IV Seminário Sul Brasileiro de Pontes e Estruturas
IV SSBPE 2024

DESENVOLVIMENTO DE UMA ROTINA DE ANÁLISE MATRICIAL DE TRELIÇAS ESPACIAIS PELO MÉTODO DA RIGIDEZ

Fernanda Segaspini Golombieski[1], Bruna Manica Lazzari[2]

[1] **Pontifícia Universidade Católica do Rio Grande do Sul / Escola Politécnica / fsegaspinigolombieski@gmail.com**

[2] **Pontifícia Universidade Católica do Rio Grande do Sul / Escola Politécnica / bruna.lazzari@pucrs.br**

Resumo

Apesar da maioria das análises de estruturas serem realizadas em *softwares* comerciais, há uma necessidade de se realizar a comparação dos resultados obtidos pelos programas com valores de referência. Muitas vezes estes são calculados a partir de métodos manuais no *Excel*. Partindo destes métodos manuais de análise matricial de treliças tridimensionais foi desenvolvido um código em *Python*. Através deste foram realizadas análises em três modelos de treliças de diferentes complexidades. Os resultados foram comparados a *softwares* comerciais e o código apresenta resultados similares, mostrando-se como uma alternativa aos métodos utilizados atualmente.

Palavras-chave

Treliça tridimensional, análise matricial, método dos deslocamentos, Pyhton.

Introdução

Nos projetos estruturais, busca-se realizar análises de maneira rápida, eficiente e segura. No geral, o objetivo destas análises é determinar os deslocamentos, forças externas e internas, deformações e tensões sob diversas circunstâncias de carregamento, vinculação, temperatura e tipos de materiais. É durante esta etapa do projeto, que é possível verificar se os elementos estruturais que estão sendo submetidos a esses esforços atingem ou não algum estado limite normatizado, seja ele último ou de serviço.

Atualmente, através de *softwares* comerciais, já é possível simular diferentes condições de carregamento e criar modelos estruturais como pórticos espaciais, pontes, passarelas, coberturas, torres de transmissão, entre outros. Ainda assim, a tarefa de interpretar, compreender e comparar resultados com rotinas criadas e desenvolvidas nos escritórios de projetos é de extrema relevância. Apesar de ser possível realizar o lançamento de dados com precisão nestes *softwares* comerciais, ter o domínio de um programa próprio e poder, a partir dele, explorar várias opções de dimensionamento, pode garantir um maior controle ao calculista estrutural. Além de tudo, uma ferramenta assim pode tornar a rotina do projetista mais rápida e produtiva, gerando mais segurança nas interpretações e tomadas de decisões e maior produtividade na hora de dimensionar as estruturas.

A Análise Matricial de Estruturas consiste na adaptação do Método dos Deslocamentos adequada para o cálculo de estruturas unidas por nós (reticuladas) de maneira computacional. No Método dos Deslocamentos são adotadas como incógnitas os deslocamentos que ocorrem nos nós da estrutura. Os esforços nas extremidades das barras são determinados em função dos deslocamentos e, mediante imposição de condições de equilíbrio, obtém-se um conjunto de equações algébricas que são organizadas na forma matricial. Resolvendo-se esse sistema de equações obtêm-se os deslocamentos e, a partir deles, os esforços nas barras.

O objetivo deste trabalho foi programar uma rotina computacional utilizando a linguagem de programação P*ython* por meio da Análise Matricial para a análise das reações externas, solicitações internas e deslocamentos em treliças espaciais. Foi analisado o impacto de diferentes carregamentos em um modelo de uma torre de transmissão. Os resultados obtidos foram comparados com os valores simulados através de outros *softwares* já utilizados na Engenharia, como o *Robot Structural Analysis*.

Referencial Teórico

O desenvolvimento desta rotina iniciou pela análise de um modelo de pequeno porte, a fim de validar o funcionamento do mesmo através da Análise Matricial no *software* Excel. Após isto, o programa desenvolvido foi utilizado para a análise de um modelo mais complexo: uma torre de transmissão. Os itens a seguir abordam as referências necessárias para o desenvolvimento deste trabalho, com base em Nunes e Vieira (2019), Pappalardo Jr. e Agnelo (2014), e Borges *et al*. (2016),.

