27th INTERNATIONAL CONGRESS OF MECHANICAL ENGENEERING
COBEM 2023
ABCM Brazilian Society of Mechanical Sciences and Engineering
27th ABCM International Congress of Mechanical Engineering
December 04-08, 2023, Florianópolis-SC, Brazil

COB-2023-0206
DEVELOPMENT OF A PYTHON SCRIPT FOR STRUCTURAL ANALYSIS BASED ON THE FINITE ELEMENT THEORY

Dawson Tadeu Izola
Roberto Wagner Bressan Junior
University Center of the Hermínio Ometto Foundation
dawson@fho.edu.br
robertowbj@alunos.fho.edu.br

Abstract. *Facing the complexity of the structures, the use of advanced software is necessary to verify the safety and the maximum load capability of it. The finite element method is implemented to solve this problem, enabling the analysis for atypical structures. Several software are available, however, to really understand all the processes that occur inside the "black box" of this type of software, a Python script that can receive, interpret and calculate the stresses acting on a plate with any shape is developed. The constant strain triangle element is used to calculate the strains on the structure. Furthermore, the Python packages used are, mainly, the numpy and matplotlib. Finally, to ensure the effectiveness of the script, two comparisons are considered: the first is with the Ansys software and the second is with the strain measurement technique using the quarter-bridge configuration. In both cases, the script developed presents great similarity with the tests described, showing that, for a linear elastic analysis and considering static loading and ductile materials, the displacements of the analyzed structure on the script and on the commercial software are obtained, basically, by the elementary equation of the finite element method.*

Keywords: *finite element, Python, strain gauge, plane stress, von Mises stress, linear elastic analysis*

1. INTRODUCTION

Considering a structural problem, the finite element method uses mathematical equations associated with physical concepts to describe the behavior of an object under load. To solve this kind of problem, numerical methods are used and, due to the nature of this, the results are not exact solutions, therefore, the precision of the analysis must always be evaluated (Bathe, 1996).

The strain gauge measurement is an instrumentation technique that is used to verify the stresses and strains acting on a structure. To measure small deformations, a sensor that varies its resistance in function of its length is used, this sensor is called strain gauge. The small variations in the resistance are converted into electrical potential difference by associating the strain gauge in an electric circuit called Wheatstone bridge (Jesus Junior and Rêgo Segundo, 2019).

The reason to develop a script in Python to solve structural problems by the finite element theory is to elucidate the calculations occurring inside the “black box” of the commercial software and to really understand the role of each parameter defined in the analysis, and how the element type influences the boundaries conditions that can be applied and the expected results in the end of the analysis.

2. BIBLIOGRAPHY REVISION

2.1 Finite element method

Considering a structural analysis that uses the stiffness (or displacement) method, the main objective is to obtain the displacement matrix, that contains, in each row, the displacement for each degree of freedom. With this solved matrix, it is possible to obtain the stresses and strains acting on the structure in the post-processing phase.

$$\{F\} = [K]\{d\} \tag{1}$$

For a linear elastic analysis, the Eq. (1) correlates the displacements $\{d\}$ and the forces $\{F\}$ through the global stiffness matrix $[K]$ (Logan, 2007).

2.2 Stiffness matrix

The physical meaning of the stiffness matrix is to describe the forces in means of the displacements. Each term K_{ij} of the matrix indicates the force acting on the node i due to a unit displacement in the node j (Alves Filho, 2000).

For complex elements, like the ones used in 2D and 3D analysis, the element stiffness matrix is obtained by the principle of the minimal potential energy, that imposes that a body tends to deform to an equilibrium state and this state occurs at a minimum point of the total potential energy of the system (Kelly, 2013).

$$\frac{\partial \Pi}{\partial d} = \frac{\partial U}{\partial d} + \frac{\partial \Omega}{\partial d} = 0 \tag{2}$$

In Eq. (2) Π corresponds to the total potential energy of the system, U is the energy of the deformation of the element and Ω is the opposite of the work generated by the external forces. It is important to notice that each term is derived in relation to the displacement d.

Firstly, the stiffness matrix for all the elements is obtained, then the global stiffness matrix is obtained by the superposition of all elements stiffness matrices. This method is called direct stiffness method (Kassimali, 2012) (Bathe, 1996).

2.3 The constant strain triangle element

According to Logan (2007) the constant strain triangle (CST) element is used for plane analysis and, as the name suggests, all the area of this element presents just one value for the strain. The CST element has three nodes with two degrees of freedom each (translations in X and Y directions).

