

Resumo

Nesta dissertação pretende-se simular o comportamento dinâmico de uma laje de betão armado aplicando o Método de Elementos Finitos através da sua implementação no programa FreeFEM++. Este programa permite-nos a análise do modelo matemático tridimensional da Teoria da Elasticidade Linear, englobando a Equação de Equilíbrio, Equação de Compatibilidade e Relações Constitutivas. Tratando-se de um problema dinâmico é necessário recorrer a métodos numéricos de Integração Directa de modo a obter a resposta em termos de deslocamento ao longo do tempo. Para este trabalho escolhemos os Método de Newmark e o Método de Euler para a discretização temporal, um pela sua popularidade e o outro pela sua simplicidade de implementação. Os resultados obtidos pelo FreeFEM++ são validados através da comparação com resultados adquiridos a partir do SAP2000 e de Soluções Teóricas, quando possível.

Palavras Chave: Método de Elementos Finitos (MEF), FreeFEM++, Dinâmica Estrutural, Elasticidade Linear, Método de Newmark, Método de Euler, Lajes, SAP2000, Solução Analítica de Navier.

Abstract

In this dissertation we are going to simulate the dynamic behavior of a reinforced concrete plate by applying the Finite Element Method with the FreeFEM++ program. Thus we study and analyze the numerical solution of a tridimensional Linear Elastic problem by FreeFEM++. We take attention to the mathematical model based on the fundamental equations of the Theory of Linear Elasticity: the Compatibility Equations, the Equilibrium Equations and the Constitutive laws. For the dynamic problem we take a numerical method by Direct Integration to obtain the displacement over time. For this, the Newmark Method and the Euler Method have been chosen, the first one by its popularity, and the other one by its simplicity. All the results, obtained by the FreeFEM++, are compared to SAP2000 results and to Analytical Solutions, when possible.

Palavras Chave: Finite Element Method (FEM), FreeFEM++, Dynamics of Structures, Linear Elasticity, Newmark Method, Euler Method, Plates, SAP2000, Navier's Analytical Solutions.

Agradecimentos

Gostaria de agradecer aos Professores Mestres José Alberto Rodrigues e Eng.º António de Sousa Gorgulho por todo o apoio e orientação prestado ao longo da realização desta dissertação e gostaria de expressar toda a minha gratidão e reconhecimento por tudo o que me foi ensinado.

Agradeço aos meus Pais e à minha irmã, de uma forma especial, por todo o apoio, carinho e amor sempre demonstrado e a ajuda dada para que fosse possível lutar por este objectivo.

E aos meus amigos, colegas e familiares pelos bons momentos proporcionados e por me acompanharem nesta jornada ao longo destes anos.

Índice

1	Introdução	11
1.1	Enquadramento do Tema	11
1.2	Objectivos	12
1.3	Metodologia	13
1.4	Estrutura da Dissertação	13
2	Dinâmica Estrutural	15
2.1	Considerações Iniciais	15
2.2	Formulação da Equação do Movimento	18
3	Teoria da Elasticidade Linear	20
3.1	Considerações Iniciais	20
3.2	Equações de Equilíbrio	22
3.3	Equações de Compatibilidade	25
3.4	Relações Constitutivas	28
3.5	Formulação Forte e Fraca do Problema	32
4	Soluções Analíticas	37
5	Resolução Numérica de Problemas Dinâmicos de Elasticidade Linear	42
5.1	Método dos Elementos Finitos	42
5.2	Método de Integração Directa	45
5.2.1	Método de Euler	47
5.2.2	Método de Newmark	49
5.3	Implementação no FreeFEM++	50

5.3.1	Descrição do Problema	51
5.3.2	Construção da Malha	51
5.3.3	Definição das Variáveis e Constantes	52
5.3.4	Definição das Macros	53
5.3.5	Definição do Espaço de Aproximação	53
5.3.6	Resolução do Problema	54
5.3.7	Tratamento e Análise dos Resultados	56
6	Análise do Problema Dinâmico	58
6.1	Considerações Iniciais	58
6.2	Laje Encastrada em todos os bordos	60
6.2.1	Malha do GMSH e Resultados do FreeFEM++	60
6.2.2	Malha e Resultados do SAP2000	63
6.2.3	Comparações	64
6.3	Laje Simplesmente Apoiada	67
6.3.1	Malha do GMSH e Resultados do FreeFEM++	67
6.3.2	Malha e Resultados do SAP2000	70
6.3.3	Comparações	71
6.4	Laje Vigada apoiada sobre Pilares Rígidos	75
6.4.1	Malha do GMSH e Resultados do FreeFEM++	75
6.4.2	Malha e Resultados do SAP2000	79
6.4.3	Comparações	80
6.5	Laje Fungiforme	83
6.5.1	Malha do GMSH e Resultados do FreeFEM++	83
6.5.2	Malha e Resultados do SAP2000	86
6.5.3	Comparações	87
6.6	Análise de Tensões	90
7	Conclusão	97
7.1	Conclusões e Considerações Finais do TFM	97
7.2	Propostas de Estudos Futuros	98
8	Bibliografia	99

9	Anexos	101
A	Programas para Caso da Laje Encastrada	102
A.1	Método de Euler	102
A.2	Método de Newmark	106
B	Programas para Caso da Laje Simplesmente Apoiada	109
B.1	Método de Euler	109
B.2	Método de Newmark	113
C	Programas para Caso da Laje Vigada sobre Pilares Rígidos	116
C.1	Método de Euler	116
C.2	Método de Newmark	120
D	Programas para Caso da Laje Fungiforme	123
D.1	Método de Euler	123
D.2	Método de Newmark	127

Lista de Figuras

2.1	Esquema representativo da caracterização do comportamento dinâmico de estruturas, cf. Mendes e Oliveira [12]	16
2.2	Esquemática dos tipos de comportamentos e métodos de avaliação	17
2.3	Idealização de um sistema com apenas um grau de liberdade (SDOF): (a) componentes básicas; (b) forças em equilíbrio, cf. Clough e Penzien [4]	19
3.1	Variáveis e equações fundamentais da Mecânica Estrutural, cf. Mendes e Oliveira [12]	21
3.2	Sólido seccionado sob acção de forças exteriores, cf. Oliveira, Castro e Gomes [15]	22
3.3	Representação de um estado de tensão de um corpo sólido, cf. Oliveira, Castro e Gomes [15]	23
3.4	Forças e tensões actuantes num ponto do interior de um sólido, cf. Oliveira, Castro e Gomes [15]	24
3.5	Deformações de fibras de um corpo sólido, cf. Oliveira, Castro e Gomes [15]	26
3.6	Deformações normais de fibras sob a hipótese de pequenos deslocamentos	27
3.7	Deformações distorcionais de fibras sob a hipótese de pequenos deslocamentos	27
3.8	Curva Tensão-extensão obtida a partir de um ensaio de compressão uniaxial	29
6.1	Malha do GMSH para a laje encastrada	60
6.2	Malha Deformada pelo Método de Euler no instante $t = 0,125 s$	61

6.3	Malha Deformada pelo método de Euler no instante $t = 0,375 s$. . .	61
6.4	Malha Deformada pelo Método de Newmark no instante $t = 0,125 s$.	62
6.5	Malha Deformada pelo método de Newmark no instante $t = 0,375 s$.	62
6.6	Malha do SAP2000 para a laje encastrada	63
6.7	Malha Deformada do SAP2000 no instante $t = 0,125 s$	63
6.8	Malha Deformada do SAP2000 no instante $t = 0,375 s$	64
6.9	Gráfico da resposta em deslocamento do ponto central da laje encastrada	66
6.10	Malha do GMSH para a laje simplesmente apoiada	67
6.11	Malha Deformada pelo Método de Euler no instante $t = 0,125 s$. . .	68
6.12	Malha Deformada pelo método de Euler no instante $t = 0,375 s$. . .	68
6.13	Malha Deformada pelo Método de Newmark no instante $t = 0,125 s$.	69
6.14	Malha Deformada pelo método de Newmark no instante $t = 0,375 s$.	69
6.15	Malha do SAP2000 para a laje simplesmente apoiada	70
6.16	Malha Deformada do SAP2000 no instante $t = 0,125 s$	70
6.17	Malha Deformada do SAP2000 no instante $t = 0,375 s$	71
6.18	Gráfico da resposta em deslocamento do ponto central da laje simplesmente apoiada	75
6.19	Malha do GMSH para a laje vigada	76
6.20	Malha Deformada pelo Método de Euler no instante $t = 0,125 s$. . .	77
6.21	Malha Deformada pelo método de Euler no instante $t = 0,375 s$. . .	77
6.22	Malha Deformada pelo Método de Newmark no instante $t = 0,125 s$.	78
6.23	Malha Deformada pelo método de Newmark no instante $t = 0,375 s$.	78
6.24	Malha do SAP2000 para a laje vigada	79
6.25	Malha Deformada do SAP2000 no instante $t = 0,125 s$	79
6.26	Malha Deformada do SAP2000 no instante $t = 0,375 s$	80
6.27	Gráfico da resposta em deslocamento do ponto central da laje vigada	82
6.28	Malha do GMSH para a laje fungiforme	83
6.29	Malha Deformada pelo Método de Euler no instante $t = 0,125 s$. . .	84
6.30	Malha Deformada pelo método de Euler no instante $t = 0,375 s$. . .	84
6.31	Malha Deformada pelo Método de Newmark no instante $t = 0,125 s$.	85
6.32	Malha Deformada pelo método de Newmark no instante $t = 0,375 s$.	85

6.33	Malha do SAP2000 para a laje fungiforme	86
6.34	Malha Deformada do SAP2000 no instante $t = 0,125 s$	87
6.35	Malha Deformada do SAP2000 no instante $t = 0,375 s$	87
6.36	Gráfico da resposta em deslocamento do ponto central da laje fungiforme	90
6.37	Distribuição de tensões ao longo da laje encastrada segundo a direcção xx para o instante $t = 0,03125 s$, pelo método de Newmark	92
6.38	Distribuição de tensões ao longo da laje encastrada segundo a direcção yy para o instante $t = 0,03125 s$, pelo método de Newmark	93
6.39	Distribuição de tensões tangenciais ao longo da laje encastrada para o instante $t = 0,03125 s$, pelo método de Newmark	93
6.40	Distribuição de tensões ao longo da laje encastrada segundo a direcção xx para o instante $t = 0,03125 s$, pelo método de Euler	94
6.41	Distribuição de tensões ao longo da laje encastrada segundo a direcção yy para o instante $t = 0,03125 s$, pelo método de Euler	94
6.42	Distribuição de tensões tangenciais ao longo da laje encastrada para o instante $t = 0,03125 s$, pelo método de Euler	95
6.43	Distribuição de tensões ao longo da laje encastrada segundo a direcção xx para o instante $t = 0,03125 s$, pelo SAP2000	95
6.44	Distribuição de tensões ao longo da laje encastrada segundo a direcção yy para o instante $t = 0,03125 s$, pelo SAP2000	96
6.45	Distribuição de tensões tangenciais ao longo da laje encastrada para o instante $t = 0,03125 s$, pelo SAP2000	96

Lista de Tabelas

6.1	Frequências naturais de vibração	59
6.2	Tabela Comparativa de resultados do deslocamento vertical do ponto central da laje encastrada	65
6.3	Tabela Comparativa de resultados do deslocamento vertical do ponto central da laje simplesmente apoiada pelo método de Euler	72
6.4	Tabela Comparativa de resultados do deslocamento vertical do ponto central da laje simplesmente apoiada	74
6.5	Tabela Comparativa de resultados do deslocamento vertical do ponto central da laje vigada	81
6.6	Tabela Comparativa de resultados do deslocamento vertical do ponto central da laje fungiforme	89
6.7	Tabela Comparativa de resultados de tensões máximas e mínimas da laje encastrada	92

Capítulo 1

Introdução

1.1 Enquadramento do Tema

A análise estrutural é tida como um dos principais objectivos de um Engenheiro de Estruturas. Tal análise poderá ser considerada estática ,quase-estática ou dinâmica dependendo essencialmente do tipo de carregamento ao qual a estrutura está exposta e da sua resposta. Pode-se dizer que a análise estática de uma estrutura representa um caso particular da análise dinâmica, pois o termo dinâmica admite quer uma variação no tempo quer a consideração do efeito das forças de inércia resultantes na determinação da resposta do sistema estrutural.

Nos dias de hoje com os avanços informáticos, os investigadores e projectistas passaram a dispor de ferramentas que lhes permitem desenvolver modelos mais eficientes de análise estrutural contribuindo tanto no aumento da confiança nos resultados obtidos da análise como também na diminuição dos custos e dos prazos de execução. Por outro lado, com o desenvolvimento de materiais cada vez mais resistentes, a tendência é a execução de estruturas mais esbeltas e flexíveis, tendo em vista tanto o factor económico como o factor estético das mesmas. No entanto, a execução de estruturas com menores secções transversais e maiores vãos torna estas mais sensíveis face a acções variáveis no tempo sobretudo em termos dos efeitos de segunda ordem. Daí ser recomendada a execução de uma análise estrutural dinâmica durante a fase de projecto de uma estrutura.

Algumas acções dinâmicas que podem despertar maior interesse, sobretudo devido

ao efeito danificador nos elementos estruturais, são: os sismos, a movimentação de pessoas (caminhar, correr, dançar, pular) e as máquinas sobre as lajes, a incidência do vento sobre edifícios altos e a aceleração de veículos sobre as pontes.

Actualmente, deparando-se com problemas cada vez mais complexos no campo de análise de estruturas que tornam a obtenção de soluções exactas bastante difícil ou até impossível, o recurso à modelação numérica por parte dos engenheiros tem sido cada vez mais frequente. Entre os diversos métodos desenvolvidos, o Método dos Elementos Finitos (MEF) tem-se revelado uma ferramenta bastante poderosa e versátil para a engenharia civil, permitindo análises com rigor de uma grande variedade de fenómenos físicos e de leis de comportamento.

Na aplicação do MEF no domínio da Mecânica dos Sólidos, o modelo de compatibilidade tem sido o mais comum devido à sua simplicidade e por apresentar uma formulação menos exigente e mais intuitiva.

As motivações subjacentes à realização deste trabalho são a análise do modelo matemático dinâmico através da sua implementação numérica pelo Método dos Elementos Finitos, usando várias técnicas de discretização.

1.2 Objectivos

O principal objectivo deste trabalho reside na simulação numérica do comportamento de lajes com condições de fronteira distintas e sujeitas a acções dinâmicas, neste caso uma solicitação vibratória harmónica. A simulação será feita com implementação do Método dos Elementos Finitos recorrendo ao software FreeFEM++ e será uma simulação em termos de resposta em deslocamentos do sistema ao longo do tempo. Com este software será possível fazer uma análise das equações que modelam o fenómeno tridimensional e verificar qual a dependência relativamente às condições de fronteira impostas. A validação dos resultados obtidos a partir do freeFEM++ será feita comparativamente aos resultados obtidos com o software SAP2000, e quando possível pelas soluções analíticas. Com base na comparação quantitativa e qualitativa dos resultados obtidos pelos diferentes meios serão estabelecidas as conclusões e considerações finais.

1.3 Metodologia

A metodologia adoptada para o trabalho é:

- Definição do modelo físico genérico;
- Formulação do modelo matemático genérico;
- Implementação em FreeFEM++;
- Obtenção dos resultados do FreeFEM++;
- Obtenção dos resultados do SAP2000;
- Obtenção das Soluções Teóricas;
- Apresentação de tabelas e gráficos comparativos;
- E a descrição das conclusões finais em relação ao trabalho.

1.4 Estrutura da Dissertação

A dissertação encontra-se dividida em sete capítulos estruturados da seguinte forma:

No capítulo 1: Introduzimos o tema, indicando os objectivos da dissertação bem como a metodologia utilizada para alcançar estes objectivos.

No capítulo 2: Descrevemos alguns conceitos duma análise dinâmica estrutural e introduzimos a formulação matemática dum problema dinâmico.

No capítulo 3: Apresentamos a formulação do problema dinâmico de Elasticidade Linear tridimensional, definindo as hipóteses admitidas e as relações fundamentais entre as suas incógnitas. Também neste capítulo é feita a transformação das equações que regem o problema conducentes à obtenção da formulação fraca, com vista à sua implementação utilizando o Método de Elementos Finitos.

No capítulo 4: É feita a dedução de uma solução Teórica, através do método de separação de variáveis e das séries de Fourier e é apresentada uma solução Teórica proposta por Navier para o caso de lajes simplesmente apoiadas.

No capítulo 5: Faz-se uma descrição dos métodos a implementar, nomeadamente o Método dos Elementos Finitos e os Métodos de Integração Directa para discretização em ordem ao tempo. Também neste capítulo é feita uma descrição da implementação do nosso problema dinâmico de elasticidade linear no software FreeFEM++

No capítulo 6: São apresentados os casos de estudo nos quais foram executadas as análises em que é feita a comparação dos resultados obtidos pelo FreeFEM++, pelo SAP2000 e pelas soluções analíticas.

No capítulo 7: Apresentamos as conclusões e as considerações finais em relação ao trabalho.

E por fim seguem-se a Bibliografia no capítulo 8 e os Anexos.

Capítulo 2

Dinâmica Estrutural

2.1 Considerações Iniciais

Sendo o termo "dinâmico", na linguagem corrente, definido como a existência de uma variação temporal, um carregamento dinâmico é entendido como um carregamento que apresenta uma variação no tempo quer em termos de intensidade, posição ou direcção. Desta forma a resposta estrutural, os esforços, as tensões e as deformações relativamente ao dito carregamento também apresenta tal variação temporal.

Numa análise estrutural linear (geométrica e material) é sempre conveniente fazer a separação entre a componente estática e a componente dinâmica das acções actuantes. Deste modo a avaliação da resposta da estrutura é executada separadamente sendo feita posteriormente a sobreposição das suas respostas (possível devido à linearidade da estrutura) por forma a se conseguir obter o efeito total.

Em comparação à típica análise estática de estruturas, a análise dinâmica apresenta um certo acréscimo quanto ao grau de dificuldade e de complexidade na obtenção das suas soluções. Isto dada a consideração da inércia e do amortecimento nas forças elásticas resistentes e a dependência do tempo de todas as forças intervenientes. O que faz com que um problema dinâmico não tenha apenas uma única solução, como acontece num problema estático, mas sim uma sucessão de soluções para todos os instantes de cálculo.

