

Aplicação voltada para análise não linear de treliças utilizando Python com programação na planilha Excel

Application focused on nonlinear analysis of trusses using Python with programming in the Excel spreadsheet

Aplicación dirigida al análisis no lineal de cerchas utilizando Python con programación en hoja de cálculo Excel

Gabriela Araújo Machado¹
Marcelo Greco²

Resumo: O objetivo do trabalho é apresentar uma ferramenta computacional para análise não linear de treliças, integrando a linguagem Python e o programa Excel. A hipótese da pesquisa baseia-se na possibilidade de desenvolvimento de uma ferramenta que permite a visualização e análise de estruturas com comportamento não linear geométrico. A metodologia é baseada na formulação posicional do Método dos Elementos Finitos, que permite a visualização da estrutura inicial e deformada após a leitura dos dados estruturais contidos na mesma planilha do script Python. Assim, a entrada e análise de dados tornam-se mais rápidas e simples. São apresentados quatro exemplos numéricos de estruturas com comportamento não linear, que permitem concluir que a metodologia utilizada é capaz de atingir o objetivo do trabalho.

Palavras-chave: Python. Excel. Elementos Finitos. Análise não linear. Treliças.

Abstract: The objective of this work is to present a computational tool for nonlinear analysis of trusses, integrating the Python language and the Excel program. The research hypothesis is based on the possibility of developing a tool that would allow the visualization and analysis of structures with geometrical nonlinear behavior. The methodology is based on the positional formulation of the Finite Element Method, which allows the visualization of the initial and deformed structures after reading the structural data contained in the same Python script spreadsheet. Thus, data entry and analysis become faster and simpler. Four numerical examples of structures with nonlinear behavior are presented, allowing us to conclude that the methodology used is capable of achieving the objectives.

Keywords: Python. Excel. Finite elements. Nonlinear analysis. Trusses.

Resumen: El objetivo del trabajo es presentar una herramienta computacional para el análisis no lineal de armaduras, integrando el lenguaje Python y el programa Excel. La hipótesis de investigación se basa en la posibilidad de desarrollar una herramienta que permitiera la visualización y análisis de estructuras con comportamiento geométrico no lineal. La metodología se basa en la formulación posicional del Método de Elementos Finitos, que permite la visualización de la estructura inicial y deformada luego de leer los datos estructurales contenidos en la misma hoja de cálculo que el script Python. Se presentan cuatro ejemplos numéricos de estructuras con comportamiento no lineal, que permiten concluir que la metodología utilizada es capaz de lograr el objetivo del trabajo.

Palabras-clave: Python. Excel. Elementos finitos. Análisis no lineal. Bragueros.

Submetido 02/12/2024

Aceito 07/04/2025

Publicado 11/04/2025

¹ Aluna de Graduação em Engenharia Aeroespacial. Universidade Federal de Minas Gerais. ORCID 0009-0005-0380-1211. E-mail: gabiaraujo2739@gmail.com.

² Doutor em Engenharia de Estruturas. Universidade Federal de Minas Gerais. ORCID 0000-0001-5500-0225. E-mail: mgreco@dees.ufmg.br.

Considerações Iniciais

A análise não linear geométrica de estruturas é um tema de grande relevância em diversas áreas da engenharia, incluindo a aeroespacial, civil, mecânica e naval. É um tipo de análise essencial quando o equilíbrio na configuração deformada de uma estrutura altera significativamente as respostas mecânicas. Pode estar associado a grandes deslocamentos, rotações e/ou deformações dos materiais constituintes da estrutura. Sistemas estruturais treliçados são concepções idealizadas de projeto nas quais os componentes estruturais trabalham à tração ou à compressão (Greco et al., 2006). São sistemas cinematicamente simples, mas podem apresentar comportamentos complexos, incluindo não linearidades física e geométrica a depender da topologia ou arranjo estrutural. As aplicações estruturais práticas podem ser encontradas em torres de linhas de transmissão, estruturas de cobertura de estádios, estruturas de aeronaves, pontes e passarelas, entre outras. As treliças se destacam pelo relativo baixo peso próprio quando comparadas com outros sistemas estruturais destinados à mesma finalidade (Machado et al., 2024).

A grande quantidade de elementos observada em estruturas treliçadas reais e a complexidade do comportamento estrutural demandam um cálculo automatizado para aplicações práticas na engenharia. O cálculo manual é praticamente inviável em aplicações reais de engenharia. Nesse sentido, o método numérico automatizado mais utilizado para resolução de problemas na Mecânica dos Sólidos e na Mecânica das Estruturas é o Método dos Elementos Finitos (MEF). A versão posicional do MEF, com referência em um único sistema de referência Lagrangiano e na descrição da cinemática calculada com base nos pontos nodais dos elementos estruturais, foi originalmente apresentada em Coda (2003). A formulação posicional foi aplicada posteriormente para treliças em Greco et al. (2006), com base na medida de deformação de Biot e seu par energético conjugado (tensão nominal). Coda (2018) apresentou o desenvolvimento da formulação posicional de treliças com base na medida de deformação de Green e seu par energético conjugado (2º tensor de Piola-Kirchhoff). No presente trabalho, utiliza-se um tipo de formulação posicional do MEF. Essa formulação é baseada em coordenadas nodais absolutas, como as apresentadas nos trabalhos de Shabana (1997), Bonet et al. (2000), Aguiar e Fosdick (2000) e Coda (2003). Entretanto, cabe ressaltar que a

formulação utilizada aqui tem semelhanças, mas não é idêntica às demais formulações baseadas em coordenadas absolutas.

Em relação ao uso do Python integrado à análise não linear de treliças, baseada na formulação não linear posicional, Bonfim et al. (2024) apresentam a interação entre um código, desenvolvido em FORTRAN, e uma interface gráfica desenvolvida em Python para aplicações de treliças planas. Os autores utilizam as bibliotecas PyQt, PyOpenGL e HETOOL. Oliveira (2015) apresentou um código em Python para resolver problemas de treliças espaciais com comportamento não linear geométrico usando a formulação posicional do método dos elementos finitos.

Em relação às outras formulações do Método dos Elementos Finitos, Souza (2019) demonstra que a formulação posicional aplicada a treliças com comportamento acentuado não linear geométrico fornece resultados de igual acurácia aos resultados obtidos pela clássica formulação corrotacional.

Assim, o presente artigo é relacionado com projeto de Iniciação Científica intitulado “Desenvolvimento de software livre para análise de treliças não lineares: Python Positional Truss 3D” (Machado, 2023) e visa apresentar uma ferramenta para a realização de análise estrutural, considerando a análise não linear. Em Machado et al. (2024) e Machado (2024) foi apresentada uma versão simplificada da formulação, utilizando-se a própria IDLE (*Integrated Development and Learning Environment*) do Python (Python Software Foundation, 2024). Isso foi possível devido à recente integração do Python com o Microsoft Excel, possibilitada pela atualização disponibilizada pelo pacote Microsoft 365 Insider. Esta integração abre um leque de possibilidades para análise de dados, visualização e automação de tarefas, tornando-se particularmente útil no contexto da engenharia estrutural. Ela pode diminuir o tempo de processamento, compactar os códigos e tornar a análise mais acessível. Qurbani (2024) foi um dos raros trabalhos encontrados na área de Mecânica dos Sólidos que utiliza a metodologia de programação do Python integrado ao Excel para resolver um problema de aquecimento e resfriamento de tubos extrudados.

A ferramenta desenvolvida utiliza bibliotecas do Python, como NumPy, para cálculos numéricos, Matplotlib, para visualização gráfica, e Pandas, para manipulação e análise de dados. Nesse âmbito, há diversos benefícios significativos para o desenvolvimento do script no

Excel, como a realização de análises complexas, modelagem estatística e até mesmo aprendizado de máquina diretamente dentro das planilhas, tornando mais simples a leitura de dados.

Metodologia

A metodologia utilizada no presente artigo é de natureza quantitativa e os instrumentos metodológicos utilizados são baseados em experimentos e estudos de caso. Para realização dos experimentos numéricos foi necessário o desenvolvimento de uma formulação numérica adequada para análise do comportamento não linear geométrico de estruturas, cuja acurácia pode ser verificada pela comparação com resultados analíticos e com os resultados obtidos por cálculo manual, quando for viável. Os estudos de caso tiveram como objetivo comprovar a viabilidade da aplicação da metodologia para o objetivo do trabalho.

1. Método numérico implementado computacionalmente

A formulação numérica é baseada no trabalho de Greco e Peixoto (2021), desenvolvida originalmente para a medida de deformação seno hiperbólico e adaptada aqui para a medida de deformação de Biot.

Os estudos sobre comportamento não linear geométrico partem da definição de estiramento (λ) (Driemeier et al., 2005).

$$\lambda = \frac{ds}{dS_0} \quad (1)$$

em que ds e dS_0 são os comprimentos de uma fibra na configuração deformada (L) e inicial (L_0), respectivamente. No caso da treliça esses comprimentos correspondem aos comprimentos dos próprios elementos finitos.