Análise Matricial de Estruturas

A Análise Matricial nada mais é do que uma adaptação do Método dos Deslocamentos, também conhecido como Método da Rigidez de Estruturas, para a realização da análise de estruturas de modo computacional. Consiste em determinar os esforços internos, reações externas e os deslocamentos que foram gerados após a aplicação de carregamentos externos. Após esta análise, através dos resultados obtidos, é iniciado o dimensionamento da estrutura.

Método dos Deslocamentos

O método dos deslocamentos, também conhecido como método da rigidez, é um método de análise de estruturas hiperestáticas reticuladas que usa a rigidez dos elementos para formar um sistema de equações, relacionando os deslocamentos com as cargas que atuam na estrutura. Esses sistemas de equações são as matrizes de rigidez compostas pelos seus coeficientes de rigidez (K_{ij}), também chamados de graus de liberdade de cada nó, que nada mais são do que a ação (força) na direção i causada por um deslocamento unitário na direção j (enquanto todos os outros deslocamentos são impostos como nulos). O método consiste na construção das matrizes de rigidez global das barras, que relacionam os deslocamentos dos nós com as forças equivalentes sobre os mesmos. Através da aplicação das condições de contorno impostas pelos vínculos (sejam eles de primeira, segunda ou terceira ordem, anulando-se os deslocamentos correspondentes às direções restringidas), se obtém um sistema de equações cujas incógnitas são os deslocamentos nodais da estrutura. Para encontrar os valores dos deslocamentos nodais, multiplica-se a matriz inversa da matriz de rigidez pelo carregamento imposto nos nós previamente definidos. Encontrados os valores dos deslocamentos, é iniciado o cálculo das reações externas e solicitações internas de cada barra da treliça. Para encontrar os valores das reações externas, multiplica-se as matrizes de rigidez de cada barra pelos valores dos deslocamentos antes encontrados. Já as solicitações são encontradas através da montagem de uma nova matriz, chamada de matriz de transformação de cada barra que é obtida fazendo o equilíbrio dos esforços globais sobre os nós do elemento segundo a direção dos eixos do sistema local. Por ter um roteiro de metodologia bem definido, o Método dos Deslocamentos pode ser facilmente implementado num programa de computador. Neste estudo, a rotina das análises apresentadas na sequência foi desenvolvida através de uma linguagem de programação *Python*.

A metodologia utilizada para aplicação do Método dos Deslocamentos é apresentada, de forma resumida, a seguir:

a) Identificação Estrutural que consiste em agrupar e reunir todas as informações pertinentes para a análise, como coordenadas nodais, conectividades entre os nós, propriedades físicas e geométricas das barras, restrições nodais, cargas sobre os nós, graus de liberdade, quantidade de nós e barras e condições de contorno (apoios externos);

b) Montagem das matrizes de rigidez em coordenadas locais e globais de cada barra da estrutura para então montar a matriz de rigidez da estrutura em coordenadas globais;
c) Cálculo das forças nodais equivalentes;
d) Montagem e resolução do sistema de equações de equilíbrio;
e) Cálculo das reações nos apoios externos após obtidos os deslocamentos da estrutura.

Identificação Estrutural

Para que a análise exista, a estrutura precisa ser definida. Nesta etapa é feito o lançamento da estrutura e de suas características, como as coordenadas nodais, que estabelecem o posicionamento dos nós das barras dentro de um plano cartesiano tridimensional segundo os eixos X, Y e Z; a conectividade entre os nós, que através das ligações entre si formam as barras da treliça, devendo ser determinadas as posições do nó inicial e do nó final, além das propriedades físicas e geométricas de cada barra, onde são definidas a área transversal e o valor do módulo de elasticidade do material.
Nesta etapa são definidas também as restrições nodais, que indicam os apoios vinculados à cada um dos nós, podendo ser eles de primeira, segunda ou terceira ordem, os quais restringem movimentos em certos planos; são determinadas as condições de contorno, tornando nulos os deslocamentos nodais nas direções de restrição de movimento; as cargas, que no caso de treliças espaciais incidem somente sobre os nós; e por último, os graus de liberdade, que em função da natureza tridimensional da treliça espacial, pode-se definir que cada nó possui três graus de liberdade.

