East Asian Journal on Applied Mathematics
doi: 10.4208/eajam.081121.161121

Vol. **12**, No. 3, pp. 487-502
August 2022

A Deep Learning Method for Elliptic Hemivariational Inequalities

Jianguo Huang[1], Chunmei Wang[2] and Haoqin Wang[1,*]

[1]*School of Mathematical Sciences, and MOE-LSC, Shanghai Jiao Tong University, Shanghai 200240, China.*
[2]*Department of Mathematics, University of Florida, Gainesville, FL 32611, USA.*

Received 8 November 2021; Accepted (in revised version) 16 November 2021.

Abstract. Deep learning method for solving elliptic hemivariational inequalities is constructed. Using a variational formulation of the corresponding inequality, we reduce it to an unconstrained expectation minimization problem and solve the last one by a stochastic optimization algorithm. The method is applied to a frictional bilateral contact problem and to a frictionless normal compliance contact problem. Numerical experiments show that for fine meshes, the method approximates the solution with accuracy similar to the virtual element method. Besides, the use of local adaptive activation functions improves accuracy and has almost the same computational cost.

AMS subject classifications: 65K15, 68T07, 68U99

Key words: Deep learning, elliptic hemivariational inequality, contact problem, mesh-free method.

1. Introduction

With the advance of deep learning technique originated in computer science, considerable efforts are made to use this approach in other areas, including numerical methods for partial differential equations (PDEs). Neural network-based numerical methods appeared in 1990s [29] and have been significantly improved recently [7, 8, 10, 25, 40, 43, 44, 47]. In those methods, deep neural networks (DNNs) are used in order to parameterize the PDE solution by the parameters defined by an optimization problem related to the PDE under consideration. The key to the success of neural networks-based methods lies in the universal approximation property [2, 9, 18, 20, 42]. It is well known that deep neural network is a powerful tool for solving high-dimensional PDEs — cf. [12, 13, 26, 36]. We are sure that this is also a valuable strategy to attack low-dimensional complicated problems in science and engineering, which are expensive to solve by traditional numerical methods. Following these ideas, Huang *et al.* [21] proposed deep learning-based methods for variational inequalities. This approach was supported by other authors [30, 41]. How-

*Corresponding author. *Email addresses:* jghuang@sjtu.edu.cn (J. Huang), chunmei.wang@ufl.edu (C. Wang), wanghaoqin@sjtu.edu.cn (H. Wang)

ever, to the best of our knowledge, there are no similar works on hemivariational inequalities (HVIs). We recall that HVIs, introduced in 1983 by Panagiotopoulos [38] in connection with engineering applications, have been rigorously studied [14, 37, 39] and are now widely used in contact mechanics. In practice, the solutions to elliptic HVIs are only available numerically. There are various numerical methods to approximate the solutions of elliptic HVIs — cf. [11, 15, 17, 46]. However, the discretization often leads to non-convex and non-smooth optimization problems, the solution of which is challenging. The iterative convexification [35, 45] is a popular method for solving the non-convex optimization problems mentioned. It consists in construction and solution of a number of convex problems approaching the original non-convex problem. Many HVIs related contact problems such as the frictional bilateral contact problem, the frictionless normal compliance contact problem and the frictionless unilateral contact problem have been solved by the method mentioned [16].

In addition to the iterative convexification approach, one can use the proximal bundle method [34], the bundle Newton method [33], and the primal-dual active-set algorithm [28]. Recently, Feng *et al.* [11] used the double bundle method [24] to solve discrete non-convex and non-smooth problems arising in the discretization of HVIs. Nevertheless, the numerical methods mentioned are rather difficult to implement or they are computationally expensive when applied to the corresponding discrete problems.

Here, we consider a deep learning method for an elliptic HVI. The method is based on an equivalent variational formulation of the problem [14]. In particular, the solution space of the HVI is parameterized by DNNs and an approximation is found by minimizing an unconstrained expectation minimization problem. The latter can be solved by stochastic gradient descent methods [3]. The unconstrained expectation minimization problem is reformulated by using the variational principle for HVIs. Therefore, the resulting deep learning optimization problem has a clear physical meaning. As applications, we employ the deep learning method to a frictional bilateral contact problem and a frictionless contact problem with normal compliance. In addition, we use a fixed activation function and a local adaptive activation function [23] to solve HVIs in numerical simulation. Numerical experiments also show that the deep learning method has the same accuracy as traditional numerical methods on fine grids. Besides, the use of local adaptive activation functions gives a better accuracy than fixed activation functions, under comparable computational cost. Finally, it is worth noting that the method is suitable for engineering applications and can be easily programmed in Python.

The rest of this paper is organized as follows. In Section 2, an elliptic HVI and its applications in contact mechanics are introduced. Section 3 provides a detailed description of the deep learning method for HVIs. In Section 4, two numerical examples demonstrate the efficiency of the deep learning methods. Finally, we summarize our work with a short conclusion in Section 5.

2. Elliptic Hemivariational Inequality and Applications

Let X be a real Banach space with norm $\|\cdot\|_X$ and X^* the dual of X with norm $\|\cdot\|_{X^*}$.

The notation $\langle\cdot,\cdot\rangle$ stands for the dual pairing between X^* and X. This naturally applies to a Hilbert space H. We denote by $\mathscr{L}(X,Y)$ the space of all continuous linear operators from a normed linear space X to a normed linear space Y and use the standard notation for Sobolev spaces, norms or seminorms [1]. For any locally Lipschitz continuous functional j on a Banach space X, let $j^0(u;w)$ refer to the generalized (Clarke) directional derivative of j at u in the direction w, i.e.

$$j^0(u;w) := \limsup_{v\to u,\ t\downarrow 0} \frac{j(v+tw)-j(v)}{t}, \quad u\in X, \quad w\in X,$$

cf. [5, 6]. The generalized subdifferential of j at $u\in X$ is defined by

$$\partial j(u) := \left\{\zeta\in X^* \mid j^0(u;w)\geq\langle\zeta,w\rangle,\ \forall w\in X\right\}.$$

2.1. Elliptic hemivariational inequality and equivalent minimization problem

Consider an elliptic HVI on a spatial domain Ω in a finite-dimensional Euclidean space. For simplicity, we denote the boundary or part of the boundary of the domain as Γ. Let H be a Hilbert space. The elliptic HVI can be described as follows.

Problem 2.1. *Find $u\in H$ such that*

$$\langle Au,v\rangle + \int_\Gamma j^0(\gamma_j u;\gamma_j v)\,\mathrm{d}s \geq \langle f,v\rangle, \quad v\in H, \tag{2.1}$$

where $\gamma_j\in\mathscr{L}(H,L^2(\Gamma;\mathbb{R}^m))$, $j:\Gamma\times\mathbb{R}^m\to\mathbb{R}$ is a locally Lipschitz continuous functional for a positive integer m and $j(x,\cdot)$ is locally Lipschitz on $\mathbb{R}^m$ for a.e. $x\in\Gamma$.