Segundo Clough e Penzien [4], para avaliação da resposta estrutural a uma solitação dinâmica estão disponíveis dois procedimentos: determinista e estocástica (não-determinista). A escolha da metodologia a adotar está inteiramente ligada à acção dinâmica a que a estrutura está sujeita. No caso da acção dinâmica apresentar uma variação temporal completamente conhecida a análise é considerada determinista. Mas por outro lado, se a acção dinâmica apresentar uma natureza aleatória, então a análise será estocástica, podendo ser caracterizada através de conceitos probabilísticos. Ainda no campo de acções dinâmicas deterministas, estas poderão ser divididas em duas categorias, podendo ser periódicas, caso o carregamento demonstre a mesma variação temporal sucessivamente em vários ciclos, ou não-periódicas.

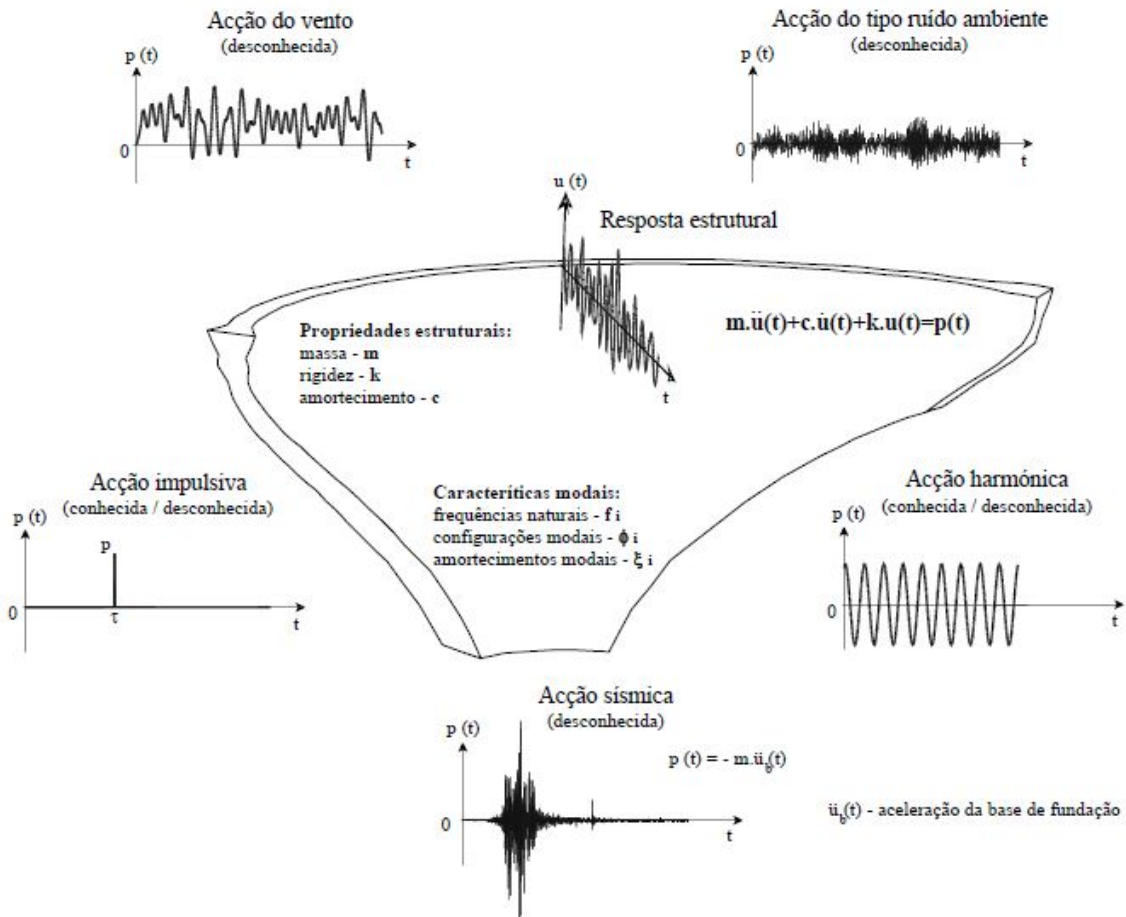


Figura 2.1: Esquema representativo da caracterização do comportamento dinâmico de estruturas, cf. Mendes e Oliveira [12]

Já Cook e Malkus [5] classificam os problemas dinâmicos segundo duas tipologias: problemas de propagação de ondas ou problemas dinâmicos estruturais. Nos problemas de propagação de ondas os carregamentos são frequentemente impactos ou explosões e quer a excitação quer a resposta da estrutura apresentam altas frequências. Já nos problemas dinâmicos estruturais a frequência de excitação encontra-se na mesma ordem de grandeza de um terço da frequência natural de vibração mais baixa da estrutura. A inércia torna-se importante se as frequências de excitação são superiores a aproximadamente um terço da frequência natural de vibração mais baixa da estrutura ou se a estrutura vibra livremente quando imposta a determinadas condições iniciais. Caso contrário o problema diz-se Quase Estático, tratando-se de um problema com uma variação temporal em que não há influência da inércia no comportamento da estrutura. Por sua vez os problemas dinâmicos estruturais subdividem-se em duas amplas classificações em que numa procuram-se as frequências naturais de vibração e correspondentes modos de vibração, ou seja uma análise modal, e noutra procura-se determinar o movimento da estrutura ao longo do tempo para determinadas solicitações, ou seja uma análise histórico-temporal. Assim sendo podemos apresentar o seguinte esquema do tipo de comportamentos estruturais:

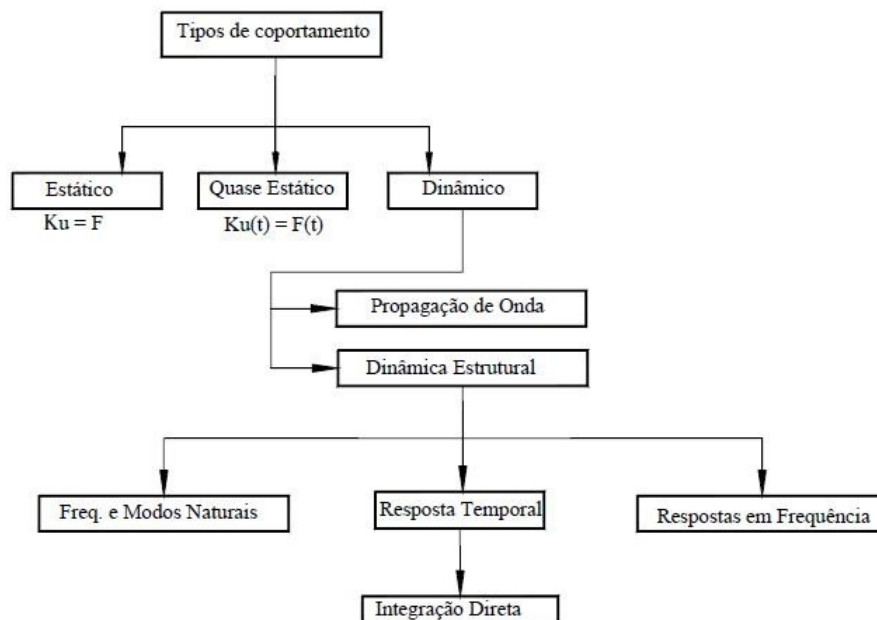


Figura 2.2: Esquematização dos tipos de comportamentos e métodos de avaliação

Para além da definição das acções actuantes, a caracterização do comportamento dinâmico das estruturas passa também pela consideração de um modelo matemático que se mostre suficientemente capaz de representar as propriedades geométricas e físicas das mesmas e de descrever o seu funcionamento estrutural por aplicação das leis da Mecânica de modo que seja possível a obtenção das relações excitação-resposta. E estes critérios aplicam-se tanto a nível determinista como a nível estocástico.

No âmbito desta dissertação apenas será abordada a metodologia determinista uma vez que a simulação do comportamento dinâmico, neste caso de uma laje, é feita face a uma excitação harmónica simples caracterizada por uma variação sinusoidal, plenamente conhecida.

2.2 Formulação da Equação do Movimento

Os deslocamentos dinâmicos são definidos por expressões matemáticas designadas de equações do movimento das estruturas em que o histórico dos deslocamentos representam as soluções destas equações. Na verdade as equações do movimento de qualquer sistema dinâmico têm implícita as expressões da segunda lei do movimento de Newton que diz que a taxa de variação da quantidade do movimento de qualquer partícula de massa m é igual à força que actua sobre o mesmo. O qual para um sistema com apenas um grau de liberdade (SDOF) poderá ser expresso matematicamente pela seguinte equação diferencial:

$$P(t) = \frac{d}{dt} \left(m \frac{du}{dt} \right) \quad (2.1)$$

Em que $F(t)$ representa as forças variáveis no tempo aplicadas e $u(t)$ é o deslocamento da partícula de massa m . Considerando que a massa é constante no tempo a Equação 2.1 poderá ser escrita da seguinte forma:

$$P(t) = m \frac{d^2u}{dt^2} = m\ddot{u}(t) \quad (2.2)$$

Segundo o princípio d'Alembert, uma massa sujeita a uma aceleração desenvolve uma força de inércia proporcional a esta mas de sentido oposto. Este conceito permite que as equações do movimento sejam expressas como equações de equilíbrio dinâmico em que o somatório de todas as forças aplicadas à massa m e das forças de inércia, $f_I(t) = m\ddot{u}(t)$, associadas à mesma se anulam:

$$P(t) = m\ddot{u}(t) \Leftrightarrow P(t) - f_I(t) = 0 \quad (2.3)$$

Como $P(t)$ podem ser consideradas vários tipos de forças actuantes tais como forças resistentes resultantes de restrições elásticas que se opõem ao deslocamento (forças de restituição elástica, $f_S(t) = ku(t)$), forças de viscosidade que resistem à velocidade (forças de dissipação de energia, $f_D(t) = c\dot{u}(t)$) e excitações externas, $f(t)$.

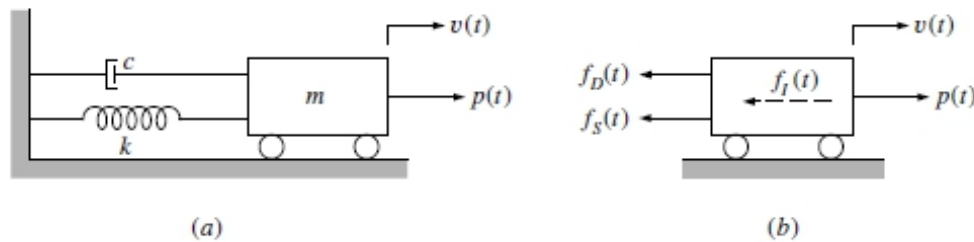


Figura 2.3: Idealização de um sistema com apenas um grau de liberdade (SDOF): (a) componentes básicas; (b) forças em equilíbrio, cf. Clough e Penzien [4]

Logo para um sistema simples com apenas um grau de liberdade a equação do movimento é dada por:

$$m\ddot{u}(t) + c\dot{u}(t) + ku(t) = f(t) \Leftrightarrow f_I(t) + f_D(t) + f_S(t) = f(t) \quad (2.4)$$

Ou seja esta formulação dá-se por meio de uma relação directa de equilíbrio em cada instante do tempo de todas as forças do sistema.

Quando o sistema estrutural se torna mais complexo, passando a ter múltiplos graus de liberdade, torna-se mais conveniente que a formulação seja feita recorrendo ao Princípio dos Trabalhos Virtuais (PTV), a qual é apresentada no capítulo 5.

Capítulo 3

Teoria da Elasticidade Linear

3.1 Considerações Iniciais

Neste capítulo são apresentadas as relações fundamentais que estão na base da formulação do problema dinâmico de elasticidade linear. O problema a ser estudado segue um conjunto de hipóteses base como:

- Os elementos estruturais do sistema são constituídos por um material homogêneo e isotrópico, ou seja, o material apresenta as mesmas propriedades ao longo de toda a estrutura e apresenta o mesmo comportamento em todas as direções.
- Admite-se um comportamento elástico linear da estrutura. Tal linearidade é verificada simultaneamente em dois níveis. No primeiro nível, o campo de tensões e o campo de deformações relacionam-se de forma linear (linearidade física ou do material) em que as deformações resultantes da aplicação de cargas exteriores são reversíveis. E no segundo nível, considerando que o corpo elástico está sujeito a pequenas deformações, o campo de deslocamentos e o campo de deformações também serão lineares em que as componentes do campo de deformações são bastante aproximadas da combinação linear das componentes do campo de deslocamentos.
- A temperatura não interfere nas características mecânicas do material.

Verificando tais hipóteses, torna-se válida a aplicação da Teoria da Elasticidade Linear que para o caso tridimensional é regida por um conjunto de 15 equações diferenciais, as quais devem satisfazer as grandezas da nossa análise estrutural nomeadamente os deslocamentos, as deformações, as forças e as tensões.

Estas equações diferenciais encontram-se divididas em três Equações de Equilíbrio, que relacionam as forças e as tensões, seis Equações de Compatibilidade, que relacionam as deformações e os deslocamentos, e seis Relações Constitutivas, que relacionam as tensões e as deformações (figura 3.1):

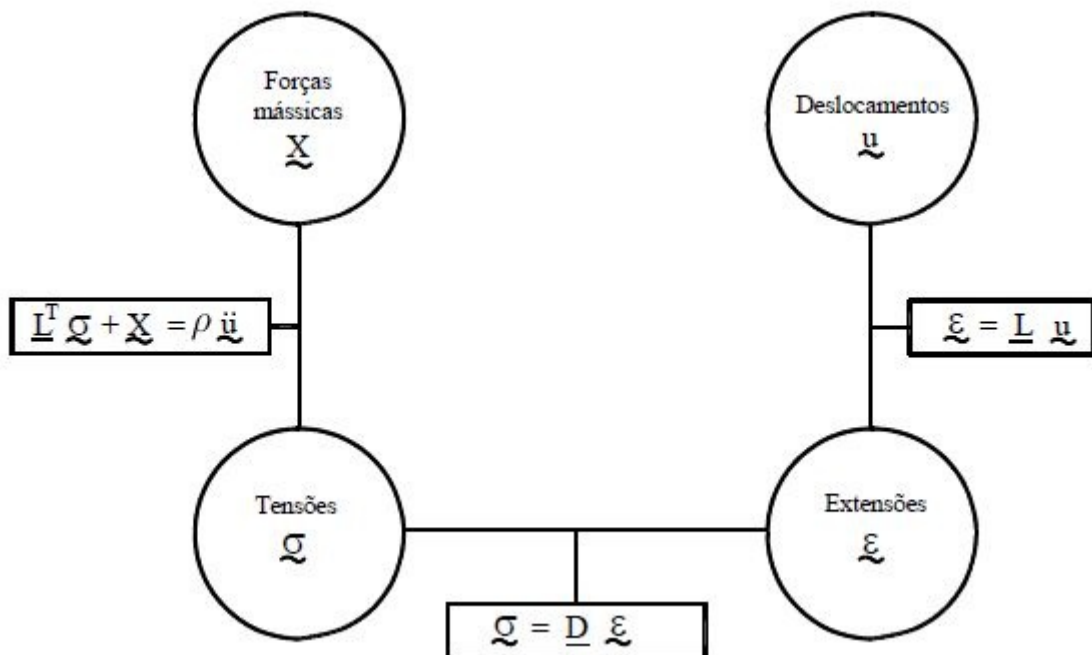


Figura 3.1: Variáveis e equações fundamentais da Mecânica Estrutural, cf. Mendes e Oliveira [12]

3.2 Equações de Equilíbrio

Um corpo sólido sob a influência de forças exteriores, mássicas e/ou de superfície, origina forças internas (ou esforços internos) cujo objectivo é manter ligadas as partículas do corpo e que se encontram distribuídas ao longo do mesmo. Para a quantificação e a distribuição destas forças internas pode-se recorrer ao "método do plano de corte". Portanto tendo um ΔA que corresponde ao elemento de superfície do plano de corte com uma normal exterior \vec{n} e que contém o ponto P e onde actua a força $\Delta \vec{F}$ (equivalente às forças internas), tal como é ilustrado pela figura 3.2.

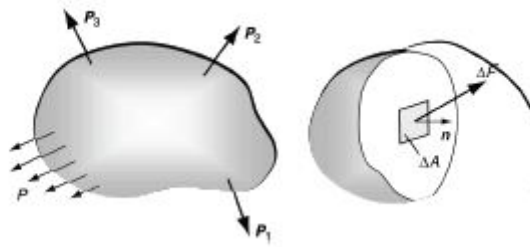


Figura 3.2: Sólido seccionado sob acção de forças exteriores, cf. Oliveira, Castro e Gomes [15]

Segundo Cauchy, o vector de tensão de um determinado ponto P , pertencente a ΔA situado no interior do sólido contínuo e sujeito a uma resultante das forças internas instaladas $\Delta \vec{F}$, é dado por:

$$\vec{\sigma}^{(n)} = \lim_{\Delta A \rightarrow 0} \frac{\Delta \vec{F}}{\Delta A} \quad (3.1)$$

Mas para definir o estado de tensão no ponto P , o vector de tensão $\vec{\sigma}^{(n)}$ revela-se insuficiente dada a existência de infinitos vectores de tensão associados ao ponto P , uma vez que se pode ter infinitos planos de corte que passam por P . Para que o estado de tensão num dado ponto fique totalmente definido é necessário conhecer três vectores de tensão de três planos cujas normais sejam ortogonais entre si.

