COBEM 2023 27TH INTERNATIONAL CONGRESS OF MECHANICAL ENGENEERING

ABCM Brazilian Society of Mechanical Sciences and Engineering
27th ABCM International Congress of Mechanical Engineering
December 04-08, 2023, Florianópolis-SC, Brazil

COB-2023-0310
PHYSICS-INFORMED NEURAL NETWORKS FOR SOLVING ELASTICITY PROBLEMS

Estevão Fuzaro de Almeida
Samuel da Silva
UNESP - Universidade Estadual Paulista, Departamento de Engenharia Mecânica, Ilha Solteira, SP, Brasil
estevao.fuzaro@unesp.br, samuel.silva13@unesp.br

Americo Cunha Jr
UERJ - Universidade do Estado do Rio de Janeiro, Departamento de Matemática Aplicada, Rio de Janeiro, RJ, Brasil
americo.cunha@uerj.br

Abstract. *Computational mechanics has seen remarkable progress in recent years due to the integration of machine learning techniques, particularly neural networks. Traditional approaches in solid mechanics, such as the finite element method (FEM), often require extensive manual labor in discretization and mesh generation, making them time-consuming and challenging for complex geometries. Moreover, these methods heavily rely on accurate and complete data, which may not always be readily available or prone to measurement errors. On the other hand, Physics-Informed Neural Networks (PINNs) are a machine learning technique that can learn from data and physics equations, allowing accurate and physically consistent predictions. Through this study, we aim to demonstrate the effectiveness of PINNs in accurately predicting the stress distribution in a triangular plate, showcasing their potential as a valuable tool in solving real-world solid mechanics problems. Combining the elasticity conservation laws and boundary conditions into the neural network architecture creates a PINN and is trained on a coarse mesh of points over the plate domain and evaluated on a fine mesh using a data-free approach, compared with the Airy analytical solution.*

Keywords: *solid mechanics, physics-informed neural networks, stress distribution, data-free modeling*

1. INTRODUCTION

The Finite Element Method (FEM) is a widely utilized and sophisticated stress analysis technique that is employed in most solid mechanics problems and was originally created to solve complicated structural mechanics problems (Maggi *et al.*, 2005; Lo, 2014). While FEM is high efficient for direct problems in stress analysis, their main drawback lies in the requirement for a fine element mesh to achieve satisfactory accuracy. The mesh refinement process, aiming for precise results, often necessitates careful consideration of aspect ratios and the placement of smaller elements in areas with significant stress gradients (Xu *et al.*, 2020). In contrast, Physics-Informed Neural Networks (PINNs) offer a complementary paradigm shift by merging the expressive capacity of neural networks with fundamental physics principles, resulting in a unique framework that can overcome some traditional solid mechanics methods' limitations. They are designed to follow the governing laws of physics, making them ideal for problems with limited or noisy data (Haghighat *et al.*, 2020). They also offer a promising technique for addressing numerical instabilities in engineering applications like Uncertainty Quantification (UQ), inverse problems, and optimization, making them ideal for extensive model evaluations.

The primary motivation behind PINNs lies in their ability to learn from both observational and physical data simultaneously. Traditional neural networks lack the explicit incorporation of physical laws, which limits their applicability in physics-driven problems. PINNs address this limitation by explicitly encoding the underlying physical equations and boundary conditions into the neural network architecture. This integration ensures that the learned solutions are not only data-driven but also adhere to the governing laws of the problem, leading to physically consistent predictions (Raissi *et al.*, 2019; Karniadakis *et al.*, 2021; Antonelo *et al.*, 2021; Rodriguez-Torrado *et al.*, 2022).

PINNs have found numerous applications across various scientific and engineering domains. In the context of solid mechanics, they have been successfully applied to problems such as structural analysis, deformation prediction, stress analysis, and fracture mechanics, being employed in diverse scenarios, including the analysis of beams, plates, shells, and more complex structures (Goswami *et al.*, 2020; Vahab *et al.*, 2021; Jin *et al.*, 2023; Roy *et al.*, 2023). PINNs offer great potential in scenarios where traditional methods face challenges, such as problems involving complex geometries, nonlinear material behavior, and multiphysics coupling (Ma *et al.*, 2022).

This article provides an overview of PINNs, emphasizing their theoretical foundations, advantages, and limitations. Specifically, we focus on their application to analyze a triangular plate subjected to a triangular load. We outline the formulation of the PINN methodology, incorporating the elasticity conservation laws and boundary conditions specific to

the problem. The training process, which optimizes the neural network weights and biases to satisfy the physics-based constraints, is described. The graphical abstract of this work is shown in Fig. 1.