O estiramento neutro ($\lambda = 1$), independentemente da medida de deformação utilizada, sempre corresponde à deformação normal nula ($\varepsilon = 0$). No caso da medida de deformação de Biot, tem-se:

$$\varepsilon = \lambda - 1 \quad (2)$$

O par energético conjugado de tensão é obtido a partir da energia de deformação específica (u) de cada um dos elementos finitos, conforme apresentado em Greco e Peixoto (2021).

$$u = \int_{\varepsilon} \sigma d\varepsilon = \int_{\lambda} \sigma \frac{\partial \varepsilon}{\partial \lambda} d\lambda = \int_{\lambda} \sigma d\lambda \quad (3)$$

Os pares energéticos de tensão-deformação devem ser conservativos. Assim, o termo $\sigma(\partial\varepsilon/\partial\lambda)$ deve ser constante e igual à tensão nominal (σ_N) medida em ensaios de laboratório. Pode-se reescrever a equação (3) da seguinte forma:

$$u = \int_{\lambda} \sigma_N d\lambda = C \int_{\lambda} (\lambda - 1) d\lambda = C \left(\frac{\lambda^2}{2} - \lambda \right) \Big|_{\lambda_1}^{\lambda_2} \quad (4)$$

em que C é um parâmetro relacionado com as propriedades do material, que no caso assume valor unitário.

No presente trabalho, será utilizada a descrição cinemática proposta originalmente por Coda (2003), que parte da parcela elástica da energia de deformação (U) de um elemento finito (m) expressa por:

$$U^{(m)}(\alpha x_i) = A_0^{(m)} \cdot L_0^{(m)} \cdot u_e(\varepsilon(\alpha x_i)) \quad (5)$$

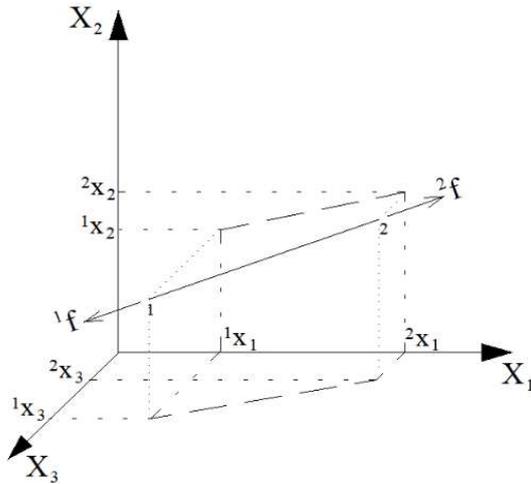
em que α é o nó do elemento, i é a direção do grau de liberdade (DoF), L_0 é o comprimento inicial do elemento, A_0 é a seção transversal do elemento na configuração indeformada e u_e é a energia de deformação específica (por unidade de volume) do elemento.

No caso do elemento de treliça 3D, a energia de deformação e a força interna de equilíbrio ($[{}^{\alpha}f_i]^{(m)}$) associada às posições nodais em determinada configuração deformada ($[{}^{\alpha}x_i]^{(m)}$), estão relacionadas por:

$$[{}^{\alpha}f_i]^{(m)} = \frac{\partial U^{(m)}}{\partial [{}^{\alpha}x_i]^{(m)}} \quad (6)$$

A Figura 1 apresenta o elemento de treliça 3D com as forças internas de equilíbrio aplicadas nos nó inicial (1f) e final (2f). Portanto, a malha de elementos finitos é definida pelos próprios elementos de barra da treliça (sem discretização adicional de domínio).

Figura 1 – Elemento de treliça 3D



Fonte: Autores (2024)

Considerando-se as Equações (5) e (6) e aplicando-se a regra da cadeia, tem-se:

$$\left[{}^{\alpha}f_i \right]^{(m)} = A_0^{(m)} \cdot L_0^{(m)} \cdot \frac{\partial u_e}{\partial [\varepsilon]^{(m)}} \frac{\partial [\varepsilon]^{(m)}}{\partial [{}^{\alpha}x_i]^{(m)}} = A_0^{(m)} \cdot L_0^{(m)} \cdot \sigma \cdot \frac{\partial [\varepsilon]^{(m)}}{\partial [{}^{\alpha}x_i]^{(m)}} \quad (7)$$

Os comprimentos inicial ($L_0^{(m)}$) e final ($L^{(m)}$) de um elemento finito m são calculados com base nas posições das configurações inicial (X) e deformada (x) e com base nos nós inicial e final de cada elemento finito.

$$L_0^{(m)} = \sqrt{({}^2X_1 - {}^1X_1)^2 + ({}^2X_2 - {}^1X_2)^2 + ({}^2X_3 - {}^1X_3)^2} \quad (8)$$

$$L^{(m)} = \sqrt{({}^2x_1 - {}^1x_1)^2 + ({}^2x_2 - {}^1x_2)^2 + ({}^2x_3 - {}^1x_3)^2} \quad (9)$$

A medida de deformação de Biot pode ser expressa em função das posições nodais na configuração deformada, conforme a seguir:

$$\varepsilon^{(m)} = \frac{\sqrt{({}^2x_1 - {}^1x_1)^2 + ({}^2x_2 - {}^1x_2)^2 + ({}^2x_3 - {}^1x_3)^2}}{L_0} - 1 \quad (10)$$

A derivada da medida de deformação em relação às posições nodais na configuração deformada fornece as seguintes equações:

$$\frac{\partial [\varepsilon]^{(m)}}{\partial [{}^{\alpha}x_i]^{(m)}} = (-1)^{\alpha} \cdot \frac{({}^2x_i - {}^1x_i)}{L^{(m)} \cdot L_0^{(m)}} \quad (11)$$

$$[{}^{\alpha}f_i]^{(m)} = A_0^{(m)} \cdot \sigma \cdot (-1)^{\alpha} \cdot \frac{({}^2x_i - {}^1x_i)}{L^{(m)}} \quad (12)$$

No equilíbrio da posição deformada há igualdade entre as forças externas aplicadas ($[{}^{\alpha}F_i]$) e as forças internas ($[{}^{\alpha}f_i]$) em cada elemento finito (m).

$$[{}^{\alpha}g_i]^{(m)} = [{}^{\alpha}f_i]^{(m)} - [{}^{\alpha}F_i]^{(m)} = 0 \quad (13)$$

Para resolver o sistema de equações não lineares, apresentado na Equação (13), deve-se minimizar o vetor dos resíduos $[{}^{\alpha}g_i]$, usando o método iterativo de Newton-Raphson, até que o módulo do vetor dos resíduos seja menor que determinada tolerância numérica.

O vetor das correções das posições ($[\Delta {}^{\alpha}x_i]$) é calculado por:

$$[\Delta {}^{\alpha}x_i] = - \left[\frac{\partial^2 U_e^{(m)}}{\partial [{}^{\alpha}x_i] \partial [{}^{\gamma}x_i]} \Big|_{\underline{x}} \right]^{-1} \cdot [g({}^{\alpha}x_i)] = - [{}^{\alpha\gamma}H_{ik}]^{-1} \cdot [{}^{\alpha}x_i] \quad (14)$$

em que $[{}^{\alpha\gamma}H_{ik}]$ é a matriz hessiana, γ é o nó e k é a direção do grau de liberdade.

A matriz hessiana é calculada pela derivada da Equação (7), aplicando-se a regra da cadeia.

$$[\nabla g({}^{\alpha}x_i)]^{(m)} = [{}^{\alpha\gamma}H_{ik}]^{(m)} = A_0^{(m)} \cdot L_0^{(m)} \cdot \left(E_t(\varepsilon) \cdot \frac{\partial \varepsilon}{\partial [{}^{\alpha}x_i]} \cdot \frac{\partial \varepsilon}{\partial [{}^{\gamma}x_i]} + \sigma \cdot \frac{\partial^2 \varepsilon}{\partial [{}^{\alpha}x_i] \partial [{}^{\gamma}x_i]} \right) \quad (15)$$

em que $E_t(\varepsilon)$ é o módulo de elasticidade longitudinal tangente.

O procedimento iterativo de Newton-Raphson é ilustrado na Figura 2.

De acordo com a Figura 2, o equilíbrio não linear é dado pelo equilíbrio das forças internas e externas relacionadas com a estrutura.