The element stiffness matrix is obtained by the principle of the minimal potential energy. Equation 3 gives the mathematical expression considering a constant thickness through all the element extensions (Logan, 2007).

$$[k] = tA[B]^T[D][B] \tag{3}$$

Where t is the thickness of the plate, A is the element area, $[B]$ is the matrix that gives the strains in means of the displacements and $[D]$ is the matrix that correlates the stresses with the strains considering a plane stress situation.

Due to the degrees of freedom of the CST element, the loads and restrictions can be set just in respect of the X and Y translations.

As the CST has a constant strain value in all the area of the element, for analyses subjected to bending loads, it is better to create a mesh with small elements to assess the real distribution of stresses and strains in relation of the height of the structure, since the constant strain triangle has a stiffer mathematical model in comparison to the real case (Logan, 2007).

2.4 Principal stresses and maximum shear stress

The principal stresses correspond to the maximum and minimum normal stresses on the element and can be obtained by the Mohr's Circle or by the application of the Eq. (4) (Hibbeler, 2004).

$$\sigma_{max,min} = \frac{\sigma_x + \sigma_y}{2} \pm \sqrt{\left(\frac{\sigma_x - \sigma_y}{2}\right)^2 + \tau_{xy}^2} \tag{4}$$

Where σ_x, σ_y and τ_{xy} are the stresses on the plane of the infinitesimal element and the results (σ_{max} and σ_{min}) are the element principal stresses.

The maximum shear stress in a plane can be understood as the maximum numerical value for the shear stress acting on the plane of the applied forces. It is important to notice that for a 2D case, the maximum shear stress given by the Eq. (5) can be an intermediate value (Hibbeler, 2004).

$$\tau_{max} = \sqrt{\left(\frac{\sigma_x - \sigma_y}{2}\right)^2 + \tau_{xy}^2} \tag{5}$$

The term τ_{max} corresponds to the maximum shear stress of a 2D study.

2.5 von Mises stress

The von Mises stress is a failure theory for ductile materials that compares the strain energy for a multiaxial stress state with the required energy to yield a specimen on a tensile testing (Norton, 2013).

$$\sigma' = \sqrt{\sigma_{max}^2 - \sigma_{max}\sigma_{min} + \sigma_{min}^2} \tag{6}$$

The term σ' represents the equivalent stress that must be compared to the yield stress of the analyzed material.

2.6 Strain gauge measurements

To measure the strains on an object, a strain gauge can be used. This sensor must be really well glued on the surface of the sample in question. This is necessary, since the length of the metallic filaments of the strain gauge has a straight relationship with its resistance, therefore any length change dictates a resistance change too.

By associating the strain gauge in the electric circuit called Wheatstone bridge, shown by Fig. 1, the variation in resistance can be measured as a variation in tension V_o (Jesus Junior and Rêgo Segundo, 2019).

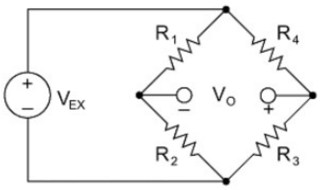

Figure 1. Wheatstone bridge circuit example (Instruments, N.D.).

The association can be with one, two or four strain gauges, configuring the quarter-bridge, half-bridge and full-bridge arrangements respectively.

3. METHODOLOGY

3.1 Script functions

The script written in Python language is able to receive an already discretized plane geometry, interpret the nodes and degrees of freedom that have a boundary condition associated, establish the global stiffness matrix considering the CST element, obtain the solution of the linear system and display the desired results to the user.

An object-oriented programming is used and each of the three implemented classes corresponds to the phases involved in a finite element analysis on a commercial software: pre-processing, processing and post-processing. The Fig. 2 represents a schematic flowchart of the script.

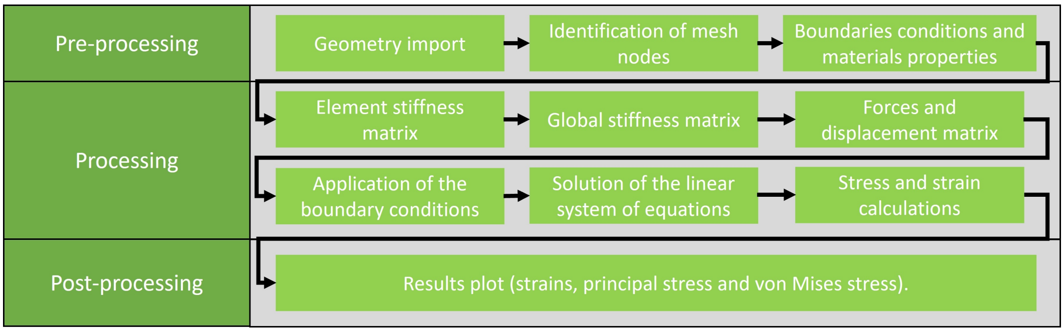

Figure 2. Simplified flowchart of the script.