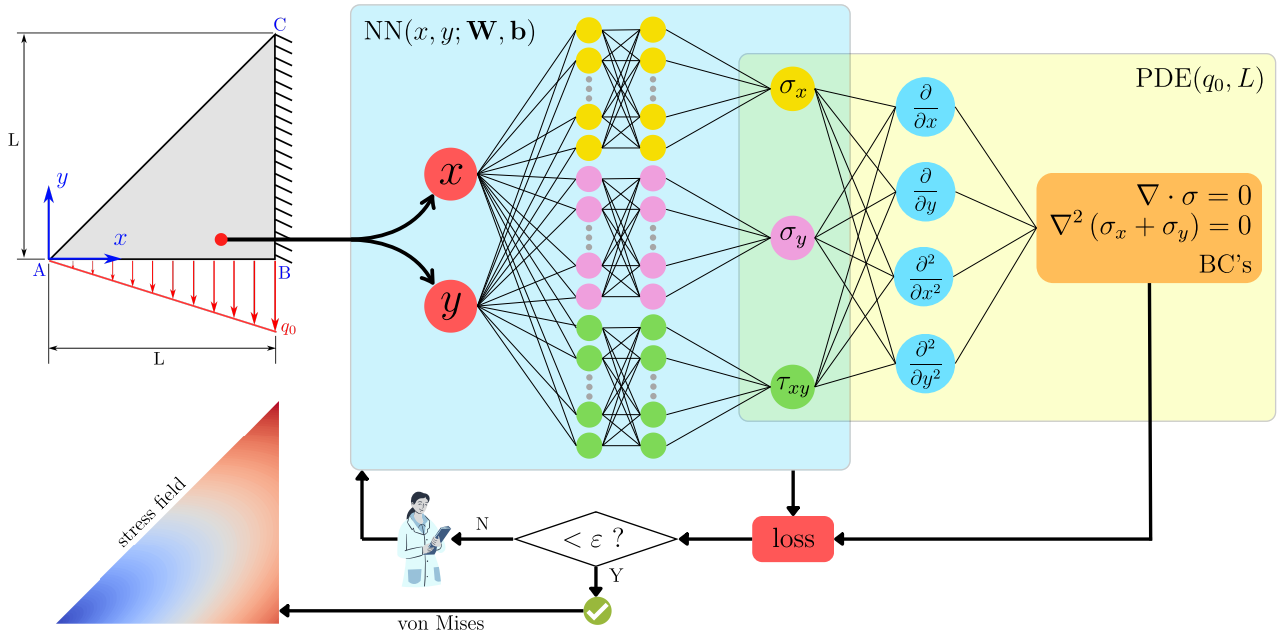

Figure 1. Graphical abstract of the developed work involving PINNs applied to a Solid Mechanics problem.

2. METHODOLOGY

In this session, it is presented the methodology of the work involving the problem to be solved in the field of solid mechanics, as well as the theoretical foundations for both the formulation of PINNs and the generation of a reference solution for the problem using the Airy stress function.

2.1 The problem: triangular plate subjected to a triangular load

The schematic of the two-dimensional triangular plate studied in this work is illustrated in Fig. 2. The plate comprises a right triangle shape with perpendicular sides equal to L $= 1$ m; face BC is fixed, while face AB experiences a triangular loading ranging from zero to $q_0 = 10$ N/m^2; the coordinate system's origin is placed at point A. The objective is to get the stress field over the triangular plate.

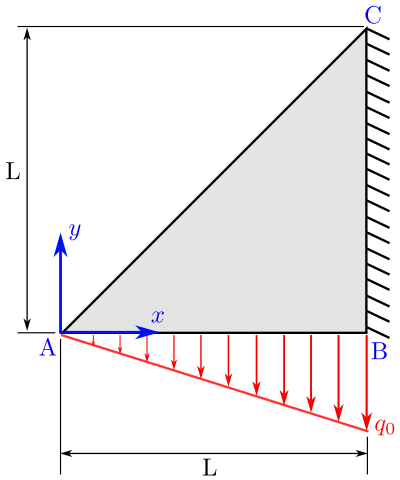

Figure 2. Scheme of the clamped-free triangular plate with a triangular load.

The boundary conditions are obtained by evaluating the stress tensor ($\boldsymbol{\sigma}$) applied to each face of the plate, as follows:

- **Face AB** ($y = 0, \forall x, 0 \leq x \leq L$):

$$\hat{n}_{\text{AB}} = -\hat{j}\ ; \quad \vec{t}_{\text{AB}} = \boldsymbol{\sigma}\hat{n}_{\text{AB}} \ \Rightarrow\ \vec{t}_{\text{AB}} = \begin{bmatrix}\sigma_x & \tau_{xy}\\ \tau_{xy} & \sigma_y\end{bmatrix}\begin{Bmatrix}0\\-1\end{Bmatrix} = \begin{Bmatrix}0\\q(x)\end{Bmatrix},\ \text{where } q(x) = -\frac{q_0}{L}x \tag{1}$$

$$\therefore\ \tau_{xy}(x,0) = 0\ \text{ and }\ \sigma_y(x,0) = \frac{q_0}{L}x. \tag{2}$$

