

ANÁLISE NÃO LINEAR GEOMÉTRICA DE PÓRTICOS TRIDIMENSIONAIS UTILIZANDO O MÉTODO DOS ELEMENTOS FINITOS.

GEOMETRICALLY NONLINEAR FINITE ELEMENT ANALYSIS OF THREE- DIMENSIONAL FRAMES.

Alysson Barbosa de Barros¹
Felipi Pablo Damasceno Fernandes²
Jordilly Reydson de Barros Silva³

RESUMO

Na engenharia estrutural, a obtenção de soluções exatas é frequentemente inviável devido à complexidade nas relações entre materiais, esforços e deformações. Esse desafio decorre da presença de não linearidades associadas às propriedades dos materiais e aos grandes deslocamentos. Diante disso, métodos numéricos e técnicas aproximativas tornam-se essenciais para a análise e previsão do comportamento estrutural. Neste contexto, este trabalho apresenta o desenvolvimento de um programa computacional em Python, baseado no Método dos Elementos Finitos (MEF), para a análise não linear geométrica de pórticos tridimensionais. A discretização da estrutura em elementos finitos permite a determinação dos esforços internos, deslocamentos, tensões e deformações, considerando a influência de grandes deslocamentos sob a hipótese de pequenas deformações. A visualização gráfica das deformações e distribuições de esforços é integrada ao programa por meio de bibliotecas do Python, como matplotlib e PyVista, facilitando a interpretação dos resultados e sua aplicação no dimensionamento estrutural. A ferramenta foi validada por meio de comparações com *softwares* comerciais amplamente utilizados, como Robot Structural Analysis (RSA), ANSYS e Abaqus, demonstrando resultados coerentes e com baixos desvios em relação às soluções de referência. Em particular, os deslocamentos não lineares apresentaram diferenças máximas de aproximadamente 0,11%. Além disso, o programa desenvolvido inclui um módulo de análise de estabilidade estrutural, que demonstrou convergência para soluções teóricas e apresentou um erro inferior a 5% em estruturas mais complexas. Os resultados confirmam a precisão do programa proposto, contribuindo para o desenvolvimento de soluções computacionais eficientes e acessíveis na análise não linear de estruturas, ampliando a compreensão de comportamentos complexos e auxiliando na previsão de falhas estruturais.

Palavras-chave: método dos elementos finitos; não linearidade geométrica; engenharia estrutural; python.

ABSTRACT

In structural engineering, obtaining exact solutions is often unfeasible due to the complexity of the relationships between materials, forces, and deformations. This challenge arises from the presence of nonlinearities associated with material properties and large displacements. Given this, numerical methods and approximate techniques become essential for analyzing and predicting structural behavior. In this context, this work presents the development of a computational program in Python,

¹ Bacharelado em Engenharia Civil pela Universidade Federal Rural de Pernambuco – Unidade Acadêmica do Cabo de Santo Agostinho, 2025.

² Doutor em Engenharia de Estruturas pela Universidade de São Paulo.

³ Doutor em Engenharia Civil pela Universidade Federal de Pernambuco.

based on the Finite Element Method (FEM), for the geometric nonlinear analysis of three-dimensional frames. The discretization of the structure into finite elements allows for the determination of internal forces, displacements, stresses, and strains, considering the influence of large displacements under the assumption of small strains. The graphical visualization of deformations and stress distributions is integrated into the program using Python libraries such as Matplotlib and PyVista, facilitating the interpretation of results and their application in structural design. The tool was validated through comparisons with widely used commercial software, such as Robot Structural Analysis (RSA), ANSYS, and Abaqus, demonstrating consistent results with low deviations from reference solutions. In particular, the nonlinear displacements exhibited maximum differences of approximately 0.11%. Additionally, the developed program includes a structural stability analysis module, which demonstrated convergence to theoretical solutions and presented an error of less than 5% in more complex structures. The results confirm the accuracy of the proposed program, contributing to the development of efficient and accessible computational solutions for nonlinear structural analysis, enhancing the understanding of complex behaviors, and aiding in the prediction of structural failures.

Keywords: finite element method; geometric nonlinearity; structural engineering; python.

INTRODUÇÃO

Em muitos campos da ciência e da engenharia, certos problemas podem ser resolvidos de forma analítica por meio de relações matemáticas lineares. No entanto, na engenharia estrutural, a complexidade das interações entre materiais, esforços e deformações frequentemente impede a obtenção de soluções exatas. Isso ocorre porque as estruturas reais estão sujeitas a comportamentos não lineares, seja devido às propriedades dos materiais e/ou grandes deslocamentos. Dessa forma, torna-se necessário recorrer a métodos numéricos e aproximativos para a obtenção de soluções aplicáveis à prática.

Tais métodos tiveram seus princípios lançados no século XVIII, quando Gauss propôs o uso de funções de aproximação para resolver problemas matemáticos complexos (Lotti *et al.*, 2006). Durante o século XVIII e XIX, o progresso foi limitado devido à ausência de tecnologias computacionais adequadas para processar as equações algébricas resultantes dos métodos de aproximação. Isso mudou na década de 1950 com o advento da computação, possibilitando a resolução de problemas complexos e o surgimento do Método dos Elementos Finitos (MEF) como é atualmente conhecido. O desenvolvimento moderno do MEF ocorreu ao longo do século XX, principalmente com os trabalhos de Courant, Turner e Clough, que

formalizaram a metodologia para aplicações em engenharia (Vaz, 2011).

O MEF é, essencialmente, uma técnica numérica que envolve a discretização (ou subdivisão) de um meio contínuo em elementos finitos, onde cada elemento mantém as propriedades do material e obedece às mesmas equações diferenciais (Lotti *et al.*, 2006).

Na engenharia estrutural, essas equações permitem determinar as tensões, deformações e outros parâmetros relevantes, como deslocamentos e esforços internos. A formulação do MEF pode ser derivada a partir de diferentes métodos, como o Princípio da Mínima Energia Potencial, o Método dos Resíduos Ponderados ou o Princípio dos Trabalhos Virtuais (Vaz, 2011).

Neste artigo, a estrutura é tratada como um meio contínuo e discretizada em elementos de pórtico (*frame elements*), que foram o foco da análise. No entanto, o MEF engloba também outros tipos de elementos, tais como: barra, casca, placa, membrana e sólido, cada um adaptado a diferentes necessidades de modelagem estrutural, aumentando a abrangência das aplicações do método.

A análise estrutural pode ser dividida em linear e não linear, dependendo do comportamento da estrutura sob carregamento. Na análise linear elástica, assume-se que o material não atingiu as tensões de escoamento e suas propriedades permanecem constantes, de modo que as relações entre forças, deslocamentos e deformações são

proporcionais, dada que as deformações são pequenas. Assim, as equações de equilíbrio são obtidas a partir da geometria indeformada da estrutura (Foletto *et al.*, 2019).

A análise não linear é subdividida em geométrica e física. A não linearidade geométrica (NLG) ainda pode tratar o material como elástico, contudo, o equilíbrio das forças é obtido considerando-se o estado atual da estrutura, isto é, na sua configuração deformada, de modo que a compatibilidade entre deslocamentos e deformações não é de natureza linear.

Já a não linearidade física (NLF) refere-se ao comportamento constitutivo dos materiais, em que a relação tensão-deformação não é mais linear. Esse tipo de não linearidade abrange materiais com diferentes respostas mecânicas, permitindo a distinção entre estruturas frágeis e dúcteis, dependendo da capacidade de redistribuir tensões antes da ruptura.

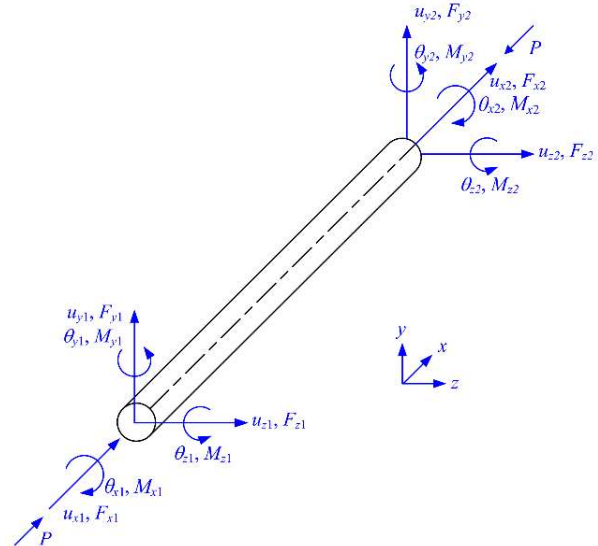
O objetivo deste trabalho é desenvolver um programa computacional em Python, utilizando o MEF para determinar os esforços internos (axiais, transversais e momentos) e os respectivos deslocamentos de estruturas tridimensionais, levando em conta a não linearidade geométrica, quando aplicável. Os resultados obtidos foram comparados com os de *softwares* comerciais amplamente utilizados na engenharia estrutural, como o Robot Structural Analysis, ANSYS e Abaqus, e as estruturas foram visualizadas graficamente no próprio programa por meio do PyVista, facilitando a interpretação dos dados e auxiliando em cálculos subsequentes, como o dimensionamento de armaduras longitudinais, transversais e de torção em elementos de concreto armado.

REVISÃO BIBLIOGRÁFICA

Os elementos são definidos em função dos nós que os conectam, devendo-se especificar os nós inicial e final. Dessa forma, todos os elementos são, de maneira simplificada, elementos de barra (*bar elements*) com seis graus de liberdade em cada nó, os quais descrevem os graus de liberdade que o elemento tem ao longo dos eixos.

Tais graus de liberdade são referentes à translação ao longo dos eixos x, y e z; e à rotação em torno dos eixos x, y e z. Os eixos e respectivos esforços são expressos na Figura 1.

Figura 1 - Deslocamentos nodais de um elemento de barra tridimensional.



Fonte: Adaptado de Pan *et al.*, 2022.

Princípio do Trabalho Virtual

Para o propósito deste artigo, o Princípio do Trabalho Virtual (PTV) foi utilizado, visto que é facilmente aplicado e válido tanto para estruturas simples como para estruturas complexas envolvendo a NLG. O PTV afirma que o trabalho virtual realizado por forças externas é igual ao trabalho virtual realizado pelas forças internas (Vasilescu, 2000), assim como mostra a Eq. (1).

$$\delta W_e = \delta W_i \quad (1)$$

Trabalhos virtuais são denotados pelo símbolo δ , com o subscrito indicando sua origem. Assim, δW_i é o trabalho virtual interno; e δW_e o trabalho virtual externo, que consiste nos trabalhos virtuais exercidos pelo carregamento externo na estrutura, como mostra a Eq. (2).

$$\delta W_e = \int_V \{\tilde{d}\}^T \{b\} dV + \int_A \{\tilde{d}\}^T \{q\} dA + \{\tilde{d}\}_i^T \{P\}_i \quad (2)$$

Onde $\{\tilde{d}\}$ é o vetor de deslocamentos virtuais, impostos no meio contínuo (pórtico formado por vários elementos de pórtico), satisfazendo as condições de contorno; $\{b\}$ o vetor de forças de corpo, sendo, nesse caso, o peso próprio; $\{q\}$ o vetor de forças distribuídas; e $\{P\}$ o vetor de forças concentradas. Já δW_i é dado pela Eq. (3).

$$\delta W_i = \int_V \{\tilde{\varepsilon}\}^T \{\sigma\} dV \quad (3)$$

Onde $\{\sigma\}$ é o vetor das tensões internas, e:

$$\{\tilde{\varepsilon}\} = [L]\{\tilde{d}\} \quad (4)$$

Sendo $\{\tilde{\varepsilon}\}$ o vetor das deformações relacionadas aos deslocamentos virtuais em função do gradiente de deslocamento, $[L]$.

