



UNIVERSIDADE FEDERAL DE SERGIPE
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA CIVIL



DANILO MENEZES SANTOS

**ANÁLISE NÃO LINEAR GEOMÉTRICA DE ESTRUTURAS
RETICULADAS ATRAVÉS DE UM ALGORITMO DE
DIFERENCIAÇÃO AUTOMÁTICA**

SÃO CRISTÓVÃO – SE
Junho, 2019

DANILO MENEZES SANTOS

**ANÁLISE NÃO LINEAR GEOMETRICA DE ESTRUTURAS
RETICULADAS ATRAVÉS DE UM ALGORITMO DE
DIFERENCIAÇÃO AUTOMÁTICA**

Trabalho de Conclusão de Curso submetido ao Departamento de Engenharia Civil da Universidade Federal de Sergipe como requisito parcial para obtenção do título de Bacharel em Engenharia Civil.

Orientador: Prof. Dr. Jorge Carvalho Costa

SÃO CRISTÓVÃO – SE

Junho, 2019

É concedida à Universidade Federal de Sergipe permissão para reproduzir cópias desta monografia e emprestar ou vender tais cópias somente para propósitos acadêmicos e científicos. O autor reserva outros direitos de publicação e nenhuma parte deste trabalho acadêmico pode ser reproduzida sem a autorização por escrito do autor.

Assinatura

DANILO MENEZES SANTOS

**ANÁLISE NÃO LINEAR GEOMETRICA DE ESTRUTURAS
RETICULADAS ATRAVÉS DE UM ALGORITMO DE
DIFERENCIAÇÃO AUTOMÁTICA**

Trabalho de Conclusão de Curso submetido ao Departamento de Engenharia Civil da Universidade Federal de Sergipe como requisito parcial para obtenção do título de Bacharel em Engenharia Civil.

Aprovado em: _____ de _____ de _____.

BANCA EXAMINADORA

Nota: _____

Prof. Dr. Jorge Carvalho Costa
(Presidente – Orientador)

Nota: _____

Prof. Dr. Marco Antonio Brasiel Sampaio
(1º Examinador)

Nota: _____

Prof. Dr. Janio Coutinho Canuto
(2º Examinador)

DEDICATORIA

Agradeço primeiramente, a Deus e Nossa Senhora das Graças que sempre me deram sabedoria e perseverança nessa caminhada.

Ao meu orientador Dr. Jorge Carvalho Costa, pelo apoio, o incentivo, o conhecimento, as muitas histórias contadas e o entusiasmo transmitido ao longo desse trabalho.

Aos amigos da vida: André, Felipe, Genisson, Fábio, Layne, Jéssica Yasmin e Rafael Nunes, que estiveram comigo e me cederam o ombro tanto nas vitórias como nas derrotas. A Fábio, Verena, Lucas, Dalila, Gilton, entre outros amigos que obtive na UFS e espero manter para a eternidade.

Aos meus familiares, em especial minha irmã Taíssa e minhas Tia Neide, Vânia e Ana, que me apoiaram ao longo de todo essa jornada.

Aos meus avôs (in memoriam) com os quais tive pouco contato, mas sei que, lá de cima, olham por mim. A minhas avós Maria Elze e Ibelza (in memoriam), duas matriarcas incríveis às quais eu e todos da família devemos tudo.

Aos meus pais Paulo e Maria José, que me transmitiram os melhores valores que uma pessoa poderia receber; além de sempre terem feito de tudo para proporcionar a melhor educação a mim e a minha irmã.

A todos os professores e profissionais do Departamento de Engenharia Civil da UFS que dedicaram parte de seu tempo a me guiar no caminho da Engenharia.

>>> `import this`

(Tim Peters, 1999)

RESUMO

Este trabalho apresenta o desenvolvimento de um algoritmo de diferenciação automática em Python 3.x. Implementado com os modos reverso e direto, sendo capaz de derivar funções representadas como funções ou strings. Para verificar sua validade o código foi utilizado para a obtenção da matriz de rigidez local linear e tangente, de treliças e pórticos planos em análise linear e geometricamente não linear, através de métodos de energia. O desenvolvimento da teoria estrutural ocorreu em Referencial Lagrangeano Total, e os resultados foram comparados com exemplos encontrados na literatura técnica. Os resultados obtidos foram satisfatórios para os exemplos estudados tendo convergência de seus valores tanto para as matrizes de rigidez quanto para deslocamentos nodais e esforços internos.

Palavras Chave: derivação automática, análise estrutural não linear, mecânica computacional, não linearidade geométrica, treliça plana, pórtico plano.

ABSTRACT

This work presents the development of an algorithm for automatic differentiation in python. Implemented through the reverse and forward mode, it is able to differentiate functions introduced through strings and functions. In order to verify its validity, the code was used to obtain the local stiffness matrix of trusses and frame planes in linear and nonlinear geometrically analyzes, through energy methods. The development of the structural theory was carried in a Total Lagrangian approach, and the results were compared with examples found in the technical literature. The results obtained were satisfactory to the examples analyzed as much for the stiffness matrix as for nodal displacements and internal forces.

Keywords: automatic differentiation, nonlinear structural analysis, computational mechanics, geometric nonlinearity, plane truss, plane frames.

Lista de Figuras

Figura 1 – Sistema de coordenada global XY e os sistemas locais y_i, x_i	16
Figura 2 – Exemplo de Treliça.	19
Figura 3 – Sistema de Referência Global.	19
Figura 4 – Deslocamentos Nodais.	20
Figura 5 - Exemplo de Pórtico.....	23
Figura 6 – Barra de Pórtico com deslocamentos nodais.....	23
Figura 7 – Grafo com os vértices (a;b;c) e arestas: [(a,b);(c,b)]......	30
Figura 8 – Árvore da função y.....	31
Figura 9 – Sequência da AD no método direto.	33
Figura 10 – Grafo de todos os caminhos da variável x_1	34
Figura 11 – Entrada de dados no módulo.	35
Figura 12 – Saída do Bytecode da equação Z.	37
Figura 13 – Vértices do grafo computacional armazenado em um dicionário para equação Z.	37
Figura 14 – Lista no modo direto.	37
Figura 15 – Caminhos no modo reverso.	38
Figura 16 – Saída de dados como variáveis simbólicas para o caso da figura 11.....	38
Figura 17 – Tabela de pontos para integração de Gauss.	40
Figura 18 – Treliça de três barras.	41
Figura 19 – Comparação das matrizes de rigidez lineares barra 1.	42
Figura 20 – Comparação das matrizes de rigidez não lineares barra 3.	42
Figura 21 – Treliça de quatro barras.....	43
Figura 22 – Comparação das matrizes de rigidez não lineares barra 2.	44
Figura 23 – Treliça de uma barra.	45
Figura 24 – Ilustração da estrutura.	47
Figura 25 – Comparação das matrizes de rigidez lineares para a barra 10.	48
Figura 26 – Comparação das matrizes de rigidez não lineares barra 10.	49
Figura 27 – Pórtico do exemplo.	50
Figura 28 – Gráfico de Cargas x Deslocamento.....	52
Figura 29 – Comparação das matrizes de rigidez não lineares barra 10.	54
Figura 30 – Ilustração da Estrutura	55
Figura 31 – Gráfico de Deslocamento x Passo de Carga	56

Lista de Tabelas

Tabela 1 – Deslocamentos Horizontais Exemplo 1.....	42
Tabela 2 - Deslocamentos Verticais Exemplo 2.....	44
Tabela 3 – Comparação forças axiais.	45
Tabela 4 – Comparação resultados.	46
Tabela 5 – Comparação de resultados análise não linear.	47
Tabela 6 – Comparação de resultados análise linear.	48
Tabela 7 – Comparação dos deslocamentos horizontais.	51
Tabela 8 – Comparativo de resultados pilar esquerdo.....	53
Tabela 9 – Comparativo de resultados	55

Sumário

1	INTRODUÇÃO	11
1.1	OBJETIVOS PRINCIPAIS	13
1.2	OBJETIVOS ESPECÍFICOS	13
2	REVISÃO DE LITERATURA	14
3	FUNDAMENTAÇÃO TEÓRICA	16
3.1	ENERGIA DE DEFORMAÇÃO E TEOREMA DE CASTIGLIANO	17
3.2	TRABALHO EXTERNO E ENERGIA POTENCIAL TOTAL.....	17
3.3	ELEMENTO DE BARRA PARA TRELIÇA PLANA	18
3.3.1	CINEMÁTICA	19
3.3.2	LINEARIZAÇÃO POR SÉRIE DE TAYLOR	22
3.4	ELEMENTO DE BARRA DE PÓRTICO PLANO	22
3.4.1	CINEMÁTICA	23
3.5	QUADRATURA DE GAUSS.....	27
3.6	MÉTODO DE NEWTON RAPHSON	27
4	DERIVAÇÃO AUTOMÁTICA	29
4.1	FUNÇÕES COMPOSTAS	29
4.2	DERIVADAS DE FUNÇÕES COMPOSTAS.....	30
4.3	INTRODUÇÃO AOS GRAFOS.....	30
4.4	ESTRATÉGIAS DE DERIVAÇÃO AUTOMÁTICA.....	32
4.4.1	MÉTODO DIRETO	32
4.4.2	MÉTODO REVERSO	33
4.5	UMA FERRAMENTA DE DIFERENCIAÇÃO AUTOMÁTICA	35
5	METODOLOGIA PARA VALIDAÇÃO DOS RESULTADOS	39
6	RESULTADOS E DISCUSSÃO	41
6.1	TRELIÇA PLANA DE TRÊS BARRAS – ANÁLISE NÃO LINEAR.....	41
6.2	TRELIÇA PLANA DE 4 BARRAS – ANÁLISE NÃO LINEAR E LINEAR ...	43
6.3	TRELIÇA DE UMA BARRA – ANÁLISE NÃO LINEAR.....	45
6.4	COLUNA SUBMETIDA A FLEXÃO COMPOSTA NORMAL	46
6.5	PÓRTICO NÃO CONTRAVENTADO DE UM ANDAR E UM VÃO.....	49
6.6	PÓRTICO DE EDIFÍCIO.....	54
7	CONCLUSÃO	57
8	REFERÊNCIAS	59
9	ANEXOS	63

1 INTRODUÇÃO

As técnicas de derivação são um conceito bastante utilizado nas Engenharias e Ciências. Sua aplicação engloba assuntos como a resolução de sistemas não lineares, otimização numérica, machine learning, problemas inversos, métodos para solução de equações diferenciais, além de outros temas, que ilustram o poder dessa ferramenta.

Seu surgimento se deu através dos esforços de Newton e Leibniz, para o emprego funcional do cálculo diferencial por volta do século XVII. Autores como Birgin (1998) citam que existem quatro formas de realizar uma diferenciação: à mão, numericamente, de maneira simbólica e através de algoritmos, este último conhecido como derivação algorítmica ou automática.

A diferenciação à mão consiste em desenvolver as expressões algébricas necessárias valendo-se das regras de cálculo para derivação, obtendo fórmulas matemáticas para as derivadas. Possui o inconveniente de, ao se trabalhar com funções de grande tamanho e complexidade, poder se tornar inviável e muito susceptível a erro nos cálculos. Além disso, para uso em sistemas computacionais, tanto a função original quando suas derivadas precisam ser traduzidas para uma linguagem de programação, passo sujeito a erros.

Já o método numérico ocorre através da aproximação do valor da derivada em um ponto, por meio de diferenças finitas. Esse procedimento elimina o uso de grandes expressões, mas introduz erros de truncamento o que reduz sua eficiência (Haftka e Gurdal, 1992 apud Barthelem e Hall, 1992).

A derivação simbólica consiste no uso de softwares específicos de análise matemática que codificam as regras do cálculo para automatizar a obtenção de expressões algébricas para as derivadas. Estas são obtidas com precisão e sem erros de truncamento e, a depender das capacidades do software usado, as expressões podem ser simplificadas e compactadas. Em contrapartida temos um custo computacional muito elevado, aumentando bastante o tempo de processamento para problemas de grande porte. Outra desvantagem ocorre quando não há compatibilidade entre o software algébrico e a plataforma de desenvolvimento do algoritmo, nessa situação pode ter que se trabalhar com grandes expressões, aumentando assim a possibilidade de erros na implementação.

No último caso temos as técnicas de derivação algorítmica, também conhecida como AD (Algorithmic Derivation). Consiste em aplicar as regras de derivação diretamente em um código ou pseudocódigo computacional. A função original é programada e o algoritmo de AD é empregado no código, retornando uma nova função computacional que calcule as derivadas.

Esse processo apresenta vantagens em comparação aos outros citados, visto que gera resultados com precisão analítica, sem erros de truncamento e com custo computacional inferior ao da derivação simbólica. Devido a essas características, a AD vem sendo bastante utilizado pela comunidade acadêmica, com aplicação nas mais diversas áreas como: *Machine Learning* (Šrajer, Kukulova and Fitzgibbon 2016), Otimização Topológica (Linn, 2010), entre outras.

Em análise não-linear de estruturas, podem ser citados dois importantes processos onde as derivadas são foco principal. Por métodos de energia, o equilíbrio de uma estrutura pode ser imposto como um ponto de mínimo da energia potencial do sistema, e a busca deste mínimo exige a avaliação das derivadas da função de energia. A própria imposição de equilíbrio, seja ela feita diretamente ou por minimização da energia, constitui uma série de funções não-lineares dos deslocamentos e as técnicas normalmente empregadas para resolução desses sistemas necessitam das derivadas das funções constituintes.

Vê-se que é um campo bastante fértil para a aplicação de técnicas de derivação. Estruturas grandes ou complexas terão as funções que descrevem sua energia potencial igualmente complexas e o número de variáveis necessárias para descrever sua deformação em número significativo. É, portanto, de grande importância a investigação do uso de técnicas algorítmicas de derivação neste campo.

1.1 OBJETIVOS PRINCIPAIS

Este trabalho visa desenvolver um algoritmo de diferenciação automática em linguagem Python 3, para derivação de equações introduzidas através de funções e *strings*, gerando como pacotes de saída, a derivada computacional da função.

1.2 OBJETIVOS ESPECÍFICOS

- Revisar a bibliografia teórica sobre derivação automática.
- Implementar um algoritmo para realização da derivação automática em linguagem Python 3.
- Validação dos resultados através da determinação da matriz de rigidez de treliças e pórticos, submetidas a análise elástica linear e geometricamente não linear.

2 REVISÃO DE LITERATURA

Para Nick Trefethen (2005 *apud* Griewank, 2012), a diferenciação automática foi um dos 30 maiores algoritmos do século XX. Walther e Griewank (2003) explicam que as técnicas de derivação automática surgiram a cerca de 50 anos, porém sua popularização através de frameworks estáveis se deu apenas em meados dos anos 90.

Linn (2010), define o conceito de diferenciação automática (AD) como um método numérico-computacional para o cálculo de derivadas de funções com precisão analítica, efetuadas de forma numérica.

Isso é possível porque a diferenciação automática baseia-se no fato de que toda função, independentemente de sua complexidade, pode ser representada através de um grafo formado por uma sequência de operações (soma, subtração, multiplicação e divisão) e funções elementares. Possibilitando assim, que ao se aplicar a regra da cadeia nos nós do grafo, calcule-se o valor da derivada da função principal. (Birgin, 1998).

Existem duas técnicas de implementação da AD, que são o método direto (*forward mode*) e o método reverso (*reverse mode*). Barthelemy e Hall (1992) explicam que no método direto ocorre a aplicação da regra da cadeia no fluxo do grafo, das variáveis independentes às dependentes, fazendo que o custo computacional esteja diretamente ligado ao número de variáveis de entrada da função. Por outro lado, o modo reverso avalia as derivadas no sentido inverso ao do grafo, vinculando assim o custo computacional ao número de saídas da função.

O uso da AD tem tido vasto emprego durante as últimas décadas. Por exemplo, Haase et al (2002), utilizaram um algoritmo de derivação automática, baseado no *reverse mode*, na otimização da massa de um quadro, submetido ao estado plano de tensão, a necessidade de se, pois, o problema em questão possuía cerca de 450 parâmetros e 7500 graus de liberdade. A derivação algorítmica foi empregada através do software ADOL-C, desenvolvido por Griewank et al.

Takahashi e Kawahara (2004) comprovaram a validade de empregar um algoritmo de diferenciação automática em um problema inverso, identificando os parâmetros de escoamento em um fluido sob as condições de Navier Stokes. Os autores avaliaram as derivadas através da biblioteca FABBAD 2.0 desenvolvida por Bendtsen e Stauning (1996).

