

UNIVERSIDADE ESTADUAL DO MARANHÃO
CENTRO DE CIÊNCIAS TECNOLÓGICAS
CURSO DE ENGENHARIA CIVIL

ANTÔNIO VITOR BEZERRA FERNANDES

**ANÁLISE COMPUTACIONAL NÃO LINEAR DE ESTRUTURAS RETICULADAS
PLANAS**

São Luís

2017

ANTÔNIO VITOR BEZERRA FERNANDES

**ANÁLISE COMPUTACIONAL NÃO LINEAR DE ESTRUTURAS RETICULADAS
PLANAS**

Trabalho de Conclusão de Curso apresentado
ao Curso de Engenharia Civil da Universidade
Estadual do Maranhão como requisito parcial
para a obtenção do título de Bacharel em
Engenharia Civil.

Orientador: Prof. Me. Carlos Cesar Pereira de
Almeida.

São Luís

2017

Fernandes, Antonio Vitor Bezerra.

Análise computacional não linear geométrica de estruturas reticuladas planas / Antonio Vitor Bezerra Fernandes. – São Luís, 2017.

72f.

Monografia (Graduação) – Curso de Engenharia Civil, Universidade Estadual do Maranhão, 2017.

Orientador: Prof. Me. Carlos Cesar Pereira de Almeida.

1.Estrutura. 2. Reticulada. 3. Plana. I. Título.

ANTÔNIO VITOR BEZERRA FERNANDES

**ANÁLISE COMPUTACIONAL NÃO LINEAR DE ESTRUTURAS RETICULADAS
PLANAS**

Trabalho de Conclusão de Curso apresentado
ao Curso de Engenharia Civil da Universidade
Estadual do Maranhão como requisito parcial
para a obtenção do título de Bacharel em
Engenharia Civil.

Aprovado em: __/12/2017

BANCA EXAMINADORA

Prof. Me. Carlos Cesar Pereira de Almeida

Prof. Orientador

Universidade Estadual do Maranhão

Prof. Silvia Ribeiro de Oliveira

Membro titular

Universidade Estadual do Maranhão

Prof. Fernando Jorge Cutrim Demétrio

Membro titular

Universidade Estadual do Maranhão

DEDICATÓRIA

Dedico este trabalho a toda minha família, amigos e professores, que me ajudaram e incentivaram nessa longa jornada, onde todos se demonstraram de extrema importância para que eu chegasse a esse ponto da minha vida.

AGRADECIMENTOS

Primeiramente agradeço à Deus, por iluminar minha jornada e permitir que eu cumpra esse sonho. Aos meus pais, Maurício e Concita, que sempre estiveram ao meu lado, me apoiando, movendo montanhas para meu bem e minha alegria, vocês com certeza são meu maior bem, porto seguro, e sei que posso contar com vocês para qualquer coisa, saibam que podem contar comigo sempre, pois sem vocês eu não seria eu, são meu passado, presente e futuro.

À minha irmã, pelo companheirismo e por ter contribuído com a construção deste trabalho.

Aos meus avós Marçalina, Marenice, Zuza “in memoriam” e Antônio “in memoriam”, que construíram minha família e o que sou, vocês são os pilares de tudo.

À tia Cibely e tia Suzi por todo o suporte, principalmente quando mudei para São Luís, a todos os meus tios que sempre se fizeram presentes e importantes. Ao meu padrinho Fernando e tia Simone pelo grande exemplo de pessoa. Ao Cabo Nélcio grande pensador, matemático e cantor do forró inglês mais autêntico.

À Maria Raimunda e Cilene, vocês me criaram, participaram da minha formação como pessoa, as várias risadas e broncas que sempre animavam os dias no Santo Antônio.

Aos meus primos, companheiros de infância, de bola e de grandes momentos marcantes.

Ao meu Orientador Professor Carlos César, por confiar sua pesquisa a mim e pacientemente me ajudar a concluir esta etapa.

Ao meu Co-orientador e grande amigo Ederson, por estar sempre disposto e disponível para me ajudar a qualquer momento, e principalmente, pela imensa contribuição de ideias e incentivo e ao desenvolvimento deste trabalho e durante toda a graduação.

Aos meus amigos do grupo “Gabriel se forma em 2020”, vocês foram um grande suporte nesta caminhada, o companheirismo tornou-a menos árdua e mais prazerosa. Em especial para Bruno e Juan por abrirem minha mente para a área de programação, justamente a ferramenta deste trabalho. Também a Guilherme e Gabriel por sempre compreenderem minha ausência na roda de conversa.

No need to hear things that they say
Life is for my own to live my own way
James Hetfield

RESUMO

Este trabalho tem como objetivo formular e apresentar um programa criado em ambiente Matlab com a função de analisar estruturas, em especial pórticos, considerando o efeito de não linearidade geométrica, para isso este trabalho também irá abordar o método dos elementos finitos aplicado ao elemento estrutural que sofre ação de viga e ação de barra de forma simultânea. Além de conter valiosas fundamentações teóricas da análise estrutural este trabalho irá expor com relevância um tratamento matemático progressivo no intuito de se obter as formulações que regem este particular tipo de análise estrutural. O código computacional será manifestado por meio de um exemplo prático onde estarão presentes comentários explicativos.

Palavras-chave: Estrutura, reticulada, plana.

ABSTRACT

This work aims to formulate and present a program created in Matlab environment with the function of analyzing structures, especially frames, considering the effect of geometric nonlinearity. For this, this work will also adopt the finite element method applied to the structural element that suffers bar action and beam action simultaneously, besides containing valuable theoretical foundations of the structural analysis this work will expose with relevance a progressive mathematical treatment in order to obtain the formulations that govern this particular type of structural analysis. The computational code will be manifested through a practical example where explanatory comments will be present.

Keywords: Structure.Reticulated.Flat.

LISTA DE FIGURAS

Figura 1: Deslocamento horizontal de um pórtico	20
Figura 2: Deslocamento horizontal de um elemento	20
Figura 3: Deformação à ação de flexão	27
Figura 4: Forças expressas na seção deformada	28
Figura 5: Ação das forças na fatia dx	29
Figura 6: Elemento de barra com seis graus de liberdade.	31
Figura 7: Elemento Arbitrario	36
Figura 8: Pórtico exemplo	40
Figura 9: Modelo computacional.....	41
Figura 10: Código Principal (1/4).....	42
Figura 11: Código Principal (2/4).....	43
Figura 12: Código Principal (3/4).....	44
Figura 13: Código Principal (4/4).....	45
Figura 14: Numeração, barras e nós	46
Figura 15: Matriz topológica ou Referencial.....	47
Figura 16: Início das interações.....	48
Figura 17: Loop para montagem da matriz de rigidez global.....	48
Figura 18: Código de desenvolvimento da matriz de rigidez do elemento.	49
Figura 19: Código de montagem da matriz de rigidez global.	50
Figura 20: Etapa seguinte no código principal.	52
Figura 21: Código de resolução do sistema de equações não lineares.	53
Figura 22: Matriz ed	54
Figura 23: Determinação dos esforços.	55
Figura 24: Valores de P	56
Figura 25: Fim das interações e geração de ilustração.	57
Figura 26: Reações e deslocamentos.	58

Figura 27: Esforços nas 3 barras.....	58
Figura 28: Ilustração do portico.....	59
Figura 29: Corpo da estrutura contendo suas dimensões, a numeração de barras e nós, em conjunto com os carregamentos atuantes no pórtico.	60
Figura 30: Propriedade das barras	61
Figura 31: Código Principal (1/5).....	62
Figura 32: Código Principal (2/5).....	63
Figura 33: Código Principal (3/5).....	64
Figura 34: Código Principal (4/5).....	65
Figura 35: Código Principal (5/5).....	66
Figura 36: Corpo da estrutura gerado pelo código.	67
Figura 37: Esforço Normal de primeira ordem.	67
Figura 38: Esforço Normal de segunda ordem.	68
Figura 39: Resultados obtidos por Leitão Giles.	69

SUMÁRIO

1. INTRODUÇÃO	15
1.1. Delimitação.....	15
1.2. Relevância e Justificativa	16
1.3. Objetivos.....	16
1.3.1. Objetivo geral.....	16
1.3.2. Objetivos específicos.....	16
1.4. Metodologia.....	17
1.5. Organização do Trabalho.....	18
2. ANÁLISE ESTRUTURAL NÃO LINEAR	19
2.1. Não Linearidade Geométrica.....	19
2.2. Método dos Elementos Finitos	21
2.3. Metodologia de Solução	22
2.4. Formulação Matemática	23
2.4.1. Equilíbrio da seção transversal deformada.....	27
2.4.2. Equações Diferenciais	30
2.4.2.1. Resolvendo a primeira equação diferencial.	31
2.4.2.2. Resolvendo a segunda equação diferencial.	33
2.5. Matriz Transformação para seis graus de liberdade.	35
3. IMPLEMENTAÇÃO COMPUTACIONAL	40
3.1. Visão geral do código computacional:	40
3.2. Entendendo as Etapas	46
3.2.1. Etapa inicial.....	46
3.2.2. Segunda etapa.....	47
3.2.3. Terceira etapa.	48

3.2.4.	Quarta Etapa.....	48
3.2.4.1.	Função matrizderigidezdaviga2Defeitossegundaordem.....	49
3.2.4.2.	Função Montagemrigidezglobal	50
3.2.5.	Quinta Etapa.....	52
3.2.5.1.	Função Resolvesistema.....	53
3.2.5.2.	Função extraideslocamento.....	54
3.2.6.	Sexta Etapa.....	55
3.2.6.1.	Função esforço	56
3.2.7.	Etapa Final.....	57
4.	VALIDAÇÃO DE RESULTADOS POR COMPARAÇÃO.	60
5	CONCLUSÃO.....	70
5.1	Sugestões para trabalhos futuros.....	70
	REFERÊNCIAS	71

1. INTRODUÇÃO

Foi observado enquanto realizado o curso de engenharia civil a atenção especial dada a área de análise das estruturas por parte de todo o corpo acadêmico, tanto por professores quanto por alunos. Tendo em vista todo o metodismo minucioso empregado nos procedimentos de resolução da análise estrutural, muitas vezes consideravelmente trabalhoso e/ou até mesmo confuso, fato esse constatado em ambos os métodos de resolução estudados, método das forças e método dos deslocamentos, fascinando e instigando a curiosidade dos alunos, além de deixar bem claro a importância da análise estrutural na idealização de um projeto, uma vez que os resultados obtidos (reações de apoio, momentos fletores, momentos torsores, esforços normais, esforços cortantes, deformações e deslocamentos) desse processo serão o ponto de partida para o dimensionamento de toda a estrutura. Por isso se faz necessário otimizar a teoria da análise estrutural ao máximo, para simular da forma mais fiel possível a maneira como a estrutura irá responder ao longo de sua vida útil.

Tendo esses princípios em destaque, o presente trabalho se propõe a criar uma programação bastante didática através do software MATLAB com foco total na realização de análises estruturais não lineares (um excepcional tipo de análise que faz por merecer maior destaque entre os profissionais da área, pois a mesma é frequentemente negligenciada) (será explicada de forma particular em capítulos posteriores), o decorrer desse trabalho contará ainda com o uso de valorosos artifícios, como o Método dos Elementos Finitos.

1.1.Delimitação

Dentro do estudo da análise estrutural há alguns pontos específicos de vital importância a serem considerados para a realização desse trabalho, donde consequentemente serão obtidas as limitações funcionais do mesmo. Seguem abaixo:

1º. Não linearidade geométrica: Na presença de deformações (ou deslocamentos) excessivas podem aparecer novos esforços internos causados pela nova configuração deformada da estrutura (também conhecidos como efeitos de segunda ordem). Existe também a não linearidade física, onde alguns materiais apresentam a particularidade de responderem de uma forma não linear na relação entre tensão e deformação, um exemplo desse fato são os materiais com características viscoelásticas que apresentam simultaneamente deformações

elásticas e viscosas. Mas este tipo de comportamento físico não será considerado nesse trabalho.

2º. Estruturas reticuladas (ou lineares): É todo tipo de sistema estrutural que “se compõem de elementos em que predomina uma dimensão sobre as demais, elementos esses que tomam o nome de barras” (LANGENDONCK,1958,p.03). Serão considerados apenas estruturas com barras de eixo reto (treliças, vigas, pórticos e grelhas).

3º. Estruturas planas: Quando tanto a estrutura quanto seus carregamentos estão embutidos em um único plano de atuação (grelhas não estão incluídas nesse grupo). Eliminando assim estruturas espaciais do escopo desse trabalho.