A fim de discretizar o elemento de pórtico, deve-se aproximar o vetor de

$$\sum_{e=1}^{n_e} \int_{V_e} ([L][N]\{\tilde{d}\})^T \{\sigma\} dV_e = \sum_{e=1}^{n_e} \int_{V_e} \{\tilde{d}\}^T [N]^T \{b\} dV_e + \sum_{e=1}^{n_e} \int_{A_e} \{\tilde{d}\}^T [N]^T \{q\} dA_e + \sum_{e=1}^{n_e} \{\tilde{d}\}^T \{P_i\} \quad (6)$$

$$\sum_{e=1}^{n_e} \int_{V_e} [B]^T \{\sigma\} dV_e = \sum_{e=1}^{n_e} \int_{V_e} [N]^T \{b\} dV_e + \sum_{e=1}^{n_e} \int_{A_e} [N]^T \{q\} dA_e + \sum_{e=1}^{n_e} \{P_i\} \quad (7)$$

A matriz $[B]$, conhecida como matriz deformação-deslocamento, estabelece a relação entre as deformações, ε , em um elemento e os deslocamentos nodais, d .

Em estruturas onde se aplica a teoria das pequenas deformações, ou seja, para comportamento estrutural linear, a matriz $[B]$ é composta pelas derivadas de primeira ordem das funções de forma $[N]$ e se mantém constante ao longo da análise. Nesse caso, $[B]$ independe dos deslocamentos nodais, o que simplifica o cálculo e permite que as equações de equilíbrio sejam estabelecidas na configuração inicial, não deformada, da estrutura.

Por outro lado, em análises não lineares geométricas, onde os deslocamentos são relevantes para o equilíbrio, a matriz $[B]$ não permanece constante. Nesse cenário, $[B]$ passa a depender dos deslocamentos nodais, e as equações de equilíbrio precisam considerar os

deslocamentos do elemento através de um conjunto de funções de interpolação, chamadas de funções de forma, $[N]$, que descrevem os deslocamentos do elemento finito, levando-se em consideração as condições de compatibilidade, dada por $\{d\}$ (Vasilescu, 2000). O vetor do campo de deslocamentos virtuais é dado, então, por:

$$\{\tilde{d}\} = [N]\{d\} \quad (5)$$

Onde $\{d\}$ refere-se aos deslocamentos nos nós do elemento.

Por se tratar do somatório das influências de todos os elementos finitos para, então, obter o comportamento global da estrutura, e substituindo as equações (3), (4) e (5) na Eq. (2), tem-se o resultado expresso na Eq. (6). Essa expressão deve ser válida para cada deslocamento nodal virtual, $\{\tilde{d}\}$, e fazendo $[B] = [L][N]$, obtém-se a Eq. (7).

deslocamentos da estrutura. Isso introduz maior complexidade ao problema, exigindo técnicas iterativas e modelos mais sofisticados para se obter uma solução precisa (McGuire; Gallagher; Ziemian, 2000; Vasilescu, 2000).

Já o vetor de tensões internas pode ser definido segundo a Lei de Hooke, tal que:

$$\{\sigma\} = [D]\{\varepsilon\} \quad (8)$$

Onde $[D]$ é a matriz constitutiva do material; e $\{\varepsilon\}$ é o vetor de deformações reais da estrutura, que pode ser dado de maneira similar à deformação virtual apresentada na Eq. (4):

$$\{\varepsilon\} = [L][N]\{d\} = [B]\{d\} \quad (9)$$

Onde $\{d\}$ corresponde ao vetor de deslocamentos reais.

Substituindo a Eq. (8) na (7), obtém-se:

$$\begin{aligned} & \sum_{e=1}^{n_e} \int_{V_e} [B]^T [D] [B] dV_e \{d\} \\ &= \sum_{e=1}^{n_e} \left(\int_{V_e} [N]^T \{b\} dV_e \right. \\ & \left. + \int_{A_e} [N]^T \{q\} dA_e + \{P_i\} \right) \end{aligned} \quad (10)$$

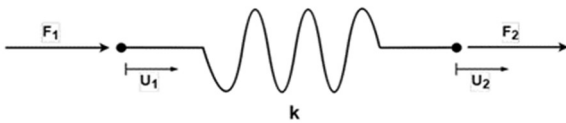
Ou:

$$[K]\{d\} = \{F\} \quad (11)$$

Onde $[K]$ corresponde à matriz de rigidez estrutural, e $\{F\}$ ao vetor das cargas nodais equivalentes.

A Eq. (11) pode ser obtida, de maneira simplificada, ao representar a estrutura elástica como um sistema de mola de rigidez k , como mostra a Figura 2.

Figura 2 - Sistema de molas para um elemento elástico.



Fonte: Parreiras, 2019.

$$\{d\} = \{u_1 \quad v_1 \quad w_1 \quad \theta_{x1} \quad \theta_{y1} \quad \theta_{z1} \quad u_2 \quad v_2 \quad w_2 \quad \theta_{x2} \quad \theta_{y2} \quad \theta_{z2}\}^T \quad (13)$$

$$\{F\} = \{F_{x1} \quad F_{y1} \quad F_{z1} \quad M_{x1} \quad M_{y1} \quad M_{z1} \quad F_{x2} \quad F_{y2} \quad F_{z2} \quad M_{x2} \quad M_{y2} \quad M_{z2}\}^T \quad (14)$$

Campos de deslocamentos

As funções de forma, $[N]$, conforme mencionado, descrevem o deslocamento do elemento finito em função dos deslocamentos nodais. Inicialmente, a análise será conduzida para um elemento de treliça sujeito a deslocamento axial. Em seguida, será considerado um elemento de viga submetido à flexão no plano x-y, para, por fim, estender a formulação ao plano x-z de maneira análoga.

Os deslocamentos do elemento deformado estão representados na Figura 3. Para deslocamentos axiais, assume-se um campo linear da forma:

$$u_a(x) = a_0 + a_1x$$

Chegando, dessa maneira, à Eq. (11) apresentada anteriormente.

Isolando os deslocamentos nodais nessa equação, chega-se à Eq. (12):

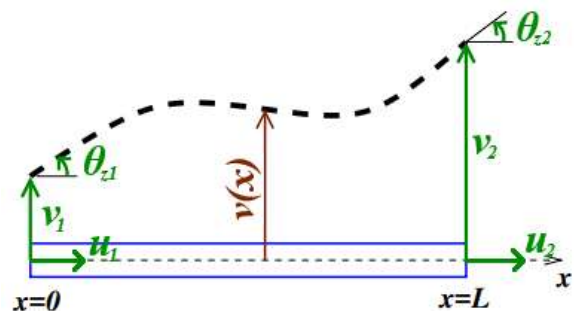
$$\{d\} = [K]^{-1}\{F\} \quad (12)$$

Como o elemento considerado possui 6 graus de liberdade em cada nó, a matriz de rigidez estrutural resultante apresenta dimensão 12×12 , o que reflete as interações entre os dois nós, cada um com seus 6 graus de liberdade. Essa matriz inclui os efeitos dos deslocamentos translacionais e das rotações em torno de cada eixo.

Além disso, tanto o vetor de forças nodais, $\{F\}$, quanto o vetor de deslocamentos nodais, $\{d\}$, têm dimensão 12×1 . Dessa forma, cada componente do vetor de forças representa uma carga aplicada em uma direção ou um momento aplicado em torno de um eixo, enquanto os componentes do vetor de deslocamentos indicam as translações e rotações dos nós. Tais vetores são dados nas Eq. (13) e (14).

Essa função deve concordar com os deslocamentos nodais u_1 e u_2 , variando de $x = 0$ a $x = L$, onde L é o comprimento do elemento.

Figura 3 - Deslocamentos de uma viga submetida à flexão.



Fonte: Adaptado de Gavin, 2012.

Realizando o devido desenvolvimento algébrico da função, chega-se ao campo de deslocamento axial, N_u , tal que:

$$u(x) = \left(1 - \frac{x}{L}\right)u_1 + \frac{x}{L}u_2$$

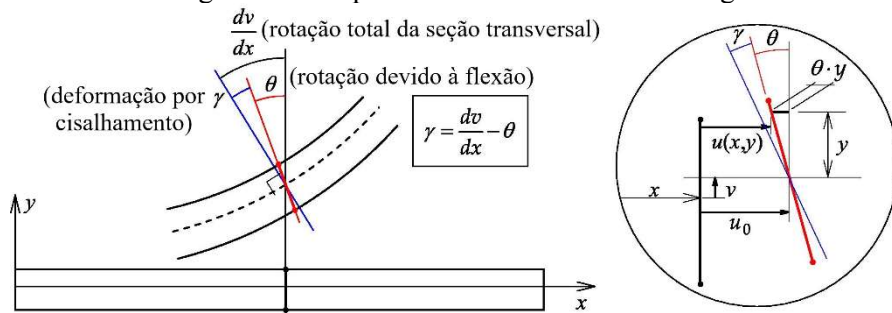
$$u(x) = N_{u1}u_1 + N_{u2}u_2 \quad (15)$$

A análise da flexão segue as hipóteses da teoria de Euler-Bernoulli, na qual a deflexão é modelada por meio de equações diferenciais parciais lineares. Dessa forma, considera-se que:

- Os planos perpendiculares à linha neutra permanecem planos e perpendiculares após a deformação;
- Pequenas rotações, permitindo o uso das aproximações $\sin(\theta) \cong \theta$ e $\cos(\theta) \cong 1$; além disso, admite-se que a rotação nodal é igual à derivada do campo de deslocamentos no ponto, ou seja, $\theta(x) \cong v'(x)$;

A Figura 4 representa o campo de deslocamentos de uma viga submetida à flexão. A parcela γ corresponde à deformação por cisalhamento, a qual é desprezada.

Figura 4 - Campo de deslocamentos de uma viga.



Fonte: Adaptado de Rodrigues; Burgos; Martha, 2021.