To simplify the notation $j(x,z)$, which may depend on the spatial variable x, we will usually write $j(z)$ with the understanding that it is allowed to depend on the spatial variable.

In the study of Problem 2.1, we need the following assumptions — cf. [14].

(A1) $A:H\to H^*$ is Lipschitz continuous and strongly monotone — i.e. there is a constant $m_A>0$ such that for any $v_1,v_2\in H$ the following inequality holds:

$$\langle Av_1-Av_2,v_1-v_2\rangle \geq m_A\|v_1-v_2\|_H^2.$$

(A2) $j(\cdot,z)$ is measurable on Γ, $z\in\mathbb{R}^m$. There are non-negative constants c_0, c_1, α_j and a function $z_0\in L^2(\Gamma;\mathbb{R}^m)$ such that $j(\cdot,z_0(\cdot))\in L^1(\Gamma)$ and

$$\begin{aligned}&\|\partial j(z)\|_{\mathbb{R}^m}\leq c_0+c_1\|z\|_{\mathbb{R}^m}, \quad z\in\mathbb{R}^m,\\ &j^0(z_1;z_2-z_1)+j^0(z_2;z_1-z_2)\leq\alpha_j\|z_1-z_2\|_{\mathbb{R}^m}^2, \quad z_1,z_2\in\mathbb{R}^m.\end{aligned}$$

(A3) $f\in H^*$.

(A4) Denote by c_Γ an upper bound of the norm of the operator γ_j. There holds

$$\|\gamma_j v\|_{L^2(\Gamma;\mathbb{R}^m)} \le c_\Gamma \|v\|_H, \quad v \in H.$$

Theorem 2.1 (cf. Han *et al.* [16]). *If $\alpha_j c_\Gamma^2 < m_A$ and the conditions* (A1)-(A4) *hold, then Problem* 2.1 *has a unique solution for any $f \in H^*$.*

Now we assume that $A \in \mathscr{L}(H, H^*)$ is symmetric — i.e.

$$\langle Av_1, v_2\rangle = \langle Av_2, v_1\rangle, \quad v_1, v_2 \in H$$

and consider the following minimization problem.

Problem 2.2. *Find $u \in H$ such that*

$$u = \arg\min_{v\in H} E(v), \tag{2.2}$$

where

$$E(v) = \frac{1}{2}\langle Av, v\rangle + \int_\Gamma j(\gamma_j v)\,\mathrm{d}s - \langle f, v\rangle, \quad v \in H.$$

Theorem 2.2 (cf. Han [14]). *If $A \in \mathscr{L}(H, H^*)$ is a symmetric matrix, $\alpha_j c_\Gamma^2 < m_A$, and conditions* (A1)-(A4) *hold, then Problem* 2.2 *is equivalent to Problem* 2.1.

Note that the functional $E(v)$ in (2.2) is reformulated from (2.1) based on the variational principle. Hence, it has clear physical meaning in mechanics.

2.2. Applications in contact mechanics

Let $\Omega \subset \mathbb{R}^2$ be the reference configuration of a linear elastic body. We assume that Ω is an open bounded connected domain with Lipschitz continuous boundary $\Gamma = \partial\Omega$. The boundary consists of three disjoint measurable parts — viz. Γ_D, Γ_T, and Γ_C such that $\mathrm{meas}(\Gamma_D) > 0$ and $\mathrm{meas}(\Gamma_C) > 0$. Let $\cdot$ and $|\cdot|$ be, respectively, the canonical inner product and the induced norm. For a vector field $\boldsymbol{v}$ on Γ, we use $v_n = \boldsymbol{v}\cdot\boldsymbol{n}$ for its normal component and $\boldsymbol{v}_\tau = \boldsymbol{v} - v_n\boldsymbol{n}$ for its tangential component, where $\boldsymbol{n}$ is the unit outward normal vector to Γ. The linearized strain tensor associated with a displacement field $\boldsymbol{u} : \Omega \to \mathbb{R}^2$ is denoted by $\boldsymbol{\varepsilon}(\boldsymbol{u})$ and the stress field is denoted by $\boldsymbol{\sigma} : \Omega \to \mathbb{S}^2$, where $\mathbb{S}^2$ is the space of second order symmetric tensors. In addition, we assume a volume force of density $\boldsymbol{f}_0 \in L^2(\Omega, \mathbb{R}^2)$ acting in Ω. Besides, the body is assumed to be fixed on Γ_D, is subject to an action of the surface traction of density $\boldsymbol{f}_2 \in L^2(\Gamma_T, \mathbb{R}^2)$ on Γ_T, and is in contact on Γ_C. The spatial variable $\boldsymbol{x}$ is usually dropped when it causes no confusion.

Consider the space

$$V := \left\{\boldsymbol{v} \in H^1\left(\Omega;\mathbb{R}^2\right) \mid \boldsymbol{v} = \boldsymbol{0} \text{ a.e. on } \Gamma_D\right\},$$

with the inner product

$$(\boldsymbol{u},\boldsymbol{v})_V = \int_\Omega \boldsymbol{\varepsilon}(\boldsymbol{u})\cdot\boldsymbol{\varepsilon}(\boldsymbol{v})\,\mathrm{d}x, \quad \boldsymbol{u},\boldsymbol{v}\in V,$$

and the associated norm $\|\boldsymbol{v}\|_V = \sqrt{(\boldsymbol{v},\boldsymbol{v})_V}$. It is known that V is a Hilbert space [4]. We also consider the Hilbert space $Q = L^2(\Omega;\mathbb{S}^2)$ with the inner product

$$(\boldsymbol{\sigma},\boldsymbol{\tau})_Q = \int_\Omega \sigma_{ij}(\boldsymbol{x})\tau_{ij}(\boldsymbol{x})\,\mathrm{d}x.$$

Here and in what follows we use the Einstein summation convention — i.e. summation implied on repeated indices.

Let us now introduce an HVI for the frictional bilateral contact problem — cf. [11, 16]. Set

$$V_1 := \big\{\boldsymbol{v}\in V \mid \boldsymbol{v}_n = 0 \text{ on } \Gamma_C\big\},$$

and define

$$\langle Au, v\rangle = \big(\mathscr{F}(\boldsymbol{\varepsilon}(\boldsymbol{u})),\boldsymbol{\varepsilon}(\boldsymbol{v})\big)_Q.$$

Besides, we consider mappings $\gamma_{j_\tau} : V_1 \to L^2(\Gamma_C;\mathbb{R}^2)$ such that $\gamma_{j_\tau}(\boldsymbol{v}) = \boldsymbol{v}_\tau$ and

$$\int_\Gamma j^0(\gamma_j u;\gamma_j v)\,\mathrm{d}s = \int_{\Gamma_C} j_\tau^0(\gamma_{j_\tau}\boldsymbol{u};\gamma_{j_\tau}\boldsymbol{v})\,\mathrm{d}s,$$
$$\langle f, v\rangle = \int_\Omega \boldsymbol{f}_0\cdot\boldsymbol{v}\,\mathrm{d}x + \int_{\Gamma_T}\boldsymbol{f}_2\cdot\boldsymbol{v}\,\mathrm{d}s, \quad \boldsymbol{v}\in V.$$

