XLVI CILAMCE
VITÓRIA - ES - BRAZIL - 2025
XLVI IBERO-LATIN AMERICAN CONGRESS ON COMPUTATIONAL METHODS IN ENGENEERING

A physically compliant surrogate constitutive model for thin-walled rods using DNN: design, training and use in FEM simulations

Marcos P. Kassab[1], Eduardo M. B. Campello[1], Adnan Ibrahimbegovic[2]

[1]*Department of Structural and Geotechnical Engineering, Polytechnic School, University of São Paulo*
P.O. Box 61548, 05424-970, São Paulo, Brazil
marcos.kassab@usp.br, campello@usp.br
[2]*University of Technology Compiègne - Alliance Sorbonne University, Laboratoire Roberval, Centre de recherche Royallieu*
Rue de docteur Schweitzer, 60200, Compiègne, France
adnan.ibrahimbegovic@utc.fr

Abstract. We propose a deep neural network (DNN)-based surrogate constitutive model tailored for thin-walled rod members. Unlike traditional multilayer perceptrons, our architecture is physically informed by design, embedding key mechanical behaviors directly into its structure. By formulating the internal energy function in terms of conventional rod strain measures, the model provides a compact yet robust representation of complex structural responses in a low-order (rod) context. The data generation workflow, training strategy and modelling choices are outlined, highlighting the essential fine-tuning required for accurate performance. Once trained, the surrogate model is integrated into our in-house finite element software, PEFSYS, which specializes in large displacement and finite strain analyses of rods and shells. The model accurately captures key nonlinear effects, including stiffness degradation due to local flange and web buckling, offering an efficient and physically consistent framework for simulating thin-walled structures with low-order rod theories.

Keywords: thin-walled rods, DNNs, surrogate model, local buckling, finite element methods (FEM)

1 Introduction

The analysis of rod members is a central aspect in structural design. Even though Reduced Order Models (ROMs) terminology is recent, classical rod models can be interpreted as an early form of model order reduction. Given an elongated geometry and appropriate loading conditions, the three-dimensional (3D) solid kinematics is constrained, effectively reducing the amount of degrees of freedom (DOFs) required to capture the studied mechanical behavior with sufficient accuracy.

Even though it is nowadays trivial to find global buckling modes for those members (Euler-Column, torsional buckling under compression, lateral torsion under bending, see Campello [1], Campello and Lago [2], Kassab et al. [3] and many others), there are not many examples in literature about how to measure the impact of local effects in the global response. In works from Dasambiagio et al. [4], Gonçalves et al. [5], geometrically exact rod with enriched kinematics are proposed. Despite the excellent results of practical interest, additional DOFs are inherently added to the solution.

To avoid the increase of computational complexity, surrogate approaches seem to be an attractive alternative. For 3D solid elements, innovative works regarding surrogate modelling of the constitutive equation have been published, e.g. Thakolkaran et al. [6], Flaschel et al. [7], Kissas et al. [8]. There, a surrogate is generated by combining multiple known constitutive equations, which goes through a regularized training process. Those methods rely on libraries of basic functions that are known for the general 3D solid problem. However, our case differs fundamentally: since we are surrogating the internal energy for the rod cross-section, there is no clear base to be known beforehand.

Recently, in Kassab et al. [9], we have explored alternative modeling techniques for rods, focusing on models that are built upon usual 6 or 7 DOFs per node rods, achieving great results using a two-level computational procedure that takes information from high-fidelity models to improve rod simulation results. Now, based on another previous study of our authorship [10], we aim to build a robust trainable surrogate model for the hyperelastic energy function, in terms of the usual generalized rod strains. In this work, we proposed a custom architectures, that

combines multiplicatively a known analytical estimative of the energy (quadratic function) with a feed-forward Deep Neural Network (DNN).

The core idea of the proposed technique is to represent how local effects (such as web/flange buckling) lead to the loss of stiffness of the cross-section, in a global manner. The readily available automatic differentiation tools from machine learning libraries can be used to evaluate the cross-sectional stress-resultants (gradients) and material tangent stiffness (hessian).Considering that this is our first attempt to develop this idea, we will be interested here in elastic analysis only, further restricted to the 2D case, and for a given rod cross-section.

