XLIII CILAMCE
21-25, November, 2022
FOZ DO IGUAÇU - PR - BRAZIL

Implementation of an elastoplastic finite element code in Python

Leonardo O. Rodriguez[1], Gabriel M. M. dos Santos [1], Silvia Corbani[2]

[1]*Escola Politécnica, Universidade Federal do Rio de Janeiro*
Av. Athos da Silveira Ramos, 149 - Bloco A, 21941-909, Rio de Janeiro, Brasil
leonardo_rodriguez@poli.ufrj.br
[2]*Programa de Projeto de Estruturas, Universidade Federal do Rio de Janeiro*
Av. Athos da Silveira Ramos, 149 - Bloco D - Sala 203, 21941-909, Rio de Janeiro, Brasil
corbani@poli.ufrj.br

Abstract. The use of the finite element method is one of the most efficient approaches to dealing with a problem with non-linearity. It is common that such problems are taken to commercial engineering programs, whose main purpose is to obtain the stresses for the user's decision-making, without a deep focus on the methodology that governs such analyses. In this paper, a finite element code with non-linearity of the material was implemented in Python, which is a language highlighted for being an Open-Source technology. The work aims to facilitate the understanding of a practical application of finite elements with perfect elastoplastic behavior and, thus, to serve as a guide for undergraduate and graduate students interested in the subject. As a practical application, a rectangular steel plate was modeled and evaluated against a vertical load on one of its edges. To this end, the Von Mises yield criterion was used to determine the stresses and, after reaching the yield of the material, the Newton-Raphson iterative method was used to determine the displacements. At the end of the work, there was a good agreement between stresses obtained with the finite element code and the chosen commercial finite element program.

Keywords: finite element method, elastoplasticity, Python

1 Introduction

The finite element method is widely used to solve non-linear analysis problems. In the linear case, its implementation is relatively simple and well documented in the literature. In this case, the relationship between force and displacement is proportional and certain properties of the model, such as modulus of elasticity, remain constant during the analysis. However, in non-linear cases, the implementation requires an iterative process relating force and displacement incrementally, updating the constitutive properties of the material, until the results converge in successive iterations.

Some authors may be highlighted by implementing codes in simple languages or even commercial educational programs of linear-elastic or elastoplastic regimes [1-3]. Tauzowski *et al.* [1] presented a new concept of object orientation in a code implemented in the Matlab program and in the C++ language, focusing on elastoplastic finite element analysis applied to structural topology optimization. Čermák *et al.* [2] proposed a more efficient and flexible implementation during the assembly of tangential stiffness matrices in the analysis of elastoplastic problems with two and three dimensions, developed in the Matlab program. Meanwhile, Cecílio and Santos [3] implemented a finite element code with triangular elements and quadratic interpolation in the commercial program Wolfram Mathematica, to help students and researchers with a better understanding and visualization of plane stress problems in the elastic regime.

Therefore, this article presents a code developed in Python computational language whose didactic objective will be to assist its users in the visualization and resolution of analyzes of steel plates with elastoplastic behavior.

2 Theory of plasticity and failure criteria

There is a linear relationship between force and displacement in linear-elastic behavior. To this end, the definition of ideal spring present in Hooke's Law can be extrapolated, which defines that, when a force F is imposed on a given spring, it will undergo a displacement, x, proportional to its stiffness, k. Therefore, in linear materials, k is directly proportional to its modulus of elasticity, E.

However, in non-linear behavior, this relationship is not directly proportional, as the modulus of elasticity is not a constant property of the material. Thus, a tangent modulus of elasticity is calculated at each point of the stress *vs.* strain diagram of the material to obtain the respective strains and, consequently, the displacements.

In the case of materials such as steel, whose behavior is linear throughout the elastic regime and not linear after the yield point, this strategy is only necessary when it enters the plastic regime, also called the non-linear regime. To do so, a failure criterion is necessary to verify the yield in the plate. In biaxial cases, there are several criteria, such as the Tresca hexagon and Von Mises failure criterion, for ductile materials, and the Mohr-Coulomb criterion, for brittle materials. During the development of the code, the Von Mises failure criterion was implemented, because, for ductile materials, it brings more accurate results than the Tresca criterion, according to Hibbeler [4]. This criterion establishes an expression of an elliptic curve that determines the domain of stresses in a material, as eq. (1):

$$\sigma_1^2 - \sigma_1\sigma_2 + \sigma_2^2 = \sigma_e^2 \,, \tag{1}$$

where σ_e is the yield stress of the material and σ_1 and σ_2 are its two principal stresses.

