

Truss Resolution Software

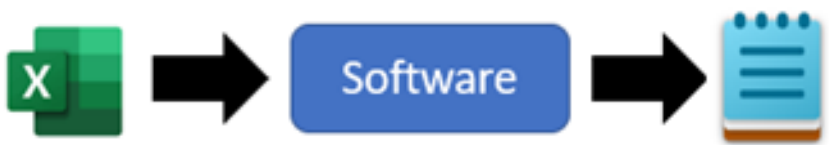
Introdução

Treliças são utilizadas em diversas estruturas presentes no nosso dia a dia, desde pontes e telhados, até torres de comunicação e satélites espaciais. O estudo dessas construções tão importantes para a sociedade evolui juntamente com os estudos de simulações computacionais, com essas o esforço matemático braçal, juntamente com a prevenção de erros e acidentes diminui drasticamente.

Com esses tipos de softwares é possível não só apenas simular forças aplicadas na treliça, como também descobrir qual o melhor material e área de seção transversal para a construção da estrutura. Sem dúvida estamos falando sobre um projeto de grande importância, portanto, para melhor compreender esse assunto, nós construímos um software com o intuito de analisar estruturas de treliças planas.

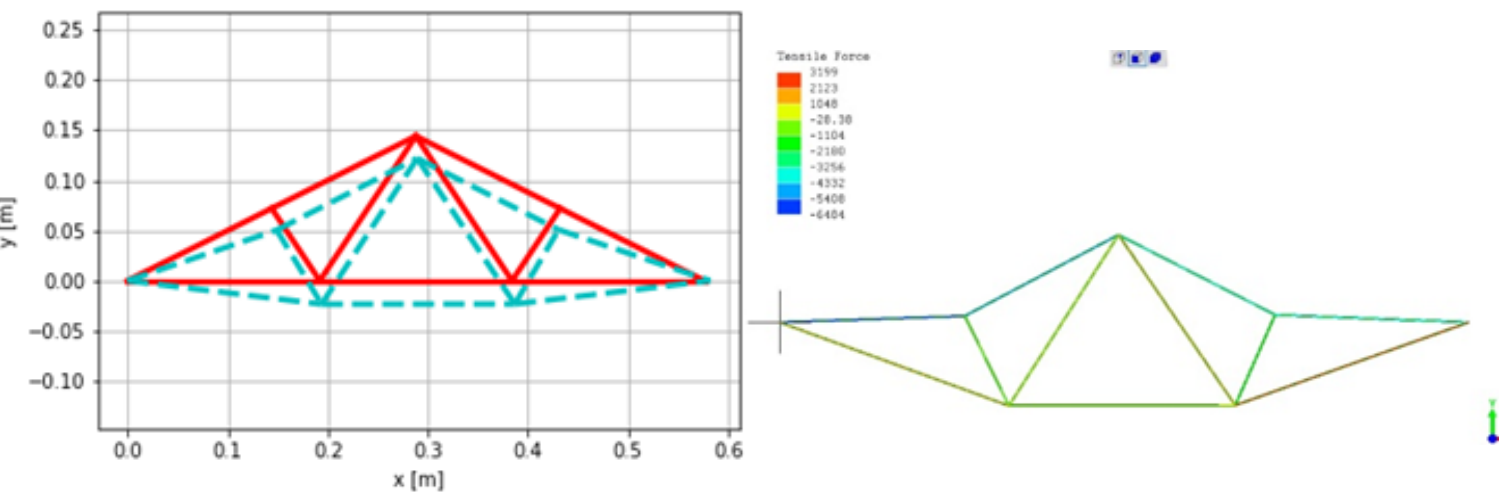
Estrutura do Software

O software fora construído em Python e utiliza técnicas numéricas para resolução de sistemas de equações, como o método de Jacobi. Fora criado uma classe Elemento que contém todos os atributos necessários para a realização dos procedimentos no cálculo da treliça. Fora também implementadas funções que realizam toda a parte algébrica do processo. O software foi desenhado para receber uma entrada em arquivo .xls e devolver uma saída com deslocamentos e reações na treliça além de gráficos que mostram os respectivos deslocamentos.