We note that $\mathscr{F} = (F_{ijkl})_{1\le i,j,k,l\le 2} : \Omega\times\mathbb{S}^2 \to \mathbb{S}^2$ is a symmetric bounded linear elasticity operator, which satisfies the following inequality:

$$\mathscr{F}(\boldsymbol{\sigma})\cdot\boldsymbol{\sigma} \ge m_{\mathscr{F}}|\boldsymbol{\sigma}|^2, \quad m_{\mathscr{F}} > 0, \quad \boldsymbol{\sigma}\in\mathbb{S}^2, \tag{2.3}$$

cf. [14]. Moreover, $j_\tau : \Gamma_C\times\mathbb{R}^2\to\mathbb{R}$ is locally Lipschitz on $\mathbb{R}^2$ for a.e. $\boldsymbol{x}\in\Gamma_C$ and satisfies the assumption (A2) with constants c_0, c_1 and α_{j_τ} [11, 16].

The frictional bilateral contact problem can be written in the following HVI form:

$$\big(\mathscr{F}(\boldsymbol{\varepsilon}(\boldsymbol{u})),\boldsymbol{\varepsilon}(\boldsymbol{v})\big)_Q + \int_{\Gamma_C} j_\tau^0(\boldsymbol{u}_\tau;\boldsymbol{v}_\tau)\,\mathrm{d}s \ge \langle\boldsymbol{f},\boldsymbol{v}\rangle, \quad \boldsymbol{v}\in V_1. \tag{2.4}$$

Let $\boldsymbol{u}\in V_1$ and λ_1 be the smallest positive eigenvalue of the eigenvalue problem

$$\int_\Omega \boldsymbol{\varepsilon}(\boldsymbol{u})\cdot\boldsymbol{\varepsilon}(\boldsymbol{v})\,\mathrm{d}x = \lambda_1\int_{\Gamma_C}\boldsymbol{u}_\tau\cdot\boldsymbol{v}_\tau\,\mathrm{d}s, \quad \boldsymbol{v}\in V_1.$$

Then the condition (A4) is satisfied if $c_\Gamma \ge \sqrt{1/\lambda_1}$.

The assumption (A1) is satisfied with $m_A = m_{\mathscr{F}}$. If $\alpha_{j_\tau} < \lambda_1 m_{\mathscr{F}}$, then Theorems 2.1 and 2.2 yield that (2.4) has a unique solution and is equivalent to the optimization problem

$$\boldsymbol{u} = \arg\min_{\boldsymbol{v}\in V_1} E(\boldsymbol{v}), \tag{2.5}$$

where

$$E(\boldsymbol{v}) = \frac{1}{2}\big(\mathscr{F}(\boldsymbol{\varepsilon}(\boldsymbol{v})), \boldsymbol{\varepsilon}(\boldsymbol{v})\big)_Q + \int_{\Gamma_C} j_\tau(\boldsymbol{v}_\tau)\,\mathrm{d}s - \langle \boldsymbol{f}, \boldsymbol{v}\rangle.$$

Next, we introduce an HVI to describe the frictionless normal compliance contact problem — cf. [11, 16]. The frictionless normal compliance contact problem is defined similar to the above considerations, where $\boldsymbol{v}_\tau$ is replaced by $\boldsymbol{v}_n$ and $\gamma_{j_\tau} : V_1 \to L^2(\Gamma_C;\mathbb{R}^2)$ by $\gamma_{j_n} : V \to L^2(\Gamma_C)$, respectively. Therefore, the frictionless normal compliance contact problem reads

$$\big(\mathscr{F}(\boldsymbol{\varepsilon}(\boldsymbol{u})), \boldsymbol{\varepsilon}(\boldsymbol{v})\big)_Q + \int_{\Gamma_C} j_n^0(\boldsymbol{u}_n; \boldsymbol{v}_n)\,\mathrm{d}s \geq \langle \boldsymbol{f}, \boldsymbol{v}\rangle, \quad \boldsymbol{v} \in V. \tag{2.6}$$

Similarly, following Theorems 2.1 and 2.2, the problem (2.6) has a unique solution and is equivalent to the following optimization problem:

$$\boldsymbol{u} = \arg\min_{\boldsymbol{v}\in V} E(\boldsymbol{v}), \tag{2.7}$$

where

$$E(\boldsymbol{v}) = \frac{1}{2}\big(\mathscr{F}(\boldsymbol{\varepsilon}(\boldsymbol{v})), \boldsymbol{\varepsilon}(\boldsymbol{v})\big)_Q + \int_{\Gamma_C} j_n(\boldsymbol{v}_n)\,\mathrm{d}s - \langle \boldsymbol{f}, \boldsymbol{v}\rangle.$$

3. Deep Learning Method for HVIs

3.1. The deep learning method

The main idea of deep learning-based HVIs solvers is to treat DNNs as an efficient parametrization of the solution space of an HVI. The HVI solution is identified via seeking a DNN $\phi(\boldsymbol{x}, \boldsymbol{\theta})$ with input $\boldsymbol{x}$ and a parameter $\boldsymbol{\theta}$ that minimizes the variational minimization problem related to the HVI. From the discussion in Section 2, we know that Problem 2.1 is equivalent to Problem 2.2. This motivates the following problem.

Problem 3.1. *Find $\boldsymbol{\theta}^*$ such that*

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} E\big(\phi(\boldsymbol{x};\boldsymbol{\theta})\big), \tag{3.1}$$

where

$$E\big(\phi(\boldsymbol{x};\boldsymbol{\theta})\big) = \frac{1}{2}\big\langle A\phi(\boldsymbol{x};\boldsymbol{\theta}), \phi(\boldsymbol{x};\boldsymbol{\theta})\big\rangle + \int_\Gamma j\big(\gamma_j \phi(\boldsymbol{x};\boldsymbol{\theta})\big)\,\mathrm{d}s - \big\langle f, \phi(\boldsymbol{x};\boldsymbol{\theta})\big\rangle. \tag{3.2}$$

In contact mechanics, the first and the third terms on the right-hand side of (3.2) usually can be formulated as integrals. Then the objective function in (3.2) can be viewed as a sum of expectations of several random variables, which can be solved by stochastic gradient descent methods (cf. [3]) or its variants. We refer to Subsection 3.2 for details along this line.

