReLU-NNs for the approximation of the solutions to the hypersingular BIEs introduced in Problems 2.7 and 2.12.

4.1. Energy Minimization.

Throughout this section, let X be a real Hilbert space equipped with the inner product $(\cdot,\cdot)_X$ and the induced norm $\|\cdot\|_X = \sqrt{(\cdot,\cdot)_X}$. In addition, let X' denote the dual space of X and let $\langle\cdot,\cdot\rangle_X$ represent the duality pairing between X' and X. As it is customary, we endow X' with the dual norm

$$\|\mathsf{f}\|_{X'} := \sup_{0\neq u\in X} \frac{|\langle \mathsf{f}, u\rangle_X|}{\|u\|_X}, \quad \mathsf{f} \in X'.$$

In addition, we say that operator $\mathsf{A} : X \to X'$ is X-elliptic if there exists a constant $C_a > 0$ such that

$$\langle \mathsf{A}v, v\rangle_X \geq C_a \|v\|_X^2, \quad \forall\, v \in X. \tag{4.1}$$

In the following, we recall a well-established result on continuous, self-adjoint and positive semi-definite operators. This property has been previously used in the construction of DNNs in the "Deep Ritz Method" framework introduced in [14].

Lemma 4.1 ([34, Lemma 3.2])**.** *Let $\mathsf{A} : X \to X'$ be a continuous, self-adjoint and positive semi-definite operator, i.e.*

$$\langle \mathsf{A}v, v\rangle_X \geq 0, \quad \forall\, v \in X,$$

and let $\mathsf{f} \in X'$. Then, $u \in X$ solves the variational problem

$$\langle \mathsf{A}u, v\rangle_X = \langle \mathsf{f}, v\rangle_X, \quad \forall\, v \in X,$$

if and only if

$$u = \arg\min_{v\in X} \left(\frac{1}{2} \langle \mathsf{A}v, v\rangle_X - \langle \mathsf{f}, v\rangle_X \right).$$

Lemma 4.1 allows us to express Problems 2.7 and 2.12 (as well as their discrete counterparts) as minimization problems over the corresponding Hilbert spaces. Let us define,

$$\ell_\Gamma : H^{\frac{1}{2}}(\Gamma) \to \mathbb{R} : \phi \mapsto \frac{1}{2}\left\langle \mathsf{W}\phi, \phi\right\rangle_\Gamma + \left\langle \phi, 1\right\rangle_\Gamma^2 - \left\langle g_\Gamma, \phi\right\rangle_\Gamma, \tag{4.2}$$

$$\ell_\Lambda : \widetilde{H}^{\frac{1}{2}}(\Lambda) \to \mathbb{R} : \varphi \mapsto \frac{1}{2}\left\langle \mathsf{W}\varphi, \varphi\right\rangle_\Lambda - \left\langle g_\Lambda, \varphi\right\rangle_\Lambda, \tag{4.3}$$

where $g_\Gamma \in H^{-\frac{1}{2}}(\Gamma)$ and $g_\Lambda \in H^{-\frac{1}{2}}(\Lambda)$ represent the right-hand sides of the variational BIEs.

For each $N \in \mathbb{N}$, we aim to find ReLU-NNs $\phi^\star_{\Gamma,N} \in \mathcal{NN}_{2,N+1,1,1}$ and $\varphi^\star_{\Lambda,N} \in \mathcal{NN}_{2,N,1,1}$ minimizing the loss functions defined in (4.2) and (4.3), i.e.,

$$\phi^\star_{\Gamma,N} \coloneqq \operatorname*{arg\,min}_{\phi \in \mathcal{NN}_{2,N+1,1,1} \cap H^{\frac{1}{2}}_{\mathrm{per}}(\mathrm{I})} \ell_\Gamma(\tau_{\boldsymbol{r}}^{-1}\phi) \qquad \text{and} \qquad \varphi^\star_{\Lambda,N} \coloneqq \operatorname*{arg\,min}_{\varphi \in \mathcal{NN}_{2,N,1,1} \cap \widetilde{H}^{\frac{1}{2}}(\mathrm{I})} \ell_\Lambda(\tau_{\boldsymbol{r}}^{-1}\varphi). \tag{4.4}$$

Remark 5. Observe that $\varphi^\star_{\Lambda,N}$ must vanish at the boundary of Λ. Hence, the width of the family of ReLU-NNs over which we search $\varphi^\star_{\Lambda,N}$ in (4.4) is exactly N. In addition, recall that $\mathcal{NN}_{2,N+1,2,1} \cap H^{\frac{1}{2}}_{\mathrm{per}}(\mathrm{I}) \subset \mathscr{C}^0_{\mathrm{per}}(\mathrm{I})$, thus enforcing periodicity of the corresponding of the ReLU-NN in the approximation of the density over a closed curve Γ.

Remark 6. While the loss functions (4.2) and (4.3) are applied to the operator W, similar loss functions can be derived for V. Consisting of piecewise constant discontinuous functions on Γ, the spaces $S^0(\Gamma, \mathcal{T}_N)$ in the Galerkin BEM Problem 2.14 for Symm's BIE can not be exactly realized via ReLU-NNs. However, with Remark 3, the corresponding loss functions for the Galerkin BEM Problem 2.14 can be realized also with ReLU-NNs on Γ. All results on convergence rates for W will have, via Remark 3, analogs for V. For reasons of length, we develop the ReLU Galerkin BEM algorithms only for W and loss functions introduced in (4.2) and (4.3).

The following result and accompanying corollaries motivate our choice of loss functions. We use this result to address the construction of ReLU–NNs by solving the minimization problems listed in (4.4).

Lemma 4.2. *Let $\mathsf{A} : X \to X'$ be a continuous, self-adjoint and X–elliptic operator. Given $f \in X'$, let $u \in X$ be the unique solution to the following variational problem:*

$$\left\langle \mathsf{A}u, v\right\rangle_X = \left\langle f, v\right\rangle_X, \quad \forall v \in X. \tag{4.5}$$

Define $\ell : X \to \mathbb{R}$ as

$$\ell(v) \coloneqq \frac{1}{2}\left\langle \mathsf{A}v, v\right\rangle_X - \left\langle f, v\right\rangle_X, \quad v \in X.$$

Then, there exist positive constants C_1 and C_2, both independent of $u \in X$, such that

$$\forall v \in X: \;\; C_1\left(\left\langle f, u\right\rangle_X + 2\ell(v)\right) \le \|u - v\|_X^2 \le C_2\left(\left\langle f, u\right\rangle_X + 2\ell(v)\right). \tag{4.6}$$

Proof. Due to the continuity and X-ellipticity of the operator $\mathsf{A} : X \to X'$, we may define an equivalent norm to $\|\cdot\|_X$ in X as follows:

$$\|v\|_{\mathsf{A}} \coloneqq \sqrt{\left\langle \mathsf{A}v, v\right\rangle_X}, \quad v \in X.$$

Indeed, if $C_c > 0$ denotes the continuity constant of $\mathsf{A} : X \to X'$ and $C_a > 0$ is as in (4.1), for all $v \in X$ it holds

$$C_a^{\frac{1}{2}}\|v\|_X \le \|v\|_{\mathsf{A}} \le C_c^{\frac{1}{2}}\|v\|_X.$$

Let $u \in X$ be the unique solution to the variational problem (4.5). Then, for all $v \in X$, we have

$$\|u - v\|_X \le C_a^{-\frac{1}{2}}\|u - v\|_{\mathsf{A}} \quad \text{and} \quad \|u - v\|_X \ge C_c^{-\frac{1}{2}}\|u - v\|_{\mathsf{A}}. \tag{4.7}$$

For $v \in X$, we calculate

$$\|u - v\|_{\mathsf{A}}^2 = \left\langle \mathsf{A}(u - v), u - v\right\rangle_X \quad = \left\langle f, u\right\rangle_X - 2\left\langle f, v\right\rangle_X + \left\langle \mathsf{A}v, v\right\rangle_X. \tag{4.8}$$

In (4.8) we use (4.5) and the self-adjointness of the operator $\mathsf{A} : X \to X'$. The bounds presented in (4.6), follow from (4.8) and (4.7) with $C_1 = C_a^{-1} > 0$ and $C_2 = C_c^{-1} > 0$. □

The next results, relevant for DNN training, follow from Lemma 4.2 and Proposition 3.3.

Corollary 4.3. *Let $\phi \in H^{\frac{1}{2}}(\Gamma)$ be the unique solution to Problem 2.7 with $g_\Gamma \in H^{-\frac{1}{2}}(\Gamma)/\mathbb{R}$. Then, there exist positive constants C_1 and C_2 (independent of ϕ and g) such that for all $N \in \mathbb{N}$ such that $N+1 > J$ and for all $\mathfrak{\phi} \in \mathcal{NN}_{2,N+1,1,1}$, it holds*

$$C_1\left(\left\langle\left(\frac{1}{2}\mathsf{Id}-\mathsf{K}'\right)g_\Gamma,\phi\right\rangle_\Gamma+2\ell_\Gamma(\tau_{\boldsymbol{r}}^{-1}\mathfrak{\phi})\right)\leq\|\tau_{\boldsymbol{r}}\phi-\mathfrak{\phi}\|^2_{H^{\frac{1}{2}}(\mathrm{I})}\leq C_2\left(\left\langle\left(\frac{1}{2}\mathsf{Id}-\mathsf{K}'\right)g_\Gamma,\phi\right\rangle_\Gamma+2\ell_\Gamma(\tau_{\boldsymbol{r}}^{-1}\mathfrak{\phi})\right),$$

where $\ell_\Gamma : H^{\frac{1}{2}}(\Gamma) \to \mathbb{R}$ is as in (4.2), the ReLU-NN $\mathfrak{\phi}$ is identified with its restriction to I and $\tau_{\boldsymbol{r}} : H^{\frac{1}{2}}(\Gamma) \to H^{\frac{1}{2}}(\mathrm{I})$ denotes the pullback operator.

Corollary 4.4. *Let $\varphi \in \widetilde{H}^{\frac{1}{2}}(\Lambda)$ be the unique solution to Problem 2.12 with right hand side $g_\Lambda \in H^{-\frac{1}{2}}(\Lambda)$. Then, there exist positive constants C_1 and C_2 independent of ϕ and g such that for all $N \in \mathbb{N}$ and for all $\mathfrak{\varphi} \in \mathcal{NN}_{3,N,1,1} \cap \widetilde{H}^{\frac{1}{2}}(\Lambda)$, it holds*

$$C_1\left(\langle g_\Lambda,\varphi\rangle_\Lambda+2\ell_\Lambda(\tau_{\boldsymbol{r}}^{-1}\mathfrak{\varphi})\right)\leq\|\tau_{\boldsymbol{r}}\varphi-\mathfrak{\varphi}\|^2_{\widetilde{H}^{\frac{1}{2}}(\mathrm{I})}\leq C_2\left(\langle g_\Lambda,\varphi\rangle_\Lambda+2\ell_\Lambda(\tau_{\boldsymbol{r}}^{-1}\mathfrak{\varphi})\right),$$

where $\ell_\Lambda : \widetilde{H}^{\frac{1}{2}}(\Lambda) \to \mathbb{R}$ is as in (4.3), the ReLU-NN $\mathfrak{\varphi}$ is identified with its restriction to I and $\tau_{\boldsymbol{r}} : \widetilde{H}^{\frac{1}{2}}(\Lambda) \to \widetilde{H}^{\frac{1}{2}}(\mathrm{I})$ denotes the pullback operator.

