

**UNIVERSIDADE FEDERAL DE PERNAMBUCO
CENTRO DE TECNOLOGIA E GEOCIÊNCIAS
DEPARTAMENTO DE ENGENHARIA CIVIL
TRABALHO DE CONCLUSÃO DE CURSO**

**UM CÓDIGO EM ELEMENTOS FINITOS PARA ANÁLISE ESTÁTICA
E DINÂMICA DE PORTICOS TRIDIMENSIONAIS**

JEAN BAPTISTE JOSEPH

Recife

2016

JEAN BAPTISTE JOSEPH

**UM CÓDIGO EM ELEMENTOS FINITOS PARA ANÁLISE ESTÁTICA
E DINÂMICA DE PORTICOS TRIDIMENSIONAIS**

Trabalho de Conclusão de Curso
apresentado à Universidade Federal de
Pernambuco para obtenção do título de
Engenheiro Civil.

Área de concentração: Engenharia de
estruturas.

Orientador: Prof. Dr. Paulo Marcelo
Vieira Ribeiro

Recife

2016

J83c	Joseph, Jean Baptiste. Um código em elementos finitos para análise estática e dinâmica de porticos tridimensionais / Jean Baptiste Joseph. - Recife: O Autor, 2016. 66 folhas, il., gráfs., tabs. Orientador: Prof. Dr. Paulo Marcelo Vieira Ribeiro. TCC (Graduação) – Universidade Federal de Pernambuco. CTG. Departamento de Engenharia Civil, 2016. Inclui Referências e Anexos. 1. Engenharia Civil. 2. EZTOOL. 3. Modelagem de estruturas. 4. Estruturas de barras. 5. Análise dinâmica. 6. Análise estática. 7. Análise modal. I. Ribeiro, Paulo Marcelo Vieira. (Orientador). II. Título. UFPE 624 CDD (22. ed.) BCTG/2016-36
------	---



SERVIÇO PÚBLICO FEDERAL

UNIVERSIDADE FEDERAL DE PERNAMBUCO

CENTRO DE TECNOLOGIA E GEOCIÊNCIAS

ESCOLA DE ENGENHARIA DE PERNAMBUCO

COORDENAÇÃO DO CURSO DE GRADUAÇÃO EM ENGENHARIA CIVIL

ATA DA DEFESA DO TRABALHO DE CONCLUSÃO DE CURSO PARA CONCESSÃO DO GRAU DE ENGENHEIRO CIVIL

CANDIDATO(S): 1 –

BANCA EXAMINADORA:

Orientador: Paulo Marcelo Ribeiro Vieira

Examinador 1:

Examinador 2:

TÍTULO DO TRABALHO DE CONCLUSÃO DE CURSO:

Um código em Elementos Finitos para Análise Estática e Dinâmica de Pórticos tridimensionais

LOCAL: CTG/UFPE; BLOCO A; 1º ANDAR; SALA 109. DATA: 03/02/2016

03/02/2016. HORÁRIO DE INÍCIO: 16 HORAS.

Em sessão pública, após exposição de cerca de 60 minutos, o candidato foi argüido oralmente pelos membros da banca, sendo considerado(s):

1)(X) aprovado(s), pois foi demonstrado suficiência de conhecimento e capacidade de sistematização no tema da monografia e o texto do trabalho aceito

(X) Sem revisões.

() Com revisões, a serem feitas e verificadas pelo orientador no prazo máximo de 30 dias. (o verso da folha da ata poderá ser utilizado para pontuar revisões).

2) (..) reprovado(s).

Na forma regulamentar foi lavrada a presente ata que é

assinada pelos membros da banca e pelo(s) candidato(s).

Orientador:

Examinador 1.....

Examinador 2.....

Candidato 1

Coordenação do Curso de Engenharia Civil-Dcivil

Rua Acadêmico Hélio Ramos s/nº. Cidade Universitária. Recife-PE CEP: 50740-530.

Fones: (081)2126.8220/8221 Fone/fax: (081)2126.8219.

DEDICATÓRIA

Primeiro à Deus, que nos criou. Seu folego de vida em mim me foi sustento e me deu coragem para questionar realidades e propor sempre um novo mundo de possibilidades. Também dedico esse trabalho a minha mãe, Gislaine Solon, meu Pai Renold Joseph, minha esposa Iracema Calixtoe minha filha Esther Marina Joseph.

AGRADECIMENTOS

Aproveito esse trabalho de conclusão de curso para agradecer a todo mundo que participou diretamente desta caminhada e meacompanharam durante os meus seis anos de aventura no Brasil, desde o primeiro dia da aprendizagem da língua portuguesa até a redação deste trabalho.

Aos meus colegas da turma 2011.1 que me acolheram muito bem tanto dentro da sala de aula como fora.

Aos meus orientadores, que me passaram experiências e conhecimentos que foram de total importância durante o curso. Quero destacar os professores Ivaldo Dario, com quem realizei meu primeiro trabalho de iniciação científica e, principalmente, o professor Paulo Marcelo, que falarei mais abaixo.

À INURED, Interuniversity Institute for Research and Development, que teve uma participação significativa durante o curso. Foi uma das instituições que me ajudou financeiramente no Brasil. Destaco o nome do professor Louis HERNS Marcelin que é o presidente da mesma.

À Universidade Federal de Pernambuco, Instituição na qual fui aceito para realizar o curso de engenharia Civil.

Ao meu grande amigo e conterrâneo William Obel, com quem vim para realizar esse grande desafio. Ele me ajudou bastante nos primeiros dias no Brasil principalmente na questão de língua portuguesa.

O prof. Paulo Marcelo merece um agradecimento especial, tendo em vista que foi o grande pilar na minha formação acadêmica. Sem ele não seria possível este trabalho, por sua generosidade em colocar sua sala à minha disposição, gastar milhares de horas comigo tirando dúvidas sobre este trabalho e foi com ele que tive a oportunidade de publicar no congresso CILAMCE do ano 2014. Eu e a minha família lhe agradecemos profundamente.

Ao grupo SAE (Software de Análise Estrutura), grupo formado pelos alunos de pesquisa do Professor Paulo Marcelo: Lucilo Victor Magalhães, Sajan Prya, Bernardo Cavalho e George Aquino. Quero destacar o nome do aluno Lucíolo Victor com quem aprendi as técnicas de programação e que me serviu de catalizador para avançar nos meus estudos.

Ao prof. Claudio Mota, esse me deu a oportunidade de colocar em prática os conceitos aprendidos na universidade, também agradeço aos colegas de trabalho Pedro, Sajan, e Lucíolo que sempre estiveram dispostos a me ajudar nos projetos.

Agradeço a minha irmã Juliette Joseph, aos meus irmãos Benedik Joseph e Davidson Joseph pelo carinho. À minha mãe e ao meu pai pelos ensinamentos a mim passados, e pela renúncia em aceitarem que eu viva longe deles para realizar o meu sonho. Agradeço a minha tia Amonita Joseph que sempre foi uma segunda mãe na minha vida. E a todos da minha família pela torcida.

O agradecimento especial vai para minha esposa Mel e a minha filha Esther, por me aceitarem em suas vidas. São duas pessoas que me ajudaram dia e noite para realizar o meu sonho. Sem Mel, a minha esposa, eu não conseguiria terminar esse curso, consequentemente não realizaria esse sonho. Ela sempre se sacrificou para cuidar de mim e da minha pequena, a princesa Esther Marina. Amo-as com todo o meu ser.

RESUMO

A evolução da tecnologia traz aos engenheiros perspectivas para a realização de projetos cada vez mais complexos e sujeitos a análises detalhadas das variáveis envolvidas. Alguns exemplos surgem como: construção de plataformas "offshore", projetos aeroespaciais e nucleares, nos quais a segurança é vital. Em casos específicos da engenharia civil, problemas especiais ocorrem com a construção de pontes, barragens e de edifícios altos em locais que possuem ações dinâmicas de natureza considerável, tais como ventos, sismos, impactos e explosões.

Todos esses projetos exigem uma avaliação de carregamento de natureza variável tanto em termos de espaço como de tempo. Para realizar esses projetos o engenheiro precisa de ferramentas ou de um modelo para simulação do comportamento físico do problema.

Considerando os aspectos citados acima esse trabalho consiste em um código computacional baseado no Método dos Elementos Finitos, denominado EZTOOL, com interface gráfica, capaz de realizar análises estática, modal e dinâmica de estruturas reticuladas em três dimensões. As rotinas foram implementadas em MATLAB e OCTAVE, com o pré-processamento realizado tanto no seu próprio interface quanto no GID desenvolvido na Universidade Politécnica da Catalunha. O código tem uma versão que pode ser utilizado para plataforma Móvel e se chama EZMOBI.

Com o código EZMOBI, o usuário é capaz de realizar Análises Estática, modal e Dinâmica de estruturas reticulas com extrema facilidade, utilizando o recurso in prompt do MATLAB. Esse funciona simplesmente como uma direta interação com usuário, com simples pergunta e resposta.

Palavras chaves: EZTOOL, Modelagem de Estruturas, Estruturas de Barras, Análise Dinâmica, Análise Estática, Análise modal.

ABSTRACT

The evolution of technology brings to prospects engineers to carry out projects increasingly complex and subject to detailed analysis of the variables involved. Some examples arise as building platforms "offshore", aerospace and nuclear projects where security is vital. In specific cases of civil engineering, special problems occur with the construction of bridges, dams and tall buildings in cities with dynamic actions of considerable nature such as winds, earthquakes, impacts and explosions.

All of these designs require a variable nature charging evaluation in terms of both space and time. To achieve these designs the engineer need tools or a model for simulating the physical behavior problem.

Considering the aspects above cited this work consists of a computer code based on the Finite Element Method, named EZTOOL, with graphic interface, capable of performing static analysis, modal and dynamic frame structures in three dimensions. The routines were implemented in MATLAB and OCTAVE, with the preprocessing performed both on its own interface as the GID developed at the Polytechnic University of Catalonia. The code has a version which can be used for mobile platform and is called EZMOBI.

With EZMOBI code, the user is able to perform Static Analysis, modal and reticles structures dynamics with ease, using the feature in MATLAB prompt. This works just like a direct iteration with user with simple question and answer.

Keywords: EZTOOL, modeling a structure, frame structures,dynamicanalysis,static analysis,Modal analysis.

LISTA DE FIGURAS

Figura 1. Viga bi-apoiada (CLOUGH & PENZIEN, 1995).....	22
Figura 2. Primeiro modo da viga (CLOUGH & PENZIEN, 1995).....	22
Figura 3. Segundo modo da viga (CLOUGH & PENZIEN, 1995).....	22
Figura 4. Terceiro modo da viga (CLOUGH & PENZIEN, 1995).....	22
Figura 5. Superposição modal (CLOUGH & PENZIEN, 1995).....	24
Figura 6. Composição dos efeitos e os graus de liberdade na barra de portico espacial. (Paulo, 2010).....	24
Figura 7. Matrizes de rigidez de todos os graus de liberdade existem no Pórtico espacial	25
Figura 8. Matriz do Pórtico Espacial/Modelo Euler-Bernouilli	25
Figura 9. Matriz do Pórtico Espacial/Modelo Timoshenko	27
Figura 10. Esquema da massa concentrada nos nós de uma viga (CLOUGH & PENZIEN, 1995).....	28
Figura 11. Matriz de massa concentrada (CLOUGH & PENZIEN, 1995).....	28
Figura 12. está descrita a composição da matriz de massa local para a barra de pórtico espacial	29
Figura 13. Composição da matriz de massa local para a barra de portico tridimensional (Paulo, 2010)	31
Figura 14. Fluxograma das principais funções.....	32
Figura 15. Fluxograma das sub-funções do Eztool.	32
Figura 16 (a). Janela principal do EZTOOL	33
Figura 16 (b). Fluxograma do software EZTOOL	33
Figura 17. Janela principal do aplicativo software EZMOBI	34
Figura 18. Fluxograma do software EMOBI.....	35
Figura 19. Resumo do aplicativo EZMOBI	35
Figura 20. Pórtico Plano com barras inclinadas e retas.....	36
Figura 21. Pórtico 3D SAP2000 (Esquerda) EZTOOL (Direita).....	37
Figura 22. Primeiro modo de vibração SAP200 (Esquerda) e EZTOOL(Direita).....	41
Figura 23. Segundo modo de vibração SAP200 (Esquerda) e EZTOOL(Direita).....	41
Figura 24. Terceiro modo de vibração SAP200 (Esquerda) e EZTOOL(Direita)	42
Figura 25. Quarto modo de vibração SAP200 (Esquerda) e EZTOOL(Direita).....	42
Figura 26. Quinto modo de vibração SAP200 (Esquerda) e EZTOOL(Direita).....	42
Figura 27. Sexto modo de vibração SAP200 (Esquerda) e EZTOOL(Direita).....	43
Figura 28. Pórtico 3D: análise dinâmica utilizando método de superposição modal /direção x	43
Figura 29. Deslocamento do nó avaliado na figura 15 na direção x (Timoshenko)	44

Figura 30. comparação do vetor de deslocamento do pórtico espacial do Software SAP2000 e EZTOOL na direção y com 10 modos	44
Figura 31. Edifício do CTG e seu modelo estrutural adaptado	46
Figura 32. Modo 1 da estrutura (EZTOOL)	47
Figura 33. Modo 2 da estrutura (EZTOOL)	47
Figura 34. Modo 3 da estrutura (EZTOOL)	48
Figura 35. Modo 4 da estrutura (EZTOOL)	48
Figura 36. Modo 5 da estrutura (EZTOOL)	48
Figura 37. Modo 6 da estrutura (EZTOOL)	49
Figura 38. Modo 7 da estrutura (EZTOOL)	49
Figura 39. Modo 8 da estrutura (EZTOOL)	49
Figura 40. Modo 9 da estrutura (EZTOOL)	50
Figura 41. Modo 10 da estrutura (EZTOOL)	50
Figura 42. Edifício, destaque do nó analisado para obtenção das respostas dinâmicas.	50
Figura 43. SAP2000 e o EZTOOL (Método de superposição modal com 50 passos) (Timoshenko) com 10 primeiros modos de vibração	51

LISTA DE TABELAS

Tabela 1. Valores de Rigidez e de Massa dos casos citados	21
Tabela 2. Propriedades mecânicas e geométricas do exemplo da figura 20	36
Tabela 3. Deslocamentos monitorados nos Pórticos Bidimensional do SAP2000-FTOOL-EZTOOL	36
Tabela 4. Propriedades mecânicas e geométricas do exemplo da figura (21).....	38
Tabela 5. Deslocamentos (Euler – Bernoulli) monitorados nos Pórticos 3D do SAP2000-FTOOL-EZTOOL	38
Tabela 6. Deslocamentos (TIMOSHENKO) monitorados nos Pórticos 3D do SAP2000-FTOOL-EZTOOL	38
Tabela 7. Propriedades mecânicas e geométricas do exemplo da figura 4	39
Tabela 8. Seis primeiras frequências obtidas/ SAP2000 com dez elementos por barra, EZTOOL com elementos por barras (Timoshenko)	39
Tabela 9. Seis primeiras frequências obtidas/SAP2000 e EZTOOL com 1 elemento por barra	39
Tabela 10. Cinco primeiras frequências obtidas/ANSYS EZTOOL com 1 elemento por barra	39
Tabela 11. Comparação entre os resultados das seis primeiras frequências obtidas no programa do SAP2000 e EZTOOL com 3 elementos por barra (Timoshenko).....	40
Tabela 12. Seis primeiras frequências obtidas/ANSYS e EZTOOL com 3 elementos por barra (Timoshenko)	40
Tabela 13. Numeração equivalente dos nós do Software SAP2000 EZTOOL.....	45
Tabela 14 Propriedades mecânicas e geométricas do exemplo da figura (31).....	45
Tabela 15 Deslocamentos monitorados em alguns nós da Estrutura SAP2000-EZTOOL (Timoshenko)	46
Tabela 16 Reações de apoios monitoradas em alguns nós da Estrutura SAP2000-EZTOOL(Timoshenko).....	46
Tabela 17 Comparação entre os resultados das seis primeiras frequências obtidas no programa do SAP2000 com 1 elemento por barra, EZTOOL com elementos por barras(Timoshenko)	47

SUMÁRIO

INTRODUÇÃO	13
1.1 Considerações Iniciais	13
1.2 Objetivos.....	14
1.3 Metodologia.....	15
1.4 Limitações	15
1.5 Organização do trabalho	15
CAPITULO 2	16
Aspectos básicos da dinâmica das Estruturas	16
2.1 Sistema de um grau de liberdade	16
2.2 Movimento forçado e amortecido.....	17
2.3 Método de Rayleigh.....	19
2.4 Metodo de Superposição Modal	23
CAPITULO 3	24
Formulação em elementos finitos para pórtico 3D	24
3.1 Viga de Euler - Bernouilli	24
3.2 Viga de Timoshenko	27
CAPITULO 4	31
codigo computacional desenvolvido	31
4.1 O EZTOOL	31
4.2 O EZMOBI.....	<u>21</u>
CAPITULO 5	36
APLICAÇÕES E validação	36
5.1 Analise Estática	36
5.1.1 Caso 1.....	36
5.2 Análise Modal.....	38
5.2.1 Pórtico Plano do caso 1	39
5.2.2 Pórtico Espacial do caso 2.....	39
5.2.3 Modos de Vibração do caso 2	40
5.3 Analise Dinâmica	43
5.4 Caso Especial.....	44
5.4.1 Analise Estática	45
5.4.2 Análise Modal	47

5.4.3	Dinâmica	50
conclusões		51
REFERÊNCIAS		52
ANEXO A.....		53
ANEXO B.....		59

CAPÍTULO 1

INTRODUÇÃO

1.1 Considerações Iniciais

A evolução atual da tecnologia traz aos engenheiros possibilidades de realizar projetos cada vez mais complexos e sujeitos a restrições cada vez mais importantes. Estamos nos referindo aos projetos espaciais, construção de plataforma de petróleo, aeronáuticos e nucleares nos quais a segurança é vital. Podemos citar outros tipos de projeto, por exemplo: Controle da poluição térmica, acústica ou química, gestão de recursos hídricos, previsão meteorológica. Para realizar esses projetos o engenheiro precisa de ferramentas ou de modelos que o permitem simular o comportamento físico do problema.

As ciências e engenharia permitem descrever o comportamento do sistema físico por meio de métodos numéricos. Dentre as ferramentas ou modelos citados anteriormente, podemos citar o Método dos Volumes Finitos (MVF) (CEBECI et al, 2005), Método das Diferenças Finitas (MDF) (AMES, 1977), Método dos Elementos Finitos (MEF) (Zienkiewicz e Morgan, 1993). Podemos destacar o MEF que é um dos mais utilizado hoje para resolver efeticamente equações diferenciais. Ele é um método que se aplica à maioria dos problemas encontrados na prática: problema estacionário, nãoestacionário, linear, não-linear, problemas com geometria variável (1D, 2D ou 3D), também se adapta muito bem em meios heterogêneos muitas vezes encontrados pelo engenheiro.

O Método dos Elementos Finitos consiste na utilização da solução aproximada para transformar as equações diferenciais em equações algébricas. Por esse motivo, o MEF requer a utilização constante de computadores, devido à necessidade de uma quantidade significativa de operações matemáticas.