The DNN-based approach provides enough flexibility to the learning process, but with a major trade-off: although our custom architecture is able to enforce important mechanical behavior by construction, special regularization shall be employed to encourage convexity of the energy function w.r.t. to its strain arguments. This is essential for robustness once the model is integrated in rod FEM frameworks. In this conference paper, we seek to introduce those techniques, essential to obtain robustness for our surrogate model, so that it can be directly employed in FEM simulations.

2 Base rod formulation

It is the scope of this work to develop a surrogate for the constitutive equation for a two-dimensional (2D) problem. Even though the equations here are provided for a fully three-dimensional (3D) problem, the 2D case is naturally covered, by just neglecting the irrelevant entries of the vectors.

Here, we will introduce the strictly needed definitions that stem from our previously published works (see e.g. the latest Kassab et al. [3]). Besides the constitutive equation employed, every other aspect of the formulation presented is unaltered. Consider a reference basis frame given by $\{\boldsymbol{e}_1^r, \boldsymbol{e}_2^r, \boldsymbol{e}_3^r\}$, with $\{\boldsymbol{e}_1^r, \boldsymbol{e}_2^r\}$ defining the plane in which the cross-section A^r is at initially, and $\boldsymbol{e}_3^r$ is the axial direction. One can define the reference configuration as

$$\boldsymbol{\xi} = \boldsymbol{a}^r + \boldsymbol{\zeta}, \quad \text{with} \quad \boldsymbol{a}^r = \xi_\alpha \boldsymbol{e}_\alpha^r \in A^r \text{and} \quad \boldsymbol{\zeta} = \zeta \boldsymbol{e}_3^r, \zeta \in \Omega^r = [0, L^r], \tag{1}$$

given cross-sectional director coordinates ξ_α, $\alpha = 1, 2$, axial coordinate ζ. The current configuration as

$$\boldsymbol{x} = \boldsymbol{z} + \boldsymbol{a} + \boldsymbol{w}, \quad \text{with} \quad \boldsymbol{z} = \boldsymbol{\zeta} + \boldsymbol{u}, \quad \boldsymbol{a} = \boldsymbol{Q}\boldsymbol{a}^r, \quad \boldsymbol{w} = p\psi \boldsymbol{e}_3, \tag{2}$$

given an axial displacement $\boldsymbol{u}$, a rotation tensor $\boldsymbol{Q} = \hat{\boldsymbol{Q}}(\boldsymbol{\theta}) \rightarrow \boldsymbol{e}_i = \boldsymbol{Q}\boldsymbol{e}_i^r$ (parametrized via Euler-Rodrigues formula) and a warping function ψ with associated intensity p. One can define the generalized displacement vector $\boldsymbol{d} = [\boldsymbol{u} \quad \boldsymbol{\theta} \quad p]^T$.

Now, the generalized rod strain vector is defined as $\boldsymbol{\varepsilon}^r = \begin{bmatrix} \boldsymbol{\eta}^r & \boldsymbol{\kappa}^r & p & p' \end{bmatrix}^T$ with $\boldsymbol{\eta}^r = \boldsymbol{Q}^T\boldsymbol{\eta}$ and $\boldsymbol{\kappa}^r = \boldsymbol{Q}^T\boldsymbol{\kappa}$, with $\boldsymbol{\eta} = \boldsymbol{z}' - \boldsymbol{e}_3$ and $\boldsymbol{\kappa} = \text{axial}(\boldsymbol{K})$, for $\boldsymbol{K} = \boldsymbol{Q}'\boldsymbol{Q}^T$. Those definitions are enough to proceed with the current paper. For more information about rod models solution, please refer to the latest Kassab et al. [3]. In the 2D case, the only non-zero strains are η_2^r,η_3^r and κ_1^r, which represent respectively the shear rate, the axial stretch and the bending curvature.

3 Surrogate constitutive equation

This section aims to briefly describe the architecture, data generation and training process of the surrogate model. By assuming a hyperelastic material, it is sufficient to create a surrogate for the internal energy function $\Psi^{sur} = \hat{\Psi}^{sur}(\boldsymbol{\varepsilon}^r)$. Once the surrogate is ready, stress resultant and material tangent stiffness are readly obtained by partial differentiation $\boldsymbol{\sigma}^r = \frac{\partial \Psi^{sur}}{\partial \boldsymbol{\varepsilon}^r}$ and $\boldsymbol{D}^m = \frac{\partial^2 \Psi^{sur}}{\partial \boldsymbol{\varepsilon}^r \partial \boldsymbol{\varepsilon}^r}$. Automatic differentiation can be employed to calculate those quantities, once the surrogate is known.