Linn (2010) em seu trabalho aplicou um algoritmo de derivação numérica, para otimização de cascas axissimétricas através de Elementos Fintos, em ambiente Fortran. O framework de diferenciação foi o TAPENADE, desenvolvido pelo *Institut National de Recherche en Informatique et en Automatique* (INRIA).

Lengiewicz, Korelc e Stupkiewicz (2010) implementaram uma abordagem híbrida entre a computação simbólica e numérica, para realizar a análise em elementos finitos para grandes deformações em problemas de contato. Esse processo foi desenvolvido através dos pacotes AceGen e AceFEM, com a junção das técnicas de derivação automática, otimização de expressões e geração automática do código, de modo integrado ao software Mathematica.

Nørgaard *et al* (2017) fizeram um estudo sobre a aplicação da AD na otimização topológica, através da simulação do escoamento de um fluido regido pelas leis de Lattice Boltzmann, com o emprego dos pacotes TAPENADE e CoDiPack.

Além dos frameworks já citados, outros projetos vêm sendo desenvolvidos para possibilitar o progresso da diferenciação algorítmica, entre eles: CasAdi (Andersson, AAkesson e Diehl), ADOL-C (Walther e Griewank), ADOL-F (Shiriaevev, Utker e Griewank), AceGen (Korelc) e etc. Além desses, existem alguns módulos implementados em Python como: PyTorch, Fenics Project, Theano, PyCppAD, entre outros.

3 FUNDAMENTAÇÃO TEÓRICA

No presente capítulo, será apresentada a teoria necessária para formulação de um elemento de barra constituído por material homogêneo e isotrópico, utilizado para a análise linear e não linear geométrica de treliças e pórticos planos, que serão aplicadas para a validação do pacote de AD.

Segundo Segnini (2000) o conceito de linearidade geométrica sugere que são válidas as hipóteses de pequenos deslocamentos, deformações e rotações. Enquanto a linearidade física baseia-se no comportamento tensão-deformação linear do material (lei de Hooke).

Ao considerar essas duas hipóteses realiza-se uma análise linear da estrutura. Por outro lado, a consideração da não linearidade geométrica, também objetivo desse trabalho, implica na possibilidade de a estrutura sofrer grandes deslocamentos, deformações e/ou rotações gerando assim o surgimento de esforços de segunda ordem. A literatura apresenta ainda a não linearidade física, caracterizada pela adoção de material com comportamento tensão-deformação não linear.

A figura 1, mostra um sistema de coordenada local e global para um pórtico:

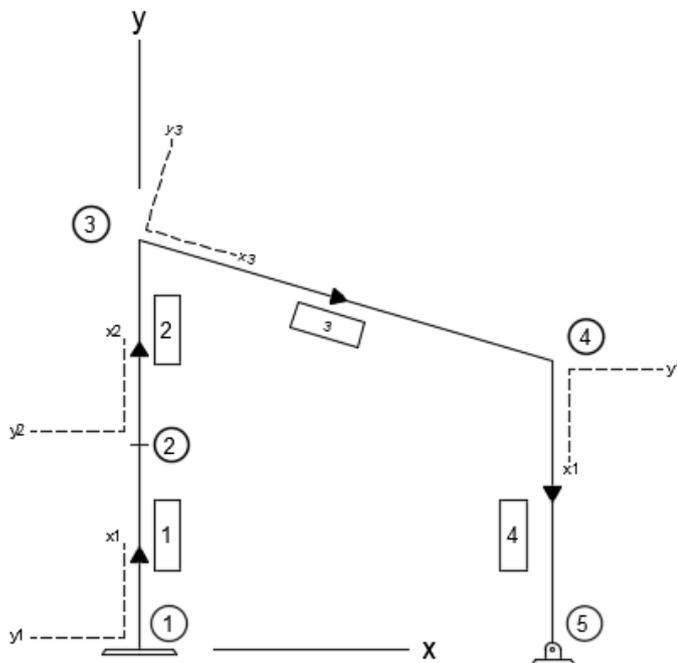


Figura 1 – Sistema de coordenada global XY e os sistemas locais y_i, x_i .

Fonte: Autor.

Foram analisadas estruturas planas de treliça e pórticos. Em ambos os casos, as estruturas são descritas em um sistema global de coordenadas XY por seus nós e seus elementos. Em cada barra, descreve-se um sistema local de coordenadas $x_i y_i$, sendo i o identificador da

barra. O eixo x_i tem sua origem no nó inicial do elemento e direção igual ao mesmo, já o eixo y_i é perpendicular a x_i e orientado de forma de modo que o sistema local de coordenadas obedeça à regra da mão direita.

Os sistemas de coordenadas local e global se relacionam por uma translação da origem ao nó inicial do elemento e uma rotação descrita pela matriz T .

3.1 ENERGIA DE DEFORMAÇÃO E TEOREMA DE CASTIGLIANO

A análise estrutural consiste em encontrar a configuração onde todos os elementos encontram-se no estado de equilíbrio, respeitadas as suas condições de contorno (restrições e carregamentos). Nesse trabalho, emprega-se os métodos de energia para a determinação do equilíbrio da estrutura.

Soriano (2003 *apud* Sousa 2017) conceitua a energia de deformação (U) como a energia que um dado corpo absorve sob a influência de um carregamento externo, desde que não ocorram deslocamentos de corpo rígido. Para uma barra de material isotrópico, sob esforços solicitantes fletores e/ou normais, a energia de deformação armazenada é:

$$U = \frac{1}{2} \int \sigma_{xx} \cdot \varepsilon_{xx} \cdot dV \quad (1)$$

Onde σ_{xx} é a tensão normal à seção transversal e ε_{xx} é a deformação específica na mesma direção. A partir do conceito de energia interna, Ramalho e Silva [s. d.] explicam que pelo Primeiro Teorema de Castigliano, a derivada parcial da energia de deformação, em relação a qualquer deslocamento u_i é igual à força interna correspondente F_i .

$$F_i = \frac{dU}{du_i} \quad (2)$$

A importância desse teorema dá-se, pois, sendo a derivada parcial da energia de deformação uma força, podemos obter um conjunto de n equações que relacionam as forças internas aos deslocamentos dos nós da estrutura, tal que:

$$F_1 = \frac{dU}{du_1}, \quad F_2 = \frac{dU}{du_2}, \dots, \quad F_n = \frac{dU}{du_n} \quad (3)$$

3.2 TRABALHO EXTERNO E ENERGIA POTENCIAL TOTAL

Além da energia de deformação, a atuação de um carregamento externo produz outra energia, chamada de Trabalho Externo (Ω). Este é o produto entre a carga (Q_i) e o deslocamento (u_i) de seu ponto de aplicação. Como carga entende-se todas as forças volumétricas (ρ), superficiais (b), concentradas (P_i), momentos (M_j) e torções (T_k) atuantes em um corpo, como pode ser visto a seguir:

$$\Omega = \int_V \rho \cdot u \, dV + \int_D b \cdot u \, dS + \sum_n P_i \cdot u_i + \sum_n M_j \cdot u_j + \sum_n T_k \cdot u_k \quad (4)$$

Em um sistema estrutural cujas deformações ocorram de modo gradual (carregamento estático) pode se desprezar a influência da energia cinética, tornando assim a combinação do trabalho externo e da energia de deformação, como a Energia Potencial Total do sistema (Π).

$$\Pi = U - \Omega \quad (5)$$

Sabe-se que energia interna e o trabalho externo variam com os deslocamentos, desse modo podemos determinar o gradiente da Energia Potencial Total em relação aos deslocamentos nodais da estrutura, resultando em:

$$\nabla \Pi = \nabla U - \nabla \Omega \quad (6)$$

Os estados de equilíbrio dos sistemas estruturais e mecânicos são caracterizados pelos pontos fixos da energia potencial total. Através dos princípios variacionais, tem-se que a condição de equilíbrio é determinada quando a variação da Energia Potencial Total apresenta valor mínimo, ou seja, $\|\nabla \Pi\| = 0$.

$$\mathbf{0} = \nabla U - \nabla \Omega \quad (7)$$

Sabendo que a derivada do Trabalho Externo em relação aos seus deslocamentos retorna a força externa da estrutura, e utilizando o que já foi mencionado sobre o Primeiro Teorema de Castigliano (3), estabelecemos que a equação (8) representa o sistema de equações responsável por descrever o equilíbrio da estrutura:

$$0 = \frac{dU}{du_1} - Q_1, \quad 0 = \frac{dU}{du_2} - Q_2, \dots, \quad 0 = \frac{dU}{du_n} - Q_n \quad (8)$$

A resolução desse sistema consiste em encontrar os valores dos deslocamentos nodais, que podem ser determinados através de métodos numéricos como o método de Newton Raphson, por exemplo.

3.3 ELEMENTO DE BARRA PARA TRELIÇA PLANA

Segundo Beer e Jonston (2009) treliças planas são elementos retilíneos conectados por rótulas localizadas em suas extremidades (nós), projetadas para suportar as cargas que atuam em seu plano e agem diretamente sobre os seus nós. Por essas considerações, costuma-se admitir que as treliças são solicitadas apenas a cargas axiais de tração e compressão, sem a presença de momentos.

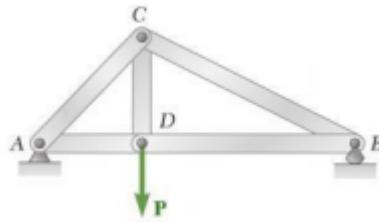


Figura 2 – Exemplo de Treliça.
Fonte: Beer e Jonston, 2009.

Tomando um sistema de coordenadas Global, como o apresentado a seguir, idealiza-se o nó de uma treliça plana sujeito a dois graus de liberdade: um relativo ao deslocamento vertical e outro ao deslocamento horizontal. Na figura abaixo, apresenta-se uma treliça de duas barras em um sistema cartesiano XY, com a indicação de seus graus de liberdade por nós.

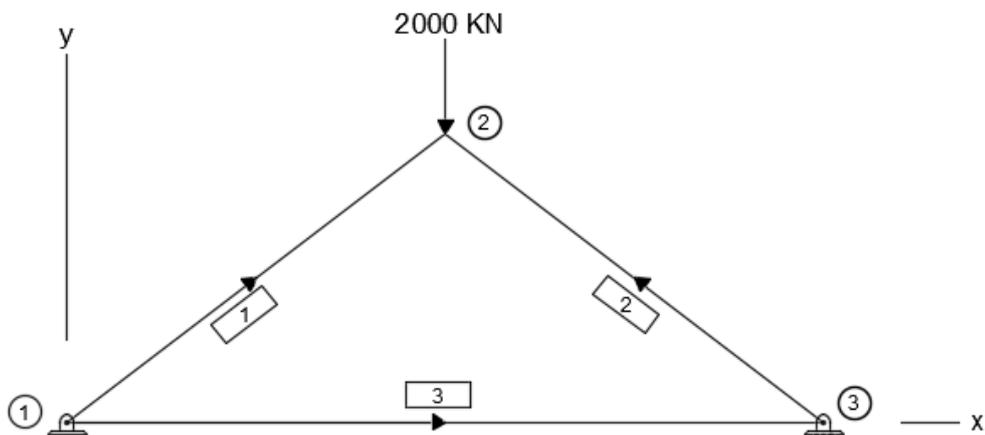


Figura 3 – Sistema de Referência Global.
Fonte: Autor.

3.3.1 CINEMÁTICA

Para determinação da energia de deformação (1) em uma barra de treliça partimos da análise da figura 4, onde temos um elemento em um sistema global XY, com sua posição antes e depois da aplicação de um carregamento genérico. Como condição do problema a barra se mantém reta todo o tempo:

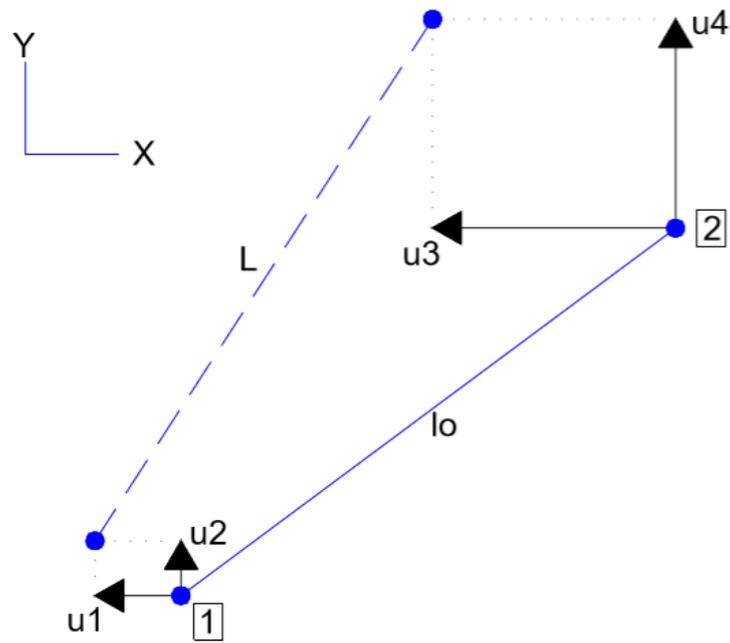


Figura 4 – Deslocamentos Nodais.
Fonte: Autor.

Pela figura 4 as coordenadas iniciais dos nós são:

$$\mathbf{x}_i = \begin{pmatrix} x1_i \\ x2_i \end{pmatrix} \text{ e } \mathbf{y}_i = \begin{pmatrix} y1_i \\ y2_i \end{pmatrix} \quad (9)$$

E as coordenadas finais:

$$\mathbf{x}_f = \begin{pmatrix} x1_f \\ x2_f \end{pmatrix} \text{ e } \mathbf{y}_f = \begin{pmatrix} y1_f \\ y2_f \end{pmatrix} \quad (10)$$

Que podem ser expressas por:

$$\mathbf{x}_f = \begin{pmatrix} x1_i \\ x2_i \end{pmatrix} + \begin{pmatrix} u1 \\ u3 \end{pmatrix} \quad (11)$$

$$\mathbf{y}_f = \begin{pmatrix} y1_i \\ y2_i \end{pmatrix} + \begin{pmatrix} u2 \\ u4 \end{pmatrix} \quad (12)$$

Os comprimentos l_0 e L , são respectivamente os comprimentos da barra antes e depois da deformação. Determinados por:

$$l_0 = \sqrt{(x2_i - x1_i)^2 + (y2_i - y1_i)^2} \quad (13)$$

$$L = \sqrt{(x2_f - x1_f)^2 + (y2_f - y1_f)^2} \quad (14)$$

Substituindo em (14) as relações (11) e (12), temos:

$$L = \sqrt{(u3 - u1 + x2_i - x1_i)^2 + (u4 - u2 + y2_i - y1_i)^2} \quad (15)$$

A variação do comprimento da barra é determinada por:

$$\Delta L = L - l_0 \quad (16)$$

Observa-se em (16) que a variação do comprimento da barra ΔL , é considerado positiva quando o comprimento da barra aumenta e negativo quando a mesma diminui. Da Mecânica dos Materiais podemos estabelecer ainda a deformação linear da barra como:

$$\varepsilon_{xx} = \frac{\Delta L}{l_0} \quad (17)$$

Nesse trabalho não se considera a não linearidade física do material, desse modo a relação constitutiva para essa barra é regida pela lei de Hooke:

$$\sigma = E \cdot \varepsilon_{xx} \quad (18)$$

A constante E é o módulo de elasticidade, ou módulo de Young, característico do material. Substituindo a equação (18) em (1) obteremos a energia de deformação interna da treliça:

$$U = \frac{1}{2} \int_V E \cdot \varepsilon_{xx}^2 dV \quad (19)$$

Como a seção transversal da barra é constante, o volume da barra prismática será dado por:

$$V = A \cdot x \quad (20)$$

E o diferencial do volume:

$$dV = A \cdot dx \quad (21)$$

Substituindo em (19):

$$U = \frac{1}{2} \int_0^{l_0} E \cdot \varepsilon_{xx}^2 \cdot A \cdot dx \quad (22)$$

Considerando-se que as barras são retas e não há cargas ao longo das mesmas, pode-se dizer que ε_{xx} é constante ao longo de cada barra. Resolvendo essa integral, temos a energia de deformação expressa em função da deformação e conseqüentemente em função dos deslocamentos:

$$U = \frac{E \cdot \varepsilon_{xx}^2 \cdot A \cdot l_0}{2} \quad (23)$$

Ao se considerar válida a equação (15) estará sendo feito uma análise não linear geometricamente exata, onde a relação entre forças e deslocamentos não ocorre de modo proporcional, e é válida a hipótese de grandes deformações gerando esforços de segunda ordem.