Since yielding occurs in steel due to the maximum shear stress, the Von Mises yield function, described by Neto *et al* [5] and defined by eq. (2) is used. The yield stress for the pure shear state is given by κ, which is a function of σ_e, according to eq. (3), in addition to J_2, given by eq. (4):

$$f(J_2) = J_2 - \sqrt{\kappa} \,, \tag{2}$$

$$\kappa = \sigma_e \,/\, \sqrt{3} \,, \tag{3}$$

$$J_2 = 1/6\,[(\sigma_X - \sigma_Y)^2 + \sigma_X^2 + \sigma_Y^2] + \tau_{XY}^2 \,, \tag{4}$$

where σ_X, σ_Y e τ_{XY} are the normal and shear stresses acting on the material. Therefore, when eq. (2) has a positive value, the stresses will be in a linear-elastic regime and within the Von Mises domain. Otherwise, the plate will fail, and the stresses will have to be recalculated in such a way that $\mathrm{J}_2 = \sqrt{\kappa}$, in other words, the stresses are in the contour of the ellipse, which will happen during the material yield.

3 Finite element approach to elastoplastic behavior

Elastoplastic behavior is characterized by a non-linear relationship between external force and nodal displacement. Therefore, a strategy used to approach this type of problem is to obtain the internal force in an incremental way. For this, a tangent stiffness matrix, $\mathbf{K}^i$, which relates an increment of internal force and displacement at each step i, is evaluated according to eq. (5). The deformation matrix **B**, which contains the derivatives of the shape functions, responsible for interpolating the coordinates and displacements of the element, and the constitutive matrix of the material, **C**, are used to obtain $\mathbf{K}^i$. This relationship is described in eq. (6).

$$\boldsymbol{\Delta F}^i = \boldsymbol{K}^i \boldsymbol{D}^i \,, \tag{5}$$

$$\boldsymbol{K}^i = \int \boldsymbol{B}^T\, \boldsymbol{C}\, \boldsymbol{B}\; dV \,. \tag{6}$$

In order to obtain the displacements with a lower computational cost, Borst *et al.* [6] propose to use the Newton-Raphson iterative method. For this purpose, force is applied in small increments. The process is illustrated in Fig. 1.

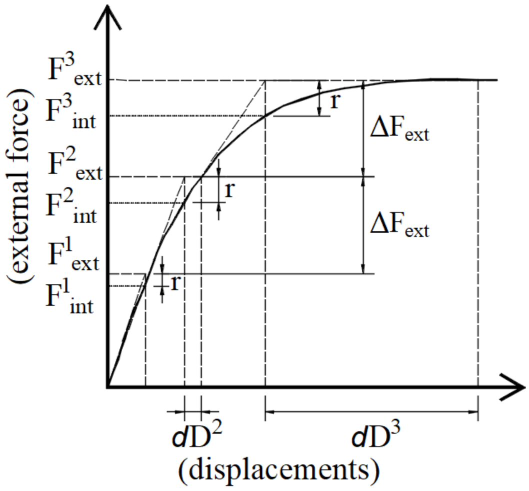


Figure 1. Procedure for iterative-incremental solution.

At each step, the system is incremented by a portion $\Delta\mathbf{F_{ext}}$, ratio of the total external force, $\mathbf{F_{ext}}$, by the number of steps adopted, *n*. For each increment of force, the displacement solution is obtained iteratively until a tolerance pre-established by the user is met. Thus, at each step *i* and sub-step *j*, there will be a force residue, $\mathbf{r}^{i,j}$, given by eq. (7). $\mathbf{F}^i\mathbf{_{ext}}$ and $\mathbf{F}^j\mathbf{_{int}}$ represent, respectively, the external force accumulated until step *i* and the internal force corresponding to all degrees of freedom of the model analyzed in step *i* and sub-step *j*. Thus, when updating the internal force $\mathbf{F}^j\mathbf{_{int}}$, a respective $\mathbf{dD}$ will be obtained at each sub-step. This process is terminated when $\mathbf{r}^{i,j}$ reaches a value smaller than the stipulated tolerance.

$$\boldsymbol{r}^{i,j} = \boldsymbol{F}^i{}_{\boldsymbol{ext}} - \boldsymbol{F}^i{}_{\boldsymbol{int}} \, . \tag{7}$$

The internal force is dependent on the initial values of assigned nodal displacements. Within the element, such force is described according to eq. (8) and is a function of the vector of the principal stress components of this step, $\boldsymbol{\sigma}^{i,j}$. From the point of view of the model as a whole, the global internal force of the problem is obtained through the sum of the internal force found in each element, mapping the location of its nodes, and classifying them as global nodes of the structure.

$$\boldsymbol{F}^{i,j}{}_{\boldsymbol{e,int}} = \int \boldsymbol{B}^T \boldsymbol{\sigma}^{i,j} dV \, . \tag{8}$$