By employing a data-based procedure, we aim to develop a DNN-based regressive function that fits accurately to a synthetically generated database, whilst complying to several mechanical requirements. Namely, it is desirable to have

1. objectivity;
2. $\Psi^{sur} \geq 0, \forall \boldsymbol{\varepsilon}^r$, with $\Psi^{sur}(\boldsymbol{o}) = 0$;
3. $\Psi^{sur} \in C_2$, so that gradient and hessian (stress resultants and material tangent stiffness) can be obtained;

4. Convexity in $\boldsymbol{\varepsilon}^r$.

In our previously work (see Kassab et al. [9]), we show that the architecture presented there (and in section 3.1), properties 1-3 are satisfied by construction. In the original work, condition 4 was not guaranteed. Here, we propose to encourage it during training, see section 3.3.

Implementation of the model and training process is done in Pytorch environment (Paszke et al. [11]).

3.1 Model architecture: enforcing properties by construction

A specially tailored architecture, taken from our previous work (see Kassab et al. [9]) is employed. Instead of using a simple multilayer perceptron to estimate the internal energy function, we propose

$$\Psi^{sur}(\boldsymbol{\varepsilon}^r) = \chi^{DNN}(\boldsymbol{\varepsilon}^r)\Psi^{quad}(\boldsymbol{\varepsilon}^r), \tag{3}$$

where $\Psi^{quad} = \frac{1}{2}\boldsymbol{\varepsilon}^r \cdot \boldsymbol{D}\boldsymbol{\varepsilon}^r$ is the usual quadratic energy function for rods under linear elasticity (see how $\boldsymbol{D}$ is built in Kassab et al. [3]) and χ^{DNN} is a scalar-valued function, with trainable DNN architecture, that acts as a penalty of the quadratic solution. Around the strain-free configuration, χ^{DNN} should tend to one (the surrogate recovers quadratic shape), decreasing towards more intense strains. This reflects the loss of stiffness of the cross-section, as local effects onset and develop. Based on previously studied shapes of the DNN, the choice was to build it with one input layer, 6 hidden layers and an output layer, 64 perceptrons wide. The chosen activation functions are

- the Exponential Linear Unity (ELU) for the input and hidden layers;
- twice the standard logistic sigmoid for the output layer, to avoid saturation around $\boldsymbol{\varepsilon}^r = \boldsymbol{o}$ (where $\chi^{DNN} = 1$ is expected).

Condition 1 from section 3 is immediatly satisfied, since objective strain variable $\boldsymbol{\varepsilon}^r$ is its only argument. The aforementioned architectural choices guarantee compliance to conditions 2-3 from section 3. Condition 4 is discussed in section section 3.3.

3.2 Data generation

As in our previous work (Kassab et al. [9]), we focused on a I-shaped cross-section (height $h = 10\,\text{cm}$, width $b_f = 5.3\,\text{cm}$ and thickness $t = 0.6\,\text{cm}$), with constant material property (Simo-Ciarlet's material, with $E = 200\,\text{GPa}$ and $G = 80\,\text{GPa}$). To establish a correspondence between higher and lower order models, a rod of representative length $L = 24\,\text{cm}$ was employed, with one clamped end and various boundary conditions (BCs) on the other one. Training and testing dataset were generated, based on values from Appendix A of the aforementioned work. In total, we have $n_{sim} = n_{train} + n_{test}$ simulation cases in our synthetic dataset.

For a given input BC, a simulation label-input pair corresponds to a higher-order 3D solid model (from FEAP [12]) and a lower-order rod model (from our in-house code PEFSYS), which is discretized in $n_{el} = 100$ elements of the type described in section 2. For a simulation BC named k, in a load-step t, we record respectively for each model the total internal energy $E_k^{(t)}$ (label) and the collection of generalized rod strains $\boldsymbol{\varepsilon}^r|_{k,e}^{(t)}$ (input) for each element e.