Para uma consideração de pequenos deslocamentos (linearidade geométrica), torna-se necessário linearizar a equação (15), transformando o comprimento pós deformação de uma função racional em uma função linear. Fazer esse procedimento é vantajoso, pois, sistemas de equações lineares possuem ferramentas menos custosas computacionalmente para sua resolução.

3.3.2 LINEARIZAÇÃO POR SÉRIE DE TAYLOR

Acerca da linearização por série de Taylor, Silva (2015) mostra que o polinômio de Taylor de grau n é o polinômio que interpola a função e suas derivadas até a ordem n no ponto A , desde que seja possível a função ser derivada pelo menos n vezes.

No nosso caso, não precisamos nos preocupar com o grau da derivada da equação (15), desse jeito basta encontrar o polinômio de Taylor de grau 1. A forma linearizada da função ficará:

$$L = (u1, u2, u3, u4) = \nabla L(u) \cdot (u - A) + L(u) \quad (24)$$

Adotando o vetor A como o ponto onde: $u1 = 0, u2 = 0, u3 = 0$ e $u4 = 0$.

$$L(0,0,0,0) = \sqrt{(x2_i - x1_i)^2 + (y2_i - y1_i)^2} = l_0 \quad (25)$$

Já o gradiente de L , pode ser determinado por:

$$\nabla L = \left(\frac{\partial L}{\partial u1} \quad \frac{\partial L}{\partial u2} \quad \frac{\partial L}{\partial u3} \quad \frac{\partial L}{\partial u4} \right) \quad (26)$$

Calculando o Gradiente no ponto A :

$$\frac{\partial L}{\partial u1} = \frac{(-x2_i + x1_i)}{l_0} \quad (27)$$

$$\frac{\partial L}{\partial u2} = \frac{(-y2_i + y1_i)}{l_0} \quad (28)$$

$$\frac{\partial L}{\partial u3} = \frac{(x2_i - x1_i)}{l_0} \quad (29)$$

$$\frac{\partial L}{\partial u4} = \frac{(y2_i - y1_i)}{l_0} \quad (30)$$

Com isso a forma linearizada de L , fica:

$$L(u1, u2, u3, u4) = \left(\frac{\partial L}{\partial u1} \quad \frac{\partial L}{\partial u2} \quad \frac{\partial L}{\partial u3} \quad \frac{\partial L}{\partial u4} \right) \cdot \left[\begin{pmatrix} u1 \\ u2 \\ u3 \\ u4 \end{pmatrix} - \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \right] + l_0 \quad (31)$$

Em sua forma final temos:

$$L(u1, u2, u3, u4) = \frac{u4 \cdot (y2_i - y1_i)}{l_0} + \frac{u2 \cdot (y1_i - y2_i)}{l_0} + \frac{u3 \cdot (x2_i - x1_i)}{l_0} + \frac{u1 \cdot (x1_i - x2_i)}{l_0} + l_0 \quad (32)$$

3.4 ELEMENTO DE BARRA DE PÓRTICO PLANO

Kassimali (2012) define um pórtico plano como uma sequência de membros conectados por conexões rígidas ou flexíveis, sujeita a carregamentos externos distribuídos ao longo de seu comprimento ou em suas conexões. Um nó de pórtico possui 3 graus de liberdade, correspondentes ao seu deslocamento vertical, horizontal e rotação.

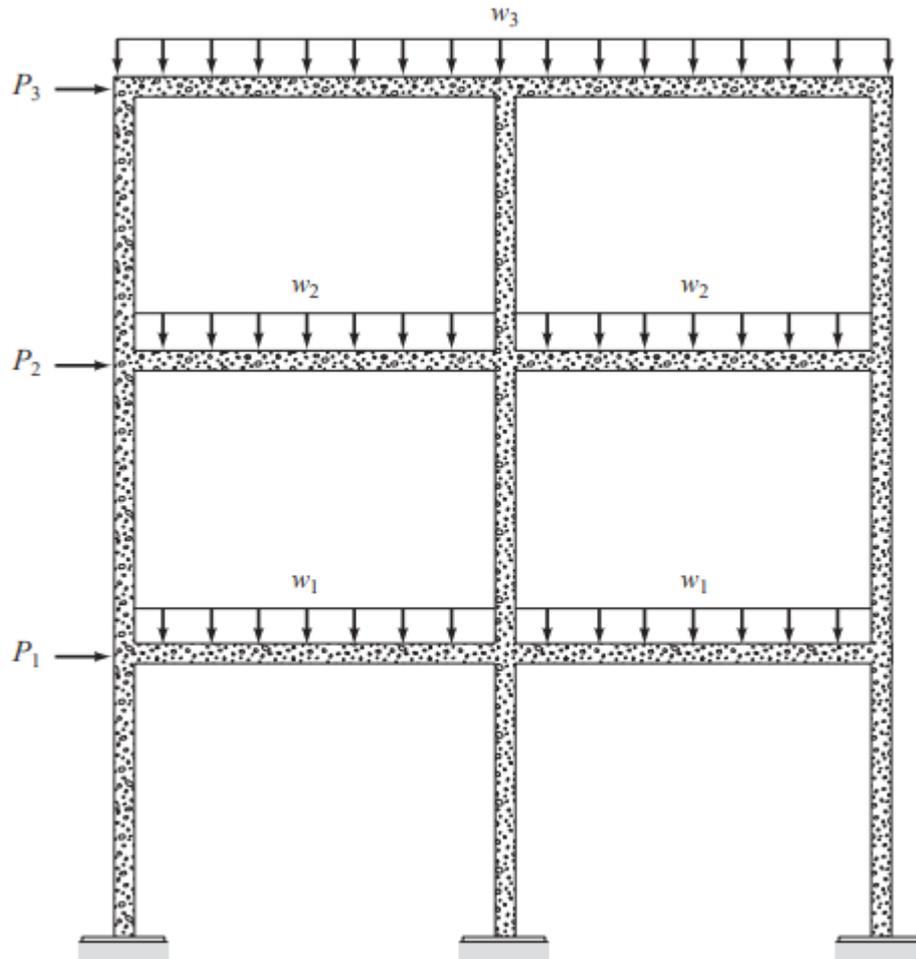


Figura 5 - Exemplo de Pórtico.
Fonte: Kassimali, 2012.

3.4.1 CINEMÁTICA

Adotando uma barra em um sistema de referência local, como ponto de partida para a determinação da energia de deformação (1) do elemento de pórtico, temos a figura 6 a seguir;

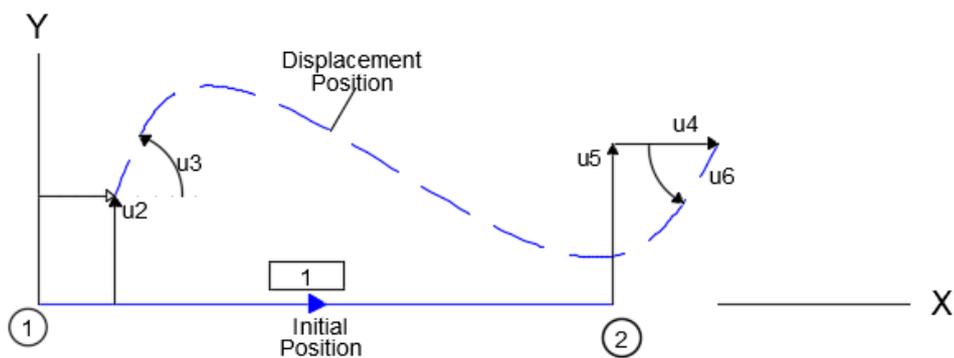


Figura 6 – Barra de Pórtico com deslocamentos nodais.
Fonte: Autor.

As considerações de partida para o elemento de barra estudado são:

- O corpo é contínuo e deformável, possuindo comprimento muito maior que as dimensões da seção transversal;
- Em cada elemento discretizado, a área da seção transversal é constante e o material é isotrópico;
- Adota-se a teoria de Euler-Bernoulli, que as seções transversais continuam planas e rígidas e as deformações por força cortante são desprezadas;
- Considera-se rotações moderadas e pequenas deformações horizontais, sendo assim, na configuração deformada o esforço normal provoca acréscimo a rigidez a flexão da estrutura.
- O deslocamento é definido por:

$$u_x = u_0 - d \cdot \frac{dv}{dx}, \text{ e } u_y = v \quad (33)$$

Ao considerar a teoria de rotações moderadas e pequenos deslocamentos, Zermiani (1998) define o tensor de deformação de Green-Lagrange por:

$$\varepsilon_{xx} = \varepsilon_L + \varepsilon_{NL} \quad (34)$$

Em que:

$$\varepsilon_L = \frac{du_0}{dx} - d \cdot \frac{d^2v}{dx^2} \quad (35)$$

$$\varepsilon_{NL} = \frac{1}{2} \cdot \left(\frac{dv}{dx} \right)^2 \quad (36)$$

Onde:

- $\frac{du_0}{dx}$ – é a deformação axial do centroide da seção;
- d – é a variação de coordenadas no eixo paralelo à altura da seção;
- $\frac{dv}{dx}$ – é a rotação da seção transversal.

Assim como no caso das treliças, ao adotar material elástico linear é válida a lei de Hooke, onde a tensão é representada pela equação (18), substituindo (18) em (1), resulta:

$$U = \frac{1}{2} \int E \cdot \varepsilon_{xx}^2 \cdot dV \quad (37)$$

Substituindo (34) em (37):

$$U = \frac{1}{2} \int E \cdot (\varepsilon_L + \varepsilon_{NL})^2 \cdot dV \quad (38)$$

Expandindo o produto notável da equação (38):

$$U = \left(\frac{E}{2} \right) \cdot \int \varepsilon_L^2 + 2 \cdot \varepsilon_L \cdot \varepsilon_{NL} + \varepsilon_{NL}^2 \cdot dV \quad (39)$$

Das regras do cálculo diferencial temos que a integral da soma é a soma das integrais, e que o diferencial do volume é equivalente a:

$$dV = dA \cdot dx \quad (40)$$

O termo não linear da integral (39) fica definido por:

$$\frac{E}{2} \cdot \iint \varepsilon_{NL}^2 \cdot dAdx \quad (41)$$

$$\frac{E}{2} \cdot \iint \left(\frac{1}{2} \cdot \left(\frac{dv}{dx} \right)^2 \right)^2 \cdot dAdx \quad (42)$$

$$\frac{E}{2} \cdot \iint \frac{1}{4} \cdot \left(\frac{dv}{dx} \right)^4 \cdot dAdx \quad (43)$$

$$\iint \varepsilon_{NL}^2 \cdot dAdx = \frac{E \cdot A}{8} \cdot \int \left(\frac{dv}{dx} \right)^4 dx \quad (44)$$

Para o termo linear teremos:

$$\frac{E}{2} \cdot \iint \varepsilon_L^2 \cdot dAdx \quad (45)$$

$$\frac{E}{2} \cdot \iint \left(\frac{du_0}{dx} - d \cdot \left(\frac{dv}{dx} \right) \right)^2 \cdot dAdx \quad (46)$$

$$\frac{E}{2} \cdot \iint \left(\frac{du_0}{dx} \right)^2 - 2 \cdot d \cdot \frac{du_0}{dx} \cdot \frac{d^2v}{dx^2} + \left(d \cdot \frac{d^2v}{dx^2} \right)^2 dAdx \quad (47)$$

Da Mecânica dos Materiais temos que:

$$\int d^2 \cdot dA = I \text{ e } \int d \cdot dA = 0 \quad (48)$$

$$\frac{E}{2} \cdot \iint \varepsilon_L^2 \cdot dAdx = \frac{E}{2} \cdot \int A \cdot \left(\frac{du_0}{dx} \right)^2 + I \cdot \left(\frac{d^2v}{dx^2} \right)^2 dx \quad (49)$$

E o termo de interação entre a deformação não linear e linear resulta em:

$$\frac{E}{2} \cdot \iint 2 \cdot \varepsilon_{NL} \cdot \varepsilon_L dAdx \quad (50)$$

$$\frac{E}{2} \cdot \iint \frac{du_0}{dx} \left(\frac{dv}{dx} \right)^2 - d \cdot \frac{d^2v}{dx^2} \cdot \left(\frac{dv}{dx} \right)^2 dAdx \quad (51)$$

$$\frac{E}{2} \cdot \iint 2 \cdot \varepsilon_{NL} \cdot \varepsilon_L dAdx = \frac{E}{2} \cdot \int A \cdot \varepsilon_0 \cdot \left(\frac{dv}{dx} \right)^2 dx \quad (52)$$

Utilizando uma mudança de variável ξ , em que ξ varia de [-1;1] para aplicação na Integração Numérica de Gauss, que será explicada mais à frente:

$$\xi = \frac{2x - x1 - x2}{(x2 - x1)} \quad (53)$$

Consequentemente:

$$d\xi = \frac{2}{l} dx \quad (54)$$

Substituindo a transformação nas integrais:

$$\frac{E}{2} \cdot \iint 2 \cdot \varepsilon_{NL} \cdot \varepsilon_L dAdx = \frac{2A \cdot \varepsilon_0 \cdot E}{l^2} \int \left(\frac{dv}{d\xi} \right)^2 d\xi \quad (55)$$

$$\frac{E}{2} \cdot \iint \varepsilon_L^2 \cdot dAdx = E \int \frac{A}{l} \cdot \left(\frac{du_0}{d\xi} \right)^2 + \frac{4 \cdot I}{l^3} \cdot \left(\frac{d^2v}{d\xi^2} \right)^2 d\xi \quad (56)$$

$$\frac{E}{2} \cdot \iint \varepsilon_{NL}^2 \cdot dAdx = \frac{E}{l^3} \int A \cdot \left(\frac{dv}{d\xi} \right)^4 d\xi \quad (57)$$

A rotação elevada a quarta potência torna-se um valor muito pequeno podendo ser desprezado, somando as equações (54 e 55) e substituindo em (39), temos a equação da energia de deformação para um caso não linear, considerando pequenas deformações e rotações moderadas. Os campos de deslocamentos u e v são determinados em Chandrupatia (2014), conforme mostrado a seguir:

$$u = N1 \cdot u1 + N2 \cdot u4 \quad (58)$$

$$v = H1 \cdot u2 + H2 \cdot u3 \cdot \frac{l}{2} + H3 \cdot u5 + H4 \cdot u6 \cdot \frac{l}{2} \quad (59)$$

Onde:

- $N1$ e $N2$ – Polinômios interpoladores de Lagrange;
- $H1, H2, H3$ e $H4$ – Polinômios interpoladores de Hermite;

Os polinômios interpoladores (de Lagrange e Hermite) são as chamadas funções de forma, e definidos para o elemento de 2 nós como:

$$N1 = \frac{1 - \xi}{2} \quad (60)$$

$$N2 = \frac{1 + \xi}{2} \quad (61)$$

$$H1 = \frac{1}{4} (2 - 3\xi + \xi^3) \quad (62)$$

$$H2 = \frac{1}{4} (1 - \xi - \xi^2 + \xi^3) \quad (63)$$

$$H3 = \frac{1}{4} (2 + 3\xi - \xi^3) \quad (64)$$

$$H4 = \frac{1}{4} (-1 - \xi + \xi^2 + \xi^3) \quad (65)$$

Caso queira se fazer uma análise linear da estrutura, elimina-se o termo ε_{NL} da equação (34), resultando assim em:

$$U = E \int \frac{A}{l} \cdot \left(\frac{du_0}{d\xi} \right)^2 + \frac{4 \cdot I}{l^3} \cdot \left(\frac{d^2v}{d\xi^2} \right)^2 d\xi \quad (66)$$

Essas integrais podem ser resolvidas por processos numéricos como a Quadratura de Gauss.

3.5 QUADRATURA DE GAUSS

A Quadratura Gaussiana consiste em encontrar o valor de uma integral a partir de valores do integrando $f(\xi)$ obtidos em n pontos específicos, ponderados.

$$\int_{-1}^1 f(\xi) d\xi = \sum wt \cdot f(t_n) \quad (67)$$

Onde:

- wt – São os pesos da interpolação de Gauss;
- t_n – Pontos analíticos de Gauss.

Para o caso de polinômios a integração retorna valores exatos, desde que o grau do polinômio seja menor ou igual a $2n + 1$.