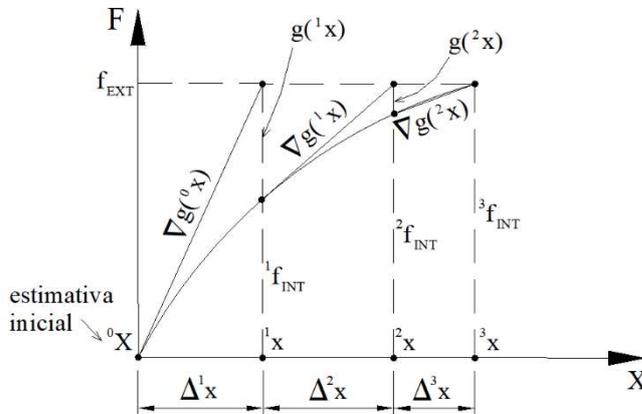
$$g(X) = f_{INT}(X) - f_{EXT}(X) = 0 \quad (16)$$

O método de Newton-Raphson considera a expansão de Taylor em 1ª ordem para aproximar o equilíbrio:

$$g(X) \cong \nabla g({}^iX) \Delta^{i+1}X + g({}^iX) = 0 \quad (17)$$

$$\nabla g({}^iX) \Delta^{i+1}X = -g({}^iX) = f_{EXT}({}^iX) - f_{INT}({}^iX) \quad (18)$$

Figura 2 – Procedimento iterativo de Newton-Raphson para busca do equilíbrio estrutural



Fonte: Autores (2024)

O resíduo para uma iteração “*i*” é calculado por:

$$g({}^iX) = f_{INT}({}^iX) - f_{EXT}({}^iX) \quad (19)$$

O processo iterativo se repete enquanto $\|g({}^iX)\| > \textit{tolerância}$.

2. Ferramenta computacional desenvolvida no Excel

A ferramenta foi desenvolvida inicialmente utilizando a IDE WinPython e principalmente a biblioteca Pandas, para leitura de um arquivo de entrada com extensão .xlsx ou .txt. Os dados do arquivo de entrada vão desde número de nós e de elementos, até quais são as coordenadas dos nós, quais graus de liberdade possuem, os carregamentos prescritos e as propriedades dos elementos, como módulo de elasticidade e área de seção transversal.

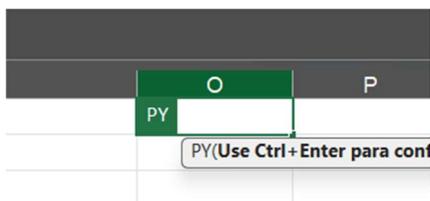
Por meio da biblioteca Pandas foi possível ler essas características da estrutura no arquivo.xlsx e manipulá-los para realizar as operações e descrever o comportamento da estrutura após aplicação da força.

Os dados obtidos após o processamento, como informações sobre os resíduos gerados com o método iterativo utilizado, dados em cada passo sobre nós especificados e como variam as coordenadas dos nós ao longo dos cálculos, são concatenados em um arquivo .xlsx ao final da execução.

Diante da grande utilização de planilhas para análises poderia otimizar o processo e facilitar as alterações sobre as informações da estrutura.

Para habilitar essa nova funcionalidade no Excel basta realizar o passo a passo disponível em Microsoft Corporation (2024). Para começar a executar códigos em Python nas células do Excel, basta digitar o comando “=PY”. Dessa forma, é possível ver que a célula irá mudar de cor para verde, conforme apresentado na Figura 3.

Figura 3 – Célula da planilha Excel preparada para programação em Python



Fonte: Autores (2024)

Alguns passos para leitura de dados foram reduzidos e/ou simplificados. Além disso, os arquivos de saída, com informações quantitativas sobre a análise, são gerados em outras abas do mesmo arquivo .xlsx que contém o código, facilitando a leitura dos resultados e a modificação das informações de entrada.

Após importar as bibliotecas utilizadas no código (Numpy, Matplotlib e Pandas), é feita a leitura das primeiras informações sobre a estrutura, a saber: Número de Elementos, de Nós, de Nós com Graus de Liberdade Prescritos, de Nós com Carregamentos Prescritos, a Tolerância admitida e dentre outras. As informações foram armazenadas em variáveis do tipo *dataframe* e depois são convertidas no formato de números inteiros.

Já os dados sobre as coordenadas dos nós, as propriedades sobre cada elemento, quais nós e em quais eixos há movimento restrito ou se já possuem deslocamento prescrito e os carregamentos aplicados, foram também armazenados em variáveis do tipo *dataframe* e depois convertidos na estrutura de dados “lista” do Python, diante da imensa gama de métodos que podem ser aplicados a ela e a facilidade de manipulação.

O *DataFrame*, da biblioteca Pandas, é uma maneira de representar e trabalhar com dados tabulares (bidimensionais), sendo facilmente criados utilizando Excel: basta apenas ativar a função “Python” no Excel, como mostrado inicialmente, e selecionar as células do Excel desejadas. Se for necessário, como nesta aplicação, manipular os dados *dataframe*,

recomenda-se atribuir a uma variável. Um exemplo de arquivo de estrutura treliçada é mostrando Figura 5.

Figura 4 – Exemplo de arquivo de entrada estruturado como dados no formato *dataframe*

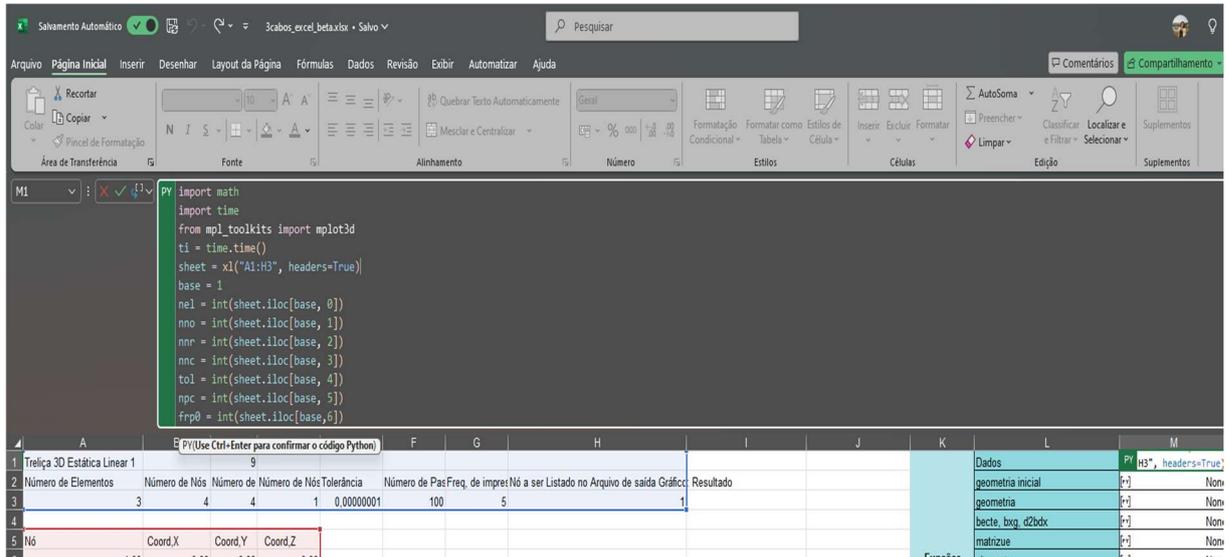
A	B	C	D	E	F	G	H	I	J
Treliza 3D Estática Linear 1									9
Número de Elementos	Número de Nós	Número de Nós com Graus de Liberdade Prescritos	Número de Nós com Carregamentos Prescritos	Tolerância	Número de Pa	Freq. de impre	Nó a ser Listado	no Arquivo de saída Gráfico	Resultado
3	4	4	4	1	0,00000001	100	5	1	
Nó	Coord,X	Coord,Y	Coord,Z						
1,00	0,00		0,00	0,00					
2,00	-432,55		-20,00	-250,00					
3,00	432,55		-20,00	-250,00					
4,00	0,00		-20,00	493,60					
Elemento	Nó Inicial	Nó Final	Elasticidade	Área	KM	HM	SY	DEFINICIAL [%]	
1,00	1,00	2,00	20500,00	6,53	0,00	0,00	100000000,00	0,00	
2,00	1,00	3,00	20500,00	6,53	0,00	0,00	100000000,00	0,00	
3,00	1,00	4,00	20500,00	6,53	0,00	0,00	100000000,00	0,00	
Restrição de Graus de Liberdade									
Nó	GL1	GL2	GL3						
1	0		0	1					
2	1		1	1					
3	1		1	1					
4	1		1	1					
Leitura dos Valores Prescritos									
Nó	Px	Py	Pz						
1	0	0	0	0					
2	0	0	0	0					
3	0	0	0	0					
4	0	0	0	0					
Nós com Carregamento Prescritos									
Nó	Fx	Fy	Fz						
1	0,00		-20,00	0,00					

Fonte: Autores (2024)

É apresentado na Figura 5 um exemplo de criação da variável nomeada como “*sheet*”, a qual deve ser seguida do sinal de igualdade (=), para ser possível atribuir valores. Sendo assim, mantendo a seta do mouse ao lado do sinal de “=”, deve-se selecionar na planilha os valores desejados e eles são atribuídos no formato de *DataFrame* à variável desejada.