Para aplicar corretamente o método, precisamos ter o conhecimento de três grandes áreas da engenharia (Combescure, 2006)

1. Ciências da engenharia, para montar as equações diferenciais;
2. Método numérico, para montar e resolver as equações algébricas;
3. A programação e a informática, a fim de implementar a ferramenta numérica em um ambiente computacional.

Não se pode esquecer a participação do professor O.C. Zienkiewicz, que trabalhou desde 1961 no Campus de Swansea da Universidade do país de Gales, no Reino Unido. Podemos citar também os trabalhos de Argyris e Kelsey, que sistematizaram a utilização do método do Princípio de Energia na análise das estruturas.

Os métodos do Princípio da Mínima Energia Potencial Total, dos Resíduos Ponderados e o do Princípio dos Deslocamentos Virtuais são na formulação do MEF, o qual utiliza o conceito de “discretização” do contínuo e de matriz de interpolação fornecendo os deslocamentos em um ponto no interior do elemento em função de seus deslocamentos nodais. (Combescure, 2006)

Podemos citar trabalhos semelhantes realizados dentro a Universidade Federal de Pernambuco, como por exemplo do aluno Lucio Victor Magalhães e Silva (Um código em Elementos Finitos para Análise Dinâmica de problema de Elasticidade Bidimensional).

A compreensão do Método exige conhecimento em várias áreas (Cobescure, 2006):

- Compreensão do problema físico estudado e conhecimento intuitivo da natureza da solução procurada;
- Aproximação das funções desconhecidas por subdomínio e construção das funções de interpolação;
- Construção das equações do sistema estudado sob forma variacional, seja a partir de método de energia, seja a partir de equações diferenciais;
- Técnicas de organização matricial dos dados;
- Métodos numéricos de integração, da solução de sistema de equações algébricas e diferenciais, lineares ou não lineares;
- Técnicas de programação adaptadas com a linguagem e com o volume de informação importantes.

Também na engenharia com o aumento dos limites da resistência do concreto e do aço, com o desenvolvimento da tecnologia das construções, levando em considerações os problemas naturais que existem (Terremoto, vulcão etc.), com o carregamento que pode sofrer variação ao longo do tempo e espaço é importante ter e entender o comportamento das estruturas diante desses fatores precitados. Portanto a Dinâmica das Estruturas tem se tornado cada vez mais presente nos projetos de engenharia e de pesquisa.

Ao longo desse trabalho estudaremos o MEF aplicado em estruturas reticuladas em três dimensões, analisaremos o comportamento de uma estrutura diante um carregamento que varia tanto no espaço quanto no tempo. Também nesse trabalho será feito um estudo sobre o comportamento do edifício do Centro de Tecnologia e Geociência (CTG) da UFPE. Esses e tantos de outros justificam o trabalho.

1.2 Objetivos

Este trabalho consiste em desenvolver um código computacional denominado EZTOOL, com interface gráfica, com base no método dos elementos finitos, cujo objetivo é a análise de estruturas reticuladas. O EZTOOL será capaz de modelar estruturas de dimensões variadas, ou seja, vigas ou pórticos (2D ou 3D). Ele foi implementado em uma linguagem de programação MATLAB que é de alto nível para cálculos numéricos e para resolver sistema de equações. Também foi desenvolvido um protótipo do mesmo código em OCTAVE, que se chama EZMOBI, esse pode ser utilizado plataforma móvel (Celular Androide). O código computacional desenvolvido é validado através de exemplos tradicionais da literatura científica utilizando software SAP2000.

Levando-se em conta o caráter acadêmico do trabalho, tem-se como objetivo a formação do aluno envolvido em métodos numéricos para resolver problemas afins, vale também destacar a participação do grupo do SAE (Software de Analise Estrutura), formado por dois alunos da graduação (Jean Baptiste Joseph, autor do trabalho e Lucíolo Victor) sob orientação do professor Paulo Ribeiro, no desenvolvimento de um código com outros tipos de análises.

1.3 Metodologia

Para a realização desse trabalho foram utilizados alguns softwares que facilitam a validação dos resultados do código desenvolvido. Foi utilizado o programa norte-americano SAP2000 que permite calcular vigas, pórticos planos ou tridimensionais, placas, cascas, sólidos utilizando o método dos Elementos Finitos.

A parte do Pré-processamento foi feito no GiD (GiD, s.d.), desenvolvido na Universidade Politécnica da Catalunha, que se mostrou bastante versátil e permitiu vários avanços no trabalho ou na própria interface do programa EZTOOL que possui um gerador de malhas próprio, gerando informações de coordenadas e de incidência.

O Software foi desenvolvido em ambiente MATLAB (MathWorks, s.d.), uma linguagem de programação de alta potência em manipulação de matrizes que possui ferramentas próprias para o desenvolvimento de códigos para solução de problemas numéricos. Também foi utilizado o método para a obtenção das propriedades de rigidez e massa da estrutura, necessários para a solução dos problemas analisados.

Foram implementadas duas formulações para análises dinâmicas de uma estrutura reticulada, o Método de Superposição modal (CLOUGH & PENZIEN, 1995), e o Método de Rayleigh que permite a obtenção de parâmetros de massa e rigidez em vigas para efeito de comparação.

Foram realizadas comparações entre os códigos desenvolvidos e solução encontradas na literatura, também problemas reais como, por exemplo, o edifício do Centro de Tecnologia e Geociência (CTG) da UFPE.

1.4 Limitações

O código desenvolvido apresenta limitações devido a seu caráter inicial. No seu desenvolvimento são considerados materiais isotrópicos em regime elástico linear e regidos pela teoria do contínuo.

Na formulação do código não foi considerado a laje, ou seja, o seu efeito tanto na matriz de rigidez quanto na matriz de massa não foi considerado.

Na Análise Estática, o código não possui ainda uma rotina própria que considera o peso próprio da estrutura, também não permite a obtenção dos diagramas dos esforços solicitantes internos.

A análise dinâmica é linear com superposição modal.

1.5 Organização do trabalho

O presente trabalho está dividido em seis capítulos, mais as referências bibliográficas, sendo:

No primeiro capítulo está a introdução, na qual estão as considerações iniciais, os objetivos do trabalho, a metodologia que foi empregada, as limitações, e esse presente tópico que é a organização do trabalho.

No capítulo 2 é descrita toda parte dos aspectos básicos da dinâmica das estruturas, abordando os sistemas de um e de n graus de liberdade, enfatizando o emprego da análise de superposição modal que é método utilizado para determinar os históricos de resposta dinâmica da estrutura.

No capítulo 3 está descrita a análise matricial para pórticos tridimensionais. Nesse capítulo foram mostradas todas matriciais, como por exemplo as matrizes de rigidez e de massa de pórticos tridimensionais. Foram apresentados o método para as vigas de Euler-Bernoulli e de Timoshenko.

No capítulo 4 é feita uma apresentação do código desenvolvido nas linguagens de Matlab e de Octave, sendo na linguagem de Matlab o programa Eztool foi desenvolvido e o Ezmobi na linguagem de Octave. Também os seus fluxogramas de funcionamento foram apresentados.

No capítulo 5 são mostrados os resultados gerados pelos aplicativos se sempre que possível, comparado com os resultados da literatura e do software SAP2000.

E finalmente o sexto capítulo trata das conclusões do trabalho e suas considerações finais do mesmo.

CAPITULO 2

ASPECTOS BÁSICOS DA DINÂMICA DAS ESTRUTURAS

A Dinâmica estrutural é um campo da mecânica das estruturas que estuda problemas variados usando métodos numéricos diferentes para a sua mobilização. Podemos citar alguns:

- O comportamento de estruturas submetido a impactos (acidente de carro, impacto de aeronave);
- O movimento causado por um terremoto ou explosões;
- O movimento de uma ponte submetida ao vento (Ponte Tacoma Narrows, EUA).

Todos esses problemas têm em comum parâmetros importantes na dinâmica de estruturas:

- O carregamento é variável no tempo;
- A força de inércia (massa da estrutura) passa a ser importante na análise.

Podemos classificar os cálculos da dinâmica de estrutura por ordem crescente de complexidade, e que são diretamente ligados aos graus de liberdade da estrutura. O caso de um sistema com um grau de liberdade é fundamental para entender o comportamento dinâmico de uma estrutura. Os conceitos básicos, tais como a ressonância, os modos de vibração, o papel do amortecimento, as excitações, a rigidez e a massa podem ser explicados considerando uma estrutura com apenas um grau de liberdade

2.1 Sistema de um grau de liberdade

O modelo de um grau de liberdade na análise dinâmica da estrutura é fundamental para entender o comportamento dinâmico de uma estrutura. As noções de base na dinâmica

das estruturas como, por exemplo, ressonância, vibração livre e forçada, a função do amortecimento, todas elas podem ser esclarecidas a partir desse modelo.

A deformada de uma estrutura submetida a uma força estática é calculada escrevendo o equilíbrio, a lei de compatibilidade, e as leis constitutivas entre as forças internas e externas:

$$Ku = f_{ext} \quad (1)$$

Sendo que f_{ext} o vetor de força estática, u o vetor de deslocamento e K a matriz de rigidez da estrutura.

A equação geral do movimento de uma estrutura em dinâmica de estrutura está representada na equação (2):

$$m \frac{d^2 u}{dt^2} + c \frac{du}{dt} + ku = F_{ext}(t) \quad (2)$$

Sendo m a massa da estrutura, c o coeficiente de amortecimento, k a rigidez da estrutura e F a excitação temporal submetida a estrutura.

Na Equação (1) acrescentam-se as forças de inércia e de amortecimento representam respectivamente por:

$$m \frac{d^2 u}{dt^2} \quad c \frac{du}{dt} \quad (3a, 3b)$$

Também na equação (2) a força depende do tempo.

Para resolver a Equação 2 deve fornecer as condições iniciais do movimento, o deslocamento inicial ou a velocidade inicial.

2.2 Movimento forçado e amortecido

Considerando a expressão geral do equilíbrio dinâmica da Equação 2, dividindo por m a equação 2, temos :

$$\frac{\partial^2 u}{\partial t^2} + \frac{c}{m} \frac{\partial u}{\partial t} + \frac{k}{m} u = \frac{P_0}{m} \sin(\omega t) \quad (4)$$

Com $F_{ext}(t) = P_0 \sin(\omega t)$

P_0 a amplitude do carregamento

ω a frequência do carregamento

A Equação (4) pode ser reescrita da seguinte maneira :

$$\frac{\partial^2 u}{\partial t^2} + 2\varepsilon\omega \frac{\partial u}{\partial t} + \omega^2 u = \frac{P_0}{m} \sin(\omega t) \quad (5)$$

Com $\varepsilon = \frac{c}{2\omega m}$

ε é a razão de amortecimento e ω a frequência natural da estrutura.

A solução homogênea da Equação(5).

$$X_h = e^{-\varepsilon\omega t} (A \cos \omega_D t + B \sin \omega_D t) \quad (6)$$

e a solução particular:

$$X_p = \frac{P_0}{k} \left(\frac{1}{(1-\beta^2)^2 + (2\varepsilon\beta)^2} \right) [(1-\beta^2) \sin \omega t - 2\varepsilon\beta \cos \omega t] \quad (7)$$

A solução da Equação 5 é a soma da solução particular (regime permanente ou forçado) e da solução homogênea das equações (6) e (7).

Podemos escrever:

$$X_t = X_p + X_h \quad (8)$$

Sendo X_p a solução particular e X_h a solução homogênea.

Ae B da solução homogênea pode ser avaliada aplicando as seguintes condições de contorno na equação 8.

$$x_t(t=0) = 0 \quad (9)$$

$$\frac{\partial x_t}{\partial t} = 0 \rightarrow t = 0 \quad (10)$$

Encontramos os valores dos constantes da equação (6).

$$A = \frac{P_0}{k} \frac{2\varepsilon\beta}{(1-\beta^2)^2 + (2\varepsilon\beta)^2} \quad (11)$$

$$B = \frac{P_0}{k} \frac{2\varepsilon^2\beta\omega - (1-\beta^2)}{(1-\beta^2)^2 + (2\varepsilon\beta)^2} \quad (12)$$

2.3 Método de Rayleigh-Ritz

Este é um método simplificado que permite calcular, de uma maneira aproximada, a frequência natural da estrutura. Além disso, ele nos permite compreender os métodos mais complexos da dinâmica da estrutura. Para um sistema de um grau de liberdade em vibração livre, o deslocamento e a velocidade podem ser escritos respectivamente da seguinte maneira (Combescure, 2006)

$$u = u_0 \cdot \sin(\omega t) \quad (13)$$

$$\frac{\delta u}{\delta t} = u_0 \cdot \omega \cos(\omega t) \quad (14)$$

Sabemos que a energia cinética é máxima quando o deslocamento é nulo, portanto:

$$E_{c, \max} = \frac{1}{2} m \cdot v_{\max}^2 \quad (15)$$

$$E_{c, \max} = \frac{1}{2} m \cdot \omega^2 \cdot u_0^2 \quad (16)$$

E a energia elástica é máxima quando a velocidade é nula, então:

$$E_{\text{elast}, \max} = \frac{1}{2} k \cdot u_{\max}^2 \quad (17)$$

$$E_{c, \max} = \frac{1}{2} k \cdot u_0^2 \quad (18)$$

Aplicando a conservação da energia, temos:

$$E_{c, \max} = E_{\text{elast}, \max} \quad (19)$$

$$\frac{1}{2} m \cdot \omega^2 \cdot u_0^2 = \frac{1}{2} k \cdot u_0^2 \quad (20)$$

Simplificando a expressão da equação 20, temos:

$$\omega^2 = \frac{k}{m} \quad (21)$$

Sendo ω a frequência natural do sistema, k a rigidez e m a massa.

Podemos generalizar utilizando separação de variáveis, considerando uma função da deformada da estrutura. O campo de deslocamento e de velocidade se escreve respectivamente:

$$U(x, t) = u_0 \cdot \Psi(x) \sin(\omega t) \quad (22)$$

$$\frac{\delta U}{\delta t}(x,t) = u_0 \cdot \omega \cdot \Psi(x) \cos(\omega t) \quad (23)$$

Aplicando as energias cinéticas e elásticas máximas, temos respectivamente:

$$E_{c, \max} = \frac{1}{2} \int_0^l m \cdot u_0 \cdot \omega^2 \Psi^2(x) dx = \frac{1}{2} u_0 \cdot \omega^2 \int_0^l m \cdot \Psi^2(x) dx \quad (24)$$

$$E_{elast, \max} = \frac{1}{2} \int_0^l EI \cdot \left(\frac{\partial^2 \Psi(x)}{\partial^2 x} \right)^2 dx = \frac{1}{2} u_0 \cdot \int_0^l EI \cdot \left(\frac{\partial^2 \Psi(x)}{\partial^2 x} \right) dx \quad (25)$$

Igualando as expressões das equações (24) e (25), temos :

$$\omega^2 = \frac{\int_0^l EI \cdot \left(\frac{\partial^2 \Psi(x)}{\partial^2 x} \right) dx}{\int_0^l m \cdot \Psi^2(x) dx} = \frac{K}{M} \quad (26)$$

Sendo E o modulo de elasticidade da estrutura, I o momento de inércia, o numerador e o denominador da equação (26) representam as matrizes de rigidez e de massa respetivamente.

Para os casos básicos da resistência dos materiais, as vigas bi apoiadas e engastada-livre, temos as seguintes funções de $\Psi(x)$ da linha elástica respectivamente (Timoshenko e Gere, 1983):

$$\Psi_1(x) = \frac{P \cdot x}{48 \cdot EI} (3L^2 - 4x^2) \quad (27)$$

$$\Psi_2(x) = \frac{P \cdot x}{6 \cdot EI} (3L - x) \quad (28)$$

Os índices 1 e 2 nas Equações (27) e (28) representam respectivamente os casos de vigas bi-apoiada e engastada e livre. Seu valor máximo é respectivamente:

$$\Psi_1(x)_{\max} = \frac{PL^3}{48EI} \quad (29)$$

$$\Psi_2(x)_{\max} = \frac{PL^3}{3EI} \quad (30)$$

A Expressão da função normalizadadas equações (29) e (30) está representada embaixo nas equações (31) e (32) respectivamente:

$$\Psi(x) = \frac{\Psi_1(x)}{\Psi_1(x)_{\max}} \quad (31)$$

$$\Psi(x) = \frac{\Psi_2(x)}{\Psi_2(x)_{\max}} \quad (32)$$

Substituindo as equações (31) e (32) na equação (26), para cada caso temos os resultados representados na tabela 1.

Tabela 1. Valores de Rigidez e de Massa dos casos citados

Casos	Rigidez (k)	Massa (m)	Frequência (w)
<i>Bi apoiada</i>	$\frac{48EI}{L^3}$	$\frac{17.m.L}{35}$	$9.94\sqrt{EI/mL^4}$
<i>Engastada-livre</i>	$\frac{3EI}{L^3}$	$\frac{33.m.L}{140}$	$3.57\sqrt{EI/mL^4}$

2.4 Análise modal

As equações de vibrações livres de um sistema de vários graus de liberdade, sem levar em conta os efeitos de amortecimento e das forças externas, podem ser escritas:

$$M.\ddot{Y}(t) + K.Y(t) = 0 \quad (33)$$

Onde M e K são respectivamente as matrizes de massa e de rigidez.

Nos sistemas de um grau de liberdade, o vetor de deslocamentos em função de tempo apresenta em cada componente um comportamento de movimento harmônico simples.

$$Y(t) = \phi.\text{sen}(\omega t) \quad (34)$$

Onde ϕ representa a forma da deformada da estrutura. Temos a segunda derivada da equação (34):

$$\ddot{Y}(t) = -\omega^2 \phi \text{sen}(\omega t) \quad (35)$$

Substituindo as equações (34) e (35) na equação (33), temos:

$$-M.\omega^2 \phi \text{sen}(\omega t) + K.\phi.\text{sen}(\omega t) = 0 \quad (36)$$

Simplificando a expressão da equação (36), obtemos:

$$[K - M\omega^2]\{\phi\} = 0 \quad (37)$$

Sendo que, para que este sistema tenha solução não trivial, deve-se atender à condição:

$$\|K - M\omega^2\| = 0 \quad (38)$$

Expandindo-se este determinante obtém-se uma equação algébrica do enésimo grau, para um sistema de n graus de liberdade, denominada equação de frequências do sistema, pois sua solução fornece as N frequências naturais (ao quadrado) do sistema, correspondentes aos N modos de vibração do sistema.

Observa-se que a equação (37) representa um problema de autovalor e autovetor, cujos autovalores são as frequências naturais ao quadrado do sistema e os autovetores correspondentes são os modos de vibração do sistema. O modo correspondente à frequência mais baixa é chamado de primeiro modo, o segundo é chamado de segundo, o terceiro é chamado de terceiro modo, etc.