4º. Será abortada apenas à análise para grandes deslocamentos e/ou rotações, mas que apresentam pequenas deformações. O estudo envolvendo grandes deformações geralmente leva em consideração o uso da análise não linear física.

1.2.Relevância e Justificativa

A escolha do tema foi causada pela necessidade de se demonstrar a importância da precisão dos cálculos de análise estrutural e a forma como eles devem moldar o real comportamento esperado de uma estrutura. Outro motivo que merece destaque é a crescente popularização do uso de softwares em todas as áreas da engenharia, tendo em vista os diversos benefícios obtidos como praticidade e capacidade de operação.

Além disso este mesmo também foi escolhido por interesse (e curiosidade) do autor na área estrutural, visando assim expandir os próprios conhecimentos, mas também tentar contribuir ao máximo às pesquisas de futuros leitores.

1.3.Objetivos

1.3.1. Objetivo geral

Desenvolver através do software Matlab um programa capaz de analisar uma estrutura plana levando em consideração o efeito da não linearidade geométrica

1.3.2. Objetivos específicos

Utilizar e descrever o Método dos Elementos Finitos no processo de determinação dos esforços solicitantes.

Apresentar resultados através de exemplos.

Mostrar a versatilidade do método dos elementos finitos para melhor difundir seu uso entre os estudantes e os profissionais da área.

1.4. Metodologia

Este projeto científico visa elaborar um programa que ofereça uma solução eficiente e compreensível prezando em primeiro lugar a análise estrutural geométrica não linear, tendo sua função canalizada para o uso em pórticos.

Na ordem de efetivar este propósito todo o desenvolvimento de conhecimento ao longo deste TCC teve seu embasamento teórico guiado por livros, publicações e artigos. É importante ressaltar que se faz extremamente necessário a compreensão de todas as bases teóricas aplicadas na metodologia de solução e idealização deste projeto (criação e o emprego desta ferramenta computacional), bem como em todas as etapas do processo de análise estrutural, e por isso será apresentado toda a fundamentação teórica desde um ponto inicial básico até um patamar conforme objetivado como necessário e consequentemente utilizado no código computacional vigente.

Primeiramente será mostrado então o que é e como funciona tanto a análise geométrica não linear quanto o método dos elementos finitos sendo feito nesse mesmo capítulo um passo a passo matemático demonstrando de que forma ambos irão se relacionar. Todas as deduções apresentadas resultantes do intenso e progressivo tratamento algébrico presente no decorrer dessa obra será voltada para a análise de pórticos planos.

Posteriormente será apresentada uma visão geral do código computacional, contendo ressalvas para sua estrutura e características particulares, em seguida será desenvolvido um exemplo como meio de comparação entre os resultados da análise estrutural considerando e desprezando o efeito de não-linearidade geométrica sobre uma mesma estrutura. Sendo este TCC concluído com propostas de aperfeiçoamento e refinamento visando abranger mais particularidades pontuais próprias do campo de conhecimento da análise estrutural.

1.5.Organização do Trabalho

O corrente trabalho científico foi organizado em 5 capítulos, este um introdutório que conta com delimitações, relevância e justificativas, objetivos, metodologia tendo por fim esta própria parte que descreve a composição deste TCC. O capítulo 2 conceitua e explica a análise estrutural não linear geométrica, também faz o mesmo com o método dos elementos finitos(MEF) além de expor a metodologia de solução não linear que será utilizada. O capítulo 3 relata o desenvolvimento computacional, suas características, estrutura de código e funcionamento. O capítulo 4 irá realizar um exemplo com a finalidade de comparar e validar os resultados obtidos. E por último o capítulo 5 onde estará presente a conclusão deste projeto assim como valorosas recomendações para trabalhos vindouros afora as referências bibliográficas.

2. ANÁLISE ESTRUTURAL NÃO LINEAR

Na análise linear há uma estipulação dos esforços levando em conta a configuração inicial de equilíbrio da estrutura (sem deslocamentos) durante todo o procedimento de cálculo o que nos leva a uma incoerência com o comportamento real de qualquer estrutura visto que é uma natureza intrínseca dos materiais apresentarem uma resposta na forma de deformações, deslocamentos e rotações quando solicitados por esforços mecânicos. Dessa forma brota daqui a necessidade de conceber uma análise mais completa e condizente com a realidade, para isso se faz uso da análise não linear, onde há a integração dos efeitos de não linearidade geométrica e a não linearidade física que dão origem aos chamados efeitos de segunda ordem. Como já comentando a não linearidade física não será abordado nesse trabalho.

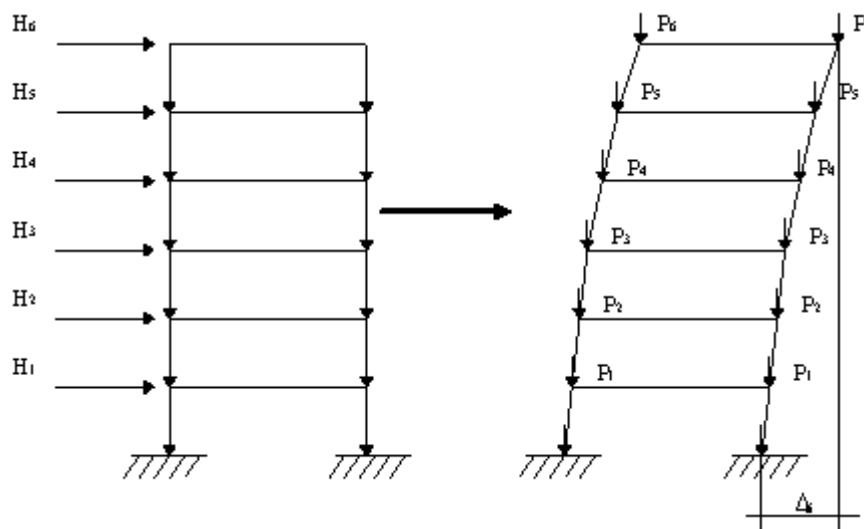
2.1. Não Linearidade Geométrica

Com relação à não linearidade geométrica, uma estrutura pode ter um comportamento não-linear, ainda que constituída de um material que obedeça à lei de Hooke. Para valores relativamente grandes de deslocamentos, a deflexão lateral de um elemento pode trazer, como consequência, o aparecimento de momentos fletores adicionais (denominados de segunda ordem), em virtude da presença de um esforço normal. A esse tipo de comportamento não linear dá-se o nome de não linearidade geométrica. Neste caso, os efeitos não lineares estão associados às equações de equilíbrio, as quais consideram a configuração deformada e as relações tensão-deformação (PEREIRA, 2002).

Segue abaixo algumas figuras que exemplificam como se dá essa origem de novos momentos fletores a partir da nova configuração deformada da estrutura, deixando em evidencia o desalinhamento provocado pelo deslocamento lateral da estrutura.

Em um pórtico:

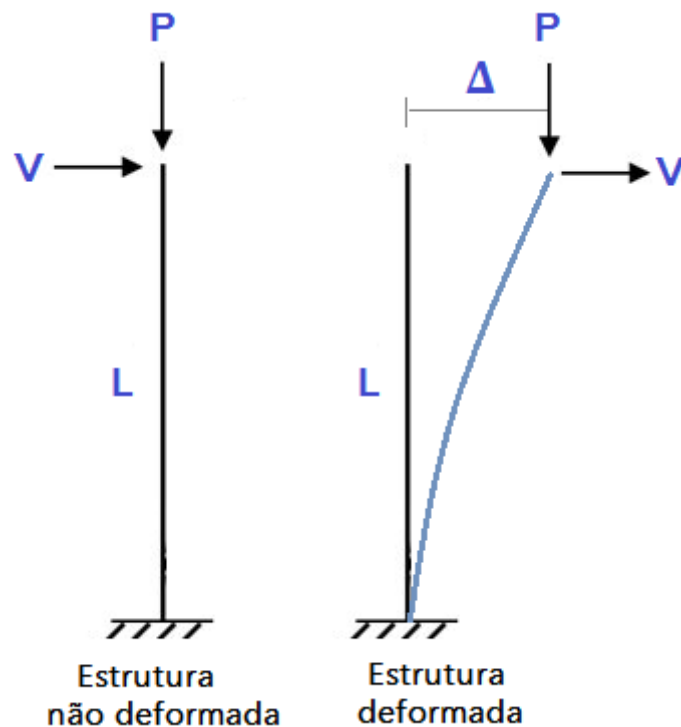
Figura 1: Deslocamento horizontal de um pórtico



Fonte: http://help.altoqi.com.br/Eberick/default.aspx?pageid=processo_pdelta

De uma maneira local:

Figura 2: Deslocamento horizontal de um elemento



Fonte: <http://maisengenharia.altoqi.com.br/estrutural/estruturas-de-nos-fixos-e-nos-moveis-aplicacao-do-p-delta/>

Então, considerando o efeito de não linearidade e obedecendo a lei de Hooke (linearidade física), teremos:

$$\Delta F = K \cdot \Delta U$$

Onde:

$\Delta F \rightarrow$ Forças deformantes.

$\Delta U \rightarrow$ Deformações produzidas.

$K \rightarrow$ Parâmetro de rigidez.

No entanto, eis aqui um ponto chave, pois a rigidez K assumirá um comportamento não linear, uma vez que a rigidez é decomposta em duas parciais, onde uma delas exibe uma dependência do caminho percorrido durante a deformação, ficando na seguinte forma geral:

$$K = K_S + K_N$$

Onde:

$K \rightarrow$ Matriz de rigidez.

$K_S \rightarrow$ Parcela linear.

$K_N \rightarrow$ Parcela não linear.

A parcial K_N se encontra em função do esforço interno axial Q_X originada da carga que o elemento estrutural esteja sofrendo, esforço esse que se encontra em função do deslocamento apresentado também pelo próprio elemento estrutural, daí vem a necessidade de um procedimento iterativo onde os referenciais sejam atualizados de forma simultânea. Todo esse raciocínio será matematicamente apresentado posteriormente.

2.2.Método dos Elementos Finitos

Em virtude da sua eficiência e capacidade em solucionar diversos problemas complexos da engenharia o MEF acabou se tornando o método mais aplicado dentro do enquadramento da análise estrutural para a obtenção de soluções numéricas.

O método consiste em discretizar (dividir) a geometria estrutural em pequenas partes denominadas de elementos estando esta quantidade em função da precisão almejada, são interligados nas extremidades pelos pontos nodais tendo como

condição de contorno os graus de liberdade (6 graus para um elemento de pórtico plano), sendo o comportamento do elemento regido pelas equações de equilíbrio.

A precisão deste método se encontra diretamente ligada às condições de convergência e ao refinamento da malha, que em condições limites tenderiam a obtenção da solução exata do problema. Em outras palavras, quanto maior o número de pontos da malha, maior o número de elementos, ou seja, mais refinada estará a malha e, portanto, mais próxima da solução analítica do problema será a resposta obtida. No entanto, deve-se ressaltar que a utilização de uma quantidade de elementos a gerar uma solução satisfatória dentro da precisão desejada e do tempo esperado refletem automaticamente em uma considerável redução no custo operacional, sem prejuízos referentes à confiabilidade dos resultados, pois após certo grau de precisão, as alterações na resposta obtida assumem valores muito pouco significativos (Galvão, 2000).

2.3. Metodologia de Solução

Será utilizado um procedimento iterativo de repetição para se obter a aproximação desejada (suficientemente próxima) nos resultados da análise estrutural não linear em comparação ao comportamento real esperado.

O procedimento terá como base inicial a equação geral do sistema estrutural que rege o comportamento da estrutura, segue abaixo:

$$K(Q_x)D = F$$

Onde:

$F \rightarrow$ Vetor de forças globais da estrutura.

$D \rightarrow$ Vetor de deslocamentos globais da estrutura.

$K(Q_x) \rightarrow$ Matriz de rigidez global da estrutura, estando essa variando em função das forças axiais internas Q_x .

Como o valor das forças internas Q_x são desconhecidas, o cálculo se inicia considerando $Q_x = 0$, de onde iremos obter, usando a última equação acima apresentada, o vetor de deslocamentos D_0 , e em seguida as reações de apoio correspondentes a esses deslocamentos, o que nos deixa aptos para determinar os esforços internos ao corpo da estrutura, inclusive os de natureza axial ($Q_{x,l}$) que serão utilizados na próxima iteração, e assim sucessivamente, até que a diferença

entre dois valores de forças axiais consecutivos convirja para um resultado suficientemente pequeno, fechando assim os ciclos de iterações, onde a última irá conter a solução aproximada ideal.

Do vetor de forças globais (F) teremos:

$$F = F_r + F_c$$

Onde:

$F_r \rightarrow$ Vetor das forças de reação de apoio (inicialmente desconhecidas).

