



2FRAME eDU: UMA FERRAMENTA BASEADA NA WEB PARA ANÁLISE GEOMÉTRICA NÃO LINEAR DE PÓRTICOS PLANOS

DOI: 10.37702/2175-957X.COBENGE.2024.5025

Autores: JOAO CARLOS CORDEIRO BARBIRATO, JOÃO FYLLIPY DE LIMA NUNES

Resumo: Este projeto envolve o desenvolvimento de um software projetado para a análise geométrica não linear de treliças planas. O software utiliza metodologias matriciais, especificamente o Método da Rigidez Direta e o Método Iterativo de Duas Etapas para considerar a não linearidade geométrica, juntamente com princípios da mecânica dos sólidos. Ele permite aos usuários inserir parâmetros da treliça e propriedades dos materiais para calcular a matriz de rigidez e as forças internas da treliça modelada. A aplicação é projetada para ser acessível tanto para estudantes de graduação e pós-graduação quanto para profissionais na área de engenharia estrutural que necessitam de capacidades de análise não linear por meio de um modelo de computação em nuvem. A parte frontal da aplicação é construída utilizando o framework React.JS. Para validar sua precisão, os resultados obtidos pelo software foram comparados com dados de exemplos resolvidos encontrados na literatura relevante.

Palavras-chave: Análise Matricial de Estruturas, Análise Geométrica Não Linear, Desenvolvimento de Aplicativos de Engenharia, Computação em Nuvem.

2FRAME eDU: UMA FERRAMENTA BASEADA NA WEB PARA ANÁLISE GEOMÉTRICA NÃO LINEAR DE PÓRTICOS PLANOS

1 INTRODUÇÃO

A análise estrutural desempenha um papel fundamental no projeto de estruturas civis e mecânicas, visando determinar as forças e os esforços que agem sobre a estrutura, bem como sua resposta às cargas aplicadas. Com o avanço da tecnologia, o uso de ferramentas computacionais tornou-se cada vez mais comum nessa área, capacitando projetistas e engenheiros a conceber estruturas mais eficientes e seguras. Nesse contexto, este trabalho propõe o desenvolvimento de um *software web* que permita aos usuários realizar análises estruturais de pórticos planos, considerando a não linearidade geométrica da estrutura. É essencial que tanto engenheiros quanto alunos de graduação e pós-graduação em engenharia estrutural possam visualizar o comportamento da estrutura sob diferentes carregamentos.

Apesar da disponibilidade de *softwares* como SAP2000, Robot e até mesmo ferramentas gratuitas como o Ftool para análise estrutural, ainda há uma demanda por uma solução que permita a análise não linear de pórticos de forma rápida e prática. Portanto, o desenvolvimento deste *software web* representa uma contribuição significativa para a área, permitindo que engenheiros (graduandos, em primeira linha) projetem estruturas mais seguras e eficientes, levando em consideração a não linearidade geométrica e proporcionando uma avaliação rápida e prática da capacidade de carga da estrutura. Além disso, este trabalho abre possibilidades futuras, como a inclusão de um módulo para dimensionamento de estruturas de aço, seguindo as normas da NBR8800:2008.

Nesse contexto, uma ferramenta foi desenvolvida com base em tecnologia *web* (*cloud computing*) para realizar análises estruturais não lineares, utilizando o Método da Rigidez Direta. Este *software* possibilita uma interação amigável com o usuário (*frontend*) através do *framework React.JS* e da linguagem de programação *JavaScript*, além de oferecer uma visualização gráfica dos elementos de barra presentes nos pórticos planos. Seu propósito é proporcionar uma nova ferramenta para estudantes de graduação e pós-graduação, assim como para professores que ministram disciplinas na área de estruturas, especialmente aquelas que abordam análises de segunda ordem em seus currículos.