Thus, the displacement corrections at each step are given by eq. (9), where n_i is the number of sub-steps necessary for $\mathbf{F}^{i,j}\mathbf{_{int}}$ to converge, approaching $\mathbf{F}^i\mathbf{_{ext}}$ as much as possible. This occurs because, as the displacements increase at each sub-step *j*, the stresses increase proportionally and, consequently, the internal force tends to increase and approach the value of the external force, which causes the convergence of displacements. These displacements, now properly corrected, are described in eq. (10).

$$\boldsymbol{r}^{i,j} = \boldsymbol{K}^i \sum_{i=1}^{n_i} \boldsymbol{dD}^{i,j} \, , \tag{9}$$

$$\boldsymbol{D}^i = \boldsymbol{D}^{i-1} + \boldsymbol{dD}^j \, . \tag{10}$$

After correction and convergence in the displacement values, we proceed to obtain the stresses. For this purpose, an elastoplastic constitutive matrix is calculated which, unlike that found in the linear regime, is not constant. According to Chen and Han [7], each term in this matrix is expressed as eq. (11):

$$C^{ep}_{ijkl} = C^{el}_{ijkl} - \frac{C^{el}_{ijmn} \frac{\partial f}{\partial \sigma_{mn}} \frac{\partial f}{\partial \sigma_{pq}} C^{el}_{pqkl}}{\frac{\partial f}{\partial \sigma_{rs}} C^{el}_{rstu} \frac{\partial f}{\partial \sigma_{tu}}} \, , \tag{11}$$

where $\mathbf{C^{el}}$ is the matrix of components of the elastic constitutive matrix and $\boldsymbol{\partial f/\partial\sigma}$ is the vector of derivatives of the Von Mises yield function (eq. (4)) with respect to the components of material stress, which in turn are given by eq. (12), where **C** is the elastoplastic constitutive matrix, if the material has yielded, and elastic, otherwise.

$$\boldsymbol{\sigma} = \boldsymbol{C}\,\boldsymbol{B}\,\boldsymbol{D}\,. \tag{12}$$

Once the stresses are obtained, the material yielding must be verified by the Von Mises criterion and, if this occurs, an algorithm is used to correct the stresses in such a way that the yield function, described in eq. (2), provide values within the adopted tolerance. This update is done with a relationship of the invariant of the stress deviator tensor and the mean stress tensor given in eq. (13):

$$\boldsymbol{\sigma}^i = \frac{\kappa}{\kappa_{teste}}\left(\boldsymbol{\sigma}^e - \boldsymbol{\sigma}^e_m\right) + \boldsymbol{\sigma}^e_m = \frac{\kappa}{\sqrt{J_2}}\left(\boldsymbol{\sigma}^e - \boldsymbol{\sigma}^e_m\right) + \boldsymbol{\sigma}^e_m\,, \tag{13}$$

where $\boldsymbol{\sigma}^{\mathbf{e}}_{\mathbf{m}}$ is the mean stress tensor and $(\boldsymbol{\sigma}^{\mathbf{e}} - \boldsymbol{\sigma}^{\mathbf{e}}_{\mathbf{m}})$ denoting the deviation matrix of the stress tensor. This function guarantees that the stresses are on the boundary of the ellipse of the von Mises criterion, which in turn corresponds to the yielding of a ductile material.

4 Numerical simulation

The code was developed from a data input file where the user necessarily defines a rectangular geometry. Therefore, the analyzes involved are limited to simulations of plates with 4 vertices and, consequently, 4 parallel edges. From this geometry, the finite element mesh with quadrilateral elements is automatically generated and the type of interpolation of the problem can be chosen: linear or quadratic. The boundary conditions, including loads and supports, are defined by the user in this same file, as well as the material parameters.

As a numerical example, a plate fixed on one edge was modeled, with a thickness of 1mm, width 48mm and height 44mm, subjected to a vertical load of magnitude 4,620N, applied in 4 increments, on the edge opposite the fixed end. The plate material is SAE 1045 steel, modulus of elasticity 206GPa, yield stress 450MPa and Poisson's ratio 0.3. To generate the mesh, it was decided to adopt 12 elements in the horizontal direction and 11 in the vertical one, totaling 132 elements with 4mm edges, as illustrated in Fig. (2).

The program is then executed and, as post-processing, the discretization of the model is provided, as well as its undeformed and deformed configuration, with proper global number nodes, as shown in Fig. (3.a) and (3.b), respectively. In addition, the same model was developed in the Abaqus finite element program to compare its results with the data outputs of the numerical analysis developed. Such comparisons can be found in Table 1, in which the respective root of the failure criterion J_2 is indicated for each step in both evaluations, as well as the error involved during this analysis.