For many of the imposed BCs, FEAP solver could not reach the final load-step $t = 1$. Whenever high-fidelity simulation fails to reach the final load-step $t = 1$, extrapolation of data is performed, by extending the curve t vs. $E_k^{(t)}$ up to $t = 1$ with a tangent line. Once curves *total internal energy vs. load-step* and *generalized rod strains vs. load-step* are sampled, usual interpolation techniques (e.g. cubic splines) can be employed to regularly sample *load-step vs. generalized rod strains vs. total internal energy*, in n_{steps} equally spaced load-steps.

Whilst constitutive equation for 3D solid model is imposed (Simo-Ciarlet's), the one for the rod model is the unknown of our problem. Therefore, the strain field derived from the rod model should be the best tentative available, e.g. using standard material for rods. Here, this estimation was obtained using linear elastic material. Then, a total energy estimative of the rod model is calculated by direct integration, which yields

$$E_k^{sur(t)} \approx \sum_{e=1,n_{el}} \Psi^{sur}(\boldsymbol{\varepsilon}^r|_{k,e}^{(t)})l_e = \frac{L}{n_{el}} \sum_{e=1,n_{el}} \Psi^{sur}(\boldsymbol{\varepsilon}^r|_{k,e}^{(t)}). \tag{4}$$

Note that now, the sum of multiple evaluations of the surrogate are needed to match one energy measure the high-fidelity simulation, instead of a one-to-one relationship. This acts naturally as a regularizer for the surrogate, as the ensemble of elements must output on average $E_k^{(t)}/L$.

Now, let us collect all the simulation data in two tensors:

- the third-order array of generalized rod strains $\boldsymbol{\mathcal{E}} = [\mathcal{E}_{iej}]$. Each entry with index i of $\mathcal{E}_{iej}$ is obtained by stacking all the load-steps t within all simulations k, so that $i = (k-1)*n_{steps}+t$ (thus, $\mathcal{E}_{[(k-1)*n_{steps}+t]ej} = \varepsilon^r_j|^{(t)}_{k,e}$), which yields a shape of $(n_{sim} * n_{steps}, n_{el}, 8)$.
- A second-order array of stacked total internal energy $\boldsymbol{E} = [E_i]$, with shape $(n_{sim} * n_{steps}, 1)$.

We proceed now to a component-wise normalization step, which is a common practice in machine learning (ML), since the optimization algorithms usually perform better in the normalized space. Here, normalization by maximum absolute value is performed for both total energy and strains (for each entry j), yielding

$$\tilde{\boldsymbol{\mathcal{E}}}_{:,:,j} = \frac{\boldsymbol{\mathcal{E}}_{:,:,j}}{|\boldsymbol{\mathcal{E}}_{:,:,j}|_\infty} \longrightarrow \tilde{\varepsilon}^r_j|^{(t)}_{k,e} = \frac{\varepsilon^r_j|^{(t)}_{k,e}}{|\boldsymbol{\mathcal{E}}_{:,:,j}|_\infty} \quad \text{and} \quad \tilde{\boldsymbol{E}} = \frac{\boldsymbol{E}}{|\boldsymbol{E}|_\infty}, \tag{5}$$

where $(\circ)_{:,:,j}$ is a slicing operation over fixed third index j, which results in second-order tensor and $|\circ|_\infty = \max(||(\circ)_{ijk}||)$. Ultimately, this means that the surrogate model that is actually trained is a normalized one in terms of normalized strains, such that $\tilde{\Psi}^{sur} = \tilde{\Psi}^{sur}(\tilde{\boldsymbol{\varepsilon}}^r)$, for $\tilde{\boldsymbol{\varepsilon}}^r = \boldsymbol{G}^{-1}\boldsymbol{\varepsilon}^r$, for $\boldsymbol{G} = \text{diag}(|\boldsymbol{\mathcal{E}}_{:,:,1}|_\infty \quad ... \quad |\boldsymbol{\mathcal{E}}_{:,:,8}|_\infty)$ holds.

Note that the rescaling process preserves the *zero strain equals zero energy* condition. It is also evident that the normalized strain inputs lay in the interval $[-1, 1]$, whilst normalized energy is in $[0, 1]$. Once the model is trained in the normalized space, de-normalization is needed for it to be used in simulations.