Logo tendo como referência os eixos cartesianos e considerando o ΔA coincidente com os planos cuja normal é paralela a estes eixos, para o tensor das tensões (notação matricial) tem-se:

$$\sigma = \begin{bmatrix} (x) \\ \vec{\sigma} \\ (y) \\ \vec{\sigma} \\ (z) \\ \vec{\sigma} \end{bmatrix} = \begin{bmatrix} (x) & (x) & (x) \\ \sigma_x & \sigma_y & \sigma_z \\ (y) & (y) & (y) \\ \sigma_x & \sigma_y & \sigma_z \\ (z) & (z) & (z) \\ \sigma_x & \sigma_y & \sigma_z \end{bmatrix} = \begin{bmatrix} \sigma_{xx} & \tau_{xy} & \tau_{xz} \\ \tau_{yx} & \sigma_{yy} & \tau_{yz} \\ \tau_{zx} & \tau_{zy} & \sigma_{zz} \end{bmatrix} \quad \text{ou} \quad \sigma = \begin{bmatrix} \sigma_{11} & \sigma_{12} & \sigma_{13} \\ \sigma_{21} & \sigma_{22} & \sigma_{23} \\ \sigma_{31} & \sigma_{32} & \sigma_{33} \end{bmatrix}$$

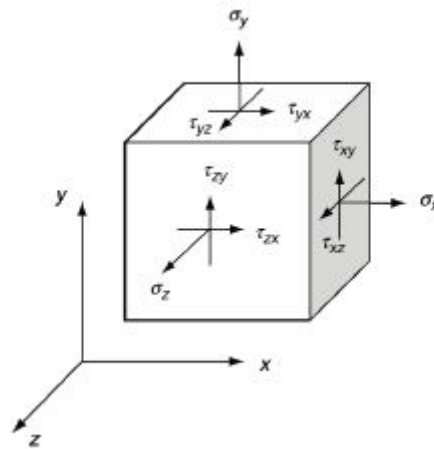


Figura 3.3: Representação de um estado de tensão de um corpo sólido, cf. Oliveira, Castro e Gomes [15]

Assim sendo o estado de tensão num ponto é definido por nove componentes escalares independentes (normais e tangenciais) em que a partir do equilíbrio de momentos do corpo livre verifica-se que a matriz do tensor das tensões é simétrica segundo a diagonal, $\sigma_{ij} = \sigma_{ji}$. Daí só ser necessário conhecer seis das componentes (três normais e três tangenciais) podendo o tensor ser escrito na seguinte forma vectorial:

$$\sigma = \begin{bmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{33} \\ \sigma_{23} \\ \sigma_{13} \\ \sigma_{12} \end{bmatrix}$$

Para a formulação das Equação de Equilíbrio Dinâmico inicialmente deve-se ter em conta o Princípio D'Alembert, o qual diz que um corpo sujeito a uma excitação desenvolve no seu interior forças de inércia cujo valor é proporcional à sua massa e com direcção oposta à aceleração imposta. E como foi referido anteriormente, um corpo sob solicitações externas gera no seu interior esforços internos, e conseqüentemente um campo de tensões internas que segue uma distribuição sob a condição de equilíbrio dinâmico em todos os pontos do corpo. Admitindo um elemento cúbico infinitesimal, dV , como sendo um determinado ponto do interior de um corpo sólido sujeito a uma excitação mássica $F(t)$ tem-se:

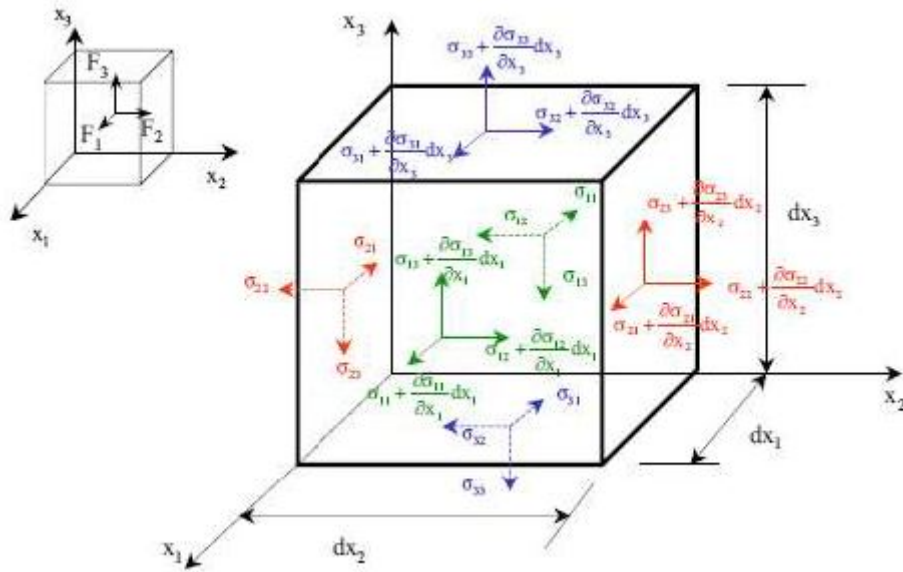


Figura 3.4: Forças e tensões actuantes num ponto do interior de um sólido, cf. Oliveira, Castro e Gomes [15]

Pela condição de equilíbrio das forças (forças internas resultantes da distribuição interna das tensões, força externa aplicada e da força de inércia interna), segundo as três direcções ortogonais, é possível escrever as três Equações de Equilíbrio Dinâmico que governam a Teoria da Elasticidade Linear numa análise tridimensional:

$$\begin{aligned} \frac{\partial \sigma_{11}}{\partial x_1} + \frac{\partial \sigma_{21}}{\partial x_2} + \frac{\partial \sigma_{31}}{\partial x_3} + F_1 &= \rho \ddot{u}_1 \\ \frac{\partial \sigma_{12}}{\partial x_1} + \frac{\partial \sigma_{22}}{\partial x_2} + \frac{\partial \sigma_{32}}{\partial x_3} + F_2 &= \rho \ddot{u}_2 \\ \frac{\partial \sigma_{13}}{\partial x_1} + \frac{\partial \sigma_{23}}{\partial x_2} + \frac{\partial \sigma_{33}}{\partial x_3} + F_3 &= \rho \ddot{u}_3 \end{aligned}$$

Que na notação de Einstein pode ser escrita da seguinte forma:

$$\sigma_{ij,j} + F_i = \rho \ddot{u}_i \quad (3.2)$$

3.3 Equações de Compatibilidade

Já no caso das deformações a análise já não é feita em termos de planos mas sim em termos de fibras. Outra particularidade da Teoria das Deformações, é que esta pode ser aplicada quer para materiais em fase elástica quer para materiais em fase plástica, e isto devido ao facto de se tratar de um problema geométrico e totalmente independente das propriedades do material do sólido em estudo.

Tal como para as tensões, o estado de deformação num ponto fica perfeitamente definida se forem conhecidos três vectores de deformação em fibras mutuamente ortogonais.

Logo de forma análoga às tensões, tem-se um tensor das deformações dado por:

$$\varepsilon = \begin{bmatrix} \vec{\varepsilon}^{(x)} \\ \vec{\varepsilon}^{(y)} \\ \vec{\varepsilon}^{(z)} \end{bmatrix} = \begin{bmatrix} \varepsilon_x^{(x)} & \varepsilon_y^{(x)} & \varepsilon_z^{(x)} \\ \varepsilon_x^{(y)} & \varepsilon_y^{(y)} & \varepsilon_z^{(y)} \\ \varepsilon_x^{(z)} & \varepsilon_y^{(z)} & \varepsilon_z^{(z)} \end{bmatrix} = \begin{bmatrix} \varepsilon_{xx} & \varepsilon_{xy} & \varepsilon_{xz} \\ \varepsilon_{yx} & \varepsilon_{yy} & \varepsilon_{yz} \\ \varepsilon_{zx} & \varepsilon_{zy} & \varepsilon_{zz} \end{bmatrix} \quad \text{ou} \quad \varepsilon = \begin{bmatrix} \varepsilon_{11} & \varepsilon_{12} & \varepsilon_{13} \\ \varepsilon_{21} & \varepsilon_{22} & \varepsilon_{23} \\ \varepsilon_{31} & \varepsilon_{32} & \varepsilon_{33} \end{bmatrix}$$

A simetria em relação à diagonal também se verifica para o tensor das deformações tendo apenas seis componentes distintas, $\varepsilon_{ij} = \varepsilon_{ji}$. Assim sendo para forma vectorial do tensor das deformações tem-se:

$$\varepsilon = \begin{bmatrix} \varepsilon_{11} \\ \varepsilon_{22} \\ \varepsilon_{33} \\ \gamma_{23} \\ \gamma_{13} \\ \gamma_{12} \end{bmatrix}$$

Na aplicação de forças a sólidos contínuos podem ser considerados dois tipos de deformações, a deformação axial/normal (ε_N) em que a fibra sofre uma variação de comprimento do estado indeformado para o estado deformado e a deformação distorcional (ε_D) em que duas fibras ortogonais no estado indeformado deixam de ser ortogonais no estado deformado.

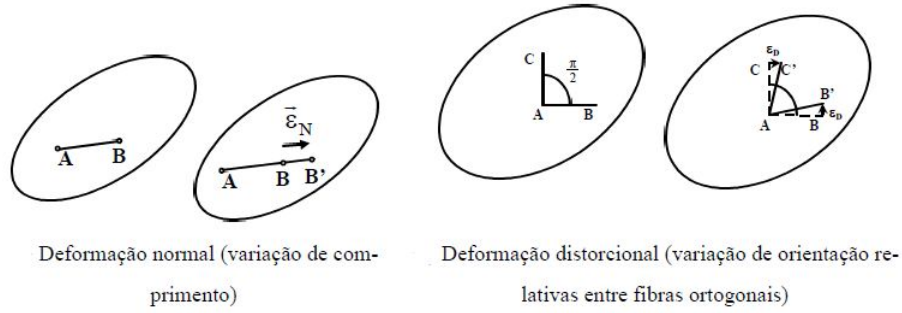


Figura 3.5: Deformações de fibras de um corpo sólido, cf. Oliveira, Castro e Gomes [15]

A deformação normal é quantificada pela taxa de variação de comprimento da fibra:

$$\varepsilon_N = \frac{L - L_0}{L_0}$$

Em que L representa o comprimento da fibra no estado deformado e L_0 representa o comprimento da fibra no estado inicial ou indeformado.

Enquanto que a deformação distorcional é quantificada a partir da distorção, ou seja, a partir da variação do ângulo entre as duas fibras ortogonais:

$$\gamma = \frac{\pi}{2} - \phi = 2\alpha$$

$$\varepsilon_D \approx \alpha = \frac{1}{2} \left(\frac{\pi}{2} - \phi \right)$$

Em que ϕ representa o ângulo formado pelas duas fibras ortogonais no estado deformado e α representa o ângulo formado por cada fibra no estado deformado e indeformado.

Seguindo as formulações de deformação normal e deformação distorcional tem-se:

- Para $\varepsilon_N \neq 0$; $\varepsilon_D = 0$:

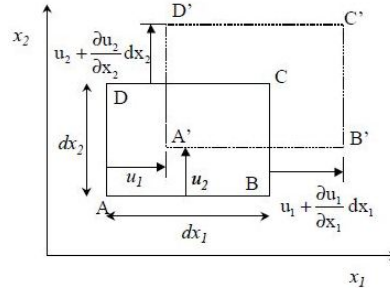


Figura 3.6: Deformações normais de fibras sob a hipótese de pequenos deslocamentos

$$\varepsilon_{11} = \frac{A'B' - AB}{AB} = \frac{\left(dx_1 - u_1 + u_1 + \frac{\partial u_1}{\partial x_1} dx_1\right) - dx_1}{dx_1} = \frac{\partial u_1}{\partial x_1} = \frac{1}{2} \left(\frac{\partial u_1}{\partial x_1} + \frac{\partial u_1}{\partial x_1} \right)$$

$$\varepsilon_{22} = \frac{A'D' - AD}{AD} = \frac{\left(dx_2 - u_2 + u_2 + \frac{\partial u_2}{\partial x_2} dx_2\right) - dx_2}{dx_2} = \frac{\partial u_2}{\partial x_2} = \frac{1}{2} \left(\frac{\partial u_2}{\partial x_2} + \frac{\partial u_2}{\partial x_2} \right)$$

Logo de forma análoga teremos:

$$\varepsilon_{33} = \frac{\partial u_3}{\partial x_3} = \frac{1}{2} \left(\frac{\partial u_3}{\partial x_3} + \frac{\partial u_3}{\partial x_3} \right)$$

- Para $\varepsilon_N = 0$; $\varepsilon_D \neq 0$:

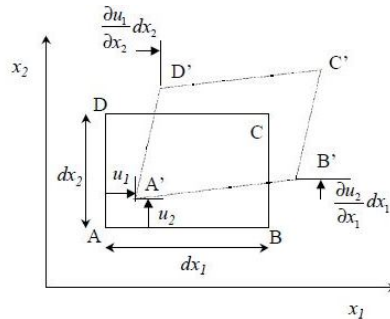


Figura 3.7: Deformações distorcionais de fibras sob a hipótese de pequenos deslocamentos

$$\alpha_1 = \arctan \left(\frac{\frac{\partial u_1}{\partial x_2} dx_2}{dx_2} \right) \approx \frac{\partial u_1}{\partial x_2}$$

$$\alpha_2 = \arctan \left(\frac{\frac{\partial u_2}{\partial x_1} dx_1}{dx_1} \right) \approx \frac{\partial u_2}{\partial x_1}$$

$$\gamma_{12} = \alpha_1 + \alpha_2 = \frac{\partial u_1}{\partial x_2} + \frac{\partial u_2}{\partial x_1} \Leftrightarrow \varepsilon_{12} = \frac{1}{2} \gamma_{12} = \frac{1}{2} \left(\frac{\partial u_1}{\partial x_2} + \frac{\partial u_2}{\partial x_1} \right)$$

Logo de forma análoga teremos:

$$\gamma_{31} = \alpha_3 + \alpha_1 = \frac{\partial u_3}{\partial x_1} + \frac{\partial u_1}{\partial x_3} \Leftrightarrow \varepsilon_{31} = \frac{1}{2} \gamma_{31} = \frac{1}{2} \left(\frac{\partial u_3}{\partial x_1} + \frac{\partial u_1}{\partial x_3} \right)$$

$$\gamma_{23} = \alpha_2 + \alpha_3 = \frac{\partial u_2}{\partial x_3} + \frac{\partial u_3}{\partial x_2} \Leftrightarrow \varepsilon_{23} = \frac{1}{2} \gamma_{23} = \frac{1}{2} \left(\frac{\partial u_2}{\partial x_3} + \frac{\partial u_3}{\partial x_2} \right)$$

Desta forma, na hipótese de pequenos deslocamentos, as Equações de Compatibilidade poderão ser escritas da seguinte forma, seguindo a notação indicial de Einstein:

$$\varepsilon_{ij} = \frac{1}{2} \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \quad (3.3)$$

3.4 Relações Constitutivas

Pela lei de Hooke, para um material que apresente um comportamento elástico linear é verificada uma relação de proporcionalidade directa entre a tensão normal instalada e a deformação normal resultante. Desta relação resulta a constante de proporcionalidade conhecida como Módulo de Elasticidade Longitudinal ou Módulo de Young (E) que poderá ser entendida como a tensão normal necessária para obter uma deformação axial unitária. Logo a lei de Hooke poderá ser representada pela seguinte expressão:

$$\sigma_x = E \varepsilon_x \quad (3.4)$$

Recorrendo a um gráfico obtido a partir de um ensaio de compressão axial o Módulo de Elasticidade Longitudinal é representada pelo declive do troço linear que caracteriza a zona de comportamento elástico do material, como se pode constatar pela figura 3.8

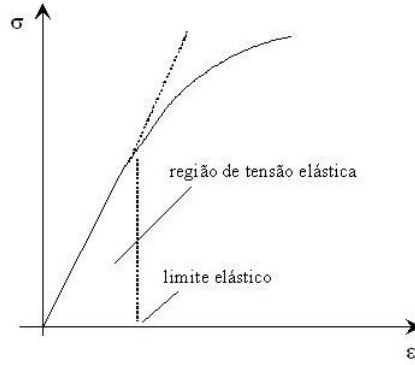


Figura 3.8: Curva Tensão-extensão obtida a partir de um ensaio de compressão uniaxial

Uma vez que para o presente estudo é admitido um material ideal elástico, homogéneo e isotrópico o Módulo de Elasticidade Longitudinal (E) toma um valor constante. Consequentemente poderá ser escrita a seguinte Relação Constitutiva entre o campo de tensões e de deformações para um problema tridimensional (Lei de Hooke Generalizada):

$$\sigma_{ij} = C_{ij,rs} \cdot \varepsilon_{rs} \quad (3.5)$$

Em que σ_{ij} e ε_{rs} são os vectores com as seis componentes independentes do tensor das tensões e das deformações respectivamente e $C_{ij,rs}$ é o tensor constitutivo (matriz de elasticidade simétrica e com 36 componentes) que pode ser escrito da seguinte forma:

$$C_{ij,rs} = \lambda \delta_{ij} \cdot \delta_{rs} + \mu (\delta_{ir} \cdot \delta_{js} + \delta_{is} \cdot \delta_{jr}) \quad (3.6)$$

Substituindo a expressão (3.6) na equação (3.5) tem-se:

$$\begin{aligned}\sigma_{ij} &= \lambda \delta_{ij} \cdot \delta_{rs} \cdot \varepsilon_{rs} + \mu (\delta_{ir} \cdot \delta_{js} \cdot \varepsilon_{rs} + \delta_{is} \cdot \delta_{jr} \cdot \varepsilon_{rs}) \\ &= \lambda \delta_{ij} \varepsilon_{kk} + 2\mu \varepsilon_{ij}\end{aligned}\quad (3.7)$$

Em que λ e μ são as denominadas Constantes de Lamé, δ_{ij} é o delta de Kronecker e ε_{kk} representa o primeiro invariante do tensor das deformações.