2 DA FORMULAÇÃO PARA A ANÁLISE ESTRUTURAL

2.1 Método da Rigidez Direta

A deformação de um material mantém relação direta com a tensão aplicada proporcionalmente a uma constante elástica E , dentro do regime elástico-linear de deformação. O método da rigidez direta faz o uso dessa estratégia ao calcular os esforços atuantes em cada barra do sistema estrutural, por meio da Equação (1) e, por meio do Princípio da Superposição dos Efeitos, representar o comportamento da estrutura como um todo de forma (Kassimali, 2010).

$$\{F\} = [k]\{d\} \quad (1)$$

A matriz $[k]$ contém os chamados coeficientes de rigidez k de cada elemento e possui ordem $3n$, sendo n a quantidade de nós. Estes são coeficientes de proporcionalidade para relacionar deslocamentos unitários com deslocamentos reais. Ou seja: é razoável entendê-los como sendo o efeito resultante de um deslocamento unitário que, ao se relacionar com o deslocamento real, leva à força externa aplicada.

$$\beta_{i0} + X_{ni} = K_i D_i = 0 \quad (2)$$

A Equação (2) descreve o método dos deslocamentos. Cada deslocamento é calculado nos nós da estrutura e cada um deles possui, no caso plano, 3 graus de liberdade e 3 eixos de atuação de força, conforme ilustrado na figura 9. Ou seja: uma estrutura é composta por $3n$ ações, e uma igual quantidade de graus de liberdade. O vetor força $\{F\}$ e o vetor deslocamento $\{d\}$ são escritos como segue:

$$\{F\} = \begin{Bmatrix} f_x^i \\ f_y^i \\ m^i \\ f_x^j \\ f_y^j \\ m^j \end{Bmatrix} \quad \{d\} = \begin{Bmatrix} u_i \\ v_i \\ \theta_i \\ u_j \\ v_j \\ \theta_j \end{Bmatrix} \quad (3)$$

O subíndice na matriz $[u]$ e o sobrescrito na matriz $[F]$, ambos genéricos, denotam os nós de início i e fim da barra j , assumindo os índices correspondentes ao respectivo nó. Com relação ao vetor $[u]$ as variáveis u , v e θ são atribuídas, respectivamente, ao deslocamento horizontal, deslocamento vertical e rotação. A visualização da posição dos elementos pode ser melhor compreendida ao olhar, por exemplo, a indicação dos elementos do vetor de força $\{F\}$ na Figura 1.

Figura 1 – Forças representadas no vetor de forças $\{F\}$.



Fonte: Autores

2.2 Análise Não Linear Geométrica

A análise estrutural busca compreender como as estruturas reagem às forças externas e essa análise deve ser realizada da maneira mais rigorosa possível. Dessa forma, faz-se necessário compreender que há situações em que a estrutura não se comporta de acordo com a lei de Hooke, ou seja, obedecendo a uma relação linear entre tensão (não linearidade física).

Também é possível que deslocamentos laterais excessivos provoquem o aparecimento de momentos fletores não previstos inicialmente (Não linearidade geométrica). Ambas as situações, não linearidades físicas e/ou geométricas, ocorrem em edifícios altos, elementos com fissuração, inconsistências no material, dentre outras. A esse tipo de análise dá-se o nome de "análise não linear" e é de fundamental importância para o melhor entendimento do comportamento do sistema estrutural (Pereira, 2002). A análise não linear também é conhecida como análise de 2ª ordem. Geralmente, diante de processos matemáticos complexos que os métodos rigorosos utilizam, é preferível utilizar métodos simplificados que, apesar de sua simplicidade comparada aos métodos rigorosos, tem mostrado boa precisão, como se observa na literatura. Um desses métodos de análise não linear geométrica consiste, como Moura e Barbirato (2020) descrevem, em analisar as alterações que deslocamentos, ocasionados por forças externas, provocam nos Esforços Internos Solicitantes (EIS). Para isso, aplica-se uma carga horizontal fictícia, de modo a tirar a estrutura de sua condição não deslocada, e verificar como os carregamentos atuantes se comportam nesta nova configuração deslocada. O processo é, então, novamente realizado, desta vez com a figuração deslocada anterior e novamente analisado. Ou seja: trata-se de um método iterativo, que busca a convergência ao equilíbrio da estrutura deslocada (Silva; Lavall, 2004).

A importância dessas metodologias simplificadas fica clara quando a NBR 8800 (2008) apresenta um modo simplificado similar, chamado de Método da Amplificação dos Esforços Solicitantes (MAES), também chamado Método B1, B2, como forma de reduzir o esforço laboral do projetista. Há métodos, entretanto, que levam em consideração a estrutura deformada de forma implícita em suas formulações, como é o caso do Método dos Dois Ciclos Iterativos (aqui chamado de M2CI). Este trabalho levou em consideração apenas a não linearidade geométrica.