4.2. **Weighted Residual Estimators.** We shortly recall the so-called *weighted residual estimators* for the a-posteriori error estimation of the numerical solution to hypersingular BIEs in a bounded Lipchitz polygon and in an open arc in $\mathbb{R}^2$, namely Problems 2.7 and 2.12, respectively. We proceed to recapitulate the result of [7], in which *reliable* a-posteriori error estimates for first-kind integral equations are analyzed.

Proposition 4.5 ([7, Theorem 2])**.** *Let Γ be a closed or open arc in $\mathbb{R}^2$. If $f \in L^2(\Gamma)$ is $L^2(\Gamma)$–orthogonal to $S^1(\Gamma,\mathcal{T}_N)$, then for $s \in [0,1]$ it holds*

$$\|f\|_{H^{-s}(\Gamma)} \leq c(s,\kappa)\,\|h_{\mathcal{T}}^s f\|_{L^2(\Gamma)},$$

where

$$c(s,\kappa) := \begin{cases} C_s & \text{if } s \neq \frac{1}{2} \\ C_{\frac{1}{2}}(\log(1+\kappa))^{\frac{1}{2}} & \text{if } s = \frac{1}{2}, \end{cases}, \quad \kappa := \max\left\{\frac{h_j}{h_k} : \Gamma_j \text{ is a neighbor of } \Gamma_k\right\}, \tag{4.9}$$

$C_s > 0$ only depending on $s \in [0,1]$, and $h_{\mathcal{T}} \in L^\infty(\Gamma)$ is the piece-wise constant function defined element-wise as $h_{\mathcal{T}}|_{\Gamma_j} = h_j$.

Corollary 4.6. *Let $\mathrm{D} \subset \mathbb{R}^2$ be a bounded Lipschitz polygon with boundary $\Gamma := \partial\mathrm{D}$, and let $\Lambda \subset \mathbb{R}^2$ be a Jordan arc. Let $c(\kappa) > 0$ denote the constant in (4.9) with $s = \frac{1}{2}$.*

(i) Let $\phi \in H^{\frac{1}{2}}(\Gamma)$ and $\phi_N \in S^1(\Gamma,\mathcal{T}_N)$ be the solution to Problem 2.7 and Problem 2.17, respectively, with $g_\Gamma \in L^2(\Gamma)$.

$$\|\phi-\phi_N\|_{H^{\frac{1}{2}}(\Gamma)} \leq c(\kappa)\left\|h_{\mathcal{T}}^{\frac{1}{2}}\mathsf{R}_N\right\|_{L^2(\Gamma)}, \quad \mathsf{R}_N := \mathsf{W}\phi_N - \left(\frac{1}{2}\mathsf{Id}-\mathsf{K}'\right)g_\Gamma.$$

(ii) Let $\varphi \in \widetilde{H}^{\frac{1}{2}}(\Lambda)$ and $\varphi_N \in S^1(\Lambda,\mathcal{T}_N)$ be the solution to Problems 2.12 and Problem 2.15, respectively, with $g_\Lambda \in L^2(\Lambda)$.

$$\|\varphi-\varphi_N\|_{\widetilde{H}^{\frac{1}{2}}(\Lambda)} \leq c(\kappa)\left\|h_{\mathcal{T}}^{\frac{1}{2}}\mathsf{R}_N\right\|_{L^2(\Lambda)}, \quad \mathsf{R}_N := \mathsf{W}\varphi_N.$$

Proof. We prove item (i). The Galerkin solution $\phi_N \in S^1(\Gamma,\mathcal{T}_N)$ to Problem 2.15 satisfies

$$\langle\mathsf{W}\phi_N,\psi_N\rangle_\Gamma = \left\langle\left(\frac{1}{2}\mathsf{Id}-\mathsf{K}'\right)g_\Gamma,\psi_N\right\rangle_\Gamma, \quad \forall\psi_N \in S^1(\Gamma,\mathcal{T}_N), \tag{4.10}$$

as by construction it holds $\langle\phi_N,1\rangle_\Gamma = 0$. Recalling that for bounded Lipschitz polygons the maps $\mathsf{W} : H^1(\Gamma) \to L^2(\Gamma)$ and $\mathsf{K}' : L^2(\Gamma) \to L^2(\Gamma)$ are continuous. Considering that $g_\Lambda \in L^2(\Gamma)$, we have that the duality pairings appearing in (4.10) can be interpreted as $L^2(\Gamma)$-inner products. Hence, we have that

$$\left(\mathsf{W}\phi_N-\left(\frac{1}{2}\mathsf{Id}-\mathsf{K}'\right)g_\Gamma,\psi_N\right)_{L^2(\Gamma)} = 0, \quad \forall\psi_N \in S^1(\Gamma,\mathcal{T}_N),$$

and we may conclude that the residual $\mathsf{R}_N \in L^2(\Gamma)$ is orthogonal to $S^1(\Gamma,\mathcal{T}_N)$. Recall the bilinear form $\mathsf{a} : H^{\frac{1}{2}}(\Gamma)\times H^{\frac{1}{2}}(\Gamma) \to \mathbb{R}$ defined in (2.3). There exists an operator $\check{\mathsf{W}} : H^{\frac{1}{2}}(\Gamma) \to H^{-\frac{1}{2}}(\Gamma)$ such that

$$\check{\mathsf{a}}(\phi,\psi) = \langle\check{\mathsf{W}}\phi,\psi\rangle_\Gamma, \quad \text{for all } \phi,\psi \in H^{\frac{1}{2}}(\Gamma).$$

Next, using the continuity and $H^{\frac{1}{2}}(\Gamma)$-ellipticity of the modified hypersingular BIO $\check{\mathsf{W}} : H^{\frac{1}{2}}(\Gamma) \to H^{-\frac{1}{2}}(\Gamma)$, stated in Propositions 2.1 and 2.2, respectively, we conclude that there exists a constant $C > 0$ independent of $\mathcal{T}_N$ such that

$$\|\phi - \phi_N\|_{H^{\frac{1}{2}}(\Gamma)} \leq C \left\|\check{\mathsf{W}}\phi - \check{\mathsf{W}}\phi_N\right\|_{H^{-\frac{1}{2}}(\Gamma)} \leq C \left\|\left(\frac{1}{2}\mathsf{Id} - \mathsf{K}'\right) g - \check{\mathsf{W}}\phi_N\right\|_{H^{-\frac{1}{2}}(\Gamma)}. \tag{4.11}$$

In addition, $\langle \phi_N, 1\rangle_\Gamma = 0$ yields $\check{\mathsf{W}}\phi_N = \mathsf{W}\phi_N$. In view of this, together with (4.11) and using Theorem 4.5 with $f = \left(\frac{1}{2}\mathsf{Id} - \mathsf{K}'\right) g - \mathsf{W}\phi_N$, we get the desired result. The proof of item (ii) follows the exact same steps of item (i), hence we omit it for the sake of simplicity. □

4.3. **Training Algorithms.** In this section, we describe two algorithms devised to construct ReLU-NNs for the approximation of the solution to Problems 2.7 and 2.12. These two approaches rely on the following observation: Each element space of piecewise linear polynomials defined on a suitable partition of the boundary can be exactly represented by a ReLU-NN according to Proposition 3.3. Moreover, the parameters of these ReLU-NNs, i.e. weights and biases, can be precisely described in terms of the parameters of the partition, namely the position of the nodes over the boundary. Therefore, we can replace the Galerkin solution to the discrete variational problem by a ReLU-NN as the ones described in Proposition 3.3 and proceed to find its parameters by minimizing a suitable loss function. The algorithms to be presented here are in the spirit of the "Deep Ritz Method" [14] and rely on the tools introduced in Sections 4.1 and 4.2.

4.3.1. *Training using minimization of the total enery.* Corollaries 4.3 and 4.4 provide a justification to use the loss functions $\ell_\Gamma(\cdot)$ and $\ell_\Lambda(\cdot)$ in the construction of the sought ReLU-NNs. Indeed, these quantities can be used a surrogates of the exact error in the minimization process. Equipped with this observation, we proceed to describe a two-step scheme to find a realization of ReLU-NN that approximates the solution to Problems 2.7 and 2.12 using the aforementioned loss functions.

We begin by describing the algorithm for Problem 2.7 only. As per usual, we consider a bounded Lipschitz polygon $\mathrm{D} \subset \mathbb{R}^2$ with boundary $\Gamma := \partial\mathrm{D}$ characterized by a number $J \in \mathbb{N}$ ($J \geq 3$) of vertices. Let $N \in \mathbb{N}$ be fixed and such that $N + 1 > J$. We consider a mesh $\mathcal{T}_N$ of Γ as described in Section 2.7, with $N+1$ points $\{\mathbf{x}_n\}_{n=0}^N \subset \Gamma$ and where $\mathbf{x}_0 = \mathbf{x}_N$ (i.e. N distinct points). Within the set $\{\mathbf{x}_n\}_{n=0}^N \subset \Gamma$ one may identify two kinds of nodes: (i) there is a first subset consisting in the vertices of the polygon Γ, which in the the following are referred to as *fixed* nodes, and (ii) there is a second set of mesh nodes that are not vertices of the polygon, which in the following are referred to as *free* nodes. Moreover, with a Lipschitz continuous and piecewise linear parametrization of Γ, $\boldsymbol{r} : \mathrm{I} \to \Gamma$ (as in Section 3.2), the set of nodes $\{\mathbf{x}_n\}_{n=0}^N \subset \Gamma$ may be identified with a set of biases $\boldsymbol{t} := \{t_n\}_{n=0}^N \subset \mathrm{I}$ such that $\boldsymbol{r}(t_n) = \mathbf{x}_n$ for each $n = 0, \ldots, N$ (see Proposition 3.3). We aim to find the position of the biases in an optimal fashion, while keeping the biases associated with fixed nodes unaltered. Figure 1 illustrates this setting for an open arc in $\mathbb{R}^2$.

FIGURE 1. Two sets of equispaced nodes for a straight open arc. Dots represent fixed nodes, while dashes represent free nodes.

Let $\boldsymbol{t}_{\mathrm{F}} \in \mathbb{R}^{N+1-J}$ be a vector containing the elements of the biases $\{\boldsymbol{t}_n\}_{n=0}^N$ that are associated to the free nodes only and let $\phi_N \in \mathcal{NN}_{3,N+2,1,1}$. In view of Proposition 3.3, there holds

$$\phi_N(t) = \sum_{j=0}^{N} c_j \zeta_j(t), \quad t \in \bar{\mathrm{I}}, \tag{4.12}$$

where $\mathbf{c} := (c_0, \ldots, c_N)^\top \in \mathbb{R}^{N+1}$ and, for each $n = 1, \ldots, N$, $\{\zeta_n\}_{n=0}^N$ corresponds to the "hat" functions introduced in Section 3.2. Then, in view of Proposition 3.3, the task of finding a ReLU-NN ϕ_N to approximate the solution of Problem 2.7 may be stated in the following form:

$$(\boldsymbol{t}^\star, \mathbf{c}^\star) = \underset{\substack{\boldsymbol{t}_{\mathrm{F}} \in \mathbb{R}^{N+2-J},\ \mathbf{c} \in \mathbb{R}^{N+1}, \\ \phi_N(-1) = \phi_N(1).}}{\arg\min} \ell_\Gamma\left(\tau_{\boldsymbol{r}}^{-1} \phi_N(\boldsymbol{t}, \mathbf{c})\right), \tag{4.13}$$

where $\ell_\Gamma(\cdot)$ has been introduced in (4.2) and by $\phi_N(\boldsymbol{t}, \mathbf{c})$ we denote the dependence of ϕ_N upon the biases $\boldsymbol{t}$ and the set of weights $\mathbf{c}$. Note our slight abuse of notation by referring to both the vector of coefficients of the BEM solution in (4.12) and to the ReLU-NN weights as $\mathbf{c}$ (see Proposition 3.3).