As Constantes de Lamé podem ser calculadas com recurso ao Módulo de Elasticidade Longitudinal (E) e ao Coeficiente de Poisson (ν) a partir das seguintes expressões:

$$\mu = G = \frac{E}{2(1+\nu)} \quad \text{e} \quad \lambda = \frac{E\nu}{(1+\nu)(1-2\nu)} \quad (3.8)$$

Em que G é o Módulo de Elasticidade Transversal ou de cisalhamento obtido a partir do ensaio de corte. O que nos permite escrever a Relação Constitutiva em função das propriedades do material como:

$$\begin{aligned}\sigma_{ij} &= \frac{E\nu}{(1+\nu)(1-2\nu)} \delta_{ij} \varepsilon_{kk} + \frac{E}{(1+\nu)} \varepsilon_{ij} \\ &= \frac{E}{(1+\nu)(1-2\nu)} [\nu \delta_{ij} \varepsilon_{kk} + (1-2\nu) \varepsilon_{ij}]\end{aligned}\quad (3.9)$$

Assim sendo, numa notação matricial, as seis componentes independentes do tensor das tensões são dadas por:

$$\begin{bmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{33} \\ \sigma_{23} \\ \sigma_{31} \\ \sigma_{12} \end{bmatrix} = \begin{bmatrix} \lambda + 2\mu & \lambda & \lambda & 0 & 0 & 0 \\ \lambda & \lambda + 2\mu & \lambda & 0 & 0 & 0 \\ \lambda & \lambda & \lambda + 2\mu & 0 & 0 & 0 \\ 0 & 0 & 0 & \mu & 0 & 0 \\ 0 & 0 & 0 & 0 & \mu & 0 \\ 0 & 0 & 0 & 0 & 0 & \mu \end{bmatrix} \cdot \begin{bmatrix} \varepsilon_{11} \\ \varepsilon_{22} \\ \varepsilon_{33} \\ \gamma_{23} \\ \gamma_{13} \\ \gamma_{12} \end{bmatrix} \quad (3.10)$$

Ou em função do Módulo de Elasticidade Longitudinal (E) e do coeficiente de Pois-

son (ν):

$$\begin{bmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{33} \\ \sigma_{23} \\ \sigma_{31} \\ \sigma_{12} \end{bmatrix} = \frac{E}{(1+\nu)(1-2\nu)} \begin{bmatrix} 1-\nu & \nu & \nu & 0 & 0 & 0 \\ \nu & 1-\nu & \nu & 0 & 0 & 0 \\ \nu & \nu & 1-\nu & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1-2\nu}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1-2\nu}{2} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1-2\nu}{2} \end{bmatrix} \cdot \begin{bmatrix} \varepsilon_{11} \\ \varepsilon_{22} \\ \varepsilon_{33} \\ \gamma_{23} \\ \gamma_{13} \\ \gamma_{12} \end{bmatrix} \quad (3.11)$$

A relação inversa da equação (3.5) é dada por:

$$\varepsilon_{ij} = -\frac{\lambda}{2\mu} \delta_{ij} \varepsilon_{kk} + \frac{\sigma_{ij}}{2\mu} \quad (3.12)$$

Onde ε_{kk} passa a ser escrita em função de σ_{kk} :

$$\begin{aligned} \sigma_{kk} &= \sigma_{11} + \sigma_{22} + \sigma_{33} \\ \Leftrightarrow \sigma_{kk} &= 3\lambda \varepsilon_{kk} + 2\mu (\varepsilon_{11} + \varepsilon_{22} + \varepsilon_{33}) \\ \Leftrightarrow \sigma_{kk} &= (3\lambda + 2\mu) \varepsilon_{kk} \\ \Leftrightarrow \varepsilon_{kk} &= \frac{\sigma_{kk}}{(3\lambda + 2\mu)} \end{aligned} \quad (3.13)$$

Que substituindo a equação (3.13) na expressão (3.12) tem-se:

$$\varepsilon_{ij} = -\frac{\lambda}{2\mu (3\lambda + 2\mu)} \delta_{ij} \sigma_{kk} + \frac{\sigma_{ij}}{2\mu} \quad (3.14)$$

Em função das propriedades do material, a expressão (3.14) toma a seguinte forma simplificada:

$$\varepsilon_{ij} = -\frac{\nu}{E} \delta_{ij} \sigma_{kk} + \frac{1+\nu}{E} \sigma_{ij} \quad (3.15)$$

Em que as seis componentes independentes do tensor das deformações são determinadas seguindo a seguinte notação matricial:

$$\begin{bmatrix} \varepsilon_{11} \\ \varepsilon_{22} \\ \varepsilon_{33} \\ \gamma_{23} \\ \gamma_{31} \\ \gamma_{12} \end{bmatrix} = \frac{1}{E} \begin{bmatrix} 1 & -\nu & -\nu & 0 & 0 & 0 \\ -\nu & 1 & -\nu & 0 & 0 & 0 \\ -\nu & -\nu & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2(1+\nu) & 0 & 0 \\ 0 & 0 & 0 & 0 & 2(1+\nu) & 0 \\ 0 & 0 & 0 & 0 & 0 & 2(1+\nu) \end{bmatrix} \cdot \begin{bmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{33} \\ \sigma_{23} \\ \sigma_{13} \\ \sigma_{12} \end{bmatrix} \quad (3.16)$$

3.5 Formulação Forte e Fraca do Problema

Sendo a obtenção da solução analítica o ideal na resolução de um problema físico governado por equações diferenciais, o que se verifica é que em diversas situações a sua obtenção é de enorme complexidade dada a impossibilidade da descrição matemática do problema. Daí se recorrer a métodos numéricos com o objectivo de se obter uma solução aproximada da solução analítica, em pontos discretos de um dado conjunto. Como métodos numéricos destaca-se o Método das Diferenças Finitas (MDF), o Método Variacional e o Método dos Elementos Finitos (MEF). Ao contrário do que acontece com o MDF, que procura obter a solução exacta do problema originando pequenos erros na verificação do equilíbrio e sem modos de os contornar (aproxima as derivadas nas equações diferenciais que governam o problema utilizando equações de diferença), o MEF procura obter uma solução aproximada que reduza a um mínimo o erro da aproximação. Assim sendo o MEF baseia-se em dois principais conceitos que o ajudam a contornar os problemas dos erros:

- Solução baseada num integral da equação.
- Aproximação da solução por uma função definida num subdomínio resultante da discretização do domínio.

O método de minimização do erro mais comum é o método dos resíduos pesados que impõe a verificação do equilíbrio em todos os pontos do domínio numa condição média (erro médio pesado nulo em todos os pontos do domínio). Deste método existem diversas versões que variam consoante a função de peso escolhida.

Assim sendo, seja Ω o domínio do nosso problema de elasticidade linear tridimensional, correspondendo ao volume do sistema estrutural a considerar, de fronteira

Γ . Se à Equação de Equilíbrio Dinâmico, Eq. (3.2), adicionarmos as condições de Dirichlet, tem-se a formulação forte do problema:

$$\begin{cases} \rho \ddot{u}_i - \sigma_{ij,j} = F_i & \text{em } \Omega, t \in [0; \infty[\\ u_i = g_i & \text{sobre } \Gamma = \partial\Omega \end{cases} \quad (3.17)$$

Que numa notação matricial é dado por:

$$\begin{cases} \rho \ddot{\underline{u}} - \nabla \cdot \underline{\sigma} = \underline{F} & \text{em } \Omega, t \in [0; \infty[\\ \underline{u} = \underline{g} & \text{sobre } \Gamma = \partial\Omega \end{cases} \quad (3.18)$$

Em que

$$\begin{aligned} u : \quad \Omega \times \mathbb{R}_0^+ &\rightarrow \mathbb{R} & \text{com } \Omega \subset \mathbb{R}^3 \\ (x_1, x_2, x_3, t) &\mapsto (u_1, u_2, u_3) \end{aligned}$$

Resolvendo a equação seguindo esta formulação estaremos a aplicar o Método das Diferenças Finitas, o que não faz parte do material de estudo do presente trabalho. Logo para que se possa aplicar o Método dos Elementos Finitos é necessário escrever a equação (3.2) na forma de um integral onde é aplicado o método dos resíduos pesados. Esta formulação, conjuntamente com as condições de fronteira, é designada de formulação fraca do problema de elasticidade linear tridimensional.

A formulação fraca do problema é obtida a partir da sua formulação forte e por aplicação da 2ª fórmula de Green, também conhecida como Igualdade de Green (Integração por Partes). Inicialmente é imposta uma função de teste, v , definida em Ω e que verifica a condição $v = 0$ sobre $\partial\Omega$. Ao multiplicar a equação (3.18) por v e integrá-la em ordem a Ω tem-se:

$$\iiint_{\Omega} \rho \ddot{\underline{u}} \cdot \underline{v} d\Omega - \iiint_{\Omega} \nabla \cdot \underline{\sigma} \cdot \underline{v} d\Omega = \iiint_{\Omega} \underline{F} \cdot \underline{v} d\Omega \quad (3.19)$$

Segundo a Igualdade de Green (Integração por partes)

$$\iiint_{\Omega} \nabla \cdot \underline{\sigma} \cdot \underline{v} \, d\Omega + \iiint_{\Omega} \underline{\sigma} \cdot \nabla \underline{v} \, d\Omega = \iint_{\Gamma} \underline{\sigma} \cdot \underline{n} \cdot \underline{v} \, dS \quad (3.20)$$

Como na fronteira da região, Γ , $v = 0$ logo tem-se:

$$\iiint_{\Omega} \nabla \cdot \underline{\sigma} \cdot \underline{v} \, d\Omega = - \iiint_{\Omega} \underline{\sigma} \cdot \nabla \underline{v} \, d\Omega \quad (3.21)$$

E substituindo na equação (3.19) obtém-se:

$$\iiint_{\Omega} \rho \underline{\ddot{u}} \cdot \underline{v} \, d\Omega + \iiint_{\Omega} \underline{\sigma} \cdot \nabla \underline{v} \, d\Omega = \iiint_{\Omega} \underline{F} \cdot \underline{v} \, d\Omega \quad (3.22)$$

Considerando o caso bidimensional, por motivos simplificativos, o tensor das deformações é dado por:

$$\underline{\varepsilon} = \begin{bmatrix} \varepsilon_{11} & \varepsilon_{12} \\ \varepsilon_{21} & \varepsilon_{22} \end{bmatrix} = \begin{bmatrix} \frac{\partial u_1}{\partial x_1} & \frac{1}{2} \left(\frac{\partial u_1}{\partial x_2} + \frac{\partial u_2}{\partial x_1} \right) \\ \frac{1}{2} \left(\frac{\partial u_2}{\partial x_1} + \frac{\partial u_1}{\partial x_2} \right) & \frac{\partial u_2}{\partial x_2} \end{bmatrix} \quad (3.23)$$

E tendo em conta a lei de Hooke generalizada determinada:

$$\sigma_{ij} = \lambda \delta_{ij} \varepsilon_{kk} + 2\mu \varepsilon_{ij} \quad (3.24)$$

Para o tensor das tensões tem-se:

$$\underline{\sigma} = \begin{bmatrix} \sigma_{11} & \sigma_{12} \\ \sigma_{21} & \sigma_{22} \end{bmatrix} = \begin{bmatrix} \lambda \left(\frac{\partial u_1}{\partial x_1} + \frac{\partial u_2}{\partial x_2} \right) + 2\mu \frac{\partial u_1}{\partial x_1} & \mu \left(\frac{\partial u_1}{\partial x_2} + \frac{\partial u_2}{\partial x_1} \right) \\ \mu \left(\frac{\partial u_2}{\partial x_1} + \frac{\partial u_1}{\partial x_2} \right) & \lambda \left(\frac{\partial u_2}{\partial x_2} + \frac{\partial u_1}{\partial x_1} \right) + 2\mu \frac{\partial u_2}{\partial x_2} \end{bmatrix} \quad (3.25)$$

Deste modo a seguinte igualdade é verificada:

$$\begin{aligned}
\sigma \cdot \nabla \underline{v} &= \lambda \left(\frac{\partial u_1}{\partial x_1} + \frac{\partial u_2}{\partial x_2} \right) \left(\frac{\partial v_1}{\partial x_1} + \frac{\partial v_2}{\partial x_2} \right) + 2\mu \left(\frac{\partial u_1}{\partial x_1} \frac{\partial v_1}{\partial x_1} + \frac{\partial u_2}{\partial x_2} \frac{\partial v_2}{\partial x_2} \right) \\
&\quad + \mu \left(\frac{\partial u_1}{\partial x_2} + \frac{\partial u_2}{\partial x_1} \right) \left(\frac{\partial v_1}{\partial x_2} + \frac{\partial v_2}{\partial x_1} \right) \\
&= \lambda \operatorname{div} \underline{u} \operatorname{div} \underline{v} + 2\mu \left[\frac{\partial u_1}{\partial x_1} \frac{\partial v_1}{\partial x_1} + \frac{\partial u_2}{\partial x_2} \frac{\partial v_2}{\partial x_2} \right. \\
&\quad \left. + \frac{1}{2} \left(\frac{\partial u_1}{\partial x_2} \frac{\partial v_1}{\partial x_2} + \frac{\partial u_2}{\partial x_1} \frac{\partial v_1}{\partial x_2} + \frac{\partial u_1}{\partial x_2} \frac{\partial v_2}{\partial x_1} + \frac{\partial u_2}{\partial x_1} \frac{\partial v_2}{\partial x_1} \right) \right] \\
&= \lambda \operatorname{div} \underline{u} \operatorname{div} \underline{v} + 2\mu \varepsilon(\underline{u}) : \varepsilon(\underline{v})
\end{aligned} \tag{3.26}$$

Em que $\varepsilon(\underline{u})$ e $\varepsilon(\underline{v})$ representam o tensor das deformações em função do campo de deslocamentos \underline{u} e campo da função de teste \underline{v} :

$$\varepsilon(\underline{u}) = \begin{bmatrix} \frac{\partial u_1}{\partial x_1} & \frac{1}{2} \left(\frac{\partial u_1}{\partial x_2} + \frac{\partial u_2}{\partial x_1} \right) & \frac{1}{2} \left(\frac{\partial u_1}{\partial x_3} + \frac{\partial u_3}{\partial x_1} \right) \\ \frac{1}{2} \left(\frac{\partial u_2}{\partial x_1} + \frac{\partial u_1}{\partial x_2} \right) & \frac{\partial u_2}{\partial x_2} & \frac{1}{2} \left(\frac{\partial u_2}{\partial x_3} + \frac{\partial u_3}{\partial x_2} \right) \\ \frac{1}{2} \left(\frac{\partial u_3}{\partial x_1} + \frac{\partial u_1}{\partial x_3} \right) & \frac{1}{2} \left(\frac{\partial u_3}{\partial x_2} + \frac{\partial u_2}{\partial x_3} \right) & \frac{\partial u_3}{\partial x_3} \end{bmatrix} \tag{3.27}$$

e

$$\varepsilon(\underline{v}) = \begin{bmatrix} \frac{\partial v_1}{\partial x_1} & \frac{1}{2} \left(\frac{\partial v_1}{\partial x_2} + \frac{\partial v_2}{\partial x_1} \right) & \frac{1}{2} \left(\frac{\partial v_1}{\partial x_3} + \frac{\partial v_3}{\partial x_1} \right) \\ \frac{1}{2} \left(\frac{\partial v_2}{\partial x_1} + \frac{\partial v_1}{\partial x_2} \right) & \frac{\partial v_2}{\partial x_2} & \frac{1}{2} \left(\frac{\partial v_2}{\partial x_3} + \frac{\partial v_3}{\partial x_2} \right) \\ \frac{1}{2} \left(\frac{\partial v_3}{\partial x_1} + \frac{\partial v_1}{\partial x_3} \right) & \frac{1}{2} \left(\frac{\partial v_3}{\partial x_2} + \frac{\partial v_2}{\partial x_3} \right) & \frac{\partial v_3}{\partial x_3} \end{bmatrix} \tag{3.28}$$

Logo ao substituir a expressão (3.26) na equação (3.22), e conjuntamente com as condições de Dirichlet, obtém-se a formulação fraca do problema:

$$\left\{ \begin{array}{l} \iiint_{\Omega} \rho \ddot{\underline{u}} \cdot \underline{v} \, d\Omega + \iiint_{\Omega} \lambda \operatorname{div} \underline{u} \operatorname{div} \underline{v} + 2\mu \varepsilon(\underline{u}) : \varepsilon(\underline{v}) \, d\Omega = \iiint_{\Omega} \underline{F} \cdot \underline{v} \, d\Omega \quad \text{em } \Omega \\ \underline{u} = \underline{g} \quad \text{sobre } \Gamma \end{array} \right. \quad (3.29)$$

De um modo simplificador, a equação (3.29) poderá ser escrita como:

$$\left\{ \begin{array}{l} \iiint_{\Omega} \rho \ddot{\underline{u}} \cdot \underline{v} \, d\Omega + a(\underline{u}, \underline{v}) = F(\underline{v}) \quad \text{em } \Omega \\ \underline{u} = \underline{g} \quad \text{sobre } \Gamma \end{array} \right. \quad (3.30)$$

Para todo o instante pertencente ao intervalo $t \in [0; \infty[$, com

$$a(\underline{u}, \underline{v}) = \iiint_{\Omega} \lambda \operatorname{div} \underline{u} \operatorname{div} \underline{v} + 2\mu \varepsilon(\underline{u}) : \varepsilon(\underline{v}) \, d\Omega$$

Onde $\varepsilon(\underline{u}) : \varepsilon(\underline{v})$ representa o duplo produto interno entre os tensores, definido matricialmente como o traço $\varepsilon(\underline{u}) : \varepsilon(\underline{v}) = \operatorname{Tr} [\varepsilon(\underline{v})^T + \varepsilon(\underline{u})]$

e

$$F(\underline{v}) = \iiint_{\Omega} \underline{F} \cdot \underline{v} \, d\Omega$$

Capítulo 4

Soluções Analíticas

Algumas soluções teóricas poderão ser facilmente deduzidas, como é o caso do problema de uma membrana vibrante, utilizando o método de separação de variáveis e as séries de Fourier [1]. Deste modo seja $\Omega = (0, a) \times (0, b)$ e $f, g : [0, a] \times [0, b] \rightarrow \mathbb{R}$ tem-se o seguinte problema:

$$\left\{ \begin{array}{ll} u_{tt} = u_{xx} + u_{yy} & \text{se } 0 < x < a, 0 < y < b \text{ e } t \geq 0 \\ u(x, 0, t) = u(x, b, t) = u(0, y, t) = u(a, y, t) = 0 & \text{se } 0 \leq x \leq a, 0 \leq y \leq b \text{ e } t \geq 0 \\ u(x, y, 0) = f(x, y) & \text{se } 0 \leq x \leq a \text{ e } 0 \leq y \leq b \\ u_t(x, y, 0) = g(x, y) & \text{se } 0 \leq x \leq a \text{ e } 0 \leq y \leq b \end{array} \right. \quad (4.1)$$

A partir do método de separação de variáveis iremos tentar encontrar uma solução para o problema que seja dado pelo produto de três funções, cada uma dependendo apenas de uma das variáveis x, y ou t :

$$u(x, y, t) = F(x) G(y) H(t) \quad (4.2)$$

Tendo

$$u_{tt} = F(x) G(y) H''(t)$$

$$u_{xx} = F''(x) G(y) H(t)$$

$$u_{yy} = F(x) G''(y) H(t)$$

Substituindo na equação (4.1) tem-se:

$$F(x) G(y) H''(t) = F''(x) G(y) H(t) + F(x) G''(y) H(t)$$

$$\Leftrightarrow \frac{H''(t)}{H(t)} = \frac{F''(x)}{F(x)} + \frac{G''(y)}{G(y)} = s$$