Cálculo da Matriz de Rigidez Global de cada Barra (K_{ij})

A matriz de rigidez global de cada barra, para uma estrutura tridimensional, é uma matriz 6x6, dado que cada um dos nós possui três graus de liberdade. As três primeiras linhas e colunas se referem ao nó inicial e a três últimas se referem ao nó final. A partir da localização global dos nós de cada barra, foram calculados os cossenos formados entre as barras da estrutura e o sistema de coordenadas global de modo que foi possível construir a Matriz de Rigidez Global de cada barra da estrutura (K_{ij}).
A Matriz de Rigidez foi montada a partir do modelo genérico, dado a seguir:

$$K_{ij} = \frac{EA}{L}\begin{bmatrix} \cos^2\theta x & \cos\theta x\ \cos\theta y & \cos\theta x\ \cos\theta z & -\cos^2\theta x & -\cos\theta x\ \cos\theta y & -\cos\theta x\ \cos\theta z \\ \cos\theta y\ \cos\theta x & \cos^2\theta y & \cos\theta y\ \cos\theta z & -\cos\theta y\ \cos\theta x & -\cos^2\theta y & -\cos\theta y\ \cos\theta z \\ \cos\theta z\ \cos\theta x & \cos\theta z\ \cos\theta y & \cos^2\theta z & -\cos\theta z\ \cos\theta x & -\cos\theta z\ \cos\theta y & -\cos^2\theta z \\ -\cos^2\theta x & -\cos\theta x\ \cos\theta y & -\cos\theta x\ \cos\theta z & \cos^2\theta x & \cos\theta x\ \cos\theta y & \cos\theta x\ \cos\theta z \\ -\cos\theta y\ \cos\theta x & -\cos^2\theta y & -\cos\theta y\ \cos\theta z & \cos\theta y\ \cos\theta x & \cos^2\theta y & \cos\theta y\ \cos\theta z \\ -\cos\theta z\ \cos\theta x & -\cos\theta z\ \cos\theta y & -\cos^2\theta z & \cos\theta z\ \cos\theta x & \cos\theta z\ \cos\theta y & \cos^2\theta z \end{bmatrix}$$

(Equação 1)

Onde:
E = Módulo de elasticidade do aço;
A = Área da seção transversal da barra;
L = Comprimento da barra;
$\theta x =$ ângulo de rotação realizado entre os eixos x, local e global;
$\theta y =$ ângulo de rotação realizado entre os eixos y, local e global;
$\theta z =$ ângulo de rotação realizado entre os eixos z, local e global.
Para a realização do cálculo dos ângulos mencionados acima, subtraiu-se os valores das coordenadas X, Y e Z do nó final de cada barra pelas coordenadas do nó inicial da mesma barra. E o resultado obtido pela subtração foi dividido pelo comprimento da respectiva barra. Conforme as equações abaixo:

$$cx = \cos\theta x = \frac{x_{final} - x_{inicial}}{L}$$

(Equação 2)

$$cy = \cos\theta y = \frac{y_{final} - y_{inicial}}{L}$$

(Equação 3)

$$cz = \cos\theta z = \frac{z_{final} - z_{inicial}}{L}$$

(Equação 4)

Já para encontrar o valor do comprimento da barra, indicado por L, segue equação:

$$L = \sqrt{(x_{final} - x_{inicial})^2 + (y_{final} - y_{inicial})^2 + (z_{final} - z_{inicial})^2}$$

(Equação 5)

Montagem da Matriz de Rigidez da Estrutura (K)
Mediante o cálculo da Matriz de Rigidez Global de cada barra (K_{ij}), se constrói a Matriz de Rigidez da estrutura (K) a partir de uma sobreposição das matrizes de cada barra, somando os elementos que apresentam os mesmos índices. O tamanho desta matriz depende do tamanho da estrutura, ou seja, os números de nós que esta apresenta.