É importante salientar que a os efeitos de segunda ordem tem um impacto significativo em estruturas de grande altura, ou seja, aumentam conforme se adicionam mais pavimentos. A variação dos deslocamentos, dessa forma, mostra uma relação linear com a altura da edificação (Konapure; Dhanshetti, 2015). De forma a diminuir a carga computacional para o cálculo do esforço de segunda ordem, pode-se adotar metodologias simplificadas. No desenvolvimento deste trabalho, dada a necessidade de diminuir o esforço computacional motivado pelo processamento ser desenvolvido no lado do *client*, optou-se por seguir esta abordagem e, portanto, utilizar-se-á, graças a sua reduzida necessidade de esforço computacional, o método sugerido por Chen e Lui (1991), chamado Método dos Dois Ciclos Iterativos. O método consiste em realizar dois ciclos de cálculo. O primeiro ciclo utiliza a matriz $[k]$ - que no contexto deste método se atribui a nomenclatura $[k_L]$ para diferenciar da matriz de rigidez atualizada - não sofre qualquer tipo de modificação de segunda ordem, ou seja, é realizada apenas uma análise de primeira ordem convencional. Após a realização do primeiro ciclo e, conseqüentemente, a obtenção dos EIS, inclui-se os efeitos da não linearidade geométrica através da adoção de uma matriz de

rigidez geométrica $[k_g]$, que depende desses esforços internos obtidos na análise do primeiro ciclo.

Inicia-se em seguida o segundo ciclo de cálculo em que a matriz de rigidez $[k_L]$ e somada a matriz de rigidez geométrica $[k_g]$, resultando em uma matriz de rigidez atualizada, como mostra a Equação (4) (em que P nesta equação se refere ao esforço axial na barra). Por fim, efetua-se o cálculo convencional, como se fosse uma análise linear, utilizando a matriz de rigidez atualizada.

$$[k] = [k_L] + [k_g] \quad (4)$$

O Quadro 1 apresenta um quadro resumo das etapas do método.

Quadro 1 – Etapas do Método dos Dois Ciclos Iterativos.

Primeiro ciclo	Passo 1	Realizar uma análise linear da estrutura.
Segundo ciclo	Passo 2	Utilizando os esforços obtidos no passo 1, calcula-se a matriz de rigidez geométrica $[k_g]$.
	Passo 3	Soma-se a matriz $[k_g]$ com a matriz $[k_L]$ gerando uma matriz de rigidez atualizada.
	Passo 4	Faz-se uma nova análise, mas utilizando a matriz de rigidez atualizada.

Fonte: Autores

Dentro do método é possível a utilização de uma matriz de rigidez geométrica $[k_g]$ ainda mais complexa, utilizando os outros esforços internos além do axial, como a sugerida por Rodrigues (2017). Porém, como observado nos resultados obtidos o aparato matemático apresentado por Chen e Lui (1991) apresentou resultados satisfatórios.

3 DO SOFTWARE 2FRAME

O programa fruto deste trabalho foi pensado para suprir a necessidade de análise de engenheiros e estudantes de engenharia para o cálculo rápido, fácil e funcional de estruturas reticuladas planas. Seu uso e interpretação dos resultados é de inteira responsabilidade do usuário, resguardando o desenvolvedor de qualquer implicação. A função do programa é de servir como ferramenta de consulta e não de cálculo final do modelo, devendo o aluno, pesquisador ou engenheiro revisar e validar os resultados. O programa é chamado 2Frame. 2 (*two* em inglês) remete a preposição *to* do inglês, significando "para", enquanto Frame vem do inglês, em tradução aplicada a este contexto, "pórtico". Ou seja: é um programa "para pórticos". Além disso, o numeral 2 indica a possibilidade de realização de análises em segunda ordem.