O que também podemos escrever:

$$\frac{F''(x)}{F(x)} = -\frac{G''(y)}{G(y)} + s = r \quad (4.3)$$

Deste modo consegue-se chegar às seguintes equações diferenciais:

$$F''(x) - r F(x) = 0$$

$$G''(y) + (r - s) G(y) = 0$$

$$H''(t) - s H(t) = 0$$

Que associadas às respectivas condições de fronteira têm como solução:

$$F_n(x) = \sin\left(\frac{n\pi x}{a}\right) \quad \text{para} \quad r = -\frac{n^2\pi^2}{a^2} \quad (4.4)$$

e

$$G_m(y) = \sin\left(\frac{m\pi y}{b}\right) \quad \text{para} \quad s - r = -\frac{m^2\pi^2}{b^2} \quad (4.5)$$

Onde

$$s = r - \frac{m^2\pi^2}{b^2} = -\pi^2 \left(\frac{n^2}{a^2} + \frac{m^2}{b^2} \right) \quad (4.6)$$

Assim sendo para o problema em t tem-se:

$$H''(t) + \pi^2 \left(\frac{n^2}{a^2} + \frac{m^2}{b^2} \right) H(t) = 0 \quad (4.7)$$

Cuja solução é dada por:

$$H_{nm}(t) = a_{nm} \cos(\lambda_{nm} t) + b_{nm} \sin(\lambda_{nm} t) \quad (4.8)$$

Em que λ_{nm} representa as ditas frequências características da membrana.

$$\lambda_{nm} = \pi \sqrt{\frac{n^2}{a^2} + \frac{m^2}{b^2}} \quad (4.9)$$

Logo por substituição na equação (4.2) tem-se os correspondentes modos normais de vibração da membrana:

$$u_{nm}(x, y, t) = \sin\left(\frac{n\pi x}{a}\right) \sin\left(\frac{m\pi y}{b}\right) (a_{nm} \cos(\lambda_{nm} t) + b_{nm} \sin(\lambda_{nm} t)) \quad (4.10)$$

Os quais irão dar origem à solução do nosso problema:

$$u(x, y, t) = \sum_{n=1}^{\infty} \sum_{m=1}^{\infty} \sin\left(\frac{n\pi x}{a}\right) \sin\left(\frac{m\pi y}{b}\right) (a_{nm} \cos(\lambda_{nm} t) + b_{nm} \sin(\lambda_{nm} t)) \quad (4.11)$$

Onde os coeficientes a_{nm} e b_{nm} são determinados como sendo os coeficientes das séries de Fourier duplas das funções apropriadas.

$$f(x, y) = u(x, y, 0) = \sum_{n=1}^{\infty} \sum_{m=1}^{\infty} a_{nm} \sin\left(\frac{n\pi x}{a}\right) \sin\left(\frac{m\pi y}{b}\right) \quad (4.12)$$

de modo que, estendendo f a uma função periódica ímpar de período $2a$ na variável x e a uma função periódica ímpar de período $2b$ na variável y , obtemos

$$a_{nm} = \frac{4}{ab} \int_0^a \int_0^b f(x, y) \sin\left(\frac{n\pi x}{a}\right) \sin\left(\frac{m\pi y}{b}\right) dx dy \quad (4.13)$$

Do mesmo modo, derivando a série de u em relação a t termo a termo, temos:

$$u_t(x, y, t) = \sum_{n=1}^{\infty} \sum_{m=1}^{\infty} \lambda_{nm} \sin\left(\frac{n\pi x}{a}\right) \sin\left(\frac{m\pi y}{b}\right) (-a_{nm} \sin(\lambda_{nm} t) + b_{nm} \cos(\lambda_{nm} t)) \quad (4.14)$$

e portanto

$$g(x, y) = u_t(x, y, 0) = \sum_{n=1}^{\infty} \sum_{m=1}^{\infty} \lambda_{nm} b_{nm} \sin\left(\frac{n\pi x}{a}\right) \sin\left(\frac{m\pi y}{b}\right) \quad (4.15)$$

logo, procedendo de modo análogo estendendo g a uma função periódica ímpar de período $2a$ na variável x e a uma função periódica ímpar de período $2b$ na variável y , obtemos:

$$b_{nm} = \frac{4}{ab \lambda_{nm}} \int_0^a \int_0^b g(x, y) \sin\left(\frac{n\pi x}{a}\right) \sin\left(\frac{m\pi y}{b}\right) dx dy \quad (4.16)$$

Para o caso de uma laje rectangular simplesmente apoiada podemos encontrar na literatura [10] a solução analítica conhecida como a solução de Navier, obtida por aplicação da análise de Fourier. Assim sendo considerando uma laje rectangular de dimensões $a \times b$, simplesmente apoiada em todo o seu bordo e sujeita a uma carga distribuída $f(x, y)$, Navier apresenta a seguinte solução para os deslocamentos transversais:

$$w = \frac{1}{\pi^4 D_f} \sum_{n=1}^{\infty} \sum_{m=1}^{\infty} \frac{a_{nm}}{\left(\frac{n^2}{a^2} + \frac{m^2}{b^2}\right)^2} \sin\left(\frac{n \pi x}{a}\right) \sin\left(\frac{m \pi y}{b}\right) \quad (4.17)$$

Em que D_f é a rigidez à flexão da laje dada por:

$$D_f = \frac{E h^3}{12(1 - \nu^2)}$$

e a_{nm} representa a expansão da função $f(x, y)$ em séries de Fourier.

Como é possível escrever qualquer função por meio de uma série de Fourier, o que Navier fez foi tomar a sobreposição das soluções para infinitos carregamentos, cada um da forma duplamente sinusoidal, como solução da equação geral de Lagrange. Assim sendo podemos representar a carga genérica como sendo:

$$f(x, y) = \sum_{n=1}^{\infty} \sum_{m=1}^{\infty} a_{nm} \sin\left(\frac{n \pi x}{a}\right) \sin\left(\frac{m \pi y}{b}\right) \quad (4.18)$$

com

$$a_{nm} = \frac{4}{ab} \int_0^a \int_0^b f(x, y) \sin\left(\frac{n \pi x}{a}\right) \sin\left(\frac{m \pi y}{b}\right) dx dy \quad (4.19)$$

Para o caso de carga uniformemente distribuída de intensidade f_0 , é possível verificar que os coeficientes da expansão são dados por:

$$a_{nm} = \frac{16 f_0}{\pi^2 n m} \quad (4.20)$$

Substituindo na equação (4.17) a solução para os deslocamentos transversais de uma laje simplesmente apoiada sujeita a uma carga uniformemente distribuída é dada por:

$$w = \frac{1}{\pi^4 D_f} \sum_{n=1}^{\infty} \sum_{m=1}^{\infty} \frac{16 f_0}{\pi^2 n m \left(\frac{n^2}{a^2} + \frac{m^2}{b^2} \right)^2} \sin \left(\frac{n \pi x}{a} \right) \sin \left(\frac{m \pi y}{b} \right) \quad (4.21)$$

No nosso problema dinâmico, fixando o tempo reduzimos o nossa solução a duas variáveis (x e y), podendo assim aplicar a solução de Navier ao nosso caso de estudo de uma laje simplesmente apoiada. Deste modo tem-se mais um elemento de comparação aos resultados obtidos pelo FreeFEM++.

Capítulo 5

Resolução Numérica de Problemas Dinâmicos de Elasticidade Linear

5.1 Método dos Elementos Finitos

O Método de Elementos Finitos tem como principal objectivo permitir a análise de sistemas complexos e/ou irregulares tendo como princípio fundamental que o sistema global é gerado a partir do agrupamento de elementos finitos, sendo estes sistemas mais simples e de fácil resolução. Garantindo que nos pontos nodais comuns aos diversos elementos haja compatibilidade dos deslocamentos e que as forças internas estejam em equilíbrio, o agrupamento destes elementos passa a funcionar como uma entidade única, equivalente ao nosso sistema global.

Na discretização do sistema global em elementos finitos estes podem assumir diversas formas sendo normalmente utilizadas triângulos ou rectângulos na análise bidimensional e tetraedros ou paralelepípedos na análise tridimensional. A cada nó do elemento estão associados um número de graus de liberdade que correspondem à redução das incógnitas do nosso problema, neste caso o campo de deslocamentos da estrutura, a um número finito. Em toda a sua essência, o procedimento de discretização do Método dos Elementos Finitos considera os elementos individuais como sendo contínuos pois os deslocamentos em qualquer ponto do elemento contínuo poderá ser obtido a partir de funções de interpolação e dos graus de liberdade do elemento no qual o ponto se encontra localizado. As funções de interpolação, nor-

malmente designadas por funções de forma (N_i), são geralmente funções polinomiais de grau reduzido e iguais para elementos do mesmo tipo, em que para cada grau de liberdade é definida tal função onde toma o valor 1 no respectivo grau de liberdade e o valor nulo nos restantes. De modo a simplificar a sua implementação as funções de forma normalmente são definidas para um elemento de referência ou mestre. A qualidade da solução pode geralmente ser melhorada através do sucessivo refinamento da malha, quer através do aumento do grau das funções de aproximação (refinamento tipo p), quer através do aumento do número de elementos considerados na discretização (refinamento tipo h), sendo este último tipo mais utilizado nestes modelos.

No nosso caso, foi feita uma análise tridimensional e foram utilizados elementos tetraédicos e funções de forma do segundo grau (P2) resultando em 10 nós por elemento em que o vector do campo de deslocamentos é dado por:

$$\underline{u} = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} = N \underline{u}^e \quad (5.1)$$

Em que \underline{u} corresponde ao vector de deslocamentos aproximados no ponto de coordenadas (x, y, z) , para o instante t , N corresponde à matriz das funções de forma e \underline{u}^e corresponde ao vector de deslocamentos dos pontos nodais. A relação da equação (5.1) é normalmente designada de aproximação fundamental do Método dos Elementos Finitos.

Partindo da Equação de Compatibilidade da Teoria de Elasticidade Linear é possível conhecer o estado de deformação em qualquer ponto de um elemento finito:

$$\underline{\varepsilon} = L \underline{u} = L N \underline{u}^e = B \underline{u}^e \quad (5.2)$$

Sendo $\underline{\varepsilon}$ o vector das deformações, L o operador diferencial e B a matriz das derivadas das funções de forma resultante do produto da matriz L pela matriz N .

Deste modo pelas Relações Constitutivas também se torna possível conhecer o estado

de tensão em qualquer ponto de um elemento finito:

$$\underline{\sigma} = D \underline{\varepsilon} = D B \underline{u}^e \quad (5.3)$$

Em que $\underline{\sigma}$ representa o vector das tensões e D representa a matriz constitutiva.

Com base nas equações (5.1), (5.2) e (5.3) e por intermédio do Princípio dos Trabalhos Virtuais (PTV) é possível formular a Equação do Movimento para um sistema com múltiplos graus de liberdade, sendo o nosso problema correspondente a um sistema de múltiplos graus de liberdade (3 graus de liberdade por nó). Pelo PTV "Se um corpo rígido está em equilíbrio, o trabalho realizado pelas forças externas e pelas forças internas ao longo de deslocamentos virtuais é nulo":

$$W_{int} = W_{ext} \quad (5.4)$$

Onde

$$W_{int} = \iiint_V \delta \underline{\varepsilon}^T \underline{\sigma} dV + \iiint_V \delta \underline{u}^T \mu \dot{\underline{u}} dV + \iiint_V \delta \underline{u}^T \rho \ddot{\underline{u}} dV \quad (5.5)$$

$$W_{ext} = \iiint_V \delta \underline{u}^T \underline{F}_V dV + \iint_S \delta \underline{u}^T \underline{F}_S dS + \sum_i^n \delta u_i^T \underline{F}_i \quad (5.6)$$

Logo, para cada elemento finito, pela igualdade dos trabalhos ao longo de um deslocamento virtual tem-se:

$$\begin{aligned} & \iiint_V \delta \underline{u}^{eT} B^T D B \underline{u}^e dV + \iiint_V \delta \underline{u}^{eT} N^T \mu N \dot{\underline{u}}^e dV + \iiint_V \delta \underline{u}^{eT} N^T \rho N \ddot{\underline{u}}^e dV = \\ & \iiint_V \delta \underline{u}^{eT} N^T \underline{F}_V dV + \iint_S \delta \underline{u}^{eT} N^T \underline{F}_S dS + \sum_i^n \delta u_i^{eT} N_i^T \underline{F}_i \\ \Leftrightarrow & \delta \underline{u}^{eT} \left(\iiint_V B^T D B dV \underline{u}^e + \iiint_V \mu N^T N dV \dot{\underline{u}}^e + \iiint_V \rho N^T N dV \ddot{\underline{u}}^e \right) = \\ & \delta \underline{u}^{eT} \left(\iiint_V N^T \underline{F}_V dV + \iint_S N^T \underline{F}_S dS + \sum_i^n N_i^T \underline{F}_i \right) \\ \Leftrightarrow & K^e \underline{u}^e(t) + C^e \dot{\underline{u}}^e(t) + M^e \ddot{\underline{u}}^e(t) = \underline{F}^e(t) \end{aligned}$$

Em que K^e é a matriz de rigidez elementar, C^e é a matriz de amortecimento elementar, M^e é a matriz de massas elementar e $\underline{u}^e(t)$, $\underline{\dot{u}}^e(t)$ e $\underline{\ddot{u}}^e(t)$ são os vectores de deslocamentos, velocidades e acelerações, respectivamente, dos nós dos elementos. ρ representa a densidade de massa do elemento e μ representa o parâmetro de amortecimento do elemento. F_V , F_S e F_i correspondem aos vectores de forças externas volumétricas, superficiais e pontuais, respectivamente, aplicadas ao elemento.

Fazendo a sobreposição de todas as componentes das matrizes (assemblagem) e dos vectores tem-se a Equação do Movimento global de toda a estrutura:

$$K \underline{u}(t) + C \underline{\dot{u}}(t) + M \underline{\ddot{u}}(t) = \underline{F}(t) \quad (5.7)$$

O processo de sobreposição das matrizes e dos vectores dos elementos é designado por método directo de rigidez. É de se salientar que as matrizes M , K e C não são dependentes do tempo sendo constantes ao longo do mesmo. Geralmente a matriz de amortecimento global C não é obtida pela assemblagem da matriz de amortecimento dos elementos mas sim através de uma combinação linear de potências das matrizes M e K do sistema global:

$$C = a_0 K^r + a_1 M^s \quad (5.8)$$

Onde a_0 e a_1 são valores reais e r e s são valores inteiros. Se r e s forem igual a um diz-se que o amortecimento é proporcional e viscoso.

Nos estudos realizados no âmbito desta dissertação não é considerado o amortecimento e é apenas considerado forças de superfície como forças externas de excitação.

5.2 Método de Integração Directa

Como foi demonstrado anteriormente, ao contrário do que acontece numa análise estática, na formulação matemática de um problema dinâmico de elasticidade linear tridimensional teremos a componente dependente do tempo que corresponde às forças de inércia. Em termos de equilíbrio dinâmico das forças, na sua formulação fraca, corresponde ao termo:

$$\iiint_{\Omega} \rho \ddot{\underline{u}} \cdot \underline{v} d\Omega \quad (5.9)$$

Para avaliação aproximada do problema, por aplicação do Método dos Elementos Finitos, existem dois métodos numéricos disponíveis: o método de sobreposição modal (possibilidade de aplicação do princípio de sobreposição de efeitos graças às hipóteses de linearidade física e geométrica admitidas) e os métodos de integração directa, que por sua vez se dividem em métodos implícitos e métodos explícitos. Na sua essência, o método numérico de integração directa baseia-se em duas ideias:

- Em vez de tentar satisfazer a equação do movimento em todo tempo t , pretende-se satisfazê-la apenas em intervalos discretos de tempo Δt .
- A segunda ideia consiste em variar os deslocamentos e acelerações dentro do intervalo de tempo Δt adoptado.

No âmbito desta dissertação, o destaque será para os métodos de integração directa uma vez que objectivo principal reside na determinação do histórico-temporal da resposta do sistema estrutural em estudo.

Deste modo a resposta da estrutura é determinada para cada instante de tempo discreto, t_k , distanciados por um intervalo de tempo Δt_k , podendo ser de valor constante ($\Delta t_k = \Delta t$) ou não em que:

$$\Delta t_k = t_k - t_{k-1}$$

Sob a condição de equilíbrio em qualquer instante e conhecidas as condições iniciais o que se tem é:

$$\iiint_{\Omega} \rho \ddot{\underline{u}}^k \cdot \underline{v} d\Omega + a(\underline{u}^k, \underline{v}) = \underline{F}^k(\underline{v}) \quad (5.10)$$

Com

$$\underline{u}^0 = \underline{u}(0) \quad \text{e} \quad \dot{\underline{u}}^0 = \dot{\underline{u}}(0) \quad (5.11)$$

Em que $\underline{u}(0)$ e $\underline{v}(0)$ representam o vector de deslocamentos e de velocidades, respectivamente, no instante inicial.

Logo o procedimento será determinar a resposta de deslocamentos da estrutura num instante u^k com base na resposta obtida no instante anterior u^{k-1} . Assim a análise em cada instante é independente uma da outra não havendo a necessidade de combinação das contribuições das diferentes análises. E este procedimento pode seguir três metodologias distintas:

- Baseando-se na interpolação da função de excitação;
- Baseando-se em expressões de diferenças finitas de velocidade e aceleração;
- Baseando-se em variação assumida da aceleração.

Como foi referido anteriormente os vários métodos de integração directa são classificadas em:

- Implícito: $u^k = f(\dot{u}^k, \ddot{u}^k, u^{k-1}, \dot{u}^{k-1}, \dots)$
- Explícito: $u^k = f(u^{k-1}, \dot{u}^{k-1}, \ddot{u}^{k-1}, u^{k-2}, \dots)$

A eficiência do método, ou seja o esforço computacional exigido para se conseguir o nível de precisão desejado, constitui o principal factor de selecção do método de integração directa a implementar. Portanto a precisão não poderá ser o único critério de selecção do método a utilizar visto que a precisão desejada poderá ser adquirida reduzindo o intervalo de tempo entre os instantes de cálculo.