- **Face AC** ($y = x,\ 0 \le x \le L$):

$$\hat{n}_{\text{AC}} = -\frac{1}{\sqrt{2}}\hat{i} + \frac{1}{\sqrt{2}}\hat{j};\quad \vec{t}_{\text{AC}} = \boldsymbol{\sigma}\hat{n}_{\text{AC}} \ \Rightarrow\ \vec{t}_{\text{AC}} = \begin{bmatrix}\sigma_x & \tau_{xy}\\ \tau_{xy} & \sigma_y\end{bmatrix}\begin{Bmatrix}-\frac{1}{\sqrt{2}}\\ \frac{1}{\sqrt{2}}\end{Bmatrix} = \begin{Bmatrix}0\\0\end{Bmatrix} \tag{3}$$

$$\therefore\ \sigma_x(x,x) = \tau_{xy}(x,x)\ \text{ and }\ \sigma_y(x,x) = \tau_{xy}(x,x). \tag{4}$$

- **Face BC** ($x = L,\ \forall y,\ 0 \le y \le L$):

$$\hat{n}_{\text{BC}} = +\hat{i}\ ; \quad \vec{t}_{\text{BC}} = \boldsymbol{\sigma}\hat{n}_{\text{BC}} \ \Rightarrow\ \vec{t}_{\text{BC}} = \begin{bmatrix}\sigma_x & \tau_{xy}\\ \tau_{xy} & \sigma_y\end{bmatrix}\begin{Bmatrix}1\\0\end{Bmatrix} = \begin{Bmatrix}t_x\\t_y\end{Bmatrix} \tag{5}$$

$$\therefore\ \sigma_x(L,y) = t_x\ \text{ and }\ \tau_{xy}(L,y) = t_y. \tag{6}$$

2.2 Elasticity equations for two-dimensional problems

Cauchy's equilibrium equations, strain-displacement relationships, and constitutive equations produce a Boundary Value Problem (BVP) with 15 unknowns and 15 equations. Numerical approximations, including the Finite Element Method (FEM), can be used for complex geometries. Indeed, several practical scenarios of relevance can be approximated as 2D problems. In such circumstances, analytical solutions can be easily achieved without resorting to numerical approaches, which will be discussed later. Assuming a plane stress condition, the equilibrium equations and the compatibility condition for strains yield three equations that explain any two-dimensional elasticity problem (Love, 1927; Timoshenko and Goodier; Boresi and LYNN, 1974; Sadd, 2009; Soutas-Little, 2012):

$$\frac{\partial\sigma_x}{\partial x} + \frac{\partial\tau_{xy}}{\partial y} = 0\ , \tag{7}$$

$$\frac{\partial\tau_{xy}}{\partial x} + \frac{\partial\sigma_y}{\partial y} = 0\ , \tag{8}$$

$$\frac{\partial^2\sigma_x}{\partial x^2} + \frac{\partial^2\sigma_y}{\partial x^2} + \frac{\partial^2\sigma_x}{\partial y^2} + \frac{\partial^2\sigma_y}{\partial y^2} = 0\ , \tag{9}$$

where the (x, y) pair is the Cartesian coordinates; σ_x and σ_y represent the normal stresses in the x and y directions respectively; and τ_{xy} represents the shear stress in the xy-plane. These equations can be simplified into two main equations:

$$\boldsymbol{\nabla}\cdot\boldsymbol{\sigma} = 0\ , \tag{10}$$

$$\nabla^2\left(\sigma_x + \sigma_y\right) = 0\ , \tag{11}$$

where Eq. (10) is the divergence form of the equilibrium equations in which $\boldsymbol{\nabla}$ represents the gradient operator, and $\boldsymbol{\sigma}$ is the stress tensor, and Eq. (11) rewrites the compatibility condition using the Laplacian operator ∇^2. By describing the boundary conditions for this BVP, it is possible to determine $\sigma_x(x,y)$, $\sigma_y(x,y)$, and $\tau_{xy}(x,y)$.

2.3 Airy stress function

The Airy stress function is a special case of the Maxwell stress functions, in which it is assumed that the third element of the main diagonal of Maxwell's tensor ($\Phi_{3,3} = \text{C}$) is a function of x and y only. As a result, this stress function can only be used for two-dimensional issues. The stress function C is commonly represented as $\phi(x,y)$ in the elasticity literature (Soutas-Little, 2012), and the stresses are given by:

$$\sigma_x = \frac{\partial^2\phi}{\partial y^2}\ ,\quad \sigma_y = \frac{\partial^2\phi}{\partial x^2}\ ,\quad \tau_{xy} = -\frac{\partial^2\phi}{\partial x\partial y}, \tag{12}$$

which satisfy the equilibrium equations (Eqs. (7) – (8)) previously described.