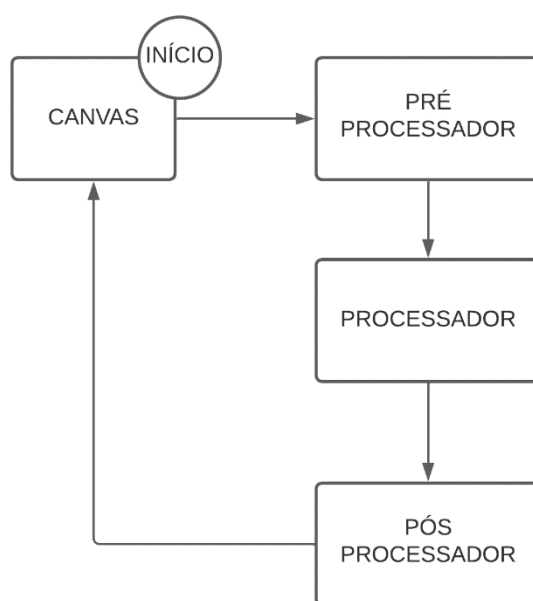
Dentre as diversas opções de programação, a Linguagem JavaScript se destacou, pois é a sintaxe *de facto* para desenvolvimento *web*. Precisa ser, no entanto, aliada a linguagem de marcação HTML para desenvolver páginas *web*. Desta forma, utilizar-se-á a versão ECMAScript 6 do JavaScript.

Com o avanço dos dispositivos móveis e da estrutura de rede da internet, os desenvolvedores de aplicativo têm se voltado para o desenvolvimento *web*. Isso possibilita a existência de aplicativos em nuvem, ou seja, aplicativos que estão armazenados por um servidor e podem ser acessados via browser de qualquer dispositivo conectado à *internet*. Esse modelo de produção em desenvolvimento de sistemas é chamado de *cloud computing*. É possível compreender este tipo de arquitetura de *software* ao fazer analogia de uma corda com dois nós em cada extremidade. Em um nó o usuário entra e recebe dados, visualiza a aplicação e, geralmente, possui um UX (*User Experience*) mais amigável,

chamado de *frontend*, enquanto o outro nó, chamado de *backend* processa as requisições, realiza os cálculos, dentre outras tarefas, de forma a centralizar o processamento, aliviando a carga computacional no lado do cliente. Toda essa comunicação é feita por meio da "corda" que, nesta analogia, é a própria internet. Essa abordagem permite que o usuário possa interagir com a aplicação em qualquer momento e em qualquer lugar, bastando apenas o acesso à *internet*.

A arquitetura de uma aplicação é a forma em que seus diferentes módulos, funções e dados são organizados, seguindo uma ordem lógica. Sua função principal é facilitar o desenvolvimento e manutenção do código. A aplicação fruto deste trabalho está dividida, conforme a Figura 2, em 4 grandes módulos: Pós-processador, Processador, Pré-processador e Canvas.

Figura 2 – Arquitetura da aplicação



Fonte: Autores

O Módulo de Canvas armazena toda a lógica da interface gráfica do usuário. Por conta disso, ela faz o gerenciamento dos outros módulos realizando a troca de dados e supervisionando as requisições do usuário. O Pré-processador faz o tratamento dos dados de entrada do usuário. É responsável por converter a entrada gráfica em *arrays* de coordenadas e forças, bem como a manipulação de dados geométricos e de materiais. Para solucionar o sistema matricial o módulo de processamento recebe, gerenciado pelo Canvas, os dados tratados pelo Processador e realiza os cálculos necessários. Após a solução do modelo da estrutura, os resultados são geridos e organizados pelo Pós-processador, enviando os resultados tratados para a exibição pelo Canvas. Todo o tratamento de dados é feito aproveitando o fato de o Javascript ser uma linguagem orientada a objetos. Dessa maneira, nós e barras são instanciados como objetos e sua manipulação se torna simplificada pelos diferentes módulos, sem a necessidade de funções e métodos complexos.

Para possibilitar um maior controle do código sobre as funções de desenho, optou-se por não utilizar nenhum *framework* gráfico externo, preferindo a criação de uma *engine* 2D própria, batizada de *Object Display on Canvas Engine* ou *ODCE*. Essa decisão foi motivada devido ao maior controle que este autor teria em relação aos aspectos gráficos



da aplicação e a não necessidade de depender de licenças externas. Uma *engine* gráfica é um conjunto de funções que propiciam que gráficos, nesse caso 2D, sejam gerados, transformando as instruções lógicas em equações lineares para exibir as formas necessárias e requisitadas na tela. A *engine 2D* aqui desenvolvida é rudimentar, utilizando os métodos de *canvas* do próprio *Javascript*. Por se tratar de algo bem simples e rudimentar, foi implementado apenas um espaço vetorial em metros e uma transformação de escala. Uma das características da *ODCE* é a possibilidade de escalonar o desenho e modificar o espaçamento entre as linhas de *grid*. Esse controle é possível com a barra de ferramentas inferior. Uma descrição da barra de ferramentas inferior é fornecida na Tabela 2.