Na sequência deste trabalho serão apresentados e aplicados dois métodos distintos cada um pertencente a um grupo diferente (implícito ou explícito). Como método implícito será aplicado o bastante conhecido método de Newmark e como método explícito será aplicado o método de Euler.

5.2.1 Método de Euler

Para uma sequência discreta de instantes afastados por um intervalo constante Δt :

$$t_k = t_0 + k \Delta t$$

E sendo o campo de deslocamentos uma função dependente do tempo, $u = u(t_k)$, tem-se:

$$u(t_k) = u(t_0 + k \Delta t) = u^k$$

Por definição a solução analítica para o campo de velocidades é dada:

$$\dot{u}(t_k) = \frac{du(t_k)}{dt} = \lim_{\Delta t \rightarrow 0} \frac{u(t_k) - u(t_k - \Delta t)}{\Delta t} = \lim_{\Delta t \rightarrow 0} \frac{u(t_k) - u(t_{k-1})}{\Delta t}$$

O método de Euler consiste em admitir que, para um intervalo de tempo suficientemente pequeno, a velocidade se aproxima da velocidade média dada pela média aritmética das velocidades no início e no fim do intervalo:

- Aproximação numérica de 1ª ordem

$$\frac{du(t_k)}{dt} \approx \frac{\Delta u(t_k)}{\Delta t} = \frac{u(t_k) - u(t_k - \Delta t)}{\Delta t} = \frac{u^k - u^{k-1}}{\Delta t}$$

- Aproximação numérica de 2ª ordem

$$\frac{d^2 u(t_k)}{dt^2} \approx \frac{\frac{du(t_k)}{dt} - \frac{du(t_k - \Delta t)}{dt}}{\Delta t} = \frac{u(t_k) - 2u(t_k - \Delta t) + u(t_k - 2\Delta t)}{\Delta t^2} = \frac{u^k - 2u^{k-1} + u^{k-2}}{\Delta t^2}$$

Deste modo substituindo a expressão anterior correspondente ao campo de acelerações na equação (5.10) tem-se:

$$\iiint_{\Omega} \rho \left(\frac{\underline{u}^k - 2\underline{u}^{k-1} + \underline{u}^{k-2}}{\Delta t^2} \right) \cdot \underline{v} d\Omega + a(\underline{u}^k, \underline{v}) = \underline{F}^k(\underline{v}) \quad (5.12)$$

E de forma simplificada a equação (5.12) pode ser escrita como:

$$\iiint_{\Omega} \rho \underline{u}^k \cdot \underline{v} d\Omega + a(\underline{u}^k, \underline{v}) \Delta t^2 = \underline{F}^k(\underline{v}) \Delta t^2 + \iiint_{\Omega} 2\rho \underline{u}^{k-1} \cdot \underline{v} d\Omega - \iiint_{\Omega} \rho \underline{u}^{k-2} \cdot \underline{v} d\Omega \quad (5.13)$$

É um método bastante simples em que quanto menor for o valor do Δt melhores resultados serão obtidos em termos de aproximação. É recomendado [4] que este seja inferior a $\frac{T}{\pi}$ para cargas dinâmicas periódicas.

5.2.2 Método de Newmark

Como foi visto um sistema de equações diferenciais de segunda ordem em dinâmica estrutural, considerando a existência de alguma excitação $F(t)$ externa aplicada no sistema ou mesmo uma condição inicial de deslocamento e velocidade não nulos, pode ser resolvido por diversos métodos. Entre estes, o método de Newmark é um dos mais versáteis e populares. Considerando a equação do movimento do sistema:

$$M \ddot{\underline{u}}(t) + C \dot{\underline{u}}(t) + K \underline{u}(t) = \underline{F}(t) \quad (5.14)$$

Em que $\ddot{\underline{u}}(t)$, $\dot{\underline{u}}(t)$, $\underline{u}(t)$ e $F(t)$ representam os vectores de aceleração, velocidade, deslocamento e excitação, respectivamente, e M , C e K representam as matrizes de massa, amortecimento e rigidez, respectivamente.

O método consiste em admitir que há uma variação linear da aceleração do sistema. Esta variação é representada através dos parâmetros numéricos de Newmark α e β que visam atribuir exactidão e estabilidade ao método. Newmark originalmente propôs o esquema conhecido como aceleração média constante em que neste caso $\alpha = \frac{1}{2}$ e $\beta = \frac{1}{6}$. Mas outros esquemas podem ser utilizados como por exemplo $\alpha = \frac{1}{2}$ e $\beta = \frac{1}{4}$, o qual foi adoptado nesta dissertação.

Na literatura existem diversos algoritmos do método de integração de Newmark sendo que para este trabalho foi considerado o seguinte [18]

$$\underline{u}^{k+1} = \left[K + \left(\frac{1}{\beta \Delta t^2} \right) M \right]^{-1} \times \left[\underline{F}^{k+1} + M \left(\frac{1}{\beta \Delta t^2} \right) \underline{u}^k + M \left(\frac{1}{2\beta} - 1 \right) \ddot{\underline{u}}^k \right] \quad (5.15)$$

$$\ddot{\underline{u}}^{k+1} = \left(\frac{1}{\beta \Delta t^2} \right) \left(\underline{u}^{k+1} - \underline{u}^k \right) + \left(\frac{1}{2\beta} - 1 \right) \ddot{\underline{u}}^k \quad (5.16)$$

Tal como foi mencionado anteriormente neste trabalho não é considerado o amortecimento da estrutura. Caso fosse considerado o algoritmo teria que sofrer algumas alterações.

5.3 Implementação no FreeFEM++

O FreeFEM++ é um programa gratuito desenvolvido por F. Hecht, O. Pironneau e A. Le Hyaric (Université Pierre et Marie Curie, Laboratoire Jacques-Louis Lions) cujo objectivo principal assenta na resolução numérica das equações diferenciais que governam os vários problemas da Física, da Engenharia e da Matemática, baseando-se no Método dos Elementos Finitos (MEF). É um pacote programado inteiramente em linguagem C++, mas que oferece ao utilizador uma linguagem própria de *input* que se aproxima bastante à escrita matemática do problema.

Algumas características do FreeFEM++ que podemos destacar são:

- Descrição do problema a partir da sua formulação fraca.
- Tem a possibilidade de trabalhar com sistemas bidimensionais ou tridimensionais, estáticos ou dinâmicos, lineares ou não-lineares, com múltiplas variáveis e múltiplas equações. Contudo é pedido ao utilizador que especifique um processo iterativo (LU, Cholesky, Crout, CG, GMRES, UMFPACK) que permite reduzir o problema para um campo linear.
- Criação da geometria desejada a partir da definição analítica dos limites fronteiras. A geometria definida será o domínio do problema que é entendida como todo o espaço à esquerda da respectiva fronteira.
- Apresenta um gerador automático de malhas em que a densidade de pontos interiores da malha é proporcional à densidade de pontos na fronteira.
- Também apresenta uma grande variedade de elementos finitos triangulares (lineares, quadráticos, entre outros)
- Consegue gerar ficheiros .txt, .edp, .pos, .gnu, .mesh que facilitam a manipulação dos dados de input e output.

No âmbito do presente trabalho, o estudo debruça-se sobre o problema da elasticidade linear tridimensional de sistemas estruturais, mais precisamente a laje, sujeita a cargas dinâmicas. As soluções do problema serão obtidas a partir de uma avaliação

do histórico temporal da resposta da laje sob aplicação de uma vibração sinusoidal com direcção vertical.

Nas secções seguintes será feita uma breve descrição da estrutura do programa do FreeFEM++ implementada para resolução do nosso problema, em que serão apresentados os principais comandos utilizados. A descrição será acompanhada de um exemplo prático de um dos casos de estudo, uma laje totalmente encastrada.

5.3.1 Descrição do Problema

Como já mencionado, para exemplo prático iremos utilizar um dos casos de estudo, em que temos uma laje totalmente encastrada com vãos de dimensões $4 \times 4 m$ (comprimento, largura) e uma espessura de $20 cm$.

Em termos de material, a laje é constituída por betão $C20/25$ com um Módulo de Elasticidade Longitudinal (E) de $30 GPa$ e um Coeficiente de Poisson (ν) igual a $0,2$.

A laje está sob acção de uma carga vertical sinusoidal (seno) uniformemente distribuída sobre a mesma, apresentando uma amplitude máxima de $3,14 kN/m^2$ e um período (T) de $0,5 s$.

5.3.2 Construção da Malha

O domínio do nosso problema é caracterizado como sendo o volume da estrutura em causa, $\Omega \subset \mathbb{R}^3$, cuja fronteira é dada por toda superfície que delimita tal região, $\Gamma = \partial\Omega$.

Embora a discretização do domínio seja possível de ser obtida directamente pelo FreeFEM++, optou-se por gerar a malha de elementos finitos a partir do software GMSH. Tal escolha baseou-se na maior eficiência demonstrada pelo GMSH tanto a nível da definição da geometria da estrutura como pelo mecanismo de geração das malhas. O GMSH ainda permite a atribuição de etiquetas às fronteiras para que posteriormente o FreeFEM++ possa aplicar as respectivas condições de fronteiras ou cargas de superfície.

Gerada a malha 3D no GMSH, constituída por elementos finitos tetraédricos, esta é guardada em um ficheiro de formato .mesh, o qual o FreeFEM++ consegue importar, atribuindo-lhe simultaneamente um nome, através do comando:

```
mesh3 nome da malha ("ficheiro a importar.mesh");
```

Mas a importação da malha pelo FreeFEM++ só é possível após o pré-carregamento de alguns comandos, que lhe permitem trabalhar com malhas 3D e malhas guardadas na extensão Medit INRIA:

```
load "msh3";  
load "medit";
```

```
// CONSTRUÇÃO DA MALHA_-----  
load "msh3"  
load "medit"  
  
mesh3 Th("dominio.mesh");
```

5.3.3 Definição das Variáveis e Constantes

Todas as constantes e variáveis do problema deverão ser declaradas no FreeFEM++ com a indicação do tipo de numeração (inteiros, reais ou complexos). Para isso será necessário a utilização do comando:

```
int, real ou complex constante = valor;
```

```
// CONSTANTES DO PROBLEMA_-----  
int      n = 128;                // Número de instantes  
  
real     T = 0.5                 // Período da excitação [s]  
         h = 0.03125,           // Intervalo entre instantes [s]  
         sqrt2 = sqrt(2),       // Raíz Quadrada de 2  
         coef = 1.;            // Factor de escala para as deformadas  
  
// CONSTANTES DO MATERIAL_-----
```

```

real    E = 30e6,                // Módulo de Young [kN/m2]
        nu = 0.2,                // Coeficiente de Poisson
        lambda = E*nu/((1+nu)*(1-2*nu)), // Constantes de Lamé
        mu = E/(2*(1+nu)),        (lambda e mu) [kN/m2]
        gamma = 25.,            // Peso Volúmico [kN/m3]
        rho = gamma/9.80665     // Massa Volúmica [T/m3]

```

5.3.4 Definição das Macros

O FreeFEM++ ainda fornece ao utilizador a possibilidade de criar macros com o intuito de facilitar a escrita de expressões relativamente extensas. O objectivo destas macros será transformar um operador específico numa função genérica. Ainda é de salientar que as variáveis aos quais se aplica a macro poderão ser diferentes das utilizadas na sua definição. Para tal basta definir o seguinte comando:

```

macro operador (função);

```

Para resolução do nosso problema utilizaram-se duas Macros:

```

// MACROS-----
macro epsilon(u1,u2,u3) [dx(u1),dy(u2),dz(u3), (dz(u2)+dy(u3))/sqrt2,
                        (dz(u1)+dx(u3))/sqrt2, (dy(u1)+dx(u2))/sqrt2]
// EOM

```

```

macro div(u1,u2,u3) ( dx(u1)+dy(u2)+dz(u3) ) // EOM

```

5.3.5 Definição do Espaço de Aproximação

Para a utilização do Método dos Elementos Finitos é necessário definir um espaço de funções admissíveis, onde será determinada a solução. Geralmente este é o espaço de funções contínuas em todo o domínio e polinomiais de grau n (P1, P2 ou P3) sobre cada elemento. Estas funções, designadas por funções de forma (polinomiais de grau n por elemento), são nulas em todos os graus de liberdade à excepção do grau de liberdade a que estão associadas, onde tomam o valor unitário. Assim a solução dos problemas estudados será uma combinação linear de todas as funções

de forma definidas no espaço de funções admissíveis. Para definição do espaço e das funções do espaço será necessário a utilização dos seguintes comandos:

```
fespace nome do espaço (nome da malha associada, grau do polinómio);
           nome do espaço funções do espaço;
```

```
// DEFINIÇÃO DO ESPAÇO DE APROXIMAÇÃO _____
fespace Vh(Th,P2);
Vh u1, u2, u3, v1, v2, v3, ua1, ua2, ua3, ub1, ub2, ub3;
```

5.3.6 Resolução do Problema

As forças exteriores aplicadas ao nosso sistema estrutural poderão ser declaradas como sendo uma função definida por uma expressão analítica utilizando o seguinte comando:

```
func nome da função = expressão analítica;
```

Enquanto que as condições iniciais são definidas atribuindo um valor às componentes do vector de deslocamentos para o instante $(i - 1)$ e $(i - 2)$.

A solução do nosso problema será obtida pela resolução da equação de equilíbrio dinâmico na sua formulação fraca com aplicação das respectivas condições de fronteira de Dirichlet. Esta é feita com recurso ao comando:

```
solve nome do problema (incógnitas, solver = processo iterativo)
      = formulação fraca da equação diferencial
      + on (fronteira, condições de fronteira de Dirichlet);
```

Tratando-se de um problema dinâmico as soluções do problema serão obtidas seguindo uma discretização temporal, i , distanciadas por um passo h . Portanto implementou-se um ciclo "for" onde em cada instante de cálculo, i , deverá executar repetidamente um conjunto de acções:

- Primeiramente, resolver o problema determinando a sua solução em todos os graus de liberdade da malha tetraédrica;

- De seguida, armazenar as coordenadas dos pontos nodais dos elementos tetraédricos e os respectivos valores de deslocamentos dos graus de liberdade (Este passo é opcional dependendo do tipo de tratamento dos resultados a executar);
- Posteriormente fazer a actualização das condições iniciais;
- E então avançar para o instante seguinte, $(i + 1)$.

```
// RESOLUÇÃO DO PROBLEMA EVOLUTIVO_____
func fd = -3,14*sin(i*pi/2);    // Força Sinusoidal aplicada
                                segundo a direcção 3 (vertical)

ua1[] = 0.;                    // ua Deslocamento no instante (i-1)
ua2[] = 0.;                    // nas direcções 1, 2 e 3
ua3[] = 0.;

ub1[] = 0.;                    // ub Deslocamento no instante (i-2)
ub2[] = 0.;                    // nas direcções 1, 2 e 3
ub3[] = 0.;

for (int i=1; i<=n; i++)      // i instante
{
solve Problema(u1, u2, u3, v1, v2, v3, solver = CG)
    = int3d(Th)( lambda*div(u1,u2,u3)*div(v1,v2,v3)*h*h
    + 2.*mu*(epsilon(u1,u2,u3)'*epsilon(v1,v2,v3))*h*h )
    - int2d(Th,1500)( fd*v3*h*h )          // Carga Superficial
    + int3d(Th)( ([u1,u2,u3]'*[v1,v2,v3])*ro )
    - int3d(Th)( ([ua1,ua2,ua3]'*[v1,v2,v3])*2*ro )
    + int3d(Th)( ([ub1,ub2,ub3]'*[v1,v2,v3])*ro )
    + on(2500, u1 = 0., u2 = 0., u3 = 0.);

mesh3 Thm = movemesh3(Th, transfo = [x+u1*coef, y+u2*coef, z+u3*coef]);
plot (Thm, wait = 1);
```

```
// ACTUALIZAÇÃO DA SOLUÇÃO-----
ub1[] = ua1[];
ub2[] = ua2[];
ub3[] = ua3[];
ua1[] = u1[];
ua2[] = u2[];
ua3[] = u3[];
}
```

5.3.7 Tratamento e Análise dos Resultados

Para análise dos resultados, o FreeFEM++ oferece-nos dois tipos de visualizações. Uma visualização gráfica permitindo-nos obter as imagens da malha deformada ou mesmo as imagens da distribuição de esforços e de tensões ao longo da estrutura. E uma visualização alfanumérica para obtenção de soluções específicas do problema. As duas visualizações poderão ser obtidas respectivamente a partir dos seguintes comandos:

```
plot (imagem de output, wait = 1, cmm = "nome da imagem",
        ps = "nome do ficheiro exportado.eps");
```

```
cout << "nome da variável ou função" << variável ou função desejada << endl;
```

Mais uma vez recorreu-se ao GMSH, para o presente trabalho, desta vez para se conseguir a visualização da malha deformada. Esta opção é apoiada pelo facto das imagens geradas pelo GMSH apresentarem uma melhor qualidade gráfica face às imagens geradas pelo FreeFEM++. Para conseguir esta visualização bastou utilizar os resultados exportados pelo FreeFEM++ para o ficheiro em formato .pos executando a sua importação no GMSH.

```
// GUARDAR RESULTADOS-----
string resul;
resul = "d"+i+".pos";
ofstream file(resul);
```

```

file << "View \" \"<< "DESLOCAMENTOS" << i << " \" {" << endl;

// REGISTO DAS COORDENADAS DOS VÉRTICES DOS TETRAÉDRICOS_____
for (int k=0; k<Th.nt; k++) // k elemento finito tetraédrico
{
file << "VS(";

for (int j=0; j<3; j++) // j vértice do tetraédrico
file << Th[k] [j].x << ","
<< Th[k] [j].y << ","
<< Th[k] [j].z << ",";
file << Th[k] [3].x << ","
<< Th[k] [3].y << ","
<< Th[k] [3].z;
file << "){" << endl;

// REGISTO DOS DESLOCAMENTOS EM CADA VÉRTICE_____
for (int j=0; j<3; j++)
file << u1[] [Vh(k,j)] << ","
<< u2[] [Vh(k,j)] << ","
<< u3[] [Vh(k,j)] << "," << endl;
file << u1[] [Vh(k,3)] << ","
<< u2[] [Vh(k,3)] << ","
<< u3[] [Vh(k,3)] << "};" << endl;
}
file << "};" << endl;

```

Assim sendo ao longo de todo o estudo efectuado, o FreeFEM++ foi utilizado apenas como uma ferramenta de cálculo deixando toda a componente gráfica a cargo do GMSH.