Em caso de maior aprofundamento indica-se ao leitor, as bibliografias de Ruggiero e Lopes (2009) e de Justo *et al* [s. d.].

3.6 MÉTODO DE NEWTON RAPHSON

Determinadas as equações de equilíbrio, têm-se um sistema de equações (7) cuja solução indica a posição de equilíbrio da estrutura analisada. Dentre os métodos disponíveis na literatura, destaca-se o método de Newton Raphson. Ruggiero e Lopes (2009), detalham a dedução do método de Newton Raphson, assim como o seu algoritmo. Abaixo apresenta-se um algoritmo adaptado.

Seja o sistema de equações de múltiplas variáveis $\mathbf{F}(\mathbf{x}) = \mathbf{0}$ e seu respectivo Jacobiano $\mathbf{J}(\mathbf{x}) = \nabla \mathbf{F}$, a solução do problema pode ser encontrada por um procedimento iterativo. Sendo garantidas as condições iniciais \mathbf{x}^k para cada iteração e as medidas de erro máximo ε :

- Calcula-se $\mathbf{J}(\mathbf{x}^k)$ e $\mathbf{F}(\mathbf{x}^k)$;
- Obtém-se \mathbf{s}^k solução de $\mathbf{J}(\mathbf{x}^k) \times \mathbf{s}^k = -\mathbf{F}(\mathbf{x}^k)$;
- Calcula $\mathbf{x}^{k+1} = \mathbf{x}^k + \mathbf{s}^k$;
- Se $\sqrt{\frac{\sum(\mathbf{s}^k)^2}{\sum(\mathbf{x}^k)^2}} < \varepsilon$, então \mathbf{x}^{k+1} é a solução do problema;
- Se não, $k = k + 1$ e repete-se o procedimento.

O jacobiano calculado para solução do sistema de equações, em análise de estruturas possui o significado físico de relacionar forças e deslocamentos, sendo definido como Matriz de Rigidez Tangente.

O teste de convergência adotado $\left(\sqrt{\frac{\sum (s^k)^2}{\sum (x^k)^2}} < \varepsilon \right)$ está ligado a variação dos deslocamentos nodais em cada iteração. Segundo Kassimali (2012) esse critério costuma trazer resultados razoavelmente precisos para estruturas que apresentam redução de rigidez quando submetidas a carregamentos maiores.

4 DERIVAÇÃO AUTOMÁTICA

Ao se analisar o procedimento numérico empregado pelo método de Newton, temos a necessidade do cálculo do jacobiano do sistema $J(x^k)$. Ruggiero e Lopes (2009), indicam que encontrar as derivadas parciais apresenta-se justamente como a dificuldade do algoritmo de Newton, sendo necessário implementar as derivadas manualmente ou utilizar as diferenças finitas.

Conforme discutido anteriormente, essas duas abordagens tornam-se desvantajosas visto que, o primeiro método está condicionado diretamente a dificuldade de realizar a implementação manual da derivada, e o segundo apresenta-se como pouco eficiente.

Uma das maneiras de contornar esses problemas é através da utilização da Derivação Automática. Neste capítulo será abordado as características da AD, apresentando sua metodologia e exemplos.

4.1 FUNÇÕES COMPOSTAS

Assim como os números reais as funções algébricas podem ser somadas, multiplicadas, subtraídas e divididas. A utilização dessas propriedades serve para gerar novas funções a partir de funções já existentes. Como exemplo, tomemos $f(x) = \sqrt{x}$ e $g(x) = x^2 + 2$, ao somarmos ou dividirmos f e g encontraremos uma nova função h definida por:

$$h(x) = f(x) + g(x) = \sqrt{x} + x^2 + 2 \quad (68)$$

$$h(x) = \frac{f(x)}{g(x)} = \frac{\sqrt{x}}{x^2 + 2} \quad (69)$$

Para que h exista é necessário que o valor adotado para x pertença ao domínio de f e g , desse modo temos que o domínio de uma função gerada por combinação é igual a intersecção dos domínios das funções que a compõem.

Além das quatro operações fundamentais anteriormente citadas, existe outra maneira possível de se combinar duas funções. Por exemplo, tomemos agora $f(u) = \sqrt{u}$ e $u = g(x) = x^2 + 2$, substituindo u por $g(x)$ em f obtemos:

$$f(u) = \sqrt{u} = \sqrt{g(x)} = \sqrt{x^2 + 2} \quad (70)$$

Pode se notar que através dessa manipulação a função f passou a ser dependente de $x^2 + 2$, esse processo é chamado de composição de funções, e a função f dá-se o nome de função composta. A representação algébrica de uma composição é definida segundo Stewart (2013), por:

$$(f \circ g)(x) = f(g(x)) \quad (71)$$

Diferentemente do que ocorre com as funções geradas por outros tipos de combinação, uma função composta f existe se x pertence ao domínio de g , e a imagem de $g(x)$ pertence ao domínio de f .

4.2 DERIVADAS DE FUNÇÕES COMPOSTAS

As fórmulas de derivação do cálculo diferencial são bem definidas para o caso de funções intrínsecas (seno, cosseno, exponencial, potenciação, entre outras) e de operações (soma, subtração, multiplicação e divisão).

Para as funções compostas como (70), não existe nenhuma fórmula que possa ser aplicada para determinação de suas derivadas diretamente. Nesses casos necessita-se da utilização de uma das principais regras de derivação, a Regra da Cadeia.

Em seu livro de cálculo diferencial, Stewart (2013) demonstra que “a derivada de uma função composta $f \circ g$ é o produto das derivadas de f e g ”, sendo essa sua formulação para a regra da cadeia. No caso da função (70):

$$(f \circ g)'(x) = f'(g(x)) \cdot g'(x) \quad (72)$$

$$(f \circ g)'(x) = \frac{1}{2\sqrt{g(x)}} \cdot (2x) \quad (73)$$

4.3 INTRODUÇÃO AOS GRAFOS

Grafos são diagramas formados por linhas que relacionam um grupo de objetos dois a dois pertencentes a um conjunto. Seu uso é aplicado a muitas situações das engenharias, computação, matemática, entre outras áreas da ciência.

Segundo Feofiloff et al. (2005 *apud* Ando, 2008) um grafo é um par (V, A) em que V é um conjunto arbitrário e A é um subconjunto de pares ordenados de V . Os elementos de V são chamados vértices ou nós, e os elementos de A são chamados arestas ou arcos.

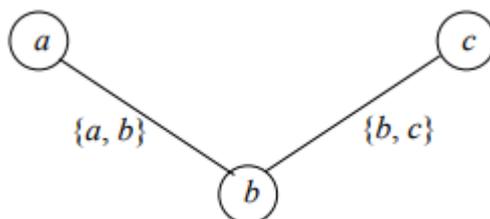


Figura 7 – Grafo com os vértices $(a;b;c)$ e arestas: $[(a,b);(c,b)]$.

Fonte: Costa Junior, 2003.

Em alguns casos as arestas que unem dois nós possuem direção associada indicada por uma seta na ponta da linha, a esse tipo de grafo dá-se o nome de dígrafo (grafo orientado). Além destes, temos ainda as árvores, que são grafos simples e acíclicos onde seus dados são organizados em sequência hierárquica.

Esses conceitos básicos de Teoria dos Grafos são fundamentais para o entendimento da derivação algorítmica, pois toda função por mais complexa que seja pode ser descrita através de uma árvore composta por operações algébricas. Como exemplo, tomemos a função:

$$y = \frac{x_1 \cdot \tan(x_1 \cdot x_2)}{x_2 - \tan(x_1 \cdot x_2)} \quad (74)$$

Tomando v_i como as operações algébricas que constituem os nós da árvore obtemos:

$$\begin{aligned} v_1 &= x_1 & v_5 &= v_4 \cdot v_1 \\ v_2 &= x_2 & v_6 &= v_2 - v_4 \\ v_3 &= v_1 \cdot v_2 & v_7 &= \frac{v_5}{v_6} \\ v_4 &= \tan(v_3) & y &= v_7 \end{aligned}$$

O que em forma de diagrama representa:

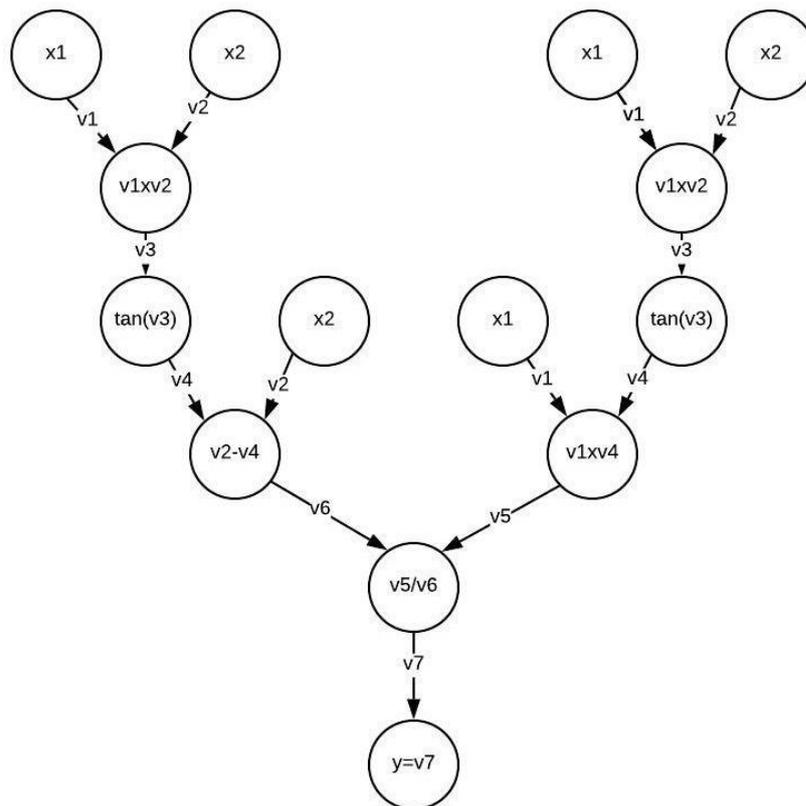


Figura 8 – Árvore da função y .
Fonte: Autor.

Ao se trabalhar com uma função através das árvores, passamos a lidar com uma sequência de funções compostas. Para a AD isso é fundamental, pois torna possível computar a derivada da função principal a partir das derivadas de seus nós, necessitando calcular apenas as derivadas de operações algébricas ($;$, $+$, $-$, $/$, *etc*) e funções elementares (seno, cosseno, tangente, entre outras).

4.4 ESTRATÉGIAS DE DERIVAÇÃO AUTOMÁTICA

Existe basicamente duas estratégias para determinar derivadas de modo automático: método direto (*forward mode*) e o método reverso (*reverse mode*). A diferença entre ambas está diretamente relacionada ao tratamento que cada processo utiliza do grafo da função principal.

4.4.1 MÉTODO DIRETO

No método direto ocorre a avaliação das derivadas no mesmo sentido do grafo, com isso as variáveis dependentes v_i são determinadas junto aos seus gradientes. Tomando o nó v_5 do grafo computacional da figura 8, o método direto avalia a derivada nesse ponto através da regra da cadeia por:

$$\nabla v_5 = \nabla v_4 \cdot v_1 + \nabla v_1 \cdot v_4 \quad (75)$$

Como ∇v_4 e ∇v_1 são vetores, a determinação de ∇v_5 pelo modo direto torna-se então a combinação linear desses dois vetores. Segundo Birgin (1998), o cálculo dessas combinações lineares apresenta-se como a tarefa mais custosa do método direto, para escapar de tal inconveniente se o gradiente da função em questão possui poucas entradas não-nulas, pode se trabalhar com vetores esparsos ou com as derivadas parciais de modo independente, porém para isso é necessário utilizar uma subrotina para cada derivada parcial aumentando assim o tempo de processamento.

O grafo a seguir ilustra o processo de obtenção das derivadas parciais da função (74), pelo método direto, onde v'_i representa o gradiente da função:

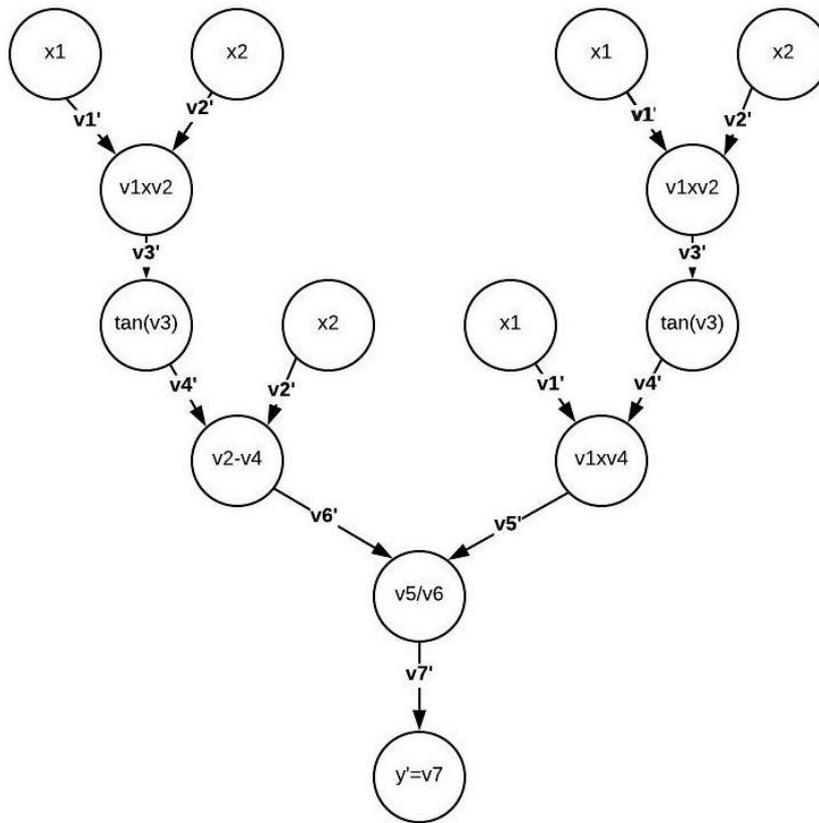


Figura 9 – Sequência da AD no método direto.

Fonte: Autor.

Por variáveis independentes o cálculo de $\frac{dy}{dx1}$ segue a sequência da figura 9, substituindo os gradientes pela derivada parcial analisada, como pode ser visto a seguir:

$$\begin{aligned}
 \frac{dv_1}{dx1} &= 1 & \frac{dv_5}{dx1} &= \frac{dv_4}{dx1} \cdot v_1 + \frac{dv_1}{dx1} \cdot v_4 \\
 \frac{dv_2}{dx1} &= 0 & \frac{dv_6}{dx1} &= \frac{dv_2}{dx1} - \frac{dv_4}{dx1} \\
 \frac{dv_3}{dx1} &= \frac{dv_2}{dx1} \cdot v_1 + \frac{dv_1}{dx1} \cdot v_2 & \frac{dv_7}{dx1} &= \frac{v_6 \cdot \frac{dv_5}{dx1} - v_5 \cdot \frac{dv_6}{dx1}}{v_6^2} \\
 \frac{dv_4}{dx1} &= \sec^2(v_3) \cdot \frac{dv_3}{dx1} & \frac{dy}{dx1} &= \frac{dv_7}{dx1}
 \end{aligned}$$

Devido as suas características o método direto torna-se computacionalmente mais eficiente (acurado e com menor tempo de processamento) que o modo reverso para casos de funções com muitos nós dependentes e poucas variáveis de entrada.

4.4.2 MÉTODO REVERSO

No modo reverso, computa-se o valor da derivada da função a partir da direção inversa da árvore. Sendo assim se faz necessário correr o gráfico em dois sentidos, uma vez no sentido

do fluxo para cálculo das funções compostas e uma vez no sentido inverso para determinação das derivadas.