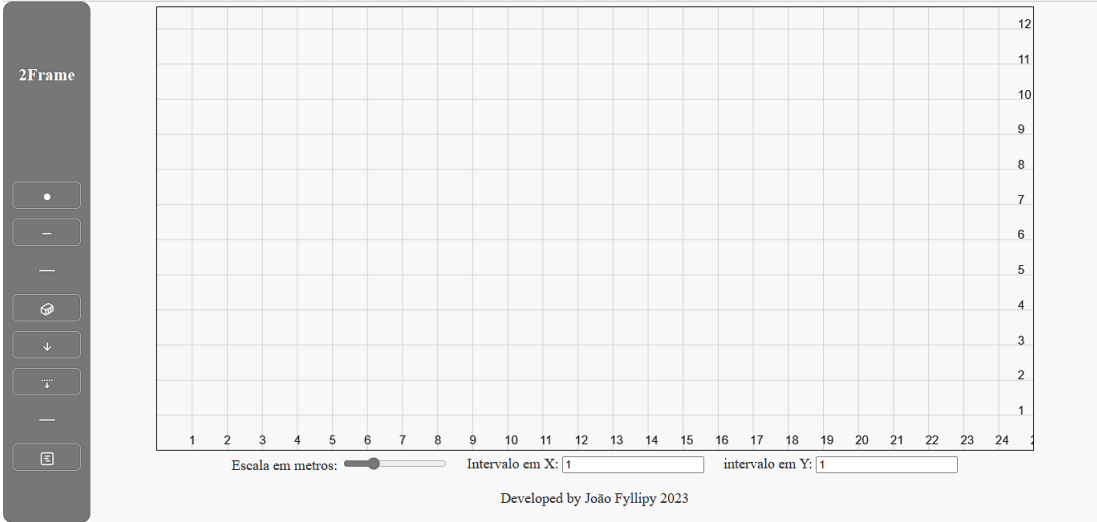
Figura 3 – Barra de ferramentas inferior.



Fonte: Autores

A interface do usuário foi pensada levando em conta as seguintes diretrizes: experiência do usuário (UX), simplicidade e leveza. Essas diretrizes foram seguidas sempre que possível, culminando com a interface padrão da aplicação (Figura 4), a qual é dividida em 4 áreas principais: barra lateral, barra inferior, mesa de desenho e *pop-ups* (ou janelas).

Figura 4 – Interface geral da aplicação.



Fonte: Autores

O Quadro 2 mostra a função de cada uma dessas áreas, aqui chamada de elementos de *UI* (*User Interface*).

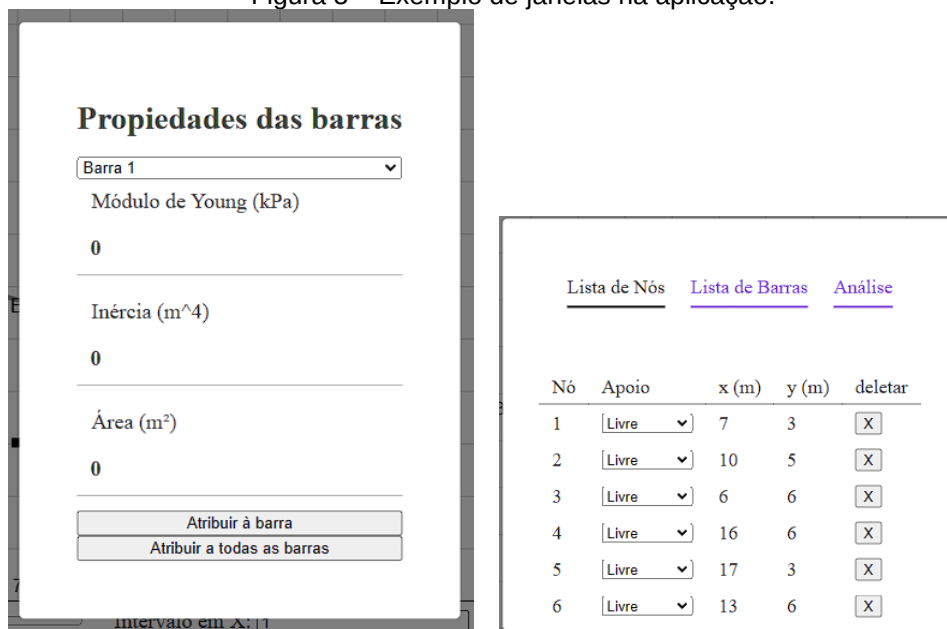
Quadro 2 – Elementos da interface de usuário.