Realizando o mesmo procedimento para os deslocamentos transversais ao longo do plano x-y, tem-se a função cúbica dada pela Eq. (16). A curvatura do elemento é dada pela sua derivada, $v'(x, y)$, como mostra a Eq. (17).

$$v(x) = a_0 + a_1x + a_2x^2 + a_3x^3 \quad (16)$$

$$v'(x) = a_1 + 2a_2x + 3a_3x^2 \quad (17)$$

Aplicando as condições de contorno da Figura 3 e adotando um domínio unitário, $\xi = x/L$, obtêm-se as funções de forma para o deslocamento transversal (Eq. (18)) e rotação (Eq. (19)).

$$v(\xi) = N_{v1}v_1 + N_{v2}\theta_{z1} + N_{v3}v_2 + N_{v4}\theta_{z2} \quad (18)$$

$$\theta_z(\xi) = N_{\theta z1}v_1 + N_{\theta z2}\theta_{z1} + N_{\theta z3}v_2 + N_{\theta z4}\theta_{z2} \quad (19)$$

Os termos dos campos de deslocamento são definidos da seguinte maneira:

$$N_{v1} = 1 - 3\xi^2 + 2\xi^3$$

$$N_{v2} = L(\xi - 2\xi^2 + \xi^3)$$

$$N_{v3} = 3\xi^2 - 2\xi^3$$

$$N_{v4} = L(-\xi^2 + \xi^3)$$

$$N_{\theta z1} = \frac{6}{L}(-\xi + \xi^2)$$

$$N_{\theta z2} = -1 + 4\xi - 3\xi^2$$

$$N_{\theta z3} = -\frac{6}{L}(-\xi + \xi^2)$$

$$N_{\theta z4} = 2\xi - 3\xi^2$$

Já no plano x-z, os cálculos são análogos, com alterações na orientação dos termos associados à rotação. As expressões para a translação e rotação nesse plano são, portanto, dadas por:

$$w(\xi) = N_{w1}w_1 + N_{w2}\theta_{y1} + N_{w3}w_2 + N_{w4}\theta_{y2} \quad (20)$$

$$\theta_y(\xi) = \mathcal{N}_{\theta y1} w_1 + \mathcal{N}_{\theta z1} \theta_{y1} + \mathcal{N}_{\theta z1} w_2 + \mathcal{N}_{\theta z1} \theta_{y2} \quad (21)$$

Sendo:

$$\begin{bmatrix} \mathcal{N}_{w1} \\ \mathcal{N}_{w2} \\ \mathcal{N}_{w3} \\ \mathcal{N}_{w4} \end{bmatrix} = \begin{bmatrix} \mathcal{N}_{v1} \\ -\mathcal{N}_{v2} \\ \mathcal{N}_{v3} \\ -\mathcal{N}_{v4} \end{bmatrix}$$

$$\begin{bmatrix} \mathcal{N}_{\theta y1} \\ \mathcal{N}_{\theta y2} \\ \mathcal{N}_{\theta y3} \\ \mathcal{N}_{\theta y4} \end{bmatrix} = \begin{bmatrix} \mathcal{N}_{\theta z1} \\ -\mathcal{N}_{\theta z2} \\ \mathcal{N}_{\theta z3} \\ -\mathcal{N}_{\theta z4} \end{bmatrix}$$

De tal modo que, matricialmente, a relação cinemática entre os nós é dada por:

$$\{u\}_{6 \times 1} = [N]_{6 \times 12} \{d\}_{12 \times 1} \quad (22)$$

Não linearidade geométrica

Em problemas não lineares, como aqueles com grandes deslocamentos, a equação de equilíbrio mostrada na Eq. (11) não é satisfeita de imediato. Por isso, é necessário utilizar métodos incrementais e iterativos, como o Método de Newton-Raphson, para

$$\frac{\partial \{R(d)\}}{\partial \{d\}} = \frac{\partial}{\partial \{d\}} (\{F_{ext}\} - \{F_{int}(d)\}) = -\frac{\partial F_{int}(d)}{\partial \{d\}} = -\frac{\partial}{\partial \{d\}} \left(\sum_{e=1}^{n_e} \int_{V_e} [B]_e^T \{\sigma\} dV_e \right)$$

$$\frac{\partial \{R(d)\}}{\partial \{d\}} = -\sum_{e=1}^{n_e} \left(\int_{V_e} \left(\frac{\partial [B]_e^T}{\partial \{d\}} \{\sigma\} + [B]_e^T \frac{\partial \{\sigma\}}{\partial \{d\}} \right) dV_e \right) \quad (25)$$

Como mencionado anteriormente, a matriz $[B]$ é composta por duas parcelas, uma contribuição constante, $[B_0]$, e uma parcela variável, $[B_L]$, tal qual mostra a Eq. (26).

$$[B] = [B_0] + [B_L] \quad (26)$$

O incremento na matriz deformação-deslocamento, $\partial[B]/\partial\{d\}$, depende, portanto, apenas da componente $[B_L]$, resultando em:

$$\frac{\partial [B]}{\partial \{d\}} = \frac{\partial}{\partial \{d\}} ([B_0] + [B_L]) = \frac{\partial}{\partial \{d\}} [B_L]$$

resolver o problema (McGuire; Gallagher; Ziemian, 2000; Vasilescu, 2000).

Para assegurar o equilíbrio estrutural, define-se o vetor de forças residuais $\{R\}$. As forças internas em cada elemento são determinadas conforme a Eq. (23).

$$\{F_{int}\} = \frac{\partial \delta W_i}{\partial \{d\}} = \int_{V_e} [B]^T \{\sigma\} dV_e \quad (23)$$

Já o vetor $\{R\}$, dependente dos deslocamentos nodais do sistema, é calculado como sendo a diferença das forças externas e internas, como expressa a Eq. (24).

$$\{R(d)\} = \{F_{ext}\} - \{F_{int}(d)\} \quad (24)$$

Aproximando o vetor $\{R(d)\}$ através de uma série de Taylor de 1ª ordem, para um incremento Δd , tem-se:

$$\{R(d + \Delta d)\} = \{R(d)\} + \frac{\partial \{R(d)\}}{\partial \{d\}} \{\Delta d\}$$

Como apenas as forças internas dependem dos deslocamentos nodais, a equação acima é reescrita conforme a Eq. (25).

A formulação não linear exige a consideração das deformações de segunda ordem, representadas pelas deformações de Green-Lagrange, as quais permitem relacionar deslocamentos e deformações de forma mais abrangente (Rodrigues; Burgos; Martha, 2021). Essa relação é analisada através do campo de deslocamentos, ilustrado na Figura 4 para o plano x-y e descrito pelas equações:

$$u(x, y) = u_0(x) - y \frac{\partial v(x)}{\partial x} \quad (27)$$

$$v(x, y) = v_0(x) \quad (28)$$

Analogamente, para o plano x-z:

$$u(x, z) = u_0(x) + z \frac{\partial w(x)}{\partial x} \quad (29)$$

$$w(x, z) = w_0(x) \quad (30)$$

Sejam ε_{xx} e η_{xx} as parcelas linear e não linear das deformações de Green-Lagrange ao longo do eixo x, tem-se:

$$\varepsilon_{xx} = \frac{\partial u}{\partial x} = \frac{\partial u_0}{\partial x} - y \frac{\partial^2 v}{\partial x^2} + z \frac{\partial^2 w}{\partial x^2} \quad (31)$$

$$\eta_{xx} = \frac{1}{2} \left(\left(\frac{\partial u}{\partial x} \right)^2 + \left(\frac{\partial v}{\partial x} \right)^2 + \left(\frac{\partial w}{\partial x} \right)^2 \right) \quad (32)$$

As demais deformações normais ao eixo do pórtico, ε_{yy} e ε_{zz} , são desconsideradas, e as componentes transversais de cisalhamento, γ_{xy} e γ_{xz} , são pequenas e podem ser desprezadas. A deformação do elemento é, então, reescrita como:

$$\begin{aligned} \{\varepsilon\} &= \{\varepsilon_{xx}\} + \{\eta_{xx}\} \\ \rightarrow \{\varepsilon\} &= \left([B_0] + \frac{1}{2} [B_L] \right) \{d\} \end{aligned} \quad (33)$$

Reescrevendo a Eq. (32), tem-se:

$$\eta_{xx} = \frac{1}{2} \begin{Bmatrix} \frac{\partial u}{\partial x} \\ \frac{\partial v}{\partial x} \\ \frac{\partial w}{\partial x} \end{Bmatrix} \begin{Bmatrix} \frac{\partial u}{\partial x} & \frac{\partial v}{\partial x} & \frac{\partial w}{\partial x} \end{Bmatrix} = \frac{1}{2} \{A\} \{\theta\}$$

Onde $\{\theta\}^T = \{A\}$.

Na sequência, pela regra da cadeia, a derivada acima em função de ξ é dada por:

$$\begin{Bmatrix} \frac{\partial u}{\partial x} \\ \frac{\partial v}{\partial x} \\ \frac{\partial w}{\partial x} \end{Bmatrix} = \frac{\partial \xi}{\partial x} \begin{Bmatrix} \frac{\partial u}{\partial \xi} \\ \frac{\partial v}{\partial \xi} \\ \frac{\partial w}{\partial \xi} \end{Bmatrix} = \frac{\partial \xi}{\partial x} \begin{Bmatrix} \frac{\partial N_u}{\partial \xi} \\ \frac{\partial N_v}{\partial \xi} \\ \frac{\partial N_w}{\partial \xi} \end{Bmatrix} \{d\}$$

$$\{A\} = [G] \{d\} \quad (34)$$

Comparando as equações (33) e (34), a matriz $[B_L]$ pode ser reescrita como:

$$[B_L] = [G] \{\theta\} \quad (35)$$

Sendo $[G]$ a matriz de gradientes do vetor campo de deslocamentos e $\{d\}$ o vetor de deslocamentos nodais do elemento.