Equação de Equilíbrio da Estrutura
A Equação de Equilíbrio da Estrutura nada mais é do que a multiplicação da Matriz de Rigidez da estrutura (K) pelos deslocamentos (U) em X, Y e Z de cada um dos nós que resulta no carregamento imposto nos nós (F). Esse carregamento já é determinado na estruturação da treliça e as incógnitas para se encontrarem nesta equação são os deslocamentos.

Segue a equação:

$$K * U = F$$

(Equação 6)

Em que:
K = matriz de rigidez da estrutura;
U = matriz coluna dos deslocamentos nodais em X, Y e Z;
F = matriz coluna dos carregamentos nodais em X, Y e Z.

Condições de Contorno
As condições de contorno são definidas através das vinculações que cada nó apresenta. Mediante eles, pode-se identificar se os deslocamentos são nulos e em qual direção dentro do plano cartesiano.
Neste momento altera-se a matriz de rigidez da estrutura (K), zerando os valores dos vetores dos nós nos sentidos em que os mesmos são nulos e aplica-se um valor unitário na diagonal principal. Mediante isso, gera-se a matriz inversa (K^{-1}) da matriz K para que se possa encontrar o valor dos deslocamentos nodais que não possuem vínculos que impendem o movimento.

Cálculo dos Deslocamentos Nodais (U)
Para encontrar os deslocamentos nodais (U), conforme descrito nos itens 2.2.4 e 2.2.5, tem-se a necessidade de se multiplicar a matriz inversa da matriz de rigidez (K^{-1}) da estrutura pelo carregamento dos nós (F).
A equação é dada por:

$$K^{-1} * F = U$$

(Equação 7)

Dado que:
K^{-1} = matriz inversa da matriz de rigidez da estrutura com as condições de contorno aplicadas;
U = matriz coluna dos deslocamentos nodais em X, Y e Z;
F = matriz coluna dos carregamentos nodais em X, Y e Z.

Cálculo das Reações Externas e Solicitações Internas

As reações externas são os esforços que os vínculos desenvolvem para manter a estrutura em equilíbrio. Já as solicitações internas são geradas a partir da deformação interna da estrutura quando esta for submetida ao carregamento. As reações e as solicitações mantêm o equilíbrio da estrutura.

Como as estruturas estudadas são caracterizadas como treliças, sendo todas as suas barras rotuladas, pode-se afirmar que nestes nós não ocorre absorção de momentos fletores, ou seja, estas barras sofrem apenas esforço normal. A existência de esforço cortante é descartada pois as barras não são carregadas ao longo de seu eixo, e tem nas suas extremidades momentos nulos.

Cálculo das Reações Externas – Forças

Para calcular as reações externas, multiplica-se a matriz de rigidez global de cada barra pela matriz coluna dos deslocamentos encontrados referentes à cada barra. O nó ao qual foi atribuído uma restrição ao movimento (vínculo), tem seu deslocamento zerado dentro da matriz coluna dos deslocamentos. Nos nós que apresentam vínculos externos, as forças resultantes nos nós de cada barra que chega neste mesmo ponto são somadas para a determinação da reação externa resultante.

A equação é apresentada abaixo como:

$$K_{ij} * U_{ij} = F_{ij}$$

(Equação 8)

Sabendo-se que:

K_{ij} = matriz de rigidez global de cada barra;

U_{ij} = matriz coluna dos deslocamentos referentes à cada barra;

F_{ij}= matriz coluna das forças externas do sistema global para cada barra.

Cálculo das Solicitações Internas

As solicitações internas são obtidas pela multiplicação da matriz transformação de translação espacial (T) de cada uma das barras da treliça pela matriz coluna das forças externas, encontradas na etapa anterior. Utiliza-se o versor local na direção y, apresentado na equação abaixo:

$$Lxy = \sqrt{\cos^2 \theta x + \cos^2 \theta y}$$

(Equação 9)

Sendo que:

Lxy = versor local na direção y;

$\theta x =$ ângulo de rotação realizado entre os eixos x, local e global;

$\theta y =$ ângulo de rotação realizado entre os eixos y, local e global.