Elemento	Descrição
Barra lateral	Guarda as ferramentas para a inserção de dados, como a inserção de nós, barras, inserção de propriedades geométricas e de material, inserção de forças e a ferramenta de revisão e análise.
Barra inferior	Controla a visualização da área de desenho, como o espaçamento do <i>grid</i> e sua escala.
Área de desenho	Local composto por um <i>grid</i> , onde o usuário faz o desenho da estrutura através de nós e barras.
janelas	<i>Pop-ups</i> que surgem ao selecionar algumas ferramentas com um formulário para sua manipulação.

Fonte: Autores

O conceito de janelas foi adotado (Figura 5) pois facilita a interação do usuário com os dados, funcionando como assistentes, exibindo campos de inserção de dados, suas descrições e unidades.

Figura 5 – Exemplo de janelas na aplicação.



Propiedades das barras

Barra 1

Módulo de Young (kPa)

0

Inércia (m⁴)

0

Área (m²)

0

Atribuir à barra

Atribuir a todas as barras

Lista de Nós

Nó	Apoio	x (m)	y (m)	deletar
1	Livre	7	3	X
2	Livre	10	5	X
3	Livre	6	6	X
4	Livre	16	6	X
5	Livre	17	3	X
6	Livre	13	6	X

Fonte: Autores

A janela de revisão e análise traz, ainda, o conceito de abas, onde é possível revisar cada elemento da estrutura em suas respectivas abas. Também é nesta janela que está a seção de análise e resultados (mostrado somente após a realização da análise).

A seção de resultados traz um relatório completo, com tabelas de reações, bem como gráficos de esforços internos solicitantes de cada uma das barras (Figura 6).

Figura 6 – Tabelas do relatório de análise de um exemplo genérico.

Reações

Nó	X	Y	Mz
1	-10.00 kN	50.00 kN	0 kN.m
2	-10.00 kN	50.00 kN	-0.00 kN.m

Deslocamentos

Nó	u	v	Theta
1	0.00000 m	0.00000 m	-0.13810 rad
2	0.00000 m	0.00000 m	0.13810 rad

Esforços Internos Solicitantes

Barra	N1	V1	M1	N2	V2	M2
1	0.00 kN	50.99 kN	-0.00 kN.m	0.00 kN	-50.99 kN	-0.00 kN.m