3.1.1 Mesh generation

The geometry of the plate is generated by an open-source software, the Gmsh. This application is able to receive the inputs of the 2D geometry passed by the user and build the mesh with triangular elements. Furthermore, the user can identify the entities used as boundary conditions, informing the respective vertices, edges or surfaces as a physical group. The software utilized has a graphical interface and is able to discretize geometry that presents holes features (Edholm, 2013).

3.1.2 Pre-processing

The pre-processing class of the script is responsible for receiving the file that contains the information about the geometry, interpreting and identifying each node of the mesh and importing the nodes according to the defined physical groups. The meshio package is used to ease the geometry processing step, this package has methods and attributes that indicate the Cartesian coordinates of all nodes and groups the nodes that describe each element of the mesh.

Still considering the pre-processing class, the user can define the properties of the material, the thickness of the plate, the external forces and store the constraints for each degree of freedom in the informed physical group.

3.1.3 Processing

To ensure the correct correlation among the force matrix, displacement matrix, element matrix and the global stiffness matrix after its assemblage, each degree of freedom is indexed by the node number as shown by Eq. (7) and Eq. (8).

$$d_x = 2N_{node} \tag{7}$$

$$d_y = 2N_{node} + 1 \tag{8}$$

Where d_x and d_y are the translational degrees of freedom of the CST element and the N_{node} is the active node number.
Seeking fast operations involving the described matrices, the numpy array object is used. In addition, the global stiffness matrix is declared as a square matrix of zeros with order equal to the total degrees of freedom of the structure. After the assemblage of each element matrix, the direct stiffness method is applied, incrementing each term of the global stiffness matrix considering the indexing system given by Eq. (7) and Eq. (8).

In sequence, the constraints stored on the pre-processing phase are employed to simplify the global stiffness matrix and the external forces matrix by eliminating rows and columns of these matrices where the translation is set as constrained.

With the reduced terms, the system of linear equations is solved by the numpy.linalg.solve() function. The solution corresponds to all unknown displacements of the nodes of the structure.

Finally, the stresses and strains acting on the element, as well as the principal stresses, the maximum shear stresses and the von Mises stresses are calculated by the use of the complete displacement matrix. These parameters must be calculated separately for each element and stored in the rows of the results matrix. All the contour types described have a specific column on the results matrix.

3.1.4 Post-processing

In this phase, the requested result by the user is shown graphically. This task is accomplished using the matplotlib package.

A general plot function is implemented and, by calling the result that needs to be evaluated by its respective method name, the contour and the corresponding values are updated in association with the results matrix obtained on the Processing phase.

3.2 Comparative methods

To evaluate the results given by the script and to ensure its effectiveness, two comparative methods are used: a computational comparison with the Ansys software and a strain gauge measurement comparison considering a sample under tensile load.