In Algorithm 1, we propose a two-step scheme for the construction of optimal ReLU-NN in the approximation of the solution to Problem 2.7. This algorithm is based on the following observation: if we fix the free biases $\boldsymbol{t}_{\mathrm{F}}$, the minimization of (4.13) for the computation of $\mathbf{c}^\star \in \mathbb{R}^N$ boils down to the solution of the Galerkin discretization of Problem 2.7, namely Problem 2.15. Hence, rather than computing the gradient with respect to $\boldsymbol{t}_{\mathrm{F}} \in \mathbb{R}^{N+2-J}$ and $\mathbf{c} \in \mathbb{R}^{N+1}$, and executing a step of the gradient descent algorithm, we perform only a gradient descent step with respect to the vector of free nodes $\boldsymbol{t}_{\mathrm{F}} \in \mathbb{R}^{N+2-J}$ while keeping that of the coefficients $\mathbf{c} \in \mathbb{R}^{N+1}$ fixed. Then, in the second step of this algorithm, we fix the biases and compute $\mathbf{c}^\star \in \mathbb{R}^{N+1}$ in (4.13) by solving the discretized boundary integral equation in Problem 2.15.

Algorithm 1 Construction of ReLU-NN by minimizing the total energy

Input: Initial set of biases $\boldsymbol{t} \in \mathbb{R}^{N+2}$; maximum number of iterations $M \in \mathbb{N}$; tolerance $\epsilon > 0$.
Output: Optimal solution $(\boldsymbol{t}^\star, \mathbf{c}^\star)$.

procedure ReLU_Optimization_Total_Energy($\boldsymbol{t}$,M,ϵ)
  Compute $\mathbf{c} \in \mathbb{R}^N$ by solving Problem 2.15 on the mesh associated with the biases $\boldsymbol{t}$;
  for $\{j = 1;\ j \le M;\ j \leftarrow j + 1\}$ **do**
    Find $\eta > 0$ by line search: $\tilde{\boldsymbol{t}}_{\mathrm{F}} \leftarrow \boldsymbol{t}_{\mathrm{F}} - \eta \nabla_{\boldsymbol{t}_{\mathrm{F}}} \ell_\Gamma(\phi_N(\boldsymbol{t}, \mathbf{c}))$;
    if $\eta \, \|\nabla_{\boldsymbol{t}_{\mathrm{F}}} \ell_\Gamma(\phi_N(\boldsymbol{t}, \mathbf{c}))\|^2_{\mathbb{R}^{N+2-J}} < \epsilon$ **then**
      return $\boldsymbol{t}$ and $\mathbf{c}$;
    end if
    $\boldsymbol{t}_{\mathrm{F}} \leftarrow \tilde{\boldsymbol{t}}_{\mathrm{F}}$;
    Compute $\mathbf{c} \in \mathbb{R}^N$ by solving Problem 2.15 on the mesh associated with the biases $\boldsymbol{t}$;
  end for
  return $(\boldsymbol{t}, \mathbf{c})$;
end procedure

For given initial set of biases (with a fixed number of biases), Algorithm 1 returns an optimal collection of biases in Γ (with the same cardinality) upon which the loss function $\ell_\Gamma(\cdot)$ is minimized. This, in turn allows us to construct a ReLU–NN for the approximation of the solution to Problem 2.7 according to Proposition 3.3. To construct a *sequence* of ReLU–NN with increasing width (corresponding to a sequence of meshes with an increasing number of nodes according to Proposition 3.3) we propose Algorithm 2. Therein, we widen the network by including neurons with (initial) biases given as the midpoints between two contiguous biases, which are subsequently optimized through Algorithm 1.

Algorithm 2 Construction of a sequence of ReLU-NN minimizing the total energy

Input:
- Initial set of biases $\boldsymbol{t} \in \mathbb{R}^{N+2}$;
- Maximum number of iterations $K \in \mathbb{N}$;
- Maximum number of inner iterations $M \in \mathbb{N}$;
- Tolerance $\epsilon > 0$.

Output: Sequence of optimal network parameters $\{(\boldsymbol{t}_j^\star, \mathbf{c_j}^\star)\}_{j=1}^K$.

procedure Sequence_ReLU_Total_Energy($\boldsymbol{t}$,K,M,ϵ)
  for $\{j = 1;\ j < K;\ j \leftarrow j + 1\}$ **do**
    $(\boldsymbol{t}_j^\star, \mathbf{c}_j^\star) \leftarrow$ ReLU_Optimization_Total_Energy($\boldsymbol{t}$,M,ϵ);
    $\boldsymbol{t}_{\mathrm{F}} \leftarrow$ Increase the number of biases in $\boldsymbol{t}_{\mathrm{F},j}^\star$ by including the middle points of contiguous biases.
  end for
  $(\boldsymbol{t}_K^\star, \mathbf{c}_K^\star) \leftarrow$ ReLU_Optimization_Total_Energy($\boldsymbol{t}$,M,ϵ);
  return $\{(\boldsymbol{t}_j^\star, \mathbf{c}_j^\star)\}_{j=1}^K$;
end procedure

Through Algorithm 2 we construct a sequence of biases $\{\boldsymbol{t}_i^\star\}_{i\in\mathbb{N}}$ from a given initial configuration $\boldsymbol{t}$ such that the sequence $\{\ell_\Gamma(\Phi(\boldsymbol{t}_i^\star, \mathbf{c}_i))\}_{i\in\mathbb{N}}$ is monotonically decreasing, where $\mathbf{c}_i$ is obtained by solving Problem 2.15 on the mesh $\mathcal{T}_{N_i}^\star$ associated with the biases $\boldsymbol{t}_i^\star$. Greedy algorithms aiming to construct *shallow* networks for function approximation characterized by a variety of activation function, e.g. the so-called ReLU^k and sigmoidal activation function, are studied in [33]. Improved convergence rates for the Orthogonal Greedy Algorithm (with respect to the well-known result presented in [4, 12]) depending on

the smoothness properties of the activation function of choice are proven. Later in [18], these findings are used for the NN approximation of the solution to elliptic boundary value problem. We mention, however, that considering Algorithm 2 with a sufficiently high number of inner iterations $M \in \mathbb{N}$ (i.e. iterations of Algorithm 1) yields, in our numerical examples, the same convergence rates proved in [18] and the references there. Furthermore, Algorithms 1 and 2 may be modified to tackle the minimization of $\ell_\Lambda(\cdot)$ in (4.3) by simply replacing, in both algorithms, Γ by Λ.

4.3.2. *Training Using Weighted Residual Estimators.* The result presented in Corollary 4.6 justifies the use of *efficiently computable, weighted residual a-posteriori estimators* in the construction of a suitable loss function to be used in the numerical implementation of the ReLU–NN BEM algorithms described herein. More precisely, the computable, local residual a-posteriori error estimates presented in Section 4.2 can be used as computable surrogates of the mismatch between the exact solution and its Galerkin approximation, in the corresponding norm.

Unlike the approach presented in Section 4.3.1, here we do not aim to construct a ReLU-NN to approximate the solution of the BIEs previously described by finding the optimal position of the mesh nodes, which in turn is driven by the minimization of a computable loss function. In turn, the algorithm presented herein aims to greedily select a set of basis functions to enrich the finite dimensional space upon which an approximation of the solution to the BIEs is built. We remark in passing that this approach is strongly related to the *adaptive basis* viewpoint elaborated in [11]. Due to Proposition 3.3, this amounts to increasing the width of the underlying ReLU-NN every time a new neuron, i.e. basis function, is added.

As in Section 4.2, we restrict our presentation to the case of a bounded Lipschitz polygon in $\mathbb{R}^2$ and, again, point out that the extension to an open arc is straightforward. The technique to be presented here is an adaptation of the *orthogonal matching pursuit algorithm* [6], and is also motivated by the recent results in [1, 18], which strongly leverage the *variational structure* of the discretization scheme.

Let $\mathcal{S} \subset H^{\frac{1}{2}}(\Gamma)$ denote a finite dimensional space of functions on Γ and let $\{\zeta_n\}_{n=1}^N$ denote a basis of $S^1(\Gamma, \mathcal{T}_N)$. For each $\xi \in \mathcal{S}$, $\operatorname{span}\left(\{\zeta_n\}_{n=1}^N \cup \{\xi\}\right)$ is itself a valid finite dimensional space on which a solution to Problem 2.15 may be sought. For given $\mathcal{S} \subset H^{\frac{1}{2}}(\Gamma)$, we aim to determine the element $\xi \in \mathcal{S}$ having the *least angle* with respect to the residual $\varphi - \varphi_N$. We aim at finding $\phi^\star \in \mathcal{S}$ such that

$$\phi^\star = \operatorname*{arg\,max}_{\phi \in \mathcal{S}} \frac{\left|(\varphi - \varphi_N, \phi)_{H^{\frac{1}{2}}(\Gamma)}\right|}{\|\phi\|_{H^{\frac{1}{2}}(\Gamma)}}. \tag{4.14}$$

However, in the computation of $\phi^\star \in \mathcal{S}$ in (4.14) we encounter the following difficulty: We can not directly compute the residual $\varphi - \varphi_N \in H^{\frac{1}{2}}(\Gamma)$. In view of Corollary 4.6, we use

$$h_{\mathcal{T}_N}^{\frac{1}{2}} \left(\mathsf{W}\varphi_N - g\right) \in L^2(\Gamma), \tag{4.15}$$

as a surrogate of the exact residual $\varphi - \varphi_N$ in the $H^{\frac{1}{2}}(\Gamma)$-norm. We proceed to find an element in $\mathcal{S}$ such that its contribution to (4.15) has the least angle, in the $L^2(\Gamma)$-inner product, with the residual (4.15) itself, i.e., find $\phi^\star \in \mathcal{S}$ such that

$$\phi^\star = \operatorname*{arg\,max}_{\phi \in \mathcal{S}} \frac{\left(h_{\mathcal{T}_N}^{\frac{1}{2}} \left(\mathsf{W}\varphi_N - g\right), h_{\mathcal{T}_N}^{\frac{1}{2}} \mathsf{W}\phi\right)_{L^2(\Gamma)}}{\left\|h_{\mathcal{T}_N}^{\frac{1}{2}} \mathsf{W}\phi\right\|_{L^2(\Gamma)}}. \tag{4.16}$$

To properly state an algorithm that allows us to construct a ReLU-NN, it remains to define how the set $\mathcal{S}$ is constructed at each step. As in Section 2.7, we consider a mesh $\mathcal{T}_N$ of Γ with $N+1$ points $\{\mathbf{x}_n\}_{n=0}^N \subset \mathbb{R}^2$, and where $\mathbf{x}_0 = \mathbf{x}_N$ (i.e. N distinct points). Let $\mathbf{x}'_n$ denote the midpoint between $\mathbf{x}_n$ and $\mathbf{x}_{n+1}$ for each $n = 0, \ldots, N-1$, and consider the set of N piecewise linear functions $\mathcal{S}_N \coloneqq \{\xi_n\}_{n=0}^N \subset H^{\frac{1}{2}}(\Gamma)$, defined as

$$\xi_n(\mathbf{x}) \coloneqq \begin{cases} 1, & \mathbf{x} = \mathbf{x}'_n, \\ 0, & \mathbf{x} \neq \mathbf{x}'_k \text{ or } \mathbf{x} \notin \{\mathbf{x}_k\}_{k=0}^N, \\ \text{linear elsewhere,} \end{cases} \tag{4.17}$$

for $n = 0, \ldots, N-1$. In this case, the set $\mathcal{S}_N$ gathers the piecewise linear functions one would add to $S^1(\Gamma, \mathcal{T}_N)$ if a uniform refinement of the mesh $\mathcal{T}_N$ of Γ were to be performed. We aim to select, among the candidates in $\mathcal{S}_N$, a function according to (4.16). Furthermore, since the incorporation of a single basis function of the set $\mathcal{S}_N$ at each step may result in a needlessly expensive procedure, we allow for

the incorporation of a subset of $\mathcal{S}_N$ at each step in order to enhance the procedure (as in the ABEM algorithm). We do so by computing

$$q_n := \frac{\left(h_{\mathcal{T}_N}^{\frac{1}{2}}\left(\mathsf{W}\varphi_N - g\right), h_{\mathcal{T}_N}^{\frac{1}{2}}\mathsf{W}\xi_n\right)_{L^2(\Gamma)}}{\left\|h_{\mathcal{T}_N}^{\frac{1}{2}}\mathsf{W}\xi_n\right\|_{L^2(\Gamma)}}, \quad n = 0, \ldots, N-1,$$

and then selecting a number of elements of $\mathcal{S}_N$ having the largest values of q_n. The number of elements of $\mathcal{S}_N$ incorporated at each iteration is controlled by an input a parameter $\theta \in (0, 1]$ specifying the fraction of elements of $\mathcal{S}_N$ to be included in our enhanced finite dimensional space at each iteration. The procedure is presented in Algorithm 3 for the setting of Problem 2.15 (recall the notation $\boldsymbol{t}$ and $\boldsymbol{c}$ denoting the biases and weights of the ReLU-NN). Note that Algorithm 3 may be implemented in tandem with Algorithm 1, as presented in Algorithm 4.