A matriz de transformação, é uma matriz 6x6, e é utilizada para transportar a matriz coluna das forças externas do sistema de eixos local para o sistema de eixos global. Esta matriz é dada por:

$$T = \begin{bmatrix} \begin{matrix} \cos\theta x & \cos\theta y & \cos\theta z \\ -\dfrac{\cos\theta y}{Lxy} & \dfrac{\cos\theta x}{Lxy} & 0 \\ -\dfrac{\cos\theta x * \cos\theta z}{Lxy} & -\dfrac{\cos\theta y * \cos\theta z}{Lxy} & Lxy \end{matrix} & 0 \\ 0 & \begin{matrix} \cos\theta x & \cos\theta y & \cos\theta z \\ -\dfrac{\cos\theta y}{Lxy} & \dfrac{\cos\theta x}{Lxy} & 0 \\ -\dfrac{\cos\theta x * \cos\theta z}{Lxy} & -\dfrac{\cos\theta y * \cos\theta z}{Lxy} & Lxy \end{matrix} \end{bmatrix}$$

(Equação 10)

A equação dada a seguir corresponde ao cálculo das solicitações internas:

$$T * F_{ij} = F^{l}$$

(Equação 11)

T = matriz transformação de translação espacial;

F_{ij} = matriz coluna das forças externas do sistema global para cada barra;
F^l = matriz coluna das solicitações internas do sistema local para cada barra.

Python

Baseado em Severgnini (2022), Parisotto (2018), Silva (2021) e Titello (2020), foi escolhida a linguagem de programação gratuita *Python 3.12* por conta, principalmente, da facilidade da implantação de um código relativamente simples para esta análise de treliça e por ser uma linguagem apropriada para a manipulação de dados. Além disso, as bibliotecas disponíveis possuem muitos pacotes específicos para tarefas de engenharia, como cálculos numéricos, análises estatísticas, visualização de dados, simulações etc. Neste trabalho, para a manipulação de matrizes, foi utilizada a biblioteca *Numpy*, e para cálculos matemáticos a biblioteca *Math*.

O *Python* também é vastamente utilizado em trabalhos acadêmicos de engenharia, como avaliação da confiabilidade de vigas em concreto armado, dimensionamentos de estruturas aporticadas e treliçadas, averiguação de cargas de ruptura de placas e lajes em flexão, para a determinação da capacidade resistente de perfis laminados e soldados e aço, dentre tantas outras infinitas possibilidades. Por conta dessa ilimitada capacidade, tem-se a condição de se analisar, como no caso deste trabalho, qualquer modelo de treliça tridimensional, favorecendo o uso do código e facilitando as rotinas dentro dos escritórios de projeto.

Além do mais, o *Python* é uma linguagem aberta e gratuita, viabilizando o acesso à esta ferramenta. Ele pode ser facilmente integrado a outras ferramentas e *softwares* utilizados na engenharia, como *CAD, CAE, SAP2000, ANSYS* etc. Isso permite a automação de tarefas repetitivas e o desenvolvimento de fluxos de trabalho mais eficientes. É executado em diversos sistemas operacionais, como *Windows, MacOS e Linux*.

A aprendizagem desta linguagem de programação fez parte do estudo desenvolvido neste trabalho, sendo utilizada pela necessidade de criar uma rotina com a possibilidade de simular estruturas treliçadas espaciais de qualquer dimensão, incluindo diferentes números de barras e de nós, com diferentes cargas aplicadas, como rotina básica para desenvolvimento de um futuro *software* completo de dimensionamento e verificação de estruturas de barras.

Detalhamento do Código na Linguagem *Python*[1]

A rotina desenvolvida neste estudo foi organizada em classes, as quais representam conjuntos de comportamentos para todos os objetos necessários à montagem da Análise Matricial de Treliça Espacial. Essas classes foram nomeadas em *point*, *line*, *line_matrix*, *equilibrium_matrix* e *rotation_matrix*. Além disso, foi desenvolvida também a função *main*, a qual executa as ordens do programa. O desenvolvimento das classes foi ordenado concordando com a necessidade e vinculação de uma classe à outra. As classes e funções do código podem ser divididas da seguinte forma:

Point

Nesta classe são apresentados os objetos "pontos", os quais representam os nós das treliças. Eles são compostos por três coordenadas, x, y e z; por suas conexões com outros pontos; por coordenadas fixas, que representam o tipo de vínculo que restringe o movimento referente à direção de cada coordenada; e pelas cargas aplicadas na direção de cada coordenada. Esta classe é utilizada como a base de dados a partir da qual será realizado o processamento.