3.2.1 Computational comparison

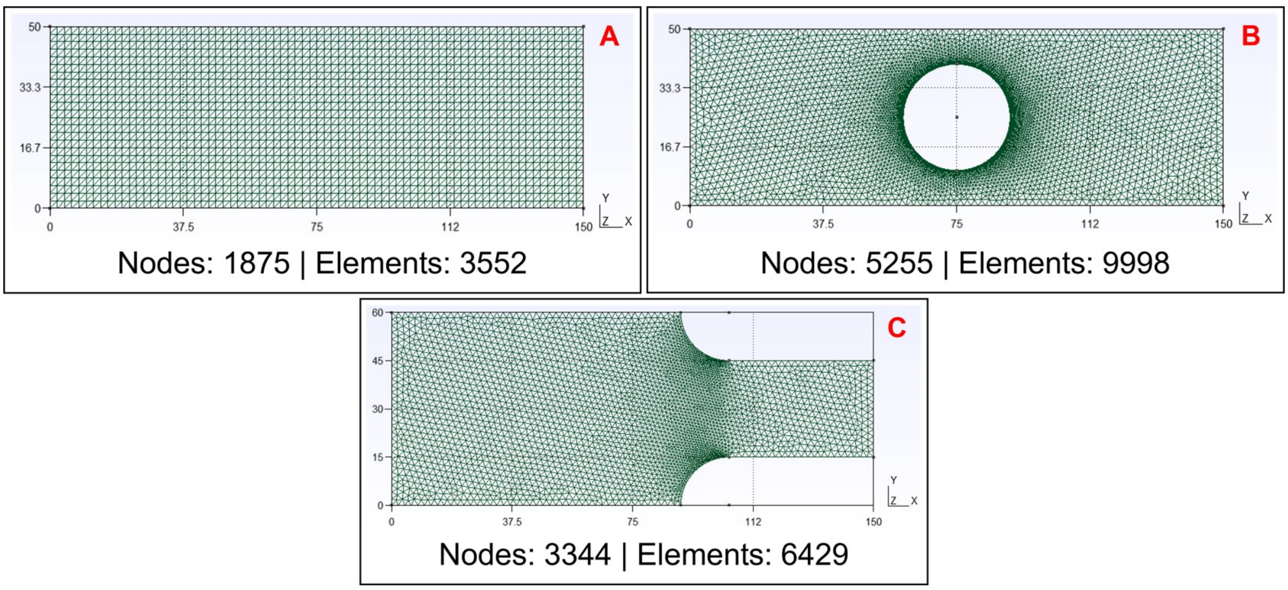

Figure 3. Geometries considered for comparison (A - plane plate / B - hole plate / C - notch plate)

The 2D geometries (and their meshing statistics) used for the comparison can be observed in Fig. 3. Each plate is evaluated considering three types of loads: tension, compression and bending. For all the cases, the magnitude of the force is 1000N and a Young's modulus of 207GPa with a 0.3 Poisson's ratio are used as the material properties. All the plates are fixed on the left extremity and the loads applied on the other side.

To ensure the robustness of the analysis, the Ansys software is restricted for a 2D analysis and the PLANE182 triangular element is used, that is analogous to the CST element of the script. Besides it, any non-linear effects are disabled on the commercial software.

Defining the maximum and minimum von Mises stresses given by the commercial software as the reference value. The relative error between these stresses in comparison with the respective values of the script should be less than 5% to frame the script as acceptable.

3.2.2 Strain gauge comparison

A steel sheet with total dimensions of 205.4mmX25.4mmX3.30mm is used in association with a quarter-bridge configuration. The strain gauge with gauge factor of 2, nominal resistance of 120 ohms and dimensions of 8mmX5mm is glued to the sanded and clean surface of the sheet.

The module NI9219 fabricated by National Instruments is used for the data acquisition. This hardware has 4 analog input channels and a configurable sampling time of 10ms, 110ms, 130ms and 500ms as well as a configurable Wheatstone bridge, dispensing the use of extra resistors. The acquisition code uses the nidaqmx package and is also written in Python language.

The calibration procedure is performed by a static transverse load, starting with 1kg and considering an increment of 1kg until the final value of 10kg. The sampling frequency used is 2Hz and corresponds to the maximum resolution frequency according to the NI9219 specifications (Instruments, 2023). The analytical strain, that is compared with the strain measured by the strain gauge, is obtained by applying Eq. (9), which is in accordance with the free body diagram in Fig. 4.

$$\varepsilon = \frac{0.2066 + 0.54F}{E \cdot 25.4 \times 10^{-3} \cdot (3.30 \times 10^{-3})^2} \tag{9}$$

Where ε is the strain, F is the force in Newton and E is the Young's modulus in Pascal.

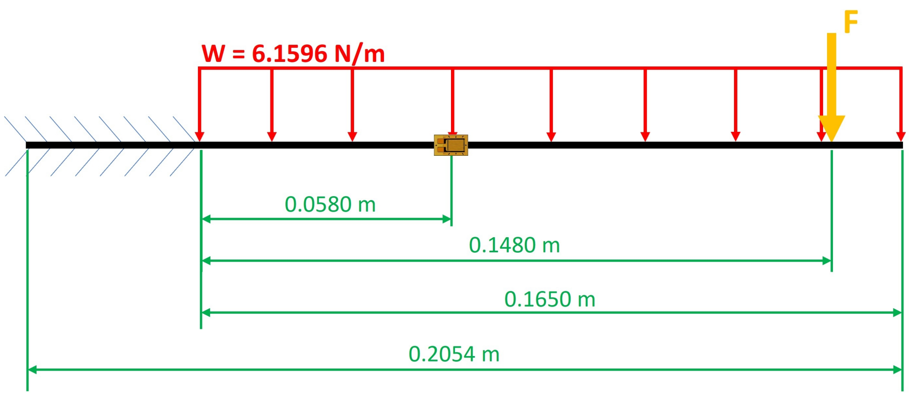

Figure 4. Corresponding free body diagram of the analytical strain equation

Considering the degrees of freedom of the CST element, a tensile load is applied in order to compare the strains measured by the strain gauge and the given ones by the script. The BioPDI tensile testing machine, which has a maximum capacity of 100kN, is used to stretch the sample.