3.3 Loss metric: encouraging desirable properties

The loss function used in the current work is the usual Mean Squared Error (MSE), with additional penalties to encourage convexity and regularization of the network. For a given state of the surrogate during training, in which the DNN's weights and biases are concatenated in a single parameter vector $\boldsymbol{\nu}$, we propose for a batch of normalized pairs $(\tilde{\boldsymbol{\mathcal{E}}}, \tilde{\boldsymbol{E}})$

$$\mathcal{L}(\boldsymbol{\nu}, \tilde{\boldsymbol{\mathcal{E}}}, \tilde{\boldsymbol{E}}) = MSE(\boldsymbol{\nu}, \tilde{\boldsymbol{\mathcal{E}}}, \tilde{\boldsymbol{E}}) + w_{L2}||\boldsymbol{\nu}||^2 + w_1\mathcal{P}_1(\boldsymbol{\nu}) + w_H\mathcal{P}_{H\geq 0}(\boldsymbol{\nu}) + w_{dh}\mathcal{P}_{dH\leq 0}(\boldsymbol{\nu}), \tag{6}$$

where $MSE(\boldsymbol{\nu}, \tilde{\boldsymbol{\mathcal{E}}}, \tilde{\boldsymbol{E}})$ is the usual MSE loss for the batch in the current model state, $w_{(\circ)}$ are weights for the penalties. The function $\mathcal{P}_1$ is a penalty that acts on sampled points of Ψ^{sur} near zero strain, penalizing deviations from $\chi^{DNN} = 1$. The function $\mathcal{P}_{H\geq 0}$ penalizes negative Hessian, by evaluating $\boldsymbol{dx} \cdot \boldsymbol{H}\, \boldsymbol{dx}$, for a random small $\boldsymbol{dx}$, for randomly sampled points of the normalized domain. Lastly, the function $\mathcal{P}_{dH\leq 0}$ penalizes increased stiffness for higher values of strains, for evolution in the radial direction: for sampled (normalized) strain points $\tilde{\boldsymbol{\varepsilon}}^r$, the function penalizes if $\beta\tilde{\boldsymbol{\varepsilon}}^r \cdot \boldsymbol{H}(\tilde{\boldsymbol{\varepsilon}}^r)\beta\tilde{\boldsymbol{\varepsilon}}^r < \beta\tilde{\boldsymbol{\varepsilon}}^r \cdot \boldsymbol{H}((1+\alpha)\tilde{\boldsymbol{\varepsilon}}^r)\beta\tilde{\boldsymbol{\varepsilon}}^r$, for a positive value α and a small positive value of β. During training, it is less expensive to estimate Hessian using a finite difference scheme than to use automatic differentiation, since only one directional increment is needed.

Encouraging $\chi^{DNN}(\boldsymbol{o}) = 1$ avoids bias that potentially stems from the refined 3D solid model, in a region of the domain where absolute error is small (thus with little learning capacity). By encouraging the surrogate to have non-negative hessian, we are trying to impose convexity. Lastly, encouraging non-increasing stiffness preserves the physical interpretation that local effects due to web/flange buckling overall reduce cross-sectional stiffness. Note that the Hessian regularization terms can be applied to a normalized range broader than $[-1, 1]$, which allows a stable extrapolation both inside and slightly beyond the explored range.

3.4 Training process

The training process consists on minimizing eq. (6). Adam optimization algorithm (see Kingma and Ba [13]) is employed. In Pytorch, L2 regularization is readily available, by passing to the optimizer the optional parameter $weight_decay$ (in general, weights decay is not equivalent to L2 regularization, but the implementation in Pytorch effectively corresponds to L2 regularization). The other penalties required custom implementation. For each training epoch, the train dataset is divided in batches, that are reshuffled every epoch.

Training is performed in steps, so that the gradients stemming from the regularization terms does not overwhelm the squared loss error too early. Initially, the penalties that are related to the hessian regularization are turned off, allowing the optimization process to learn the general aspects of the model. Then, the weights of the

disabled penalties are enabled, allowing the model to readjust its internal weights. For now, learning rate (lr) and hyperparameters associated to the penalties are manually tuned as training progresses.