Algorithm 3 Construction of a sequence of ReLU networks minimizing the weighted residual estimator

Input: Initial set of biases $\boldsymbol{t} \in \mathbb{R}^N$; cardinality parameter $\theta \in (0, 1]$; maximum number of iterations $K \in \mathbb{N}$.

Output: Optimal Numerical Approximation $(\boldsymbol{t}_{\mathrm{F}}^\star, \mathbf{c}^\star)$.

procedure SEQUENCE_RELU_WEIGHTED_RESIDUAL_ESTIMATORS($\boldsymbol{t}$,θ,K)
  for $\{j = 1,\ j \leq K,\ j = j + 1\}$ **do**
    Compute $\mathbf{c} \in \mathbb{R}^{N+1}$ and $\phi(\boldsymbol{t}, \mathbf{c})$ by solving Problem 2.15 on the corresponding mesh of Γ;
    $\boldsymbol{q} \leftarrow \left\{ \frac{1}{\|\mathsf{W}\xi_n\|_{L^2(\Gamma)}} \left(h_{\mathcal{T}_N}^{\frac{1}{2}}\left[\mathsf{W}(\tau_{\boldsymbol{r}}^{-1}\phi(\boldsymbol{t}, \mathbf{c})) - g\right], \mathsf{W}\xi_n\right)_{L^2(\Gamma)} \right\}_{n=0}^{N-1}$ with ξ_n as in (4.17);
    $I_N \leftarrow$ Set of the $\lceil \theta N \rceil$ nodes $\{\tilde{\mathbf{x}}'_n\}_{n=1}^{\lceil \theta N \rceil} \subset \Gamma$ associated with the highest values in $\boldsymbol{q}$ (where $\xi_n(\tilde{\mathbf{x}}'_n) = 1$ as in (4.17));
    $\boldsymbol{t} \leftarrow$ include the biases $\{t_n\}_{n=1}^{\lceil \theta N \rceil} \subset \mathrm{I}$ associated with $\{\tilde{\mathbf{x}}'_n\}_{n=1}^{\lceil \theta N \rceil} \subset \Gamma$ through $\boldsymbol{r} : \mathrm{I} \to \Gamma$;
    $N \leftarrow N + \lceil \theta N \rceil$;
    $\boldsymbol{t}_j^\star \leftarrow \boldsymbol{t}$;
    Compute $\mathbf{c}_j^\star \in \mathbb{R}^N$ by solving Problem 2.15 on the mesh associated with the biases $\boldsymbol{t}_j^\star$;
  end for
  return $\{(\mathbf{t}_j^\star, \mathbf{c}_j^\star)\}_{j=1}^K$;
end procedure

Algorithm 4 Combination of Algorithms 1 and 3

Input: Initial set of biases $\boldsymbol{t} \in \mathbb{R}^N$; cardinality parameter $\theta \in (0, 1]$; maximum number of iterations $K \in \mathbb{N}$; maximum number of inner iterations $M \in \mathbb{N}$; tolerance $\epsilon > 0$.

Output: Sequence of optimal network parameters $\{(\boldsymbol{t}_j^\star, \mathbf{c}_j^\star)\}_{j=1}^K$.

procedure SEQUENCE_RELU_COMBINATION($\boldsymbol{t}$,θ,K,M,ϵ)
  for $\{j = 1,\ j < K,\ j = j + 1\}$ **do**
    $(\boldsymbol{t}_j^\star, \mathbf{c}_j^\star) \leftarrow$ ReLU_Optimization_Total_Energy($\boldsymbol{t}, M, \epsilon$);
    $\boldsymbol{q} \leftarrow \left\{ \frac{1}{\|\mathsf{W}\xi_n\|_{L^2(\Gamma)}} \left(h_{\mathcal{T}_N}^{\frac{1}{2}}\left[\mathsf{W}(\tau_{\boldsymbol{r}}^{-1}\phi(\boldsymbol{t}_j^\star, \mathbf{c}_j^\star)) - g\right], \mathsf{W}\xi_n\right)_{L^2(\Gamma)} \right\}_{n=0}^{N-1}$ with ξ_n as in (4.17);
    $I_N \leftarrow$ Set of the $\lceil \theta N \rceil$ nodes $\{\tilde{\mathbf{x}}'_n\}_{n=1}^{\lceil \theta N \rceil} \subset \Gamma$ associated with the highest values in $\boldsymbol{q}$ (where $\xi_n(\tilde{\mathbf{x}}'_n) = 1$ as in (4.17));
    $\boldsymbol{t} \leftarrow \boldsymbol{t}_j^\star$ together with the biases $\{t_n\}_{n=1}^{\lceil \theta N \rceil} \subset \mathrm{I}$ associated with $\{\tilde{\mathbf{x}}'_n\}_{n=1}^{\lceil \theta N \rceil} \subset \Gamma$ through $\boldsymbol{r} : \mathrm{I} \to \Gamma$;
    $N \leftarrow N + \lceil \theta N \rceil$;
  end for
  $(\boldsymbol{t}_K^\star, \mathbf{c}_K^\star) \leftarrow$ ReLU_Optimization_Total_Energy($\boldsymbol{t}, M, \epsilon$);
  return $\{(\boldsymbol{t}_j^\star, \mathbf{c}_j^\star)\}_{j=1}^K$
end procedure

Finally, note that setting $\theta = 1$ on Algorithm 4 results in Algorithm 2, while setting $M = 0$ on Algorithm 4 results in Algorithm 3.

5. Numerical Results

In this section we present numerical results obtained using the algorithms described in Section 4.

5.1. **Setting.** For the sake of simplicity, we provide results only for the setting described in Problem 2.12 on the so-called *slit* $\Lambda := (-1,1)\times\{0\} \subset \mathbb{R}^2$. For the numerical implementation of the BEM and the evaluation of the weighted residual estimators we use the `MATLAB` based [25] BEM library `HILBERT` [2] and custom `Fortran` codes. We remark that, in the numerical examples presented in Sections 5.2 and 5.3, we also compare the performance of Algorithm 4 to the built-in adaptive strategies implemented in `HILBERT`. These codes have been made available in `https://gitlab.epfl.ch/fhenriqu/relugalerkinbem`.

5.2. **Example I.** We consider Problem 2.12 with right-hand side $g = 1$ on Λ. The exact solution to this problem in Λ is known and given by $\phi(\mathbf{x}) = 2\sqrt{1-x_1^2}$, $\mathbf{x} = (x_1, x_2)^\top \in \Lambda$ (*cf.* [21, Section 4.2]). As a consequence of Lemma 4.1, ℓ_Λ in (4.3) achieves its minimum at ϕ, with value

$$\ell_\Lambda^\star := \ell_\Lambda(\phi) = -\frac{1}{2}\int_\Lambda g(\mathbf{x})\phi(\mathbf{x})\,\mathrm{d}s_{\mathbf{x}} = -\frac{1}{2}\int_{-1}^{1}\sqrt{1-x^2}\,\mathrm{d}x = -\frac{\pi}{2}. \tag{5.1}$$

We begin by considering Algorithm 1 on four different inital configurations of the biases $\boldsymbol{t} \in \mathbb{R}^{N+2}$ corresponding to $N \in \{8, 32, 128, 512\}$ equidistant free biases $\boldsymbol{t}_F$ on $\mathrm{I} = [-1,1]$. Figure 2 portrays the evolution of both the loss function ℓ_Λ and the error in the $\widetilde{H}^{\frac{1}{2}}(\Lambda)$–norm to the exact solution $2\sqrt{1-x_1^2}$ attained by the ReLU-NN generated by Algorithm 1 throughout $M = 10^4$ iterations. The decrease of the loss function displayed in Figure 2 is accompanied by the decrease of the error in the $\widetilde{H}^{\frac{1}{2}}(\Lambda)$–norm to the exact solution, as indicated by Lemma 4.2.

Then, we consider Algorithm 2 with an initial configuration of only one free bias ($N_0 = 1$), located at $t_{F,1} = 0$, and display the evolution of the loss function ℓ_Λ and the $\widetilde{H}^{\frac{1}{2}}(\Lambda)$–error attained by the NN on Figure 3, where $M = 10^4$ inner iterations of Algorithm 1 are considered, together with a tolerance of $\epsilon = 10^{-15}$ and $K = 8$ outer iterations. The convergence of the loss function to its minimum value is compared to that of Algorithm 2 with no optimization of the position of the biases (which is analogous to solving Problem 2.17 on meshes with a uniform refinement), for which we expect a convergence rate $\mathcal{O}(N^{-1})$ (the double of the expected convergence rate of the $\widetilde{H}^{\frac{1}{2}}(\Lambda)$–error, see Lemma 4.2). The convergence of the $\widetilde{H}^{\frac{1}{2}}(\Lambda)$–error, on the other hand, is compared to that attained by the ABEM algorithm. The trained NN displays convergence rates close to 3, for the loss function, and an EOC[1] 1.4 for the $\widetilde{H}^{\frac{1}{2}}(\Lambda)$–error (with respect to the number of free biases).

The performance of Algorithm 3 is compared with that of the ABEM algorithm in Figure 4, through the weighted residual estimator and $\widetilde{H}^{\frac{1}{2}}(\Lambda)$–error, for two different values of the parameter θ (0.25 and 0.5). The numerical results display a decay of the convergence rate for the value $\theta = 0.5$, due to the inclusion of suboptimal biases. For the value $\theta = 0.25$, on the other hand, an optimal convergence rate of 1.5 with respect to the free biases is observed.