Pode ser visto, na imagem abaixo, o código “xl(...)” que é criado pelo Excel automaticamente. Assim, tem-se o primeiro *dataframe* e após isso, as informações foram separadas em variáveis do tipo inteiro (int), como o número de elementos (primeiro valor do *dataframe*, posição 0) e foi armazenado na variável “nel”.

Figura 5 – Exemplo de criação da variável “sheet” criada no formato de *DataFrame*



Fonte: Autores (2024)

As coordenadas dos nós também foram armazenadas no formato *dataframe* e depois convertidas em listas, utilizando o método “*to_list*”, conforme ilustrado na Figura 6. Nessa parte, foram criadas três listas, cada uma armazenando as coordenadas dos eixos x, y e z (equivalente aos eixos X_1 , X_2 e X_3 da formulação numérica desenvolvida).

Visando tornar o preenchimento de dados mais flexível, não é necessário preencher os nós ou elementos em ordem crescente, sendo possível preencher em qualquer ordem. Para isso, após a leitura de dados, eles são posicionados na lista em ordem crescente, por meio do método “*sort_values*”.

Figura 6 – Exemplo de conversão de um conjunto de dados do formato *DataFrame* para o formato de lista

```

nodes_coord = xl("A5:D9", headers=True)
nodes_coord.columns = ["Node", "x", "y", "z"]
nodes_coord = nodes_coord.sort_values(by="Node")
Node = nodes_coord["Node"].tolist()
x = nodes_coord["x"].tolist()
y = nodes_coord["y"].tolist()
z = nodes_coord["z"].tolist()

elements = xl("A11:I14", headers=True)
elements.columns = ["Element", "Initial node", "Final node", "Elasticity", "Area", "KM", "HM", "SY", "INITIAL STRAIN [%]"]
    
```

Nó	Coord,X	Coord,Y	Coord,Z
1,00	0,00	0,00	0,00
2,00	-432,55	-20,00	-250,00
3,00	432,55	-20,00	-250,00
4,00	0,00	-20,00	499,60

Fonte: Autores (2024)

As funções para realização dos cálculos sobre comportamento da estrutura foram armazenadas em várias células do Excel, marcadas por meio da cor azul para facilitar a visualização. É possível colocar todo o código em apenas uma célula do Excel. Contudo, visando auxiliar na correção de erros e no entendimento do *script*, as funções foram distribuídas em 14 células, conforme ilustrado na Figura 7. O Excel executa o código linha por linha da planilha e considera nos cálculos das funções os resultados obtidos nas funções da linha anterior.

Figura 7 – Exemplo de conversão de um conjunto de dados do formato *DataFrame* para o formato de lista

K	L	M
Funções	Dados	[*] None
	geometria inicial	[*] None
	geometria	[*] None
	becte, bxg, d2bdx	[*] None
	matrizue	[*] None
	elemento	[*] None
	monta	[*] None
	matriz, plastific	[*] None
	condon, destroca, calcula norma	[*] None
	esforços e saída	[*] None
MAIN	Gráfico estrutura inicial	[*] Image
	Gráfico DOF	[*] Image
	Gráfico estrutura deformada	[*] Image

Fonte: Autores (2024)

Na Figura 7, as funções realizam as seguintes operações:

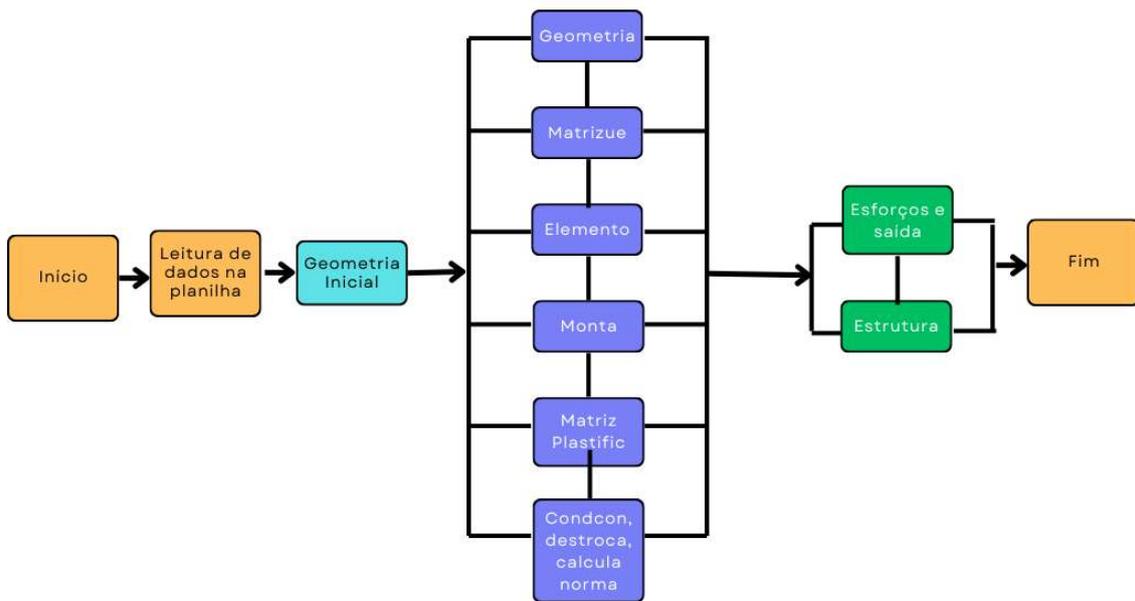
- Dados: faz a leitura dos dados de entrada do problema (número de nós, número de elementos, nós com apoios, nós carregados, posições iniciais dos nós, conectividade das barras, propriedades físicas e geométricas das barras);
- geometria inicial: calcula os parâmetros geométricos da configuração inicial (indeformada) da estrutura;
- geometria: calcula os parâmetros geométricos da configuração deformada da estrutura;
- matrizue: calcula a matriz hessiana com base na energia de deformação de cada elemento;
- elemento: calcula a força interna de equilíbrio de cada elemento;
- monta: monta o vetor das forças internas de todos os elementos da estrutura;
- matriz, plastific: monta a matriz hessiana de todos os elementos da estrutura e aplica correções elastoplásticas (quando houver não linearidade física envolvida na análise);
- condcon, destroca, calcula norma: aplica as condições de contorno, resolve o sistema de equações e calcula a norma euclidiana do vetor dos resíduos (para comparação com a tolerância numérica adotada);
- esforços e saída: gera as abas na planilha com resultados das forças axiais, posições nodais deformadas e posições nodais de um nó específico;

- estrutura: parametriza a estrutura para geração das saídas gráficas usando a biblioteca Matplotlib.

Dessa forma, o Python foi incorporado como funções internas no Excel e a leitura de dados é feita na sequência mostrada na Figura 8.

É possível perceber que apenas as três últimas funções têm retorno gráfico, podendo ser visualizada a representação da estrutura inicial e da estrutura deformada. No desenvolvimento do *script*, foram utilizadas diversas bibliotecas, funções e métodos, os quais são mostrados no Quadro 1, tornando o processo mais eficiente.

Figura 8 – Fluxograma mostrando o fluxo da leitura de dados e cálculos realizados



Fonte: Autores (2024)

Quadro 1 – Bibliotecas e componentes utilizadas no código desenvolvido

BIBLIOTECA	NUMPY	MATPLOTLIB	PANDAS
Métodos	-	subplot , title, plot, legend, grid	tolist, sort_values, loc, iloc
Funções	array, linalg (submódulo),		concat

	solve (função dentro do linalg)	figure, show	
Classes	-	-	dataframe

Fonte: Autores (2024)

A presente seção apresentou a formulação numérica essencial para o desenvolvimento da ferramenta computacional destinada à análise não linear de treliças. Também foram apresentados os aspectos computacionais relacionados com a integração entre a linguagem Python e o programa Excel. Com a ferramenta desenvolvida são realizadas as análises dos dados e resultados da próxima seção.

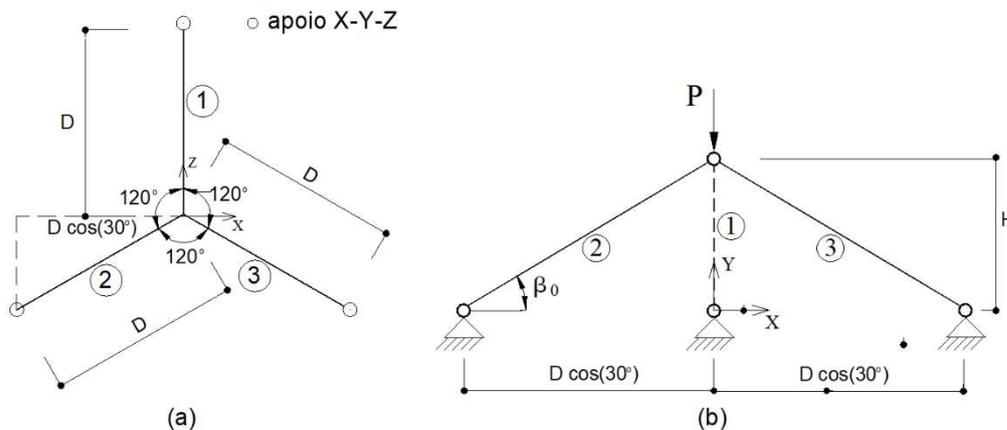
Análise dos dados e resultados

São apresentados quatro exemplos com os resultados obtidos após analisar estruturas treliçadas representativas. Para todos os exemplos de aplicação foi considerada uma tolerância numérica para o erro absoluto de 10^{-8} .