4 Results: the trained model

Using the proposed architecture, data generation and training process, a surrogate was trained. From Fig. 1, the nonlinear effects due to local buckling are clearly seen: the multiplicative factor plummets quickly after the onset of local processes. Important coupling between bending curvature and axial stretch is present: bending and compression simultaneously importantly decrease χ^{DNN} (note darker regions in Fig. 1b) in regions with negative axial stretch and non-zero bending curvature). Note also that values outside the known data range are also shown, illustrating the extrapolation capacity without hallucination of the model.

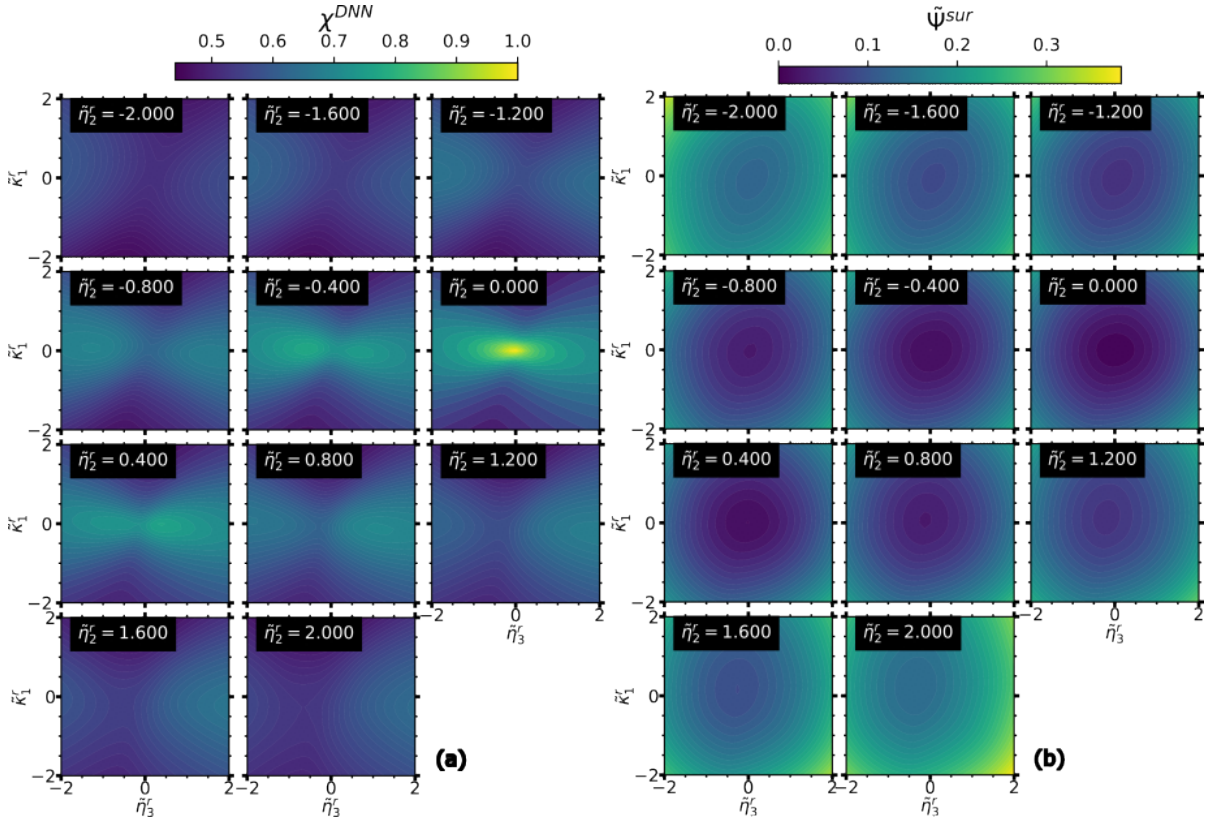


Figure 1. **a)** Multiplicative factor χ^{DNN} after training surrogate model and **b)** final normalized surrogate model $\tilde{\Psi}^{sur}$. Slices of constant $\tilde{\eta}_2^r$ levels, as functions of $\tilde{\eta}_3^r$ and $\tilde{\kappa}_1^r$.