Therefore, to ensure the existence of an Airy function, it is sufficient to verify if $\phi(x,y)$ is a sufficiently smooth (at continuously twice differentiable function), satisfying Eq. (9) as follows:

$$\nabla^2\left(\sigma_x + \sigma_y\right) = 0\ \Rightarrow\ \nabla^2\left(\frac{\partial^2\phi}{\partial y^2} + \frac{\partial^2\phi}{\partial x^2}\right) = 0\ \Rightarrow \tag{13}$$

$$\nabla^2(\nabla^2\phi(x,y)) = \frac{\partial^4\phi}{\partial x^4} + 2\frac{\partial^4\phi}{\partial x^2\partial y^2} + \frac{\partial^4\phi}{\partial y^4} = 0\ \Rightarrow\ \nabla^4\phi = 0. \tag{14}$$

with this, $\forall\phi(x,y)$ that ensures $\nabla^4\phi = 0$ can be used to describe $\sigma_x(x,y)$, $\sigma_y(x,y)$, and $\tau_{xy}(x,y)$.

2.4 Analytical stress field

To obtain an analytical solution to compare with PINN methodology, we propose an Airy function for the triangular plate problem as follows:

$$\phi(x,y) = \mathcal{A}x^3 + Bx^2y + Cxy^2 + Dy^3. \tag{15}$$

First it were obtained σ_x, σ_y and τ_{xy} based on Eq. (12):

$$\sigma_x = 2\mathcal{C}x + 6\mathcal{D}y\ , \quad \sigma_y = 6\mathcal{A}x + 2\mathcal{B}y\ , \quad \tau_{xy} = -2\mathcal{B}x - 2\mathcal{C}y\ , \tag{16}$$

where the coefficients $\mathcal{A}$, $\mathcal{B}$, $\mathcal{C}$, and $\mathcal{D}$ can be obtained by applying the boundary conditions (Eqs. (2), (4) and (6)) corresponding to each face of the plate. Solving the system with 6 equations (BCs) and 6 variables ($\mathcal{A}, \mathcal{B}, \mathcal{C}, \mathcal{D}, t_x, t_y$) can be obtained the analytical stress field of the triangular plate to be used as a comparison with PINN's solution:

$$\sigma_x(x,y) = -\frac{q_0}{L}\left(x - 2y\right)\ , \tag{17}$$

$$\sigma_y(x,y) = \frac{q_0}{L}x\ , \tag{18}$$

$$\tau_{xy}(x,y) = \frac{q_0}{L}y\ , \tag{19}$$

$$t_x(y) = q_0\left(\frac{2}{L}y - 1\right)\ , \tag{20}$$

$$t_y(y) = q_0. \tag{21}$$

2.5 von Mises yield criterion

The von Mises yield criterion in materials science and engineering can be expressed in terms of the von Mises equivalent stress σ_v, a scalar stress value calculated from the Cauchy stress tensor. Based on the results of simple tension tests, the von Mises equivalent stress is used to estimate the yielding of the material under any loading scenario. In the case of plane stress, as present in this study, the von Mises criterion states that:

$$\sigma_v = \sqrt{\sigma_1^2 - \sigma_1\sigma_2 + \sigma_2^2}\ , \tag{22}$$

where σ_1 and σ_2 are the principal stresses obtained from the eigenvalues of the stress tensor $\boldsymbol{\sigma}$:

$$\boldsymbol{\sigma} = \begin{bmatrix} \sigma_x & \tau_{xy} \\ \tau_{xy} & \sigma_y \end{bmatrix} \tag{23}$$

2.6 Physics-Informed Neural Networks (PINNs)

PINNs combine the strength of neural networks with the incorporation of fundamental physics concepts to handle complicated issues in a data-efficient manner. PINNs seek to learn the underlying physics equations directly from data, allowing for accurate predictions while adhering to the problem's governing principles. PINNs are created by embedding the physics equations and boundary conditions into the neural network architecture (Karniadakis *et al.*, 2021).

Consider a general partial differential equation (PDE) representing the physics of the problem:

$$\mathcal{F}(u, \nabla u) = 0, \tag{24}$$

where u represents the unknown field, and ∇ denotes the gradient operator. The goal of PINNs is to find an approximation $u(\boldsymbol{x};\boldsymbol{\theta})$ that satisfies both the given PDE and the associated boundary conditions.