1. Treliça 3D simétrica com três barras

Trata-se de um exemplo analítico original de uma treliça 3D simétrica com três barras. Na Figura 9 apresenta-se a geometria, condições iniciais e condições de contorno do problema, em que a dimensão “D” é representada em planta.

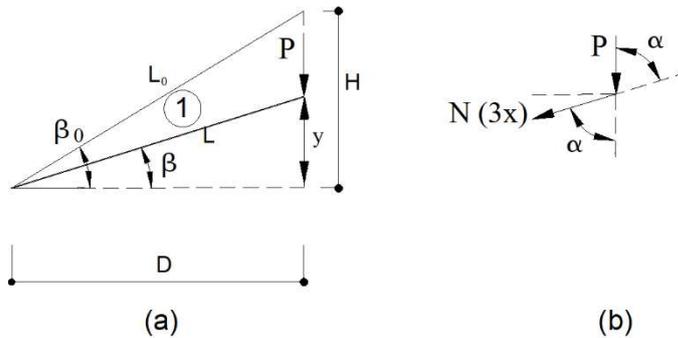
Figura 9 – Treliça 3D simétrica com três barras: (a) vista superior; (b) vista lateral



Fonte: Autores (2024)

Na Figura 10 apresenta-se a cinemática de um elemento da estrutura e o equilíbrio do nó central (nó 1). Cabe ressaltar que, devido à simetria, as cinemáticas das três barras são iguais.

Figura 10 – Elemento 1 da treliça de três barras: (a) cinemática; (b) equilíbrio



Fonte: Autores (2024)

A relação cinemática da posição deformada, apresentada na Figura 9(a) fornece as seguintes relações:

$$\tan(\beta_0) = \frac{H}{D} \quad (20)$$

$$\tan(\beta) = \frac{y}{D} \quad (21)$$

$$L = \frac{D}{\cos(\beta)} \quad (22)$$

$$L_0 = \frac{D}{\cos(\beta_0)} \quad (23)$$

A aplicação definição de estiramento, apresentada na Eq. (1), fornece:

$$\lambda = \frac{L}{L_0} = \frac{\cos(\beta_0)}{\cos(\beta)} \quad (24)$$

Pela Figura 9(b) verifica-se que a projeção da força axial \vec{N} na direção da coordenada vertical \vec{y} fornece o ângulo α :

$$\cos(\alpha) = \frac{\vec{N} \cdot \vec{y}}{|\vec{N}| |\vec{y}|} = \frac{(0, y, -D)(0, y, 0)}{y \sqrt{y^2 + D^2}} = \frac{y}{\sqrt{y^2 + D^2}} \quad (25)$$

Ainda pela Figura 9(b) obtém-se a relação de equilíbrio na posição deformada na direção da coordenada vertical \vec{y} .

$$|\vec{N}| = N = \frac{-P}{3 \cos(\alpha)} \quad (26)$$

Considerando-se a Lei de Hooke (que relaciona proporcionalmente a medida de deformação de Biot com a tensão nominal) como relação constitutiva, tem-se:

$$\sigma = E\varepsilon \quad (27)$$

$$\frac{N}{A_0} = E(\lambda - 1) \quad (28)$$

Substituindo-se as Eqs. (24) e (26) na Eq. (28) chega-se à equação que permite relacionar a configuração deformada (definida pelo ângulo β) com a força aplicada (P).

$$P = -3EA_0 \cos(\alpha) \left[\frac{\cos(\beta_0) - \cos(\beta)}{\cos(\beta)} \right] \quad (29)$$

Com uma planilha de cálculo é possível arbitrar posições (y) do ó central, calcular o ângulo β , definir a força vertical P necessária para atingir a configuração deformada e finalmente calcular as forças axiais (N) de equilíbrio interno nas barras da treliça.

Para efeitos de comparação da solução analítica com a solução numérica, foram adotados: $EA_0 = 133865$ kN, $H = 20$ cm e $D = 500$ cm. A Tabela 1 apresenta a comparação dos resultados analíticos com os encontrados pelo programa computacional desenvolvido.

Na Figura 11 são apresentadas as configurações inicial e deformada no Excel para a treliça 3D de três barras analisadas numericamente.

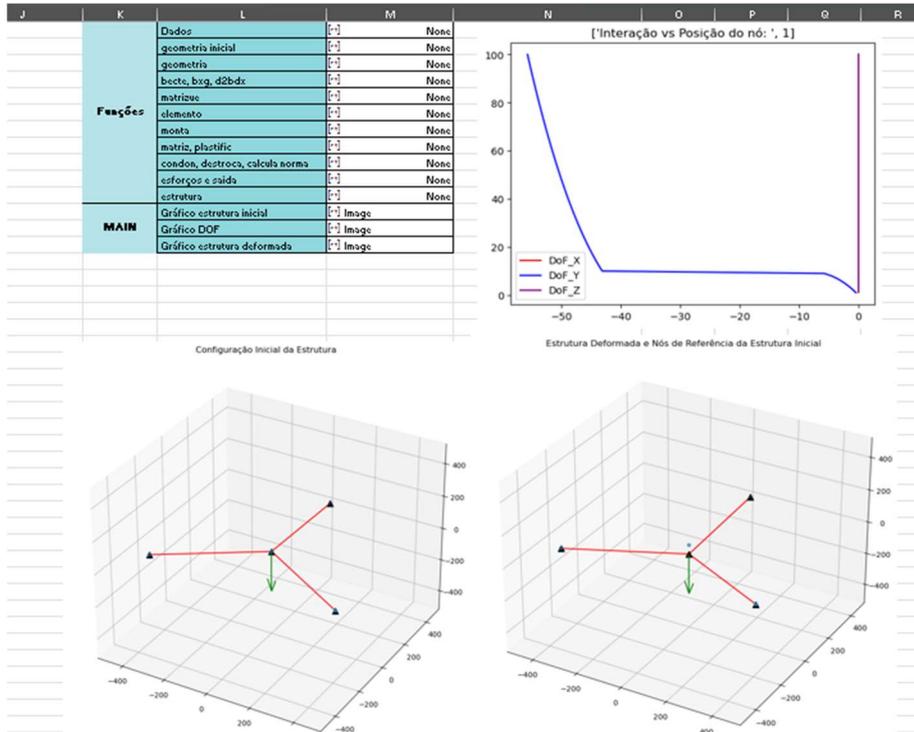
Tabela 1 – Comparação entre as forças axiais obtidas pelas soluções analítica e numérica

POSIÇÃO Y [cm]	FORÇA AXIAL ANALÍTICA [kN]	FORÇA AXIAL NUMÉRICA [kN]	ERRO RELATIVO [%]
20	0,00	0,00	-
10	-80,14	-80,26	0,14
0*	-106,84	-107,02	0,17
-10	-80,14	-80,26	0,14
-20	0,00	0,00	-
-30	133,78	133,66	0,10
-40	321,53	320,55	0,30

Fonte: Autores (2024)

* No caso analítico foi utilizada a posição de 0,001 nos cálculos para evitar singularidade.

Figura 11 – Resultados cinemáticos obtidos com o *script* implementado no Excel para a treliça 3D de três barras - [cm, kN]



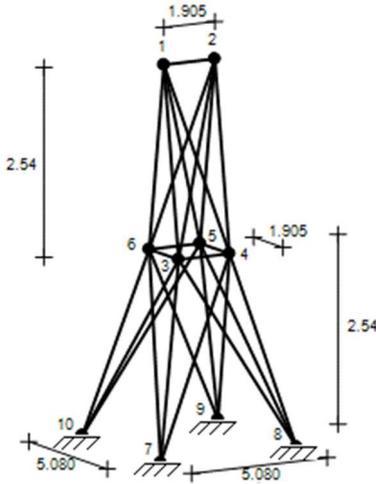
Fonte: Autores (2024)

Por fim, é importante citar o fenômeno do *snap-through* (salto de deslocamento), o qual ocorre quando uma estrutura passa por uma mudança súbita de configuração após atingir uma determinada carga crítica. No controle de força usado para rodar o exemplo não é possível captar o *snap-through*. Por isso, o deslocamento correspondente à posição na qual ocorre o *snap-through* não é registrado com precisão na Tabela 1 (posição 0*).