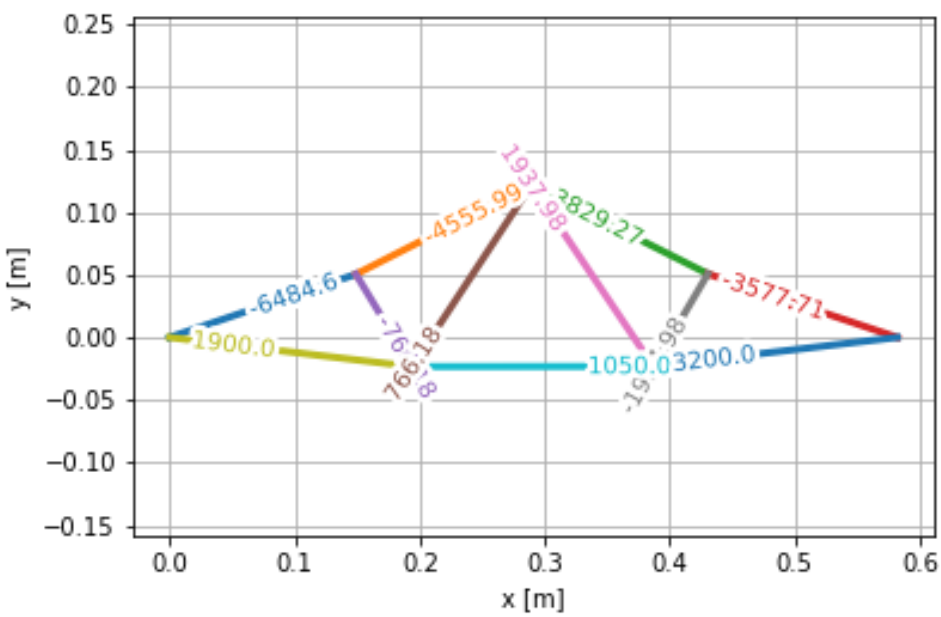


Validação

A validação do software foi realizada com o auxílio de outro software já desenvolvido e utilizado no mercado LISA. Para comparar os resultados do software proposto com o LISA, nós plotamos um gráfico que faz a comparação dos deslocamentos nodais da treliça



Outro método de validação utilizado foi a comparação de forças internas.



Element	Local node	Tensile Force (LISA)	Tensile Force (Software)	Erro Relativo
1	1	-6484.003965	-6484.597	9.14529E-05
1	2	-6484.003965	-6484.597	9.14529E-05
2	1	-765.7522889	-766.1796	0.000557717
2	2	-765.7522889	-766.1796	0.000557717
3	1	1899.385942	1900	0.000323189
3	2	1899.385942	1900	0.000323189
4	1	-4555.469151	-4555.989	0.000114102
4	2	-4555.469151	-4555.989	0.000114102
5	1	765.5663599	766.17965	0.000800452
5	2	765.5663599	766.17965	0.000800452
6	1	-3828.695676	-3829.266	0.000148938
6	2	-3828.695676	-3829.266	0.000148938
7	1	-1937.534761	-1937.984	0.000231807
7	2	-1937.534761	-1937.984	0.000231807
8	1	-3577.160403	-3577.709	0.000153337
8	2	-3577.160403	-3577.709	0.000153337
9	1	1937.280647	1937.9838	0.000362827
9	2	1937.280647	1937.9838	0.000362827
10	1	1050.109708	1050	0.000104484
10	2	1050.109708	1050	0.000104484
11	1	3199.424721	3200	0.000179775
11	2	3199.424721	3200	0.000179775

Nessa tabela nós realizamos a comparação entre as forças internas do software e do LISA sempre calculando o erro. A média dos erros do nosso software foi de aproximadamente 0,2789%.

Conclusão

Nota-se uma proximidade entre a simulação dos deslocamentos nodais entre o software e o LISA. Já na comparação entre forças internas observa-se um erro relativamente pequeno. Consideramos os resultados satisfatórios em virtude da nossa experiência na área e das metodologias utilizadas.

Para futuras iterações é necessário implementar outros métodos numéricos para diversificar as possíveis soluções do problema, implementar técnicas de backtracking para não precisar repetir contas similares, considerar mais fenômenos físicos como atrito, e implementar soluções para treliças 3D.

Bibliografia

- Link para a implementação do software: clique aqui.
https://www.researchgate.net/publication/357279908_Analysis_of_Plane_Truss_using_C_Programme
- <https://www.engineersdaily.com/2011/01/3-methods-for-truss-analysis.html>
- <https://www.geeksforgeeks.org/backtracking-algorithms/>