Alguns modos de vibração da viga da figura (1) podem ser vistos a seguir:

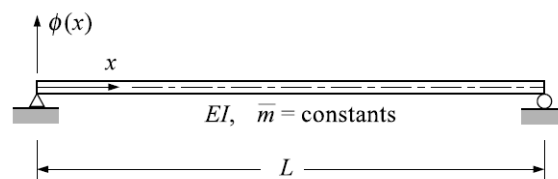


Figura 1. Viga bi-apoiada (CLOUGH & PENZIEN, 1995)



Figura 2. Primeiro modo da viga (CLOUGH & PENZIEN, 1995)

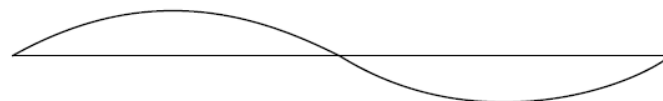


Figura 3. Segundo modo da viga (CLOUGH & PENZIEN, 1995)



Figura 4. Terceiro modo da viga (CLOUGH & PENZIEN, 1995)

2.5 Metodo de Superposição Modal

O metodo de Superposição Modal é uma das técnicas utilizado para calcular a resposta dinamica de um sistema no domínio da frequência . Esta técnica baseia-se na identificação de uma base adequada para representar a resposta do sistema . O princípio básico é que a resposta da estrutura é a combinação de diferentes modos naturais de vibração(CLOUGH & PENZIEN, 1995)

$$v = \sum_{n=1}^N \phi_n Y_n \quad (39)$$

Em notação matricial, a equação 13fica

$$v = \phi Y \quad (40)$$

Onde $[\phi]$ é a matriz dos modos de vibração natural da estrutura e $[Y]$ o vetor de deslocamento nodal, a equação do equilíbrio dinâmico do problema é definido por (CLOUGH & PENZIEN, 1995)

$$M_n \ddot{Y}_n(t) + C_n \dot{Y}_n(t) + K_n Y_n(t) = P_n(t); \quad n=1,...,N \quad (41)$$

Sendo N o numero de modos de vibração e:

$$M_n = \phi_n^T m \phi_n \quad (42)$$

$$K_n = \phi_n^T k \phi_n \quad (43)$$

$$C_n = \phi_n^T c \phi_n \quad (44)$$

$$P_n = \phi_n^T p \quad (45)$$

As Equações (42),(43),(44) e (45)são respectivamente as matrizes de massa, rigidez, amortecimento e vetor de carga desacoplado. A matriz contém os N vetores dos modos de vibração natural da estrutura.

A grande vantagem deste método é que ele não necessita a resolução completa para cada frequência discreta. A diagonalização das matrizes do sistema reduz consideravelmente o esforço numerico.

Na figura (5) está uma representação de alguns modos de vibração de uma viga engasta e livre, aplicando o metodo de superposição na determinação do campo de deslocamento da estrutura.

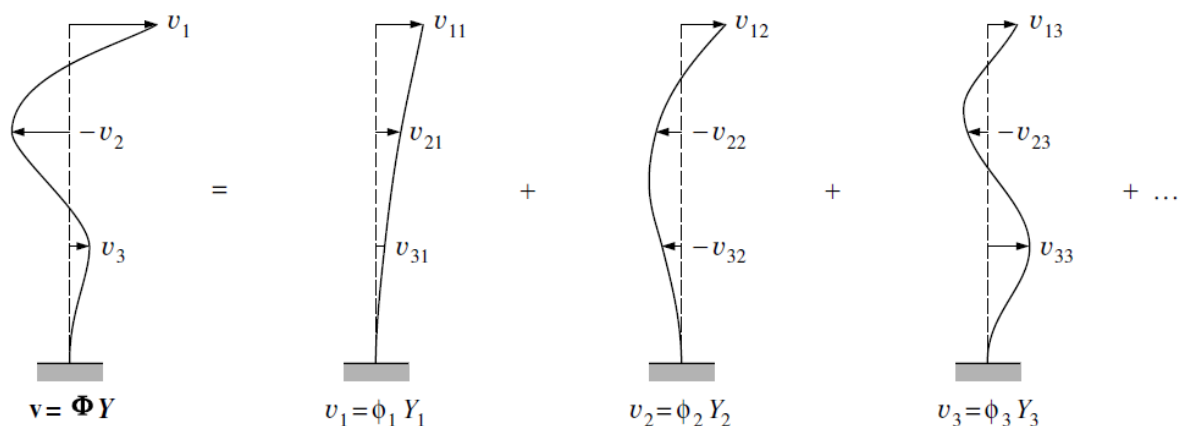


Figura 5. Superposição modal (CLOUGH & PENZIEN, 1995).

CAPITULO 3

ANÁLISE MATRICIAL PARA PÓRTICO TRIDIMENSIONAL

3.1 Viga de Euler - Bernoulli

No elemento de pórtico espacial, vários efeitos estão presentes na montagem da matriz de rigidez, temos os efeitos de flexão bidirecional, esforço axial e torção. Temos para esse elemento doze graus de liberdade. Vide Figura6(Paulo, 2010)

O problema de análise estática de estrutura consiste na resolução do sistema da equação (46):

$$[K]\{Y\} = \{f\} \quad (46)$$

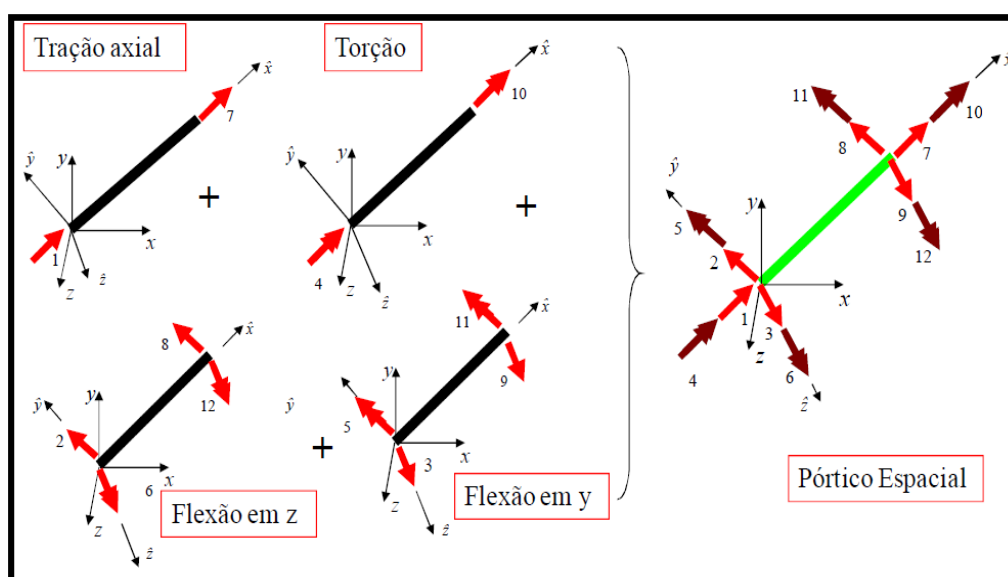


Figura 6. Composição dos efeitos e os graus de liberdade na barra de portico espacial. (Paulo, 2010)

Na figura (7) mostram as matrizes de rigidez correspondentes a cada grau de liberdade que existe no pórtico espacial.

Tração Axial	→	$[k_a] = \frac{AE}{L} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}$
Torção	→	$[k_t] = \frac{GI_x}{L} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}$
Flexão em y	→	$k_y = \frac{EI_y}{L^3} \begin{bmatrix} 12 & 6L & -12 & 6L \\ 6L & 4L^2 & -6L & 2L^2 \\ -12 & -6L & 12L & -6L \\ 6L & 2L^2 & -6L & 4L^2 \end{bmatrix}$
Flexão em z	→	$k_z = \frac{EI_z}{L^3} \begin{bmatrix} 12 & 6L & -12 & 6L \\ 6L & 4L^2 & -6L & 2L^2 \\ -12 & -6L & 12L & -6L \\ 6L & 2L^2 & -6L & 4L^2 \end{bmatrix}$

Figura 7. Matrizes de rigidez de todos os graus de liberdade existem no Pórtico espacial

A matriz de rigidez final do pórtico espacial pode ser obtida por um conveniente espalhamento considerando as contribuições de cada matriz representada na figura (8):

	1	2	3	4	5	6	7	8	9	10	11	12
K	$\frac{E \cdot A}{L}$	0	0	0	0	0	$-\frac{E \cdot A}{L}$	0	0	0	0	0
	$\frac{12 \cdot E \cdot I_z}{L^3}$	0	0	0	0	$\frac{6 \cdot E \cdot I_z}{L^2}$	0	$-\frac{12 \cdot E \cdot I_z}{L^3}$	0	0	0	$\frac{6 \cdot E \cdot I_z}{L^2}$
	$\frac{12 \cdot E \cdot I_y}{L^3}$	0	0	$-\frac{6 \cdot E \cdot I_y}{L^2}$	0	0	0	0	$-\frac{12 \cdot E \cdot I_y}{L^3}$	0	$-\frac{6 \cdot E \cdot I_y}{L^2}$	0
	$\frac{G \cdot I_x}{L}$	0	0	0	0	0	0	0	0	$-\frac{G \cdot I_x}{L}$	0	0
	$\frac{4 \cdot E \cdot I_y}{L}$	0	0	0	0	0	0	$\frac{6 \cdot E \cdot I_y}{L^2}$	0	$\frac{2 \cdot E \cdot I_y}{L}$	0	0
	$\frac{4 \cdot E \cdot I_z}{L}$	0	0	0	0	0	$-\frac{6 \cdot E \cdot I_z}{L^2}$	0	0	0	0	$\frac{2 \cdot E \cdot I_z}{L}$
	$\frac{E \cdot A}{L}$	0	0	0	0	0	0	0	0	0	0	0
	$\frac{12 \cdot E \cdot I_z}{L^3}$	0	0	0	0	0	0	0	0	0	$-\frac{6 \cdot E \cdot I_z}{L^2}$	0
	$\frac{12 \cdot E \cdot I_y}{L^3}$	0	0	0	0	0	0	0	0	$\frac{6 \cdot E \cdot I_y}{L^2}$	0	0
	$\frac{G \cdot I_x}{L}$	0	0	0	0	0	0	0	0	$\frac{G \cdot I_x}{L}$	0	0
	$\frac{4 \cdot E \cdot I_y}{L}$	0	0	0	0	0	0	0	0	$\frac{4 \cdot E \cdot I_y}{L}$	0	0
	$\frac{4 \cdot E \cdot I_z}{L}$	0	0	0	0	0	0	0	0	0	$\frac{4 \cdot E \cdot I_z}{L}$	0

simétrica

Figura 8. Matriz do Pórtico Espacial/Modelo Euler-Bernouilli

Para resolver a Equação (1) apresentada anteriormente, é necessário montar o vetor de força nodal equivalente.

Força Axial

$$\begin{Bmatrix} f_1 \\ f_7 \end{Bmatrix} = \begin{Bmatrix} \frac{pL}{2} \\ \frac{pL}{2} \end{Bmatrix} \quad (47)$$

Torção

$$\begin{Bmatrix} f_4 \\ f_{10} \end{Bmatrix} = \begin{Bmatrix} \frac{tL}{2} \\ \frac{tL}{2} \end{Bmatrix} \quad (48)$$

As equações (47) e (48) representam respectivamente as parcelas de tração axial e Torção axial.

Flexão em y

$$\begin{Bmatrix} f_2 \\ f_6 \\ f_8 \\ f_{12} \end{Bmatrix} = \begin{Bmatrix} \frac{-pL}{2} \\ \frac{-pL^2}{12} \\ \frac{-pL}{2} \\ \frac{pL^2}{12} \end{Bmatrix} \quad (49)$$

Flexão em z

$$\begin{Bmatrix} f_3 \\ f_5 \\ f_9 \\ f_{11} \end{Bmatrix} = \begin{Bmatrix} \frac{-pL}{2} \\ \frac{-pL^2}{12} \\ \frac{-pL}{2} \\ \frac{pL^2}{12} \end{Bmatrix} \quad (50)$$

Sendo as Equações (49) e (50) representam respectivamente as parcelas de Flexão em y e Flexão em z.

3.2 Viga de Timoshenko

No caso do modelo de flexão de Timoshenko onde a deformação por cortante é incorporada, a montagem da matriz de rigidez final do elemento de pórtico espacial pode também ser obtida da seguinte maneira (Paulo, 2010)

$$[K] = \begin{bmatrix} \frac{EA}{L} & 0 & 0 & 0 & 0 & 0 & -\frac{EA}{L} & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{12m_1}{L^2} & 0 & 0 & 0 & \frac{6m_1}{L} & 0 & -\frac{12m_1}{L^2} & 0 & 0 & 0 & \frac{6m_1}{L} \\ 0 & 0 & \frac{12m_2}{L^2} & 0 & -\frac{6m_2}{L} & 0 & 0 & 0 & -\frac{12m_2}{L^2} & 0 & -\frac{6m_2}{L} & 0 \\ 0 & 0 & 0 & \frac{GI_P}{L} & 0 & 0 & 0 & 0 & 0 & -\frac{GI_P}{L} & 0 & 0 \\ 0 & 0 & 0 & 0 & (4+3\gamma_y)m_2 & 0 & 0 & 0 & \frac{6m_2}{L} & 0 & (2-3\gamma_y)m_2 & 0 \\ 0 & 0 & 0 & 0 & 0 & (4+3\gamma_z)m_1 & 0 & -\frac{6m_1}{L} & 0 & 0 & 0 & (2-3\gamma_z)m_1 \\ -\frac{EA}{L} & 0 & 0 & 0 & 0 & 0 & \frac{EA}{L} & 0 & 0 & 0 & 0 & 0 \\ 0 & -\frac{12m_1}{L^2} & 0 & 0 & 0 & \frac{6m_1}{L} & 0 & \frac{12m_1}{L^2} & 0 & 0 & 0 & -\frac{6m_1}{L} \\ 0 & 0 & -\frac{12m_2}{L^2} & 0 & \frac{6m_2}{L} & 0 & 0 & 0 & \frac{12m_2}{L^2} & 0 & \frac{6m_2}{L} & 0 \\ 0 & 0 & 0 & -\frac{GI_P}{L} & 0 & 0 & 0 & 0 & 0 & \frac{GI_P}{L} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & (4+3\gamma_y)m_2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & (4+3\gamma_z)m_1 \end{bmatrix}$$

simétrica

Figura 9. Matriz do Pórtico Espacial/Modelo Timoshenko

Onde:

$$m_1 = \frac{EI_z}{L(1+3\gamma_z)} \quad (51)$$

$$m_2 = \frac{EI_y}{L(1+3\gamma_y)} \quad (52)$$

$$\gamma_z = \frac{4EI_z}{\kappa GL^2} \quad (53)$$

$$\gamma_y = \frac{4EI_y}{\kappa GL^2} \quad (54)$$

Nas equações(53) e (54) o parâmetro representa o fator de forma da seção. Para seção retangular o seu valor é $\kappa=5/6$.

A matriz obtida no modelo de Euler-Bernouilli é igual á matriz de Timoshenko quando $\gamma_z = \gamma_y = 0$.

3.3 Matriz de massa local para pórtico tridimensional

3.3.1 Matriz de massa concentrada

A procedimento mais simples para definir a matriz de massa de uma estrutura, é considerar que a massa é distribuída de acordo com a região de influencia, ou seja para os graus de liberdade de translação. O procedimento mais comum, é localizar o ponto mássico em cada nó após a divisão da barra ou elemento estrutural em vários segmentos, que serve o ponto de conexão para os segmentos. A soma das massas concentradas em cada nó dos segmentos é a massa total da estrutura. Essa matriz assume a forma de uma matriz diagonal e geralmente se chama matriz de massa concentrada, apresentada na equação (49)

Para sistema que possui apenas o efeito de grau de liberdade translacional, exemplo da figura (8),

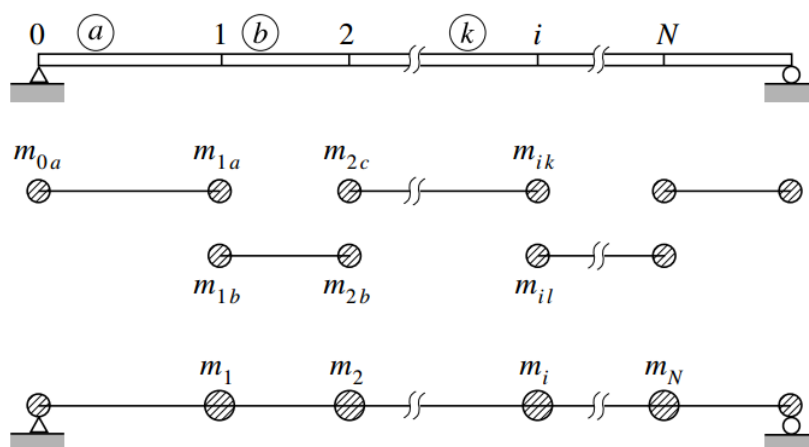


Figura 10. Esquema da massa concentrada nos nós de uma viga (CLOUGH & PENZIEN, 1995)

a matriz de massa concentrada tem a seguinte forma:

$$\mathbf{m} = \begin{bmatrix} m_1 & 0 & 0 & \cdots & 0 & \cdots & 0 \\ 0 & m_2 & 0 & \cdots & 0 & \cdots & 0 \\ 0 & 0 & m_3 & \cdots & 0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & m_i & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & 0 & \cdots & m_N \end{bmatrix}$$

Figura 11. Matriz de massa concentrada (CLOUGH & PENZIEN, 1995)

3.3.2 Matriz de massa consistente

A forma mais eficiente para apresentar a matriz de massa de uma estrutura é considerar todos os efeitos de todos os graus de liberdade dos nós do sistema. Como já foi mostrado anteriormente na determinação da matriz de rigidez para o modelo da viga de Timoshenko, na barra de pórtico espacial estão presentes os efeitos de flexão bidirecional (flexão em z e em y), tração axial e torção uniforme, resultando em doze graus de liberdade. Assim, de forma análoga, para compor a matriz de massa será utilizado o mesmo procedimento.

Somando-se todos os efeitos de tração, de flexão (x e y) e de torção, obtém-se a matriz de massa local mostrada na figura (10)

$$[M_l] = \begin{bmatrix} \frac{\rho AL}{3} & 0 & 0 & 0 & 0 & 0 & \frac{\rho AL}{6} & 0 & 0 & 0 & 0 & 0 \\ (T.j_1 + R.j_7) & 0 & 0 & 0 & (T.p_2 + R.p_8) & 0 & (T.j_3 + R.j_7) & 0 & 0 & 0 & (T.p_4 + R.p_8) \\ & (T.p_1 + R.p_7) & 0 & (T.j_2 + R.j_8) & 0 & 0 & 0 & (T.p_3 + R.p_7) & 0 & (T.j_4 + R.j_8) & 0 \\ & & \frac{\rho_p L}{3} & 0 & 0 & 0 & 0 & 0 & \frac{\rho_p L}{6} & 0 & 0 \\ & & & (T.j_2 + R.j_8) & 0 & 0 & 0 & 0 & 0 & (T.j_6 + R.j_{10}) & 0 \\ & & & & (T.p_5 + R.p_9) & 0 & -(T.j_4 + R.j_8) & 0 & 0 & 0 & (T.p_6 + R.p_{10}) \\ & & & & & 0 & 0 & 0 & 0 & 0 & 0 \\ & & & & & & (T.j_1 + R.j_7) & 0 & 0 & 0 & -(T.p_2 + R.p_8) \\ & & & & & & & (T.p_1 + R.p_7) & 0 & -(T.j_2 + R.j_8) & 0 \\ & & & & & & & & \frac{\rho_p L}{3} & 0 & 0 \\ & & & & & & & & & (T.j_5 + R.j_9) & 0 \\ & & & & & & & & & & 0 & (T.p_3 + R.p_9) \end{bmatrix}$$

Simetrica

Figura 12. A composição da matriz de massa local para a barra de pórtico espacial .