2. Torre treliçada

O exemplo trata da análise não linear de uma torre treliçada com 25 barras, apresentada na Figura 12, que é utilizada como referência em estudos de otimização estrutural (Hasançebi et al., 2013). A treliça é construída com barras de alumínio, que tem um módulo de elasticidade longitudinal $E = 68,9$ GPa. São utilizados os valores de área de seção transversal das barras apresentados em Peixoto (2024).

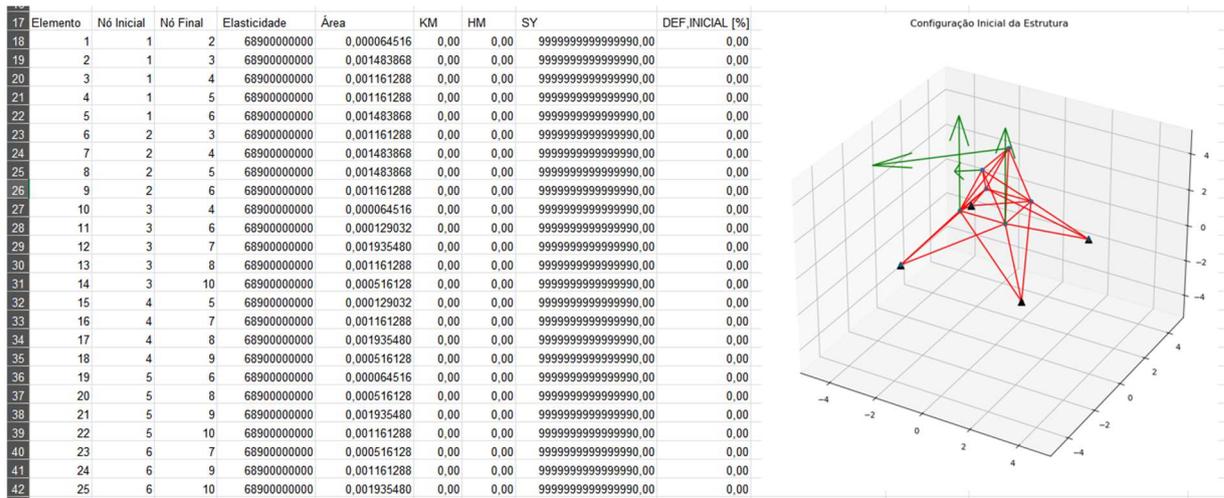
Figura 12 – Geometria da Torre treliçada com condições de contorno – [m]



Fonte: Adaptado de Peixoto (2024)

Na Figura 12, os nós 7, 8, 9 e 10 são bloqueados ao deslocamento nas direções X, Y e Z. Foram considerados 50 incrementos de força $\Delta P_x = \Delta P_y = \Delta P_z 4,45374$ kN para os nós 1 e 2 e $\Delta P_x = 2,22687$ kN para o nó 3 e $\Delta P_x = 2,67224$ kN para o nó 6. Na Figura 13 são apresentados os dados de entrada da torre treliçada.

Figura 13 – Dados de entrada da torre treliçada – [N; m]



Fonte: Autores (2024)

O resultado de saída referente às forças axiais da torre deformada para o último passo de força, em formato de planilha, é apresentado na Figura 14.

Figura 14 – Resultados numéricos obtidos com o *script* implementado no Excel para a torre treliçada – [kN]

Load Step:	50	
Element	Initial node	Final node
1	-69946,31	-69946,31
2	-225729,38	-225729,38
3	-260381,19	-260381,19
4	-5975,00	-5975,00
5	210531,47	210531,47
6	-252826,57	-252826,57
7	-107688,51	-107688,51
8	328458,92	328458,92
9	-11090,60	-11090,60
10	13345,71	13345,71
11	24221,57	24221,57
12	-317394,36	-317394,36
13	-47871,79	-47871,79
14	-59287,43	-59287,43
15	-521,30	-521,30
16	-47909,71	-47909,71
17	-314415,04	-314415,04
18	-74736,01	-74736,01
19	19828,91	19828,91
20	62062,64	62062,64
21	242422,61	242422,61
22	152313,77	152313,77
23	90404,99	90404,99
24	166609,94	166609,94
25	236609,09	236609,09

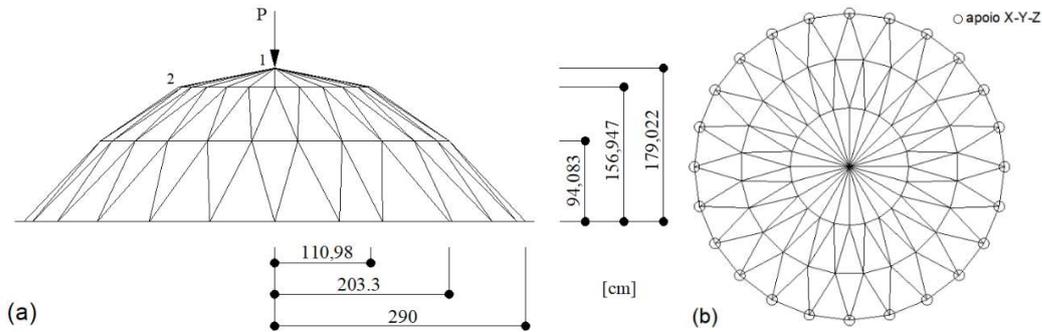
Gráfico **Forças Axiais** +

Fonte: Autores (2024)

3. Domo treliçado

O exemplo trata da análise não linear de um domo treliçado com 168 barras, apresentado na Figura 15. Todas as barras da estrutura possuem as mesmas seções, com módulo de rigidez axial $EA_0 = 10$ kN. Esse exemplo foi baseado no trabalho de Forde e Stierner (1987) e a validação numérica pode ser encontrada em Greco e Ferreira (2009).

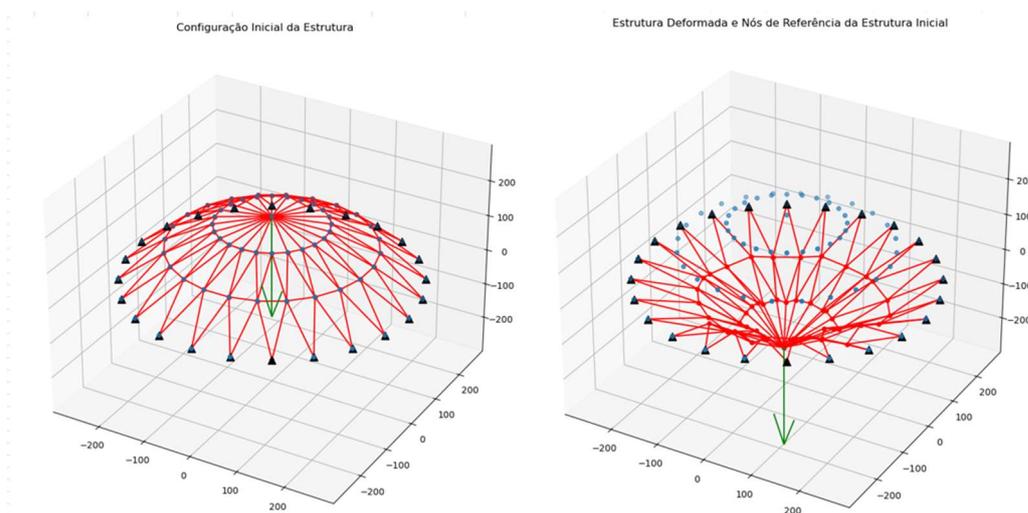
Figura 15 – Geometria e condições de contorno do Domo treliçado: (a) vista lateral; (b) vista superior – [cm]



Fonte: Adaptado de Greco e Ferreira (2009)

Na Figura 16 são apresentadas as configurações inicial e deformada da estrutura analisada.