Em geral, o método reverso pode ser expresso pela expressão seguinte:

$$\frac{dy}{dx_j} = \sum_{i=1}^c \prod \frac{dy}{dv_k} \text{ em que } j \text{ são as derivadas parciais elementares de } \frac{dy}{dx_j}.$$

A ilustração do funcionamento do *reverse mode*, inicia-se pelo armazenamento dos caminhos que a variável analisada percorre. Para nossa função (74), ambos os caminhos de x_1 são representados a seguir:

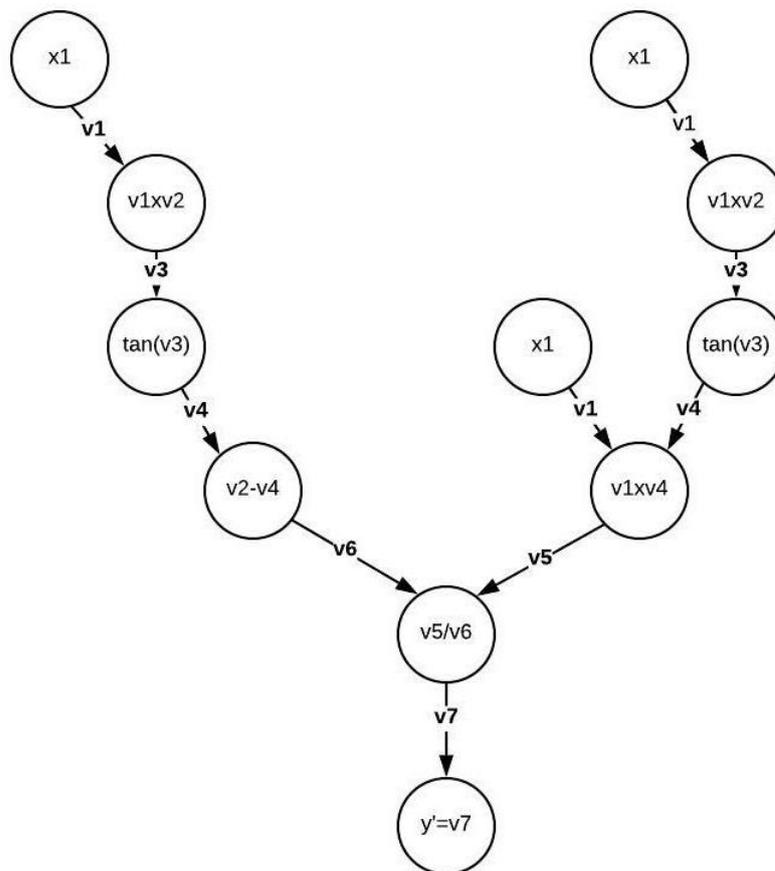


Figura 10 – Grafo de todos os caminhos da variável x_1 .

Fonte: Autor.

Analisando a figura 10 pode se constatar que a variável x_1 está ligada a 3 trajetos: c_1 (v_1, v_3, v_4, v_6 e v_7), c_2 (v_1, v_5 e v_7) e c_3 (v_1, v_3, v_4, v_5 e v_7). Determinados os caminhos calcula-se o produtório das derivadas parciais de cada sequência no seu sentido inverso:

➤ Caminho c_1 :

$$\frac{dv_7}{dv_6} = -\frac{v_5}{v_6^2} \qquad \frac{dv_6}{dv_4} = -1$$

$$\begin{aligned} \frac{dv_4}{dv_3} &= \sec^2(v_3) & \frac{dv_1}{dx_1} &= 1 \\ \frac{dv_3}{dv_1} &= v_2 \end{aligned}$$

Sendo o produtório de c_1 definido por:

$$\prod c_1 = \left(\frac{dv_7}{dv_6}\right) \left(\frac{dv_6}{dv_4}\right) \left(\frac{dv_4}{dv_3}\right) \left(\frac{dv_3}{dv_1}\right) \left(\frac{dv_1}{dx_1}\right) \quad (76)$$

➤ Caminho c_2 :

$$\frac{dv_7}{dv_5} = \frac{1}{v_6} \qquad \frac{dv_5}{dv_1} = v_4 \qquad \frac{dv_1}{dx_1} = 1$$

Sendo o produtório de c_2 definido por:

$$\prod c_2 = \left(\frac{dv_7}{dv_5}\right) \left(\frac{dv_5}{dv_1}\right) \left(\frac{dv_1}{dx_1}\right) \quad (77)$$

➤ Caminho 3:

$$\begin{aligned} \frac{dv_7}{dv_5} &= \frac{1}{v_6} & \frac{dv_4}{dv_3} &= \sec^2(v_3) & \frac{dv_1}{dx_1} &= 1 \\ \frac{dv_5}{dv_4} &= v_1 & \frac{dv_3}{dv_1} &= v_2 \end{aligned}$$

Sendo o produtório de c_3 definido por:

$$\prod c_3 = \left(\frac{dv_7}{dv_5}\right) \left(\frac{dv_5}{dv_4}\right) \left(\frac{dv_4}{dv_3}\right) \left(\frac{dv_3}{dv_1}\right) \left(\frac{dv_1}{dx_1}\right) \quad (78)$$

Ao obtermos todos os produtórios da função principal, basta realizarmos a sua soma para a determinação da derivada parcial de y em relação a x_1 :

$$\frac{dy}{dx_1} = \prod c_1 + \prod c_2 + \prod c_3$$

(79)

Devido a essa necessidade de analisar todos os caminhos da variável, o método reverso torna-se mais vantajoso computacionalmente que o modo direto conforme diminui o número de nós dependentes da função.

4.5 UMA FERRAMENTA DE DIFERENCIAÇÃO AUTOMÁTICA

Nesta seção apresenta-se o funcionamento da ferramenta de diferenciação computacional DAALGPY (*Python Algoritmo de Diferenciação Automática*), desenvolvida pelo autor. O pacote DAALGPY é um algoritmo de diferenciação computacional implementada em linguagem Python 3x, utilizando o método direto e o método reverso para determinação de gradientes.

O algoritmo aceita que os dados de entrada sejam sistemas de equações ou funções individuais, sendo necessário que cada equação esteja representada por meio de uma *string* ou uma função. A semântica das funções implementadas segue o mesmo padrão do módulo nativo *math*, sendo possível utilizar:

- Operadores binários: adição (+), subtração (−), divisão (/) e multiplicação (*);
- Funções trigonométricas: seno (*sin(·)*), cosseno (*cos(·)*), tangente (*tan(·)*), secante (*sec(·)*), cossecante (*cossec(·)*) e cotangente (*cotg(·)*);
- Funções elementares: exponencial (*exp(·)*), potenciação (**), logaritmo neperiano (*log(·)*) e radiciação (*sqrt(·)*).

```
7 import DAALGPY.Gradient as gr
8
9 X = 'x1 * (x2 ** 2) - x3 ** 2'
10
11 def Y(x1,x2,x3):
12     return log(x1 * x2) ** 2 + x2 * (x3 ** 2) - a - x2
13
14 def Z(x1,x2,x3):
15     f1 = - x2 * x3 + a
16     return sqrt(log(x1 * x2) + (x2/x3)) + f1
17 variaveis = ['x1', 'x2', 'x3']
18 constantes = ['a']
19 a = gr.GradientRM([X,Y,Z],variaveis,constantes,'teste.py')
```

Figura 11 – Entrada de dados no módulo.

Fonte: Autor.

Na figura 11, temos um exemplo de uso do pacote de derivação, onde:

- Linha 7 – importa o módulo de derivação do DAALGPY;
- Linha 9 – introduz a equação X, em formato de *string*;

- Linha 11 e 12 – constrói a função Y;
- Linha 14 a 15 – constrói a função Z;
- Linha 17 – define a lista com as variáveis a serem analisadas;
- Linha 18 – define a lista com as constantes do problema;
- Linha 19 – utiliza o módulo DAALGPY, através do modo reverso (*reverse mode*). Da esquerda para direita temos: uma lista contendo todas as equações que serão derivadas, as variáveis, as constantes do problema e a *string* final indicando o nome do arquivo *py*, gerado para calcular o valor numérico do jacobiano.

Ao introduzir funções ou strings, o algoritmo utiliza do módulo nativo *dis* para interpretar e transformar o Bytecode do objeto (função ou string) em arquivo txt. O módulo *dis*, retorna uma representação simples para compreensão humana do modo como a máquina interpreta o código trabalhado. A sua estrutura apresenta 5 colunas, das quais:

- A primeira indica a linha inicial daquela função ou string;
- A segunda representa o endereço em máquina daquela instrução bytecode;
- A terceira o nome da instrução bytecode;
- A quarta coluna traz o índice em que aquele argumento está alocado na tabela de variáveis e constantes;
- A última coluna o nome da variável no código humano.

1	21	0	LOAD_FAST	1	(x2)
2		2	UNARY_NEGATIVE		
3		4	LOAD_FAST	2	(x3)
4		6	BINARY_MULTIPLY		
5		8	LOAD_GLOBAL	0	(a)
6		10	BINARY_ADD		
7		12	STORE_FAST	3	(f1)
8					
9	22	14	LOAD_GLOBAL	1	(sqrt)
10		16	LOAD_GLOBAL	2	(log)
11		18	LOAD_FAST	0	(x1)
12		20	LOAD_FAST	1	(x2)
13		22	BINARY_MULTIPLY		
14		24	CALL_FUNCTION	1	
15		26	LOAD_FAST	1	(x2)
16		28	LOAD_FAST	2	(x3)
17		30	BINARY_TRUE_DIVIDE		
18		32	BINARY_ADD		
19		34	CALL_FUNCTION	1	
20		36	LOAD_FAST	3	(f1)
21		38	BINARY_ADD		
22		40	RETURN_VALUE		

Figura 12 – Saída do Bytecode da equação Z.
Fonte: Autor.

Os grafos são construídos através da interpretação do arquivo txt, os dados gerados pelo módulo *dis* são compreendidos através da análise por Notação Polonesa Reversa. Porém para utilização nas próximas etapas do DAALGPY, faz-se necessário a sua conversão para notação algébrica, onde torna-se mais simples a aplicação da regra da cadeia. Após a conversão para cada função $f_i (\mathbb{R}^n \rightarrow \mathbb{R})$, cria-se um dicionário responsável por guardar o grafo computacional construído como explicado em 4.3.

```
{'t0': 'x1', 't1': 'x2', 't2': 'x3', 't3': '(-1)', 't4': '(t3*t1)',
't5': '(t4*t2)', 't6': '(t5+a)', 't7': '(t0*t1)', 't8': '(log(t7))',
't9': '(t1/t2)', 't10': '(t8+t9)', 't11': '(sqrt(t10))', 't12':
'(t11+t6)'}
```

Figura 13 – Vértices do grafo computacional armazenado em um dicionário para equação Z.
Fonte: Autor.

Determinado o grafo ocorre uma bifurcação no tratamento dos dados conforme método de diferenciação escolhido. No modo direto serão criadas listas que indicam de quais variáveis esse nó é dependente. Enquanto no modo reverso as listas irão conter os caminhos percorridos por cada variável.

```
{'t2': ['t2'], 't1': ['t1'], 't0': ['t0'], 't3': ['t1'], 't4': ['t1', 't2'], 't5':
['t1', 't2'], 't6': ['t1', 't2'], 't7': ['t1', 't2'], 't8': ['t1', 't2'], 't9':
['t0', 't1'], 't10': ['t0', 't1'], 't11': ['t1', 't0', 't2'], 't12': ['t1', 't0',
't2'], 't13': ['t1', 't0', 't2']}
```

Figura 14 – Lista no modo direto.
Fonte: Autor.

```
{'t0': [[9, 10, 11, 12, 13]], 't1': [[3, 4, 5, 6, 13], [7, 8, 11, 12, 13], [9, 10, 11, 12, 13]], 't2': [[4, 5, 6, 13], [7, 8, 11, 12, 13]]}
```

Figura 15 – Caminhos no modo reverso.

Fonte: Autor.

A partir dos caminhos e dependências, calcula-se as derivadas de todos os nós sejam elas no modo reverso ou direto. Para a implementação do método direto percorre-se o grafo uma vez para a determinação de cada derivada, semelhante ao que Birgin (1998) indica para os casos de gradientes com poucas entradas não nulas. Infelizmente essa estratégia acabou por diminuir a eficiência do modo direto sob o ponto de vista de tempo de processamento.

O último passo do módulo está ligado a criar as saídas da função que ocorrem de dois modos. O primeiro através de uma lista contendo em seu interior as respectivas derivadas simbólicas. Enquanto no segundo é criado um arquivo *.py* contendo um algoritmo responsável por avaliar as derivadas numericamente, conforme a técnica de derivação escolhida.

```
[[ '(0.5*(((log((x1*x2)))+(x2/x3)))**(-0.5))*1/((x1*x2))*(x2)',
  '(x3)*(-1)+(0.5*(((log((x1*x2)))+(x2/x3)))**(-0.5))*1/((x1*x2))*(x1)+(0.5*(((log((x1*x2)))+(x2/x3)))**(-0.5))*1/(x3)',
  '((-1)*x2)+(0.5*(((log((x1*x2)))+(x2/x3)))**(-0.5))*((-1)*x2/(x3**2))' ]]
```

Figura 16 – Saída de dados como variáveis simbólicas para o caso da figura 11.

Fonte: Autor.

5 METODOLOGIA PARA VALIDAÇÃO DOS RESULTADOS

O pacote DAALGPY foi validado através da sua aplicação em um algoritmo de análise estrutural de treliças e pórticos planos, desenvolvido em linguagem Python. Foram analisados 3 treliças e 3 pórticos por análise elástica-linear e não linear geométrica, e seus resultados foram comparados com os de outros autores.

Para a análise, foi utilizado um processo incremental iterativo através de passos de carga, obtendo assim resultados mais expressivos sobre o comportamento das estruturas. Para cada incremento de carga, o código utiliza o método de Newton-Raphson para resolver o sistema de equações sejam elas lineares ou não. Ao se atingir um valor de convergência considerado satisfatório passa-se para o próximo valor de carga, até que o ponto de parada seja alcançado. Como o método de Newton necessita de um chute inicial na sua primeira iteração, considera-se sempre o valor final dos deslocamentos do passo anterior como ponto inicial da carga atual. Quando não ocorrer convergência do passo anterior o chute inicial será obtido como o resultado da análise linear.

São necessários como parâmetros de entrada para a análise: as coordenadas dos nós, as cargas solicitantes da estrutura, suas condições de apoio, a conectividade dos nós, a área da seção transversal, inércia e Módulo de Young de cada barra, também podem ser ajustados o número desejado de passos de carga, número máximo de iterações e a tolerância permitida para a análise.

Todos os processos de utilização de derivadas foram realizados a partir do pacote de AD, sendo as matrizes de rigidez e os vetores de cargas internas da estrutura obtidos a partir da derivação da energia de deformação. Para os casos de treliças utilizou-se a consideração (15) para análise não linear geometricamente exata e (32) para casos lineares, no caso dos pórticos planos não lineares com rotações moderadas (55, 56) e para análise linear (66). Além disso, as derivadas dos campos de deslocamento (u e v) também foram determinadas através do DAALGPY.

Para o caso de pórticos foi necessário a criação de um algoritmo para a realização da integração numérica de Gauss. Como o código de análise não linear e linear é o mesmo, teve que ser adotado o número de pontos de Gauss que contemple a integração do polinômio de maior grau, conforme estabelecido em 3.5. Ao se elevar a primeira derivada de v ao quadrado

acaba-se gerando um polinômio de quarto grau, desse modo, a integração numérica adotada foi a de 3 pontos de Gauss, com os valores descritos a seguir:

n	t_j	w_j
1	0	2
2	$\pm \frac{\sqrt{3}}{3}$	1
3	0	$\frac{8}{9}$
	$\pm \sqrt{\frac{3}{5}}$	$\frac{5}{9}$

Figura 17 – Tabela de pontos para integração de Gauss.

Fonte: Justo et al, [s. d.].

As matrizes de rigidez geradas com o pacote de AD foram comparadas com uma implementação das matrizes analíticas definidas por Kassimali (2012) e Zermiani (1998). Além disso, buscou-se comparar os resultados dos deslocamentos nodais e esforços solicitantes das estruturas.

6 RESULTADOS E DISCUSSÃO

6.1 TRELIÇA PLANA DE TRÊS BARRAS – ANÁLISE NÃO LINEAR

Este exemplo proposto por Kassimali (2012) consiste em uma treliça plana formada por três barras, constituídas por um material elástico linear com módulo de Young $E = 70 \text{ GPa}$ e área de seção transversal $A = 645,2 \text{ mm}^2$. A estrutura é sujeita a uma carga concentrada vertical para baixo $P_l = 2000 \text{ KN}$ no nó 2. A Figura 17 abaixo apresenta o desenho esquemático e o seu modelo analítico.