Analisando separadamente o primeiro integrando da Eq. (25), com base nas equações (8), (9), (26) e (35), tem-se que:

$$\begin{aligned} \frac{\partial [B]_e^T}{\partial \{d\}} \{\sigma\} &= \frac{\partial [B_L]_e^T}{\partial \{d\}} \{\sigma\} \\ \rightarrow \frac{\partial (\{\theta\} [G])^T}{\partial \{d\}} \{\sigma\} &= \frac{\partial (\{\theta\}^T [G]^T)}{\partial \{d\}} \{\sigma\} \\ \rightarrow [G]^T \frac{\partial \{A\}}{\partial \{d\}} \{\sigma\} &= [G]^T \frac{\partial \{[G] \{d\}\}}{\partial \{d\}} \{\sigma\} \end{aligned}$$

$$\frac{\partial [B]_e^T}{\partial \{d\}} \{\sigma\} = [G]^T \{\sigma\} [G] \quad (36)$$

Já para o segundo integrando:

$$\begin{aligned} [B]_e^T \frac{\partial \{\sigma\}}{\partial \{d\}} &= [B]^T \frac{\partial ([D] [B] \{d\})}{\partial \{d\}} \\ [B]_e^T \frac{\partial \{\sigma\}}{\partial \{d\}} &= [B]^T [D] [B] \end{aligned} \quad (37)$$

Substituindo ambos os termos na Eq. (25), pode-se reescrever o incremento de forças internas, dado por:

$$\begin{aligned} \frac{\partial \{R(d)\}}{\partial \{d\}} &= - \sum_{e=1}^{n_e} \left(\int_{V_e} ([G]^T \{\sigma\} [G] \right. \\ &\quad \left. + [B]^T [D] [B]) dV_e \right) \\ \frac{\partial \{R(d)\}}{\partial \{d\}} &= - \sum_{e=1}^{n_e} (K_\sigma + K_e) \end{aligned} \quad (38)$$

Onde K_σ é a matriz de rigidez geométrica, ou matriz de tensão inicial, e K_e a matriz de rigidez elástica, sendo dadas por:

$$K_e = \int_{V_e} [B]^T [D] [B] dV_e \quad (39)$$

$$K_\sigma = \int_{V_e} [G]^T \{\sigma\} [G] dV_e \quad (40)$$

E a soma de ambas as parcelas é chamada de matriz de rigidez tangente, K_T :

$$K_T = K_e + K_\sigma \quad (41)$$

Integração numérica

Devido à complexidade dos termos que compõem as matrizes de rigidez, a resolução por métodos analíticos pode ser ineficiente do ponto de vista computacional. Desse modo, foi adotada a integração Gaussiana para a resolução de tais integrais.

De maneira simplificada, a quadratura de Gauss-Legendre é um método numérico para a aproximação de integrais definido com base na escolha otimizada de pontos de integração (nós) e pesos associados (Suescún-Díaz; Chala-Casanova; Gómez-Noguera, 2022).

O método consiste em aproximar uma função através de uma soma de n pontos de integração, definidos em um novo domínio $[-1, 1]$, tal que:

$$\int_{-1}^1 f(t) dt \approx [J] \sum_{i=1}^n w_i f(t_i) \quad (42)$$

Onde w_i e t_i são o i -ésimo peso e ponto de Gauss, respectivamente; e $[J]$ é o Jacobiano do novo domínio de integração.

Sendo tal domínio definido como:

$$x(t) = \frac{b-a}{2} t + \frac{a+b}{2}$$

E sua respectiva derivada:

$$\frac{dx}{dt} = \frac{b-a}{2} \rightarrow dx = \frac{b-a}{2} dt = [J] dt$$

No caso do MEF, as integrais a serem modificadas são as expressas pelas Eq. (39) e (40). Por se tratar de elementos de pórtico, onde a dimensão de interesse é unicamente o comprimento do elemento (eixo x local), as integrais volumétricas apresentadas podem ser simplificadas para uma única dimensão. Isto posto, para uma matriz volumétrica, K :

$$K = \int_{V_e} dV_e = \int_0^L \left(\int_A dA \right) dx$$

Transformação para o domínio padrão:

$$x(\xi) = \frac{L}{2} (\xi + 1) \rightarrow dx = \frac{L}{2} d\xi$$

Assim, a integral no novo domínio é dada por:

$$K = \int_{-1}^1 \left(\int_A dA \right) \frac{L}{2} d\xi \quad (43)$$

Ressalta-se que, como a matriz do campo de deslocamento, $[N]$, e a matriz deformação-deslocamento, $[B]$, dependem diretamente da posição x ao longo do elemento, seus termos devem ser avaliados nos pontos de integração da quadratura de Gauss.

Métodos iterativos-incrementais

Como já comentado, a fim de prever a resposta estrutural não linear, é utilizado um procedimento incremental-iterativo, sendo o método de Newton-Raphson adotado neste trabalho devido à sua eficiência computacional e acurácia.

A formulação do método admite algumas abordagens distintas em relação ao caráter iterativo, como: Newton-Raphson Completo e Modificado, e Quase-Newton. Além disso, o processo incremental pode ser conduzido por meio do controle de deslocamentos ou de cargas (Vasilescu, 2000).

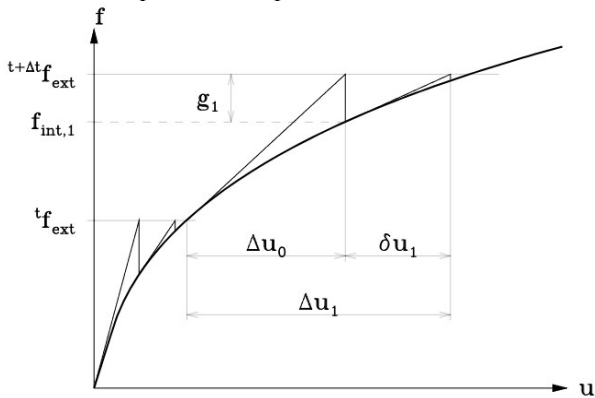
Independentemente da formulação adotada, todos os métodos visam determinar o incremento de deslocamento em cada passo da análise, $\{\Delta d\}^i$, até que o critério de convergência estabelecido seja satisfeito.

Na abordagem de Newton-Raphson Completo (NRC), a matriz de rigidez tangente, $[K_T]$, é recalculada a cada iteração e utilizada na obtenção do próximo vetor de incremento de deslocamento. Essa abordagem tende a proporcionar uma convergência mais rápida, porém com um alto custo computacional.

No método de Newton-Raphson Modificado (NRM), $[K_T]$ é recalculada apenas no início de cada incremento (seja de carga ou deslocamento). Assim, esse método precisa de mais iterações até atingir a convergência, contudo, com menor esforço computacional.

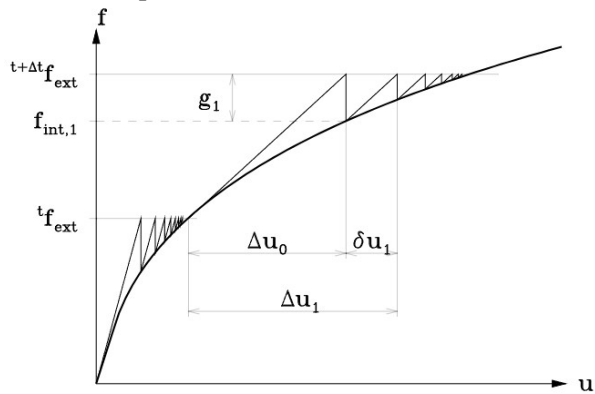
Já o método Secante (ou Quase-Newton) busca um equilíbrio entre eficiência e custo computacional. Nesse caso, a matriz de rigidez $[K]^{i+1}$ é atualizada com base na posição anterior da curva de equilíbrio. Por exemplo, cita-se o método conhecido como BFGS (Broyden-Fletcher-Goldfarb-Shanno) (Nonlinear..., 2025). Os métodos podem ser visualizados nas figuras a seguir.

Figura 5 - Representação gráfica do Método de Newton-Raphson Completo.



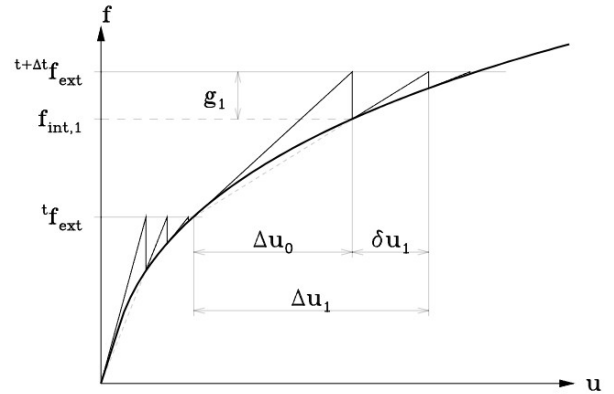
Fonte: Nonlinear..., 2025.

Figura 6 - Representação gráfica do Método de Newton-Raphson Modificado.



Fonte: Nonlinear..., 2025.

Figura 7 - Representação gráfica do Método Quase-Newton.



Fonte: Nonlinear..., 2025.

Quanto aos métodos incrementais, há a subdivisão do vetor de forças externas ou deslocamentos nodais em incrementos menores até atingir o seu valor total. Essa subdivisão depende do grau de complexidade da estrutura.

A precisão da resposta estrutural obtida por meio dos métodos mencionados está diretamente relacionada aos critérios de convergência adotados, os quais podem considerar deslocamentos, forças, energia interna ou uma combinação desses parâmetros (Vasilescu, 2000). Uma tolerância de convergência excessivamente alta pode interromper as iterações prematuramente, resultando em uma resposta estrutural imprecisa. Por outro lado, tolerâncias muito baixas garantem maior confiabilidade na solução, mas aumentam significativamente o custo computacional.

A convergência dos deslocamentos, forças e energia interna é definida a seguir:

$$\frac{\|\{\Delta d\}^i\|}{\|\{d\}^i\|} \leq d_{tol} \quad (44)$$

$$\frac{\|\{R\}^i\|}{\|\{\Delta F\}^i\|} \leq F_{tol} \quad (45)$$

$$\frac{\|\Delta \Pi_R^i\|}{\|\Delta \Pi_F^i\|} \leq \Pi_{tol} \quad (46)$$

Onde $\|\cdot\|$ representa a norma euclidiana dos vetores correspondentes. Os numeradores referem-se aos vetores de incremento na i -ésima iteração, enquanto os denominadores

referem-se aos valores acumulados de cada variável ao longo do processo iterativo. Os parâmetros com subscrito “tol” correspondem aos limites de tolerância estabelecidos para cada critério de convergência

Matriz de transformação

Todos os dados apresentados até o momento referem-se às coordenadas globais do elemento, ou seja, alinhadas paralelamente ao eixo x global. No entanto, na maioria dos casos, essa disposição não ocorre, visto que as estruturas podem ter orientações diversas, como verticais (pilares) ou inclinadas.

Assim, para representar o elemento corretamente, necessita-se de uma matriz de rotação, ou matriz de transformação, [T]. A obtenção da matriz de rotação envolve a definição de uma nova base de vetores ortogonais que representem a orientação de um elemento no espaço, a qual converte o elemento das coordenadas globais para locais, ou vice-versa.

Nota-se que as rotações podem acontecer nos eixos x, y e/ou z, as quais compõem a matriz de transformação. Para determiná-la, é necessário estabelecer três vetores mutuamente ortogonais que definam o sistema de coordenadas local do elemento, gerando uma matriz [R]_{3x3}.

A primeira linha é composta pelo vetor unitário do x local (λ_x), a segunda pelo vetor unitário do y local (λ_y) e, por fim, a terceira pelo vetor unitário do z local (λ_z). Algumas considerações são necessárias (Parreiras, 2019):

- O eixo x local é o eixo que liga o nó inicial ao final do elemento finito;
- O eixo y local é sempre perpendicular ao eixo x local e z global;
- O eixo z local é perpendicular aos eixos x e y locais.