Figura 16 – Resultados cinemáticos obtidos com o *script* implementado no Excel para o domo treliçada – [kN, cm]



Fonte: Autores (2024)

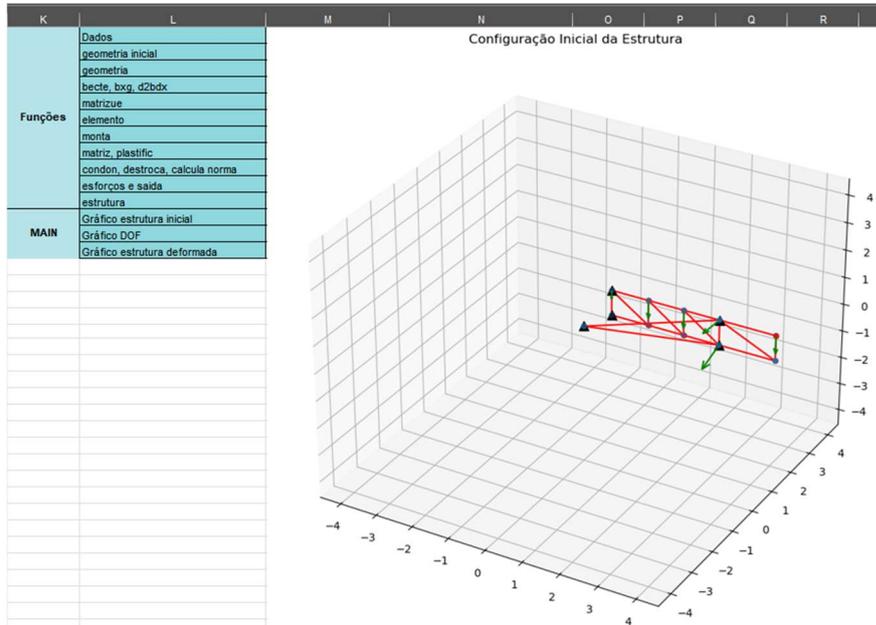
Para esse exemplo, ao utilizar um número elevado de passos de carga (como 500), o Microsoft Excel apresenta um erro de “Tempo limite”, indicando que o tempo padrão foi excedido para execução daquela célula. Assim, para valores elevados de passos de carga

aplicados a estruturas complexas, algumas análises podem não ser concluídas e será necessário alterar a configuração padrão do Excel, dificultando a realização imediata dos resultados.

4. Estrutura interna de uma semiasa de avião monomotor

O exemplo trata da análise de uma estrutura da semiasa de avião monomotor com 17 barras, apresentada na Figura 17. Esse exemplo foi baseado no livro de Bruhn (1973), que é uma das principais referências para o projeto de estruturas aeronáuticas até os dias atuais, auxiliando no entendimento e dimensionamento de componentes de aeronaves. A obra aborda de forma abrangente os princípios de análise estrutural de veículos aeronáuticos, combinando fundamentos teóricos sólidos com aplicações práticas. Juntamente com o livro de Niu (1999), Bruhn (1973) é ainda hoje o livro de referência para projeto de estruturas aeronáuticas mais usado em empresas como Embraer e Boeing. O exemplo trata o problema como linear e transfere os efeitos da sustentação dos dois montantes para a estrutura treliçada da semiasa. As unidades do problema foram convertidas do sistema inglês para o sistema internacional de unidades. Foram consideradas nas simulações tirantes de aço (barras 14 a 17) com módulos de rigidez axiais $EA_0 = 2068 \text{ kN}$ e demais elementos de alumínio (barras 1 a 13) com módulos de rigidez axiais $EA_0 = 5334 \text{ kN}$.

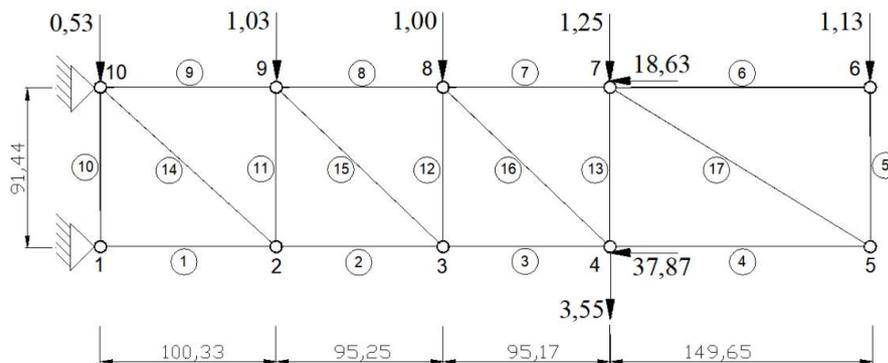
Figura 17 – Estrutura treliçada de uma semiasa de avião monomotor na configuração inicial – [kN, m]



Fonte: Autores (2024)

Foi simulada apenas a treliça plana, considerando-se as forças de sustentação e de arrasto e travando-se os nós perpendiculares ao plano da estrutura, conforme apresentado na Figura 18.

Figura 18 – Estrutura interna treliçada de uma semiasa de avião monomotor com dados geométricos, condições de contorno e condições iniciais [kN, cm]



Fonte: Autores (2024)

A Tabela 2 apresenta a comparação entre as forças axiais obtidas no *script* do Excel e os resultados de Bruhn (1973).

Tabela 2 – Comparação entre as forças axiais obtidas pelas soluções analítica e numérica

BARRA	FORÇA AXIAL ANALÍTICA [kN]	FORÇA AXIAL NUMÉRICA [kN]	ERRO RELATIVO [%]
1	-61,90	-61,80	0,16
2	-53,15	-53,08	0,13
3	-45,91	-45,87	0,07
4	-1,84	-1,84	0,05
5	-1,13	-1,13	0,01
6	0,00	0,00	-
7	-16,80	-16,80	0,00
8	-10,59	-10,63	0,32
9	-3,35	-3,42	2,13
10	0,00	0,00	-
11	-7,98	-7,96	0,26
12	-6,96	-6,93	0,34
13	-2,39	-2,38	0,37
14	11,84	11,83	0,07
15	10,04	10,01	0,33
16	8,60	8,59	0,18
17	2,16	2,16	0,06

Fonte: Autores (2024)

Esta seção apresentou quatro exemplos numéricos de aplicação com diferentes características geométricas: uma treliça espacial com três barras, uma torre, um domo e uma estrutura treliçada utilizada na semiasa de uma aeronave tipo monoplano.

A treliça de três barras serviu para validar a metodologia desenvolvida na seção anterior, comparando-se as respostas numéricas, obtidas pela ferramenta desenvolvida, com a solução analítica do problema. Nesse sentido, a treliça da semiasa de uma aeronave tipo monoplano também serviu para validação da metodologia. São duas estruturas mais simples que podem ser resolvidas manualmente. No caso da estrutura da semiasa, o cálculo manual é restrito à análise linear, com a comparação das forças de equilíbrio axiais dos elementos.

Os outros dois exemplos numéricos (torre e domo) são estudos de caso que ilustram aplicações com severo comportamento não linear geométrico nas quais o cálculo manual é inviável para efeitos de comparação. No entanto, são exemplos que servem para auxiliar no entendimento do que é o efeito não linear geométrico, que pode apresentar ganho ou perda de rigidez, a depender da configuração dos elementos estruturais e das condições iniciais (forças aplicadas) e de contorno do problema (apoios).

À medida que o número de nós na estrutura aumenta, observa-se um crescimento correspondente no tempo de processamento, como mostrado na Tabela 3. Para pequenas quantidades de elementos, essa variação é pouco significativa. No entanto, para estruturas mais complexas, como o domo treliçado, o tempo de processamento foi aproximadamente cinco vezes superior ao da semiasa, que possui 19 elementos. Contudo, o tempo de processamento para essa estrutura maior permaneceu em um nível aceitável, em torno de 27 segundos, considerando que o objetivo do estudo é a análise de estruturas mais simples.

Tabela 3 – Dados quanto ao número de iterações e tempo de processamento

EXEMPLO	NÚMERO MÁXIMO DE ITERAÇÕES	TEMPO DE PROCESSAMENTO [s]
Treliça 3D de três barras (3 elementos)	142	2,8748
Torre treliçada (25 elementos)	2	5,3115
Domo treliçado (168 elementos)	195	27,3602
Semiasa de avião treliçada (19 elementos)	2	5,4938

Fonte: Autores (2024).

Considerações finais

A partir da acurácia das respostas encontradas nos dois exemplos com comparações de resultados e das duas aplicações mais complexas, conclui-se que a metodologia utilizada atinge o objetivo do trabalho, visto que todas as simulações descreveram aquilo que é previsto pela teoria. Assim, foi desenvolvida com sucesso uma ferramenta computacional para análise não linear de treliças baseada na integração entre a linguagem Python e o programa Excel.

Durante os cursos de graduação em engenharia no Brasil, o tema da não linearidade é, geralmente, abordado de maneira introdutória. Pode-se afirmar que a fonte de não linearidade mais relacionada com a instabilidade estrutural e ao mesmo tempo com ganho de rigidez em sistemas estruturais tem tido muito pouca atenção nos currículos dos cursos de graduação. Os exemplos da estrutura interna de uma semiasa de avião monomotor e da torre treliçada são ilustrativos do ganho de rigidez estrutural sob efeito não linear geométrico. Por se tratar de um tema complexo, que pode ser associado a outras fontes de não linearidade, o tema acaba sendo objeto de estudo de cursos de pós-graduação. O exemplo do domo treliçado ilustra a complexidade da análise não linear geométrica em estruturas. Na área das engenharias aeroespacial e aeronáutica o tema da não linearidade geométrica é fundamental, mas os conteúdos amplos desses cursos acabam restringindo a oferta de disciplinas específicas sobre o tema. Outra dificuldade para lecionar o tema é que há poucas soluções analíticas disponíveis para tratar problemas com comportamento não linear geométrico, geralmente destinadas a resolver problemas muito simples, como o exemplo da treliça 3D simétrica com três barras que possui solução analítica.