$F_c \rightarrow$ Vetor de carregamento global da estrutura. É formado pela soma dos carregamentos nodais (já em coordenadas globais pela matriz topológica) de cada elemento de acordo com seus respectivos posicionamentos.

Nesse caso F_c precisa ser dividido em duas partes, ficando:

$$F_c = F_w + F_l$$

$F_w \rightarrow$ Vetor de cargas em função dos carregamentos distribuídos q_y , sendo necessário um tratamento matemático para transportar as cargas para as extremidades nodais do elemento.

$F_l \rightarrow$ Vetor de cargas localizadas que atuam nos nós do elemento.

2.4. Formulação Matemática

Partindo das seguintes premissas:

Deformação de Cauchy (deformação específica):

$$\varepsilon_x = \frac{du}{dx} \quad (1)$$

Onde:

$\varepsilon_x \rightarrow$ Deformação específica na direção x .

$dx \rightarrow$ Distância original infinitesimal entre dois pontos adjacentes na direção x .

$du \rightarrow$ Variação da distância entre os pontos (deformação).

Do diagrama tensão-deformação temos (fase linear elástica):

$$\sigma_x = E \cdot \varepsilon_x \quad (2)$$

Onde:

$\sigma_x \rightarrow$ Tensão axial na direção x.

E $\rightarrow$ Módulo de elasticidade. (Módulo de Young)

Tensão axial:

$$\sigma_x = \frac{N}{A} \quad (3)$$

Onde:

N $\rightarrow$ Força axial agindo no elemento.

A $\rightarrow$ Área da seção transversal.

De (2) e (3) teremos:

$$N_x = AE \cdot \varepsilon_x \quad (4)$$

Para acharmos a equação que governa o comportamento do momento devemos considerar um elemento estrutural que apresente deformação à ação de flexão, daí teremos:

Considerando pequenas rotações e usando semelhança de triângulos, tiramos:

$$\frac{y}{r} = \frac{du}{dx} \quad (5)$$

Onde:

y $\rightarrow$ Distância da superfície neutra da barra até uma fibra qualquer.

r $\rightarrow$ Raio de curvatura do eixo da barra deformado.

Substituindo (5) em (1):

$$\varepsilon_x = \frac{y}{r} \quad (6)$$

Substituindo (6) em (2):

$$\sigma_x = \frac{Ey}{r} \quad (7)$$

E sendo:

$$\sigma_x = \frac{N_x}{dA} \quad (8)$$

Onde:

$dA \rightarrow$ Área elementar da seção transversal distante y do eixo neutro.

De (7) e (8) teremos:

$$N_x = \frac{E y dA}{r} \quad (9)$$

$N_x \rightarrow$ Força axial única correspondente apenas à área dA da seção transversal.

E sabendo que:

$$M_1 = N_x \cdot y \quad (10)$$

Substituindo (9) em (10) teremos:

$$M_1 = \frac{E y^2 dA}{r} \quad (11)$$

Onde:

$M_1 \rightarrow$ Momento fletor único, correspondente à ação em dA .

Sabendo que o conjugado de momentos resistentes (M_1) na seção transversal devem serem somados para se equilibrar com o conjugado exterior M , teremos:

$$M = \int M_1$$

$$M = \int \frac{E y^2}{r} dA$$

Sendo E e r constantes ao longo da seção transversal.

Fica:

$$M = \frac{E}{r} \int y^2 dA \quad (12)$$

Observação: O somatório acontece com y variando da altura de zero (linha neutra) até um dos bordos externos.

Sabendo que:

$$I = \int y^2 dA \quad (13)$$

$I \rightarrow$ Momento de inercia da seção transversal.

E substituindo (13) em (12):

$$M = \frac{EI}{r} \quad (14)$$

Continuando considerando o elemento estrutural flexionado, temos ainda:

$$\tan(\theta) = \frac{dv}{dx}$$

Onde:

$dv \rightarrow$ Deslocamento transversal entre um ponto inicial e final do elemento dx pertencentes a uma mesma fibra.

$\theta \rightarrow$ Ângulo formado entre um ponto inicial e final do elemento dx pertencentes a uma mesma fibra.

Considerando que para pequenos ângulos (rotações) teremos:

$$\cos(\theta) \approx 1$$

$$\sin(\theta) \approx \tan(\theta) \approx \theta$$

Então:

$$\frac{dv}{dx} = \theta \quad (15)$$

Derivando uma vez em função de x :

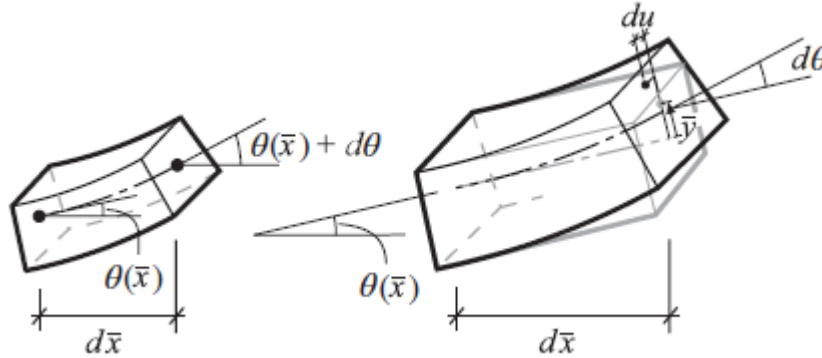
$$\frac{d^2v}{dx^2} = \frac{d\theta}{dx} \quad (16)$$

$d\theta \rightarrow$ Representa assim a mudança de ângulo que surge entre dois pontos adjacentes com uma distância dx entre eles.(Figura 3)

Podemos também formalizar e generalizar a expressão (1/r) da relação (14), introduzindo a notação $R(x)$ para representá-la, pois a mesma se trata de uma função que descreve a rotação sofrida ao longo da barra em consequência do deslocamento, variando assim de acordo com o posicionamento (x) em questão (coordenadas localizadas), ficando na seguinte forma:

$$R(x) = \frac{1}{r}$$

Figura 3: Deformação à ação de flexão



Fonte: OLSSON (2016)

Que ainda assume a forma:

$$R(x) = \frac{d\theta}{dx}$$

$R(x) \rightarrow$ Representa a função de curvatura em função de x (r varia de acordo com x).

Assumindo isso as equações (14) e (16) podem ser reescritas nas seguintes formas:

$$M = EI.R(x) \quad (17)$$

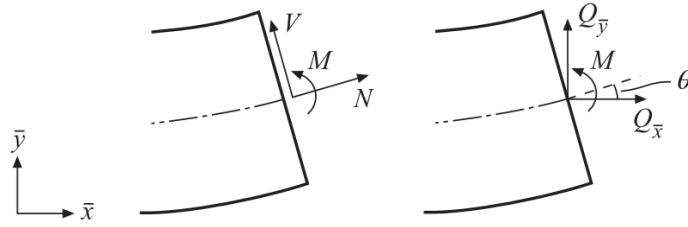
E:

$$R(x) = \frac{d^2v}{dx^2} \quad (18)$$

2.4.1. Equilíbrio da seção transversal deformada

Sabe-se que a força normal (N) e a força cortante (V) agem de forma perpendicular e paralela, respectivamente, ao plano da seção transversal, no entanto ao se considerar o equilíbrio no estado deformado, haverá um desalinhamento das forças (N e V) com o sistema de eixos locais, em função da rotação θ (figura 4), por isso iremos adicionar as notações Q_x (perpendicular) e Q_y (transversal) para representar as resultantes em relação ao sistema de eixos locais.

Figura 4: Forças expressas na seção deformada



Fonte: OLSSON (2016)

Daí tiramos:

$$\begin{cases} N_x = N \cdot \cos \theta \\ N_y = N \cdot \sin \theta \end{cases} \quad (19)$$

$$\begin{cases} V_x = V \cdot \sin \theta \\ V_y = V \cdot \cos \theta \end{cases} \quad (20)$$

$$\begin{cases} Q_x = N_x - V_x \\ Q_y = N_y + V_y \end{cases} \quad (21)$$

Substituindo (19) e (20) na (21), temos:

$$\begin{cases} Q_x = N \cdot \cos \theta - V \sin \theta \\ Q_y = N \cdot \sin \theta + V \cos \theta \end{cases} \quad (22)$$

A equação (22) na forma matricial, fica:

$$\begin{bmatrix} Q_x \\ Q_y \end{bmatrix} = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} N \\ V \end{bmatrix}$$

Inversamente (matriz transposta), teremos:

$$\begin{bmatrix} N \\ V \end{bmatrix} = \begin{bmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} Q_x \\ Q_y \end{bmatrix}$$

Considerando pequenas rotações, temos:

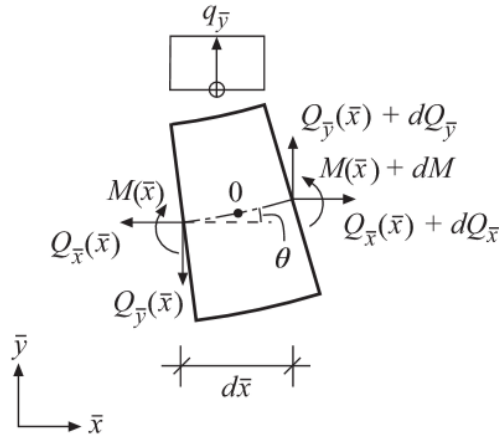
$$\begin{bmatrix} N \\ V \end{bmatrix} = \begin{bmatrix} 1 & \theta \\ -\theta & 1 \end{bmatrix} \begin{bmatrix} Q_x \\ Q_y \end{bmatrix}$$

De onde podemos tirar:

$$N = Q_x + \theta Q_y \quad (23)$$

Da fatia dx da barra flexionada tiramos:

Figura 5: Ação das forças na fatia dx



Fonte: OLSSON (2016)

Fazendo $\sum F_x = 0$, temos:

$$Q_x = Q_x + dQ_x$$

Ou:

$$\frac{dQ_x}{dx} = 0 \quad (24)$$

Fazendo $\sum F_y = 0$, temos:

$$-Q_y + (Q_y + dQ_y) + q_y dx = 0$$

$$\frac{dQ_y}{dx} + q_y = 0 \quad (25)$$

$q_y \rightarrow$ Carga atuante.

Fazendo $\sum M = 0$ (ponto de referência na direita), temos:

$$-M + (M + dM) + Q_y dx - Q_x dx \theta + q_y dx \frac{dx}{2} = 0$$

Desconsiderando o termo $q_y dx \frac{dx}{2}$, teremos:

$$\frac{dM}{dx} + Q_y - Q_x \theta = 0 \quad (26)$$

Substituindo (15) na (26), teremos:

$$\frac{dM}{dx} - Q_x \frac{dv}{dx} + Q_y = 0 \quad (27)$$

Derivando e assumindo Q_x constante segundo a relação (24), temos:

$$\frac{dM}{dx} - Q_x \frac{d^2v}{dx^2} + \frac{dQ_y}{dx} = 0 \quad (28)$$

Substituindo (25) na (28), teremos:

$$\frac{dM}{dx} - Q_x \frac{d^2v}{dx^2} - q_y = 0 \quad (29)$$

2.4.2. Equações Diferenciais

Teremos duas equações onde cada uma irá servir como ponto de partida para se obter a sua respectiva matriz de rigidez (localizada), sendo assim a matriz total é constituída de duas parcelas como citado no início deste capítulo, segue abaixo:

Associando (1) e (4) na (24), teremos:

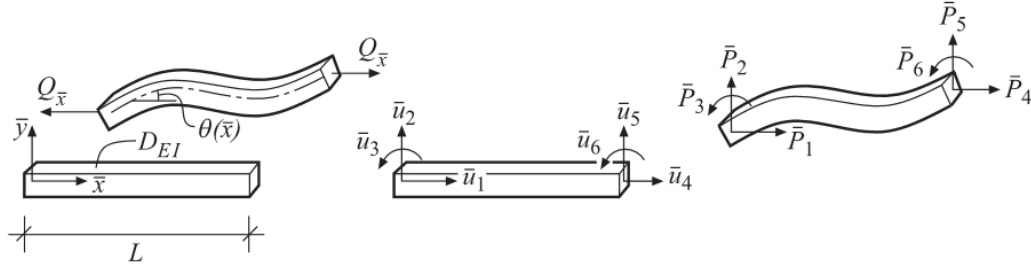
$$EA \frac{d^2u}{dx^2} = 0 \quad (30)$$

Associando (17) e (18) na (29), teremos:

$$EI \frac{d^4v}{dx^4} - Q_x \frac{d^2v}{dx^2} - q_y = 0 \quad (31)$$

A equação (31) irá dar origem à parcela não linear da rigidez por ser a única que sofre influência da força axial Q_x , que irá variar até se obter o equilíbrio da estrutura.