The sampling frequency for the test above corresponds to 100Hz and the maximum force during the load application is equal to 8289.51N. Furthermore, by the Force vs. Position graph given by the tensile testing machine, the Young's modulus of the material corresponds to 10.53GPa, which, although outside the values described in the literature, is used because it is an empirical value and the exact origin of the sheet material is not known.

Following the same acceptance criterion as the computational comparison, the relative error between the strain values given by the strain gauge and by the script, when considering the force of 8289.51N, should be less than 5%.

4. RESULTS

4.1 Computational comparison

The Figures 5, 6 and 7 show the results of the script (pyFHOFEA) and of the Ansys software.

The maximum and minimum von Mises stress value of each geometry and load case is exactly the same, therefore the relative error between the script and the Ansys software is equal to zero.

By a basic visual inspection, the similarity between the color distribution of the two solutions is remarkable. Furthermore, considering the elementary theory of solid mechanics, it is possible to notice the stress concentration in the geometries with the hole and with the notch. In a similar manner, the stresses over the neutral axis of the plane geometry under the effect of the bending load is near to zero as expected.

Finally, in every situation, a stress singularity in the region of the left edge can be observed. This behavior can also be seen in the Ansys software solution and is due to the constraint applied.

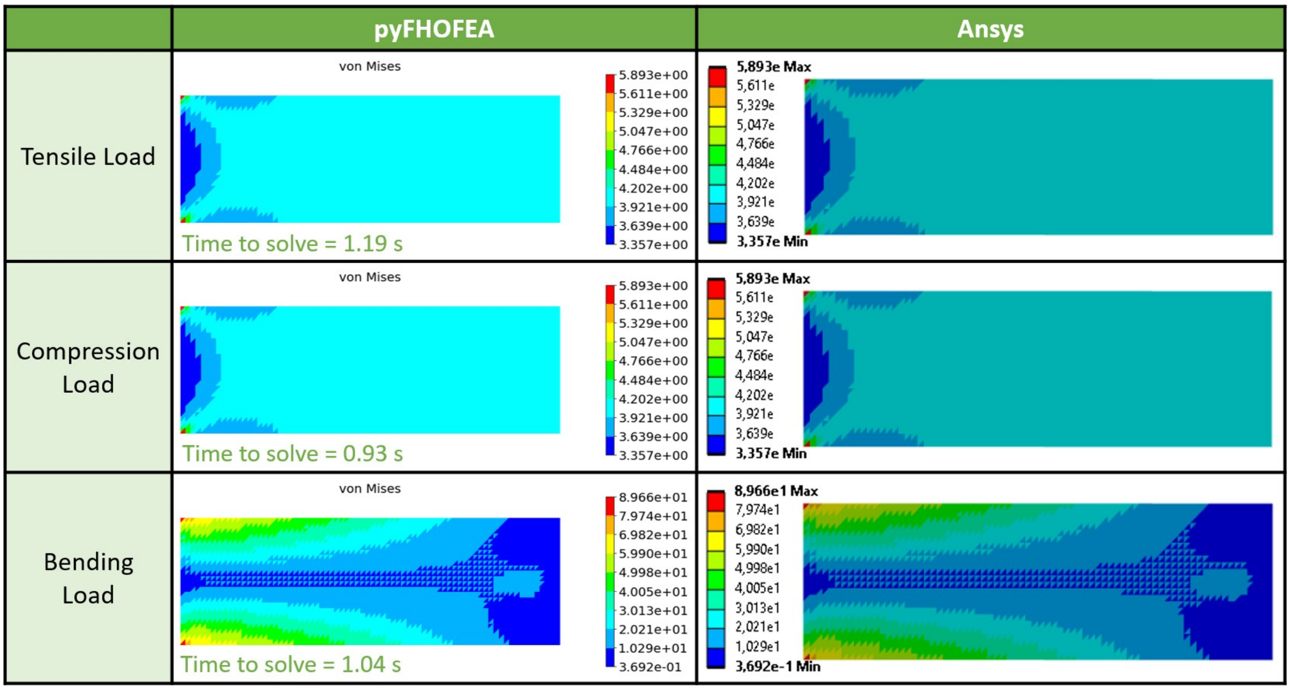

Figure 5. Results comparison for the plane geometry

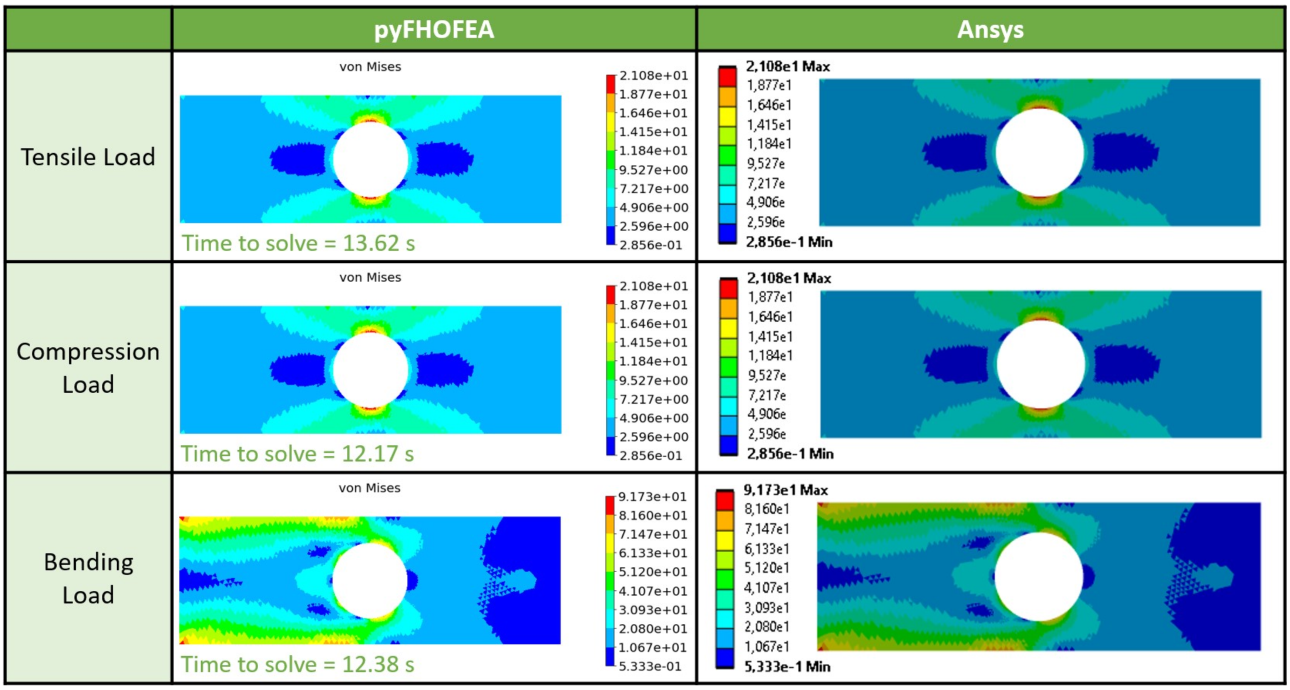

Figure 6. Results comparison for the hole geometry

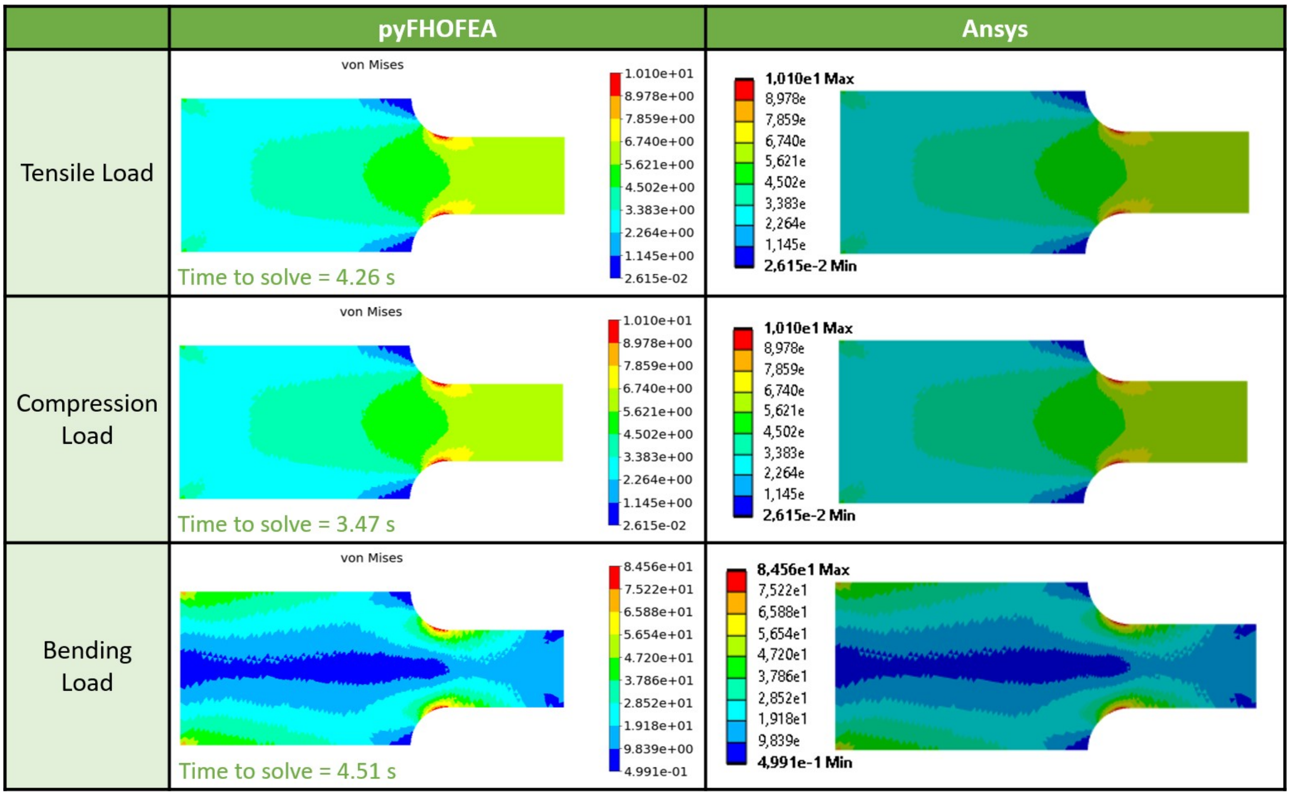