The neural network $u(\boldsymbol{x};\boldsymbol{\theta})$ takes the spatial coordinate $\boldsymbol{x}$ as input and is parameterized by $\boldsymbol{\theta}$, which represents the weights and biases of the network. The network output is defined as $u(\boldsymbol{x};\boldsymbol{\theta}) \approx \sigma(\boldsymbol{x};\boldsymbol{\theta})$, where σ represents the activation function. PINNs employ automatic differentiation to calculate the derivatives of $u(\boldsymbol{x};\boldsymbol{\theta})$ concerning $\boldsymbol{x}$ to enforce the physics constraints and use them to construct the residual function:

$$\mathcal{R}(\boldsymbol{x};\boldsymbol{\theta}) = \mathcal{F}(u(\boldsymbol{x};\boldsymbol{\theta}), \nabla u(\boldsymbol{x};\boldsymbol{\theta})). \tag{25}$$

The residual function $\mathcal{R}(\boldsymbol{x};\boldsymbol{\theta})$ quantifies the deviation from the governing physics equations. By minimizing the mean squared residual over a set of training points $\mathcal{D}$, the neural network parameters $\boldsymbol{\theta}$ can be optimized:

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} \frac{1}{N}\sum_{\boldsymbol{x}\in\mathcal{D}} \left(\mathcal{R}(\boldsymbol{x};\boldsymbol{\theta})\right)^2, \tag{26}$$

where N represents the number of training points.

Additional terms are added to the loss function to incorporate the boundary conditions. For example, for a Dirichlet boundary condition $u(\boldsymbol{x}_b;\boldsymbol{\theta}) = g(\boldsymbol{x}_b)$, where $\boldsymbol{x}_b$ represents a point on the boundary and $g(\boldsymbol{x}_b)$ is a given boundary function, the corresponding term in the loss function is:

$$\mathcal{L}_b = \frac{1}{M}\sum_{\boldsymbol{x}_b\in\mathcal{B}}\left[u(\boldsymbol{x}_b;\boldsymbol{\theta}) - g(\boldsymbol{x}_b)\right]^2, \tag{27}$$

where $\mathcal{B}$ represents the set of boundary points and M is the number of boundary points.

By minimizing the combined loss function, which includes the mean squared residual and the boundary terms, PINNs can learn the underlying physics and provide accurate solutions. The ability of PINNs to handle complicated geometries, nonlinear behaviors, and multiphysics interaction without the requirement for explicit mesh generation is one of their main features. Since they are designed to learn from both observational and physical data simultaneously, PINNs can efficiently manage situations with limited or noisy data (Brunton and Kutz, 2022). It is crucial to point out, however, that PINNs face multiple restrictions. To achieve accurate and consistent convergence, the training procedure can be computationally expensive, especially for large-scale issues, and careful consideration must be given to the choice of network architecture, activation, and loss functions.

2.6.1 Building a PINN to solve the problem

The process utilized to build the Physics-Informed Neural Network (PINN) used in this study is outlined below. SciANN, a high-level artificial neural network API written in Python with Keras and TensorFlow backends, was used (Haghighat and Juanes, 2021). It allows rapid experimentation with various network designs, emphasizing scientific computations, physics-informed deep learning, and inversion. With this API one can start deep-learning in a very few lines of code.

The initial step is to build the stress approximation space described by $\hat{\sigma}_x$, $\hat{\sigma}_y$ and $\hat{\tau}_{xy}$. The Cartesian pair (x, y) are the independent variables and $\mathcal{N}_{\sigma_x}$, $\mathcal{N}_{\sigma_y}$, and $\mathcal{N}_{\tau_{xy}}$ are the neural networks for each of the stresses:

$$\hat{\sigma}_x : (x, y) \rightarrow \mathcal{N}_{\sigma_x}(x, y; \mathbf{W}, \mathbf{b}) \tag{28}$$
$$\hat{\sigma}_y : (x, y) \rightarrow \mathcal{N}_{\sigma_y}(x, y; \mathbf{W}, \mathbf{b}) \tag{29}$$
$$\hat{\tau}_{xy} : (x, y) \rightarrow \mathcal{N}_{\tau_{xy}}(x, y; \mathbf{W}, \mathbf{b}) \tag{30}$$

A deep neural network with four layers of twenty neurons each was employed for each stress. The networks receive as input the Cartesian pair (x, y) created by a mesh of points over the problem's domain (plate surface) and output the respective stress for each network. A coarse mesh that comprises 1275 points on the plate (50×50 points excluding points above the diagonal) was chosen for PINN's training. On the other hand, a fine mesh was used to evaluate the analytical solution over 125250 points on the plate (500×500 points excluding points above the diagonal). The objective is to extrapolate the network training and perform a comparison with the analytical solution. The produced meshes for each approach are shown in Fig. 3.