Figura 6: Elemento de barra com seis graus de liberdade.



Fonte: OLSSON (2016)

2.4.2.1. Resolvendo a primeira equação diferencial.

No intuito de solucionar o problema, a equação (30) terá o termo EA separado temporariamente por ser constante, o que acaba contribuindo no cálculo. Fica:

$$\frac{d^2 u}{dx^2} = 0 \quad (32)$$

Integrando (32) duas vezes:

$$u_h(x) = c_1 + c_2 x \quad (33)$$

Onde:

$u_h(x) \rightarrow$ Solução homogênea da equação diferencial, está em função do posicionamento ao longo do eixo x.

c_1 e $c_2 \rightarrow$ Constantes de integração.

Condições de contorno para $x=0$ e $x=L$ (nós do elemento, figura 6), seguem abaixo:

$$u_h(0) = u_1$$

$$u_h(L) = u_4$$

u_1 e $u_4 \rightarrow$ Deslocamentos nodais.

Aplicando-as na (33), teremos:

$$u_1 = c_1$$

$$u_4 = c_1 + c_2 L$$

Ou:

$$c_1 = u_1$$

$$c_2 = -\frac{u_1}{L} + \frac{u_4}{L}$$

Substituindo na (33), fica:

$$u_h(x) = u_1 - \frac{u_1 x}{L} + \frac{u_4 x}{L}$$

$$u_h(x) = \left(1 - \frac{x}{L}\right)u_1 + \left(\frac{x}{L}\right)u_4 \quad (34)$$

Na forma matricial temos:

$$u_h(x) = \begin{bmatrix} 1 - \frac{x}{L} & \frac{x}{L} \end{bmatrix} \begin{bmatrix} u_1 \\ u_4 \end{bmatrix} \quad (35)$$

Substituindo a (35) na (1) e sucessivamente na (4), obtemos a relação entre força normal e deslocamento nodal, na seguinte forma:

$$N = EA \begin{bmatrix} -\frac{1}{L} & \frac{1}{L} \end{bmatrix} \begin{bmatrix} u_1 \\ u_4 \end{bmatrix} \quad (36)$$

Da (23), teremos:

$$Q_x = N - \theta Q_y$$

Presumindo pequenas rotações o termo θQ_y pode ser negligenciado, ficando:

$$Q_x = N \quad (37)$$

Substituindo na (36) fica:

$$Q_x = EA \begin{bmatrix} -\frac{1}{L} & \frac{1}{L} \end{bmatrix} \begin{bmatrix} u_1 \\ u_4 \end{bmatrix} \quad (38)$$

Da figura 6, temos:

Para $x=0$:

$$P_1 = -Q_x \quad (39)$$

Para $x=L$:

$$P_4 = Q_x$$

Substituindo na (38), teremos:

$$P_1 = EA \begin{bmatrix} \frac{1}{L} & -\frac{1}{L} \end{bmatrix} \begin{bmatrix} u_1 \\ u_4 \end{bmatrix}$$

$$P_4 = EA \begin{bmatrix} -\frac{1}{L} & \frac{1}{L} \end{bmatrix} \begin{bmatrix} u_1 \\ u_4 \end{bmatrix}$$

Na forma matricial, temos:

$$\begin{bmatrix} P_1 \\ P_4 \end{bmatrix} = \frac{EA}{L} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_4 \end{bmatrix} \quad (40)$$

Sendo:

$$K_s = \begin{bmatrix} \frac{EA}{L} & -\frac{EA}{L} \\ -\frac{EA}{L} & \frac{EA}{L} \end{bmatrix} \quad (41)$$

Onde:

$K_s \rightarrow$ Matriz de rigidez local para ação de barra.

2.4.2.2. Resolvendo a segunda equação diferencial.

A resolução exata da equação diferencial (31) será dada pela equação homogênea que segue abaixo:

$$EI \frac{d^4 v}{dx^4} - Q_x \frac{d^2 v}{dx^2} = 0$$

E para $Q_x < 0$, teremos a seguinte solução:

$$v(x) = c_1 + c_2 x + c_3 \cos \lambda x + c_4 \sin \lambda x$$

Onde:

$$\lambda = \sqrt{-\frac{Q_x}{EI}}$$

E para $Q_x > 0$, teremos:

$$v(x) = c_1 + c_2 x + c_3 \cosh \lambda x + c_4 \sinh \lambda x$$

Onde:

$$\lambda = \sqrt{\frac{Q_x}{EI}}$$

E resulta na seguinte matriz de rigidez do elemento:

$$K_N = \frac{EI}{L^3} \begin{bmatrix} 12\phi_5 & 6L\phi_2 & -12\phi_5 & 6L\phi_2 \\ 6L\phi_2 & 4L^2\phi_2 & -6L\phi_2 & 2L^2\phi_4 \\ -12\phi_5 & -6L\phi_2 & 12\phi_5 & -6L\phi_2 \\ 6L\phi_2 & 2L^2\phi_4 & -6L\phi_2 & 4L^2\phi_3 \end{bmatrix} \quad (42)$$

$K_N \rightarrow$ Matriz de rigidez local para ação de viga.

Onde para $Q_x < 0$ teremos:

$$\phi_2 = \frac{\lambda^2 L^2}{12(1-\phi_1)}$$

$$\phi_3 = \frac{\phi_1}{4} + \frac{3\phi_2}{4}$$

$$\phi_4 = -\frac{\phi_1}{2} + \frac{3\phi_2}{2}$$

$$\phi_5 = \phi_1\phi_2$$

Sendo:

$$\phi_1 = \frac{\lambda L}{2} \cot \frac{\lambda L}{2}$$

E para $Q_x > 0$, teremos:

$$\phi_2 = -\frac{\lambda^2 L^2}{12(1-\phi_1)}$$

$$\phi_3 = \frac{\phi_1}{4} + \frac{3\phi_2}{4}$$

$$\phi_4 = -\frac{\phi_1}{2} + \frac{3\phi_2}{2}$$

$$\phi_5 = \phi_1\phi_2$$

Sendo:

$$\phi_1 = \frac{\lambda L}{2} \coth \frac{\lambda L}{2}$$

Na presença de carregamento distribuído teremos o seguinte vetor de cargas do elemento:

$$F_w = q_y \begin{bmatrix} \frac{L}{2} \\ \frac{L^2}{12} \Psi \\ \frac{L}{2} \\ -\frac{L^2}{12} \Psi \end{bmatrix}$$

$\Psi \rightarrow$ Fator de variação (causada pela deformação do elemento) para os momentos oriundos do carregamento distribuído.

Sendo:

Para $Q_x < 0$, teremos:

$$\Psi = \frac{6}{\lambda L} \left(\frac{2}{\lambda L} - \frac{1 + \cos \lambda L}{\sin \lambda L} \right)$$

E para $Q_x > 0$, teremos:

$$\Psi = -\frac{6}{\lambda L} \left(\frac{2}{\lambda L} - \frac{1 + \cosh \lambda L}{\sinh \lambda L} \right)$$

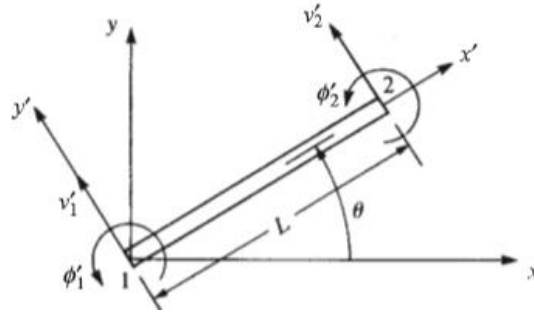
2.5. Matriz Transformação para seis graus de liberdade.

É a matriz com a função de transformar os referenciais (sistema de eixos) locais (de cada elemento) para globais, tornando possível uma associação entre as diferentes partes de uma estrutura bem como a resolução do problema como um todo.

Matriz Transformação

A essa matriz cabe a função de transformar (transportar) o sistema de coordenadas globais para coordenadas locais do elemento, e vice-versa utilizando a transposta da matriz transformação.

Figura 7: Elemento Arbitrario



Fonte: Daryl Logan (2012)

$(x', y') \rightarrow$ Sistema de eixo local do elemento.

$(x, y) \rightarrow$ Sistema de eixo global, situado para ser conveniente para toda a estrutura.

Da figura tiramos as seguintes relações:

$$\begin{cases} u_1 = H_1 \cos \theta + P_1 \sin \theta \\ v_1 = P_1 \cos \theta - H_1 \sin \theta \\ \phi_1 = \phi_1 \\ u_2 = H_2 \cos \theta + P_2 \sin \theta \\ v_2 = P_2 \cos \theta - H_2 \sin \theta \\ \phi_2 = \phi_2 \end{cases}$$

Na forma matricial fica:

$$\begin{bmatrix} u_1 \\ v_1 \\ \phi_1 \\ u_2 \\ v_2 \\ \phi_2 \end{bmatrix} = \begin{bmatrix} \cos \theta & \sin \theta & 0 & 0 & 0 & 0 \\ -\sin \theta & \cos \theta & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \cos \theta & \sin \theta & 0 \\ 0 & 0 & 0 & -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} H_1 \\ P_1 \\ \phi_1 \\ H_2 \\ P_2 \\ \phi_2 \end{bmatrix}$$

Sendo:

$$T_r = \begin{bmatrix} \cos \theta & \sin \theta & 0 & 0 & 0 & 0 \\ -\sin \theta & \cos \theta & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \cos \theta & \sin \theta & 0 \\ 0 & 0 & 0 & -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

E sua transposta:

$$T_r^t = \begin{bmatrix} \cos \theta & -\sin \theta & 0 & 0 & 0 & 0 \\ \sin \theta & \cos \theta & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \cos \theta & -\sin \theta & 0 \\ 0 & 0 & 0 & \sin \theta & \cos \theta & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Substituindo os vetores de carregamento (que inclui momentos nos nós) e deslocamento (que inclui a rotação sofridas nos nós) dentro do sistema de equações generalizadas, teremos:

$$f = T_r * F$$

E:

$$d = T_r * D$$

$f \rightarrow$ Vetor de forças locais.

$d \rightarrow$ Vetor de deslocamentos locais.

$F \rightarrow$ Vetor de força em coordenadas globais com relação a um elemento (n vetores de força para os n elementos da estrutura).

$D \rightarrow$ Vetor de deslocamentos globais.

$T_r \rightarrow$ Matriz transformação.

Para se obter a matriz de rigidez global iremos partir da equação fundamental:

$$f = d * K_L$$

$K_L \rightarrow$ Matriz de rigidez local.

Substituindo teremos:

$$F * T_r = D * T_r * K_L$$

$$F = D * T_r * K_L * T_r^t$$

De forma correspondente:

$$F = D * K_G$$

Portanto:

$$K_G = T_r^t * K_L * T_r$$

$T_r^t \rightarrow$ Transposta da matriz transformação.

$K_G \rightarrow$ Matriz de rigidez em coordenadas globais, mas com relação a apenas um elemento da estrutura, após se obter as n matrizes de rigidez globais dos n elementos será montada a matriz de rigidez global da estrutura através da superposição de linhas e colunas com relação aos seus respectivos graus de liberdade (no código faremos uso da matriz topológica).