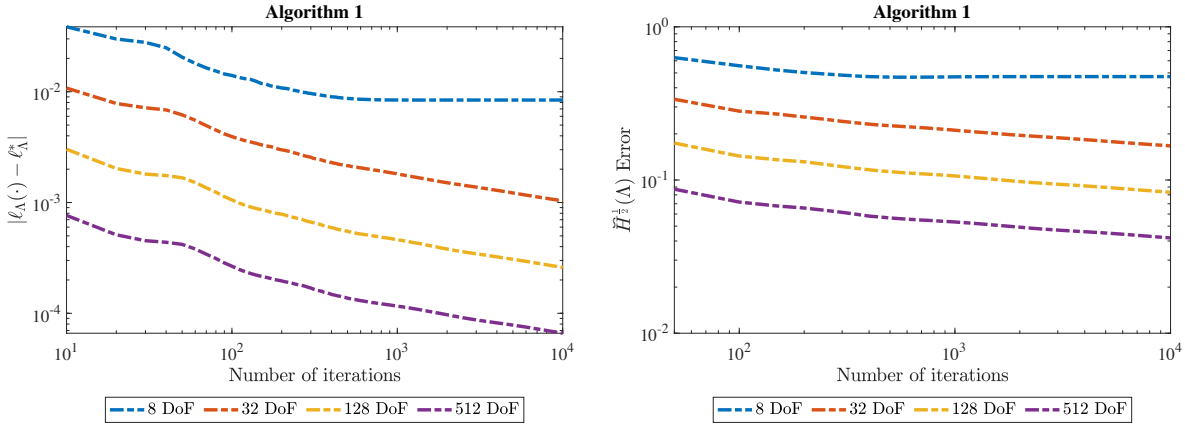
Finally, Figure 5 portrays the convergence of the $\widetilde{H}^{\frac{1}{2}}(\Lambda)$–norm for the ReLU-NNs returned by Algorithm 4, with $\theta \in \{0.25, 0.5\}$ and an initial configuration of one free bias, as before ($N_0 = 1$). Algorithm 1 was called with $M = 5000$ and $\epsilon = 10^{-15}$ and, on most iterations, fewer than 1000 inner iterations of Algorithm 1 were taken thanks to the stopping criterion. The results of Algorithm 4 are compared with those of Algorithm 3, and display how the optimization of the free bias through Algorithm 1 enhances the convergence rates obtained for $\theta = 0.5$, for which the optimal convergence rate of 1.5 (with respect to the number of free biases) is observed. The locations of the free biases generated by 8 iterations of Algorithm 4 are shown in Figure 6, where an accumulation towards the boundary of I, in order to accomodate to the singular behaviour of the solution, is discerned. In addition, in Figure 7 we show the computed solution on the meshes generated by Algorithm 4 together with the exact solution of this problem. Figures 7a 7b, 7c, and 7d portray the solution obtained on the meshes generated by Algorithm 4 with $N = 2, 4, 7, 11$, respectively.

5.3. **Example II.** Here, we consider Problem 2.12 but with the right-hand side

$$g(\mathbf{x}) = \begin{cases} -1, & \mathbf{x} \in (-1,0)\times\{0\}, \\ +1, & \mathbf{x} \in [0,1)\times\{0\}, \end{cases}, \quad \mathbf{x} \in \Lambda, \tag{5.2}$$

and repeat the numerical experiments presented in Section 5.2. That is to say, Figures 8, 9, 10 and 11 correspond to our implementations of Algorithms 1, 2, 3 and 4 exactly as in Section 5.2, but considering

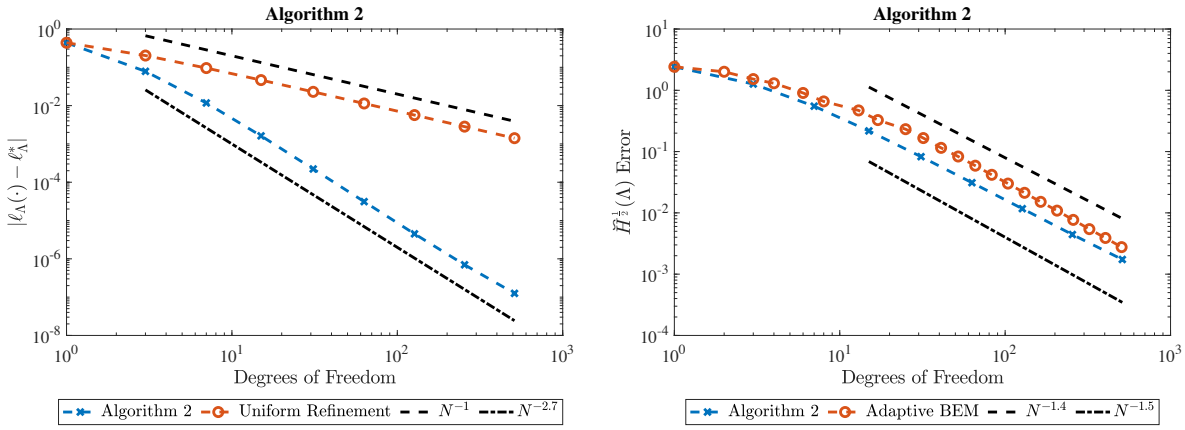
[1] Empirical Order of Convergence



(A) Evolution of the loss function ℓ_Λ.

(B) Evolution of the error in the $\widetilde{H}^{\frac{1}{2}}(\Lambda)$-norm.

FIGURE 2. Values of the loss function ℓ_Λ and $\widetilde{H}^{\frac{1}{2}}(\Lambda)$–error throughout the training process of Algorithm 1 for different values of N (i.e., number of degrees of freedom/free biases). The initial mesh (i.e. the biases in the hidden layer) was uniform for all instances. Excluding the mesh with 8 free nodes (which saturates before the 1000th iteration), a decrease of the loss function is observed to parallel the descent of the $\widetilde{H}^{\frac{1}{2}}(\Lambda)$–error of the solution of Problem 2.17 on each mesh.

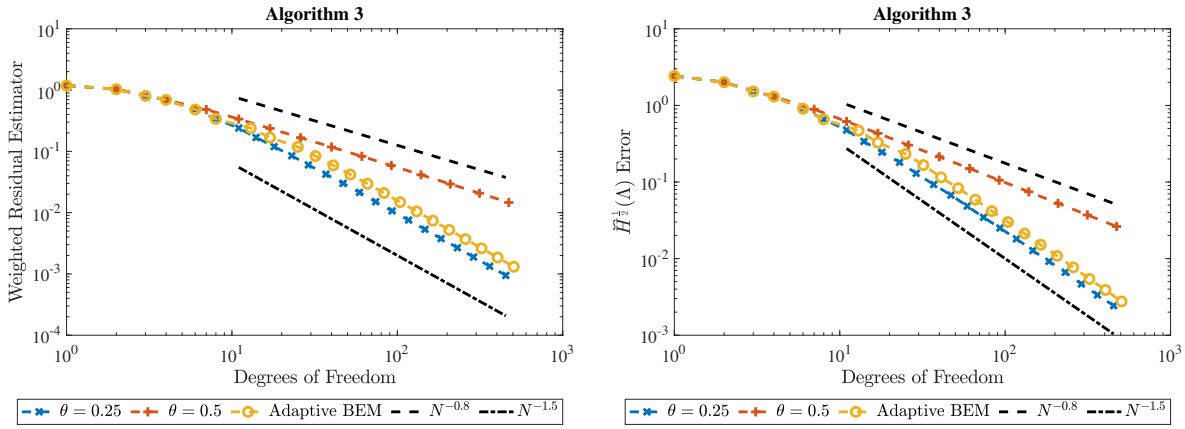


(A) Evolution of the loss function ℓ_Λ.

(B) Evolution of the $\widetilde{H}^{\frac{1}{2}}(\Lambda)$-error.

FIGURE 3. Evolution of the loss function ℓ_Λ and of the $\widetilde{H}^{\frac{1}{2}}(\Lambda)$–error throughout the training process of Algorithm 2 with respect to the number of free biases of each ReLU-NN (degrees of freedom of the associated mesh, see Proposition 3.3). On subfigure 3a, the difference between the loss function ℓ_Λ and its minimum value $-\frac{\pi}{2}$ (see (5.1) and Lemma 4.1) is compared for the sequence of meshes resulting from Algorithm 2 and for uniform mesh refinement. Substantially faster convergence of the training procedure to the optimum is observed for Algorithm 2 than for a uniform mesh refinement. Figure (3b) portrays the convergence in $\widetilde{H}^{\frac{1}{2}}(\Lambda)$–norm of the solutions to Problem 2.17 given by Algorithm 2 and by the adaptive BEM algorithm. The adaptive BEM algorithm achieves the optimal convergence rate of 1.5, while the sequence of meshes returned by Algorithm 2 attains a convergence rate of, approximately, 1.38 with respect to the number of degrees of freedom that are active in the ReLU-NN.

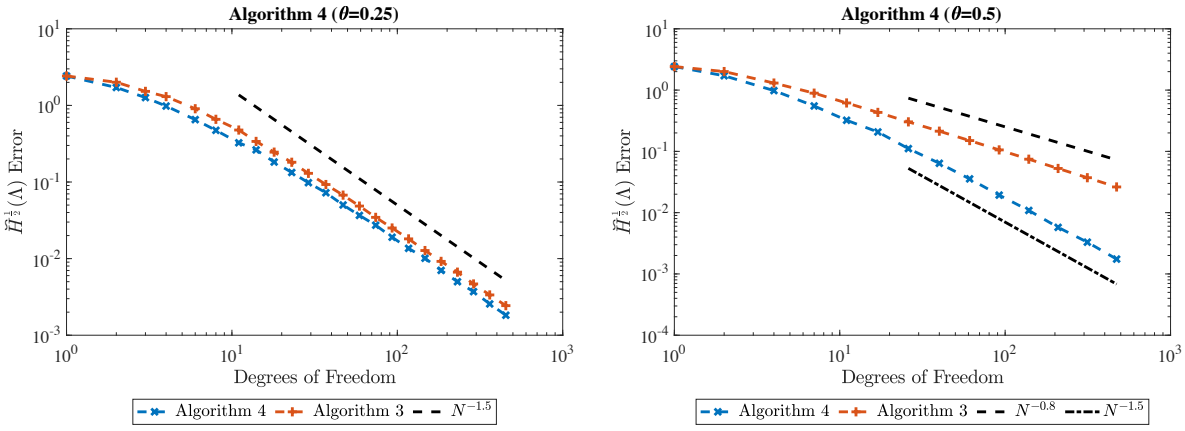
the right-hand side defined in (5.2). However, all initial configurations in this section require a fixed bias at the origin to accommodate for the discontinuity of g at $\mathbf{x} = (0, 0)$. Figure 12 shows the locations of



(A) Evolution of the weighted residual estimator.

(B) Evolution of the $H^{\frac{1}{2}}(\Lambda)$ error.

FIGURE 4. Evolution of the weighted residual estimator and $\widetilde{H}^{\frac{1}{2}}(\Lambda)$–error throughout the training process of Algorithm 3 and by the ABEM algorithm. The ABEM algorithm in the above figure corresponds to $\theta = 0.25$, and achieves, for the $\widetilde{H}^{\frac{1}{2}}(\Lambda)$-error in Figure 4b, the optimal convergence rate of 1.5 with respect to the number of degrees of freedom. On the other hand, Algorithm 3 achieves convergence rates of, approximately, 1.48 and 0.85 for $\theta = 0.25$ and $\theta = 0.5$, respectively. The convergence rates of the weighted residual estimators in Figure 4a follow a similar behavior.



(A) $\theta = 0.25$.

(B) $\theta = 0.5$.

FIGURE 5. Convergence in $\widetilde{H}^{\frac{1}{2}}(\Lambda)$-norm of the solutions of Problem 2.17 generated by Algorithms 3 and 4 for $\theta = 0.25$ and $\theta = 0.5$. The sequences of meshes returned by Algorithm 4 both achieve the optimal convergence rate of 1.5, independent of θ (though for $\theta = 0.25$ Algorithm 3 had already achieved a convergence rate close to 1.5). The figure displays how optimizing the positions of the free biases $\boldsymbol{t}_F$ through Algorithm 1 in between successive refinements by Algorithm 3 helps to improve convergence.

the biases generated by 8 iterations of Algorithm 4, where an accumulation at 0 due to the discontinuity of g, besides the expected accumulation at $\mathbf{x} = (-1, 0)$ and $\mathbf{x} = (1, 0)$, may be observed.