Onde,

$$p_1 = 312 + 1764\gamma_z + 2520\gamma_z^2 \quad p_2 = (44 + 231\gamma_z + 315\gamma_z^2).L \quad (55,56)$$

$$p_3 = 108 + 756\gamma_z + 1220\gamma_z^2 \quad p_4 = -(26 + 189\gamma_z + 315\gamma_z^2).L \quad (57,58)$$

$$p_5 = (8 + 42\gamma_z + 63\gamma_z^2).L^2 \quad p_6 = -3.(2 + 14\gamma_z + 21\gamma_z^2).L^2 \quad (59,60)$$

$$p_7 = 36 \quad p_8 = -3.(-1 + 15\gamma_z).L \quad (61,62)$$

$$p_9 = (4 + 15\gamma_z + 90\gamma_z^2).L^2 \quad p_{10} = (-1 - 15\gamma_z + 45\gamma_z^2).L^2 \quad (63,64)$$

As Eqs (55) até Eqs (58) são obtidos considerando os efeitos de flexão na direção z de barra não articulada no modelo de Timoshenko.

Também temos:

$$j_1 = 312 + 1764\gamma_y + 2520\gamma_y^2 \quad j_2 = -(44 + 231\gamma_y + 315\gamma_y^2).L \quad (65,66)$$

$$j_3 = 108 + 756\gamma_y + 1220\gamma_y^2 \quad j_4 = (26 + 189\gamma_y + 315\gamma_y^2).L \quad (67,68)$$

$$j_5 = (8 + 42\gamma_y + 63\gamma_y^2).L^2 \quad j_6 = -3.(2 + 14\gamma_y + 21\gamma_y^2).L^2 \quad (69,70)$$

$$j_7 = 36 \quad j_8 = 3.(-1 + 15\gamma_y).L \quad (71,72)$$

$$j_9 = (4 + 15\gamma_y + 90\gamma_y^2).L^2 \quad j_{10} = (-1 - 15\gamma_y + 45\gamma_y^2).L^2 \quad (73,74)$$

Com

$$\gamma_z = \frac{4EI_z}{\kappa GAL^2} \quad ; \quad \gamma_y = \frac{4EI_y}{\kappa GAL^2} \quad (75,76)$$

$$T = \frac{\rho AL}{840 \cdot (1 + 3\gamma_y)^2} \quad ; \quad R = \frac{\rho I_z}{30L \cdot (1 + 3\gamma_y)^2} \quad (77,78)$$

Para a obtenção da matriz de massa para o modelo de Euler-Bernouilli, o valor dos parâmetros $\gamma_y = \gamma_z = 0$

Na sequencia será apresentada o fluxograma da composição dos efeitos que compõe a matriz de massa local do pórtico tridimensional.

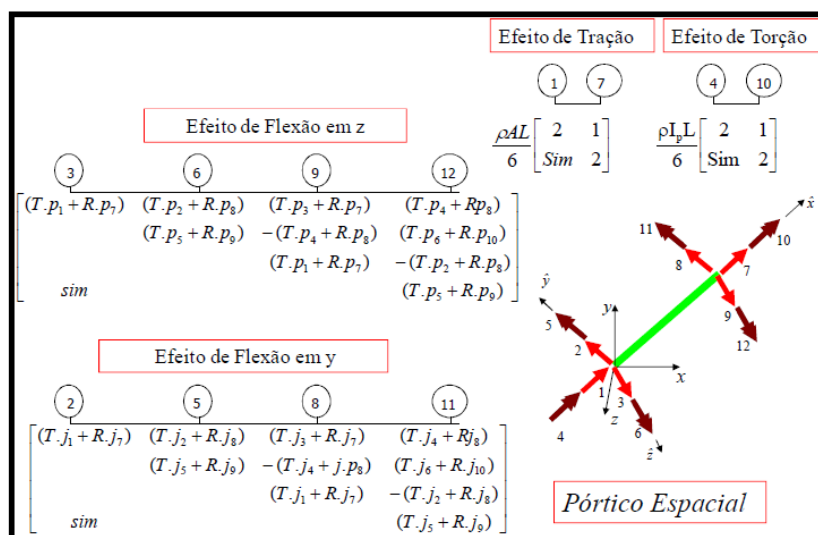


Figura 13. Composição da matriz de massa local para a barra de pórtico tridimensional (Paulo, 2010)

CAPITULO 4

CODIGO COMPUTACIONAL DESENVOLVIDO

4.1 O EZTOOL

O programa EZTOOL desenvolvido utilizando a linguagem Matlab, é um dos módulos do código computacional denominado SAE (Software de Análise Estrutural), esse está sendo desenvolvido pelos alunos Jean e Lucíolo, orientados pelo mesmo orientador deste trabalho, o professor Paulo Marcelo. Este código é um passo inicial para a construção de um software de análise de estruturas com módulos que serão implementados por alunos interessados na área. Em princípio o código é meramente acadêmico com o intuito da formação dos alunos em Análise Estrutural e Modelagem Numérica. No estágio atual foram desenvolvidos módulos de análise de vigas, pórticos planos e tridimensionais. Também foram desenvolvidos módulos bidimensionais para problema de estado plano utilizando três formulações de elementos finitos. Também foram desenvolvidas rotinas para a análise modal e dinâmica dessas estruturas além da integração desses códigos com outras plataformas.

O programa EZTOOL possibilita analisar estruturas reticuladas quaisquer, pode ser executado tanto no Matlab, o próprio ambiente que ele foi desenvolvido, quanto em Octave uma outra linguagem de programação similar ao de Matlab. O programa possui dois módulos de entrada de dados. Isso pode ser feito manualmente ou através do software GID. Um software de Pré e Pós-processamento.

Como foi dito anteriormente, foi utilizado a linguagem de programação **MATLAB (MathWorks)**. Uma linguagem de programação muito versátil, que possui uma biblioteca ampla, possibilitando as operações matriciais, adição, multiplicação, inversa etc.

Para o pré-processamento de dados, o EZTOOL disponibiliza dos modos de entrada de dados: Através do GIG, um gerador de malha muito poderoso, ou manualmente através da própria função do aplicativo. Uma grande vantagem de usar o GID são as possibilidades para os formatos dos arquivos de saída.

Para modelar uma estrutura no EZTOOL, alguns passos são primordiais:

1. Modelagem da estrutura no GiD ou no próprio Aplicativo EZTOOL
2. Caso a estrutura seja modelada noGID, salvar o arquivo de saída.
3. Salvar o arquivo com extensão ‘m’
4. Carregar o EZTOOL e Abrir o arquivo que foi salvo

Tipo de análise

1. Estática
 2. Dinâmica
5. Processamento
 6. Pós Processamento

O código é composto por sete funções principais, na figura (14) pode-se visualizar a disposição das funções.



Figura 14. Fluxograma das principais funções

Na figura (15) pode-se visualizar a disposição das sub-funções.

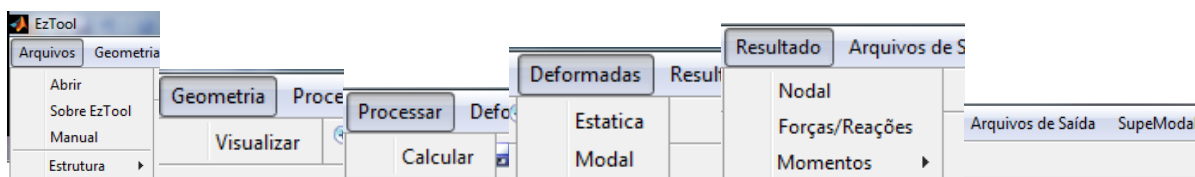


Figura 15. Fluxograma das sub-funções do Eztool.

A função *Arquivom* mostrada na figura (15) é utilizada para modelar a estrutura, pode ser na própria interface acionando a função *Estrutura* ou utilizando a sub função *abrir* para arquivos vieram do software GID.

Após a modelagem da estrutura é necessário o acionamento do botão *Processar* para calcular a estrutura e em seguida realizar as análises necessárias. Nas figuras (16(a) e 16(b)) está apresentada a janela do software EZTOOL e o seu fluxograma.

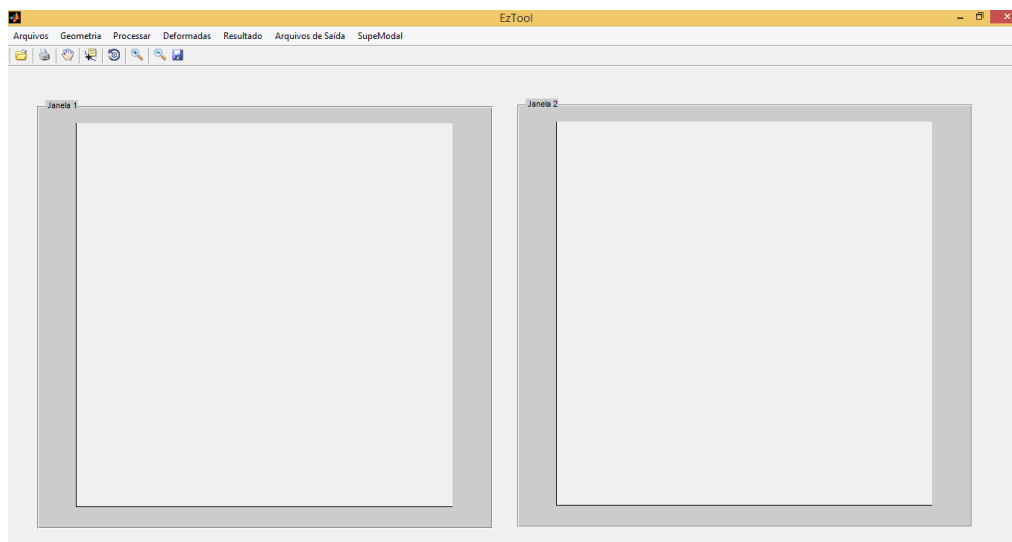


Figura 16 (a). Janela principal do EZTOOL

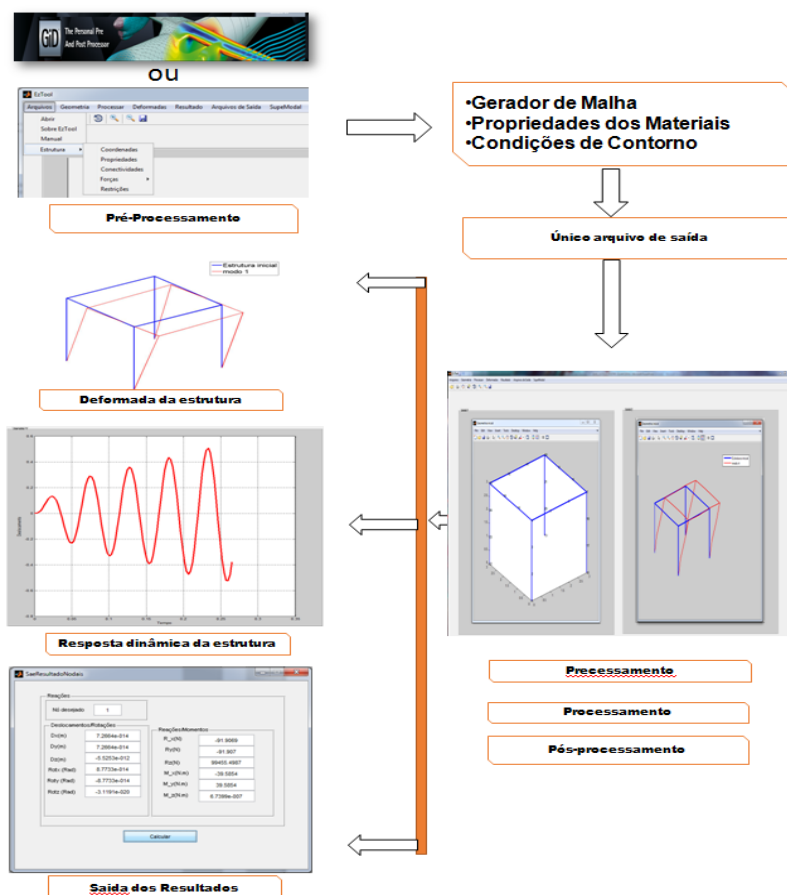


Figura 17 (b). Fluxograma do software EZTOOL

4.2 O EZMOBI



Figura 18. Janela principal do aplicativo software EZMOBI

O fluxograma da figura (18) explica os procedimentos para utilizar o aplicativo EZMOBI:

Primeiro deve baixar na loja do seu celular os arquivos Octave (Serve para sistema operacional Android/grátis), em seguida deve colocar os arquivos “EZMOBI.m”, “global_var.m” e os arquivos que pretende-se analisar (nessa versão já contém três arquivos testes que são “viga1.m”, “port2d.m” e “port3dm”, na pasta “home”. No caso de um celular Galaxy Gran Prime Duos, o caminho a ser seguido está descrito em seguida.

Computador>> Galaxy Gran Prime Duos>>Phone>>GNU Octave>>home>>EZMOBI

Após a inserção dos arquivos na pasta home, para iniciar uma análise, primeiro deve digitar preferencialmente o comando “clc” para limpar a tela e em seguida o nome do aplicativo “EZMOBI”, em maiúsculo. Na sequência o usuário seguir as instruções dadas pelo aplicativo. O usuário deve ficar atento aos avisos do próprio aplicativo tanto em inglês (aviso do **Octave**) como em português (do aplicativo **EZMOBI**).

Devido a seu caráter inicial, essa versão não se encontra nas lojas virtuais.

Pelo fato de que o aplicativo está em fase inicial, apresenta algumas limitações:

1. Foram implementados dois tipos de análise: Estática e Modal
2. Não foi considerado o efeito da laje na estrutura
3. Não existe ainda uma rotina que permite de editar valores já digitados;
4. Precisa um programa auxiliador para funcionar (Octave por exemplo)
5. Se o usuário errar um passo, precisa recomeçar o procedimento.

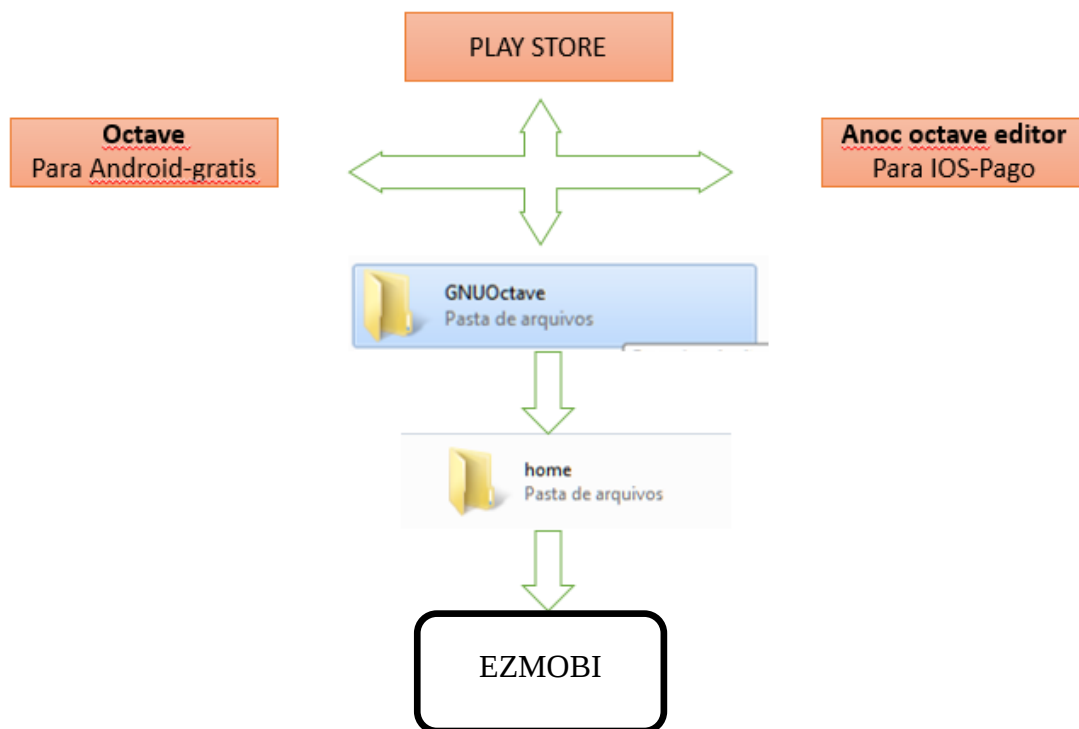


Figura 19. Fluxograma do software EMOBI

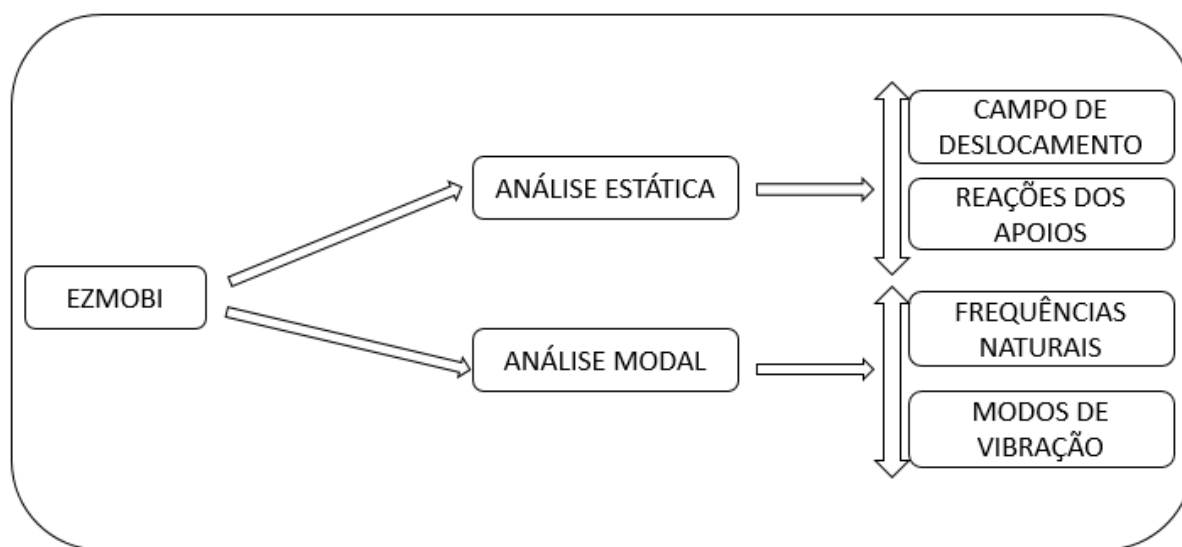


Figura 20. Resumo do aplicativo EZMOBI

CAPITULO 5

APLICAÇÕES E VALIDAÇÃO

5.1 Análise Estática

5.1.1 Caso 1 (Euler –Bernoulli)

Pórtico Plano com barras inclinadas e retas

Na figura (20) está indicado um pórtico plano cujas propriedades mecânicas e geométricas estão apresentadas na Tabela 2.