A primeira linha é composta pela diferença das coordenadas entre os nós inicial e final normalizada. Seja $u_i = (x_i \ y_i \ z_i)$, $u_f = (x_f \ y_f \ z_f)$ e $\Delta u = u_f - u_i$, tem-se:

$$\lambda_x = \frac{\Delta u}{\|\Delta u\|} \quad (47)$$

Para garantir a ortogonalidade com λ_x , o vetor λ_y é definido em um plano perpendicular a λ_x . Esse vetor pode ser calculado utilizando o produto vetorial entre λ_x e um vetor arbitrário que não seja paralelo a ele (geralmente, o vetor z global é utilizado, ou $k = [0, 0, 1]$).

O vetor λ_y é, então, normalizado:

$$\lambda_y = \frac{k \times \lambda_x}{\|k \times \lambda_x\|} \quad (48)$$

Por fim, o vetor λ_z é obtido a partir do produto vetorial entre λ_x e λ_y :

$$\lambda_z = \lambda_x \times \lambda_y \quad (49)$$

Caso o vetor λ_x seja paralelo ao eixo z global, o vetor do eixo y local é igualado ao global, tal que $\lambda_y = [0, 1, 0]$, realizando analogamente o produto vetorial expresso na Eq. (49).

A matriz de rotação é, então, dada pela Eq. (50), a qual compõe a matriz de transformação, [T], conforme a Eq. (51).

$$[R] = \begin{bmatrix} \lambda_{x,1} & \lambda_{x,2} & \lambda_{x,3} \\ \lambda_{y,1} & \lambda_{y,2} & \lambda_{y,3} \\ \lambda_{z,1} & \lambda_{z,2} & \lambda_{z,3} \end{bmatrix} \quad (50)$$

$$[T] = \begin{bmatrix} [R] & 0 & 0 & 0 \\ 0 & [R] & 0 & 0 \\ 0 & 0 & [R] & 0 \\ 0 & 0 & 0 & [R] \end{bmatrix} \quad (51)$$

Onde todos os termos são compostos por matrizes 3x3.

Análise de estabilidade estrutural

Esforços axiais, como forças de compressão, influenciam os deslocamentos laterais em pilares e podem causar deslocamentos indesejáveis em chapas e cascas, enquanto forças de tração tendem a reduzir esses efeitos. Avaliar o impacto de cargas axiais requer uma análise não linear geométrica com as equações de equilíbrio formuladas na configuração deformada. Essa abordagem, que considera grandes deslocamentos e pequenas deformações, é

eficaz para estimar a carga crítica de flambagem (Vaz, 2011).

A Eq. (41) representa a contribuição das matrizes de rigidez elástica e geométrica. Contudo, a relação entre as matrizes é não linear, visto que K_σ depende da tensão que, por sua vez, depende dos deslocamentos axiais encontrados (Vaz, 2011). Para grandes deslocamentos, substituindo a Eq. (41) na Eq. (11), obtém-se a Eq. (52).

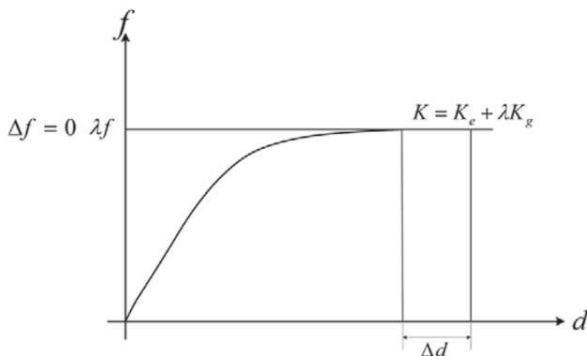
$$(K_e + K_\sigma)\Delta d = \Delta f \quad (52)$$

Para que ocorra a flambagem elástica em uma estrutura, calcula-se um fator λ que deve ser aplicado às cargas nodais, f , para que a estrutura produza deslocamentos não triviais, $\Delta d \neq 0$, mesmo sem incrementos nas cargas atuantes, isto é, $\Delta f = 0$. Assim:

$$(K_e + \lambda K_\sigma)\Delta d = 0 \quad (53)$$

Tal problema representa um sistema de n autovalores λ e n autovetores, sendo n a dimensão da matriz de rigidez. O menor valor de λ calculado é o fator de carga crítica, e o autovetor associado a este representa os deslocamentos da estrutura. O problema pode ser visualizado graficamente na Figura 8.

Figura 8 - Carga crítica de flambagem.



Fonte: Vaz, 2011.

METODOLOGIA

A presente seção descreve os métodos, ferramentas e etapas seguidas para o desenvolvimento da ferramenta computacional e a análise estrutural não linear geométrica de estruturas tridimensionais. A implementação do Método dos Elementos Finitos (MEF) foi realizada em linguagem Python, utilizando algoritmos específicos para a discretização da

estrutura, a resolução das equações de equilíbrio e a visualização dos resultados.

Todo o código foi realizado utilizando a versão mais recente disponível do Python (3.13.2), com bibliotecas como NumPy, Scipy, e PyVista, permitindo a geração de gráficos interativos e a manipulação eficiente de dados estruturais.

Estrutura do código

O programa desenvolvido foi nomeado SimuFrame (*Simulation Frame*), sendo dividido em três módulos principais:

1. Pré-processamento:
 - a. Definição da geometria da estrutura e das propriedades dos elementos;
 - b. Discretização dos elementos e obtenção da matriz de transformação, $[T]$;
 - c. Montagem do vetor de forças externas, $\{F\}$;
 - d. Construção da matriz de rigidez global elástica, $[K_e]$, e aplicação das condições de contorno.
2. Processamento computacional:
 - a. Obtenção dos esforços internos e deslocamentos nodais através da resolução do sistema apresentado na Eq. (12);
 - b. Construção da matriz de rigidez geométrica, $[K_\sigma]$, e obtenção dos modos de flambagem;
 - c. Aplicação do Método de Newton-Raphson Completo (NRC) para obtenção dos deslocamentos não lineares e seus respectivos esforços;
 - d. Armazenar todos os dados dos elementos para posterior plotagem.
3. Pós-processamento
 - a. Geração de uma interface gráfica (GUI) por meio da biblioteca oficial do PySide6, permitindo ao usuário selecionar os esforços e/ou

- deslocamentos a serem visualizados;
- b. Exibição dos gráficos tridimensionais com controle interativo de escala e parâmetros de análise;
 - c. Tabela de comparação dos resultados linear e não linear (esforços e deslocamentos).

Validação

A partir dos dados obtidos, foram analisadas diferentes estruturas de referência, desde elementos simples, como vigas e pilares, até sistemas estruturais mais complexos, como pórticos e treliças espaciais. O objetivo dessas análises foi validar a ferramenta computacional desenvolvida, comparando os resultados obtidos com aqueles gerados por *softwares* comerciais amplamente utilizados, como ANSYS, Abaqus e o Robot Structural Analysis (RSA).

Primeiramente, foram comparados os deslocamentos nodais para avaliar a precisão do método implementado, confrontando os resultados obtidos pelo código desenvolvido com as soluções fornecidas por *softwares* comerciais. Essa comparação abrangeu tanto a análise linear como a não linear geométrica, permitindo verificar a aderência dos valores calculados.

Além disso, foi realizada a análise de estabilidade estrutural para determinar os modos de flambagem e suas respectivas cargas críticas. Os resultados foram comparados diretamente com os fornecidos pelos *softwares* de referência, possibilitando a avaliação da eficiência do programa proposto neste estudo.

Dessa forma, busca-se garantir que a metodologia empregada forneça resultados coerentes e representativos da realidade estrutural, possibilitando sua aplicação em diferentes cenários de análise.

RESULTADOS E DISCUSSÕES

Para garantir a precisão dos resultados, todas as verificações foram realizadas com base em *softwares* comerciais amplamente

utilizados na engenharia estrutural. O ANSYS destacou-se por sua abordagem intuitiva e abrangente, consolidando-se como uma das ferramentas mais completas para uso acadêmico. O Abaqus, por sua vez, demonstrou maior coerência com a metodologia adotada neste estudo, sendo utilizado principalmente para determinar os deslocamentos na estrutura deformada. Já o RSA, embora apresente limitações na análise não linear, serviu como referência para a validação dos resultados obtidos tanto na análise linear quanto na não linear, além dos esforços internos. A comparação abrangeu deslocamentos e esforços na estrutura deformada, assegurando a confiabilidade da solução proposta.

Para as análises de estabilidade estrutural, todos os programas mencionados foram utilizados para comparar as 5 menores cargas críticas de flambagem.

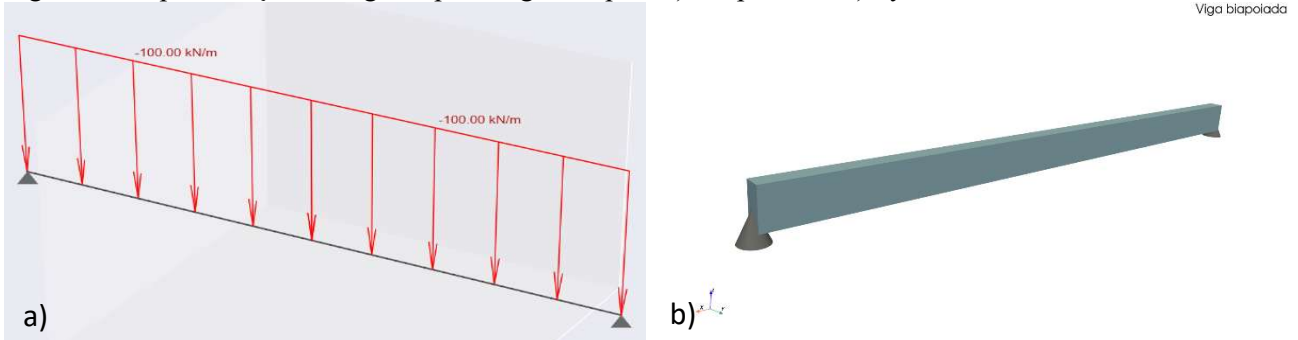
Os parâmetros da integração numérica pela quadratura de Gauss-Legendre foram mantidos constantes em todas as estruturas analisadas. Para o comprimento do elemento (eixo x local), adotaram-se três pontos de integração, assegurando maior precisão na determinação do campo de deslocamento, descrito por uma função cúbica.

No método de Newton-Raphson Completo, adotaram-se critérios de convergência baseados tanto na força quanto nos deslocamentos, ambos com uma tolerância mínima de 1.10^{-6} . O número de incrementos de carga é adaptável, sendo inicialmente definido como 5.

Viga biapoiada

Inicialmente, foi analisada uma viga biapoiada de 8 metros de comprimento, feita de concreto com seção transversal de 20×40 cm, submetida a uma carga distribuída de 100 kN/m, conforme ilustrado na Figura 9a. A Figura 9b representa a estrutura tridimensional. A magnitude da carga foi aumentada intencionalmente para destacar os efeitos da não linearidade geométrica. Adotou-se um módulo de elasticidade de 27000 MPa e um coeficiente de Poisson de 0,2.