2.6 Associação de Matrizes

Reescrevendo a relação (41) para os seis graus de liberdade do elemento de pórtico, teremos:

$$K_s = \begin{bmatrix} \frac{EA}{L} & 0 & 0 & -\frac{EA}{L} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ -\frac{EA}{L} & 0 & 0 & \frac{EA}{L} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}; F = \begin{bmatrix} P_1 \\ P_2 \\ P_3 \\ P_4 \\ P_5 \\ P_6 \end{bmatrix} \text{ e } D = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \end{bmatrix}$$

Fazendo o mesmo para a relação (42), teremos;

$$K_N = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{12EI\phi_5}{L^3} & \frac{6EI\phi_2}{L^2} & 0 & \frac{-12EI\phi_5}{L^3} & \frac{6EI\phi_2}{L^2} \\ 0 & \frac{6EI\phi_2}{L^2} & \frac{4EI\phi_3}{L} & 0 & \frac{-6EI\phi_2}{L^2} & \frac{2EI\phi_4}{L} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{-12EI\phi_5}{L} & \frac{-6EI\phi_2}{L} & 0 & \frac{12EI\phi_5}{L} & \frac{-6EI\phi_2}{L} \\ 0 & \frac{6EI\phi_2}{L} & \frac{2EI\phi_4}{L} & 0 & \frac{-6EI\phi_2}{L} & \frac{4EI\phi_3}{L} \end{bmatrix};$$

$$F_c = \begin{bmatrix} P_1 \\ P_2 \\ P_3 \\ P_4 \\ P_5 \\ P_6 \end{bmatrix} \quad e \quad D = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \end{bmatrix}$$

E fazendo (relação presente na seção 2.1 deste trabalho):

$$K = K_S + K_N \text{ (No código computacional será representada por 'Kel')}$$

Teremos por fim:

$$K = \begin{bmatrix} \frac{EA}{L} & 0 & 0 & -\frac{EA}{L} & 0 & 0 \\ 0 & \frac{12EI\phi_5}{L^3} & \frac{6EI\phi_2}{L^2} & 0 & \frac{-12EI\phi_5}{L^3} & \frac{6EI\phi_2}{L^2} \\ 0 & \frac{6EI\phi_2}{L^2} & \frac{4EI\phi_3}{L} & 0 & \frac{-6EI\phi_2}{L^2} & \frac{2EI\phi_4}{L} \\ -\frac{EA}{L} & 0 & 0 & \frac{EA}{L} & 0 & 0 \\ 0 & \frac{-12EI\phi_5}{L} & \frac{-6EI\phi_2}{L} & 0 & \frac{12EI\phi_5}{L} & \frac{-6EI\phi_2}{L} \\ 0 & \frac{6EI\phi_2}{L} & \frac{2EI\phi_4}{L} & 0 & \frac{-6EI\phi_2}{L} & \frac{4EI\phi_3}{L} \end{bmatrix}$$

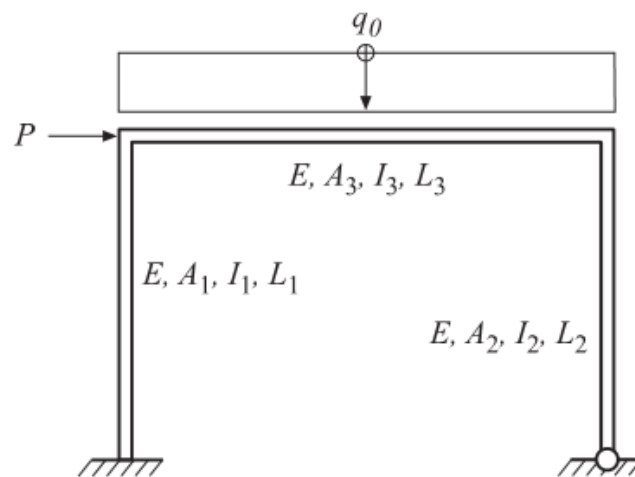
3. IMPLEMENTAÇÃO COMPUTACIONAL

O código computacional vigente neste trabalho foi elaborado no meio Matlab, no intuito de ser compreendido e utilizado até por aqueles que não apresentam um domínio técnico aprimorado na área da programação.

3.1. Visão geral do código computacional:

O funcionamento do código será exibido por meio da execução de um exemplo.

Figura 8: Pórtico exemplo



Fonte: OLSSON (2016)

Dados de entrada:

E, A, I e L . → Sua inserção ocorre entre as linhas 40 e 47 do código principal.

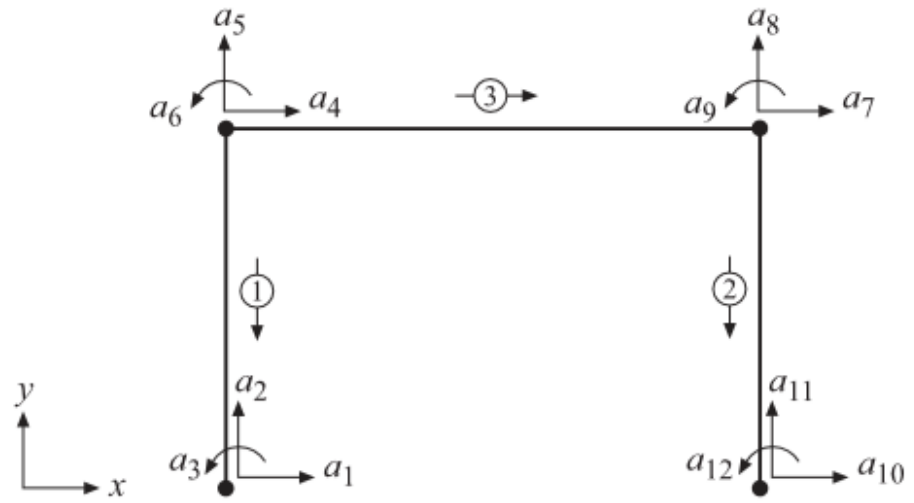
q_0 → Sua inserção ocorre na linha 52 do código principal, evidenciando em qual elemento o carregamento distribuído está atuando.

p → Sua inserção ocorre na linha 67 do código principal, evidenciando em qual grau de liberdade a carga localizada está atuando.

Matriz coord → Inserção dos pares ordenados dos nós, ocorre na linha 7 do código principal.

Matriz conec → Inserção das ligações (dois nós por linha da matriz) entre os nós, ocorre na linha 10 do código principal.

Figura 9: Modelo computacional



Fonte: OLSSON (2016)

Figura 10: Código Principal (1/4)

```
1 - clear all
2 - clc
3 - % Exemplo 1
4
5 - format short
6 - % Matriz coordenadas dos nós
7 - coord=[0 0;0 4;6 4;6 0];
8
9 - % Matriz conectividade dos elementos
10 - conec=[2 1;3 4;2 3];
11
12 - nel=size(conec,1);
13
14 - ex=zeros(nel,2);
15 - ey=zeros(nel,2);
16
17 - for i=1:nel
18 -     ex(i,:)=[coord(conec(i,1),1) coord(conec(i,2),1)];
19 -     ey(i,:)=[coord(conec(i,1),2) coord(conec(i,2),2)];
20 - end
21
22 - % Nós I e J
23 - I=conec(:,1);
24 - J=conec(:,2);
25
26 - Mtopologica=zeros(nel,7);
27 - Mtopologica(:,1)=[1:nel]';
28 - Mtopologica(:,2)=3*I-2;
29 - Mtopologica(:,3)=3*I-1;
30 - Mtopologica(:,4)=3*I;
31 - Mtopologica(:,5)=3*J-2;
32 - Mtopologica(:,6)=3*J-1;
33 - Mtopologica(:,7)=3*J;
34
35 - Mtopologica
36
37 - gdlest=max(max(Mtopologica));
38
39 - % Propriedades do elementos pe
40 - A=2e-3;
```

Fonte: Elaborado pelo autor (2017)

Figura 11: Código Principal (2/4)

```

39      % Propriedades do elementos pe
40 -    A=2e-3;
41 -    I=1.6e-5;
42 -    E=200e9;
43 -    pe=zeros(nel,3);
44 -    for i=1:nel
45 -        pe(i,:)=[E A I];
46 -    end
47 -    pe(3,:)=[200e9 6e-3 5.4e-5];
48
49
50      % carga nodal equivalente para carga uniformemente distribuida
51 -    eq=zeros(1,nel);
52 -    eq(3)=-50e3;
53
54      % INICIA A INTERAÇÃO
55
56 -    eps=0.0001;           % norma de parada
57 -    N=[0 0 0];           % força normal inicial
58 -    N0=[1 1 1];         % força normal da interação inicial anterior
59 -    n=0;                 % contador de interação
60
61 -    while (abs((N(1)-N0(1))/N0(1))>eps)
62 -        n=n+1
63 -        N
64 -        N0
65 -        K=zeros(gdlest)
66 -        f=zeros(gdlest,1);
67 -        f(4)=10e3;
68
69      % Propriedades do elementos pe
70      %A=2e-3;
71      %I=1.6e-5;
72      %E=200e9;
73      %pe=zeros(nel,3);
74      %for i=1:nel
75      %pe(i,:)=[E A I];
76      %end
77      %pe(3,:)=[200e9 6e-3 5.4e-5];

```

Fonte: Elaborado pelo autor (2017)

Figura 12: Código Principal (3/4)

```

77     %pe(3,:)= [200e9 6e-3 5.4e-5];
78
79
80     % carga nodal equivalente para carga uniformemente distribuida
81     %eq=zeros(nel,1);
82     %eq(3)=-50e3;
83
84     % rigidez e força de cada elemento e matriz de rigidez global
85     %ml=0;
86
87 -   for i=1:nel
88
89         %eq(3)=-50e3;
90 -     [Ke,fe]=matrizderigidezdaviga2Defeitossegundaordem(ex(i,:),ey(i,:),pe(i,:),N(i),eq(i));
91 -     [K,f]=Montagemrigidezglobal(Mtopologica(i,:),K,Ke,f,fe);
92
93 -   end
94
95     % condições de contorno
96 -   cc=[1 0;2 0;3 0;10 0;11 0];
97
98     % Resolve o sistema de equações e encontra os deslocamentos
99     % e as reações de apoio
100 -   [d,R]=Resolvesistema(K,f,cc);
101
102 -   Reacoes=[R(1);R(2);R(3);R(10);R(11)]
103 -   d
104
105     % Extrai os deslocamentos
106 -   Ed=extraideslocamento(Mtopologica,d);
107
108
109
110 -   VIGA=0;
111     %N2=zeros(1,nel);

```

Fonte: Elaborado pelo autor (2017)

Figura 13: Código Principal (4/4)

```
93 - for i=1:nel
94 -     es=esforco(ex(i,:),ey(i,:),pe(i,:),Ed(i,:),N(i),eq(i));
95
96 -     N1(:,i)=es(1,1);
97 -     Q(:,i)=es(:,2);
98 -     M(:,i)=es(:,3);
99
100
101
102
103     %N2(i)=N1;
104 - end
105 - NO=N;
106 - N=N1
107 - Q
108 - M
109
110 - if (n>20)
111 -     disp('A solução não converge')
112 -     return
113 - end
114
115
116
117 - end
118
119
120
121     %echo off
122 -     x=ex';
123 -     y=ey';
124 -     xc=x;
125 -     yc=y;
126 -     hold on
127 -     plot(x,y)
128 -     plot(xc,yc)
129 -     hold off
130 -     mx=min(min(ex)); my=min(min(ey)); Mx=max(max(ex)); My=max(max(ey));
131 -     axis([mx-2 Mx+2 my-2 My+2]);
```

Fonte: Elaborado pelo autor (2017)

3.2. Entendendo as Etapas

A seguir será exposto uma serie de figuras com o proposito de explicar o código computacional (além dos comentarios já presentes ilustrados na cor verde) apresentado desta vez em etapas, segue a serie de figuras a baixo:

3.2.1. Etapa inicial

Figura 14: Numeração, barras e nós

```
1 - clear all
2 - clc
3 - % Exemplo 1
4
5 - format short
6 - % Matriz coordenadas dos nós
7 - coord=[0 0;0 4;6 4;6 0];
8
9 - % Matriz conectividade dos elementos
10 - conec=[2 1;3 4;2 3];
11
12 - nel=size(conec,1);
13
14 - ex=zeros(nel,2);
15 - ey=zeros(nel,2);
16
17 - for i=1:nel
18 -     ex(i,:)=[coord(conec(i,1),1) coord(conec(i,2),1)];
19 -     ey(i,:)=[coord(conec(i,1),2) coord(conec(i,2),2)];
20 - end
```

Fonte: Elaborado pelo autor (2017)

format short → Estabelece notação com 4 dígitos decimais.

coord → Nome dado a matriz de coordenadas dos nós da estrutura.

conec → Nome dado a matriz que representa a conexão entre os nós, formando o corpo da estrutura.

nel → Número de elementos da estrutura obtido por meio da função size que cria um vetor linha de onde iremos tirar o valor presente na sua primeira coluna.

Criação das matrizes ex (contém o valor das abcissas dos nós) e ey (contém o valor das ordenadas dos nós) montadas com relação à

conectividade dos elementos, ambas serão inseridas em funções posteriores. Com base no exemplo segue abaixo:

$$ex = \begin{bmatrix} 0 & 0 \\ 6 & 6 \\ 0 & 6 \end{bmatrix} \text{ e } ey = \begin{bmatrix} 4 & 0 \\ 4 & 0 \\ 4 & 4 \end{bmatrix}$$

3.2.2. Segunda etapa.

Figura 15: Matriz topológica ou Referencial.

```

22 % Nós I e J
23 - I=conec(:,1);
24 - J=conec(:,2);
25
26 - Mtopologica=zeros(nel,7);
27 - Mtopologica(:,1)=[1:nel]';
28 - Mtopologica(:,2)=3*I-2;
29 - Mtopologica(:,3)=3*I-1;
30 - Mtopologica(:,4)=3*I;
31 - Mtopologica(:,5)=3*J-2;
32 - Mtopologica(:,6)=3*J-1;
33 - Mtopologica(:,7)=3*J;
34
35 - Mtopologica
36
37 - gdlest=max(max(Mtopologica))

```

Fonte: Elaborado pelo autor (2017)

Mtopologica → Nome dado a matriz que estabelece a relação entre os graus de liberdade locais e os graus de liberdade globais, auxiliando na montagem da matriz de rigidez global e no vetor de força global.

gdlest → Representa o número de graus de liberdade da estrutura, obtida através do uso da função max duas vezes extraindo assim o maior valor da matriz Mtopologica.