Tabela 2. Propriedades mecânicas e geométricas do exemplo da figura 20

Propriedades	E (MPa)	Poisson , ν	Área (m ²)
Mecânicas	20000	0.2	0.09

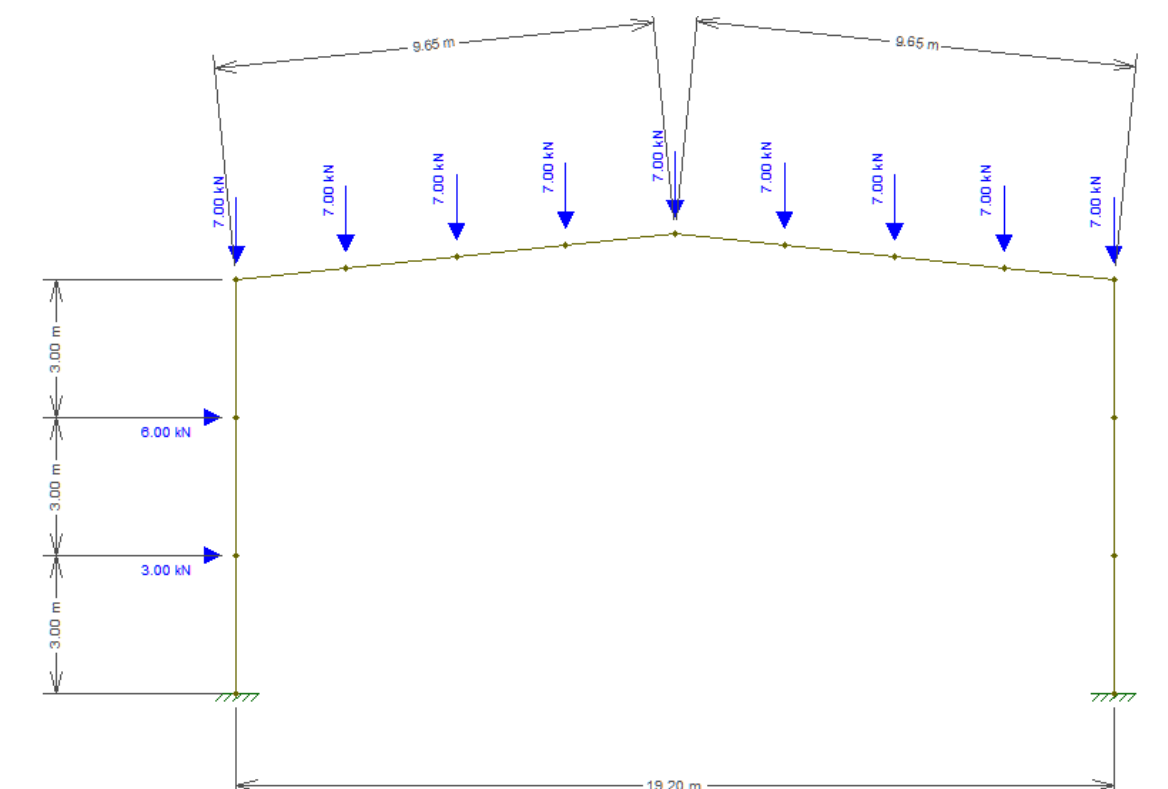


Figura 21. Pórtico Plano com barras inclinadas e retas

Tabela 3. Deslocamentos monitorados nos Pórticos Bidimensional do SAP2000-FTOOL-EZTOOL

Programa	Nó	Trans X (m)	Trans Y (m)	Trans Z (m)	Rot.X (rad)	Rot.Y (rad)	Rot.Z (rad)
SAP2000	7	0.0129	0	-0.0816	0	0.00663	0
	8	0.0137	0	-0.0891	0	-0.00047	0
	9	0.0147	0	-0.0795	0	-0.00740	0
FTOOL	7	0.0129	0	-0.0816	0	0.00663	0
	8	0.0137	0	-0.0890	0	-0.00059	0
	9	0.0147	0	-0.0795	0	-0.00740	0
EZTOOL	7	0.0129	0	-0.0816	0	0.00663	0
	8	0.0137	0	-0.0891	0	-0.00047	0
	9	0.0147	0	-0.0795	0	-0.00740	0

Pode-se observar o erro entre o SAP2000 e O EZTOOL é nulo.

5.1.2 Caso 2 (Euler –Bernoulli)

Pórtico Espacial

O pórtico espacial apresentado abaixo na figura (21) tem respectivamente 3 metros de comprimento em x, y e z. Este exemplo é útil na avaliação dos deslocamentos gerados em uma análise estática. Neste exemplo são aplicadas duas forças de 100 kN que estão aplicadas nas direções x e z respectivamente.

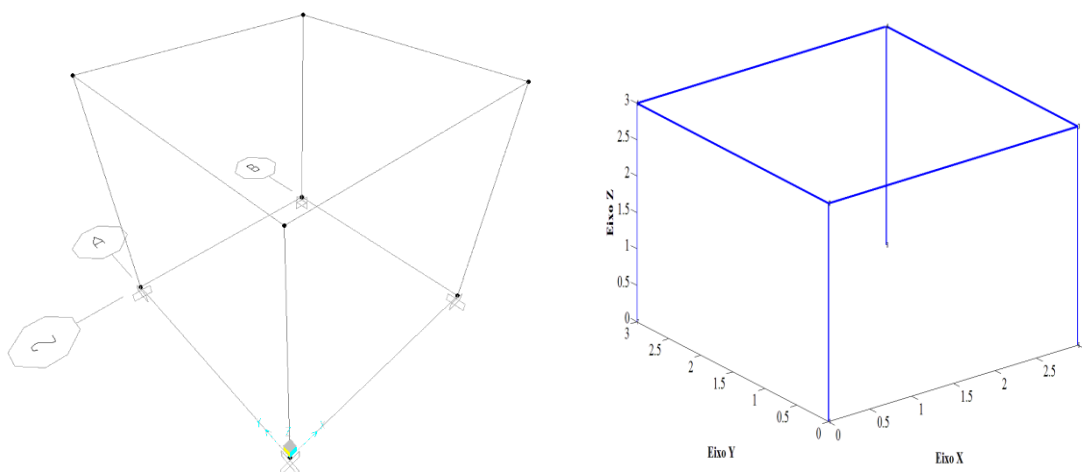


Figura 22. Pórtico 3D SAP2000 (Esquerda) EZTOOL (Direita)

A tabela 4 contém as propriedades geométricas e mecânicas da estrutura do caso 2

Tabela 4. Propriedades mecânicas e geométricas do exemplo da figura (21)

Propriedades	E (MPa)	Poisson (v)	Área (m ²)
Mecânicas	200000	0.3	0.09

A seguir os resultados obtidos na análise estática do pórtico tridimensional mencionado no caso 2 da figura (21)

Tabela 5. Deslocamentos (Euler – Bernoulli) monitorados nos Pórticos 3D do SAP2000-FTOOL-EZTOOL

Programa	Nó	Trans X (mm)	Trans Y (mm)	Trans Z (mm)	Rot.X (10 ⁻⁴ rad.)	Rot.Y (10 ⁻⁴ rad)	Rot.Z (10 ⁻⁴ rad)
SAP2000	2	0.2691	0.1679	-0.01381	-0.262	0.60	1.580
	5	0.9246	0.1679	0.00429	-0.262	1.793	-1.581
EZTOOL	2	0.2691	0.1679	-0.01381	-0.262	0.60	1.580
	5	0.9246	0.1679	0.00429	-0.262	1.793	-1.581
Erro (%)	2	0.00	0.00	0.00	0.00	0.00	0.00
Erro (%)	5	0.00	0.00	0.00	0.00	0.00	0.00

Tabela 6. Deslocamentos (TIMOSHENKO) monitorados nos Pórticos 3D do SAP2000-FTOOL-EZTOOL

Programa	Nó	Trans X (mm)	Trans Y (mm)	Trans Z (mm)	Rot.X (rad.)	Rot.Y (rad)	Rot.Z (rad)
SAP2000	2	0.2782	0.1719	-0.01382	0.00002675	0.000062	0.0001615
	5	0.9509	0.1719	0.00427	0.00002675	0.0001836	-0.0001618
EZTOOL	2	0.2782	0.1719	-0.01382	0.00002675	0.000062	0.0001615
	5	0.9509	0.1719	0.00427	0.00002674	0.0001836	-0.0001618
Erro (%)	2	0.00	0.00	0.00	0.00	0.00	0.00
Erro (%)	5	0.00	0.00	0.00	0.00	0.00	0.00

5.2 Análise Modal

Nesta seção serão feitas análises modal e dinâmica dos exemplos casos 1 e 2 da seção anterior.

5.2.1 Pórtico Plano do caso 1 (Timoshenko)

O pórtico plano apresentado anteriormente na figura(20) será empregado novamente na análise modal da estrutura. Na tabela 2 da seção anterior será acrescida a densidade do material. Vide tabela (7) abaixo

Tabela 7. Propriedades mecânicas e geométricas do exemplo da figura 4

Propriedades Mecânicas	E (MPa)	Poisson (ν)	Área (m^2)	γ_c (kg / m^3)
	25000	0.2	0.09	2500

Tabela 8. Seis primeiras frequências obtidas/ SAP2000 com dez elementos por barra, EZTOOL com elementos por barras (Timoshenko)

	W1 (Rad/s)	W2 (Rad/s)	W3 (Rad/s)	W4 (Rad/s)	W5 (Rad/s)	W6 (Rad/s)
SAP2000	4.29	7.29	8.90	12.96	17.01	34.31
EZTOOL	4.29	7.32	8.87	13.24	16.83	36.63
Erro (%)	0.00	0.41	0.34	2.11	1.06	6.33

5.2.2 Pórtico Espacial do caso 2 (Timoshenko)

Tabela 9. Seis primeiras frequências obtidas/SAP2000 e EZTOOL com 1 elemento por barra

	W1 (Rad/s)	W2 (Rad/s)	W3 (Rad/s)	W4 (Rad/s)	W5 (Rad/s)	W6 (Rad/s)
SAP2000	113.30	113.30	124.60	224.26	1378.2	1383.5
EZTOOL	118.10	119.98	146.75	265.68	505.20	605.44
Erro (%)	4.06	5.57	15.09	15.59	63.34	56.24

Tabela 10. Cinco primeiras frequências obtidas/ANSYS EZTOOL com 1 elemento por barra

	W1 (Rad/s)	W2 (Rad/s)	W3 (Rad/s)	W4 (Rad/s)	W5 (Rad/s)	W6 (Rad/s)
ANSYS	122.32	122.32	157.77	280.18	560.69	-

EZTOOL	118.10	119.98	146.75	265.68	505.20	-
Erro (%)	3.44	1.91	7.07	5.18	9.90	-

Tabela 11. Comparação entre os resultados das seis primeiras frequências obtidas no programa do SAP2000 e EZTOOL com 3 elementos por barra (Timoshenko)

	W1	W2	W3	W4	W5	W6
	(Rad/s)	(Rad/s)	(Rad/s)	(Rad/s)	(Rad/s)	(Rad/s)
<u>SAP2000</u>	120.13	120.13	149.67	263.45	498.27	580.56
EZTOOL	120.80	121.00	153.30	296.60	532.20	603.2
Erro (%)	0.55	0.72	2.37	11.17	6.38	3.75

O erro relativo apresentado na tabela 9 é muito expressivo devido a função de interpolação utilizada no SAP2000 para montar a matriz de massa, ou seja, no SAP2000 é utilizada a matriz de massa discreto, portanto no EZTOOL foi empregada matriz de massa consistente.

Tabela 12. Seis primeiras frequências obtidas/ANSYS e EZTOOL com 3 elementos por barra (Timoshenko)

	W1	W2	W3	W4	W5	W6
	(Rad/s)	(Rad/s)	(Rad/s)	(Rad/s)	(Rad/s)	(Rad/s)
<u>ANSYS</u>	120.84	120.84	154.97	265.24	501.82	580.56
EZTOOL	120.80	121.00	153.30	296.60	532.20	603.2
Erro (%)	0.033	0.132	1.08	10.57	5.71	3.75

5.2.3 Modos de Vibração do caso 2

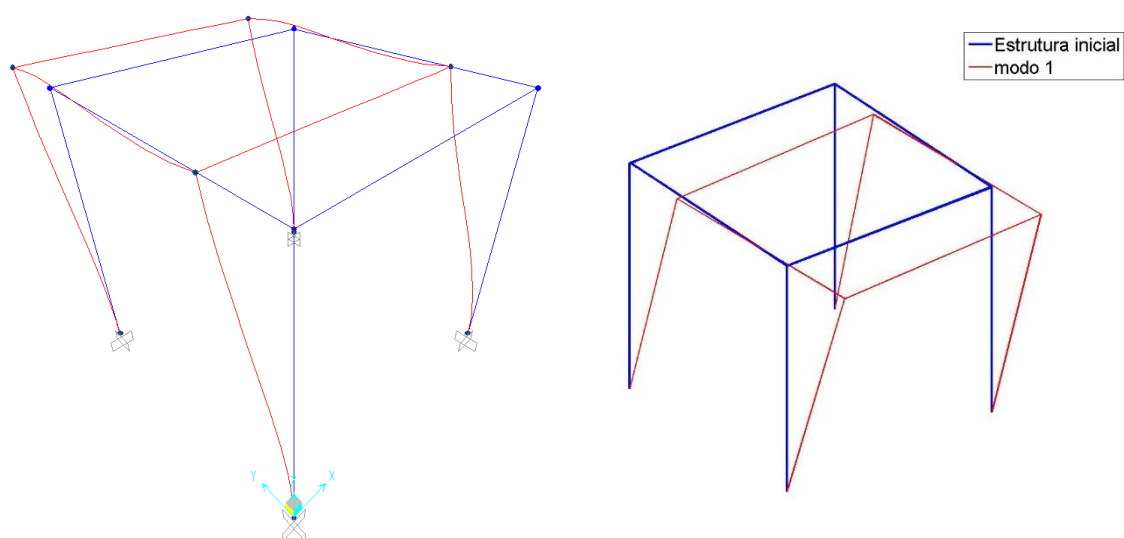


Figura 23. Primeiro modo de vibração SAP200 (Esquerda) e EZTOOL(Direita)

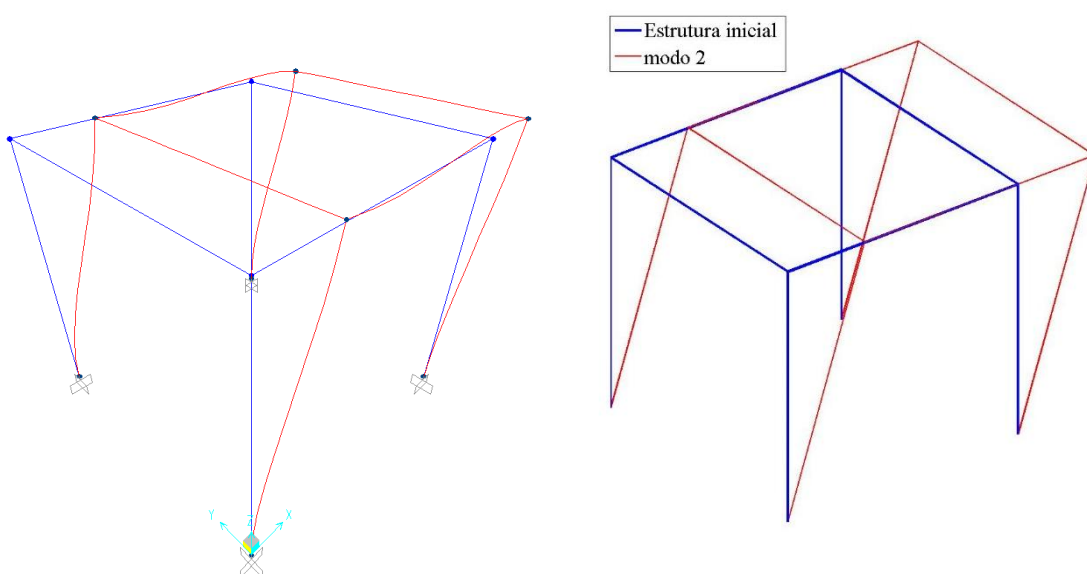


Figura 24. Segundo modo de vibração SAP200 (Esquerda) e EZTOOL(Direita)

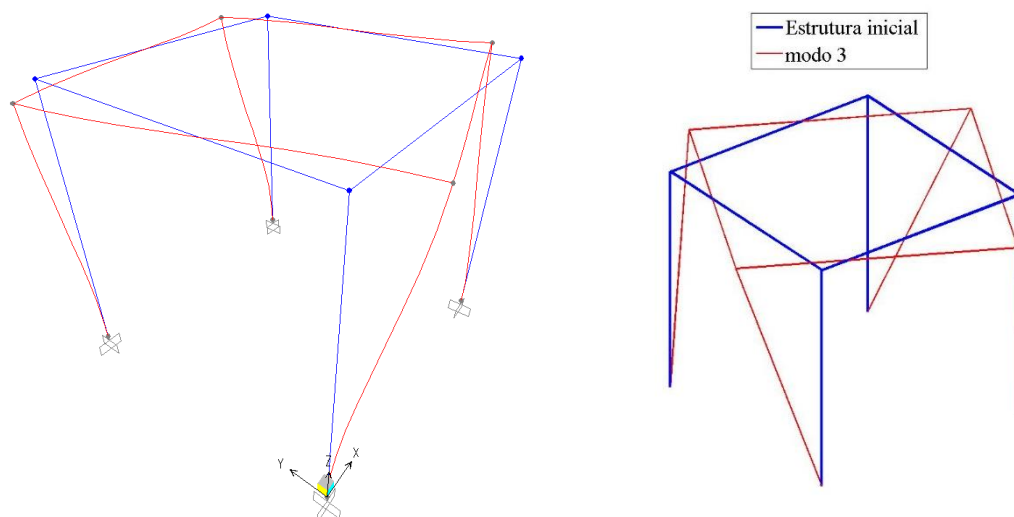


Figura 25. Terceiro modo de vibração SAP200 (Esquerda) e EZTOOL(Direita)

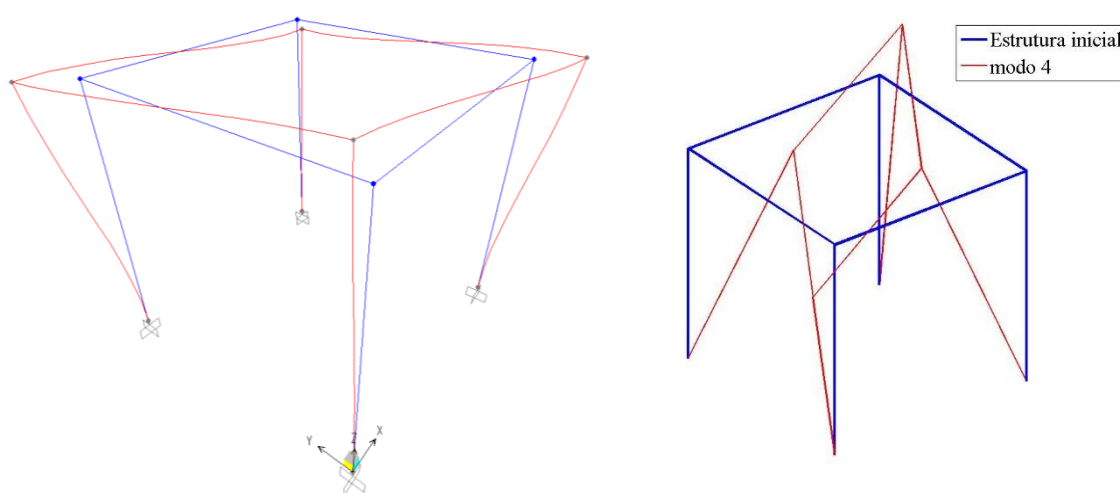


Figura 26. Quarto modo de vibração SAP200 (Esquerda) e EZTOOL(Direita)