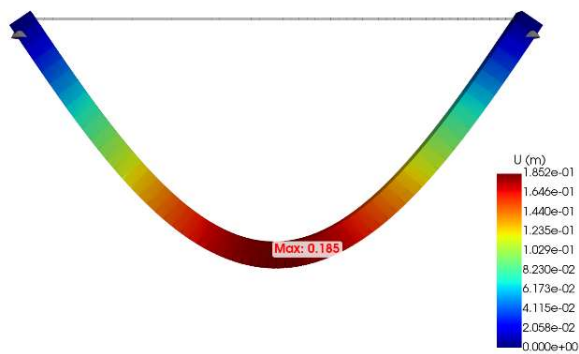
Figura 9 - Representação da viga biapoçada gerada pelo a) matplotlib e b) PyVista.



Fonte: Elaborado pelo próprio autor.

Sob tais condições, o deslocamento linear máximo no centro da viga é dado segundo a Figura 10, acompanhado da distribuição dos deslocamentos na região inferior direita. Já a Tabela 1 apresenta a comparação entre os resultados obtidos pelo código desenvolvido e aqueles fornecidos por *softwares* comerciais, destacando a diferença percentual entre as soluções e demonstrando a precisão da metodologia adotada.

Figura 10 - Deslocamento linear máximo da viga biapoçada, em metro (fora de escala).



Fonte: Elaborado pelo próprio autor.

Tabela 1: Deslocamentos lineares da viga biapoçada.

Fonte	$d_{\text{máx}}$ (cm)	Diferença (%)
SimuFrame	18,519	—
Abaqus	18,519	0,00
ANSYS	18,623	-0,56
RSA	18,519	0,00

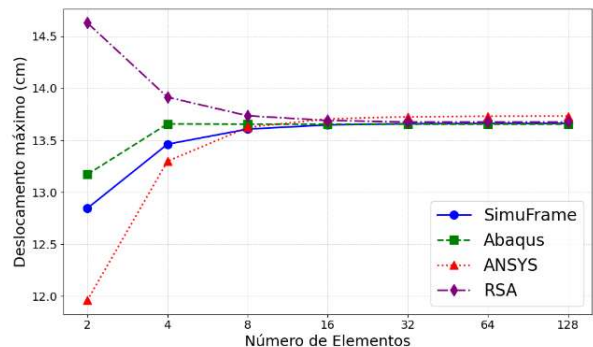
Fonte: Elaborado pelo próprio autor.

Todos os deslocamentos obtidos foram precisos, com exceção dos resultados do ANSYS, que apresentaram pequenas diferenças devido ao modelo estrutural empregado, baseado na Teoria de Timoshenko, a qual inclui a deformação por cisalhamento

em sua análise. Essas diferenças, no entanto, não comprometem a precisão dos resultados, uma vez que são consistentes com a abordagem teórica adotada pelo *software*.

Já os deslocamentos não lineares foram determinados por meio de uma análise iterativa-incremental, cuja precisão depende diretamente do número de elementos na malha. A Figura 11 mostra a relação entre o deslocamento máximo e o número de elementos, destacando como o aumento progressivo de elementos resulta na convergência dos resultados. Isso reforça a necessidade de equilibrar precisão e custo computacional durante a modelagem.

Figura 11 - Comparação do deslocamento não linear da viga biapoçada.



Fonte: Elaborado pelo próprio autor.

Percebe-se que a convergência dos resultados ocorreu em torno de 32 elementos, com o ANSYS apresentando o maior deslocamento máximo, equivalente a 13,732 cm, em virtude da inclusão da parcela de deformação por cisalhamento. O RSA, por sua vez, resultou em 13,676 cm.

Em contrapartida, tanto o código autoral quanto o Abaqus convergiram para valores muito próximos, com deslocamentos máximos de 13,663 cm e 13,654 cm,

respectivamente. Essa diferença percentual de apenas 0,066% entre os dois resultados pode ser atribuída a nuances nas formulações numéricas, visto que, segundo a documentação do *software*, é utilizada uma formulação corrotacional para a NLG, diferente da abordagem deste artigo.

Pilar engastado

A análise de estabilidade estrutural foi realizada através do estudo clássico de flambagem em pilares. Adotou-se um pilar circular de concreto com 6 m de altura e raio de 10 cm, mantendo as propriedades do material definidas no exemplo anterior.

Uma carga unitária foi aplicada no topo do pilar, enquanto sua base foi engastada. Essa configuração permite uma comparação direta entre os resultados computacionais e a solução analítica fornecida pela teoria de Euler.

A carga crítica de flambagem depende da seção transversal, material e comprimento do pilar, tal que:

$$P_{cr} = \frac{\pi^2 \cdot E \cdot I}{L_f^2}$$

Tabela 2: Cargas críticas de flambagem para o pilar.

Modo	1 elemento				8 elementos			
	SimuFrame	Abaqus	ANSYS	RSA	SimuFrame	Abaqus	ANSYS	RSA
1	146,435	146,440	147,150	146,435	145,342	145,340	144,970	145,342
2	146,435	146,440	147,150	146,435	145,342	145,340	144,970	145,342
3	1895,600	1895,600	3342,200	1895,600	1308,292	1308,300	1300,200	1308,091
4	1895,600	1895,600	3342,200	1895,600	1308,292	1308,300	1300,200	1308,091
5	—	—	—	—	3638,015	3638,000	3591,800	3633,840

Fonte: Elaborado pelo próprio autor.

Na primeira análise, observa-se que, embora os resultados sejam próximos entre si, nenhum deles atinge exatamente o valor teórico. Em particular, o ANSYS apresenta uma discrepância significativa, possivelmente devido à utilização de uma rotina de cálculo distinta dos demais *softwares*, o que pode envolver diferentes formulações numéricas, métodos de resolução ou considerações adicionais, como os efeitos de cisalhamento.

Já na segunda análise (8 elementos), verifica-se que, com exceção do ANSYS, todos os programas convergiram para o resultado teórico. Isso sugere que, à medida

Onde:

$$I_y = I_z = \pi \cdot \frac{r^4}{4} = 78,54 \cdot 10^{-6} m^4$$

$$L_f = 2 \cdot L = 12 m$$

$$\rightarrow P_{cr} = \frac{\pi^2 \cdot 2,7 \cdot 10^7 \cdot 78,54 \cdot 10^{-6}}{12^2}$$

$$\therefore P_{cr} = 145,342 kN$$

Portanto, para que ocorra a flambagem do pilar sob as dadas condições, é necessária uma carga mínima de 145,342 kN.

Inicialmente, a estrutura foi modelada sem discretização, considerando um único elemento com comprimento total de 6 m. Posteriormente, a estrutura foi subdividida em 8 elementos para avaliar a convergência dos resultados.

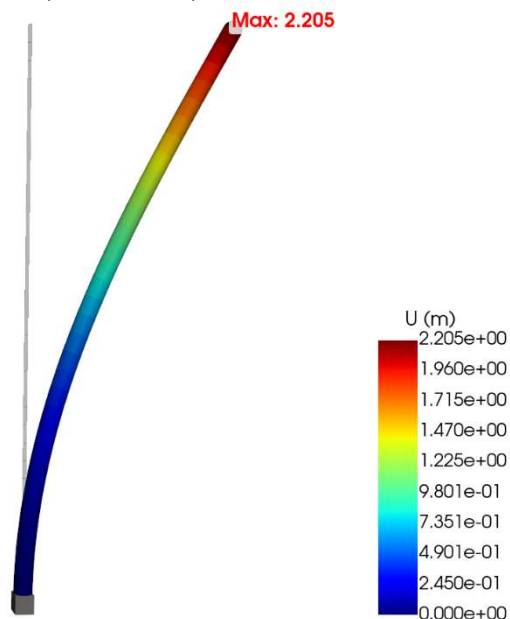
As cargas críticas de flambagem correspondentes aos cinco primeiros modos, obtidas tanto pelo código desenvolvido quanto pelos *softwares* comerciais, estão apresentadas na Tabela 2.

que a discretização do elemento aumenta, a acurácia na determinação da carga crítica também se aprimora.

Além disso, nota-se a recorrência de pares de cargas críticas idênticas. Esse comportamento está diretamente relacionado à simetria da estrutura, uma vez que, em relação aos eixos y e z, os momentos de inércia são equivalentes, permitindo que a flambagem ocorra de maneira similar em ambas as direções.

A Figura 12 demonstra como é o primeiro modo de flambagem, acompanhado dos deslocamentos.

Figura 12 - Primeiro modo de flambagem do pilar engastado ($\lambda = 145,342$); dados em metros.

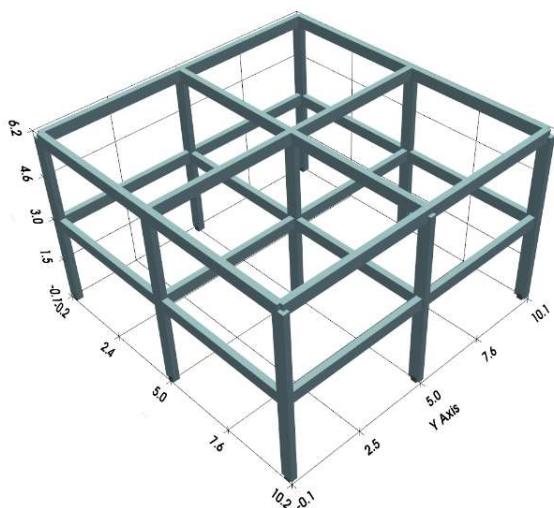


Fonte: Elaborado pelo próprio autor.

Pórtico espacial

Em seguida, foi analisado um pórtico espacial composto por vigas e pilares de seção 20 x 40 cm, conforme ilustrado na Figura 13. Os pilares possuem altura de 3 m, enquanto as vigas têm comprimento de 5 m. Todas as bases são engastadas.

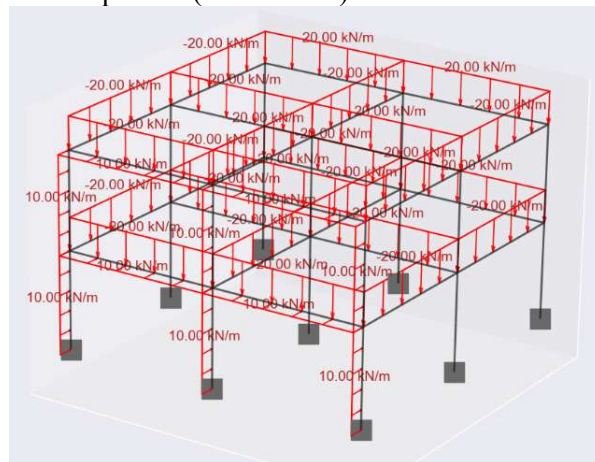
Figura 13 - Representação do pórtico espacial (SimuFrame).