3.2.3. Terceira etapa.

Figura 16: Início das interações.

```
54 % INICIA A INTERAÇÃO
55
56 - eps=0.0001; % norma de parada
57 - N=[0 0 0]; % força normal inicial
58 - N0=[1 1 1]; % força normal da interação inicial anterior
59 - n=0; % contador de interação
60
61 - while (abs ( (N(1)-N0(1)) /N0(1)) >eps)
62 -     n=n+1
63 -     N
64 -     N0
65 -     K=zeros(gdlest)
66 -     f=zeros(gdlest,1);
67 -     f(4)=10e3;
```

Fonte: Elaborado pelo autor (2017)

eps → Valor de convergência escolhido para delimitar o número de iterações.

Declaração de variáveis.

N, N0 e n.

while → Enquanto o valor eps for menor do que o obtido na expressão matemática da linha 61 (função da diferença entre duas forças internas axiais sucessivas, como já explicado no capítulo anterior), uma serie de comandos continuarão a serem executados configurando assim as interações.

3.2.4. Quarta Etapa.

Figura 17: Loop para montagem da matriz de rigidez global.

```
87 - for i=1:nel
88
89 -     %eq(3)=-50e3;
90 -     [Ke,fe]=matrizderigidezdaviga2Defeitossegundaordem(ex(i,:),ey(i,:),pe(i,:),N(i),eq(i));
91 -     [K,f]=Montagemrigidezglobal(Mtopologica(i,:),K,Ke,f,fe);
92
93 - end
```

Fonte: Elaborado pelo autor (2017)

for → Todos os comandos entre for e end serão repetidos para i=1 até i=nel ou seja, igual ao número de elementos.

Os arquivos de ambas as funções presentes dentro do laço estão a seguir.

3.2.4.1. Função matrizderigidezdaviga2Defeitossegundaordem

Figura 18: Código de desenvolvimento da matriz de rigidez do elemento.

```

1 function [Ke,fe]=matrizderigidezdaviga2Defeitossegundaordem(ex,ey,pe,N,eq)
2 - L=sqrt((ex(2)-ex(1))^2+(ey(2)-ey(1))^2);
3 - costheta=(ex(2)-ex(1))/L; sentheta=(ey(2)-ey(1))/L;
4 - E=pe(1); A=pe(2); I=pe(3); rho=-N*L^2/(pi^2*E*I);
5 - KL=pi*sqrt(abs(rho))+eps;
6 - if rho>0
7 -     f1=(KL/2)/tan(KL/2);
8 -     f2=-(1/12)*KL^2/(1-f1);
9 -     f3=f1/4+3*f2/4;
10 -    f4=-f1/2+3*f2/2;
11 -    f5=f1*f2;
12 -    h=6*(2/KL^2-(1+cos(KL))/(KL*sin(KL)));
13 - elseif rho<0
14 -     f1=(KL/2)/tan(KL/2);
15 -     f2=-(1/12)*KL^2/(1-f1);
16 -     f3=f1/4+3*f2/4;
17 -     f4=-f1/2+3*f2/2;
18 -     f5=f1*f2;
19 -     h=6*(2/KL^2-(1+cos(KL))/(KL*sin(KL)));
20 - else
21 -     f1=1; f2=1; f3=1; f4=1; f5=1; h=1;
22 - end
23 - Kel=[E*A/L      0      0      -E*A/L      0      0 ;
24 -      0      12*E*I*f5/L^3      6*E*I*f2/L^2      0      -12*E*I*f5/L^3      6*E*I*f2/L^2;
25 -      0      6*E*I*f2/L^2      4*E*I*f3/L      0      -6*E*I*f2/L^2      2*E*I*f4/L;
26 -      -E*A/L      0      0      E*A/L      0      0 ;
27 -      0      -12*E*I*f5/L^3      -6*E*I*f2/L^2      0      12*E*I*f5/L^3      -6*E*I*f2/L^2;
28 -      0      6*E*I*f2/L^2      2*E*I*f4/L      0      -6*E*I*f2/L^2      4*E*I*f3/L];
29
30 - fel=eq*L*[0 1/2 L*h/12 0 1/2 -L*h/12]';
31 % fel=L*[7*qx/20 7*qy/20
32
33 - T=[costheta sentheta      0      0      0      0;
34 -     -sentheta costheta      0      0      0      0;
35 -     0      0      1      0      0      0;
36 -     0      0      0      costheta sentheta 0;
37 -     0      0      0      -sentheta costheta 0;
38 -     0      0      0      0      0      1];
39 - Ke=T'*Kel*T; fe=T'*fel;
40 %-----end-----

```

Fonte: Elaborado pelo autor (2017)

$f(1:5) \rightarrow$ Fatores oriundos da força axial interna, seus valores variam para cada interação, (em função de N).

$h \rightarrow$ Fator de variação (causada pela deformação do elemento) para os momentos oriundos do carregamento distribuído.

$K_{el} \rightarrow$ Matriz de rigidez do elemento, com as parciais linear e não linear juntas em uma única matriz.

$e_q \rightarrow$ Valor da carga distribuída.

$f_{el} \rightarrow$ Vetor de forças oriundas de cargas distribuídas que atuam no elemento.

$f_e \rightarrow$ Vetor de forças oriundas de cargas distribuídas que atuam no elemento, agora em coordenadas globais.

$T \rightarrow$ Matriz de transformação (já mencionada no capítulo anterior).

$K_e \rightarrow$ Matriz de rigidez em coordenadas globais mas com relação à apenas um elemento da estrutura.

3.2.4.2. Função Montagemrigidezglobal

Figura 19: Código de montagem da matriz de rigidez global.

```
1 function [K,f]=Montagemrigidezglobal(Mtopologica,K,Ke,f,fe)
2
3     [nel,n]=size(Mtopologica);
4     % nel numero de elemento da estrutura
5     % n numero da coluna da matriz topologica
6     st=Mtopologica(:,2:n);
7     % st submatriz topologica que contem só o grau de liberdade da
8     % estrutura
9     for i = 1:nel
10        K(st(i,:),st(i,:)) = K(st(i,:),st(i,:))+Ke;
11        if nargin==5
12            f(st(i,:))=f(st(i,:))+fe;
13        end
14    end
15    %-----end-----
```

Fonte: Elaborado pelo autor (2017)

$nargin \rightarrow$ Número de argumentos inseridos (nesse caso são 5: $Mtopologica$, k , ke , f , fe).

Nesse exemplo:

$$M_{topologica} = \left[\begin{array}{c|cccccc} 1 & 4 & 5 & 6 & 1 & 2 & 3 \\ 2 & 7 & 8 & 9 & 10 & 11 & 12 \\ 3 & 4 & 5 & 6 & 7 & 8 & 9 \end{array} \right]$$

$$st = \left[\begin{array}{cccccc} 4 & 5 & 6 & 1 & 2 & 3 \\ 7 & 8 & 9 & 10 & 11 & 12 \\ 4 & 5 & 6 & 7 & 8 & 9 \end{array} \right]$$

st → Matriz semitopológica.

$K \rightarrow$ Matriz de rigidez global da estrutura, é montada pela soma da matriz K_e com os valores da matriz K (que no primeiro ciclo do comando FOR na linha 87 do código principal estará preenchida por zeros) nos respectivos espaços (o número do grau de liberdade faz a localização dentro da matriz K).

Tomando como amostragem o exemplo, para onde $i=1$, teremos:

$$K([1 \ 2 \ 3 \ 4 \ 5 \ 6], [1 \ 2 \ 3 \ 4 \ 5 \ 6]) = K([1 \ 2 \ 3 \ 4 \ 5 \ 6], [1 \ 2 \ 3 \ 4 \ 5 \ 6]) + K_e$$

Ou seja, os espaços da matriz K contidos nas linhas 1, 2, 3, 4, 5 e 6 com a interseção das colunas 1, 2, 3, 4, 5 e 6, serão preenchidos pela soma do que já estava lá (no primeiro loop apenas zeros) com a matriz de rigidez global do elemento 'i' (nesse caso, do elemento 1), e assim sucessivamente.

O vetor de cargas globais será gerado de forma análoga a matriz K , no entanto nesse exemplo já haverá uma carga localizada inicial no grau de liberdade 4, como ilustrado na figura 8.

3.2.5. Quinta Etapa.

Figura 20: Etapa seguinte no código principal.

```
93 - end
94
95 % condições de contorno
96 - cc=[1 0;2 0;3 0;10 0;11 0];
97
98 % Resolve o sistema de equações e encontra os deslocamentos
99 % e as reações de apoio
100 - [d,R]=Resolvesistema(K,f,cc);
101
102 - Reacoes=[R(1);R(2);R(3);R(10);R(11)]
103 - d
104
105 % Extrai os deslocamentos
106 - Ed=extraideslocamento(Mtopologica,d);
107
```

Fonte: Elaborado pelo autor (2017)

cc→ Matriz que contém as condições de contorno conhecidas do problema, informando onde se localiza e qual o valor dos deslocamentos prescritos (nos apoios que impedem o movimento em uma direção torna o deslocamento nessa direção igual a zero, como visto na resolução do exemplo).

3.2.5.1. Função Resolvesistema

Figura 21: Código de resolução do sistema de equações não lineares.

```
1 function [d,R]=Resolvesistema(K,f,cc)
2 % d=Resolve(K,f)
3 % [d,R=Rx+Ry]=Resolve(K,f,bc)
4
5 if nargin==2 ;
6     d=K\f ;
7 elseif nargin==3;
8     [nd,nd]=size(K);
9     % gdl grau de liberdade total
10    gdl=[1:nd]';
11    %
12    d=zeros(size(gdl));
13    R=zeros(size(gdl));
14    % gdlp grau de liberdade preescrito
15    gdlp=cc(:,1);
16    % dp deslocamento preescrito
17    dp=cc(:,2);
18    % elimina do grau de liberdade total o grau de liberdade preescrito
19    gdl(gdlp)=[];
20    % dl deslocamento livre
21    dl=K(gdl,gdl)\(f(gdl)-K(gdl,gdlp)*dp);
22    %
23    d(gdlp)=dp;
24    d(gdl)=dl;
25 end
26 R=K*d-f;
27
28 %-----end-----
```

Fonte: Elaborado pelo autor (2017)

gdl → Vetor coluna partindo do valor inicial 1 até o grau de liberdade máximo.

(Ver figura 9).

Declarando 'R' e 'd'.

O vetor grau de liberdade preescrito será igual a primeira coluna da matriz cc.

O vetor deslocamento preescrito será igual a segunda coluna da matriz cc.

[] → Comando que elimina um arranjo que está inserido em outro.

gdl → Graus de liberdade totais, é aonde a estrutura irá apresentar deslocamentos não preescritos (não conhecidos), se

tornando as incógnitas da equação (onde os deslocamentos são livres) sendo as forças atuantes já conhecidas nesses graus de liberdade.

R→ Reações, sua determinação ocorre pela equação fundamental já citada na seção 2.3 deste trabalho, linha 26.

3.2.5.2. Função extraideslocamento

Figura 22: Matriz ed

```
1  function [ed]=extraideslocamento(Mtopologica,d)
2
3
4  [nel,n]=size(Mtopologica);
5  %
6  st=Mtopologica(:,2:n);
7  %
8  for i = 1:nel
9      ed(i,1:(n-1))=d(st(i,:))';
10 end
11 %
12 %-----end-----
```

Fonte: Elaborado pelo autor (2017)

Formação da matriz 'ed' preenchida pelo valor dos deslocamentos oriundos do vetor de deslocamento global 'd', de acordo com seus graus de liberdade. No exemplo corrente a matriz ed terá 3 linhas e 6 colunas (n-1, sempre coincide com número de graus de liberdade do elemento).