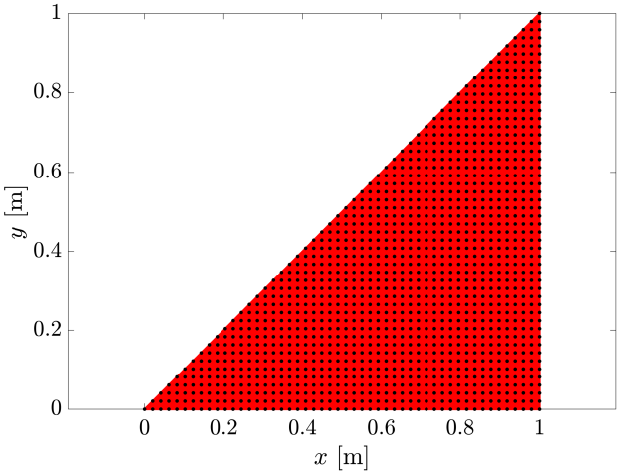

Figure 3. Representation of the generated mesh for evaluating the stress field of the plate. The analytical solution is obtained over the red points (•) of the mesh, while the black points (•) represent the mesh evaluated by the PINN.

It should be noted that no analytical data was used to train the PINN: the physics provided to the networks is based on the previously stated constraints, which include three conservation equations (Eqs. (7) – (9)) and four boundary conditions

(Eqs. (2) and (4)) only. Equation (6) was not used because its sole purpose is to estimate specific stresses at the fixed boundary, which are inevitably dependent on the network solution and may delay the training process.

The PINN was trained for 1000 epochs, with an initial learning rate of 0.01 and a batch size of 50 (equal to the number of points for each plate dimension). The loss function used was the mean squared error (MSE), the activation function was hyperbolic tangent (tanh), and the optimizer was Adam, a stochastic gradient descent method that relies on the adaptive estimation of first and second-order moments. Once again, it is worth noting that no analytical data was used to train the PINN, only conservation laws and boundary conditions.

3. RESULTS AND DISCUSSIONS

Figure 4 depicts the MSE during the training process and the variation of the learning rate. Figure 5 presents the same MSE for each of the seven objective functions, with the conservation equations represented by the letter "L" and the boundary conditions represented by the letter "C". The methodology entails minimizing the loss function over the issue domain throughout a set number of steps (epochs), as described in Eq. (27). The PINN training takes roughly 40 minutes[1].

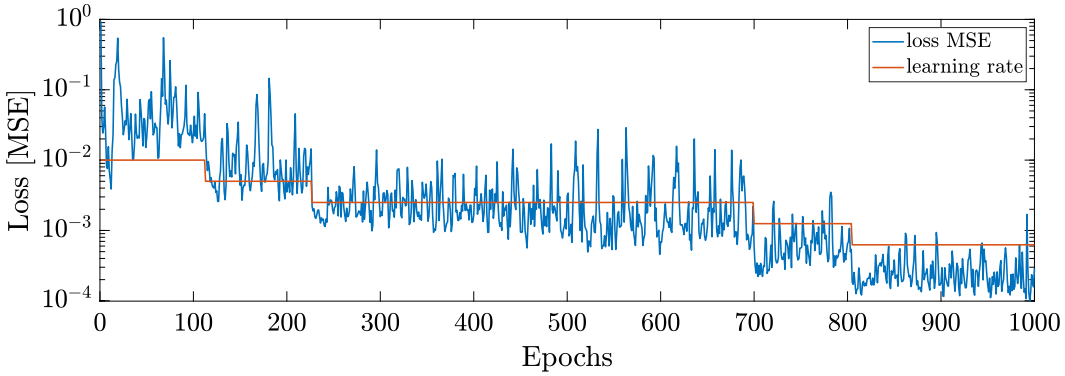

Figure 4. Representation of the MSE loss (•) and the learning rate (•) of the PINN throughout the training process as a function of the number of epochs.