Fonte: Elaborado pelo próprio autor.

As vigas foram submetidas a um carregamento distribuído de 20 kN/m, enquanto as vigas e pilares posteriores sofrem influência do vento, considerando-se uma carga de 10 kN/m (Figura 14).

Figura 14 - Representação dos esforços atuantes sobre o pórtico (SimuFrame).



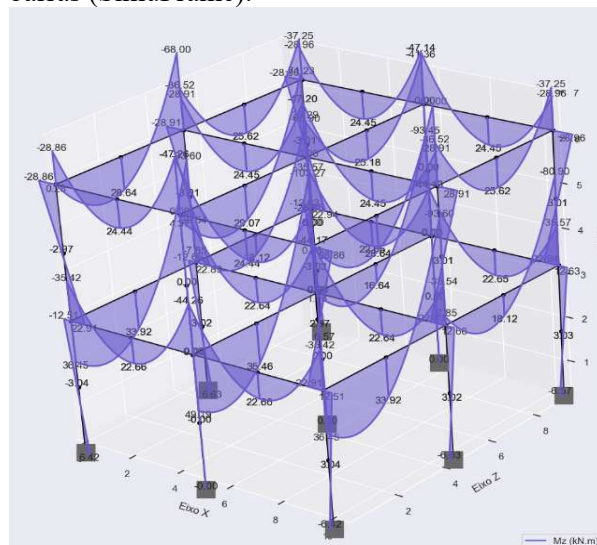
Fonte: Elaborado pelo próprio autor.

Este exemplo teve por objetivo validar os esforços internos, deslocamentos máximos não lineares e cargas críticas de flambagem, abrangendo todos os aspectos estruturais relevantes.

Os esforços podem ser representados de duas formas principais: por gráficos de barras, comum em *softwares* como Ftool e RSA, ou por mapas de cores, amplamente utilizados em *softwares* de elementos finitos, como Abaqus e ANSYS.

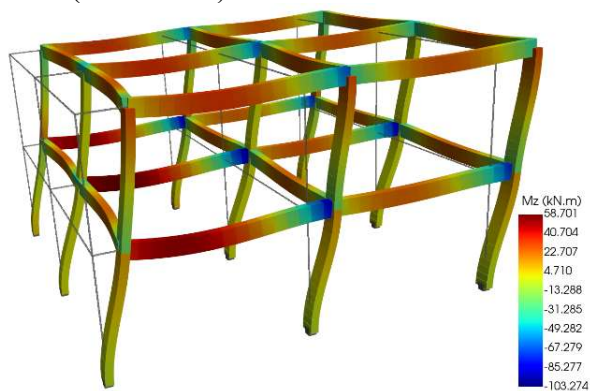
Na Figura 15, é apresentado o gráfico de barras dos momentos em relação ao eixo z (M_z), enquanto a Figura 16 o representa em mapa de cores.

Figura 15 - Representação de M_z por gráfico de barras (SimuFrame).



Fonte: Elaborado pelo próprio autor.

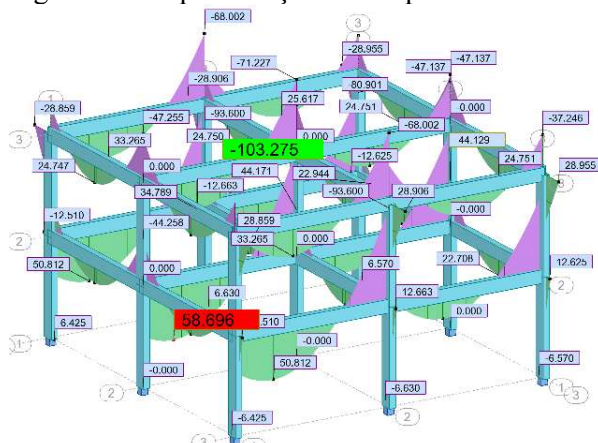
Figura 16 - Representação de M_z por mapas de cores (SimuFrame).



Fonte: Elaborado pelo próprio autor.

A partir das figuras, observa-se que os momentos máximo e mínimo são 58,701 kN.m e -103,274 kN.m, estando o elemento tracionado na face superior ou inferior, respectivamente. Os resultados obtidos pelo RSA são apresentados na Figura 17.

Figura 17 - Representação de M_z pelo RSA.



Fonte: RSA (AutoDesk, 2025).

Os momentos fletores extremos obtidos foram de -103,275 kN.m (máximo negativo) e 58,596 kN.m (máximo positivo), valores que corroboram a precisão da análise linear realizada pelo programa desenvolvido.

Para a análise de flambagem, determinaram-se os cinco primeiros modos de instabilidade, adotando-se uma discretização de 16 elementos. O ANSYS foi excluído desta comparação devido à incompatibilidade metodológica com a abordagem teórica empregada no estudo. O comparativo está expresso na Tabela 3.

Tabela 3: Cargas críticas de flambagem para o pórtico espacial.

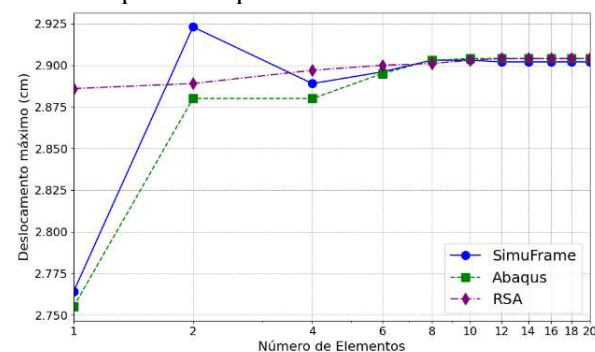
Modo	SimuFrame	Abaqus	RSA
1	24,814	26,077	24,814
2	36,381	36,343	36,386
3	39,848	40,082	39,850
4	50,650	50,157	50,654
5	57,901	57,886	57,418

Fonte: Elaborado pelo próprio autor.

Nota-se que a carga crítica mínima foi idêntica à obtida pelo RSA. O Abaqus, todavia, apresentou um valor relativamente distante dos demais para o primeiro modo de flambagem, aproximadamente 5,09% superior. Em análises subsequentes, com um refinamento progressivo da malha, observou-se uma tendência de redução dos autovalores calculados pelo Abaqus. Esse comportamento resultou em um valor mínimo de 25,836 para 24 subdivisões dos elementos, evidenciando a convergência dos resultados do Abaqus em direção aos valores obtidos pelo código autoral e pelo RSA. Com essa última convergência, a diferença foi reduzida para 4,12%. O refinamento não pôde avançar devido às limitações da versão estudiantil.

Por fim, os deslocamentos não lineares foram obtidos de maneira similar. O refinamento da malha foi limitado a 20, com os resultados expressos na Figura 18.

Figura 18 - Comparação do deslocamento não linear do pórtico espacial.



Fonte: Elaborado pelo próprio autor.

Nesse exemplo, os resultados obtidos pelo RSA e Abaqus convergiram para um deslocamento de 2,904 cm, enquanto o programa desenvolvido apresentou um valor de 2,902 cm, resultando em uma diferença de apenas 0,069%.

Por outro lado, o RSA demonstrou maior aderência aos resultados esperados em situações em que a diferença entre os deslocamentos linear e não linear não é tão expressiva, como observado neste caso, onde o deslocamento linear foi de 2,806 cm.

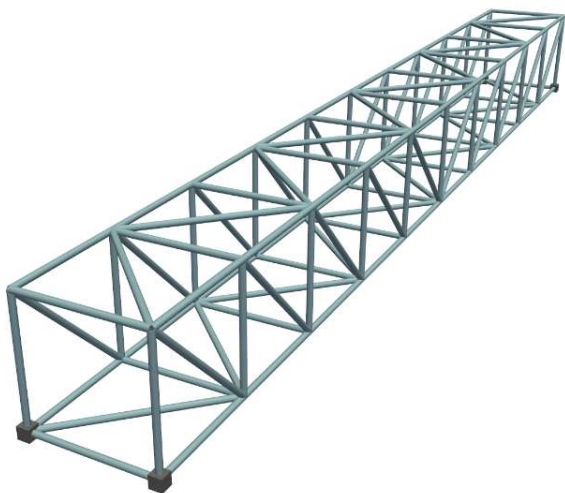
Treliça espacial

Por fim, foi analisada uma treliça espacial representando uma passarela metálica de 20 metros de comprimento, com largura e altura de 2 metros. A estrutura é composta por seções tubulares de 10 cm de diâmetro e 1 cm de espessura, em aço CA-50 ($E = 210.000$ GPa). Os nós foram distribuídos a cada 2 metros em todos os eixos, garantindo elementos de mesmo comprimento, exceto as diagonais. As extremidades dos banzos inferiores foram apoiadas, com todos os nós rotulados.

Adotou-se um carregamento fictício, aplicando 30 kN nos nós dos banzos inferiores e 15 kN nos superiores. O objetivo foi validar os deslocamentos lineares e não lineares da estrutura treliçada.

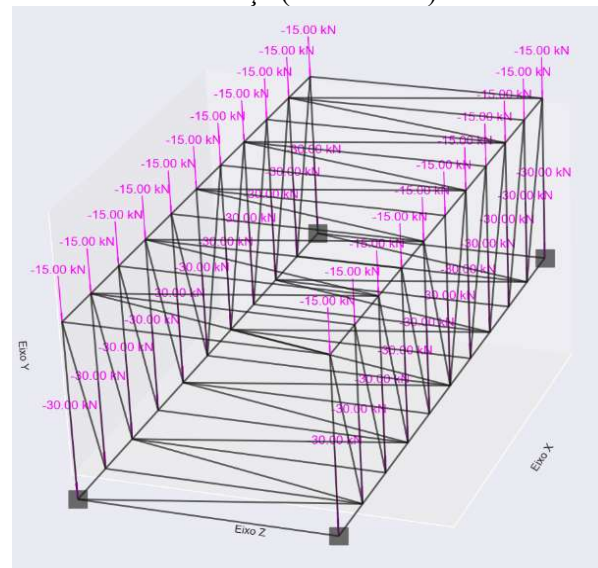
A estrutura é indicada nas figuras abaixo, com as deformações linear e não linear representadas nas Figuras 21 e 22.

Figura 19 - Representação da treliça espacial (SimuFrame).



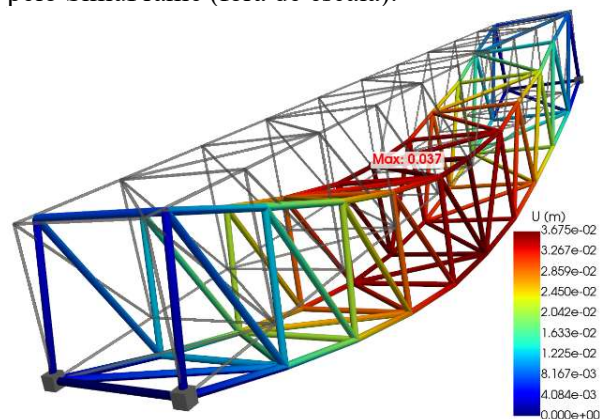
Fonte: Elaborado pelo próprio autor.