Figure 7. Results comparison for the notch geometry

4.2 Strain gauge comparison

The calibration curve and the corresponding equation can be observed in Fig. 8. Considering the R^2 coefficient, the linear regression for the analytical strain (Y axis) and the measured strain (X axis) is acceptable.

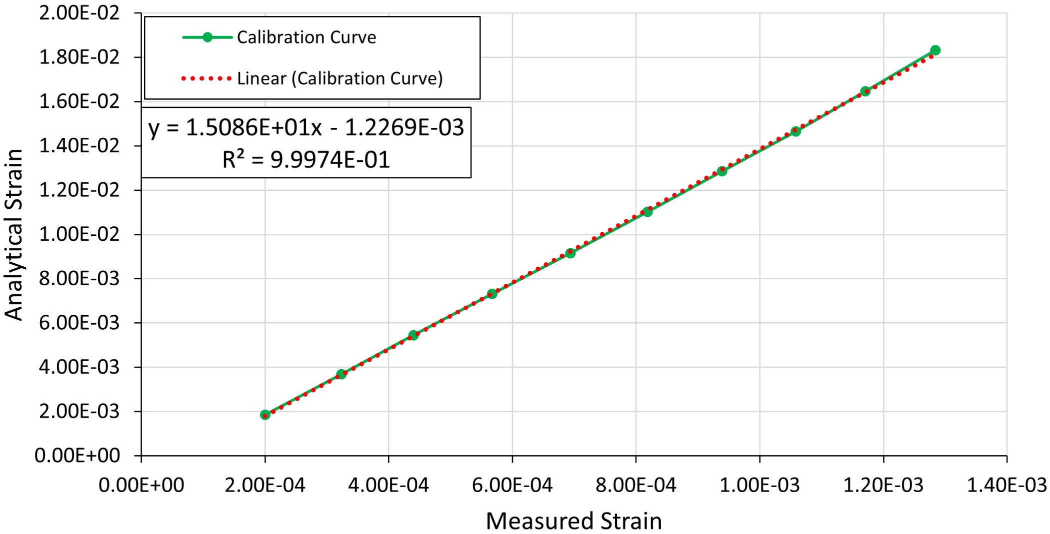

Figure 8. Calibration curve of the strain gauge (quarter-bridge)

The graph in Fig. 9 corresponds to measurements of the strain gauge during the tensile testing. Using the last part of the data as reference, the mean value of the strain is, approximately, 8.970×10^{-3}. The corresponding strain in the region of the strain gauge, calculated by the script when the force of 8289.51 N is applied, is equal to 9.392×10^{-3} as can be observed on Fig. 10.

The relative error in respect of the two strain values above is 4,70%, that is near, but still under the limit defined in the methodology phase.