Capítulo 6

Análise do Problema Dinâmico

6.1 Considerações Iniciais

Neste capítulo serão apresentados os casos estudados para testar os métodos implementados em FreeFEM++ para resolução de problemas dinâmicos de Elasticidade Linear. A selecção dos casos analisados teve como critério a utilização dos casos mais correntes em termos de tipologias de apoio de uma laje.

A validação dos resultados obtidos pelo FreeFEM++ é feita com base na comparação com os resultados obtidos no SAP2000 e, quando possível, com Soluções Teóricas determinadas por outros autores. Um aspecto a salientar da comparação dos resultados, é a distinção entre modelos utilizados pelos programas referidos, já que em FreeFEM++ se implementa um modelo tridimensional de elasticidade linear e em SAP2000 se utiliza os modelos de Kirchoff (lajes finas) ou Reissner-Mindlin (lajes espessas), para modelar o mesmo comportamento. Em termos de material admite-se que todos os elementos estruturais são de betão *C 20/25*, com um peso volúmico $\gamma = 25 \text{ kN/m}^3$, um módulo de Young $E = 30 \text{ GPa}$ e um coeficiente de Poisson $\nu = 0,2$. O material será o mesmo em ambos os programas. Como acção dinâmica actuante escolheu-se uma carga vertical de superfície distribuída uniformemente sobre a laje cuja variação é representada pela seguinte função matemática simulando a actividade humana sobre as lajes:

$$F_3(t) = F_0 \times \sin(\omega t) \quad [\text{kN/m}^2]$$

Em que f_0 representa a intensidade máxima da força aplicada, ω representa a frequência angular da excitação e t é o tempo. Para os casos de estudo a força considerou-se uma intensidade máxima de $3,14 \text{ kN/m}^2$, equivalente a 4 pessoas de 80 kg em um metro quadrado, e uma frequência angular igual a 4π , equivalente a um período de $0,5 \text{ s}$ e uma frequência de 2 Hz . A função da força aplicada poderá assumir diversas formas (sinusoidal, série de cossenos, exponencial, entre outros) conforme o tipo de acção mecânica que se pretende implementar. A frequência de excitação atribuída está conforme à gama de valores propostos por diversos autores, para simular uma vibração induzida pelas actividades humanas sobre uma laje. Um dos factores tidos em conta foi o efeito de ressonância. Por isso procurou-se determinar a menor frequência natural de vibração dos casos de estudo de modo a garantir que estas não se aproximavam da frequência de excitação:

	Frequência [Hz]
Caso 1	72,1
Caso 2	39,6
Caso 3	41,5
Caso 4	4,47

Tabela 6.1: Frequências naturais de vibração

A comparação de resultados é feita seguindo dois processos de avaliação:

- Cálculo dos erros, expressos em termos percentuais, resultantes da análise comparativa do deslocamento vertical ou flecha do ponto central da laje (ponto onde ocorre o deslocamento máximo) num tempo total de cálculo igual a 1 s :

$$erro = \left(1 - \frac{\min(R_{FF++}; R_{SAP2000})}{\max(R_{FF++}; R_{SAP2000})} \right) \times 100 \quad (6.1)$$

De certa forma já são expectáveis erros entre os resultados dos dois modelos, embora não significativos, uma vez que o modelo bidimensional do SAP2000 resulta de uma simplificação do modelo tridimensional.

- Apresentação de gráficos comparativos do percurso temporal do ponto central da laje para um tempo total de cálculo igual a 4 s

6.2 Laje Encastrada em todos os bordos

O primeiro caso a ser estudado corresponde a uma laje quadrada com vãos de 4 m em ambas as direcções horizontais e com uma espessura de 20 cm , estando encastrada em todos os bordos (cf. Anexo A.1).

6.2.1 Malha do GMSH e Resultados do FreeFEM++

Neste caso de estudo o modelo físico tridimensional da laje, gerada a partir do programa GMSH, corresponde a um paralelepípedo com as respectivas dimensões $(4 \times 4 \times 0,2\text{ m})$. A discretização é feita através de elementos finitos tetraédricos obtendo um total de 3803 elementos. Visto tratar-se de uma laje com os bordos totalmente encastrados, na atribuição das condições de fronteira impediu-se o deslocamento dos mesmos em todas as direcções: $u_1 = 0, u_2 = 0$ e $u_3 = 0$.

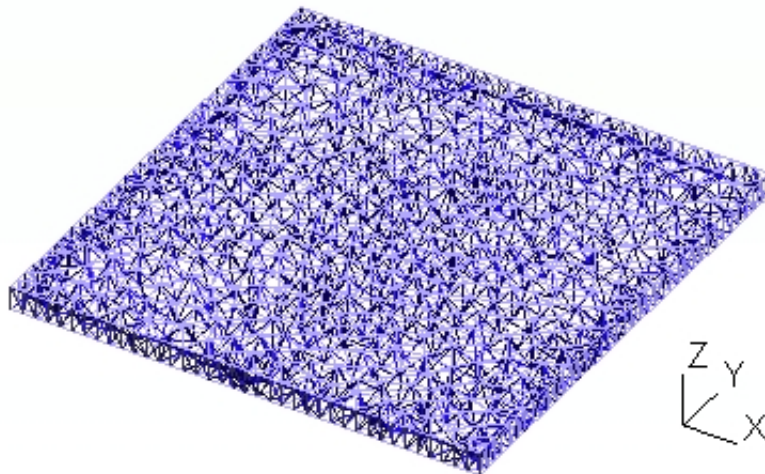


Figura 6.1: Malha do GMSH para a laje encastrada

Uma vez obtido o campo de deslocamentos da malha, é possível visualizar a malha deformada, usando o programa GMSH, com o respectivo valor do deslocamento vertical máximo para um determinado instante. Como exemplo, nas figuras 6.2, 6.3, 6.4 e 6.5 temos as malhas deformadas da laje para os instantes $t = 0,125\text{ s}$ e $t = 0,375\text{ s}$, respectivamente, para os diferentes métodos:

- Pelo Método de Euler

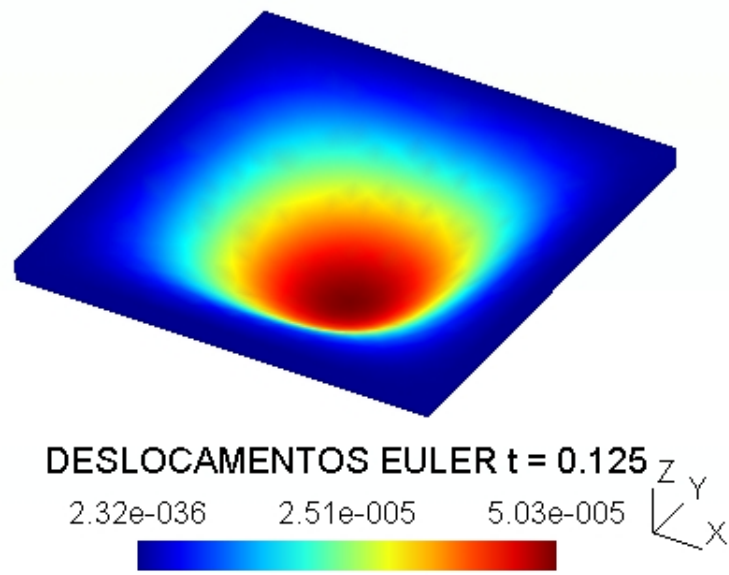


Figura 6.2: Malha Deformada pelo Método de Euler no instante $t = 0,125 s$

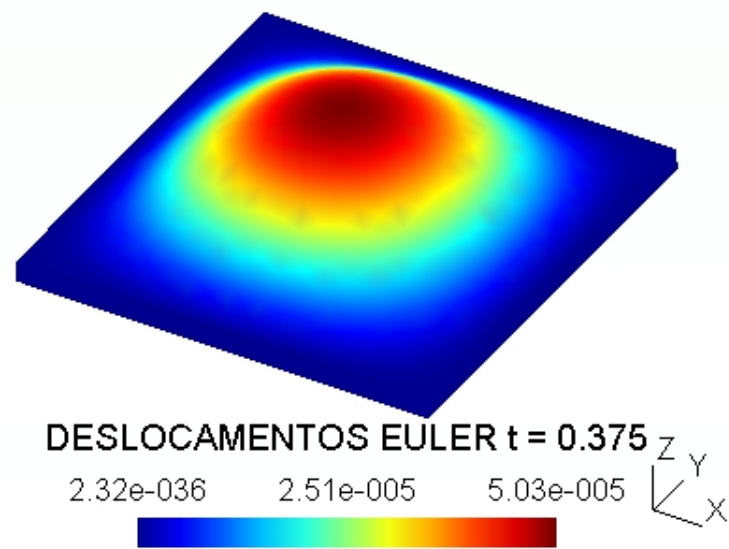


Figura 6.3: Malha Deformada pelo método de Euler no instante $t = 0,375 s$

- Pelo Método de Newmark

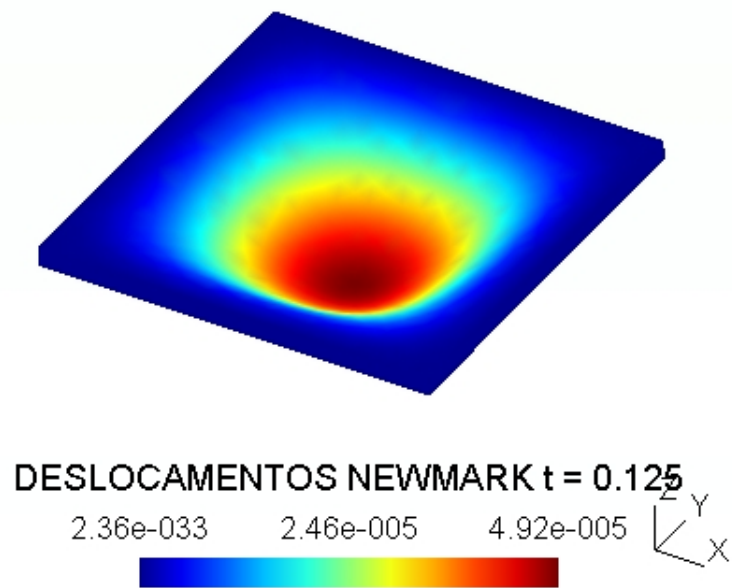


Figura 6.4: Malha Deformada pelo Método de Newmark no instante $t = 0,125$ s

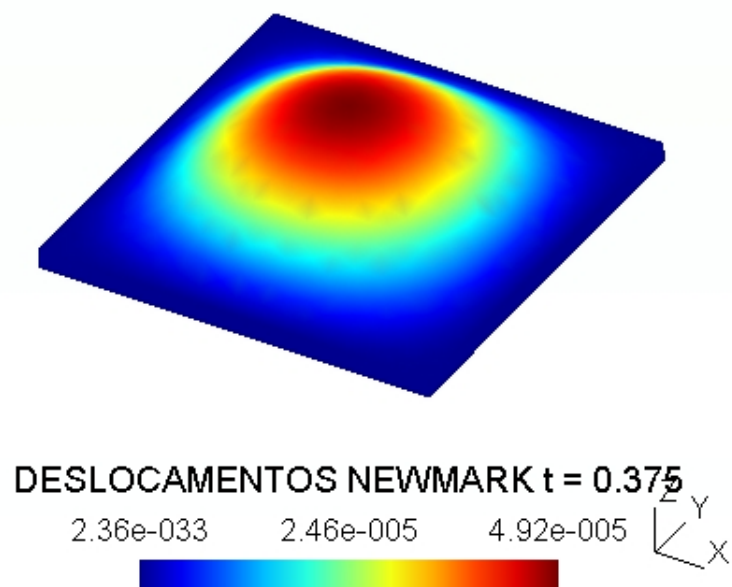


Figura 6.5: Malha Deformada pelo método de Newmark no instante $t = 0,375$ s

6.2.2 Malha e Resultados do SAP2000

No SAP2000 o modelo físico é dado pelo modelo bidimensional de Kirchoff que corresponde a um quadrado com vãos iguais ao modelo tridimensional do FreeFEM++ ($4 \times 4 m$) e discretizada em 400 elementos finitos quadrados ($0,2 \times 0,2 m$). Para simular o encastramento dos bordos foram encastrados todos os nós pertencentes aos mesmos tal como é ilustrado na figura 6.6:

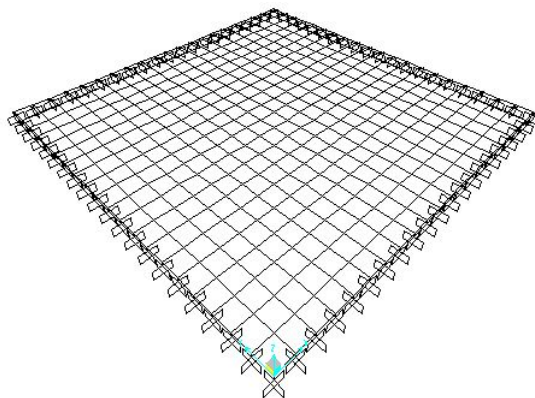
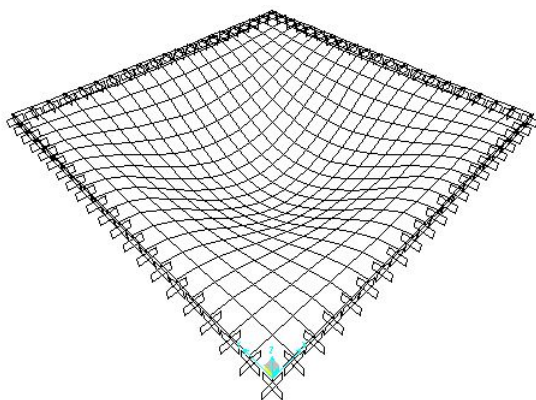


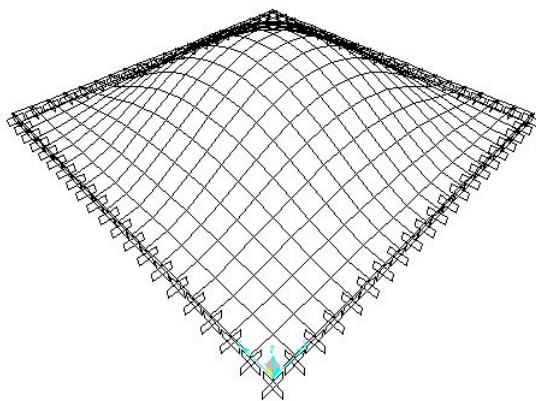
Figura 6.6: Malha do SAP2000 para a laje encastrada

Resolvido o problema, utilizando o método de Newmark, é obtido o campo de deslocamentos cuja malha deformada poderá ser visualizada directamente no programa com o respectivo valor do deslocamento vertical máximo para um determinado instante. Como exemplo, nas figura 6.7 e 6.8 temos as malhas deformadas da laje para os instantes $t = 0,125 s$ e $t = 0,375 s$ respectivamente:



Factor de Escala: 10000

Figura 6.7: Malha Deformada do SAP2000 no instante $t = 0,125 s$



Factor de Escala: 10000

Figura 6.8: Malha Deformada do SAP2000 no instante $t = 0,375 s$

6.2.3 Comparações

A seguir apresentamos a tabela comparativa dos resultados obtidos para o deslocamento vertical do ponto central da laje para um tempo total de cálculo igual a 1 s, discretizada em intervalos de tempo $\Delta t = 0,03125 s$:

Instante	FF++ Euler [mm]	FF++ Newmark [mm]	SAP2000 [mm]	Erro Euler vs. SAP2000	Erro Newmark vs. SAP2000
0,03125	-0,0191	-0,0188	-0,018	5,8	4,3
0,0625	-0,0355	-0,0348	-0,036	1,4	3,3
0,09375	-0,0464	-0,0454	-0,045	3,0	0,9
0,125	-0,0502	-0,0492	-0,05	0,4	1,6
0,15625	-0,0464	-0,0454	-0,044	5,2	3,1
0,1875	-0,0355	-0,0348	-0,036	1,4	3,3
0,21875	-0,0192	-0,0188	-0,017	11,5	9,6
0,25	-1,58E-05	0,0000	-0,001207	-	-
0,28125	0,0192	0,0188	0,021	8,6	10,5

Instante	FF++ Euler [mm]	FF++ Newmark [mm]	SAP2000 [mm]	Erro Euler vs. SAP2000	Erro Newmark vs. SAP2000
0,3125	0,0355	0,0348	0,034	4,2	2,3
0,34375	0,0464	0,0454	0,047	1,3	3,4
0,375	0,0502	0,0492	0,048	4,4	2,4
0,40625	0,0464	0,0454	0,047	1,3	3,4
0,4375	0,0355	0,0348	0,033	7,0	5,2
0,46875	0,0192	0,0188	0,02	4,0	6,0
0,5	1,58E-05	0,0000	-0,001948	-	-
0,53125	-0,0192	-0,0188	-0,018	6,2	4,3
0,5625	-0,0355	-0,0348	-0,037	4,1	5,9
0,59375	-0,0464	-0,0454	-0,044	5,2	3,1
0,625	-0,0502	-0,0492	-0,05	0,4	1,6
0,65625	-0,0464	-0,0454	-0,044	5,2	3,1
0,6875	-0,0355	-0,0348	-0,035	1,4	0,6
0,71875	-0,0192	-0,0188	-0,018	6,2	4,3
0,75	-1,58E-05	0,0000	0,0005113	-	-
0,78125	0,0192	0,0188	0,019	1,0	1,1
0,8125	0,0355	0,0348	0,036	1,4	3,3
0,84375	0,0464	0,0454	0,045	3,0	0,9
0,875	0,0502	0,0492	0,05	0,4	1,6
0,90625	0,0464	0,0454	0,044	5,2	3,1
0,9375	0,0355	0,0348	0,036	1,4	3,3
0,96875	0,0192	0,0188	0,017	11,5	9,6
1	-1,58E-05	0,0000	-0,0001452	-	-

Tabela 6.2: Tabela Comparativa de resultados do deslocamento vertical do ponto central da laje encastrada

Nota: (-) comparação de diferentes gamas de zeros

Da análise da tabela 6.2, em termos de deslocamentos, podemos observar que os erros variam de instante para instante, sendo que pelo método de Euler o menor erro verificado ocorre no instante $t = 0,375 s$, com um valor de 0,4 %, e o maior ocorre no instante $t = 0,21875 s$, com um valor de 11,5 %, enquanto que pelo método de Newmark o menor erro ocorre no instante $t = 0,6875 s$, com o valor de 0,6 % e o maior ocorre no instante $t = 0,28125 s$, com o valor de 10,5 %. De uma forma geral temos erros na gama dos 1 %, 3 %, 4 % e 6 %, sendo que o instante $t = 0,21875 s$ no método de Euler representa o único instante em que o valor do erro ultrapassa a marca dos 10 %. Logo pode-se dizer que há uma certa compatibilidade entre os resultados obtidos pelo diferentes programas. Como foi referido inicialmente no início deste capítulo, estes erros poderão ter origem na distinção dos modelos utilizados pelo FreeFEM++ (modelo tridimensional) e pelo SAP2000 (modelo de Kirchoff). Uma justificação pelos erros do método de Euler serem maiores aos do método de Newmark prende-se ao facto de que no SAP2000 é utilizado o método de Newmark.