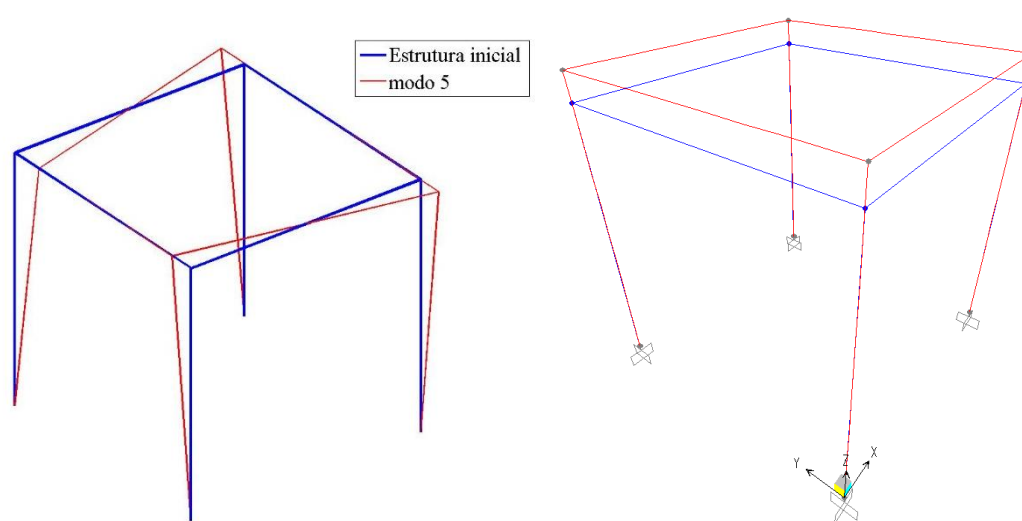


Figura 27. Quinto modo de vibração SAP200 (Esquerda) e EZTOOL(Direita)

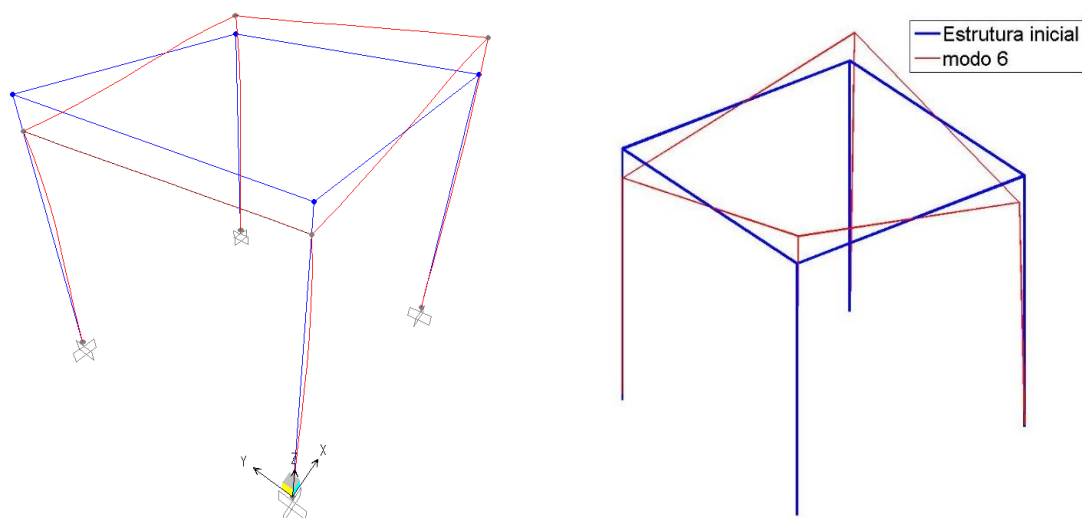


Figura 28. Sexto modo de vibração SAP2000 (Esquerda) e EZTOOL(Direita)

5.3 Análise Dinâmica

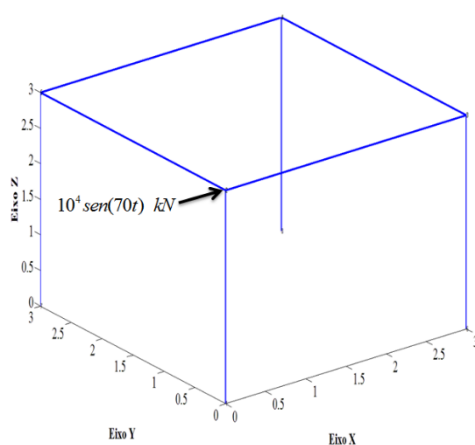


Figura 29. Pórtico 3D: análise dinâmica utilizando método de superposição modal /direção x

Foram realizadas simulações utilizando o Método de Superposição Modal e os resultados foram comparados com o software de SAP2000 e os últimos mostram uma concordância entre o código desenvolvido e o SAP2000 validando assim o código desenvolvido.

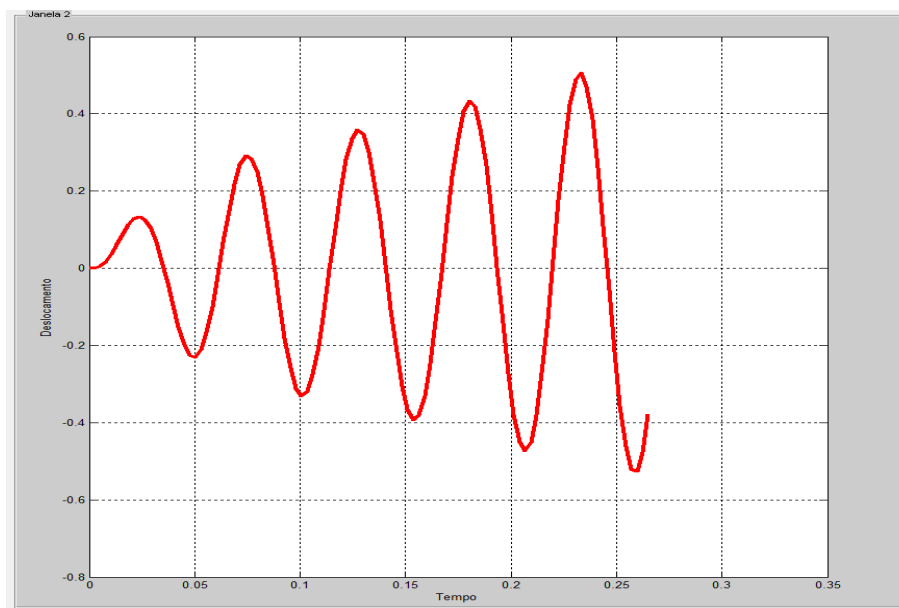


Figura 30. Deslocamento do nó avaliado na figura 15 na direção x (Timoshenko)

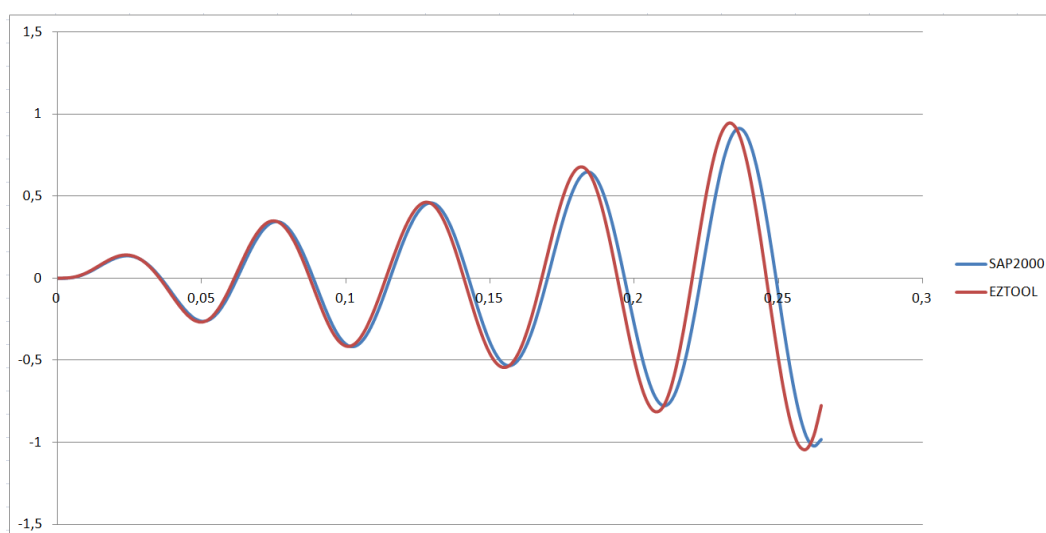


Figura 31. comparação do vetor de deslocamento do pórtico espacial do Software SAP2000 e EZTOOL na direção y com 10 modos

Para a obtenção da resposta dinâmica foi mapeado o ponto em destaque na figura (27) e para isso foi analisado os valores do deslocamento referentes ao grau de liberdade na direção x

5.4 CASO ESPECIAL

Levando em consideração os resultados apresentados nos exemplos anteriores podemos notar que as formulações apresentam resultados satisfatórios validando assim o código computacional desenvolvido para análise estática e dinâmica de estruturas. Assim para verificar o código para grande estrutura foi estudado um modelo adaptado do edifício do Centro de Tecnologia e Geociência (CTG).

Tabela 13. Numeração equivalente dos nós do Software SAP2000 EZTOOL

/Software	EZTOOL	SAP2000
Numeração equivalente dos nós	430	468
	232	465
	446	564
	1	902
	33	740
	34	661
	66	693

Modelo adaptado do edifício do Centro de Tecnologia e Geociência (CTG) da Universidade Federal de Pernambuco. Os dados que serão utilizados não são exatamente os mesmo do edifício real porque houve varias simplificações tantos nos valores das características geométricas e mecânicas que foram alvo de uma pesquisa. O edifício possui um comprimento de 128 metros e 4 metros de largura. Nesse modelo o prédio possui 66 pilares uniformemente espaçados de 4 metros tanto na direção x como y. Cada pilar recebe uma carga vertical no eixo z de 500 kN e duas cargas horizontais de 30 kN tanto na direção x como y. Foi adotado pilares e vigas de seção quadrada de 80cmx80, justamente para facilidade a modelagem da estrutura no EZTOOL.

5.4.1 Análise Estática

Tabela 14 Propriedades mecânicas e geométricas do exemplo da figura (31)

Propriedades Mecânicas	E (MPa)	Poisson (v)	Área (m ²)	$\gamma_c (kg / m^3)$
	28000	0.2	0.64	2500

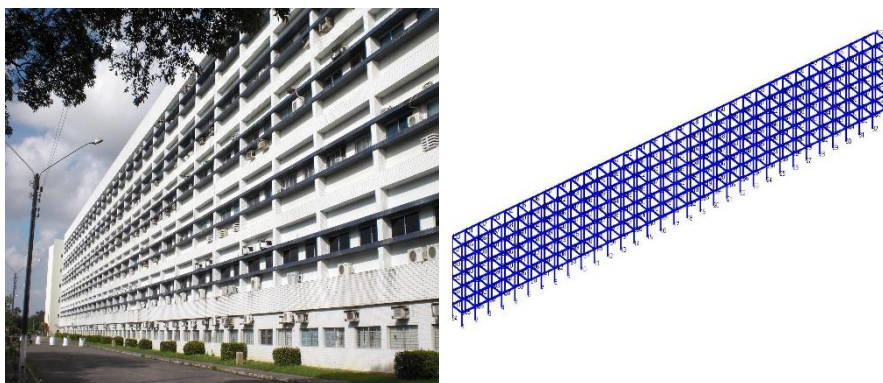


Figura 32. Edifício do CTG e seu modelo estrutural adaptado

Tabela 15 Deslocamentos monitorados em alguns nós da Estrutura SAP2000-EZTOOL (Timoshenko)

Programa	Nó	Trans X (10^{-3} m)	Trans Y (10^{-3} m)	Trans Z (10^{-4} m)	Rot.X (10^{-4} rad)	Rot.Y (10^{-5} rad)	Rot.Z (rad)
SAP2000	468	1.100	2.330	-5.703	-1.068	4.040	0
	465	0.519	0.974	-3.011	-1.179	5.133	0
	564	1.080	2.330	-6.350	-1.068	-1.961	0
EZTOOL	430	1.099	2.332	-5.703	-1.068	4.043	0
	232	0.519	0.974	-3.011	-1.180	5.137	0
	446	1.080	2.332	-6.350	-1.068	-1.963	0

Tabela 16 Reações de apoios monitoradas em alguns nós da Estrutura SAP2000-EZTOOL (Timoshenko)

Programa	Nó	R X (kN)	RY (kN)	RZ (kN)	MX (kN.m)	MY (kN.m)	MZ (kN.m)
SAP2000	661	-22.35	-30.00	616.34	69.56	-47.84	0
	902	-22.35	-30.00	145.91	69.56	-47.84	0
	693	-22.35	-30.00	854.09	69.56	-47.84	0
	740	-22.35	-30.00	383.66	69.56	-47.84	0
EZTOOL	34	-22.35	-30.00	616.34	69.56	-47.84	0
	1	-22.35	-30.00	145.89	69.56	-47.84	0
	66	-22.35	-30.00	854.11	69.56	-47.84	0
	33	-22.35	-30.00	383.66	69.56	-47.84	0

5.4.2 Análise Modal

A seguir serão apresentados os resultados das análises modais realizadas. Considerando as mesmas propriedades anteriores da estrutura. Os valores obtidos serão comparados com o SAP2000.

Tabela 17 Comparação entre os resultados das seis primeiras frequências obtidas no programa do SAP2000 com 1 elemento por barra, EZTOOL com elementos por barras (Timoshenko)

	W1 (Rad/s)	W2 (Rad/s)	W3 (Rad/s)	W4 (Rad/s)	W5 (Rad/s)	W6 (Rad/s)
SAP2000	19.80	20.15	21.20	23.94	27.31	28.38
EZTOOL	19.83	20.18	21.25	24.02	27.35	28.49
Erro (%)	0.15	0.15	0.24	0.33	0.15	0.39

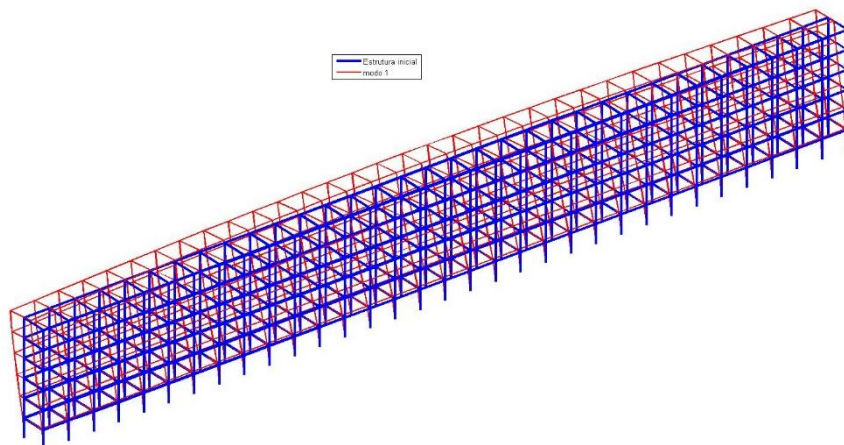


Figura 33. Modo 1 da estrutura (EZTOOL)

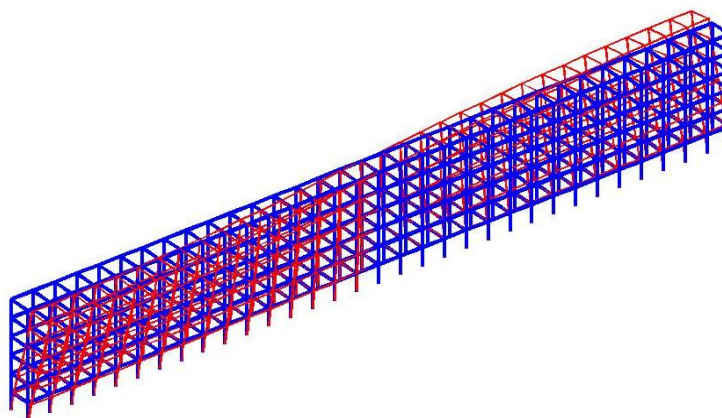


Figura 34. Modo 2 da estrutura (EZTOOL)

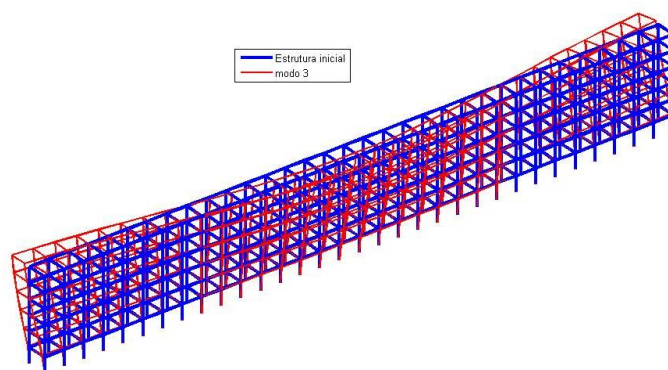


Figura 35. Modo 3 da estrutura (EZTOOL)

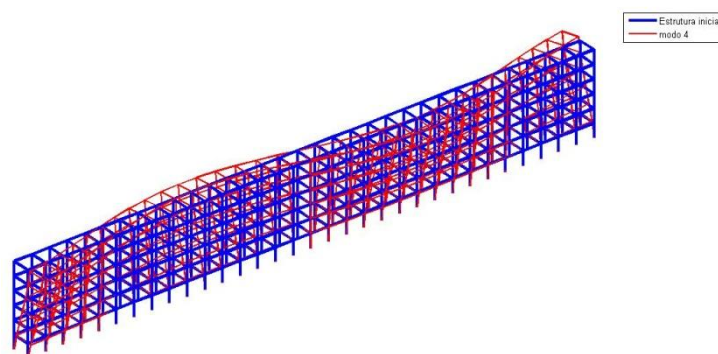


Figura 36. Modo 4 da estrutura (EZTOOL)