[1] O código pode ser acessado a partir da página https://github.com/FernandaSegaspiniGolombieski/treliss-matrix. Para acessá-lo, é necessário criar uma conta no site e pedir permissão de acesso à autora Fernanda Segaspini Golombieski.

Line
A classe *Line* representa os objetos "linhas" e na treliça são as barras. As linhas são geradas a partir da conexão entre dois objetos da classe *Point* (nós). Nesta classe tem-se as informações da seção transversal, do comprimento, e a propriedade do material (módulo de elasticidade). Esta classe também é utilizada como base de dados para o processamento do código. A classe também possui três métodos:

- *calculate_length*: realiza o cálculo do comprimento da linha (barra);
- *calculate_cosines*: realiza o cálculo dos cossenos referentes à cada plano;
- *get_material_property*: calcula a propriedade do material da barra a partir dos dados fornecidos.

Line_Matrix
Esta classe representa a matriz de rigidez de cada barra (linha), e possui a matriz coluna das forças externas geradas. Esta classe possui dois métodos:

- *generate_matrix*: calcula a matriz de rigidez através dos cossenos, dos comprimentos e da propriedade do material calculados na classe *Line;*
- *calculate_movement:* realiza o cálculo dos deslocamentos.

Equilibrium_Matrix
Na classe *Equilibrium_Matrix*, a matriz de rigidez da estrutura é representada. Possui também a matriz inversa que é calculada também na criação do objeto. Esta possui dois métodos, e são esses:

- *generate_matrix*: realiza a montagem da matriz de rigidez da estrutura, insere as condições de contorno, o que impede os movimentos nos nós de acordo com o vínculo.
- *calculate_value:* função auxiliar que calcula os valores para os campos da matriz a partir dos cossenos e da elasticidade.

Rotation_Matrix
Esta classe corresponde à matriz de rotação e possui a matriz coluna das solicitações internas. Detém dois métodos:

- *generate_matrix:* calcula a matriz de rotação de cada barra através de um versor e dos cossenos das linhas (barras) rotacionadas;
- *calculate_internal_solicitations:* realiza o cálculo da matriz coluna das solicitações internas de cada barra.

Main
Main é a função principal do código, e é a partir dela que se realiza o processamento. Primeiro, são inseridos todos os dados dos pontos em uma lista de objetos *Point*. Após isso, são geradas as linhas e colocadas em uma lista de objetos *Line*. As linhas são criadas a partir das conexões definidas nos pontos, e é também criada uma lista com as cargas dos pontos em sequência.
Para cada linha gerada é criada uma matriz da classe *Line_Matrix* e inserida em uma lista, juntamente com a adição de uma matriz de rotações em uma lista de objetos da classe *Rotation_Matrix*. São então geradas a matriz de equilíbrio, da classe *Equilibrium_Matrix* e calculados os deslocamentos a partir da matriz de equilíbrio invertida e a lista de cargas gerada anteriormente.
Por fim, é realizado um processamento para calcular as solicitações internas, onde são buscados os deslocamentos a partir dos pontos equivalentes, e calculadas as solicitações internas para cada matriz de rotação. Finalmente os resultados são impressos. Este método utiliza alguns métodos auxiliares para melhorar a compreensão do código:

- *generate_lines*: gera as linhas a partir de uma lista de pontos;
- *extract_loads*: extrai uma lista de cargas geral a partir das cargas de cada ponto;
- *calculate_internal_solicitations*: calcula as solicitações internas de cada matriz de rotação;
- *print_results*: imprime os resultados;
- *build_array*: cria lista de deslocamentos condizentes a cada ponto a partir de uma lista geral de movimentos.