3.2.6. Sexta Etapa

Figura 23: Determinação dos esforços.

```
112 - for i=1:nel
113 -     es=esforco(ex(i,:),ey(i,:),pe(i,:),Ed(i,:),N(i),eq(:,i));
114 -
115 -     NO=N
116 -
117 -     N1=es(:,1)
118 -     Q=es(:,2);
119 -     M=es(:,3);
120 -
121 -     %esforcosnaviga(:,1)=N;
122 -     %esforcosnaviga(:,2)=Q;
123 -     %esforcosnaviga(:,3)=M;
124 -     %VIGA=VIGA+1
125 -     %disp('Esforço Normal Esforço Cortante Momento fletor')
126 -     %disp(' ')
127 -     %disp(esforcosnaviga)
128 -
129 -     N2(i)=N1;
130 -     N2
```

Fonte: Elaborado pelo autor (2017)

Do comando ‘for’ determinamos que será utilizado um elemento por vez, implicando assim numa sistematização de seleção (para utilização dentro da função ‘esforco’) da respectiva linha ou coluna das matrizes dos argumentos que serão inseridos no correspondente ciclo que estará embutida a função esforço.

3.2.6.1. Função esforço

Figura 24: Valores de P

```

1  function es=esforco(ex,ey,pe,ed,N,eq)
2  -   L=sqrt((ex(2)-ex(1))^2+(ey(2)-ey(1))^2);
3  -   costheta=(ex(2)-ex(1))/L; sentheta=(ey(2)-ey(1))/L;
4  -   E=pe(1); A=pe(2); I=pe(3); rho=-N*L^2/(pi^2*E*I);
5  -   KL=pi*sqrt(abs(rho))+eps;
6  -   if rho>0
7  -       f1=(KL/2)/tan(KL/2);
8  -       f2=-(1/12)*KL^2/(1-f1);
9  -       f3=f1/4+3*f2/4;
10 -       f4=-f1/2+3*f2/2;
11 -       f5=f1*f2;
12 -       h=6*(2/KL^2-(1+cos(KL))/(KL*sin(KL)));
13 -   elseif rho<0
14 -       f1=(KL/2)/tan(KL/2);
15 -       f2=-(1/12)*KL^2/(1-f1);
16 -       f3=f1/4+3*f2/4;
17 -       f4=-f1/2+3*f2/2;
18 -       f5=f1*f2;
19 -       h=6*(2/KL^2-(1+cos(KL))/(KL*sin(KL)));
20 -   else
21 -       f1=1; f2=1; f3=1; f4=1; f5=1; h=1;
22 -   end
23 -   Kel=[E*A/L      0      0      -E*A/L      0      0 ;
24 -        0      12*E*I*f5/L^3      6*E*I*f2/L^2      0      -12*E*I*f5/L^3      6*E*I*f2/L^2;
25 -        0      6*E*I*f2/L^2      4*E*I*f3/L      0      -6*E*I*f2/L^2      2*E*I*f4/L;
26 -       -E*A/L      0      0      E*A/L      0      0 ;
27 -        0      -12*E*I*f5/L^3      -6*E*I*f2/L^2      0      12*E*I*f5/L^3      -6*E*I*f2/L^2;
28 -        0      6*E*I*f2/L^2      2*E*I*f4/L      0      -6*E*I*f2/L^2      4*E*I*f3/L];
29 -   fel=eq*L*[0 1/2 L*h/12 0 1/2 -L*h/12]';
30 -   % fel=L*[7*qx/20 7*qy/20
31 -   T=[costheta sentheta      0      0      0      0;
32 -      -sentheta costheta      0      0      0      0;
33 -        0      0      1      0      0      0;
34 -        0      0      0      costheta sentheta 0;
35 -        0      0      0      -sentheta costheta 0;
36 -        0      0      0      0      0      1];
37 -   u=ed';
38 -   P=(Kel*T*u-fel);
39 -   es=[-P(1) -P(2) -P(3) P(4) P(5) P(6)];
40 -   %-----end-----

```

Fonte: Elaborado pelo autor (2017)

Está função irá calcular os esforços atuantes nas extremidades de cada elemento, sendo armazenados no arranjo 'es' na seguinte ordem: Esforço normal (p1 e p4), esforço cortante (p2 e p5) e momento fletor (p3 e p6).

3.2.7. Etapa Final.

Figura 25: Fim das interações e geração de ilustração.

```
109 - end
110 - NO=N;
111 - N=N1
112
113 - if (n>20)
114 -     disp('A solução não converge')
115 -     return
116 - end
117
118
119
120 - end
121
122
123
124 %echo off
125 - x=ex';
126 - y=ey';
127 - xc=x;
128 - yc=y;
129 - hold on
130 - plot(x,y)
131 - plot(xc,yc)
132 - hold off
133 - mx=min(min(ex)); my=min(min(ey)); Mx=max(max(ex)); My=max(max(ey));
134 - axis([mx-2 Mx+2 my-2 My+2]);
```

Fonte: Elaborado pelo autor (2017)

Através do comando ‘if’ se estipula o fim das interações caso as mesmas ultrapassem 20 voltas sem cumprir com a condição imposta pelo comando ‘while’ na linha 61 do código principal. Entre as linhas 125 e 134 teremos o código que irá gerar uma ilustração do corpo da estrutura.

Figura 26: Reações e deslocamentos.

```

Reacoes =

    1.0e+05 *

    0.0817
    1.4241
    0.0803
   -0.1817
    1.5759

d =

    0
    0
    0
    0.0452
   -0.0014
   -0.0281
    0.0451
   -0.0016
    0.0239
    0
    0
   -0.0296
  
```

Fonte: Elaborado pelo autor (2017)

Figura 27: Esforços nas 3 barras.

```

N =

    1.0e+05 *

   -1.4241   -1.5759   -0.1817

Q =

    1.0e+05 *

    0.0817   -0.1817   -1.4241
    0.0817   -0.1817    1.5759

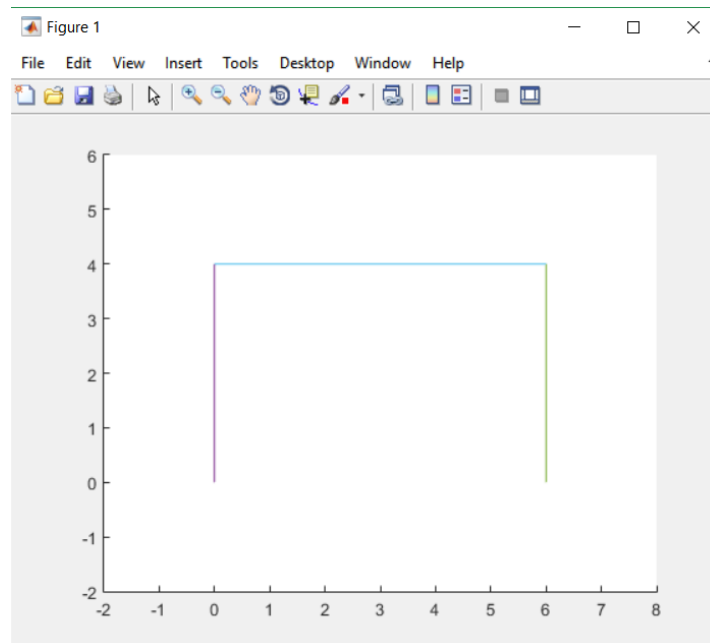
M =

    1.0e+04 *

    3.4280   -7.9796   -3.4280
    0.8027    0.0000   -7.9796
  
```

Fonte: Elaborado pelo autor (2017)

Figura 28: Ilustração do portico



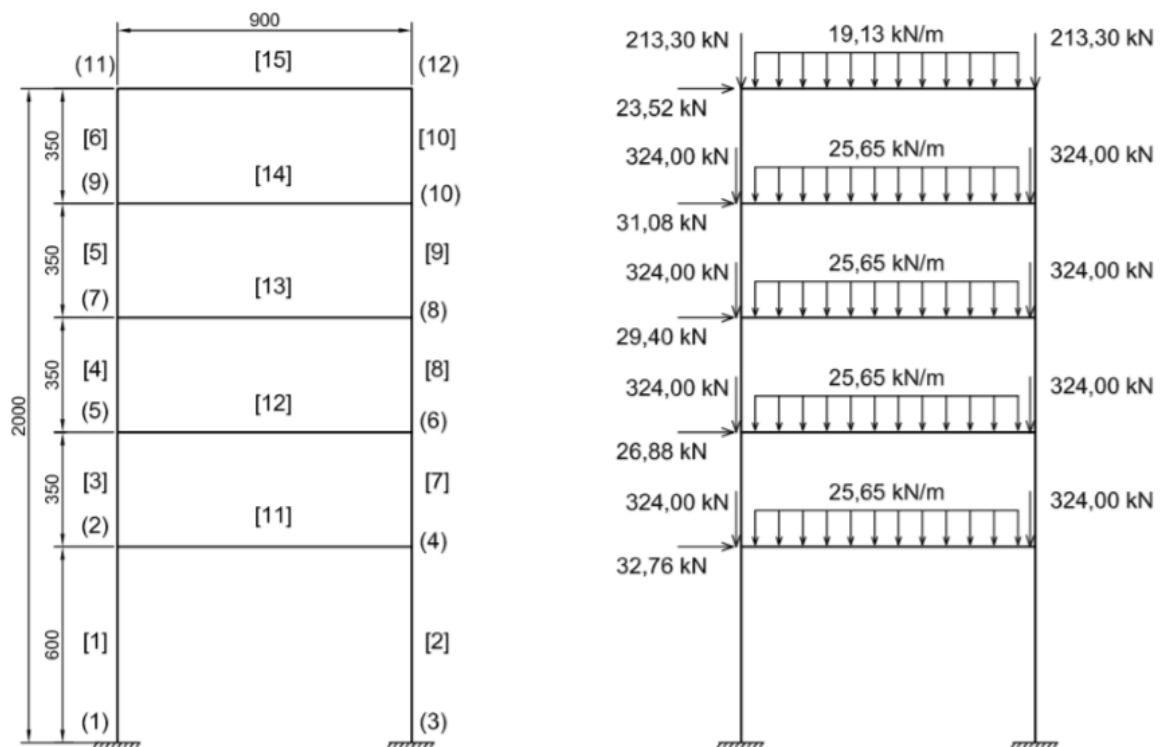
Fonte: Elaborado pelo autor (2017)

4. VALIDAÇÃO DE RESULTADOS POR COMPARAÇÃO.

Neste capítulo será efetuado uma situação de comparação na solução obtida para uma mesma estrutura, entre o programa aqui criado e um exemplo extraído da obra: Leitão, Giles Bortolon. Análise Numérica de Segunda Ordem de Pórticos Planos de Estruturas de Aço. 2014. 120. Dissertação de Mestrado – Universidade Estadual de Campinas, Campinas, 2014. Onde o exemplo também é solucionado através de um programa.

Exemplo:

Figura 29: Corpo da estrutura contendo suas dimensões, a numeração de barras e nós, em conjunto com os carregamentos atuantes no pórtico.