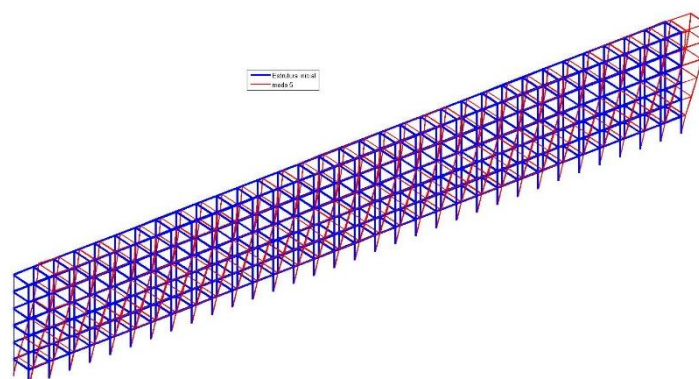


Figura 37. Modo 5 da estrutura (EZTOOL)

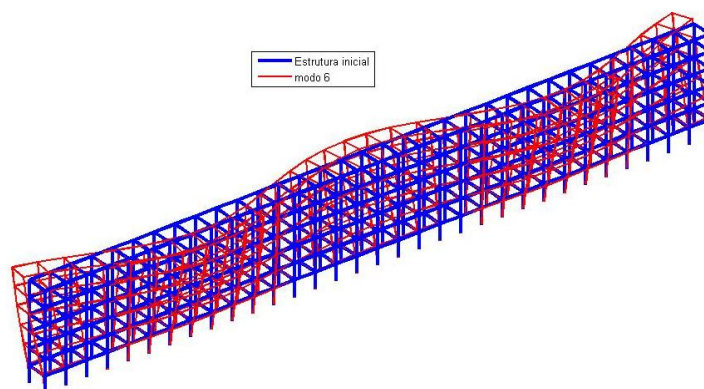


Figura 38. Modo 6 da estrutura (EZTOOL)

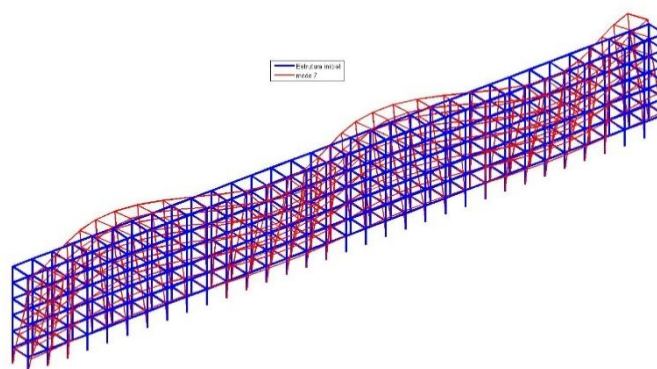


Figura 39. Modo 7 da estrutura (EZTOOL)

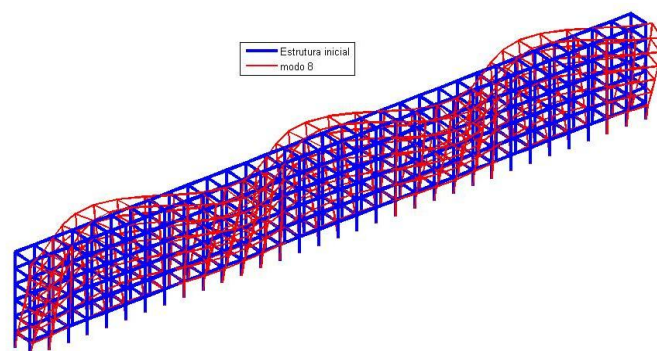


Figura 40. Modo 8 da estrutura (EZTOOL)

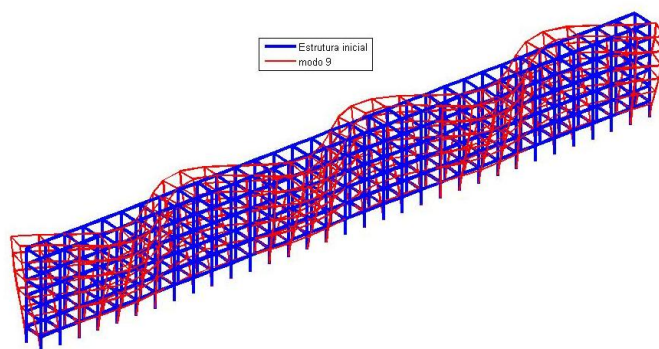


Figura 41. Modo 9 da estrutura (EZTOOL)

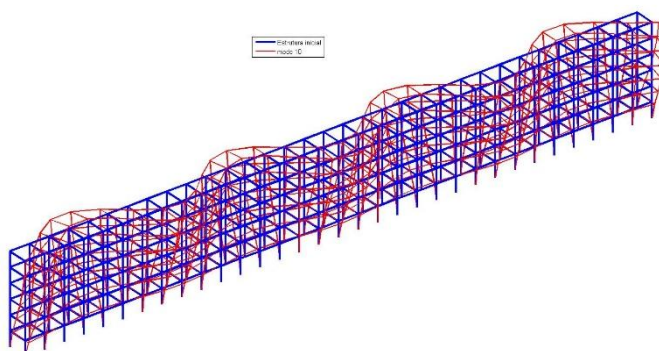


Figura 42. Modo 10 da estrutura (EZTOOL)

5.4.3 Dinâmica

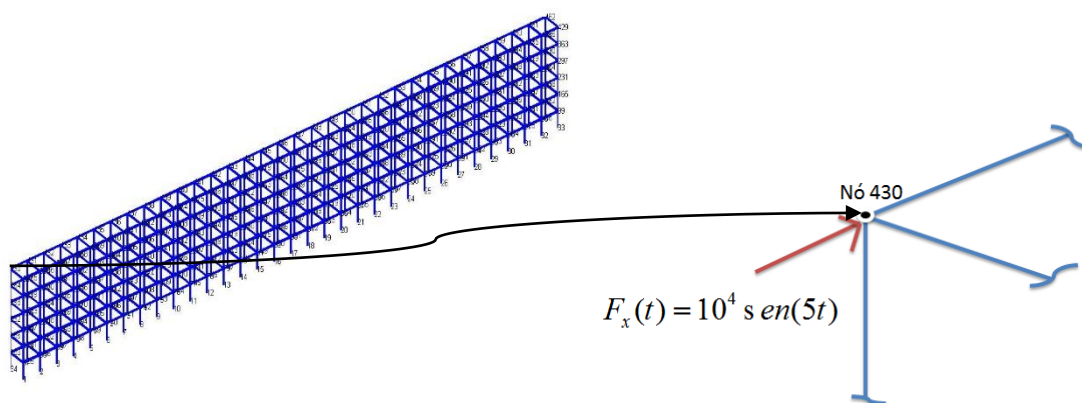


Figura 43. Edifício, destaque do nó analisado para obtenção das respostas dinâmicas.

Foram realizadas simulações do edifício da figura (42) utilizando o Método de Superposição Modal e os resultados foram comparados com o software de SAP2000. Nesse exemplo foi localizado o nó 430 da estrutura. Para esse exemplo foi utilizado uma malha com

50 passos de tempo com valor de 0.1256. Na figura (43) mostra os resultados para esse caso. Os resultados mostram uma concordância entre o código desenvolvido e o SAP2000 validando assim o código desenvolvido.

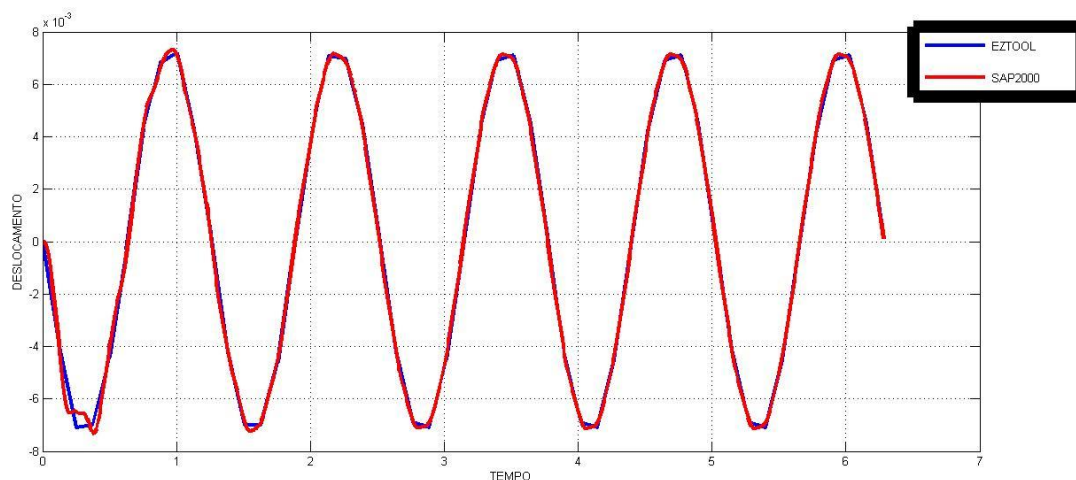


Figura 44. SAP2000 e o EZTOOL (Método de superposição modal com 50 passos) (Timoshenko) com 10 primeiros modos de vibração

CAPITULO 6

CONCLUSÕES

6.1. Considerações Finais

Este trabalho apresenta uma rápida visão geral dos vários estudos realizados e, geralmente, os métodos de cálculo utilizado no campo de estática e dinâmica de estruturas reticuladas.

Embora a solução de estruturas reticuladas com o uso de computadores já esteja amplamente difundida, neste trabalho será estudada mais adiante a interação de pórtico com o elemento de casca, que são técnicas e teorias pouco desenvolvidas em estudos para conclusão de alunos de graduação. Esse tema é mais comum em trabalho de mestrado.

O código desenvolvido neste trabalho, em elementos finitos, pode representar fenômenos mais complexos usando o método numérico de superposição modal.

6.2. Perspectivas ou recomendações para Trabalhos futuros

Para continuação desta linha de pesquisa são sugeridos , entre outros, os seguintes temas:

- Incluir o efeito das lajes na estabilidade da estrutura;
- Implemantar formulações para plotar os diagramas dos esforços internos;
- Incluir rotinas para considerar o peso proprio da estrutura
- Aprofundar para um estudo de não linearidade na formação do MEF

REFERÊNCIAS

- AMES, W. (1977). Numerical Methods for Partial Differential Equations. *Orlando: AcademicAcademic Press* .
- ASSAN, Eloísio Ernesto. Método dos elementos finitos: primeiros passos. 2ª edição, São Paulo: Unicamp, 2003.
- Bathe, J. K. (1996). Finite Element Procedures. *Upper Saddle River: Prentice Hall* .
- C. B., & J. D. (1992). *Boundary Elements: An Introductory Course*. Southampton: WIT Press.
- CEBECI, T., SHAO, J. P., KAFYERE, F., LAURENDEAU, E., 2005, Computational Fluid Dynamics for Engineers. 1 ed. Springer, Inc.
- GiD. (s.d.). Fonte: <http://www.gidhome.com/>
- http://civil.fe.up.pt/pub/apoio/ano5/de/acetatos/capitulo_7_8.pdf. (s.d.). *fe*. Acesso em 07 de 03 de 2015
- Logan, D. L. (2012). *A First Course in the Finite Element Method*. Stamford: Cengage Learning.
- MathWorks. (s.d.). *MATLAB The Language of Techical Computing*. Fonte: MathWorks Accelerating the pace of engineering of science: <http://www.mathworks.com/products/matlab/>
- Paliga, A. (s.d.). *UFPEL*. Fonte: <http://wp.ufpel.edu.br/alinepaliga/files/2013/09/Unidade-71.pdf>
- Proença, J. M. (s.d.). *SMEE - DECivil*. Acesso em 7 de 03 de 2015, disponível em ulisboa: https://fenix.tecnico.ulisboa.pt/downloadFile/3779571671152/Acetatos_Rayleigh.pdf
- Salete. (s.d.). *UFF*. Fonte: <http://www.professores.uff.br/salete/res1/aula11.pdf>
- Soriano, H. d. (Introdução à Dinâmica das Estruturas). *Introdução à Dinâmica das Estruturas*. Elsever.
- ZIENKIEWICZ, O. C.; MORGAN, K. Finite elements and approximations. Singapore: John Willey & Sons, 1993.
- Paulo, C. Queiroz. (2010). Análise estática e dinâmica de estruturas reticuladas: Ambiente de simulação em java.
- Clough, R. W. (1995). Dynamics of Structures. *McGraw-Hill*

ANEXO A

EzTool.m

O Anexo A contém o script de um código computacional desenvolvido em MATLAB que há como função de chamar a janela principal do software Eztool, nessa janela contém todos os widgets necessários para a modelagem da estrutura.

```
%disp( '%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%');
%disp( '%                UNIVERSIDADE FEDERAL DE PERNAMBUCO                %');
%disp( '%                Código de Pórtico Tridimensional EZTOOL                %');
%disp( '%                Orientador: Dr Paulo Marcelo Ribeiro                %');
%disp( '%                Colaborador:Luciolo Victor Magalhães                %');
%disp( '%                Aluno/Autor: Jean Baptiste Joseph                %');
%disp( '%                EMAIL: paulo.marcelo.1980@gmail.com                %');
%disp( '%                : luciolo_victor@hotmail.com                %');
%disp( '%                : jbluxe40@yahoo.fr                %');
%disp( '%                %');
%disp( '%                Data: 10/08/2015                %');
%disp( '%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%');

% Trabalho de Conclusão de Curso ( TCC )/ de Engenharia civil/**10/08/2015

function varargout = EzTool(varargin)
%
% Última modificação v2.5 29-May-2015 12:36:23

% Inicialização do código - Não modifique
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
'gui_Singleton',  gui_Singleton, ...
'gui_OpeningFcn', @EzTool_OpeningFcn, ...
'gui_OutputFcn',  @EzTool_OutputFcn, ...
'gui_LayoutFcn',  [] , ...
'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% Fim da inicialização

function EzTool_OpeningFcn(hObject, eventdata, handles, varargin)

handles.output = hObject;

guidata(hObject, handles);

function varargout = EzTool_OutputFcn(hObject, eventdata, handles)

varargout{1} = handles.output;

% --- Executar quando pressionar pushbutton1_DADOS.
```

```

function pushbutton1_DADOS_Callback(hObject, eventdata, handles)

SaeEntradasDados

% --- Executar quando pressionar radiobutton3_Desl_Rot.
function radiobutton3_Desl_Rot_Callback(hObject, eventdata, handles)

% --- Executar quando pressionar radiobutton4_R_Mom.
function radiobutton4_R_Mom_Callback(hObject, eventdata, handles)

% --- Executar quando pressionar radiobutton5_Ke.
function radiobutton5_Ke_Callback(hObject, eventdata, handles)

function edit29_Num_Elem_Callback(hObject, eventdata, handles)

% --- Executes during object creation, after setting all properties.
function edit29_Num_Elem_CreateFcn(hObject, eventdata, handles)

%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on button press in radiobutton7_KG.
function radiobutton7_KG_Callback(hObject, eventdata, handles)

function edit1_no_Callback(hObject, eventdata, handles)

% --- Executes during object creation, after setting all properties.
function edit1_no_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit4_dx_Callback(hObject, eventdata, handles)

% --- Executes during object creation, after setting all properties.
function edit4_dx_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit12_y_Callback(hObject, eventdata, handles)

% --- Executes during object creation, after setting all properties.
function edit12_y_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit13_z_Callback(hObject, eventdata, handles)

```

```

% --- Executes during object creation, after setting all properties.
function edit13_z_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit14_rotx_Callback(hObject, eventdata, handles)

% --- Executes during object creation, after setting all properties.
function edit14_rotx_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit15_roty_Callback(hObject, eventdata, handles)

% --- Executes during object creation, after setting all properties.
function edit15_roty_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit16_rotz_Callback(hObject, eventdata, handles)

% --- Executes during object creation, after setting all properties.
function edit16_rotz_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit23_rx_Callback(hObject, eventdata, handles)

% --- Executes during object creation, after setting all properties.
function edit23_rx_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit24_ry_Callback(hObject, eventdata, handles)

% --- Executes during object creation, after setting all properties.
function edit24_ry_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))

```

```

        set(hObject, 'BackgroundColor', 'white');
    end
    function edit25_rz_Callback(hObject, eventdata, handles)

% --- Executes during object creation, after setting all properties.
function edit25_rz_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject, 'BackgroundColor'),
get(0, 'defaultUicontrolBackgroundColor'))
    set(hObject, 'BackgroundColor', 'white');
end

function edit26_My_Callback(hObject, eventdata, handles)

% --- Executes during object creation, after setting all properties.
function edit26_My_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject, 'BackgroundColor'),
get(0, 'defaultUicontrolBackgroundColor'))
    set(hObject, 'BackgroundColor', 'white');
end

function edit27_Mz_Callback(hObject, eventdata, handles)

% --- Executes during object creation, after setting all properties.
function edit27_Mz_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject, 'BackgroundColor'),
get(0, 'defaultUicontrolBackgroundColor'))
    set(hObject, 'BackgroundColor', 'white');
end

function edit28_Mx_Callback(hObject, eventdata, handles)

% --- Executes during object creation, after setting all properties.
function edit28_Mx_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject, 'BackgroundColor'),
get(0, 'defaultUicontrolBackgroundColor'))
    set(hObject, 'BackgroundColor', 'white');
end

function edit31_Callback(hObject, eventdata, handles)

function edit31_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject, 'BackgroundColor'),
get(0, 'defaultUicontrolBackgroundColor'))
    set(hObject, 'BackgroundColor', 'white');
end

% --- Executes during object creation, after setting all properties.
function axes_number1_CreateFcn(hObject, eventdata, handles)
% hObject      handle to axes_number1 (see GCBO)
% eventdata    reserved - to be defined in a future version of MATLAB
% handles      empty - handles not created until after all CreateFcns called

% Hint: place code in OpeningFcn to populate axes_number1

```

```

% -----
function Untitled_Arqui_Callback(hObject, eventdata, handles)
% hObject      handle to Untitled_Arqui (see GCBO)
% eventdata    reserved - to be defined in a future version of MATLAB
% handles      structure with handles and user data (see GUIDATA)

% -----
function Untitled_Estruturas_Callback(hObject, eventdata, handles)
% hObject      handle to Untitled_Estruturas (see GCBO)
% eventdata    reserved - to be defined in a future version of MATLAB
% handles      structure with handles and user data (see GUIDATA)

% -----
function Untitled_Coord_Callback(hObject, eventdata, handles)
% hObject      handle to Untitled_Coord (see GCBO)
% eventdata    reserved - to be defined in a future version of MATLAB
% handles      structure with handles and user data (see GUIDATA)
%hload('estrutura.fig');
estrutura

% -----
function Untitled_Propr_Callback(hObject, eventdata, handles)
% hObject      handle to Untitled_Propr (see GCBO)
% eventdata    reserved - to be defined in a future version of MATLAB
% handles      structure with handles and user data (see GUIDATA)
Propriedades_mateirais

% -----
function Untitled_abrir_Callback(hObject, eventdata, handles)
% hObject      handle to Untitled_abrir (see GCBO)
% eventdata    reserved - to be defined in a future version of MATLAB
% handles      structure with handles and user data (see GUIDATA)
SaeEntradasDados

% -----
function Untitled_nos_Callback(hObject, eventdata, handles)
% hObject      handle to Untitled_nos (see GCBO)
% eventdata    reserved - to be defined in a future version of MATLAB
% handles      structure with handles and user data (see GUIDATA)
SaeResultadoNodais

% -----
function Untitled_fletorzz_Callback(hObject, eventdata, handles)
% hObject      handle to Untitled_fletorzz (see GCBO)
% eventdata    reserved - to be defined in a future version of MATLAB
% handles      structure with handles and user data (see GUIDATA)
SaeMomento_zz

% -----
function Untitled_defor_est_Callback(hObject, eventdata, handles)
% hObject      handle to Untitled_defor_est (see GCBO)
% eventdata    reserved - to be defined in a future version of MATLAB
% handles      structure with handles and user data (see GUIDATA)
%SaeDeformada
SaePlotModal

% -----
function Untitled_Calc_Callback(hObject, eventdata, handles)
% hObject      handle to Untitled_Calc (see GCBO)
% eventdata    reserved - to be defined in a future version of MATLAB
% handles      structure with handles and user data (see GUIDATA)