5 Conclusions

A surrogate model for constitutive equation of a particular thin-walled I-section has been trained for the 2D case. Mechanical consistency conditions are satisfied, some by construction, some by penalty during training. The model is readily available to be used in a rod FEM framework. A comprehensive description of the framework proposed here, along with novel results, will be given in a forthcoming journal paper by the authors.

Acknowledgements. This work was supported by Fapesp (São Paulo State Research Foundation), Brazil, grants number 2023/16272-1, 2022/15644-0 and 2020/13362-1, CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico), Brazil, under the grants 313046/2021-2 and 306132/2024-9, as well as by ANR (Agence Nationale Recherche) in France under the grant ANR-20-CE46-0012-01 and IUF (Institut Universitaire France) under grant 1479.

Authorship statement. The authors hereby confirm that they are the sole liable persons responsible for the authorship of this work, and that all material that has been herein included as part of the present paper is either the property (and authorship) of the authors, or has the permission of the owners to be included here.

References

[1] E. M. B. Campello. Análise não-linear de perfis metálicos conformados a frio. Master's thesis, University of São Paulo, 2000. doi: 10.11606/D.3.2000.tde-24052024-074053.

[2] E. M. B. Campello and L. B. Lago. Effect of higher order constitutive terms on the elastic buckling of thin-walled rods. *Thin-Walled Structures*, vol. 77, pp. 8–16, 2014. doi: 10.1016/j.tws.2013.11.001.

[3] M. P. Kassab, E. M. B. Campello, and P. M. Pimenta. Advances on kinematically exact rod models for thin-walled open-section members: Consistent warping function and nonlinear constitutive equation. *Computer Methods in Applied Mechanics and Engineering*, vol. 407, pp. 115933, 2023. doi: 10.1016/j.cma.2023.115933.

[4] E. R. Dasambiagio, P. M. Pimenta, and E. M. B. Campello. A finite strain rod model that incorporates general cross section deformation and its implementation by the finite element method. In *Mechanics of Solids in Brazil*, volume 1, pp. 145–168, Rio de Janeiro. H. S. da Costa Mattos and M. Alves, Brazilian Society of Mechanical Sciences and Engineering, 2009.

[5] R. Gonçalves, M. Ritto-Corrêa, and D. Camotim. A large displacement and finite rotation thin-walled beam formulation including cross-section deformation. *Computer Methods in Applied Mechanics and Engineering*, vol. 199, pp. 1627–1643. Number: 23-24, 2010. doi: 10.1016/j.cma.2010.01.006.

[6] P. Thakolkaran, A. Joshi, Y. Zheng, M. Flaschel, L. De Lorenzis, and S. Kumar. Nn-euclid: Deep-learning hyperelasticity without stress data. *Journal of the Mechanics and Physics of Solids*, vol. 169, pp. 105076, 2022. doi: 10.1016/j.jmps.2022.105076.

[7] M. Flaschel, S. Kumar, and L. De Lorenzis. Automated discovery of generalized standard material models with euclid. *Computer Methods in Applied Mechanics and Engineering*, vol. 405, pp. 115867, 2023. doi: doi.org/10.1016/j.cma.2022.115867.

[8] G. Kissas, S. Mishra, E. Chatzi, and L. De Lorenzis. The language of hyperelastic materials. *Computer Methods in Applied Mechanics and Engineering*, vol. 428, pp. 117053, 2024. doi: 10.1016/j.cma.2024.117053.

[9] M. P. Kassab, E. M. B. Campello, and A. Ibrahimbegovic. A geometrically exact thin-walled rod model with warping and stress-resultant-based plasticity obtained with a two-level computational approach. *Computer Methods in Applied Mechanics and Engineering*, vol. 433, pp. 117497, 2025a. doi: 10.1016/j.cma.2024.117497.

[10] M. P. Kassab, E. d. M. B. Campello, and A. Ibrahimbegovic. A DNN-based surrogate constitutive equation for geometrically exact thin-walled rod members. *Computation*, vol. 13, n. 3, 2025b. doi: 10.3390/computation13030063.

[11] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Köpf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala. Pytorch: An imperative style, high-performance deep learning library, 2019. doi: 10.48550/arXiv.1912.01703.

[12] R. L. Taylor. FEAP - finite element analysis program, 2014.

[13] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *CoRR*, vol. abs/1412.6980, 2014. doi: 10.48550/arXiv.1412.6980.