Figura 20 - Representação dos esforços atuantes sobre os nós da treliça (SimuFrame).



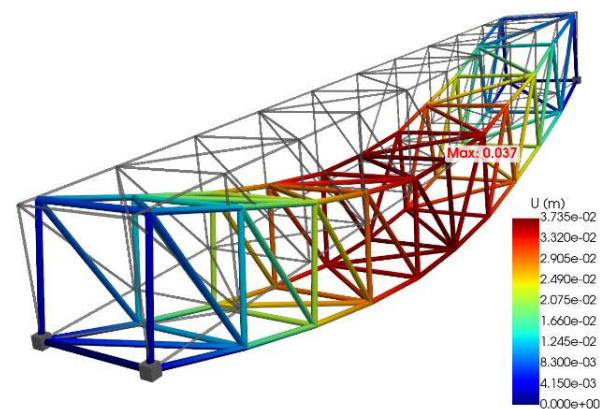
Fonte: Elaborado pelo próprio autor.

Figura 21 - Deformação linear da treliça espacial pelo SimuFrame (fora de escala).



Fonte: Elaborado pelo próprio autor.

Figura 22 - Deformação não linear da treliça espacial pelo SimuFrame (fora de escala).



Fonte: Elaborado pelo próprio autor.

Os deslocamentos máximos obtidos foram de 3,675 cm para a análise linear e 3,735 cm para a análise não linear. Devido à maior

escala da estrutura em comparação ao pórtico, a limitação no número de nós no Abaqus tornou-se ainda mais restritiva, permitindo apenas duas subdivisões. Essa restrição resultou em um deslocamento linear de 3,675 cm e um deslocamento não linear de 3,731 cm, com uma diferença de apenas 0,11% em relação ao programa desenvolvido.

Essa pequena discrepância pode ser atribuída à formulação corrotacional do Abaqus, que ajusta o sistema de coordenadas local conforme a deformação do elemento. Isso pode gerar pequenas variações em relação aos métodos convencionais.

No entanto, o resultado já se mostrou satisfatório, uma vez que a discrepância é praticamente desprezível.

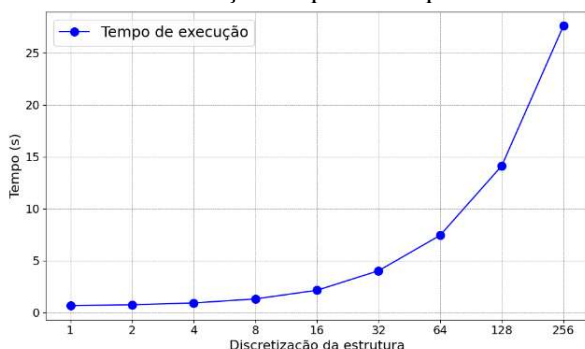
No RSA, a convergência não foi alcançada, mesmo após ajustes nos parâmetros incrementais da análise.

Tempo de processamento

Apesar da precisão satisfatória obtida com o programa autoral, algumas limitações foram identificadas, principalmente no que diz respeito à eficiência computacional. O Python, embora prático e de fácil utilização, apresenta restrições quanto ao uso de múltiplas *threads* da CPU, o que dificulta a execução de cálculos dinâmicos simultâneos para diversos elementos, possível em linguagens como o C++. Além disso, por não utilizar nativamente a GPU para cálculos numéricos, o processamento torna-se mais lento à medida que o número de elementos aumenta.

Na Figura 23 representam-se os tempos de execução do pórtico espacial em função da discretização estrutural.

Figura 23 - Tempo de execução para diferentes níveis de discretização do pórtico espacial.



Fonte: Elaborado pelo próprio autor.

Observa-se que o tempo computacional cresce exponencialmente com o aumento do número de elementos. Entretanto, malhas extremamente refinadas - como no caso analisado com 256 divisões - geralmente não são necessárias para a maioria das aplicações práticas em análise de pórticos espaciais.

CONCLUSÕES

Os resultados obtidos ao longo deste estudo evidenciam a precisão e confiabilidade do método implementado para a análise de estruturas tridimensionais considerando a não linearidade geométrica. A validação incluiu comparações com soluções analíticas e *softwares* como Abaqus, ANSYS e RSA.

Na viga biapoiada, os deslocamentos lineares apresentaram plena convergência, enquanto os não lineares diferiram em apenas 0,066% em relação ao Abaqus, demonstrando alta precisão. Para o pilar engastado, a solução convergiu com oito elementos, alinhando-se aos resultados teóricos e aos demais *softwares*, exceto pelo ANSYS, que mostrou maior discrepância.

No estudo do pórtico espacial, os esforços internos foram consistentes com os valores fornecidos pelo RSA, incluindo a carga crítica de flambagem. O Abaqus, por sua vez, se distanciou dos demais nesse aspecto. Quanto aos deslocamentos não lineares, a diferença em relação ao Abaqus e ao RSA foi aproximadamente desprezível (0,069%).

Já na treliça espacial, a diferença nos deslocamentos não lineares foi de apenas 0,11%, potencialmente menor sem as limitações do Abaqus estudantil. Embora os deslocamentos lineares convergiram em todos os programas, o RSA falhou na convergência não linear.

Desse modo, os resultados demonstram que o SimuFrame apresenta validação consistente tanto teórica quanto computacional frente a *softwares* comerciais, com resultados consistentes para deslocamentos, esforços internos e modos de flambagem. Limitações computacionais surgem em modelos excessivamente discretizados, mas o método prova-se eficaz e precisa para análises não lineares de pórticos espaciais.

REFERÊNCIAS

- FOLETTTO, Evandro Paulo; PAVAN, Roberto Carlos; PILZ, Silvio Edmundo; COSTELLA, Marcelo Fabiano. Estudo da não linearidade geométrica, instabilidade e ruptura de estruturas espaciais reticuladas. **Engevista**, [s. l.], v. 21, n. 1, p. 63–78, 2019. DOI 10.22409/engevista.v21i1. Disponível em: <https://periodicos.uff.br/engevista/article/view/10957>. Acesso em: 25 out. 2024.
- GAVIN, Henri. **Geometric Stiffness Effects in 2D and 3D Frames**. [S. l.]: Duke University, 2012.
- LOTTI, Raquel S.; MACHADO, André Wilson; MAZZIEIRO, Ênio Tonani; LANDRE JÚNIOR, Janes. Aplicabilidade científica do método dos elementos finitos. **Revista Dental Press de Ortodontia e Ortopedia Facial**, [s. l.], v. 11, p. 35–43, 2006. DOI <https://doi.org/10.1590/S1415-54192006000200006>. Disponível em: <https://www.scielo.br/j/dpress/a/6WCw3yW7WhXWnRYJRGtYzPq/>. Acesso em: 25 out. 2024.
- MCGUIRE, William; GALLAGHER, Richard H.; ZIEMIAN, Ronald D. **Matrix Structural Analysis**. 2. ed. New York: John Wiley & Sons, 2000.
- NONLINEAR Analysis. **Diana FEA**. [S. l.: s. n.], 2025. Disponível em: <https://manuals.dianafea.com/d102/Theory/Theorych54.html>. Acesso em: 7 fev. 2025.
- PAN, Wen-Hao; ZHAO, Chuan-Hao; TIAN, Yuan; LIN, Kai-Qi. Exact Solutions for Torsion and Warping of Axial-Loaded Beam-Columns Based on Matrix Stiffness Method. **Nanomaterials**, [s. l.], v. 12, n. 3, p. 538, 2022. DOI 10.3390/nano12030538. Disponível em: <https://www.mdpi.com/2079-4991/12/3/538>. Acesso em: 30 out. 2024.
- PARREIRAS, Emanuel Antônio. **Modeling and Simulation of 3D Frame Structures**. 2019. 57 f. Monografia (Bacharelado em Ciências da Computação) – Instituto de Ciências Exatas, Universidade Federal de Juiz de Fora, Juiz de Fora, 2019. Disponível em: <https://www.gcg.ufjf.br/pub/doc132.pdf>. Acesso em: 20 out. 2024.
- RODRIGUES, Marcos Antônio Campos; BURGOS, Rodrigo Bird; MARTHA, Luiz Fernando. A unified approach to the Timoshenko 3D beam-column element tangent stiffness matrix considering higher-order terms in the strain tensor and large rotations. **International Journal of Solids and Structures**, [s. l.], v. 222–223, p. 111003, 1 jul. 2021. DOI 10.1016/j.ijsolstr.2021.02.014. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0020768321000688>. Acesso em: 18 nov. 2024.
- SUESCÚN-DÍAZ, Daniel; CHALACASANOVA, Jesús A.; GÓMEZ-NOGUERA, Rosemberg. Cuadraturas de Gauss-Legendre para resolver numéricamente la ecuación inversa de la cinética puntual. **Información tecnológica**, [s. l.], v. 33, n. 3, p. 43–50, jun. 2022. DOI 10.4067/S0718-07642022000300043. Disponível em: https://www.scielo.cl/scielo.php?script=sci_abstract&pid=S0718-07642022000300043&lng=es&nrm=iso&tlng=es. Acesso em: 27 nov. 2024.
- VASILESCU, Adrian. **Analysis of geometrically nonlinear and softening response of thin structures by a new Facet Shell Element**. 2000. 207 f. Dissertação (Mestrado em Engenharia Civil e Ambiental) – Carleton University, 2000. Disponível em: <https://hdl.handle.net/20.500.14718/35846>. Acesso em: 23 nov. 2024.
- VAZ, Luiz Eloy. **Método dos Elementos Finitos em Análise de Estruturas**. Rio de Janeiro: Elsevier, 2011.

AGRADECIMENTOS

Primeiramente, expresso minha sincera gratidão aos meus orientadores, Prof. Dr. Felipi Pablo Damasceno Fernandes e Prof. Dr. Jordlly Reidson de Barros Silva, pelo incentivo contínuo e por todos os conhecimentos compartilhados ao longo deste projeto. Suas contribuições foram essenciais para o desenvolvimento deste trabalho.

Agradeço também à minha mãe e à minha namorada, cujo apoio e encorajamento foram fundamentais durante a minha trajetória acadêmica. Às amigas que construí ao longo do curso, deixo meu reconhecimento pelo companheirismo, pelas trocas de experiências e pelos momentos que tornaram essa jornada mais leve e enriquecedora.

Por fim, estendo meus agradecimentos à Universidade Federal Rural de Pernambuco e ao corpo docente do curso de Engenharia Civil, que contribuíram significativamente para minha formação e para a construção do conhecimento que tornou este trabalho possível.