```

SaeCalcular

```
% -----
function Untitled_view_Callback(hObject, eventdata, handles)
% hObject      handle to Untitled_view (see GCBO)
% eventdata    reserved - to be defined in a future version of MATLAB
% handles      structure with handles and user data (see GUIDATA)
%SaePlotagem_inicial
SaePlotagem_inicial

% -----
function Untitled_defo_modal_Callback(hObject, eventdata, handles)
% hObject      handle to Untitled_defo_modal (see GCBO)
% eventdata    reserved - to be defined in a future version of MATLAB
% handles      structure with handles and user data (see GUIDATA)
SaePlot_modal

% -----
function Untitled_arquivos_saida_Callback(hObject, eventdata, handles)
% hObject      handle to Untitled_arquivos_saida (see GCBO)
% eventdata    reserved - to be defined in a future version of MATLAB
% handles      structure with handles and user data (see GUIDATA)

% -----
function Untitled_9_view_arq_saida_Callback(hObject, eventdata, handles)
% hObject      handle to Untitled_9_view_arq_saida (see GCBO)
% eventdata    reserved - to be defined in a future version of MATLAB
% handles      structure with handles and user data (see GUIDATA)
SaeArquivosSaida

% --- Executes on button press in pushbutton8.
function pushbutton8_Callback(hObject, eventdata, handles)
% hObject      handle to pushbutton8 (see GCBO)
% eventdata    reserved - to be defined in a future version of MATLAB
% handles      structure with handles and user data (see GUIDATA)

% -----
function Untitled_9_desl_rot_Callback(hObject, eventdata, handles)
% hObject      handle to Untitled_9_desl_rot (see GCBO)
% eventdata    reserved - to be defined in a future version of MATLAB
% handles      structure with handles and user data (see GUIDATA)

% -----
function Untitled_Arqui_ButtonDownFcn(hObject, eventdata, handles)
% hObject      handle to Untitled_Arqui (see GCBO)
% eventdata    reserved - to be defined in a future version of MATLAB
% handles      structure with handles and user data (see GUIDATA)

% --- Executes during object creation, after setting all properties.
function axes3_number2_CreateFcn(hObject, eventdata, handles)
% hObject      handle to axes3_number2 (see GCBO)
% eventdata    reserved - to be defined in a future version of MATLAB
% handles      empty - handles not created until after all CreateFcns called

% Hint: place code in OpeningFcn to populate axes3_number2

% -----
function Untitled_9_sup_modal_Callback(hObject, eventdata, handles)
% hObject      handle to Untitled_9_sup_modal (see GCBO)
```

```

% eventdata reserved - to be defined in a future version of MATLAB
% handles      structure with handles and user data (see GUIDATA)

% -----
function Untitled_9_calcular_SM_Callback(hObject, eventdata, handles)
% hObject      handle to Untitled_9_calcular_SM (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles      structure with handles and user data (see GUIDATA)
SaeSuperpo_modal

% -----
function Untitled_9_distribuida_Callback(hObject, eventdata, handles)
% hObject      handle to Untitled_9_distribuida (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles      structure with handles and user data (see GUIDATA)
Sae_Carregamento_distribuido

```

ANEXO B

SaeCalcular.m

Esse anexo é o mais importante do software, embora todos os arquivos que compõem a pasta do programa Eztool são indispensáveis para o bom funcionamento dele. Esse anexo B é responsável para calcular o Deslocamento da estrutura

```

%disp( '%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%' );
%disp( '% UNIVERSIDADE FEDERAL DE PERNAMBUCO %' );
%disp( '% Código de Pórtico Tridimensional EZTOOL %' );
%disp( '% Orientador: Dr Paulo Marcelo Ribeiro %' );
%disp( '% Aluno/Autor: Jean Baptiste Joseph %' );
%disp( '% EMAIL: paulo.marcelo.1980@gmail.com %' );
%disp( '% : luciolo_victor@hotmail.com %' );
%disp( '% : jbluxe40@yahoo.fr %' );
%disp( '% %' );
%disp( '% Data: 10/08/2015 %' );
%disp( '%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%' );

% Trabalho de Conclusão de Curso ( TCC )/ de Engenharia civil/**10/08/2015

%%
global_var      % Arquivo dos variaveis globais
%%
    TAN=2;      % 1 para viga de Bernouilli e 2 pra de Timoshenko
%%
    denss=2500; % Densidade do material Kg/m^3
%%
Nnode = size(Coord,1); % Numero Total de nós da estrutura

Nelem = size(Elem,1); % Numero de elemento da malha

Ndes = 6*Nnode; % Número total de graus de liberdade.

NDIRES = size(fixnodes,1); % Número de direções restringidas

NRES = zeros(NDIRES,1); % Matriz nula dos nós restrigidos

```

```

%%
% O for a seguir transforma as informações na matriz fixnodes na matriz
% NRES. A matriz fixnodes foi contruida de maneira que as entradas sejam
% mais naturais, está sendo modificada por ser mais fácil operar com ela
% dessa maneira.

for i=1:NDIRES

    if fixnodes(i,2)==1

        NRES(i) = fixnodes(i,1)*6-5;

    elseif fixnodes(i,2)==2;

        NRES(i) = fixnodes(i,1)*6-4;

    elseif fixnodes(i,2)==3;

        NRES(i) = fixnodes(i,1)*6-3;

    elseif fixnodes(i,2)==4 ;

        NRES(i) = fixnodes(i,1)*6-2;

    elseif fixnodes(i,2)==5;

        NRES(i) = fixnodes(i,1)*6-1;

    elseif fixnodes(i,2)==6;

        NRES(i) = fixnodes(i,1)*6;

    end

end

%%
% Inicializando o vetor de forças nodais (f)
f = zeros(Ndes,1);
% Inicializando da matriz de rigidez K
K=zeros(Ndes,Ndes);
% Inicializando da matriz de massa
M=zeros(Ndes,Ndes);
% Inicializando da matriz dAS REAÇÕES
Reacoes = sparse(Ndes,1);
% Inicializando o vetor com os comprimentos
L1 = zeros(size(Elem,1));
% Inicializando da matriz dAS REAÇÕES
Desloc = sparse(Ndes,1);
Inc(1,:) = Elem(:,2);
Inc(2,:) = Elem(:,3);
%%
% NFA
NFA = size(FA,1); % Número de forças aplicadas.

% Montagem da matriz de força.
%%

for i=1:NFA

```

```

f(FA(i,1)*6-5) = FA(i,2);

f(FA(i,1)*6-4) = FA(i,3);

f(FA(i,1)*6-3) = FA(i,4);

f(FA(i,1)*6-2) = FA(i,5);
f(FA(i,1)*6-1) = FA(i,6);

f(FA(i,1)*6)    = FA(i,7);

end

%%
% Ponteiros que relacionam os graus de liberdade do elemento com os graus
% de liberdade do elemento'
for i=1:Nelem

    dg(1,i) = 6*Inc(1,i)-5;
    dg(2,i) = 6*Inc(1,i)-4;
    dg(3,i) = 6*Inc(1,i)-3;
    dg(4,i) = 6*Inc(1,i)-2;
    dg(5,i) = 6*Inc(1,i)-1;
    dg(6,i) = 6*Inc(1,i);
    dg(7,i) = 6*Inc(2,i)-5;
    dg(8,i) = 6*Inc(2,i)-4;
    dg(9,i) = 6*Inc(2,i)-3;
    dg(10,i) = 6*Inc(2,i)-2;
    dg(11,i) = 6*Inc(2,i)-1;
    dg(12,i) = 6*Inc(2,i);

end

%%

%%
% Loop para montar as matriz de Rigidez e de Massa
for i=1:Nelem
% Nós que compõem a barra
    no1=(Elem(i,2)) ;
    no2=(Elem(i,3)) ;
coord_no1 = Coord(no1,:);
    coord_no2 = Coord(no2,:);
% Comprimento
    Delta = ( coord_no2 - coord_no1);

    L1 = norm(Delta);
    dx = Delta(1);
    dy = Delta(2);
    dz = Delta(3);
    l = dx/L1;
    m = dy/L1;
    n = dz/L1;
D = sqrt(l^2 + m^2);
if dx == 0 & dy == 0;
if dz > 0
lambda = [ 0 0 1;

           0 1 0;
          -1 0 0 ];

```

```

else

    lambda = [ 0 0 -1;
               0 1 0;
               1 0 0 ];

end
else
    lambda = [ 1      m      n ;
    -m/D      1/D      0;
              -1*n/D   -m*n/D   D];

end

E = E1(i);           %Vetor de modulo de young
A = A1(i);           %Vetor de modulo de young
L = L1 ;             %Vetor de Area
J = I1(i,1);         %Vetor de inercia em x
Iy = I1(i,2);        %Vetor de inercia em y
Iz = I1(i,3);        %Vetor de inercia em z
G = G1(i);           %Vetor de modulo de Cisalhamento

%%

if TAN==1
% ke
%
% Matriz de rigidez local da estrutura ( BERNOUILLI)
V1=E*A/L;
V2=12*E*Iz/L^3;
V3=6*E*Iz/L^2;
V4=12*E*Iy/L^3;
V5=6*E*Iy/L^2;
V6=G*J/L;
V7=4*E*Iy/L;
V8=4*E*Iz/L;
V9=2*E*Iy/L;
V10=2*E*Iz/L;

ke = [ V1  0  0  0  0  0  -V1  0  0  0  0  0;
      0  V2  0  0  0  0  V3  0  -V2  0  0  0;
      0  0  V4  0  -V5  0  0  0  0  -V4  0  -V5;
      0  0  0  V6  0  0  0  0  0  0  -V6  0;
      0  0  -V5  0  V7  0  0  0  0  V5  0  V9;
      0  V3  0  0  0  0  V8  0  -V3  0  0  0;
      -V1  0  0  0  0  0  0  V1  0  0  0  0;
      0  -V2  0  0  0  -V3  0  V2  0  0  0  -V3;
      0  0  -V4  0  V5  0  0  0  V4  0  V5  0;
      0  0  0  -V6  0  0  0  0  0  V6  0  0;
      0  0  -V5  0  V9  0  0  0  V5  0  V7  0;
      0  V3  0  0  0  V10  0  -V3  0  0  0  V8];

%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Matriz de Massa local da estrutura (BERNOUILLI)
beta =denss*A*L/420;
V1=140;
V2=70;
V3=156;
V4=22*L;
V5=54;

```

```

V6=13*L;
V7=140*J/A;
V8=70*J/A;
V9=4*L^2;
V10=3*L^2;
%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
me =beta*[ V1  0    0    0    0    0    V2  0    0    0    0    0;
           0  V3    0    0    0    V4  0  V5    0    0    0 -V6;
           0  0    V3    0 -V4    0  0  0    V5    0    V6  0;
           0  0    0    V7    0    0  0  0    0    V8    0  0;
           0  0 -V4    0    V9    0  0  0 -V6    0 -V10 0;
           0  V4    0    0    0    V9  0  V6    0    0    0 -V10;
           V2  0    0    0    0    0  V1  0    0    0    0  0;
           0  V5    0    0    0    V6  0  V3    0    0    0 -V4;
           0  0    V5    0 -V6    0  0  0    V3    0    V4  0;
           0  0    0    V8    0    0  0  0    0    V7    0  0;
0    0    V6    0 -V10 0    0    0    V4    0    V9    0;
           0 -V6    0    0    0 -V10 0 -V4    0    0    0  V9];

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%
else
% Parametros que considera que considera a deformação por cisalhamento
%
%                               TIMOSHENKO
ks=5/6;
phiy=12*E*Iy/(ks*A*G*L^2);
phiz=12*E*Iz/(ks*A*G*L^2);

%%
%%                               Axial                               %%%
V1=E*A/L;

%%
%%                               Flexão em Z                               %%%
alphakey=E*Iy/(L^3*(1+phiy));
alphakez=E*Iz/(L^3*(1+phiz));
V2=12*alphakey;
V3=6*L*alphakez;
V8=(4+phiz)*L^2*alphakez;
V10=(2-phiz)*L^2*alphakez;

%%
%%                               Flexão em y                               %%%
V4=12*alphakey;
V5=6*L*alphakey;
V7=(4+phiy)*L^2*alphakey;
V9=(2-phiy)*L^2*alphakey;

%%
%%                               Torção                               %%%
V6=G*J/L;
%%

ke = [ V1  0    0    0    0    0 -V1  0    0    0    0    0;
       0  V2  0    0    0    V3  0 -V2  0    0    0  V3;
       0  0  V4  0 -V5    0  0  0 -V4  0 -V5  0;
       0  0  0  V6  0    0  0  0  0 -V6  0  0;
       0  0 -V5  0  V7    0  0  0  0  V5  0  V9  0;

```

```

0      V3      0      0      0      V8      0      -V3      0      0      0      V10;
-V1     0      0      0      0      0      V1      0      0      0      0      0;
0      -V2     0      0      0      -V3     0      V2      0      0      0      -V3;
0      0      -V4     0      V5      0      0      0      V4      0      V5      0;
0      0      0      -V6     0      0      0      0      0      V6      0      0;
0      0      -V5     0      V9      0      0      0      V5      0      V7      0;
0      V3      0      0      0      V10     0      -V3      0      0      0      V8];

%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
k=5/6;
% Matriz de massa (Timoshenko)
tes=1;
if tes==1
psiy=4*E*Iy/(k*G*A*L^2);
psiz=4*E*Iz/(k*G*A*L^2);
else
psiy=0;
psiz=0;
end

f_Tz=denss*A*L/(840*(1+3*psiz)^2);
f_Rz=denss*Iz/(30*L*(1+3*psiz)^2);

f_Ty=denss*A*L/(840*(1+3*psiy)^2);
f_Ry=denss*Iy/(30*L*(1+3*psiy)^2);
%
p1=312+1764*psiz+2520*psiz^2;
p2=(44+231*psiz+315*psiz^2)*L;
p3=108+756*psiz+1260*psiz^2;
p4=-(26+189*psiz+315*psiz^2)*L;
p5=(8+42*psiz+63*psiz^2)*L^2;
p6=-3*(2+14*psiz+21*psiz^2)*L^2;
p7=36;
p8=-3*(-1+15*psiz)*L;
p9=(4+15*psiz+90*psiz^2)*L^2;
p10=(-1-15*psiz+45*psiz^2)*L^2;

%%
j1=312+1764*psiy+2520*psiy^2;
j2=(44+231*psiy+315*psiy^2)*L;
j3=108+756*psiy+1260*psiy^2;
j4=-(26+189*psiy+315*psiy^2)*L;
j5=(8+42*psiy+63*psiy^2)*L^2;
j6=-3*(2+14*psiy+21*psiy^2)*L^2;
j7=36;
j8=-3*(-1+15*psiy)*L;
j9=(4+15*psiy+90*psiy^2)*L^2;
j10=(-1-15*psiy+45*psiy^2)*L^2;

%%
%
z1=f_Tz*p1+f_Rz*p7;
z2=f_Tz*p2+f_Rz*p8;
z3=f_Tz*p3 - f_Rz*p7;
z4=f_Tz*p4+f_Rz*p8;
%
z5=f_Tz*p5+f_Rz*p9;
z6=-(f_Tz*p4+f_Rz*p8);

```

```

z7=(f_Tz*p6+f_Rz*p10);

%%
y1=f_Ty*j1+f_Ry*j7;
y2=f_Ty*j2+f_Ry*j8;
y3=f_Ty*j3 - f_Ry*j7;
y4=f_Ty*j4+f_Ry*j8;
%
y5=f_Ty*j5+f_Ry*j9;
y6=-(f_Ty*j4+f_Ry*j8);
y7=(f_Ty*j6+f_Ry*j10);

V1=denss*A*L/3;
V2=denss*A*L/6;

V7=denss*J*L/3;
V8=denss*J*L/6;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

me =
0   [ V1  0  0  0  0  0  V2  0  0  0  0  0;
      z1  0  0  0  z2  0  z3  0  0  0  z4 ;
           0  0  y1  0  y2  0  0  0  y3  0  y4  0;
           0  0  0  V7  0  0  0  0  0  0  V8  0;
           0  0  y2  0  y5  0  0  0  y6  0  y7  0;
           0  z2  0  0  0  z5  0  z6  0  0  0  z7;
           V2  0  0  0  0  0  V1  0  0  0  0  0;
           0  z3  0  0  0  z6  0  z1  0  0  0 -z2;
0   0   y3  0  y6  0  0  0  y1  0  z2  0;
           0  0  0  V8  0  0  0  0  0  0  V7  0;
           0  0  y4  0  y7  0  0  0  z2  0  y5  0;
0   z4   0  0  0  z7  0 -z2  0  0  0  z5];

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Montagem da matriz global
Rot = zeros(12,12);

Z = zeros(3,3);

Rot = [ lambda      Z      Z      Z;
        Z      lambda      Z      Z;
        Z      Z      lambda      Z;
        Z      Z      Z      lambda];

Ke = Rot'*ke*Rot; % Matriz de rigidez rotacionada
% Montagem da matriz de rigidez Global da estrutura
for ii=1:12
for jj=1:12
% Matriz de rigidez Global
K(dg(ii,i),dg(jj,i)) = K(dg(ii,i),dg(jj,i)) + Ke(ii,jj);
% Matriz de Massa Global
M(dg(ii,i),dg(jj,i)) = M(dg(ii,i),dg(jj,i)) + me(ii,jj);
end
end

```

```

end
% Introdução dos vínculos na matriz de rigidez com a técnica do número
% grande

for i=1:NDIRES

    K(NRES(i),NRES(i)) = 10^6*K(NRES(i),NRES(i));

end
%%
format short

Desloc = pinv(K)*f;

% ANALISE MODAL
[modo freq]=eig(K,M);
freq=sqrt(freq);

for i=1:size(freq,1);

    F(i) = freq(i,i);

end;
n = size(freq,2);

aux = sortrows([ F ; modo ]')';

F = aux(1,:);

modo = aux(2:n+1,:);
format short
omega=F(1:Nnode)
Periodo=zeros(Nnode,1);
freq_HZ=zeros(Nnode,1);

% PERIODOS
for j=1:Nnode
    Periodo(j,1)=2*pi/omega(j);
    freq_HZ(j,1)=1/Periodo(j);
end

% REACÕES DOS NOS

%Reacoes(NRES) = K(NRES,1:Ndes)*Desloc(1:Ndes)
for i=1:NDIRES

    K(NRES(i),NRES(i)) = K(NRES(i),NRES(i))/10^6;

end
R = K*Desloc-f;
%%

```