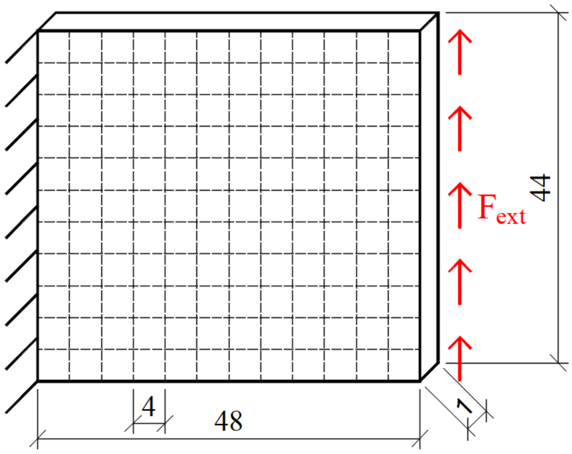
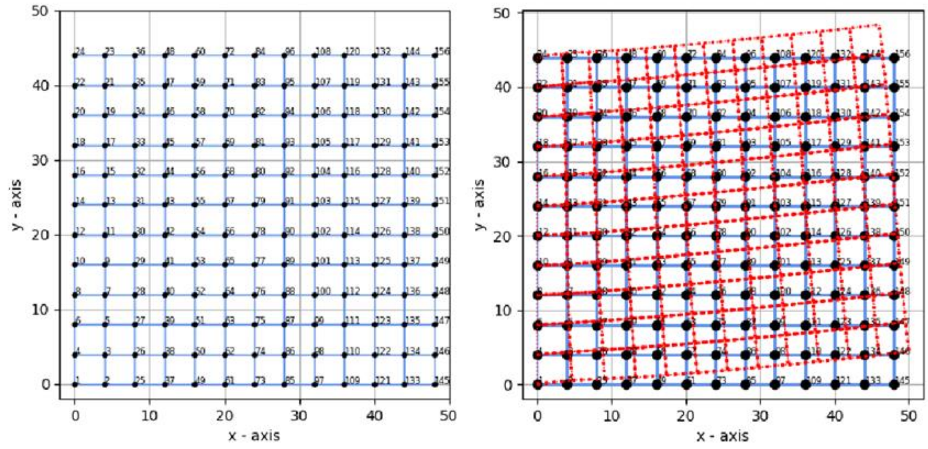


Figure 2. Numerical model under study (units in mm).



(a) (b)

Figure 3. Numerical model: (a) Discretization and (b) Deformed and undeformed configuration.

Table 1. Results comparison

	Step 1	Step 2	Step 3	Step 4
$\sqrt{J_2}$, python [MPa]	114.3	228.7	232.9	252.5
$\sqrt{J_2}$, abaqus [MPa]	114.3	228.7	259.8	259.8
Error	0.0%	0.0%	10.4%	2.8%

5 Conclusions

During the present study, it was sought to develop a didactic finite element code, capable of evaluating elastoplastic models of steel plates with various geometries and loads, in such a way that its user can visualize relevant steps of these stress analyses, such as for example the use of the von Mises criterion and the Newton-Raphson iterative method. In the end, the results of the incremental application of forces were given as a function of J_2, calculated through the acting normal and shear stresses (eq. (4)). The comparison of these values with those provided by the commercial program Abaqus was satisfactory, since there was a maximum error of only 10.4%, based on the parameters of the commercial program. For future work, it is expected some adaptations in the code to allow the adoption of cracked plate models, whose behavior is governed by the elastic-plastic fracture mechanics.

Acknowledgements. I would like to express my special thanks to the institutional program Pibic/UFRJ-CNPq, which supports my research financially.

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] P. Tauzowski, B. Blachowski and J. Lógó "Functor-oriented topology optimization of elasto-plastic structures". Advances in Engineering Software, vol. 135, pp. 102690, 2019.
[2] M. Čermák, S. Sysala and J. Valdman "Efficient and flexible MATLAB implementation of 2D and 3D elastoplastic problems". Applied Mathematics and Computation, vol. 355, n. 6, pp. 595-614, 2019.
[3] D. L. Cecílio and T. D. D. Santos "A finite element elasticity programming in Mathematica software". Computer Applications in Engineering Education, vol. 26, n. 6, pp. 1968-1985, 2018.
[4] R. C. Hibbeler. *Mechanics of materials*. Pearson Education, 2005.
[5] E. A. S. Neto, D. Peric and D. R. J. Owen. *Computational Methods for Plasticity: Theory and Aplications*. John Wiley & Sons, 2008.
[6] R. Borst; M. A. Crisfield, J. J. C. Verhoosel and V. Clemens. *Non-Linear Finite Element Analysis of Solids and Structures.* John Wiley & Sons, 2012
[7] W. F. Chen and D. J. Han. *Plasticity for Structural Engineers.* Springer-Verlag, 1987