In this paper, we use the residual neural network (ResNet) proposed in [19] to approximate the solution of the HVI introduced in Section 2. Mathematically, the ResNet can be formulated as follows:

$$\begin{aligned}
&\boldsymbol{h}_0 = \boldsymbol{V}\boldsymbol{x}, \quad \boldsymbol{g}_\ell = \sigma(\boldsymbol{W}_\ell \boldsymbol{h}_{\ell-1} + \boldsymbol{b}_\ell),\\
&\boldsymbol{h}_\ell = \boldsymbol{h}_{\ell-1} + \boldsymbol{U}_\ell \boldsymbol{g}_\ell, \quad \ell = 1, 2, \ldots, L,\\
&\phi(\boldsymbol{x};\boldsymbol{\theta}) = \boldsymbol{a}^T \boldsymbol{h}_L,
\end{aligned}$$

where $\boldsymbol{V} \in \mathbb{R}^{N_0 \times d}$, $\boldsymbol{W}_\ell \in \mathbb{R}^{N_\ell \times N_0}$, $\boldsymbol{U}_\ell \in \mathbb{R}^{N_0 \times N_\ell}$, $\boldsymbol{b}_\ell \in \mathbb{R}^{N_\ell}$ for $\ell = 1, \ldots, L$, $\boldsymbol{a} \in \mathbb{R}^{N_0 \times m}$. $\sigma(x)$ is a non-linear activation function. For the purpose of simplicity, we consider $N_0 = N_\ell = N$ and $\boldsymbol{U}_\ell$ is set as the identity matrix. Here, L is the depth of the ResNet, and N is the width of the network, and $\boldsymbol{\theta} = \{\boldsymbol{V}, \boldsymbol{a}, \boldsymbol{W}_\ell, \boldsymbol{b}_\ell : 1 \le \ell \le L\}$ denotes the set of all parameters in $\boldsymbol{\phi}$, which uniquely determines the neural network.

We also use locally adaptive activation functions [23] to accelerate the training. In this framework, the neural network can be described as:

$$\begin{aligned}
&\boldsymbol{h}_0 = \boldsymbol{V}\boldsymbol{x}, \quad \boldsymbol{g}_\ell = \sigma\big(n\boldsymbol{a}_\ell * (\boldsymbol{W}_\ell \boldsymbol{h}_{\ell-1} + \boldsymbol{b}_\ell)\big),\\
&\boldsymbol{h}_\ell = \boldsymbol{h}_{\ell-1} + \boldsymbol{g}_\ell, \quad \ell = 1, 2, \ldots, L, \quad \phi(\boldsymbol{x};\boldsymbol{\theta}) = \boldsymbol{a}^T \boldsymbol{h}_L,
\end{aligned}$$

where n and $\boldsymbol{a}_\ell \in \mathbb{R}^N$ for $\ell = 1, \ldots, L$ are the parameters for adaptive activation function, which change the slope of activation functions, and "$*$" stands for the Hadamard product. Notice that the factor n is fixed, so the set of all parameters is $\boldsymbol{\theta} = \{\boldsymbol{V}, \boldsymbol{a}, \boldsymbol{W}_\ell, \boldsymbol{b}_\ell, \boldsymbol{a}_\ell : 1 \le \ell \le L\}$.

In order to update the parameters of the networks, we use the following training algorithm.

Algorithm 3.1 Training Algorithm.

Require: The desired HVI.

Ensure: The parameter set $\boldsymbol{\theta}$ in the solution ResNet $\phi(\boldsymbol{x};\boldsymbol{\theta})$.

Set $Epoch$ as the total iteration number, and assign N, N_Γ as the sample sizes in the domain Ω, the boundary Γ, respectively, the accelerate factor n if we need.

Initialize $\phi(\boldsymbol{x};\boldsymbol{\theta})$ following the default random initialization of PyTorch.

for $k = 1, \ldots, Epoch$ **do**

Generate uniformly distributed samples $\{\boldsymbol{x}_i\}_{i=1}^{N} \subset \Omega$, $\{\boldsymbol{x}_i\}_{i=1}^{N_\Gamma} \subset \Gamma$.

Evaluate the loss function (3.2) at the generated samples.

Use suitable optimizer to update the parameters $\boldsymbol{\theta}$.

end for

3.2. The algorithm description of deep learning methods for contact problems

Considerations in Section 2 show that the frictional bilateral contact problem (2.4) and the frictionless normal compliance contact problem (2.6) are equivalent to optimization problems (2.5) and (2.7), respectively. In order to numerically solve these problems by deep learning method, we first have to parameterize the solution space of HVIs according to the approach of Section 3.1. Note that the frictional bilateral contact problem (2.5) has the solution space V_1 and the frictionless normal compliance contact problem (2.7) has the solution space V. In order to handle the constraints in the admissible spaces V_1 or V, one can employ a penalty method which penalizes the loss function with extra terms to enforce these constraints. However, the tuning of the penalty method parameters can be very inefficient in practice. Another strategy is to construct neural network functions satisfying essential boundary conditions — e.g.

$$\boldsymbol{\phi}(\boldsymbol{x};\boldsymbol{\theta}) = \boldsymbol{b}(\boldsymbol{x}) * \boldsymbol{\psi}(\boldsymbol{x};\boldsymbol{\theta}),$$

where $*$ refers to the Hadamard product, $\boldsymbol{b}(\boldsymbol{x})$ is a known smooth vector-valued function such that $\boldsymbol{\phi} = \mathbf{0}$ on the Dirichlet boundary Γ_D and $\phi_n = 0$ on the contact boundary Γ_C for the frictional bilateral contact problem, or $\boldsymbol{\phi} = \mathbf{0}$ on the Dirichlet boundary Γ_D for the frictionless normal compliance contact problem; $\boldsymbol{\psi}(\boldsymbol{x};\boldsymbol{\theta})$ is an arbitrary DNN in $\{\boldsymbol{\psi}(\boldsymbol{x};\boldsymbol{\theta})\}_{\boldsymbol{\theta}} \approx H^1(\Omega;\mathbb{R}^2)$. Such a method is not always feasible for general problems in complex (non-regular) domains, and here we only consider contact problems in a cube. In this case, we can determine the function $\boldsymbol{b}(\boldsymbol{x})$. For more details about deep learning methods for variational problems with essential boundary conditions the reader can consult [22, 31] and references therein.

Constructing neural network functions which exactly satisfy the essential boundary conditions, we write (3.1) as follows:

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} E\big(\boldsymbol{b}(\boldsymbol{x}) * \boldsymbol{\psi}(\boldsymbol{x};\boldsymbol{\theta})\big), \tag{3.3}$$

and $\boldsymbol{u}^{DL}(\boldsymbol{x};\boldsymbol{\theta}^*) = \boldsymbol{b}(\boldsymbol{x}) * \boldsymbol{\psi}(\boldsymbol{x};\boldsymbol{\theta}^*)$ is the approximate solution to the targeted HVI.