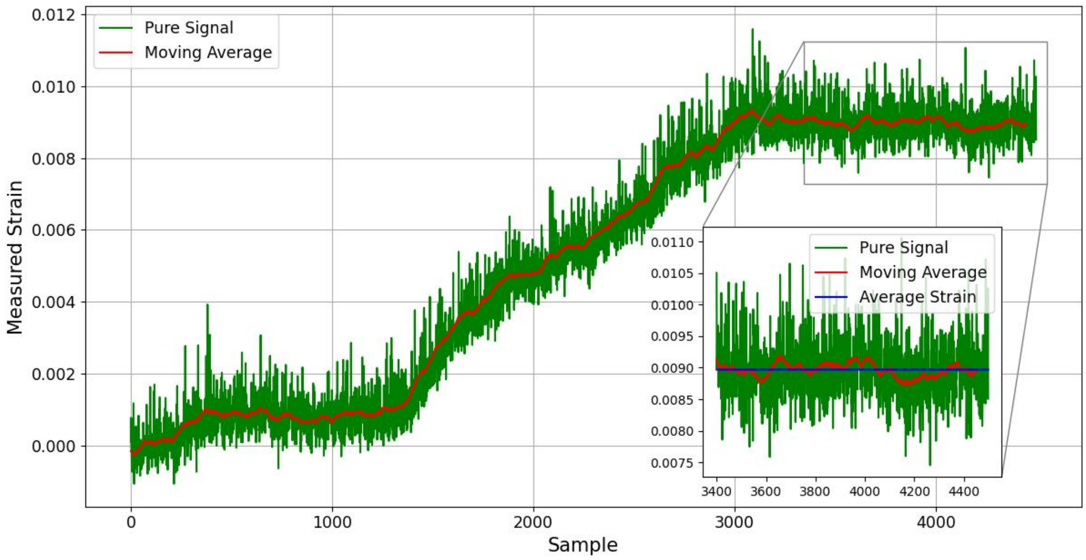

Figure 9. Measured strain during the tensile testing

Figure 10. Strain on the area of the strain gauge

5. CONCLUSION

The script for structural analysis, based on the finite element theory written in Python language, presents analogous results to those of the Ansys software, a fact that can be explained, mainly, by the employed methodology, considering the same simplifications, boundary conditions, material properties and degrees of freedom of the element. Thus, it is evident that the calculations executed during the Processing phase of the commercial software for a linear elastic analysis under static load corresponds to the solution given by Eq. (1).

Although the Young's modulus of the plate used as a sample is far from the limits accepted by the literature, the strain values obtained by the strain gauge and the one given by the script are close and are below the maximum allowed relative error.

Therefore, the script developed to solve structural problems, considering linear elastic regime, static load and ductile materials, is effective, presents reliable results and is able to solve any kind of plane geometry with unconventional shapes.

The development of this work contributes to the understanding of the inputs, outputs and all the phases of any commercial software that uses the finite element method. It is also remarkable the importance of knowledge about the theory related to the chosen element and its respective degrees of freedom, considering the boundary conditions and the expected results.

Some possible improvements of the script are the implementation of other elements types and geometric and material nonlinearities. Finally, the code is available at GitHub and can be accessed by the following link:
github.com/RobertoWBJ/pyFHOFEA.

6. REFERENCES

Alves Filho, A., 2000. *Elementos Finitos: a base da tecnologia CAE*. Érica, Tatuapé.
Bathe, K.J., 1996. *Finite Element Procedures*. Prentice Hall, S.L.
Edholm, A., 2013. *Meshing and Visualisation Routines in the Python Version of Calfem*. Master's thesis, Division of

Structural Mechanics, Lund University, Lund, Sweden.

Hibbeler, R.C., 2004. *Resistência dos Materiais*. Prentice Hall, São Paulo.

Instruments, N., 2023. "Ni-9219 specifications". https://www.ni.com/docs/en-US/bundle/ni-9219-specs/page/specs.html. Accessed 14 June 2023.

Instruments, N., N.D. "Measuring strain with strain gages". https://www.ni.com/pt-br/shop/data-acquisition/sensor-fundamentals/measuring-strain-with-strain-gages.html. Accessed 14 June 2023.

Jesus Junior, B.F. and Rêgo Segundo, A.K., 2019. "Sistema de aquisição de dados de extensometria aplicado a um tambor descascador de toras de madeira". In *Simpósio Brasileiro de Automação Inteligente*. SBAI - Simpósio Brasileiro de Automação Inteligente, Ouro Preto, MG, Brazil.

Kassimali, A., 2012. *Matrix Analysis of Structures*. Cengage Learning, Carbondale.

Kelly, P.A., 2013. *Solids Mechanics Part I: An Introduction to Solid Mechanics*. S.N., S.L.

Logan, D.L., 2007. *A First Course in the Finite Element Method*. Thomson, Platteville.

Norton, R.L., 2013. *Projeto de Máquinas: Uma Abordagem Integrada*. Bookman, Porto Alegre.

7. RESPONSIBILITY NOTICE

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