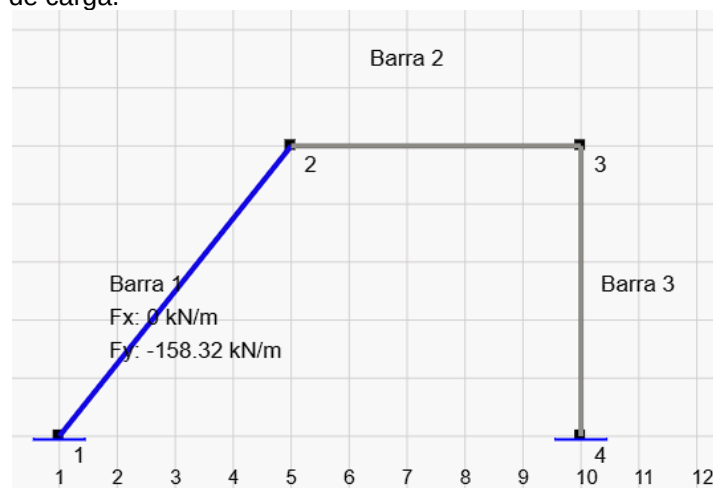
Fonte: Autores

Um diferencial no presente trabalho é o caráter de facilitação de uso *on line*. Para tanto, foi escolhida a forma e hospedagem para disponibilizar o programa. O termo hospedagem, no meio da computação, significa "alocar um espaço na *web* para uma página", ou seja, é o ato de colocar no ar o servidor da aplicação. A hospedagem pode ser realizada pelo próprio produto (neste caso, *software*, com a implantação de um servidor *in house*). Porém, por conta da praticidade e custo, sempre é preferível utilizar serviços que realizem a hospedagem de páginas. No momento do desenvolvimento deste trabalho, o programa encontra-se hospedado em um *host* gratuito, ficando à mercê das políticas e cláusulas do serviço. Esta foi a principal dificuldade durante a elaboração do programa, pois funcionalidades pensadas originalmente para o 2Frame tiveram que ficar fora da *web* por questões relacionadas ao serviço gratuito e suas limitações. Assim, o autor deste trabalho não pode garantir a sua permanência na rede, ficando reservado ao mesmo apenas o código fonte.

4 APLICAÇÃO

Para a comparação dos resultados foi modelado, no 2Frame e no *software* AcadFrame, que realiza análise não linear geométrica utilizando o método dos elementos finitos posicionais, um pórtico com 3 barras, sendo uma destas inclinada, e carregamento linearmente distribuído nesta barra, conforme visto na Figura 7. O vão vencido pela viga (barra 2) é de 5 metros e a altura do pavimento e também de 5 metros. Os apoios são do tipo engaste. Foi utilizada uma única geometria de seção para todas as barras, atribuindo o perfil CVS 500x217. O módulo de elasticidade adotado foi de $E = 205000$ MPa. A análise foi realizada com 8 casos de carregamento, com incrementos de 12,5% na carga até o limite de 1266,56 kN/m. Portanto, o carregamento do primeiro caso equivale a 158,32 kN/m. Salienta-se que os resultados apresentados são oriundos da análise não linear geométrica apenas. Todos os resultados comparados foram os referentes ao nó 2 da estrutura, conforme indicado na Figura 7.

Figura 7 – Pórtico modelado no 2Frame com o primeiro caso de carga.



Fonte: Autores

Na Tabela 1 são apresentados os resultados obtidos pelo AcadFrame e 2Frame para os deslocamentos do nó 2.

Tabela 1 – Comparação dos deslocamentos horizontais (m) no AcadFrame e no 2Frame no nó 2

P/P_{ref}	Deslocamento no AcadFrame (m)	Deslocamento no 2Frame (m)	Erro (%)
0,125	0,01248	0,0121	3,04
0,250	0,02497	0,0242	2,92
0,375	0,03746	0,0364	2,82
0,500	0,04996	0,0486	2,72
0,625	0,06246	0,0608	2,65
0,750	0,07497	0,0731	2,49
0,875	0,08748	0,0854	2,37
1,000	0,09999	0,0977	2,23

Fonte: Autores

Nota-se que, à medida em que a carga aumenta, chegando ao limite $P/P_{ref} = 1$, o erro diminui, chegando a 2,23%. Considerando que cargas maiores exigem maior confiabilidade, um erro cada vez menor a medida em que se aumenta a carga, considerando apenas o AcadFrame como referência, a segurança também está aumentando, pois se converge a um resultado cada vez mais similar. Isso mostra a segurança do 2Frame para o cálculo de flechas, por exemplo.

Por fim, as comparações de resultados e os erros percentuais obtidos para os Momentos Fletores são apresentados na Tabela 2.