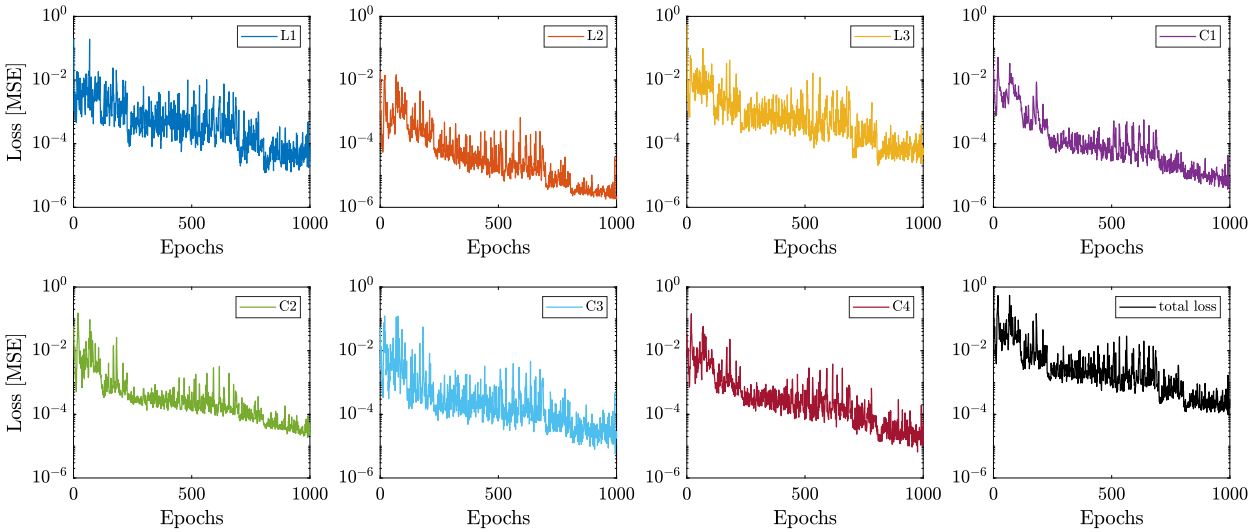

Figure 5. MSE loss for each physical condition presented to the network. Conditions labeled with the letter "L" correspond to the conservation equations, while those labeled with the letter "C" correspond to the boundary conditions.

The PINN was extrapolated after training to evaluate the fine mesh. Then, the corresponding stress tensors for each approach were computed over the 125250 points. The principal stresses were then obtained, and the von Mises stress for the plate was determined. Figure 6 presents the von Mises stress field for each of the methodologies: (a) shows the analytical solution, (b) displays the PINN solution, and (c) illustrates the absolute error between the methodologies. The maximum error occurs near the plate diagonal and has a magnitude of 0.2 MPa, equivalent to 1%.

[1]DELL G15: i5-12500H, 16GB RAM DDR5, RTX 3050 4GB DDR6

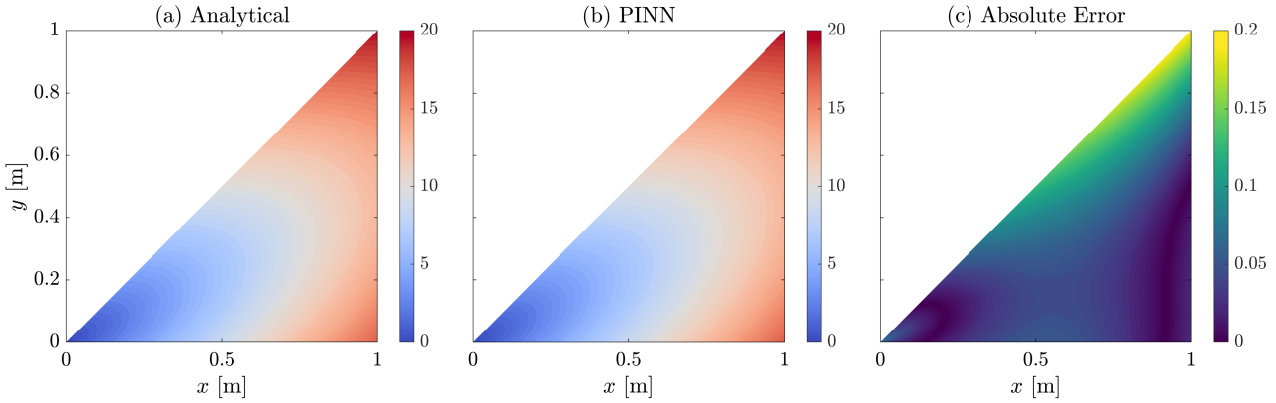

Figure 6. Stress field in MPa via von Mises yield criterion for the cases: (a) Analytical solution via Airy; (b) PINN, trained solely using conservation equations and boundary conditions; (c) Absolute error between the methodologies.

4. FINAL REMARKS

In this work, a Physics-Informed Neural Network (PINN) technique is used to accurately forecast the stress distribution in a triangular plate subjected to a triangle load. It was proved that PINNs can solve a solid mechanics problem without analytical data, relying on conservation equations and boundary conditions. When the PINN solution was compared to the analytical solution generated from the Airy stress function, there was good agreement with a maximum error of about 1% using the von Mises stress field. The results show that PINNs have the potential to be a valuable tool for solving complex problems in computational mechanics, particularly in cases where traditional methods fail. The fact that the PINN was trained on a coarse mesh and evaluated on a fine mesh is highlighted. This is a significant difference from FEM which requires computation and evaluation on the same pre-established mesh, which is a significant drawback. Future research will examine how PINNs can solve various solid mechanics problems involving nonlinear material behavior, multiphysics interaction, and complex geometries.

5. ACKNOWLEDGEMENTS

The first author would like to thank São Paulo Research Foundation (FAPESP) for providing financial support under grant number 2022/16156-9.