Next, we consider specific functions $E(\boldsymbol{b} * \boldsymbol{\psi})$ in (3.3) for the frictional bilateral contact problem (2.5) and the frictionless normal compliance contact problem (2.7). In particular, for the frictional bilateral contact problem (2.5) we set

$$\begin{aligned} E\big(\boldsymbol{b}(\boldsymbol{x}) * \boldsymbol{\psi}(\boldsymbol{x};\boldsymbol{\theta})\big) &= \frac{1}{2}\Big(\mathscr{F}\Big(\boldsymbol{\varepsilon}\big(\boldsymbol{b}(\boldsymbol{x}) * \boldsymbol{\psi}(\boldsymbol{x};\boldsymbol{\theta})\big)\Big), \boldsymbol{\varepsilon}\big(\boldsymbol{b}(\boldsymbol{x}) * \boldsymbol{\psi}(\boldsymbol{x};\boldsymbol{\theta})\big)\Big)_Q \\ &\quad + \int_{\Gamma_C} j_\tau\big((\boldsymbol{b} * \boldsymbol{\psi})_\tau(\boldsymbol{x};\boldsymbol{\theta})\big)\,\mathrm{d}s - \big\langle \boldsymbol{f}, \boldsymbol{b}(\boldsymbol{x}) * \boldsymbol{\psi}(\boldsymbol{x};\boldsymbol{\theta})\big\rangle \\ &= |\Omega|\mathbb{E}_{\xi_1}\Big[\frac{1}{2}\mathscr{F}\Big(\boldsymbol{\varepsilon}\big(\boldsymbol{b}(\xi_1) * \boldsymbol{\psi}(\xi_1;\boldsymbol{\theta})\big)\Big)_{ij}\boldsymbol{\varepsilon}\big(\boldsymbol{b}(\xi_1) * \boldsymbol{\psi}(\xi_1;\boldsymbol{\theta})\big)_{ij} \\ &\qquad - \boldsymbol{f}_0(\xi_1)\cdot\big(\boldsymbol{b}(\xi_1) * \boldsymbol{\psi}(\xi_1;\boldsymbol{\theta})\big)\Big] \end{aligned}$$

$$-|\Gamma_T|\mathbb{E}_{\xi_2}\Big[\boldsymbol{f}_2(\xi_2)\cdot\big(\boldsymbol{b}(\xi_2)*\boldsymbol{\psi}(\xi_2;\boldsymbol{\theta})\big)\Big]+|\Gamma_C|\mathbb{E}_{\xi_3}\Big[j_\tau\big((\boldsymbol{b}*\boldsymbol{\psi})_\tau(\xi_3;\boldsymbol{\theta})\big)\Big] \tag{3.4}$$

with random vectors ξ_1, ξ_2, and ξ_3 of uniform distribution over Ω, Γ_T, and Γ_C, respectively. In practice, the minimization problem (3.3) is usually solved by the Adam method by randomly sampling the loss function [27]. At each iteration of the optimization algorithm, the stochastic loss function

$$\begin{aligned}&\hat{E}\big(\boldsymbol{b}(\boldsymbol{x})*\boldsymbol{\psi}(\boldsymbol{x};\boldsymbol{\theta})\big)\\ &=\frac{|\Omega|}{N}\sum_{l=1}^{N}\Big[\frac{1}{2}\mathscr{F}\Big(\boldsymbol{\varepsilon}\big(\boldsymbol{b}(\xi_l)*\boldsymbol{\psi}(\xi_l;\boldsymbol{\theta})\big)\Big)_{ij}\boldsymbol{\varepsilon}\big(\boldsymbol{b}(\xi_l)*\boldsymbol{\psi}(\xi_l;\boldsymbol{\theta})\big)_{ij}-\boldsymbol{f}_0(\xi_l)\cdot\big(\boldsymbol{b}(\xi_l)*\boldsymbol{\psi}(\xi_l;\boldsymbol{\theta})\big)\Big]\\ &\quad-\frac{|\Gamma_T|}{N_T}\sum_{l=1}^{N_T}\Big[\boldsymbol{f}_2(\boldsymbol{\eta}_l)\cdot\big(\boldsymbol{b}(\boldsymbol{\eta}_l)*\boldsymbol{\psi}(\boldsymbol{\eta}_l;\boldsymbol{\theta})\big)\Big]+\frac{|\Gamma_C|}{N_T}\sum_{l=1}^{N_C}\Big[j_\tau\big((\boldsymbol{b}*\boldsymbol{\psi})_\tau(\boldsymbol{\zeta}_l;\boldsymbol{\theta})\big)\Big]\end{aligned} \tag{3.5}$$

is minimized instead of the original loss function in (3.4). Note that $\{\xi_l\}_{l=1}^{N}$, $\{\boldsymbol{\eta}_l\}_{l=1}^{N_T}$, and $\{\boldsymbol{\zeta}_l\}_{l=1}^{N_C}$ are independent random vectors with uniform distributions over Ω,Γ_T, and Γ_C, respectively.

The form $E(\boldsymbol{b}(\boldsymbol{x})*\boldsymbol{\psi}(\boldsymbol{x};\boldsymbol{\theta}))$ and the stochastic loss function for the frictionless normal compliance contact problem is defined similar to the frictional bilateral contact problem, but $j_\tau((\boldsymbol{b}*\boldsymbol{\psi})_\tau)$ is replaced by $j_n((b*\psi)_n)$.

4. Numerical Experiments

In this section, we demonstrate the performance of the deep learning method for contact problems under consideration. Numerical comparisons of the deep learning method and the virtual element method (VEM) are provided. Besides, we investigate the numerical performance in terms of fixed and adaptive activation function.

In all numerical experiments, we employ ResNet $\boldsymbol{\phi}(\boldsymbol{x};\boldsymbol{\theta})$ introduced in Section 3 as the solution ansatz to HVIs. In the network, the depth of $\boldsymbol{\phi}$ is set to $L=8$ and the width to $N=50$. Therefore, the total number of parameters in $\boldsymbol{\phi}(\boldsymbol{x};\boldsymbol{\theta})$ with fixed activation function is 18050, and the total number of parameters in $\boldsymbol{\phi}(\boldsymbol{x};\boldsymbol{\theta})$ with local adaptive activation function is 18450. Note that one often uses tanh as the activation function. All numerical experiments are implemented in Python 3.7 using Pytorch 1.3 in an Nvidia Geforce RTX 2080 Ti GPU card. The code can be shared upon request.