Tabela 2 – Comparação dos Momentos Fletores (kN.m) no AcadFrame x 2Frame no nó 2

P/P_{ref}	Fletor no AcadFrame (kN.m)	Fletor no 2Frame (kN.m)	Erro (%)
0,125	272,19	271,74	0,16
0,250	546,89	545,12	0,32
0,375	824,1	820,17	0,47
0,500	1103,81	1096,87	0,62
0,625	1386,04	1375,25	0,78
0,750	1670,78	1655,3	0,92
0,875	1958,03	1937,04	1,07
1,000	2247,8	2220,46	1,21

Fonte: Autores

Os Momentos Fletores, como observado na Tabela 2, são os que apresentam a menor diferença, tendo um erro de apenas 1,21% para a carga limite P_{ref} . Importante ressaltar que, para a carga mais baixa, houve um erro também muito pequeno, de 0,16%. Esses resultados mostram, como os demais, a confiabilidade do 2Frame quando comparado ao AcadFrame. Em suma, os resultados comparativos apresentados demonstram a confiabilidade dos cálculos realizados pelo *software* desenvolvido.

5 CONSIDERAÇÕES FINAIS

O desenvolvimento do *software web*, nomeado de 2Frame, foi apresentado em detalhes, descrevendo as funcionalidades e recursos disponíveis para os usuários, além de destacar as principais vantagens e desafios enfrentados durante o desenvolvimento. Em seguida, foram realizadas análises comparativas entre os resultados obtidos pelo *software* desenvolvido e *software* acadêmico AcadFrame.

Os resultados obtidos demonstraram que o 2Frame apresentou resultados satisfatórios, comparáveis com os demais métodos de cálculo. Além disso, foi possível destacar a vantagem do *software* em relação a facilidade de uso, acessibilidade e praticidade, uma vez que pode ser acessado por qualquer dispositivo com conexão à internet. Conclui-se, por fim, que o *software* desenvolvido é uma ferramenta valiosa para os profissionais da área e alunos de graduação e pode contribuir para o entendimento do comportamento de estruturas, tanto em análises lineares quanto em análises não lineares.

REFERÊNCIAS

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **NBR 8800: Projeto de estruturas de aço e de estruturas mistas de aço e concreto de edifícios**. Rio de Janeiro, 2008. 247 p.

CHEN, W.-F.; LUI, E. M. **Stability Design of Steel Frames**. [S.l.]: CRC Press, 1991.

CODA, H. B.; PACCOLA, R. R. **AcadFrame: Software Acadêmico para análise de pórticos e treliças planas**. 2006. Acesso em 18/05/2023. Disponível em: <http://web.set.eesc.usp.br/software_depto/acadframe/>.

KASSIMALI, A. **Matrix Analysis of Structures**. 2. ed. [S.l.]: Cengage, 2010

KONAPURE, C. G.; DHANSHETTI, P. V. **Effect of p-delta action on multi-storey buildings**. International Journal of Engineering Research & Technology, v. 4, 2015.

MOURA, B. C. de A.; BARBIRATO, J. C. C. **Contribuição para análise não linear de pórticos planos de edifícios utilizando o p-delta com a técnica das cargas laterais fictícias**. XLI CILAMCE, 2020.

PEREIRA, A. **Projeto ótimo de pórticos planos com restrição à flambagem**. 99 p. Dissertação (Mestrado em Engenharia Civil) — Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, 2002.

RODRIGUES, M. **Soluções integradas para as formulações do problema de não linearidade geométrica**. Tese (Doutorado) — Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, 2017.

SILVA, R. G. L. da; LAVALL, A. C. C. **Utilização de análises elásticas estimadas ou aproximadas de 2ª ordem para o dimensionamento de pórticos planos de aço**. XXXI Jornadas Sud-Americanas de Ingeniería Estructural, 2004.

2FRAME eDU: A WEB-BASED TOOL FOR NON-LINEAR GEOMETRIC ANALYSIS OF 2-DIMENSIONAL FRAMES

Abstract: *This project involves the development of software designed for the non-linear geometric analysis of plane frames. The software utilizes matrix methodologies, specifically the Direct Stiffness Method and the Two Cycle Iterative Method to account for geometric nonlinearity, alongside principles from solid mechanics. It allows users to input frame parameters and material properties to calculate the stiffness matrix and internal forces of the modelled frame. The application is designed to be accessible to both undergraduate and graduate students as well as professionals in the structural engineering field who require non-linear analysis capabilities via a cloud-based computing model. The front-end of the application is built using the React.JS framework. To validate its accuracy, the results obtained from the software were compared against data from solved examples found in relevant literature.*

Keywords: *Structural Analysis, Matrix Analysis of Structures, Nonlinear Geometric Analysis, Engineering App Development, Cloud Computing.*