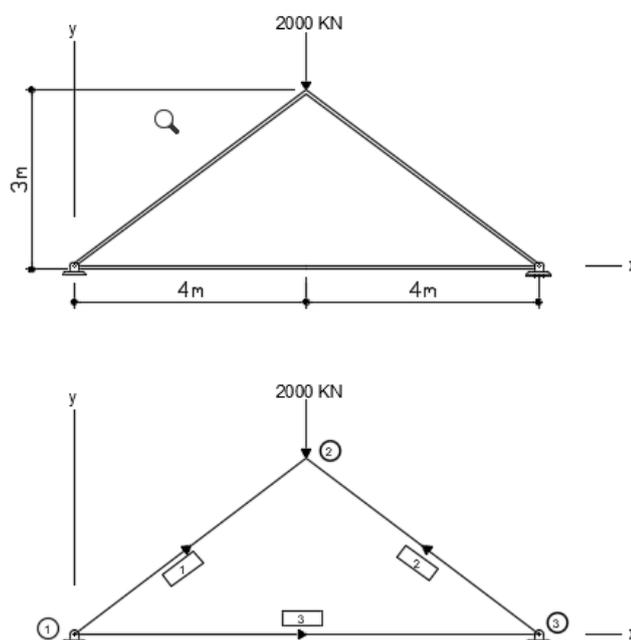


Figura 18 – Treliça de três barras.

Fonte: Autor.

A análise foi realizada através da equação (23) levando-se em conta a não linearidade geométrica, foi utilizado apenas um passo de carga e uma tolerância de convergência de $\varepsilon = 0,001$. O exemplo foi analisado por Kassimali (2012) com as equações de equilíbrio e matriz de rigidez tangente obtidas analiticamente. Para comparação de resultados, foi elaborada a tabela a seguir contendo os deslocamentos finais em cada grau de liberdade livre.

Tabela 1 – Deslocamentos Horizontais Exemplo 1.

Nó	Deslocamento Horizontal(cm)		Diferença %
	Kassimali	AD	
1	0,15664	0,15664	0,00%
2	-0,64975	-0,64975	0,00%
3	0,31327	0,31327	0,00%

Fonte: Autor.

Pode se perceber que a utilização do pacote de AD não provocou nenhum erro ao processo, convergindo assim para os resultados esperados. Além dos deslocamentos horizontais, compara-se também a matriz de rigidez global analítica do próprio Kassimali e a obtida através do DAALGPY para o elemento 1 em sua primeira iteração e para a barra 3 em sua última iteração:

```

BARRA
1
MATRIZ DA DERIVAÇÃO
[[ 5780.992  4335.744 -5780.992 -4335.744]
 [ 4335.744  3251.808 -4335.744 -3251.808]
 [-5780.992 -4335.744  5780.992  4335.744]
 [-4335.744 -3251.808  4335.744  3251.808]]
MATRIZ ANALÍTICA KASSIMALI
[[ 5780.992  4335.744 -5780.992 -4335.744]
 [ 4335.744  3251.808 -4335.744 -3251.808]
 [-5780.992 -4335.744  5780.992  4335.744]
 [-4335.744 -3251.808  4335.744  3251.808]]
SUBTRAÇÃO
[[ 9.09494702e-13  1.81898940e-12 -9.09494702e-13 -1.81898940e-12]
 [ 1.81898940e-12  1.36424205e-12 -1.81898940e-12 -1.36424205e-12]
 [-9.09494702e-13 -1.81898940e-12  9.09494702e-13  1.81898940e-12]
 [-1.81898940e-12 -1.36424205e-12  1.81898940e-12  1.36424205e-12]]

```

Figura 19 – Comparação das matrizes de rigidez lineares barra 1.

Fonte: Autor.

```

BARRA
3
MATRIZ DA DERIVAÇÃO
[[ 5645.5      0.      -5645.5      0.      ]
 [ 0.      212.61407168  0.      -212.61407168]
 [-5645.5     0.      5645.5     0.      ]
 [ 0.      -212.61407168  0.      212.61407168]]
MATRIZ ANALÍTICA KASSIMALI
[[ 5645.5      0.      -5645.5      0.      ]
 [ 0.      212.61407168  0.      -212.61407168]
 [-5645.5     0.      5645.5     0.      ]
 [ 0.      -212.61407168  0.      212.61407168]]
SUBTRAÇÃO
[[ -9.09494702e-13  0.00000000e+00  9.09494702e-13  0.00000000e+00]
 [ 0.00000000e+00 -2.84217094e-14  0.00000000e+00  2.84217094e-14]
 [ 9.09494702e-13  0.00000000e+00 -9.09494702e-13  0.00000000e+00]
 [ 0.00000000e+00  2.84217094e-14  0.00000000e+00 -2.84217094e-14]]

```

Figura 20 – Comparação das matrizes de rigidez não lineares barra 3.

Fonte: Autor.

Percebe-se que para o exemplo proposto só ocorreu divergência entre as matrizes de rigidez no 12º algarismo decimal, do ponto de vista computacional pode ser adotado como resultado satisfatório, podendo ser atribuído as condições de arredondamento do sistema.

6.2 TRELIÇA PLANA DE 4 BARRAS – ANÁLISE NÃO LINEAR E LINEAR

Este exemplo proposto por Segnini (2000) consiste em uma treliça plana formada por cinco barras, todas constituídas por um material elástico linear com módulo de Young $E = 20 \text{ GPa}$ e área de seção transversal $A = 0,001 \text{ m}^2$. A estrutura é sujeita a uma carga concentrada vertical para baixo $P_l = 180 \text{ kN}$ no nó 1. A Figura abaixo apresenta um desenho esquemático da treliça.

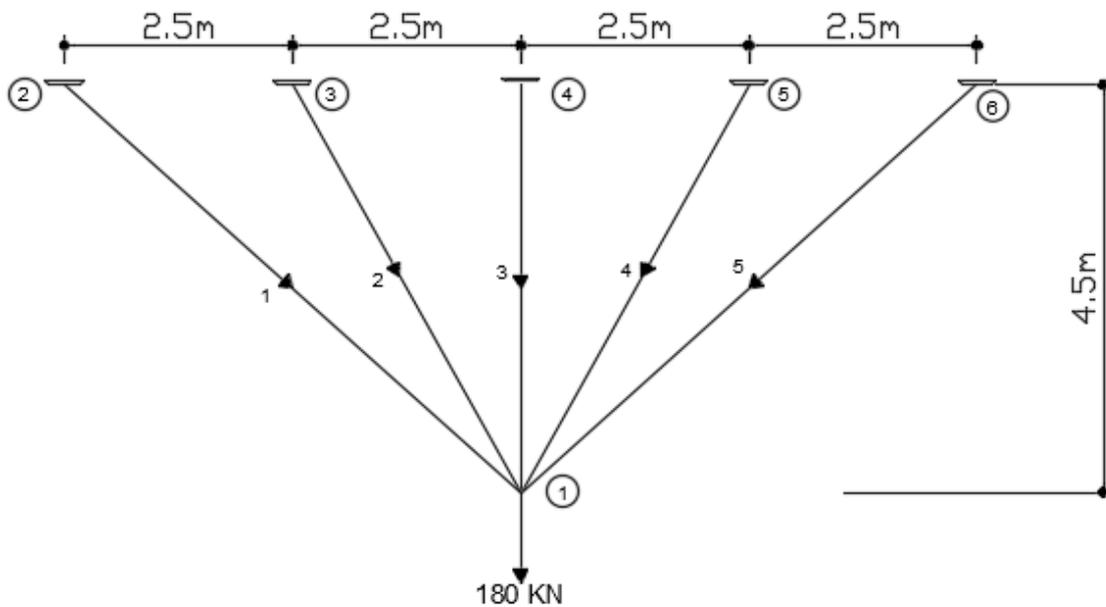


Figura 21 – Treliça de quatro barras.
Fonte: Autor.

A análise foi realizada através da equação (23) levando-se em conta a não linearidade geométrica, foram utilizados dez passos de carga com uma tolerância de $\varepsilon = 0,001$. Os resultados foram comparados com os obtidos por Segnini (2000) para uma análise não linear. A tabela a seguir contém os deslocamentos finais em cada grau de liberdade livre.

Tabela 2 - Deslocamentos Verticais Exemplo 2.

P/Pl	Deslocamento Vertical nó 1 (cm)		%
	Segnini	AD	
0,1	-0,001380	-0,001380	0,00%
0,2	-0,002760	-0,002760	0,00%
0,3	-0,004139	-0,004139	0,00%
0,4	-0,005518	-0,005518	0,00%
0,5	-0,006897	-0,006897	0,00%
0,6	-0,008275	-0,008275	0,00%
0,7	-0,009653	-0,009653	0,00%
0,8	-0,011031	-0,011031	0,00%
0,9	-0,012409	-0,012409	0,00%
1,0	-0,013786	-0,013786	0,00%

Fonte: Autor.

Como pode ser visto houve convergência para todos os valores. Além dos deslocamentos horizontais, compara-se também a matriz de rigidez global analítica obtida por Kassimali e a obtida através do DAALGPY para a barra 2. A análise foi não linear e estava em sua última iteração submetida a carga de 180 KN:

```

BARRA
2
MATRIZ DA DERIVAÇÃO
[[ 918.97714384 -1642.83710524 -918.97714384 1642.83710524]
 [-1642.83710524 2975.24377547 1642.83710524 -2975.24377547]
 [-918.97714384 1642.83710524 918.97714384 -1642.83710524]
 [ 1642.83710524 -2975.24377547 -1642.83710524 2975.24377547]]
MATRIZ ANALÍTICA KASSIMALI
[[ 918.97714384 -1642.83710524 -918.97714384 1642.83710524]
 [-1642.83710524 2975.24377547 1642.83710524 -2975.24377547]
 [-918.97714384 1642.83710524 918.97714384 -1642.83710524]
 [ 1642.83710524 -2975.24377547 -1642.83710524 2975.24377547]]
SUBTRAÇÃO
[[ 2.27373675e-13 0.00000000e+00 -2.27373675e-13 0.00000000e+00]
 [-2.27373675e-13 0.00000000e+00 2.27373675e-13 0.00000000e+00]
 [-2.27373675e-13 0.00000000e+00 2.27373675e-13 0.00000000e+00]
 [ 2.27373675e-13 0.00000000e+00 -2.27373675e-13 0.00000000e+00]]

```

Figura 22 – Comparação das matrizes de rigidez não lineares barra 2.

Fonte: Autor.

Nesse exemplo a divergência entre as matrizes de rigidez foi ainda menor que no anterior, ocorrendo apenas no 13º algarismo decimal, e em alguns casos não sendo perceptível dentro do sistema de aritmética de ponto flutuante do Python.

Para verificação da validade do cálculo do vetor de Forças Internas através do pacote de AD, realiza-se uma comparação entre as forças axiais calculadas por Sales (2018) para uma

análise linear da estrutura, submetida a carga de -180 KN . Para isso adotou-se a consideração (32).

Tabela 3 – Comparação forças axiais.

Barra	Força Axial (kN)		Diferença %
	Sales	AD	
1	27,4436	27,4481	0,02%
2	46,8386	46,8690	0,06%
3	61,2728	61,3347	0,10%
4	46,8386	46,8690	0,06%
5	27,4436	27,4481	0,02%

Fonte: Autor.

Assim como nos casos de deslocamentos a determinação dos esforços axiais resultou em valores condizentes com os encontrados com a literatura mostrando assim a validade da implementação.

6.3 TRELIÇA DE UMA BARRA – ANÁLISE NÃO LINEAR

Este exemplo proposto por Segnini (2000) consiste em uma treliça plana formada por uma barra, constituída por material elástico linear com módulo de Young $E = 20500\text{ KN/cm}^2$ e área de seção transversal $A = 6,526\text{ cm}^2$. A estrutura é sujeita a uma carga concentrada vertical para baixo $Pl = 10\text{ kN}$ no nó 2. A Figura abaixo apresenta um desenho esquemático da treliça.

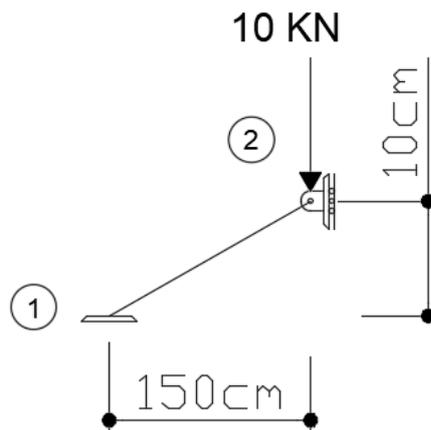


Figura 23 – Treliça de uma barra.

Fonte: Autor.

O equilíbrio foi imposto como a minimização da energia de deformação, dada pela equação (23) levando-se em conta a não linearidade geométrica. Foram utilizados dez passos

de carga com uma tolerância de $\varepsilon = 0,001$. Os resultados foram comparados com os obtidos por Segnini (2000) para uma análise não linear. A tabela a seguir contém os deslocamentos finais para a direção vertical no nó 2.

Tabela 4 – Comparação resultados.

Carga (KN)	Deslocamento Vertical nó 2 (cm)				
	AD	Segnini	Diferença %	Sales	Diferença %
1	-0,264	-0,264	0,00%	-0,264	0,00%
2	-0,553	-0,553	0,00%	-0,553	0,00%
3	-0,872	-0,872	0,00%	-0,872	0,00%
4	-1,234	-1,234	0,00%	-1,234	0,00%
5	-1,658	-1,658	0,00%	-1,658	0,00%
6	-2,187	-2,187	0,00%	-2,187	0,00%
7	-2,957	-2,957	0,00%	-2,957	0,00%
8	-21,619	-2,902	644,97%	-21,619	0,00%
9	-21,783	-21,783	0,00%	-21,783	0,00%
10	-21,941	-21,941	0,00%	-21,941	0,00%

Fonte: Autor.

Neste exemplo, observou-se uma grande divergência entre os valores para carga de 8 KN. O valor de $-21,619\text{ cm}$ foi encontrado na presente aplicação com AD e em Sales (2018) com a matriz de rigidez obtida analiticamente, ambos com um grande número de iterações. O valor de $-2,902\text{ cm}$ obtido por Segnini pode ser reproduzido ignorando-se a convergência e limitando o processo de Newton-Rahpson a 25 iterações (conforme descrito pela autora). Logo, pode se admitir que o valor correto é próximo de $-21,619\text{ cm}$ e a presente aplicação é satisfatória para treliças geometricamente lineares ou não lineares.

6.4 COLUNA SUBMETIDA A FLEXÃO COMPOSTA NORMAL

Este exemplo proposto por Zermiani (1998) consiste em uma coluna constituída por material elástico linear com módulo de Young $E = 943\text{ KN/cm}^2$, área de seção transversal $A = 225,00\text{ cm}^2$ e inércia $I = 4218,75\text{ cm}^4$. A estrutura é sujeita no seu topo a uma carga concentrada vertical para baixo $Pl = 39,24\text{ kN}$ e um momento de $Ml = -220,43\text{ kNcm}$. Foram utilizadas três malhas: 2 nós, 3 nós e 11 nós. A Figura abaixo apresenta um desenho esquemático da coluna:

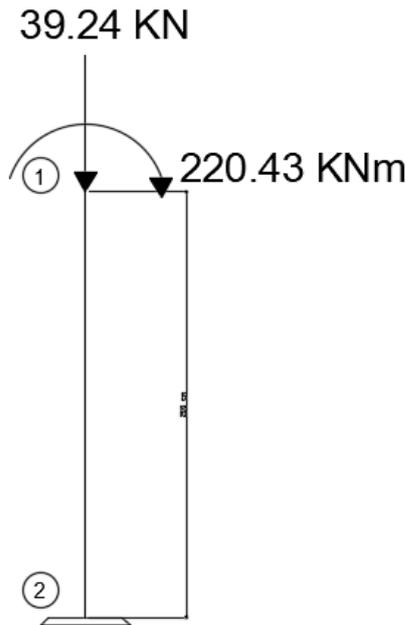


Figura 24 – Ilustração da estrutura.
Fonte: Zermiani, 1998.

A análise foi realizada através da equação (55 e 56) levando-se em conta a não linearidade geométrica e (64) para uma análise linear, foram utilizados dez passos de carga com uma tolerância de $\varepsilon = 0,001$. Os resultados foram comparados com os obtidos pelo próprio Zermiani (1998), para deslocamento no topo do pilar e o momento no engaste.

Tabela 5 – Comparação de resultados análise não linear.

Malha	Deslocamento Horizontal (cm)		Diferença %	Momento no Engaste (kNcm)		Diferença %
	Zermiani	AD		Zermiani	AD	
2	2,323334	2,323334	0,00%	311,5974	311,5976	0,00%
3	2,324826	2,324824	0,00%	311,6566	311,6561	0,00%
11	2,324733	2,324926	0,01%	311,6391	311,6601	0,01%

Fonte: Autor.