Before reporting the numerical results, let us summarize notations used in this section. We write $\boldsymbol{u}^{DL}=(u_1^{DL},u_2^{DL})^T$ for the approximate solution estimated by the deep learning method. Suppose $\mathscr{T}_h$ is a uniform triangulation of Ω into K and $h=\text{diam}(K)$. Since the true solution is unavailable, for the bilateral contact problem, we use $\boldsymbol{u}^{ref}$ as the reference solution evaluated by VEM with $h=2^{-7}$ [11]. Analogously, for the frictionless contact with normal compliance problem, we take $\boldsymbol{u}^{ref}$ as the reference solution evaluated by VEM with $h=2^{-9}$ [11]. Denote the relative difference between the deep learning solution and the

reference solution by

$$\mathscr{E}_r = \frac{\left\|\boldsymbol{u}^{DL} - \boldsymbol{u}^{ref}\right\|_E}{\|\boldsymbol{u}^{ref}\|_E},$$

where $\|\cdot\|_E$ is the energy norm

$$\|\boldsymbol{v}\|_E := \frac{1}{\sqrt{2}}\big(\mathscr{F}(\boldsymbol{\varepsilon}(\boldsymbol{v})), \boldsymbol{\varepsilon}(\boldsymbol{v})\big)_Q^{\frac{1}{2}}.$$

4.1. Bilateral contact problem

Consider the bilateral contact problem (2.4). The domain $\Omega = (0,4)\times(0,4)$ is the cross section of a three-dimensional linearly elastic body and the plane stress condition is imposed. The body is clamped on $\Gamma_D = \{4\}\times(0,4)$ and the vertical traction acts on $\Gamma_T = (\{0\}\times(0,4))\cup((0,4)\times\{4\})$. The frictional contact happens on the boundary $\Gamma_C = (0,4)\times\{0\}$, and the linear elasticity tensor $\mathscr{F}$ has the form

$$(\mathscr{F}\varepsilon)_{ij} = \frac{E\kappa}{(1-\kappa^2)}(\varepsilon_{11}+\varepsilon_{22})\delta_{ij} + \frac{E}{1+\kappa}\varepsilon_{ij}, \quad 1\le i,j\le 2,$$

where E is the Young modulus, κ the Poisson ratio of the material, and δ_{ij} the Kronecker symbol. In numerical simulations, we use the following data:

$$\begin{aligned}
&E = 2000 \text{ daN/mm}^2, \quad \kappa = 0.4,\\
&\boldsymbol{f}_0(x,y) = (0,0)^T \text{ daN/mm}^2 \quad \text{in } \Omega,\\
&\boldsymbol{f}_2(x,y) = \begin{cases} (200(5-y),-200)^T \text{ daN/mm}^2 & \text{on } \{0\}\times(0,4),\\ (0,0)^T \text{ daN/mm}^2 & \text{on } (0,4)\times\{4\},\end{cases}\\
&j_\tau(\boldsymbol{z}) = \int_0^{\|\boldsymbol{z}\|}\left(450e^{-2000t}+450\right)\mathrm{d}t, \quad \boldsymbol{z} = (x,y).
\end{aligned}$$

The HVI(2.4) is approximated by the ResNet $\boldsymbol{\phi}(\boldsymbol{x};\boldsymbol{\theta})$ with the fixed activation function $\tanh(x)$ and locally adaptive activation function (LAA) $\tanh(nax)$. The parameter $n = 3$ is fixed and the adjustable adaptive parameter a is initialized by 1. These neural networks are trained by the Adam optimizer [27] with default initial learning rate $\eta = 0.001$. In optimization, the learning rate is adjusted in an exponentially decaying scheme with decaying rate $0.01^{1/50000}$. The domain has batch size 4096 and on each boundary there are 1024 training data. Then we discuss the numerical performance of different neural network structures.

Fig. 1 shows the change of the loss function during the iterations. We note that in early iterations the fixed activation function has a faster convergence rate than the local adaptive activation one. Fig. 2 (left and middle) displays the numerical solution to the above problem obtained by ResNet with locally adaptive activation after 50.000 epochs. For the contact boundary, the results obtained by deep learning method and VEM are presented in Fig. 2

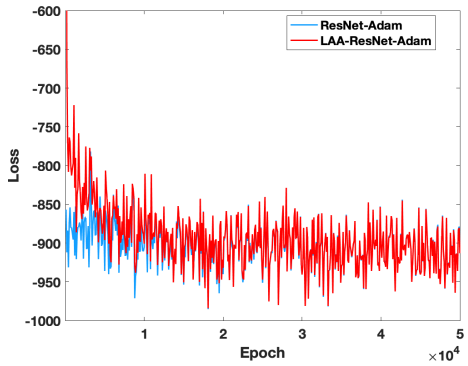

Figure 1: Changes of loss function during iterations.

(right). We note that for fine mesh, the deep learning method has numerical accuracy similar to VEM.

Aiming to quantify the accuracy of deep learning methods and to compare the efficiency of different networks, we evaluate the relative error between numerical and reference solutions. In Table 1, we list the mean value and standard deviation of relative errors for all approaches. It is worth noting that relative errors are reduced to 3% by all deep learning methods. In particular, the use of the local adaptive activation function instead of the fixed activation one improves accuracy by 32%.

Table 1: The relative error of different activation functions for the bilateral contact problem.

Algorithm	ResNet-Adam	LAA-ResNet-Adam
$\mathscr{E}_r$	$0.0295 \pm 0.15\%$	$0.0200 \pm 0.36\%$

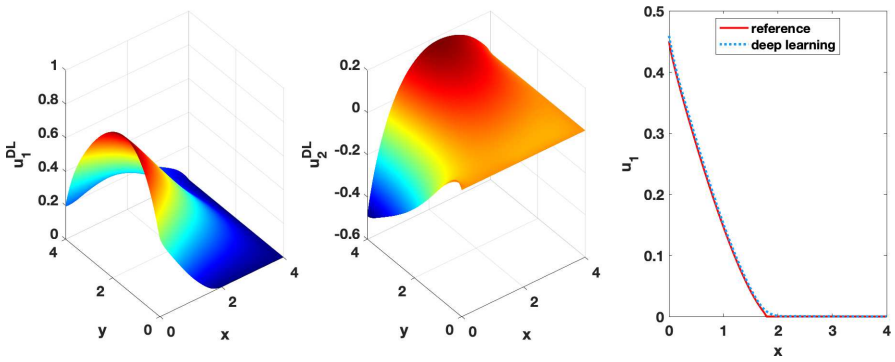

Figure 2: The numerical solution in the domain (left and middle) and on the contact boundary (right).

Remark 4.1. Since for this problem the function $j_\tau(\boldsymbol{z})$ in the contact condition has a good regularity, we also use the L-BFGS method [32] to train the network with fixed activation function. After about 150s (GPU time), we get an approximation with the relative error

$0.0536 \pm 0.89\%$. That means the L-BFGS method is very efficient in solving this problem. However, it exhibits different behaviour in the second example.