All errors and differences $|\ell_\Lambda - \ell_\Lambda^\star|$ displayed on the figures referenced on this section are computed with respect to an overkill solution (implemented in `HILBERT`) with $\ell_\Lambda^\star \approx -0.63662$, attained by Algorithm 4.

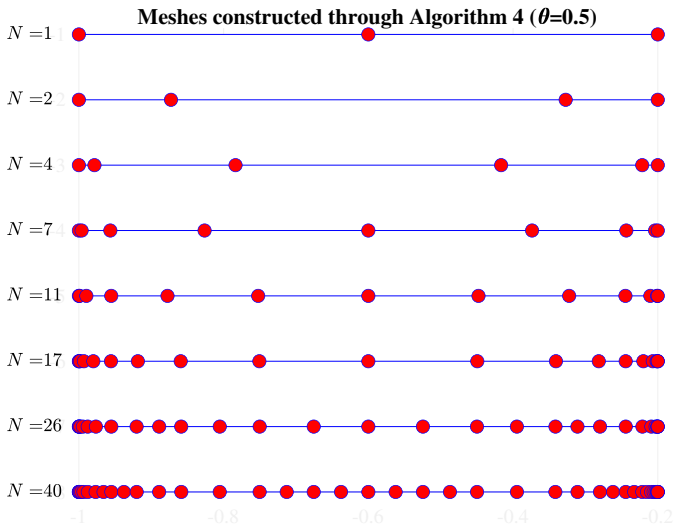


FIGURE 6. First 8 meshes generated by Algorithm 4.

6. Concluding Remarks and Outlook

We recapitulate principal findings of the present paper, and indicate extensions and possible directions for further research.

We developed a novel class of Galerkin boundary element methods, on polygonal domains $\mathrm{D} \subset \mathbb{R}^2$. They are based on trial spaces comprising deep neural networks with ReLU activation function.

Similar approaches are conceivable on boundaries of polyhedra $\mathrm{D} \subset \mathbb{R}^3$, using the fact that also surfaces, continuous first order Lagrangean boundary elements admit representations as ReLU NNs [19].

We investigated the approximation rates of corresponding BEM in dependence on the DNN architecture. Essentially, the size and structure of the triangulation is encoded in the NN architecture, with the location of the nodes being NN weights in the hidden layer of the NN.

We proved, also for singularities due to corners of $\partial\mathrm{D}$, that optimal algebraic convergence rates can be achieved with shallow ReLU DNN BEM, by suitable choice of NN weights and biases in the hidden layer. *Deep ReLU DNN trial spaces* will facilitate exponential convergence of the corresponding deep ReLU BEM, by emulating *hp*-boundary element methods. These can in principle achieve exponential rates of convergence, see [27].

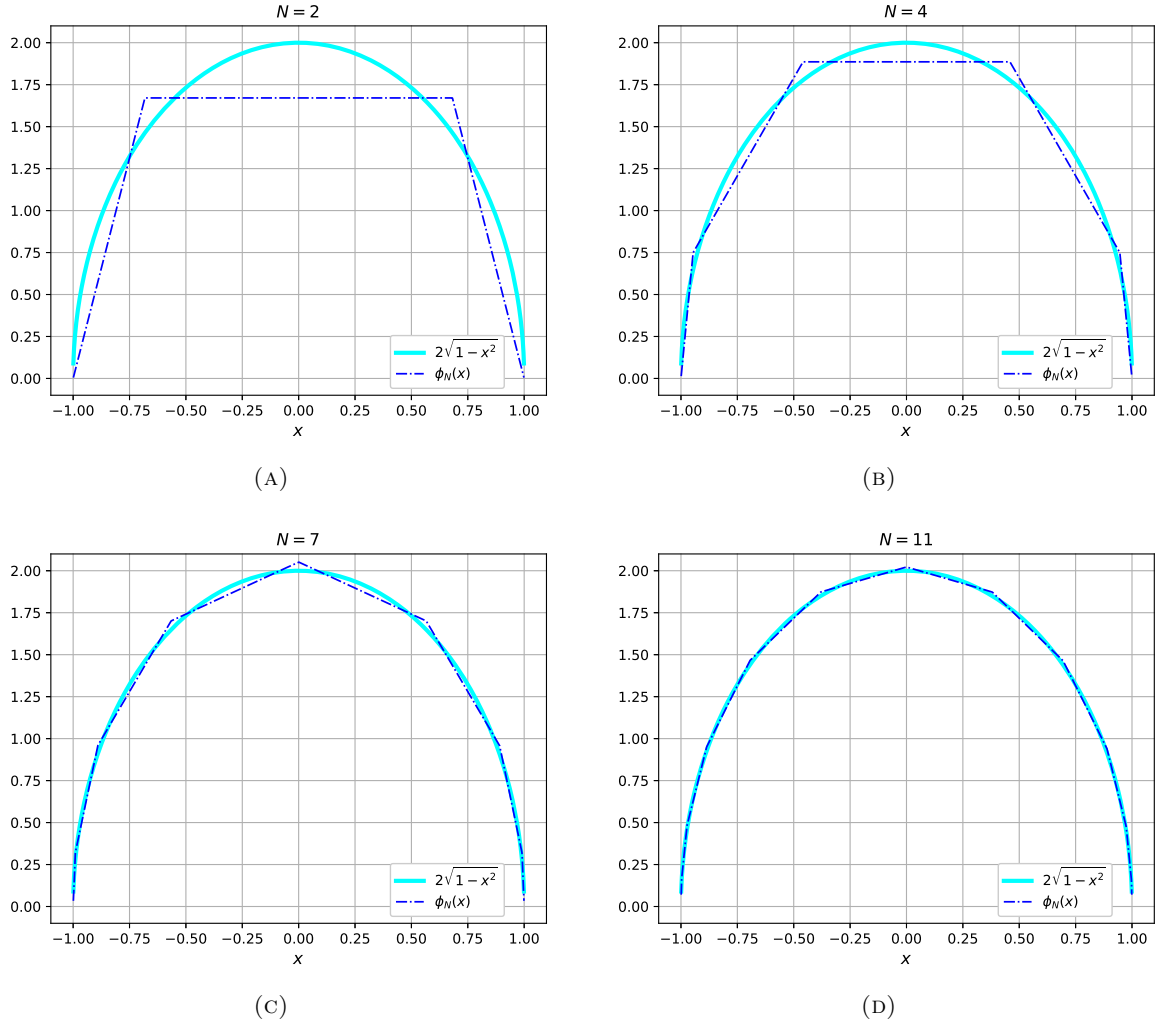
We proposed DNN training in the "natural" energy spaces being fractional, hilbertian Sobolev spaces on the boundary Γ which underlie the variational theory of first kind BIEs. While NN based discretizations have been proposed recently for PDEs, the nonlocal nature of the boundary integral operators renders efficient numerical evaluation of loss functions costly. We leveraged existing, computable local residual a-posteriori error estimators to obtain novel, computationally efficient loss functions. They are based on local, reliable a-posteriori residual discretization error estimators.

The present exposition was developed for plane, polygonal domains. However, the ReLU DNN expression results extend also to polyhedral domains $\mathrm{D} \subset \mathbb{R}^3$ with boundaries comprising a finite number of plane faces Γ_j. Here, again exact expression results of ReLU DNNs for continuous, piecewise affine BEM spaces on Γ are available in [19]. Corresponding approximation results on corner- and edge-graded meshes on Γ (see, e.g., [17]) will hold.

The general principle described in the present work, namely that ReLU DNNs are capable of emulating a wide range of spline-based approximation spaces with, essentially, identical convergence rate bounds, extends well beyond the presently considered setting.

The presently proposed formulation of boundary integral equations with DNN - based approximation spaces can serve as a vehicle to leverage powerful machine learning methodologies for the numerical treatment of boundary integral equations. Here, one single, unifying ReLU DNN based construction of approximation spaces on Γ will allow to achieve performance of adaptive mesh refinements and exponential convergence of *hp*-BEM without any revision of implementations.

For more general BIEs arising, for example, in BIEs on polyhedral surfaces resulting from the boundary reduction of the time-harmonic Maxwell equations, it is well-known that Galerkin discretizations must be based on certain *compatible subspaces*. See, e.g., [5], and [20]. Also in these cases, DNNs which



(A) (B)

(C) (D)

FIGURE 7. Exact solution to Problem 2.12 with right-hand side $g = 1$ on the slit $\Lambda = (-1, 1) \times \{0\}$ together with the solution obtained with the mesh produced by Algorithm 4 for $N = 2, 4, 7, 11$ in Figures 7a 7b, 7c, and 7d, respectively.

are *structure preserving* can be constructed. We refer to [23] for a development of DNNs for De Rham compatible Finite Element spaces. Development of details for BIEs of, e.g., electromagnetic scattering is beyond the scope of the present work.

REFERENCES

[1] M. Ainsworth and J. Dong. Galerkin neural networks: a framework for approximating variational equations with error control. *SIAM J. Sci. Comput.*, 43(4):A2474–A2501, 2021.

[2] M. Aurada, M. Ebner, M. Feischl, S. Ferraz-Leite, T. Führer, P. Goldenits, Michael Karkulik, M. Mayr, and D. Praetorius. HILBERT: a MATLAB implementation of adaptive 2D-BEM. *Numerical Algorithms*, 67(1):1–32, 2014.

[3] I. Babuška, B. Q. Guo, and E. P. Stephan. On the exponential convergence of the *h-p* version for boundary element Galerkin methods on polygons. *Math. Methods Appl. Sci.*, 12(5):413–427, 1990.

[4] A. R. Barron, A. Cohen, W. Dahmen, and R. A. DeVore. Approximation and learning by greedy algorithms. *The annals of statistics*, 36(1):64–94, 2008.

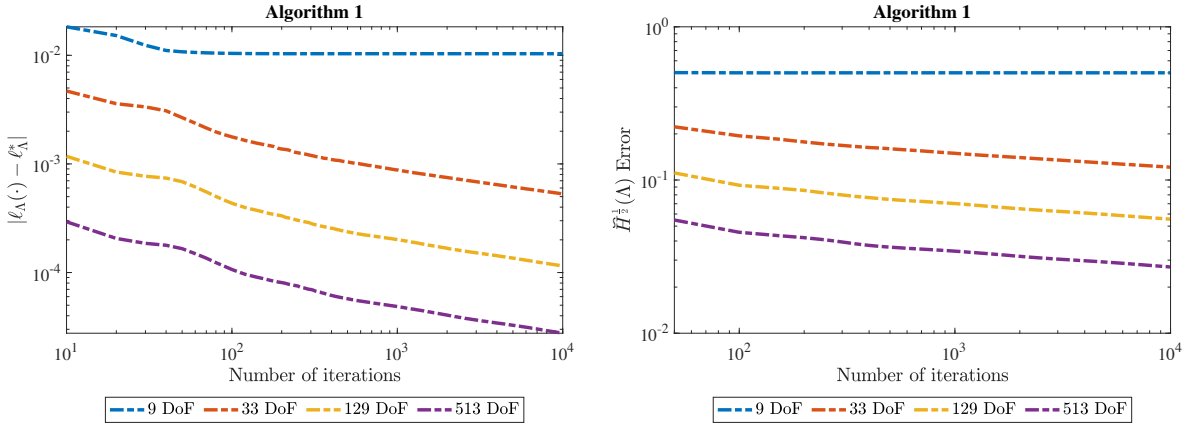
[5] A. Buffa, R. Hiptmair, T. von Petersdorff, and C. Schwab. Boundary element methods for Maxwell transmission problems in Lipschitz domains. *Numer. Math.*, 95(3):459–485, 2003.