Como pode ser visto, para a análise não linear houve uma pequena divergência nos valores do momento reativo no engaste e no deslocamento horizontal no topo do pilar, como essa diferença foi em geral inferior a 0,02% podemos considerar como corretas ambas as análises. Assim como no caso das treliças, compara-se também a matriz de rigidez local analítica obtida por Zermiani e a gerada pela aplicação da derivação automática, para a barra 10 em sua última iteração.

```

BARRA
10
MATRIZ DA DERIVAÇÃO
[[ 8487.      0.      0.     -8487.      0.
   0.      ]
 [  0.      3053.43648  38187.576      0.     -3053.43648
 38187.576 ]
 [  0.      38187.576  636394.2      0.     -38187.576
318295.2 ]
 [ -8487.      0.      0.      8487.      0.
   0.      ]
 [  0.     -3053.43648 -38187.576      0.      3053.43648
-38187.576 ]
 [  0.      38187.576  318295.2      0.     -38187.576
636394.2 ]]]
MATRIZ ANALÍTICA
[[ 8487.      0.      0.     -8487.      0.
   0.      ]
 [  0.      3053.43648  38187.576      0.     -3053.43648
 38187.576 ]
 [  0.      38187.576  636394.2      0.     -38187.576
318295.2 ]
 [ -8487.      0.      0.      8487.      0.
   0.      ]
 [  0.     -3053.43648 -38187.576      0.      3053.43648
-38187.576 ]
 [  0.      38187.576  318295.2      0.     -38187.576
636394.2 ]]]
SUBTRAÇÃO
[[ 0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00
 0.00000000e+00 0.00000000e+00]
 [ 0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00
 0.00000000e+00 0.00000000e+00]
 [ 0.00000000e+00 0.00000000e+00 -2.32830644e-10 0.00000000e+00
 0.00000000e+00 -5.82076609e-11]
 [ 0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00
 0.00000000e+00 0.00000000e+00]
 [ 0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00
 0.00000000e+00 0.00000000e+00]
 [ 0.00000000e+00 0.00000000e+00 -5.82076609e-11 0.00000000e+00
 0.00000000e+00 -2.32830644e-10]]

```

Figura 25 – Comparação das matrizes de rigidez lineares para a barra 10.

Fonte: Autor.

Nesse exemplo a divergência entre a matriz de rigidez ocorreu no 13º algarismo decimal, e em alguns casos não foi perceptível dentro do sistema de aritmética de ponto flutuante do Python.

Para o caso da análise linear obtivemos:

Tabela 6 – Comparação de resultados análise linear.

Malha	Deslocamento Horizontal (cm)		Diferença %	Momento no Engaste (kNcm)		Diferença %
	Zermiani	AD		Zermiani	AD	
2	1,73151	1,73151	0,00%	220,4299	220,4300	0,00%
3	1,73151	1,73151	0,00%	220,4301	220,4300	0,00%
11	1,73143	1,73151	0,00%	220,4298	220,4300	0,00%

Fonte: Autor.

Para a análise linear também obteve-se uma pequena divergência para todos os valores do momento reativo no engaste, há de se notar que os valores propostos por Zermiani apresentam leve perturbação se comparados com os resultados feitos através de uma análise linear manual, talvez esse fenômeno também possa ter causado essa distorção para o caso da análise não linear. Do ponto de vista da AD os resultados estão plenamente condizentes com os oriundos de uma análise analítica, e com os obtidos com o autor em questão. A matriz linear é demonstrada a seguir:

```

BARRA
10
MATRIZ DA DERIVAÇÃO
[[ 8487.      0.      0.    -8487.      0.      0. ]
 [ 0.    3055.32 38191.5    0.    -3055.32 38191.5 ]
 [ 0.    38191.5 636525.    0.    -38191.5 318262.5 ]
 [-8487.      0.      0.     8487.      0.      0. ]
 [ 0.    -3055.32 -38191.5    0.     3055.32 -38191.5 ]
 [ 0.    38191.5 318262.5    0.    -38191.5 636525. ]]]
MATRIZ ANALÍTICA
[[ 8487.      0.      0.    -8487.      0.      0. ]
 [ 0.    3055.32 38191.5    0.    -3055.32 38191.5 ]
 [ 0.    38191.5 636525.    0.    -38191.5 318262.5 ]
 [-8487.      0.      0.     8487.      0.      0. ]
 [ 0.    -3055.32 -38191.5    0.     3055.32 -38191.5 ]
 [ 0.    38191.5 318262.5    0.    -38191.5 636525. ]]]
SUBTRAÇÃO
[[ 0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00
 0.00000000e+00 0.00000000e+00]
 [ 0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00
 0.00000000e+00 -7.27595761e-12]
 [ 0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00
 0.00000000e+00 0.00000000e+00]
 [ 0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00
 0.00000000e+00 0.00000000e+00]
 [ 0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00
 0.00000000e+00 7.27595761e-12]
 [ 0.00000000e+00 -7.27595761e-12 0.00000000e+00 0.00000000e+00
 7.27595761e-12 0.00000000e+00]]

```

Figura 26 – Comparação das matrizes de rigidez não lineares barra 10.

Fonte: Autor.

Assim como no caso não linear a diferença entre a matriz de rigidez obtida foi perceptível no 12º algarismo decimal.

6.5 PÓRTICO NÃO CONTRAVENTADO DE UM ANDAR E UM VÃO

Este exemplo proposto por Martha *et al* (2016) consiste em um pórtico constituído por material elástico linear, com módulo de Young $E = 205000 \text{ MPa}$, tanto os pilares quanto as vigas são compostos por perfis soldados, sendo estes CS 400x137 e CVS 500x217, respectivamente. A estrutura é sujeita no seu topo a uma carga concentrada vertical para baixo

$P_l = 18270 \text{ kN}$ e uma força horizontal de $F_l = 1820 \text{ kN}$. A Figura abaixo apresenta um desenho esquemático da coluna:



Figura 27 – Pórtico do exemplo.
Fonte: Autor.

A análise foi realizada através da equação (55 e 56) levando-se em conta a não linearidade geométrica, foram utilizados 15 passos de carga com uma tolerância de $\varepsilon = 0,001$. Os resultados foram comparados com os disponibilizados em Martha *et al* (2016), que apresenta os valores através de diferentes modelagens quanto a análise linear geométrica, a primeira feita por Silva (2014 *apud* Martha *et al*, 2016) desenvolvida em um referencial corrotacional, e a outra encontrados no software STAAD por meio do método P-delta.

Tabela 7 – Comparação dos deslocamentos horizontais.

P/Pl	Deslocamento Horizontal (m)				
	AD	Silva	%	STAAD	%
0,03	0,00421	0,00423	0,47%	0,00423	0,47%
0,07	0,00858	0,00862	0,46%	0,00862	0,46%
0,10	0,01313	0,01317	0,30%	0,01318	0,38%
0,13	0,01786	0,01791	0,28%	0,01793	0,39%
0,17	0,02278	0,02284	0,26%	0,02287	0,39%
0,20	0,02790	0,02797	0,25%	0,02801	0,39%
0,23	0,03325	0,03331	0,18%	0,03338	0,39%
0,27	0,03883	0,03888	0,13%	0,03898	0,38%
0,30	0,04466	0,0447	0,09%	0,04483	0,38%
0,33	0,05075	0,05078	0,06%	0,05095	0,39%
0,50	0,08594	0,08578	0,19%	0,08628	0,39%
0,67	0,13158	0,13097	0,47%	0,1321	0,39%
1,00	0,28079	0,27702	1,36%	0,28188	0,39%

Fonte: Autor.

Ao se comparar os resultados da AD com a análise P- Δ do software STAAD e a análise em 2ª ordem de Silva (2004 *apud* Martha *et al*, 2016), notamos que os valores estão satisfatórios e convergindo para resultados semelhantes aos desses softwares, notando que em média a divergência para os valores de deslocamento horizontal ficou em 0,401% para o STAAD e 0,346% comparado com a análise de Silva. Nota-se que, conforme se eleva o valor do carregamento aumenta-se a diferença entre os resultados da análise de Silva e da AD, já no caso do software STAAD mantém-se praticamente constante durante toda a análise uma diferença de 0,39%. O gráfico abaixo traz um comparativo dos deslocamentos pelos passos de carga.

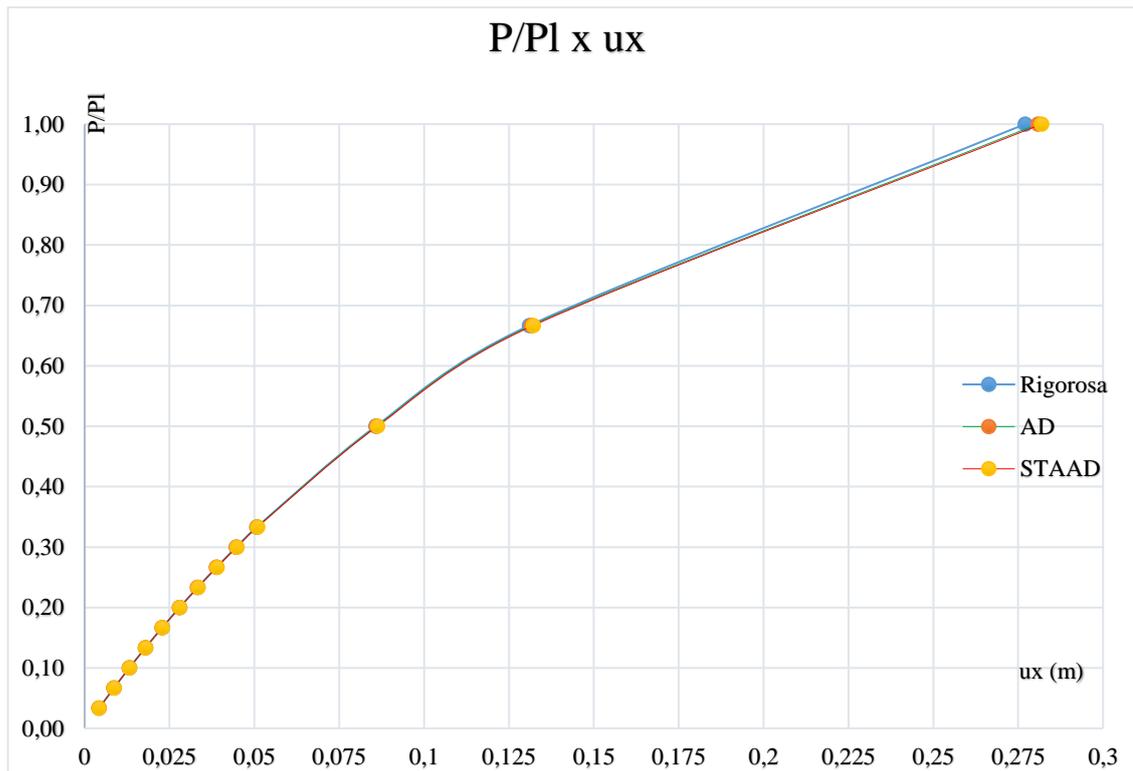


Figura 28 – Gráfico de Cargas x Deslocamento.

Fonte: Autor.

Também se comparou os valores de esforço cortante atuantes no pilar esquerdo resultando em:

Tabela 8 – Comparativo de resultados pilar esquerdo.

P/Pl	Força Cortante (KN)				
	AD	Silva	Diferença %	STAAD	Diferença %
0,03	30,48	30,72	0,78%	30,60	0,39%
0,07	60,99	61,69	1,13%	61,22	0,38%
0,10	91,52	92,94	1,53%	91,87	0,38%
0,13	122,07	124,88	2,25%	122,54	0,38%
0,17	152,65	156,31	2,34%	153,24	0,39%
0,20	183,27	188,48	2,76%	183,98	0,39%
0,23	213,92	221,00	3,20%	214,75	0,39%
0,27	244,63	254,00	3,69%	245,57	0,38%
0,30	275,38	287,00	4,05%	276,45	0,39%
0,33	306,20	321,00	4,61%	307,38	0,38%
0,50	461,48	497,00	7,15%	463,28	0,39%
0,67	620,16	690,00	10,12%	622,59	0,39%
1,00	965,98	1174,00	17,72%	969,92	0,41%

Fonte: Autor.

A análise do esforço cortante corrobora com os resultados encontrados anteriormente, e explicita ainda mais como os valores da análise de Silva distanciam-se dos resultados gerados pelas duas outras análises, o que já era esperado visto que a teoria em segunda ordem realizada por Silva apresenta-se menos simplificada do que nos outros dois casos. Já em relação aos valores da STAAD, o esforço cortante assim como para os deslocamentos horizontais manteve-se em torno dos 0,39% todo o tempo.

Compara-se ainda a matriz de rigidez analítica deduzida por Zermiani (1998) com os resultados através da AD, escolheu-se a barra número 3 (pilar direito), em sua última iteração para a carga Pl:

```

BARRA
3
MATRIZ DA DERIVAÇÃO
[[ 713400.          0.          0.         -713400.
   0.          0.          ]
 [ 0.          5716.35190956  23865.81329565  0.
 -5716.35190956  23865.81329565]
 [ 0.          23865.81329565  73169.42197101  0.
 -23865.81329565  46159.64450725]
 [-713400.          0.          0.          713400.
   0.          0.          ]
 [ 0.          -5716.35190956 -23865.81329565  0.
  5716.35190956 -23865.81329565]
 [ 0.          23865.81329565  46159.64450725  0.
 -23865.81329565  73169.42197101]]
MATRIZ ANALÍTICA
[[ 713400.          0.          0.         -713400.
   0.          0.          ]
 [ 0.          5716.35190956  23865.81329565  0.
 -5716.35190956  23865.81329565]
 [ 0.          23865.81329565  73169.42197101  0.
 -23865.81329565  46159.64450725]
 [-713400.          0.          0.          713400.
   0.          0.          ]
 [ 0.          -5716.35190956 -23865.81329565  0.
  5716.35190956 -23865.81329565]
 [ 0.          23865.81329565  46159.64450725  0.
 -23865.81329565  73169.42197101]]
SUBTRAÇÃO
[[-1.16415322e-10  0.00000000e+00  0.00000000e+00  1.16415322e-10
  0.00000000e+00  0.00000000e+00]
 [ 0.00000000e+00 -9.09494702e-13 -7.27595761e-12  0.00000000e+00
  9.09494702e-13 -1.09139364e-11]
 [ 0.00000000e+00 -7.27595761e-12 -1.45519152e-11  0.00000000e+00
  7.27595761e-12  0.00000000e+00]
 [ 1.16415322e-10  0.00000000e+00  0.00000000e+00 -1.16415322e-10
  0.00000000e+00  0.00000000e+00]
 [ 0.00000000e+00  9.09494702e-13  7.27595761e-12  0.00000000e+00
 -9.09494702e-13  1.09139364e-11]
 [ 0.00000000e+00 -7.27595761e-12  0.00000000e+00  0.00000000e+00
  7.27595761e-12 -1.45519152e-11]]

```

Figura 29 – Comparação das matrizes de rigidez não lineares barra 10.

Fonte: Autor.

Assim como nos casos anteriores a matriz de rigidez não linear registrou uma diferença no 10º algarismo decimal.

6.6 PÓRTICO DE EDIFÍCIO

Neste exemplo proposto por Banki (1999) e estudado por Junges, La Rovere e Loriggio (2012), trata-se de um pórtico de contraventamento, engastado na base submetido a esforços verticais e horizontais. As características geométricas do problema são: colunas e vigas compostas por material elástico linear com módulo de Young $E = 27000 \text{ MPa}$, seção transversal das vigas composta por 13x55cm e para os pilares 30x20cm. Foi utilizado uma

malha com 12 barras de portico para análise. A Figura abaixo apresenta um desenho esquemático da estrutura com o carregamento proposto:

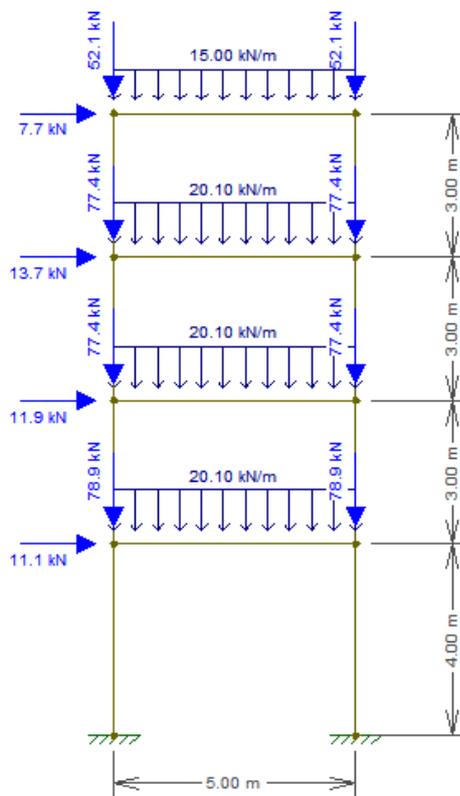


Figura 30 – Ilustração da Estrutura
Fonte: Autor.