6. REFERENCES

Antonelo, E.A., Camponogara, E., Seman, L.O., de Souza, E.R., Jordanou, J.P. and Hubner, J.F., 2021. "Physics-informed neural nets for control of dynamical systems". doi:10.48550/arXiv.2104.02556.

Boresi, A. and LYNN, P., 1974. "Elasticity in". *Engineering Mechanics. New JerseY, Prentice-Hall*, pp. 2–5.

Brunton, S.L. and Kutz, J.N., 2022. *Data-driven science and engineering: Machine learning, dynamical systems, and control*. Cambridge University Press. ISBN 1-009-09848-9.

Goswami, S., Anitescu, C., Chakraborty, S. and Rabczuk, T., 2020. "Transfer learning enhanced physics informed neural network for phase-field modeling of fracture". *Theoretical and Applied Fracture Mechanics*, Vol. 106, p. 102447.

Haghighat, E. and Juanes, R., 2021. "Sciann: A keras/tensorflow wrapper for scientific computations and physics-informed deep learning using artificial neural networks". *Computer Methods in Applied Mechanics and Engineering*, Vol. 373, p. 113552.

Haghighat, E., Raissi, M., Moure, A., Gomez, H. and Juanes, R., 2020. "A deep learning framework for solution and discovery in solid mechanics". doi:10.48550/arXiv.2003.02751.

Jin, H., Zhang, E. and Espinosa, H.D., 2023. "Recent advances and applications of machine learning in experimental solid mechanics: A review". doi:10.48550/arXiv.2303.07647.

Karniadakis, G.E., Kevrekidis, I.G., Lu, L., Perdikaris, P., Wang, S. and Yang, L., 2021. "Physics-informed machine learning". *Nature Reviews Physics*, Vol. 3, No. 6, pp. 422–440. doi:10.1038/s42254-021-00314-5. URL `https://doi.org/10.1038/s42254-021-00314-5`.

Lo, D.S., 2014. *Finite element mesh generation*. CRC press. ISBN 0-415-69048-X.

Love, A.E.H., 1927. *A treatise on the mathematical theory of elasticity*. University press.

Ma, Y., Xu, X., Yan, S. and Ren, Z., 2022. "A preliminary study on the resolution of electro-thermal multi-physics coupling problem using physics-informed neural network (PINN)". *Algorithms*, Vol. 15, No. 2, p. 53. doi:10.3390/a15020053. URL `https://doi.org/10.3390/a15020053`.

Maggi, Y.I., Gonçalves, R.M., Leon, R.T. and Ribeiro, L.F.L., 2005. "Parametric analysis of steel bolted end plate connections using finite element modeling". *Journal of Constructional Steel Research*, Vol. 61, No. 5, pp. 689–708.

Raissi, M., Perdikaris, P. and Karniadakis, G.E., 2019. "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations". *Journal of Computational Physics*, Vol. 378, pp. 686–707. ISSN 0021-9991. doi:10.1016/j.jcp.2018.10.045. URL `https://www.sciencedirect.com/science/article/pii/S0021999118307125`.

Rodriguez-Torrado, R., Ruiz, P., Cueto-Felgueroso, L., Green, M.C., Friesen, T., Matringe, S. and Togelius, J., 2022. "Physics-informed attention-based neural network for hyperbolic partial differential equations: application to the buckley–leverett problem". *Scientific Reports*, Vol. 12, No. 1. doi:10.1038/s41598-022-11058-2. URL `https://doi.org/10.1038/s41598-022-11058-2`.

Roy, A.M., Bose, R., Sundararaghavan, V. and Arróyave, R., 2023. "Deep learning-accelerated computational framework based on physics informed neural network for the solution of linear elasticity". *Neural Networks*, Vol. 162, pp. 472–489. doi:10.1016/j.neunet.2023.03.014. URL `https://doi.org/10.1016/j.neunet.2023.03.014`.

Sadd, M.H., 2009. *Elasticity: theory, applications, and numerics*. Academic Press.

Soutas-Little, R.W., 2012. *Elasticity*. Courier Corporation. ISBN 0-486-40690-3.

Timoshenko, S. and Goodier, J., ???? "1970, theory of elasticity, mcgraw-hill, new york".

Vahab, M., Haghighat, E., Khaleghi, M. and Khalili, N., 2021. "A physics informed neural network approach to solution and identification of biharmonic equations of elasticity". doi:10.48550/ARXIV.2108.07243. URL `https://arxiv.org/abs/2108.07243`.

Xu, R., Yang, J., Yan, W., Huang, Q., Giunta, G., Belouettar, S., Zahrouni, H., Zineb, T.B. and Hu, H., 2020. "Data-driven multiscale finite element method: From concurrence to separation". *Computer Methods in Applied Mechanics and Engineering*, Vol. 363, p. 112893.

7. RESPONSIBILITY NOTICE

The authors are the only ones responsible for the printed material included in this paper.