Fonte: Leitão Giles (2014)

Figura 30: Propriedade das barras

Barra	Perfil	A (cm²)	I_z(cm⁴)
1	W460x158	201,00	79500,00
2	W460x158	201,00	79500,00
3	W460x128	144,00	63683,00
4	W460x128	144,00	63683,00
5	W460x128	144,00	63683,00
6	W460x128	144,00	63683,00
7	W460x128	144,00	63683,00
8	W460x128	144,00	63683,00
9	W460x128	144,00	63683,00
10	W460x128	144,00	63683,00
11	W460x68	87,60	29851,00
12	W460x68	87,60	29851,00
13	W460x52	66,60	21370,00
14	W460x52	66,60	21370,00
15	W360x44	66,60	12112,00

Fonte: Leitão Giles (2014)

Figura 31: Código Principal (1/5)

```

1 - clear all
2 - clc
3 - % Exemplo 2
4
5 - format short
6 - % Matriz coordenadas dos nós
7 - coord=[0 0;0 6;9 0;9 6;0 9.5;9 9.5;0 13;9 13;0 16.5;9 16.5;0 20;9 20];
8
9 - % Matriz conectividade dos elementos
10 - conec=[1 2;3 4;2 5;5 7;7 9;9 11;4 6;6 8;8 10;10 12;2 4;5 6;7 8;9 10;11 12];
11
12 - nel=size(conec,1);
13
14 - ex=zeros(nel,2);
15 - ey=zeros(nel,2);
16
17 - for i=1:nel
18 -     ex(i,:)=[coord(conec(i,1),1) coord(conec(i,2),1)];
19 -     ey(i,:)=[coord(conec(i,1),2) coord(conec(i,2),2)];
20 - end
21
22 - % Nós I e J
23 - I=conec(:,1);
24 - J=conec(:,2);
25
26 - Mtopologica=zeros(nel,7);
27 - Mtopologica(:,1)=[1:nel]';
28 - Mtopologica(:,2)=3*I-2;
29 - Mtopologica(:,3)=3*I-1;
30 - Mtopologica(:,4)=3*I;
31 - Mtopologica(:,5)=3*J-2;
32 - Mtopologica(:,6)=3*J-1;
33 - Mtopologica(:,7)=3*J;
34
35
36 - gdlest=max(max(Mtopologica));

```

Fonte: Elaborado pelo autor (2017)

Figura 32: Código Principal (2/5)

```
38      % Propriedades do elementos pe
39 -    A1=201e-4;
40 -    I1=79500e-8;
41 -    A2=144e-4;
42 -    I2=63683e-8;
43 -    A3=876e-5;
44 -    I3=29851e-8;
45 -    A4=666e-5;
46 -    I4=21370e-8;
47 -    A5=A4;
48 -    I5=12112e-8;
49 -    E=200e9;
50
51 -    for i=1:2
52 -        pe(i,:)=[E A1 I1];
53 -    end
54 -    for i=3:10
55 -        pe(i,:)=[E A2 I2];
56 -    end
57 -    for i=11:12
58 -        pe(i,:)=[E A3 I3];
59 -    end
60 -    for i=13:14
61 -        pe(i,:)=[E A4 I4];
62 -    end
63 -    pe(15,:)=[E A5 I5];
64
65      % carga nodal equivalente para carga uniformemente distribuida
66 -    eq=zeros(1,nel);
67 -    for i=11:14
68 -        eq(i)=-25.63e3;
69 -    end
70 -    eq(15)=-19.13e3;
```

Fonte: Elaborado pelo autor (2017)

Figura 33: Código Principal (3/5)

```

72 % INICIA A INTERAÇÃO
73
74 - eps=0.0001; % norma de parada
75 - N=zeros(1,nel); % força normal inicial
76 - N0=ones(1,nel); % força normal da interação inicial anterior
77 - n=0; % contador de interação
78
79 - while (abs ( N(1)-N0(1) )/N0(1))>eps)
80 -     n=n+1
81
82 -     K=zeros(gdlest);
83 -     f=zeros(gdlest,1);
84 -     f(4)=32.76e3;
85 -     f(13)=26.88e3;
86 -     f(19)=29.4e3;
87 -     f(25)=31.08e3;
88 -     f(31)=23.52e3;
89 -     f(5)=-324e3;
90 -     f(14)=-324e3;
91 -     f(20)=-324e3;
92 -     f(26)=-324e3;
93 -     f(32)=-213.3e3;
94 -     f(11)=-324e3;
95 -     f(17)=-324e3;
96 -     f(23)=-324e3;
97 -     f(29)=-324e3;
98 -     f(35)=-213.3e3;
99
100
101 - for i=1:nel
102
103
104 -     [Ke,fe]=matrizderigidezdaviga2Defeitossegundaordem(ex(i,:),ey(i,:),pe(i,:),N(i),eq(i));
105 -     [K,f]=Montagemrigidezglobal(Mtopologica(i,:),K,Ke,f,fe);
106
107 - end

```

Fonte: Elaborado pelo autor (2017)

Figura 34: Código Principal (4/5)

```
110 % condições de contorno
111 - cc=[1 0;2 0;3 0;7 0;8 0;9 0];
112
113 % Resolve o sistema de equações e encontra os deslocamentos
114 % e as reações de apoio
115 - [d,R]=Resolvesistema(K,f,cc);
116
117 - Reacoes=[R(1);R(2);R(3);R(7);R(8);R(9)]
118 - d
119
120 % Extrai os deslocamentos
121 - Ed=extraideslocamento(Mtopologica,d);
122
123 - M=zeros(2,nel);
124 - Q=zeros(2,nel);
125 - Nl=zeros(1,nel);
126 - VIGA=0;
127
128 - for i=1:nel
129 - es=esforco(ex(i,:),ey(i,:),pe(i,:),Ed(i,:),N(i),eq(i));
130
131 - Nl(:,i)=es(1,1);
132 - Q(:,i)=es(:,2);
133 - M(:,i)=es(:,3);
134
135
136
137
138 %N2(i)=Nl;
139 - end
```

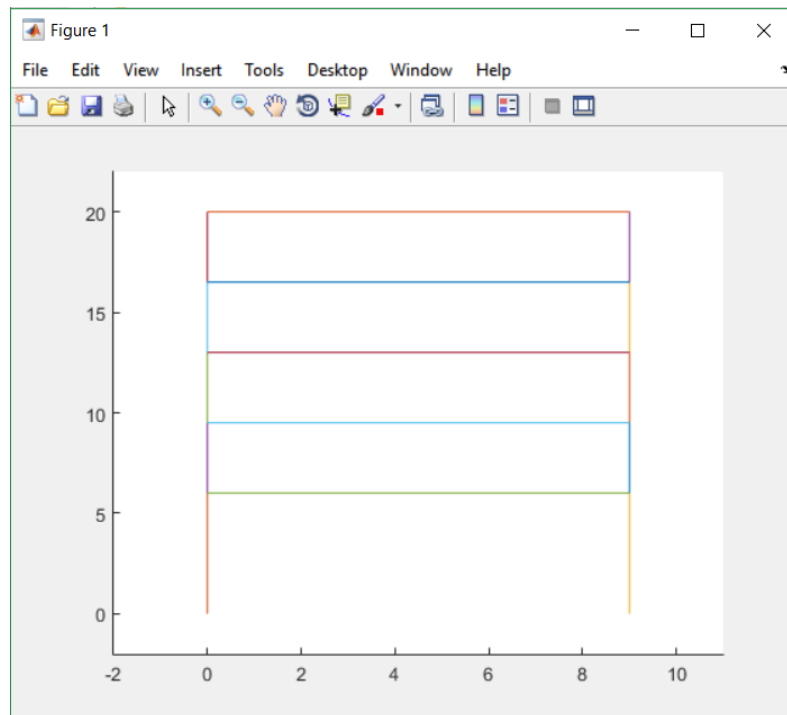
Fonte: Elaborado pelo autor (2017)

Figura 35: Código Principal (5/5)

```
139 - end
140 - N0=N;
141 - N=N1
142 - Q
143 - M
144
145 - if (n>20)
146 -     disp('A solução não converge')
147 -     return
148 - end
149
150
151
152 - end
153
154
155
156 %echo off
157
158 - x=ex';
159 - y=ey';
160 - xc=x;
161 - yc=y;
162 - hold on
163 - plot(x,y)
164 - plot(xc,yc)
165 - hold off
166 - mx=min(min(ex)); my=min(min(ey)); Mx=max(max(ex)); My=max(max(ey));
167 - axis([mx-2 Mx+2 my-2 My+2]);
```

Fonte: Elaborado pelo autor (2017)

Figura 36: Corpo da estrutura gerado pelo código.



Fonte: Elaborado pelo autor (2017)

Figura 37: Esforço Normal de primeira ordem.

```

N =
    1.0e+06 *

Columns 1 through 8

    -1.8862    -2.2273    -1.5763    -1.0396    -0.7944    -0.3550    -1.6585    -1.3165

Columns 9 through 15

    -0.6831    -0.2437     0.0234    -0.0215    -0.0178         0    -0.0737
    
```

Fonte: Elaborado pelo autor (2017)

Figura 38: Esforço Normal de segunda ordem.

```
N =  
  
1.0e+06 *  
  
Columns 1 through 8  
-1.8862    -2.2273    -1.5763    -1.0396    -0.7944    -0.3550    -1.6585    -1.3165  
  
Columns 9 through 15  
-0.6831    -0.2437     0.0234    -0.0215    -0.0178         0    -0.0737
```

Fonte: Elaborado pelo autor (2017)

Figura 39: Resultados obtidos por Leitão Giles.

Barra	Nó	N_{Sd} (kN)	
		Primeira ordem	Segunda ordem
1	1	1929,91	1894,53
	2	-1929,91	-1894,53
2	3	2184,26	2219,64
	4	-2184,26	-2219,64
3	2	1530,46	1506,91
	5	-1530,46	-1506,91
4	5	1127,97	1115,13
	7	-1127,97	-1115,13
5	7	712,12	705,64
	9	-712,12	-705,64
6	9	290,81	288,86
	11	-290,81	-288,86
7	4	1704,86	1728,41
	6	-1704,86	-1728,41
8	6	1228,5	1241,34
	8	-1228,5	-1241,34
9	8	765,5	771,98
	10	-765,5	-771,98
10	10	307,96	309,91
	12	-307,96	-309,91
11	2	-22,18	-21,22
	4	22,18	21,22
12	5	20,45	21,61
	6	-20,45	-21,61
13	7	14,8	16,09
	8	-14,8	-16,09
14	9	1,44	2,15
	10	-1,44	-2,15
15	11	73,06	74,2
	12	-73,06	-74,2

Fonte: Leitão Giles (2014)

Realizando a comparação visual (figuras 34 e 35 com figura 36) dos valores dos esforços normais tanto da primeira interação (esforços de primeira ordem) quanto dos esforços de segunda ordem obtidos da última interação do ciclo nos darão uma aproximação bastante congruente e satisfatória.

5 CONCLUSÃO

Após expor todo o conhecimento teórico ao longo deste trabalho, foi possível a criação e aplicação de um programa que tem a capacidade de obter esforços, deslocamentos e reações de apoio de uma estrutura (pórtico plano) considerando o efeito da não linearidade geométrica.

O programa apresenta na sua maioria, um código de nível acessível em termos de dificuldade, favorecendo uma boa compreensão e principalmente um manejo sistemático dos valores de entrada, essenciais para o pleno funcionamento do mesmo. Salientou-se também a eficácia do mesmo, através dos bons resultados obtidos em comparação a um exemplo retirado da bibliografia.

5.1 Sugestões para trabalhos futuros

Criação de um sistema de interfaces que proporcionem uma melhora na praticidade do uso deste programa.

Aprimoramento do código para exibição de gráficos objetivando ilustrar as ações dos esforços variando ao longo do corpo da estrutura.

Inclusão da capacidade de se analisar estruturas planas somente constituídas de treliças.

REFERÊNCIAS

1. GILAT A. MATLAB com Aplicações em Engenharia: 2 ed. Porto Alegre: Bookman, 2006.
2. GILAT A.; SUBRAMANIAM V. Métodos Numéricos para Engenheiros e Cientistas: uma introdução com aplicações usando MATLAB. Porto Alegre: Bookman, 2008.
3. Iglesia – Socrate Muñoz – Estruturas de Nós Móveis e Aplicação do P-Delta – 2016 - <http://maisengenharia.altoqi.com.br/estrutural/estruturas-de-nos-fixos-e-nos-moveis-aplicacao-do-p-delta/> - Acesso em 28 Set. 2017.
4. Lacerda, Estéfane George Macedo de. Análise Não Linear de Treliças pelo Método dos Elementos Finitos Posicional. 2014. 92. Dissertação de Pós-Graduação – Universidade Federal do Rio Grande do Norte, Natal, 2014.
5. LANGENDONCK T. V. Vigas Articuladas, Arcos e Pórticos Triarticulados: 1 ed. Rio de Janeiro: Científica, 1958.
6. Leitão, Giles Bortolon. Análise Numérica de Segunda Ordem de Pórticos Planos de Estruturas de Aço. 2014. 120. Dissertação de Mestrado – Universidade Estadual de Campinas, Campinas, 2014.
7. LOGAN D. L. A first course in Finite Element Method: 5 ed. Stamford: Cengage Learning, 2012.
8. Não Linearidade Geométrica: Processo P-Delta – 2016 - http://help.altoqi.com.br/Eberick/default.aspx?pageid=processo_pdelta – Acesso em 28 Set. 2017.

9. Neto, Ederson Rosental Pinheiro. Desenvolvimento de uma Aplicação Computacional para o Dimensionamento de uma Tesoura de Madeira de Cobertura, utilizando o Matlab. 2017. 116. Dissertação de Conclusão de Curso – Universidade Estadual da UEMA, São Luís, 2017
10. OLSSON K.; DAHLBOLM O. Structural Mechanics: modeling and analysis of frames and trusses: 1ed. United Kingdom: Wiley, 2016.
11. Paraski, Natalie Von. Análise Estática Não Linear de Pórticos Planos via MATLAB. 2012. 150. Dissertação de Pós-Graduação – Universidade Federal Fluminense, Volta Redonda, 2012.
12. TIMOSHENKO S. P. Resistência dos Materiais: 1 ed. Rio de Janeiro: Ao Livro Técnico S. A., 1967.