A análise foi realizada levando-se em conta a não linearidade geométrica e uma análise linear, foram utilizados dez passos de carga com uma tolerância de $\varepsilon = 0,001$. Os resultados foram comparados com os obtidos pelos autores citados, para deslocamento horizontal no topo do pórtico obteve-se:

Tabela 9 – Comparativo de resultados

Deslocamento Horizontal (cm)				
AD	Banki	Diferença %	Junges et. al (SAP 2000)	Diferença %
4,868	4,866	0,04%	4,872	0,08%

Fonte: Autor.

A fim de análise submeteu-se a estrutura em questão até próximo de sua carga limite obtendo assim a seguinte curva de deslocamentos:

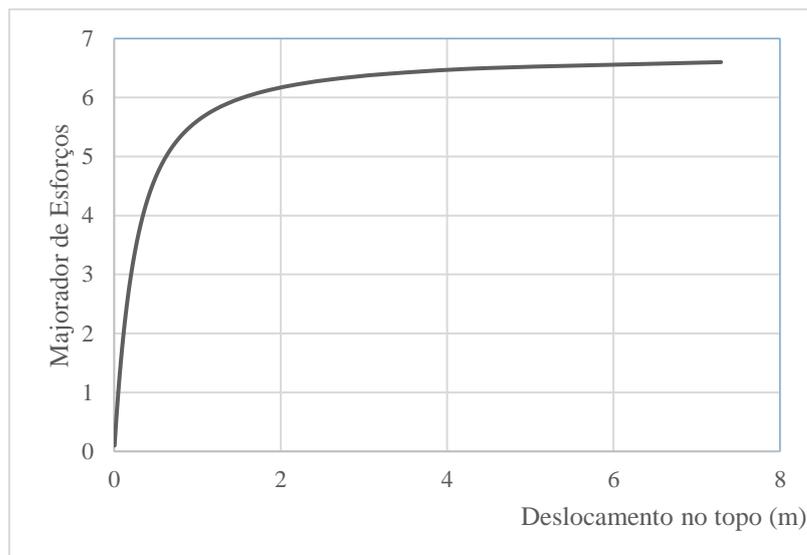


Figura 31 – Gráfico de Deslocamento x Passo de Carga

Fonte: Autor.

Como pode ser observado a curva acima demonstra que os deslocamentos nesse portico são regidos por um critério não linear, tendo sua carga de ruptura sendo alcançada próximo a 6 vezes o valor do carregamento da figura 30.

7 CONCLUSÃO

O objetivo principal deste trabalho foi o desenvolvimento de um código computacional capaz de possibilitar a derivação automática de equações algébricas. Com especial interesse a sua utilização na análise de sistemas estruturais lineares e não lineares.

As soluções obtidas na seção 6, mostram que os resultados obtidos com a utilização do pacote de diferenciação automática DAALGPY para a análise de estruturas, foram bastante satisfatórios se comparados aos obtidos com resultados numéricos disponíveis na literatura. Os dois únicos pontos de grande discrepância encontrados foram nas seções **6.3** (relacionado ao número máximo de iterações utilizada por Segnini) e **6.6** (se compararmos aos valores do esforço normal obtidos pela análise completa de Silva).

Constatou-se que os valores encontrados para a matriz de rigidez local em pórticos e global em treliças planas, apresentou pequena diferença a partir do 10º algarismo decimal, no pior dos casos, o que se apresentou como um erro relativo de grandeza próxima a 10^{-16} . Como já explicado anteriormente, um dos motivos para ocorrer essa pequena discrepância pode ser atribuído aos sistemas de aritmética de ponto flutuante, tendo em vista que a matriz de rigidez da diferenciação automática, possui muito mais operações utilizando o sistema de aritmética de ponto flutuante do que os resultados analíticos, desse modo torna-se mais suscetível a erros de arredondamento, que estão diretamente ligados a limitações computacionais (tamanho da mantissa, tempo de execução e etc.).

Também foi possível perceber como os esforços axiais interagem no comportamento não linear geométrico das estruturas. Pode se perceber principalmente nos exemplos **6.4** onde ocorreu uma comparação entre os resultados lineares e não lineares, como os valores de momento fletor e deslocamento destoaram cerca de 35%, entre as análises.

Um ponto interessante da utilização do algoritmo de AD, foi a possibilidade de mudar da análise não linear para a análise linear, dentro do algoritmo modificando-se apenas a indicação de qual fórmula da energia de deformação estava sendo utilizada. Isso possibilitou realizar diferentes tipos de análises dentro do mesmo código, cada uma tomando diferentes considerações para a cinemática da estrutura, gerando assim programas simples, mas muito abrangentes.

Indica-se para futuros estudos:

- A utilização de formulações mais completas para a não linearidade geométrica de pórticos levando em conta os efeitos de grandes deformações;
- A consideração da não linearidade física para treliças e pórticos;
- Problemas de otimização topológica;
- Formulação de análise não linear geométrica para estruturas tridimensionais;
- Adaptar a etapa de derivação do grafo computacional, a computação paralela.
- Acoplar um algoritmo de simplificação de expressões, como o desenvolvido por Korelc no AceGen.

8 DIVULGAÇÃO

Os principais conceitos e resultados descritos nesse trabalho foram apresentados no XL Ibero-Latin-American Congress on Computational Methods in Engineering (CILAMCE 2019) em Natal, RN. O artigo completo foi publicado em seus anais como Santos e Costa (2019).

9 REFERÊNCIAS

- ANDO, J. K. **Análise da Estabilidade de Estruturas Metálicas com Comportamento Não-Linear**. Tese (Doutorado em Engenharia Civil) – Programa de Pós-Graduação em Engenharia Civil, Universidade Federal Fluminense, Niterói, 2008.
- BANKI, A. L. **Estudo Sobre a Inclusão da Não Linearidade Geométrica em Projetos de Edifícios**. Dissertação (Mestrado em Engenharia Civil) – Programa de Pós-Graduação em Engenharia Civil, Universidade Federal de Santa Catarina, Florianópolis, 1999.
- BARTHELEMY, J. F. M; HALL, L. E. **Automatic Differentiation as a Tool in Engineering Design**. NASA Technical Memorandum, 107661, Agosto, 1992.
- BIRGIN, E. J. G. **Diferenciação Computacional e Aplicações**. Tese (Doutorado em Matemática Aplicada) – Instituto de Matemática, Estatística e Computação Científica, Universidade Estadual de Campinas, Campinas, 1998.
- BUENO, J. R; LORIGGIO, D. D. **Análise de Efeitos de 2ª Ordem: Estudo de Caso**. RIEM - Revista IBRACON de Estruturas e Materiais vol.9 n.4: 494 – 509, Agosto, 2016.
- CARVALHO, M. F. M. S. de. **Formulação Corrotacional para Análise de Vigas com Elementos Finitos**. Dissertação (Mestrado Integrado em Engenharia Mecânica) – Departamento de Engenharia Mecânica e Industrial, Universidade Nova de Lisboa, Lisboa, 2010.
- CHANDRUPATIA, T; BELEGUENDU, A. **Elementos Finitos**. 4ª ed. São Paulo: Editora Pearson, 2014.
- COSTA JUNIOR, E. F. da. **Resolução Automática de Equações Algébrico-diferenciais de Índice Superior**. Tese (Doutorado em Engenharia Química) – COPPE, Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2003.
- DE SOUZA, A. S. C; PEREIRA, M. F; NARDIN, S. D. **Análise Inelástica de Segunda Ordem em Pórticos de Aço**. REEC – Revista Eletrônica de Engenharia Civil, vol.12, n.13: 17-30, Jun./Dez., 2016.
- ESPATH, L. F. R; LINN, R. V; AWRUCH, A. M. **Shape Optimization of Shell Structures Based on NURBS Description Using Automatic Differentiation**. International Journal for Numerical Methods in Engineering, n.88: 613-636, Fevereiro,2011.
- GRIEWANK, A. **On Automatic Differentiation**. Mathematical Programming: Recent Development and Applications, 83-108, Novembro, 1988.
- GRIEWANK, A; WALTHER, A. **Introduction to Automatic Differentiation**. PAMM – Proceedings in Applied Mathematics & Mechanics, n.2: 45-49, 2003.
- GRIEWANK, A. **Who Invented the Reverse Mode of Differentiation?** Documenta Mathematica, Extra Volumee: 389-400, 2012.

- HAASE, G; *et al.* **Optimal Sizing and Shape Optimization in Structural Mechanics.** Institute of Computational Mathematics, Johannes Kepler University of Linz, Linz, Maio, 2003.
- JUNGES, E.; LORIGGIO, D.; LA ROVERE, H. L. **Análise de Segunda Ordem Global de Estruturas de Concreto Armado Utilizando Programas computacionais de dimensionamento.** 54º Congresso Brasileiro do Concreto, Outubro, 2012.
- JUSTO, D. A. R; *et al.* **Calculo Numérico: Um Livro Colaborativo.** Disponível em: <https://www.ufrgs.br/reatmat/CalculoNumerico/livro-py/in-quadratura_de_gauss-legendre.html> .
- KASSIMALI, A. **Matrix Analysis of Structures.** 2ª ed. Stamford: Cengage Learning, 2012.
- KORELC, J. **Automatic Generation of Numerical Codes.** University of Ljubljana, Slovenia, 2003.
- LENGIEWICZ, J; KORELC, J; STUPKIEWICZ, S. **Automation of Finite Element Formulations for Large Deformation Contact Problems.** International Journal for Numerical Methods in Engineering, n.85: 1252-1279, Agosto, 2010.
- LIMA, J. M. F; GARCIA, L. F. T. **Princípio dos Trabalhos Virtuais Aplicado à Flexão de Barras com Forte Não Linearidade Geométrica.** *Sitientibus*, n.26: 131-149, Jan./Jun., 2002.
- NØRGAARD, S. A; *et. al.* **Applications of Automatic Differentiation in Topology Optimization.** Structural and Multidisciplinary Optimization, n.56: 1135-1146, 2017.
- Python Software Foundation. **Disassembler for Python bytecode.** Disponível em: <https://docs.python.org/3.4/library/dis.html>.
- RAMALHO, J. B; SILVA, R. S. e. **Aplicações de Métodos de Energia a Problemas de Instabilidade em Estruturas.** Departamento de Engenharia Civil – Pontifícia Universidade Católica do Rio de Janeiro, [20~].
- RIGITANO, A. C. **Contribuição para a Determinação de Matrizes de Rigidez e Vetores de Ações Nodais Equivalentes com o Emprego da Formulação Hermitiana Livre.** Tese (Doutorado em Engenharia de Estruturas) – Escola de Engenharia de São Carlos, Universidade de São Paulo, 1999.
- RUGGIERO, M. A. G.; LOPES, V. L. R. **Cálculo Numérico – Aspectos Teóricos e Computacionais.** 2ª ed. São Paulo: Pearson, 1996.
- SALES, D. C. **Análise Geometricamente Não Linear de Treliças Planas.** Trabalho de Conclusão de Curso – Departamento de Engenharia Civil, Universidade Federal de Sergipe, Sergipe, 2018.
- SANTOS, D. M; COSTA, J; C. **Geometrically nonlinear analysis of plane trusses and frames using an automatic derivation algorithm.** Em Proceedings of the Ibero-Latin-American Congress on Computational Methods in Engineering (CILAMCE 2019). ISSN 2675-6269. Disponível em <https://cilamce.com.br/anais/arearestrita/apresentacoes/104/7343.pdf>. Natal, RN, 2019.

SEGNINI, S. C. A. **Estudo Comparativo de Formulações para a Análise Não-Linear Geométrica de Treliças.** Dissertação (Mestrado em Engenharia Civil) – Faculdade de Engenharia Civil, Universidade Estadual de Campinas, Campinas, 2000.

SILVA, M. F. D. S; MENEZES, I. F. M; MARTHA, L. F. **Um Método Simplificado para Análise Não-Linear Geométrica no Ftool.** XXXVII CILAMCE – Iberian Latin American Congress on Computational Methods in Engineering, Novembro, 2016.

SOUSA, C. M. L. de. **Determinação de Deslocamentos em Treliças Planas Via Métodos de Energia.** Trabalho de Conclusão de Curso - Centro de Ciências Tecnologia e Saúde, Universidade Estadual da Paraíba, 2017.

ŠRAJER, F; KUKELOVA, Z; FITZGIBBON, A. **A Benchmark of Selected Algorithmic Differentiation Tools on Some Problems in Computer Vision and Machine Learning.** AD2016—7th International Conference on Algorithmic Differentiation, and in Optimization Methods and Software, Taylor and Francis, Fevereiro 2018.

TAKAHASHI, Y; KAWAHARA, M. **Identification of Reynolds Number Using Automatic Differentiation.** Inverse Problems, Design and Symposium, 2004.

ZERMIANI, F. L. **Contribuição à Análise Não-Linear Geométrica de Pórticos Planos.** Dissertação (Mestrado em Engenharia Mecânica) – Programa de Pós-Graduação em Engenharia Mecânica, Universidade Federal de Santa Catarina, Florianópolis, 1998.

10 ANEXOS

10.1 MATRIZ DE ROTAÇÃO (T)

$$T = \begin{bmatrix} \cos(\theta) & \sin(\theta) & 0 & 0 & 0 & 0 \\ -\sin(\theta) & \cos(\theta) & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \cos(\theta) & \sin(\theta) & 0 \\ 0 & 0 & 0 & -\sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

10.2 MATRIZ DE RIGIDEZ TANGENTE KASSIMALI

$$K_t = \frac{EA}{L} \begin{bmatrix} c_x^2 & c_x c_y & -c_x^2 & -c_x c_y \\ c_x c_y & c_y^2 & -c_x c_y & -c_y^2 \\ -c_x^2 & -c_x c_y & c_x^2 & c_x c_y \\ -c_x c_y & -c_y^2 & c_x c_y & c_y^2 \end{bmatrix} + \frac{Q}{L} \begin{bmatrix} -c_y^2 & c_x c_y & c_y^2 & -c_x c_y \\ c_x c_y & -c_x^2 & -c_x c_y & c_x^2 \\ c_y^2 & -c_x c_y & -c_y^2 & c_x c_y \\ -c_x c_y & c_x^2 & c_x c_y & -c_x^2 \end{bmatrix}$$

Onde:

$$C_x = \cos(\theta) \text{ e } C_y = \sin(\theta)$$

10.3 MATRIZ DE RIGIDEZ TANGENTE ZERMIANI

$$K = \frac{EI}{L} \cdot \begin{bmatrix} \frac{A}{I} & 0 & 0 & -\frac{A}{I} & 0 & 0 \\ 0 & \frac{12}{L^2} & \frac{6}{L} & 0 & -\frac{12}{L^2} & \frac{6}{L} \\ 0 & \frac{6}{L} & 4 & 0 & -\frac{6}{L} & 2 \\ \frac{A}{I} & 0 & 0 & -\frac{A}{I} & 0 & 0 \\ 0 & -\frac{12}{L^2} & -\frac{6}{L} & 0 & \frac{12}{L^2} & -\frac{6}{L} \\ 0 & \frac{6}{L} & 2 & 0 & -\frac{6}{L} & 4 \end{bmatrix} + \frac{P}{L} \cdot \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{6}{5} & \frac{L}{10} & 0 & -\frac{6}{5} & \frac{L}{10} \\ 0 & \frac{L}{10} & \frac{2L^2}{15} & 0 & -\frac{L}{10} & -\frac{L^2}{30} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -\frac{6}{5} & -\frac{L}{10} & 0 & \frac{6}{5} & -\frac{L}{10} \\ 0 & \frac{L}{10} & -\frac{L^2}{30} & 0 & -\frac{L}{10} & \frac{2L^2}{15} \end{bmatrix}$$