Resultados

Para validação do código desenvolvido no *Python* foram comparadas as informações resultantes do funcionamento do código e os resultados dos *softwares Excel* e *Robot* para a torre de transmissão demonstrada na Figura 1. A análise é feita realizando a comparação dos deslocamentos e das solicitações internas geradas pelo código e os resultantes tanto do método de análise matricial por *Excel* quanto os dados resultantes do programa *Robot*.

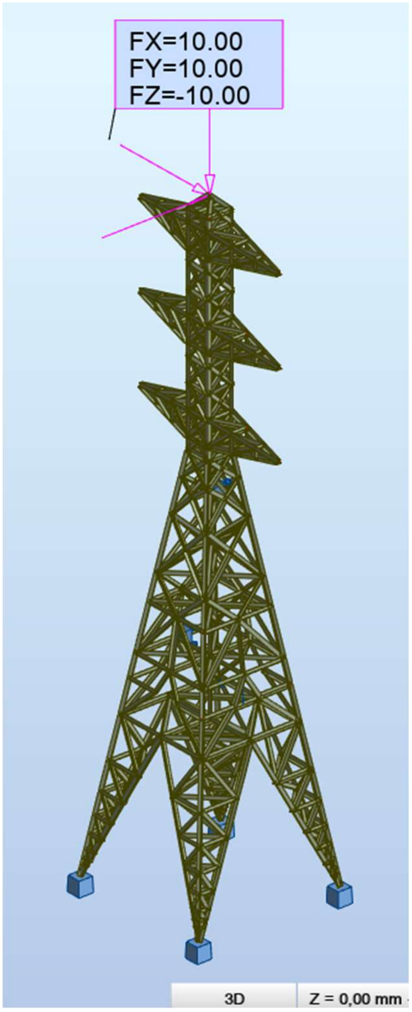

Figura 1 - Torre de Transmissão

A torre de transmissão analisada, de altura de 25,50m e base de 6x6 (Fig. 1) foi desenvolvida pela autora deste trabalho, com base em um modelo de torre de transmissão (Brametal, 2021). A torre é composta por 318 nós, 1025 barras metálicas de perfil L de dimensões 152mmx152mmx42mm de material aço carbono no padrão ASTM A36 possuindo módulo de elasticidade de 200 GPa e área da seção de 54,44cm². Não foi considerado o peso próprio da estrutura e a mesma foi submetida apenas ao carregamento no nó 3: 10kN em X, 10kN em Y e -10kN em Z. Possui engastes nos nós da base: 2, 73, 75 e 76. Dado que a estrutura apresentou instabilidades, ao ser analisada no *software Robot,* por não apresentar um projeto e dimensionamento prévios, em alguns nós e em diferentes direções, houve a necessidade de inserir apoios de primeira ordem nos nós 136 e 347 e segunda ordem no nó 143. Nas tabelas a seguir encontram-se dados selecionados gerados pelo *software Robot* e os dados correspondentes gerados pela rotina *Python*. Esta seleção é necessária pois uma análise exaustiva dos dados foge do escopo do presente trabalho.

Tabela 1 – Deslocamentos.

	Robot (cm)			*Python* (mm)		
Nó	X	Y	Z	X	Y	Z
1	0,1	0,2	0	1,14667	1,41753	0,04466
2	0	0	0	0	0	0
3	0,7	0,8	0	7,23117	7,82949	0,03262
4	0,7	0,7	0,1	6,84400	7,43554	0,80555
5	0,7	0,7	0	6,84170	7,43409	0,03042

Tabela 2 – Reações externas.

	Robot (kN)			*Python* (kN)		
Nó	X	Y	Z	X	Y	Z
2	0,39	-0,39	2,5	0,39976	-0,39976	2,55738
73	-7,52	-7,52	-48,11	-7,53454	-7,53454	-48,20041
75	2,52	-2,52	-16,12	2,52853	-2,53239	-16,20036
76	-5,39	-5,39	34,51	-5,39001	-5,39823	34,5339

Tabela 3 – Solicitações internas.