4.2. Frictionless normal compliance contact problem

Consider the frictionless normal compliance contact problem (2.6). The domain $\Omega = (0,1)\times(0,1)$ is the cross section of a three-dimensional linearly elastic body and the plane stress condition is imposed. The body is clamped on $\Gamma_D = (\{0\}\times(0,1))\cup(\{1\}\times(0,1))$ and the vertical traction acts on $\Gamma_T = (0,1)\times\{1\}$. The frictional contact happens on the boundary $\Gamma_C = (0,1)\times\{0\}$, and the linear elasticity tensor $\mathscr{F}$ has the form

$$(\mathscr{F}\varepsilon)_{ij} = \frac{E\kappa}{(1+\kappa)(1-2\kappa)}(\varepsilon_{11}+\varepsilon_{22})\delta_{ij} + \frac{E}{1+\kappa}\varepsilon_{ij}, \quad 1 \le i,j \le 2,$$

where E is the Young modulus, κ the Poisson ratio of the material, and δ_{ij} the Kronecker symbol. In numerical simulations, we use the following data:

$$\begin{aligned}
&E = 70 \text{ GPa}, \quad \kappa = 0.3,\\
&\boldsymbol{f}_0(x,y) = (0,0)^T \text{ GPa} \qquad \text{in } \Omega,\\
&\boldsymbol{f}_2(x,y) = (0,-52)^T \text{ GPa} \quad \text{on } \Gamma_T,\\
&j_n(u_n) = \begin{cases} 0, & u_n \le 0,\\ 50u_n^2 + 0.1u_n, & u_n \in (0,0.1],\\ 20.1u_n - 50u_n^2 - 1, & u_n \in (0.1,0.15),\\ 200u_n^2 - 54.9u_n + 4.625, & u_n \ge 0.15. \end{cases}
\end{aligned}$$

Similar to the bilateral contact problem, we solve (2.6) by using different activation functions. The parameters for training algorithms and local adaptive activation functions (LAA) are the same as in Section 4.1. In this problem, the domain has the batch size 1024 and there are 256 training data on each boundary. Fig. 3 shows the change of the loss function

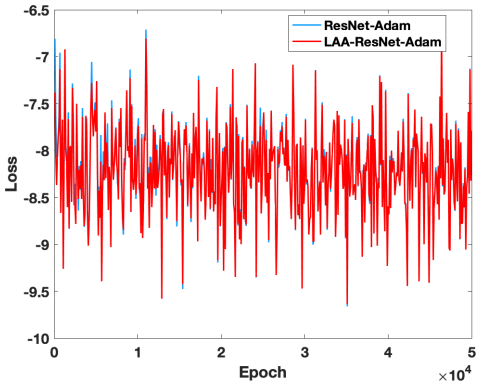

Figure 3: The changes of loss function with iterations.

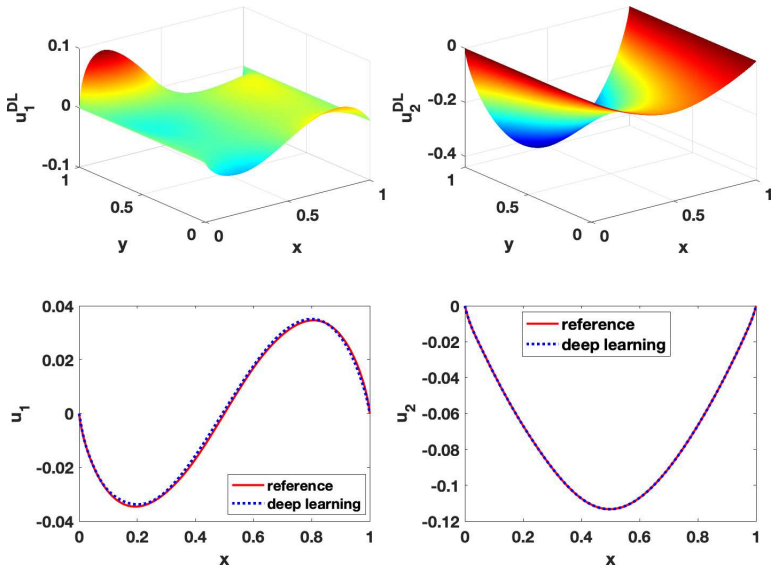

Figure 4: Numerical solution. Top: In the domain. Bottom: On contact boundary.

during iterations. It suggests that the deep learning method related to a fixed activation function or a local adaptive activation function converges.

Fig. 4 (top) displays the numerical solution to the above problem obtained by ResNet with locally adaptive activation after 50,000 epochs. Besides, Fig. 4 (bottom) suggests that for fine mesh on the contact boundary, the deep learning method has numerical accuracy similar to VEM.

Analogously, evaluating the relative error between the deep learning method and the reference solution for different networks — cf. Table 2, we note that the relative errors are reduced to nearly 3% by the Adam method. In particular, the use of the local adaptive activation function instead of the fixed activation one can improve the accuracy by 9%.

Table 2: The relative error of different activation functions for the frictionless normal compliance contact problem.

Algorithm	ResNet-Adam	LAA-ResNet-Adam
$\mathscr{E}_r$	$0.0343 \pm 0.26\%$	$0.0312 \pm 0.75\%$

5. Conclusion

We develop a deep learning method for solving HVIs based on their variational formulations. The solution space is parameterized via DNNs. After that, the HVI is reformulated as an expectation minimization problem, which allows to apply the stochastic gradient descent method or its variants — e.g. the Adam method. As applications to contact mechanics, we consider a frictional bilateral contact problem and a frictionless normal compliance contact problem. Numerical experiments show the following features of the method:

1. For fine meshes, the deep learning method can approximate HVIs with accuracy similar to VEM.

2. The local adaptive activation function increases about 2% parameters, but it improves the accuracy by 10%, at least.

Acknowledgments

The authors would like to thank Dr. Fang Feng for offering comparison data of numerical solutions for contact problems by the virtual element method used in numerical experiments. We also thank the referees and Prof. Victor Didenko for their valuable comments and suggestions which helped to improve the manuscript.

J. H. was partially supported by the National Key Research and Development Project (Grant 2020YFA0709800), NSFC (Grant 12071289) and by the Shanghai Municipal Science and Technology Major Project (Grant 2021SHZDZX0102). C. W. was partially supported by the National Science Foundation Award (Grant DMS-2136380).

References

[1] R.A. Adams, *Sobolev Spaces*, Academic Press (1975).
[2] A.R. Barron, *Universal approximation bounds for superpositions of a sigmoidal function*, IEEE Trans. Inform. Theory **39**, 930–945 (1993).
[3] L. Bottou, F.E. Curtis and J. Nocedal, *Optimization methods for large-scale machine learning*, SIAM Rev. **60**, 223–311 (2018).
[4] S.C. Brenner, *Korn's inequalities for piecewise H^1 vector fields*, Math. Comp. **73**, 1067–1087 (2004).
[5] F.H. Clarke, *Generalized gradients and applications*, Trans. Amer. Math. Soc. **205**, 247–262 (1975).
[6] F.H. Clarke, *Optimization and Nonsmooth Analysis*, John Wiley & Sons, Inc. (1983).
[7] W. E, *Machine learning and computational mathematics*, Commun. Comput. Phys. **28**, 1639–1670 (2020).
[8] W. E, J. Han and A. Jentzen, *Deep learning-based numerical methods for high-dimensional parabolic partial differential equations and backward stochastic differential equations*, Commun. Math. Stat. **5**, 349–380 (2017).
[9] W. E, C. Ma and L. Wu, *A priori estimates of the population risk for two-layer neural networks*, Commun. Math. Sci. **17**, 1407–1425 (2019).
[10] W. E and B. Yu, *The deep Ritz method: a deep learning-based numerical algorithm for solving variational problems*, Commun. Math. Stat. **6**, 1–12 (2018).
[11] F. Feng, W. Han and J. Huang, *Virtual element method for an elliptic hemivariational inequality with applications to contact mechanics*, J. Sci. Comput. **81**, 2388–2412 (2019).
[12] J. Han, A. Jentzen and W. E, *Solving high-dimensional partial differential equations using deep learning*, Proc. Natl. Acad. Sci. USA **115**, 8505–8510 (2018).
[13] J. Han, J. Lu and M. Zhou, *Solving high-dimensional eigenvalue problems using deep neural networks: a diffusion Monte-Carlo like approach*, J. Comput. Phys. **423**, 109792, 13 (2020).
[14] W. Han, *Minimization principles for elliptic hemivariational inequalities*, Nonlinear Anal. Real World Appl. **54**, 103114 (2020).