Os autores consideram que o objetivo complementar de facilitar o entendimento do comportamento não linear geométrico por meio de ferramenta computacional também tenha sido atingido. Tanto para alunos de graduação quanto para alunos de pós-graduação, esse tipo de ferramenta faz-se importante. A exemplificação visual do comportamento estrutural facilita o entendimento e reforça a importância do entendimento de fenômenos complexos como o da não linearidade geométrica.

Em relação a implementações futuras no código, destacam-se os efeitos não lineares físicos oriundos de modelos de plastificação (Barbosa et al.; 2023) e oriundos de modelos de dano (Felipe et al., 2019). Outros efeitos, como os oriundos do acoplamento termo-estrutural (Carrazedo e Coda; 2010) e oriundos de modelos constitutivos hiperelásticos (Perônica, 2022), também são de interesse para aplicações em treliças não lineares modeladas pelo método dos elementos finitos posicional.

Agradecimentos

Os autores agradecem o apoio financeiro concedido pelo CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico).

Referências

- AGUIAR, A. R.; FOSDICK, R. L. A singular problem in incompressible nonlinear elastostatics. **Mathematical Models and Methods in Applied Sciences**, Toh Tuck Link, v. 10, n. 8, p. 1181-1207, 2000.
- BARBOSA, G. B.; FERNANDES, W. L.; GRECO, M.; PEIXOTO, D. H. P. Numerical modeling of lyapunov exponents for structural damage identification. **Buildings**, Basel, v. 13, n. 7, p. 1802, 2023.
- BOMFIM, D. S.; MARTHA, L. F.; PACCOLA, R. R. An educational computational program for nonlinear geometric analysis of truss structures using the Finite Element Method based on Positions. In: XLV IBERO-LATIN AMERICAN CONGRESS ON COMPUTATIONAL METHODS, 2024, Alagoas. **CILAMCE**. 2024. Disponível em: <https://www.researchgate.net/profile/Danilo-Bomfim-3/publication/387069426_An_educational_computational_program_for_nonlinear_geometric_analysis_of_truss_structures_using_the_Finite_Element_Method_based_on_Positions/links/675ed661e9427c77e914d56f/An-educational-computational-program-for-nonlinear-geometric-analysis-of-truss-structures-using-the-Finite-Element-Method-based-on-Positions.pdf>. Acesso em: 19 mar. 2025.
- BONET, J.; WOOD, R. D.; MAHANEY, J.; Heywood, P. Finite element analysis of air supported membrane structures. **Computer Methods in Applied Mechanics and Engineering**, Amsterdam, v. 190, n. 5, p. 579-595, 2000.
- BRUHN, E. F. **Analysis and design of flight vehicles structures**. Cincinnati, Tri-State Offset Company, 1973.
- CARRAZEDO, R.; CODA, H. B. Alternative positional FEM applied to thermomechanical impact of truss structures. **Finite Elements in Analysis and Design**, Amsterdam, v. 46, n. 11, p. 1008-1016, 2010.
- CODA, H. B. An exact FEM geometric non-linear analysis of frames based on position description. In: XVIII BRAZILIAN CONGRESS OF MECHANICAL ENGINEERING, 2003, São Paulo. **COBEM**. 2003. Disponível em: <<https://repositorio.usp.br/item/001372218>>. Acesso em: 23 nov. 2024.
- CODA, H. B. **O método dos elementos finitos posicional: sólidos e estruturas – não linearidade geométrica e dinâmica**. EESC/USP, São Carlos, 2018.
- DRIEMEIER, L.; PROENÇA, S. P. B.; ALVES, M. A contribution to the numerical nonlinear analysis of three-dimensional truss systems considering large strains, damage and plasticity. **Communications in Nonlinear Science and Numerical Simulation**, Amsterdam, v. 10, n. 5, p. 515-535, 2005.

FELIPE, T. R. C.; LEONEL, E. D.; HAACH, V.G.; BECK, A. T. A comprehensive ductile damage model for 3D truss structures. **International Journal of Non-Linear Mechanics**, Amsterdam, v. 112, p. 13-24, 2019.

FORDE, B. W. R.; STIEMER, S. F. Improved arc-length orthogonality methods for nonlinear finite element analysis. **Computers & Structures**, Amsterdam, v. 27, n. 5, p. 625–630, 1987.

GRECO, M.; PEIXOTO, D. H. N. Comparative assessments of strain measures for nonlinear analysis of truss structures at large deformations. **Engineering Computations**, Leeds, v. 39, n. 5, p. 1621-1641, 2021.

GRECO, M.; FERREIRA, I. P. Logarithmic strain measure applied to the nonlinear positional formulation for space truss analysis. **Finite Elements in Analysis and Design**, Amsterdam, v.45, p. 632-639, 2009.

GRECO, M.; GESUALDO, F. A. R.; VENTURINI, W. S.; CODA, H. B. Nonlinear positional formulation for space truss analysis. **Finite Elements in Analysis and Design**, Amsterdam, v. 42, n. 12, p. 1079-1086, 2006.

HASANÇEBI, O.; TEKE, T.; PEKCAN, O. A bat-inspired algorithm for structural optimization. **Computers & Structures**, Amsterdam, v. 128, p. 77–90, 2013.

MACHADO, G. A.; GRECO, M.; BECHO, J. S. Ferramenta computacional voltada ao ensino de análise não linear geométrica de estruturas treliçadas. In: 52º COBENGE - CONGRESSO BRASILEIRO DE EDUCAÇÃO EM ENGENHARIA, 2024, Vitória. Anais do COBENGE 2024. **COBENGE**. 2024. Associação Brasileira de Educação em Engenharia (ABENGE), 2024. v. 1. p. 1-13. Disponível em: <<http://dx.doi.org/10.37702/2175-957X.COBENGE.2024.4933>>. Acesso em: 23 nov. 2024.

MACHADO, G.A. **Ferramenta computacional voltada ao ensino de análise não linear geométrica de estruturas treliçadas**. Disponível em: <<https://youtu.be/p2E8WYFuoJk>>. Acesso em: 23 de nov. 2024.

MACHADO, G. A. **Desenvolvimento de software livre para análise de treliças não lineares: Python Positional Truss 3D**. Iniciação científica (Graduação em Engenharia Aeroespacial). Universidade Federal de Minas Gerais, Belo Horizonte, 2023.

MICROSOFT CORPORATION. **Introdução ao Python no Excel**. Disponível em: <<https://support.microsoft.com/pt-br/office/introdu%C3%A7%C3%A3o-ao-python-no-excel-55643c2e-ff56-4168-b1ce-9428c8308545>>. Acesso em: 22 nov. 2024.

NIU, M. C. **Airframe stress analysis and sizing**. 2.ed. Califórnia, Hong Kong Conmilit Press, 1999.

OLIVEIRA, R. L. **Elaboração de algoritmo com formulação não linear geométrica para cálculo de treliças tridimensionais.** Trabalho de Conclusão de Curso (Graduação em Engenharia Civil). Universidade Tecnológica Federal do Paraná, Campo Mourão, 2015.

PEIXOTO, D. H. N. **Estudo sobre diferentes medidas de deformação e apresentação da família senohiperbólica generalizada, com aplicação na modelagem de polímeros, biomateriais e tecidos biológicos moles.** Tese (Doutorado em Engenharia de Estruturas). Universidade Federal de Minas Gerais, Belo Horizonte, 2024.

PERÔNICA, D. S. **Análise não linear de treliças considerando modelos hiperelásticos pelo método dos elementos finitos posicional.** 2022. Dissertação (Mestrado em Engenharia Civil). Universidade Federal do Rio Grande do Norte, Natal, 2022.

PYTHON SOFTWARE FOUNDATION. **Integrated Development and Learning Environment (IDLE).** Disponível em: <<https://docs.python.org/3/library/idle.html>>. Acesso em: 23 nov. 2024.

QURBANI, M. **Python simulation of heating and cooling of extruded tubes using Excel as interface.** 2024. Degree Project in Technology (PhD Thesis) - KTH Royal Institute of Technology, Stockholm, 2024. Disponível em: <<https://www.researchgate.net/publication/387724223>>. Acesso em: 23 mar. 2025.

SOUZA, L. A. F. de. Implementação computacional das formulações posicional e corrotacional de elementos finitos para treliças 2D. **Revista de Engenharia e Tecnologia**, Ponta Grossa v. 11, n. 4, 2019.

SHABANA, A. A. Definition of the slopes and the finite element absolute nodal coordinate formulation. **Multibody System Dynamics**, Berlin, v. 1, n. 3, p. 339-348, 1997.