Barra	Nó	*Robot* (kN)	*Python* (kN)
2	3	-5,95	-5,94933
2	4	-5,95	-5,94933
15	21	41,64	41,67124
15	2	41,64	41,67124
244	142	-20,61	-20,67148
244	115	-20,61	-20,67148
763	84	-85,47	-85,40599
763	46	-85,47	-85,40599
1645	293	10,09	10,2939
1645	64	10,09	10,2939

Ao comparar os resultados obtidos pela rotina em *Python* com o software *Robot*, observou-se que, nos deslocamentos, a diferença percentual variou entre 0,03% e 29%. Nas reações externas, as discrepâncias foram menores, com percentuais entre 0,0002% e 2,5%. Já nas solicitações internas, a variação ficou entre 0,07% e 2,02%. Pode-se verificar que em todos os casos as soluções de ambos os *softwares* encontram resultados aproximados, o que atesta o funcionamento da rotina gerada no presente trabalho. Estas aproximações, ao invés de resultados iguais, podem ser atribuídos aos diferentes métodos de arredondamento utilizados por cada software, que criam, ao longo de sucessivas operações, estas pequenas diferenças.

Conclusão

Pôde-se ver que o código gera resultados compatíveis com outros *softwares* de análise de estruturas, demonstrando a efetividade do método de análise matricial de treliças e a possibilidade de automação do mesmo em código. Assim, é possível utilizar o código como uma ferramenta de auxílio e validação na hora do dimensionamento de estruturas e comparação de resultados com *softwares* existentes atualmente.

Uma das limitações deste trabalho foi a não possiblidade de utilizar projeto estrutural prévio. Não foi possível equiparar os valores quando adicionados peso próprio da estrutura ou cargas de vento incididas na mesma.

Para a validação do código com projetos reais e a elaboração de métodos mais fáceis de inserção de dados, trabalhos ainda devem ser realizados, mas a robusteza da ferramenta em gerar os resultados corretos em vários cenários diferentes mostra que esta possui um grau de confiabilidade. Outra possibilidade é a integração com *softwares* de análise de estruturas. Há, portanto, amplo espaço para melhorias partindo dos resultados deste trabalho.

Referências

AGNELO, D. P.; PAPPALARDO JR., A. Desenvolvimento de Programa - Análise de Treliças espaciais. São Paulo: Faculdade de Engenharia de São Paulo, 2014.

BORGES, R. D. A.; SILVA, S. S. D.; BEZERRA, A. A. B. Estudo de Treliças Planas e Espaciais utilizando a Linguagem de Programação Python e o Software VTK. Brasília: Revista Interdisciplinar de Pesquisa em Engenharia, 2016.

BRAMETAL. Infraestrutura de Transmissões Elétricas - Torres e estruturas metálicas em aço galvanizado BRAMETAL. [S.l.]: [s.n.], 2021. Disponivel em: <https://www.brametal.com.br/wp-content/uploads/2021/12/brametal-transmissao-28p.pdf>. Acesso em: 21 jun. 2024.

NUNES, D. C.; VIEIRA, G. S. Aplicação do Método da Rigidez Direta para uma Treliça Espacial. Uberlândia: Universidade Federal de Uberlândia, 2019.

PARISOTTO, M. Elaboração de um Softwre para determinação via Método dos Elementos Finitos de Cargas de Ruptura de Placas e Lajes em Flexão. Porto Alegre: Universidade Federal do Rio Grande do Sul, 2018.

SEVERGNINI, J. A. B. Metalpy: Módulo Computacional para determinação da capacidade de resistente de perfis laminados e soldados de aço. Porto Alegre: Universidade Federal do Rio Grande do Sul, 2022.

SILVA, L. J. L. D. Aplicação de Métodos Heurísticos no Dimensionamento Otimizado de uma estrutura de aço aporticada com perfis de seção variável. Porto Alegre: Universidade Federal do Rio Grande do Sul, 2021.

TITELLO, E. P. Análise da confiabilidade de vigas em concreto armado reforçado com fibras de aço em relação aos esforços transversais. Porto Alegre: Universidade Federal do Rio Grande do Sul, 2020.

TORRES, F. C.; INOUE, H. Análise Estrutural e Dimensionamento de Linhas de Transmissão de Energia. [S.l.]: XII CBPE 2021, 2021.