[15] W. Han and M. Sofonea, *Numerical analysis of hemivariational inequalities in contact mechanics*, Acta Numer. **28**, 175–286 (2019).
[16] W. Han, M. Sofonea and M. Barboteu, *Numerical analysis of elliptic hemivariational inequalities*, SIAM J. Numer. Anal. **55**, 640–663 (2017).
[17] J. Haslinger, M. Miettinen and P.D. Panagiotopoulos, *Finite Element Method for Hemivariational Inequalities*, Kluwer Academic Publishers (1999).
[18] J. He, L. Li, J. Xu and C. Zheng, *ReLU deep neural networks and linear finite elements*, J. Comput. Math. **38**, 502–527 (2020).
[19] K. He, X. Zhang, S. Ren and J. Sun, *Deep residual learning for image recognition*, in: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 770–778 (2016).
[20] K. Hornik, M. Stinchcombe and H. White, *Multilayer feedforward networks are universal approximators*, Neural Networks **2**, 359–366 (1989).
[21] J. Huang, H. Wang and H. Yang, *Int-deep: A deep learning initialized iterative method for nonlinear problems*, J. Comput. Phys. **419**, 109675 (2020).
[22] J. Huang, H. Wang and T. Zhou, *An augmented Lagrangian deep learning method for variational problems with essential boundary conditions*, arXiv e-prints, 2106.14348 (2021).
[23] A.D. Jagtap, K. Kawaguchi and G.E. Karniadakis, *Locally adaptive activation functions with slope recovery for deep and physics-informed neural networks*, Proc. A. **476**, 20200334 (2020).
[24] K. Joki, A.M. Bagirov, N. Karmitsa, M.M. Mäkelä and S. Taheri, *Double bundle method for finding Clarke stationary points in nonsmooth DC programming*, SIAM J. Optim. **28**, 1892–1919 (2018).
[25] G.E. Karniadakis, I.G. Kevrekidis, L. Lu, P. Perdikaris, S. Wang and L. Yang, *Physics-informed machine learning*, Nat. Rev. Phys. **3**, 422–440 (2021).
[26] Y. Khoo, J. Lu and L. Ying, *Solving for high-dimensional committor functions using artificial neural networks*, Res. Math. Sci. **6**, 1–13 (2019).
[27] D.P. Kingma and J. Ba, *Adam: a method for stochastic optimization*, arXiv e-prints, 1412.6980 (2014).
[28] V.A. Kovtunenko, *A hemivariational inequality in crack problems*, Optimization **60**, 1071–1089 (2011).
[29] H. Lee and I.S. Kang, *Neural algorithm for solving differential equations*, J. Comput. Phys. **91**, 110–131 (1990).
[30] W. Li, M.Z. Bazant and J. Zhu, *A physics-guided neural network framework for elastic plates: comparison of governing equations-based and energy-based approaches*, Comput. Methods Appl. Mech. Engrg. **383**, 113933 (2021).
[31] Y. Liao and P. Ming, *Deep Nitsche method: Deep Ritz method with essential boundary conditions*, Commun. Comput. Phys. **29**, 1365–1384 (2021).
[32] D.C. Liu and J. Nocedal, *On the limited memory BFGS method for large scale optimization*, Math. Programming **45**, 503–528 (1989).
[33] L. Lukšan and J. Vlček, *A bundle-Newton method for nonsmooth unconstrained minimization*, Math. Programming **83**, 373–391 (1998).
[34] M.M. Mäkelä, *Survey of bundle methods for nonsmooth optimization*, Optim. Methods Softw. **17**, 1–29 (2002).
[35] E.S. Mistakidis and P.D. Panagiotopoulos, *Numerical treatment of problems involving nonmonotone boundary or stress-strain laws*, Comput. & Structures **64**, 553–565 (1997).
[36] T. Nakamura-Zimmerer, Q. Gong and W. Kang, *Adaptive deep learning for high-dimensional Hamilton-Jacobi-Bellman equations*, SIAM J. Sci. Comput. **43**, A1221–A1247 (2021).
[37] Z. Naniewicz and P.D. Panagiotopoulos, *Mathematical Theory of Hemivariational Inequalities and Applications*, Marcel Dekker, Inc. (1995).

[38] P.D. Panagiotopoulos, *Nonconvex energy functions. Hemivariational inequalities and substationarity principles*, Acta Mech. **48**, 111–130 (1983).
[39] P.D. Panagiotopoulos, *Hemivariational Inequalities. Applications in Mechanics and Engineering*, Springer (1993).
[40] 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).
[41] E. Samaniego, C. Anitescu, S. Goswami, V.M. Nguyen-Thanh, H. Guo, K. Hamdia, X. Zhuang and T. Rabczuk, *An energy approach to the solution of partial differential equations in computational mechanics via machine learning: concepts, implementation and applications*, Comput. Methods Appl. Mech. Engrg. **362**, 112790 (2020).
[42] Z. Shen, H. Yang and S. Zhang, *Deep network approximation characterized by number of neurons*, Commun. Comput. Phys. **28**, 1768–1811 (2020).
[43] Y. Shin, J. Darbon and G.E. Karniadakis, *On the convergence of physics informed neural networks for linear second-order elliptic and parabolic type PDEs*, Commun. Comput. Phys. **28**, 2042–2074 (2020).
[44] J. Sirignano and K. Spiliopoulos, *DGM: a deep learning algorithm for solving partial differential equations*, J. Comput. Phys. **375**, 1339–1364 (2018).
[45] M.A. Tzaferopoulos, E.S. Mistakidis, C.D. Bisbos and P.D. Panagiotopoulos, *Comparison of two methods for the solution of a class of nonconvex energy problems using convex minimization algorithms*, Comput. & Structures **57**, 959–971 (1995).
[46] F. Wang and H. Qi, *A discontinuous Galerkin method for an elliptic hemivariational inequality for semipermeable media*, Appl. Math. Lett. **109**, 106572 (2020).
[47] Y. Zang, G. Bao, X. Ye and H. Zhou, *Weak adversarial networks for high-dimensional partial differential equations*, J. Comput. Phys. **411**, 109409 (2020).