[6] T. T. Cai and L. Wang. Orthogonal matching pursuit for sparse signal recovery with noise. *IEEE Transactions on Information Theory*, 57(7):4680–4688, 2011.

[7] C. Carstensen. An a posteriori error estimate for a first-kind integral equation. *Mathematics of Computation*, 66(217):139–155, 1997.

[8] C. Carstensen and E. P. Stephan. A posteriori error estimates for boundary element methods. *Mathematics of Computation*, 64(210):483–500, 1995.

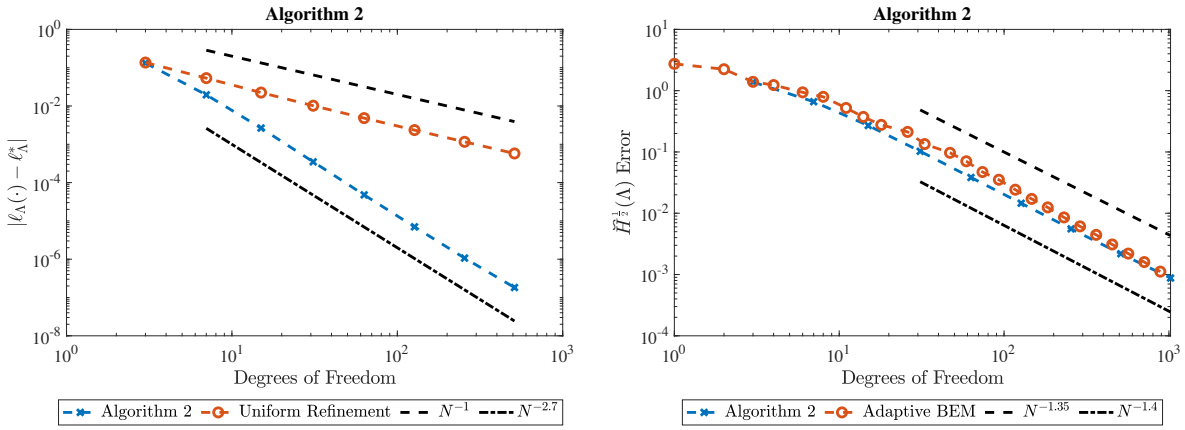
[9] C. Carstensen and E. P. Stephan. Adaptive boundary element methods for some first kind integral equations. *SIAM journal on numerical analysis*, 33(6):2166–2183, 1996.



(A) Evolution of the loss function ℓ_Λ.

(B) Evolution of the $H^{\frac{1}{2}}(\Lambda)$ error.

FIGURE 8. Evolution of the loss function ℓ_Λ and $\widetilde{H}^{\frac{1}{2}}(\Lambda)$–error throughout the training process of Algorithm 1 for different values of N. The initial mesh was uniform for all instances. As in Figure 2, a reduction of the loss function is observed together with a reduction $\widetilde{H}^{\frac{1}{2}}(\Lambda)$–error of the solution of Problem 2.17 on each mesh (excluding the mesh with 9 degrees of freedom, that saturates before 10^2 iterations).



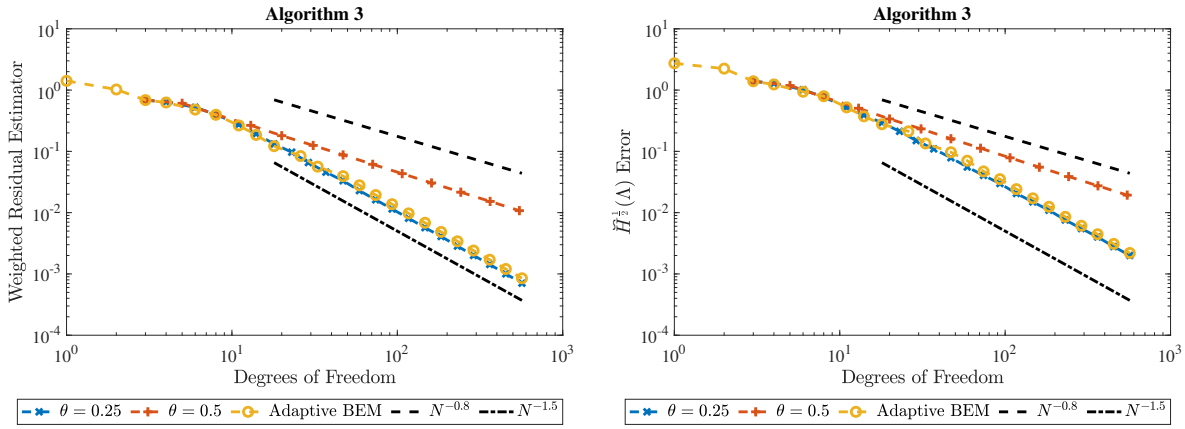
(A) Evolution of the loss function ℓ_Λ.

(B) Evolution of the $\widetilde{H}^{\frac{1}{2}}(\Lambda)$ error.

FIGURE 9. Evolution of the loss function ℓ_Λ and of the $\widetilde{H}^{\frac{1}{2}}(\Lambda)$–error throughout the training process of Algorithm 2 with an initial uniform mesh given by $N_0 = 3$ (i.e., 5 nodes and 4 elements). Subfigure 9a displays the difference between the loss function ℓ_Λ and its minimum attained value $\ell_\Lambda^\star \approx -0.63662$ for the sequence of meshes resulting from Algorithm 2 and for uniform mesh refinement. Again, faster convergence to the optimum is observed for Algorithm 2 than for uniform mesh refinement. Subfigure 9b shows the convergence in $\widetilde{H}^{\frac{1}{2}}(\Lambda)$–norm of the solutions to Problem 2.17 for Algorithm 2 and the adaptive BEM algorithm. The adaptive BEM algorithm achieves the optimal convergence rate of 1.5, while Algorithm 2 attains a convergence rate of 1.35 with respect to the degrees of freedom active in the ReLU-NN.

[10] M. Costabel. Boundary integral operators on Lipschitz domains: elementary results. *SIAM J. Math. Anal.*, 19(3):613–626, 1988.

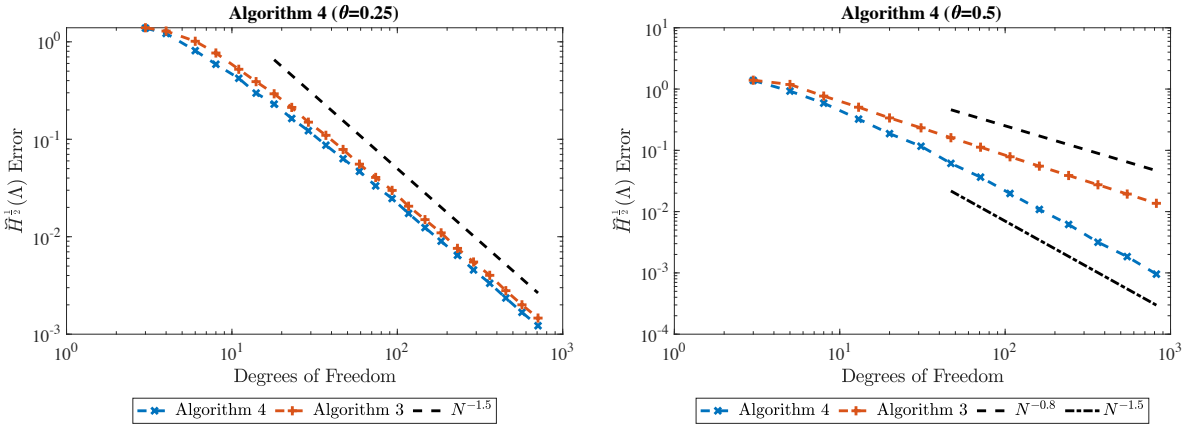
[11] E. C. Cyr, M. A. Gulian, R. G. Patel, M. Perego, and N. A. Trask. Robust training and initialization of deep neural networks: An adaptive basis viewpoint. In *Mathematical and Scientific Machine Learning*, pages 512–536. PMLR, 2020.



(A) Evolution of the weighted residual estimator.

(B) Evolution of the $H^{\frac{1}{2}}(\Lambda)$ error.

FIGURE 10. Evolution of the weighted residual estimator and $\widetilde{H}^{\frac{1}{2}}(\Lambda)$–error throughout the training process of Algorithm 3 and by the ABEM algorithm. Two values of θ were considered for the implementation of Algorithm 3 (0.25 and 0.5), while the ABEM algorithm considered $\theta = 0.5$. The initial mesh was uniform and consisted of 3 nodes (2 elements). The ABEM algorithm achieves, for both the weighted residual estimator in subfigure 10a and for the $\widetilde{H}^{\frac{1}{2}}(\Lambda)$–error in Figure 10b, the optimal convergence rate of 1.5 with respect to the number of degrees of freedom, while Algorithm 3 achieves convergence rates of, approximately, 1.48 and 0.85 for $\theta = 0.25$ and $\theta = 0.5$, respectively.



(A) Evolution of the weighted residual estimator.

(B) Evolution of the error in the $H^{\frac{1}{2}}(\Lambda)$ norm.

FIGURE 11. Convergence in the $\widetilde{H}^{\frac{1}{2}}(\Lambda)$–norm of the solutions of Problem 2.17 on the sequence of meshes generated by Algorithms 3 and 4 for $\theta = 0.25$ and $\theta = 0.5$. The sequences of meshes returned by Algorithm 4 both achieve the optimal convergence rate of 1.5, independent of θ (though for $\theta = 0.25$ Algorithm 3 had already achieved the a convergence rate close to the optimum). The figure displays how optimizing the mesh nodes through Algorithm 1 in between successive refinements by Algorithm 3 helps improve convergence. Here, Algorithm 1 was called with $M = 5 \cdot 10^3$ and $\epsilon = 10^{-15}$ and, on most meshes, fewer than 10^3 iterations were required thanks to the stopping criterion.

[12] R. A. DeVore and V. N. Temlyakov. Some remarks on greedy algorithms. *Advances in computational Mathematics*, 5(1):173–187, 1996.

[13] C. Duan, Y. Jiao, Y. Lai, X. Lu, and Z. Yang. Convergence rate analysis for deep Ritz method. *arXiv preprint arXiv:2103.13330*, 2021.

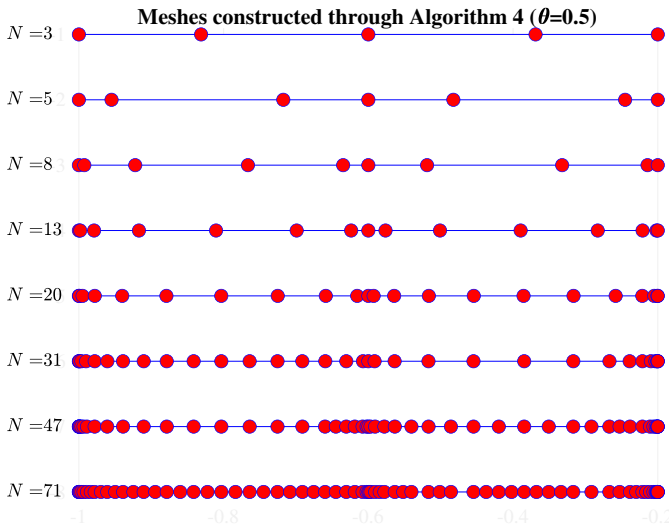


FIGURE 12. First 8 meshes generated by Algorithm 4. The non symmetric behaviour of the positions of the biases is due to $\lceil N\theta \rceil$ being an odd integer for some iterations of the Algorithm. This results, in this case, in more biases on the interval $(-1, 0)$ than on $(0, 1)$

.

[14] W. E and B. Yu. The deep Ritz method: a deep learning-based numerical algorithm for solving variational problems. *Commun. Math. Stat.*, 6(1):1–12, 2018.
[15] M. Feischl, T. Führer, N. Heuer, M. Karkulik, and D. Praetorius. Adaptive boundary element methods. *Archives of Computational Methods in Engineering*, 22(3):309–389, 2015.
[16] B. Guo and N. Heuer. The optimal rate of convergence of the p-version of the boundary element method in two dimensions. *Numerische Mathematik*, 98(3):499–538, 2004.
[17] J. Gwinner and E. P. Stephan. *Advanced boundary element methods*, volume 52 of *Springer Series in Computational Mathematics*. Springer, Cham, 2018. Treatment of boundary value, transmission and contact problems.
[18] W. Hao, X. Jin, J. W. Siegel, and J. Xu. An efficient greedy training algorithm for neural networks and applications in pdes. *arXiv preprint arXiv:2107.04466*, 2021.
[19] J. He, L. Li, J. Xu, and C. Zheng. ReLU Deep Neural Networks and Linear Finite Elements. *Journal of Computational Mathematics*, 38, 2020.
[20] George C. Hsiao and Wolfgang L. Wendland. *Boundary integral equations*, volume 164 of *Applied Mathematical Sciences*. Springer, Cham, [2021] ©2021. Second edition [of 2441884].
[21] C. Jerez-Hanckes and J.C. Nédélec. Explicit variational forms for the inverses of integral logarithmic operators over an interval. *SIAM Journal on Mathematical Analysis*, 44(4):2666–2694, 2012.
[22] Y. Jiao, Y. Lai, Y. Luo, Y. Wang, and Y. Yang. Error Analysis of Deep Ritz Methods for Elliptic Equations. *arXiv preprint arXiv:2107.14478*, 2021.
[23] M. Longo, J. A. A. Opschoor, N. Disch, Ch. Schwab, and J. Zech. De Rham compatible Deep Neural Networks. Technical Report 2022-03, Seminar for Applied Mathematics, ETH Zürich, Switzerland, 2022.
[24] 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 *Conference on Learning Theory*, pages 3196–3241. PMLR, 2021.
[25] MATLAB. *version 9.7.0 (R2019b)*. The MathWorks Inc., Natick, Massachusetts, 2019.
[26] W. McLean. *Strongly elliptic systems and boundary integral equations*. Cambridge University Press, Cambridge, 2000.
[27] J. A. A. Opschoor, P. C. Petersen, and Ch. Schwab. Deep ReLU networks and high-order finite element methods. *Anal. Appl. (Singap.)*, 18(5):715–770, 2020.
[28] A. Pinkus. Approximation theory of the MLP model in neural networks. In *Acta numerica, 1999*, volume 8 of *Acta Numer.*, pages 143–195. Cambridge Univ. Press, Cambridge, 1999.
[29] 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.*, 378:686–707, 2019.
[30] J. Saranen and G. Vainikko. *Periodic integral and pseudodifferential equations with numerical approximation*. Springer Science & Business Media, 2013.
[31] S. A. Sauter and Ch. Schwab. Boundary element methods, volume 39 of Springer Series in Computational Mathematics, 2011.
[32] Ch. Schwab and J. Zech. Deep learning in high dimension: neural network expression rates for generalized polynomial chaos expansions in UQ. *Anal. Appl. (Singap.)*, 17(1):19–55, 2019.
[33] J. W. Siegel and J. Xu. Improved Convergence Rates for the Orthogonal Greedy Algorithm. *arXiv preprint arXiv:2106.15000*, 2021.

[34] O. Steinbach. *Numerical approximation methods for elliptic boundary value problems: finite and boundary elements*. Springer Science & Business Media, 2007.
[35] E. P. Stephan and W. L. Wendland. An augmented Galerkin procedure for the boundary integral method applied to two-dimensional screen and crack problems. *Applicable Analysis*, 18(3):183–219, 1984.
[36] R. K. Tripathy and I. Bilionis. Deep UQ: learning deep neural network surrogate models for high dimensional uncertainty quantification. *J. Comput. Phys.*, 375:565–588, 2018.
[37] T. von Petersdorff and Ch. Schwab. Wavelet approximations for first kind boundary integral equations on polygons. *Numer. Math.*, 74(4):479–516, 1996.
[38] W. L. Wendland and E. P. Stephan. A hypersingular boundary integral method for two-dimensional screen and crack problems. *Arch. Rational Mech. Anal.*, 112(4):363–390, 1990.

Appendix A. Proof of Proposition 3.4

We only prove the second case in (3.7) which gives the maximum rate of convergence. We consider separately the regular part u_0 and the singularities x^{λ_j}. As in the statement of Proposition 3.4, we assume $\beta \geq 1$, $0 < s \leq 1$ (we will mainly need the case $s = 1/2$). We write x_k in place of $x_k^{N,\beta} = (k/N)^\beta \in [0,1]$ for $k = 0, 1, ..., N$. The assumptions of Proposition 3.4 imply that $u \in \mathscr{C}(\mathrm{I})$. Hence the linear interpolant I_N^β of u on $\mathcal{T}_{N,\beta}$ is well-defined and nodally exact: $(u - I_N^\beta u)(x_k) = 0$ for $k = 0, 1, ..., N$. Recall that $u_0 \in H^2(I)$, and that $0 \leq s \leq 1$. On each $\mathrm{I}_k = (x_{k-1}, x_k) \in \mathcal{T}_{N,\beta}$,

$$\|v_0 - S_0\|_{\widetilde{H}^s(\mathrm{I}_k)} \leq C h_k^{2-s} \|v_0''\|_{L^2(\mathrm{I}_k)} .$$

Here, $\|\cdot\|_{\widetilde{H}^s(\mathrm{I}_k)}$ is the interpolation norm between $L^2(\mathrm{I}_k)$ and $H_0^1(\mathrm{I}_k)$, and $C > 0$ depends on $s \in [0,1]$, but not on h_k. For $s = 0, 1$, we get

$$\begin{aligned} \|u_0 - I_N^\beta u_0\|^2_{\widetilde{H}^s(\mathrm{I})} &\leq C \sum_{k=1}^N \|u_0 - I_N^\beta u_0\|^2_{\widetilde{H}^s(\mathrm{I}_k)} \leq C \sum_{k=1}^N h_k^{4-2s} \|u_0''\|^2_{L^2(\mathrm{I}_k)} \\ &\leq \max_{k=1,..,N} \{h_k\}^{4-2s} \sum_{k=1}^N \|u_0''\|^2_{L^2(\mathrm{I}_k)} \leq C h^{2(2-s)} \|u_0''\|^2_{L^2(\mathrm{I})} , \end{aligned}$$

and the general case ($s \in [0,1]$) follows by interpolation.

Consider now a singularity, i.e., $f(x) = x^\lambda$ with $\lambda \in \{\lambda_j\}$ and take $\tilde{\gamma} \in \mathbb{R}$ such that $1 - 1/\beta \leq \tilde{\gamma} \leq 1$. Assume, first, $k \geq 2$, so that $\mathrm{dist}(0, \mathrm{I}_k) > 0$. Then, we have that $f \in H^2(\mathrm{I}_k) \subset C(\overline{\mathrm{I}_k})$ and, therefore, the linear nodal interpolant of f on I_k is well-defined and $f - I_N^\beta f \in H_0^1(\mathrm{I}_k)$. Furthermore,

$$\sum_{k=2}^N \|f - I_N^\beta f\|^2_{\widetilde{H}^s(\mathrm{I}_k)} \leq C \sum_{k=2}^N h_k^{4-2s} \|f''\|^2_{L^2(\mathrm{I}_k)} \lesssim \sum_{k=2}^N h^{(1-\tilde{\gamma})\beta(4-2s)} x_k^{\tilde{\gamma}(4-2s)} \|f''\|^2_{L^2(\mathrm{I}_k)} ,$$

since, for $k = 1, \ldots, N$, it holds that

$$h_k \leq \beta h^\beta k^{\beta-1} = \beta x_k^{\tilde{\gamma}} h^{(1-\tilde{\gamma})\beta - 1} \leq \beta h^{(1-\tilde{\gamma})} x_k^{\tilde{\gamma}} .$$

For $k \geq 2$, and for any $x \in I_k = (x_{k-1}, x_k)$, we have that

$$x_k = (kh)^\beta = \left(\frac{k}{k-1}\right)^\beta x_{k-1} \leq 2^\beta x_{k-1} \leq 2^\beta x .$$

Therefore

$$\begin{aligned} \sum_{k=2}^N \|f - I_N^\beta f\|^2_{\widetilde{H}^s(\mathrm{I}_k)} &\leq C h^{(1-\tilde{\gamma})\beta(4-2s)} \sum_{k=2}^N \|f''(x) x^{(2-s)\tilde{\gamma}}\|^2_{L^2(\mathrm{I}_k)} \\ &\leq C h^{(1-\tilde{\gamma})\beta(4-2s)} \int_{x_1}^1 |f''(x)|^2 x^{2(2-s)\tilde{\gamma}} dx \\ &\leq C h^{(1-\tilde{\gamma})\beta(4-2s)} \int_0^1 |f''(x)|^2 x^{2(2-s)\tilde{\gamma}} dx . \end{aligned}$$

Since $f''(x) = \lambda(\lambda-1)x^{\lambda-2}$, the latter integral exists if

$$2\lambda - 4 + 2(2-s)\tilde{\gamma} > -1 \Longleftrightarrow 1 - \tilde{\gamma} < \frac{\lambda - s + 1/2}{2-s} .$$

To bound the contribution from $\mathrm{I}_1 = (0, x_1) = (0, h^\beta)$, we use a scaling argument. Let $0 < h < 1$ be arbitrary. Then, for any $g \in H_0^1(\mathrm{I})$ there holds that

$$\|g(x/x_1)\|_{L^2(\mathrm{I}_1)} \leq x_1^{1/2} \|g(x)\|_{L^2(\mathrm{I})}, \;\; |g(x/h)|_{H^1(\mathrm{I}_1)} \leq x_1^{-1/2} |g|_{H^1(\mathrm{I})} .$$

We interpolate (or directly estimate the fractional order seminorm) to obtain

$$\|g(x/x_1)\|_{\widetilde{H}^s(\mathrm{I}_1)} \leq x_1^{1/2-s}\|g(x)\|_{\widetilde{H}^s(\mathrm{I})}.$$

Taking $g(x) = (f - I_N^\beta f)(x_1 \cdot x)$, so that $g(x/x_1) := (f - I_N^\beta f)(x)$ on $\mathrm{I}_1 = (0, x_1)$, yields

$$\|(f - I_N^\beta f)(x)\|_{\widetilde{H}^s(\mathrm{I}_1)} \leq x_1^{1/2-s}\|(f - I_N^\beta f)(x_1 \cdot x)\|_{\widetilde{H}^s(\mathrm{I})} \leq x_1^{1/2-s+\lambda}\|f - \tilde{f}\|_{\widetilde{H}^s(0,1)} \leq C(\lambda, s)x_1^{1/2-s+\lambda},$$

where $\tilde{f}$ is the linear interpolant of f on I, so that $\|f - \tilde{f}\|_{\tilde{H}^s(\mathrm{I})}$ is independent of x_1. Inserting $\tilde{\gamma} = 1 - 1/\beta \Longleftrightarrow 1 - \tilde{\gamma} = 1/\beta$ into the constraints implies

$$1/\beta = 1 - \tilde{\gamma} < \frac{\lambda - s + 1/2}{2 - s}$$

which is the second case in (3.7). The first case is proved in the same fashion.