Para observar a resposta em deslocamentos do ponto central da laje ao longo do tempo é apresentado o seguinte gráfico:

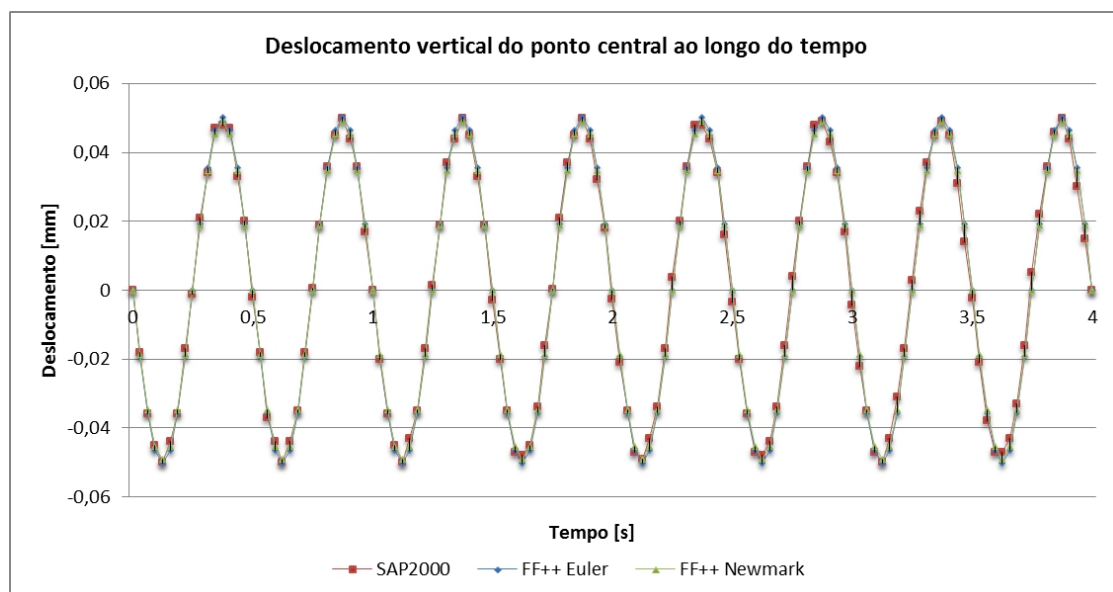


Figura 6.9: Gráfico da resposta em deslocamento do ponto central da laje encastrada

Pela observação do gráfico 6.9 é possível constatar a grande aproximação dos resultados obtidos pelos dois programas (FreeFEM++ e SAP2000), que se traduz pela quase sobreposição das curvas.

6.3 Laje Simplesmente Apoiada

O segundo caso a ser estudado corresponde a uma laje quadrada com vãos de 4 m em ambas as direcções horizontais e com uma espessura de 20 cm , estando todos os bordos apoiados sobre vigas com elevada rigidez à flexão (cf. Anexo A.2).

6.3.1 Malha do GMSH e Resultados do FreeFEM++

Neste caso de estudo o modelo físico tridimensional da laje, gerada a partir do programa GMSH, corresponde a um paralelepípedo com as respectivas dimensões ($4 \times 4 \times 0,2\text{ m}$). A discretização é feita através de elementos finitos tetraédricos obtendo um total de 3926 elementos. Na atribuição das condições de fronteira impediu-se apenas os deslocamentos verticais dos bordos, $u_3 = 0$, permitindo deste modo a rotação em torno dos mesmos (torção).

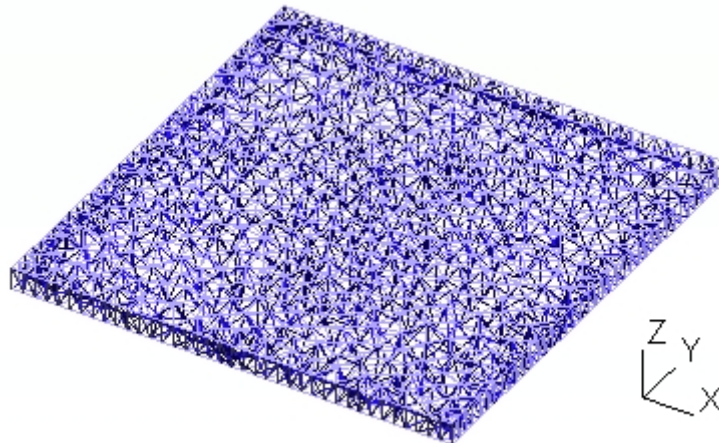


Figura 6.10: Malha do GMSH para a laje simplesmente apoiada

A partir da solução numérica do problema posto, visualizamos com GMSH o campo de deslocamentos da malha e por sua vez a malha deformada com o respectivo valor do deslocamento vertical máximo para um determinado instante. Como exemplo,

nas figuras 6.11, 6.12, 6.13 e 6.14 temos as malhas deformadas da laje para os instantes $t = 0,125\text{ s}$ e $t = 0,375\text{ s}$ respectivamente:

- Pelo Método de Euler

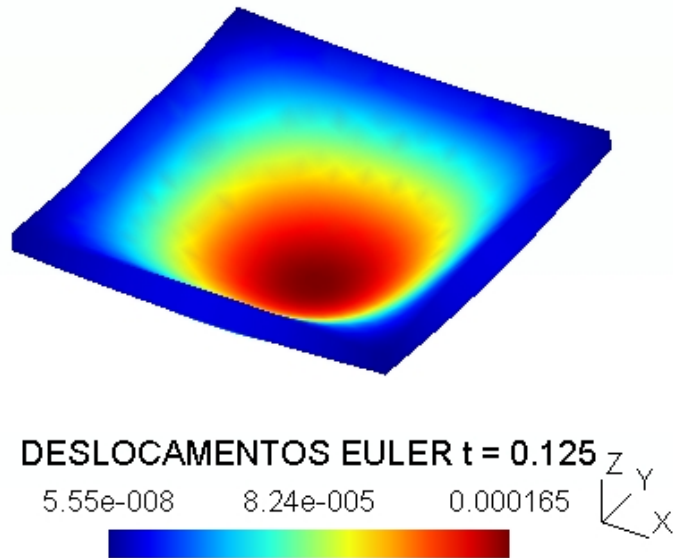


Figura 6.11: Malha Deformada pelo Método de Euler no instante $t = 0,125\text{ s}$

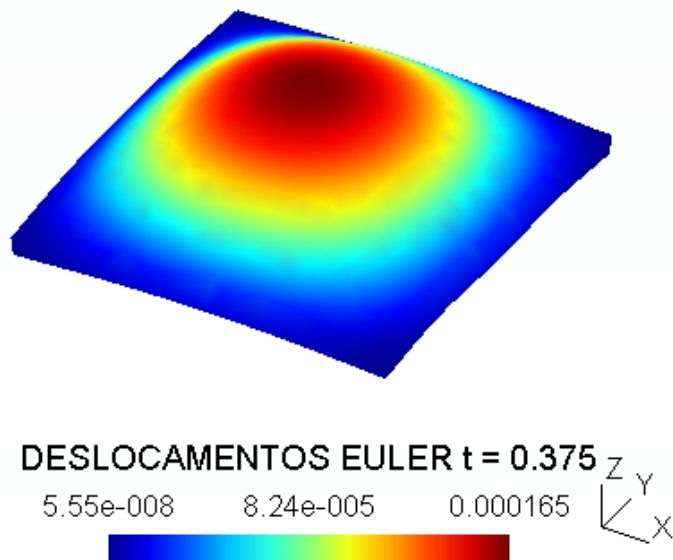


Figura 6.12: Malha Deformada pelo método de Euler no instante $t = 0,375\text{ s}$

- Pelo Método de Newmark

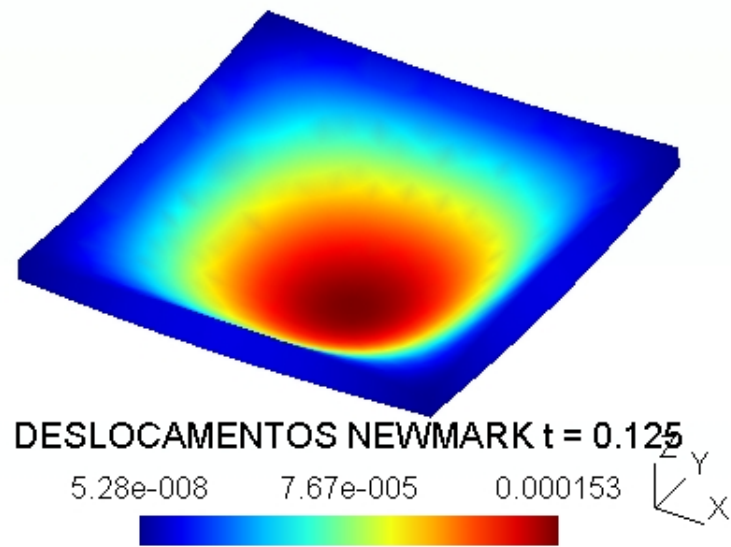


Figura 6.13: Malha Deformada pelo Método de Newmark no instante $t = 0,125$ s

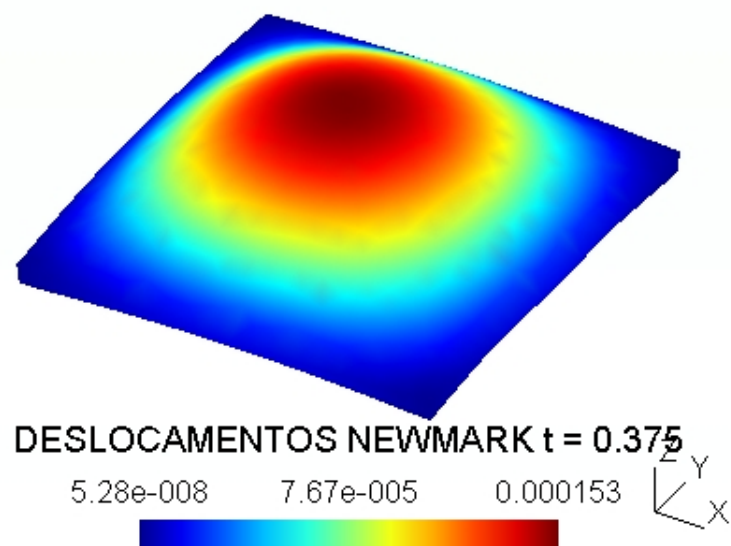


Figura 6.14: Malha Deformada pelo método de Newmark no instante $t = 0,375$ s

6.3.2 Malha e Resultados do SAP2000

No SAP2000 o modelo físico é dado pelo modelo bidimensional de Kirchoff que corresponde a um quadrado com vãos iguais ao modelo tridimensional do FreeFEM++ ($4 \times 4 m$) e discretizada em 400 elementos finitos quadrados ($0,2 \times 0,2 m$). Para simular as condições de fronteira foram colocados apoios fixos em todos os nós pertencentes aos bordos tal como é ilustrado na figura :

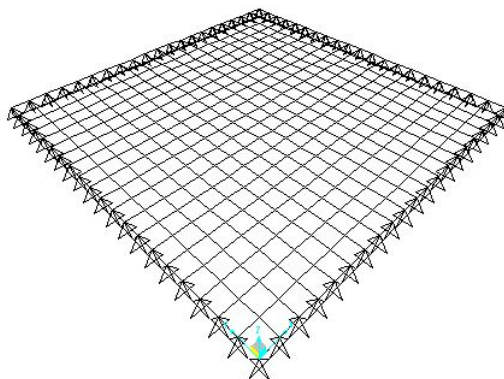
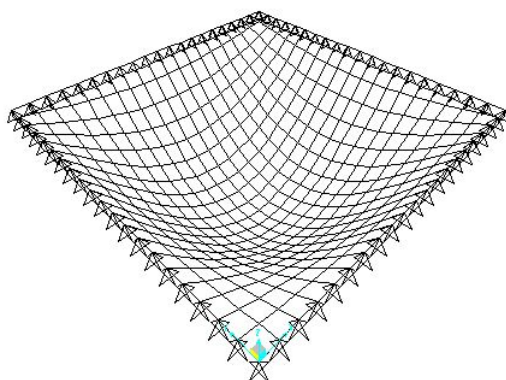


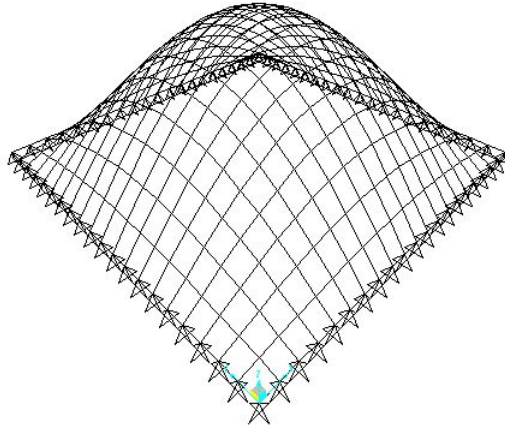
Figura 6.15: Malha do SAP2000 para a laje simplesmente apoiada

Resolvido o problema, utilizando o método de Newmark, obtemos o campo de deslocamentos cuja malha deformada poderá ser visualizada directamente no programa com o respectivo valor do deslocamento vertical máximo para um determinado instante. Como exemplo, nas figura 6.16 e 6.17 temos as malhas deformadas da laje para os instantes $t = 0,125 s$ e $t = 0,375 s$ respectivamente:



Factor de Escala: 10000

Figura 6.16: Malha Deformada do SAP2000 no instante $t = 0,125 s$



Factor de Escala: 10000

Figura 6.17: Malha Deformada do SAP2000 no instante $t = 0,375 s$

6.3.3 Comparações

Apresentamos de seguida as tabelas comparativas dos resultados obtidos para o deslocamento vertical do ponto central da laje para um tempo total de cálculo igual a 1 s, discretizada em intervalos de tempo $\Delta t = 0,03125 s$:

- Pelo Método de Euler

Instante	FF++ Euler [mm]	SAP2000 [mm]	Solução Teórica [mm]	Erro Euler vs. SAP2000	Erro Euler vs. S. Teórica
0,03125	-0,0618	-0,056	-0,0614	9,4	0,6
0,0625	-0,1164	-0,118	-0,1130	1,4	2,9
0,09375	-0,1523	-0,137	-0,1480	10,0	2,8
0,125	-0,1649	-0,165	-0,1600	0,1	3,0
0,15625	-0,1524	-0,139	-0,1480	8,8	2,9
0,1875	-0,1167	-0,114	-0,1130	2,3	3,2
0,21875	-0,0632	-0,06	-0,0614	5,1	2,8
0,25	-1,72E-04	0,0044	0,0000	-	-

Instante	FF++ Euler [mm]	SAP2000 [mm]	Solução Teórica [mm]	Erro Euler vs. SAP2000	Erro Euler vs. S. Teórica
0,28125	0,0629	0,055	0,0614	12,6	2,4
0,3125	0,1165	0,12	0,1130	2,9	3,0
0,34375	0,1523	0,137	0,1480	10,0	2,8
0,375	0,1649	0,165	0,1600	0,1	3,0
0,40625	0,1524	0,139	0,1480	8,8	2,9
0,4375	0,1167	0,114	0,1130	2,3	3,2
0,46875	0,0633	0,058	0,0614	8,4	3,0
0,5	-1,72E-04	-0,003033	0,0000	-	-
0,53125	-0,0629	-0,058	-0,0614	7,8	2,4
0,5625	-0,1165	-0,118	-0,1130	1,3	3,0
0,59375	-0,1523	-0,139	-0,1480	8,7	2,8
0,625	-0,1649	-0,165	-0,1600	0,1	3,0
0,65625	-0,1524	-0,136	-0,1480	10,8	2,9
0,6875	-0,1167	-0,117	-0,1130	0,3	3,2
0,71875	-0,0633	-0,051	-0,0614	19,4	3,0
0,75	-1,72E-04	-0,003805	0,0000	-	-
0,78125	0,0629	0,067	0,0614	6,1	2,4
0,8125	0,1165	0,109	0,1130	6,4	3,0
0,84375	0,1523	0,148	0,1480	2,8	2,8
0,875	0,1649	0,156	0,1600	5,4	3,0
0,90625	0,1524	0,143	0,1480	6,2	2,9
0,9375	0,1167	0,111	0,1130	4,9	3,2
0,96875	0,0633	0,054	0,0614	14,7	3,0
1	-1,72E-04	0,001097	0,0000	-	-

Tabela 6.3: Tabela Comparativa de resultados do deslocamento vertical do ponto central da laje simplesmente apoiada pelo método de Euler

- Pelo Método de Newmark

Instante	FF++ Newmark [mm]	SAP2000 [mm]	Solução Teórica [mm]	Erro Newmark vs. SAP2000	Erro Newmark vs. S. Teórica
0,03125	-0,0587	-0,056	-0,0614	4,6	4,4
0,0625	-0,108	-0,118	-0,1130	8,5	4,4
0,09375	-0,142	-0,137	-0,1480	3,5	4,1
0,125	-0,153	-0,165	-0,1600	7,3	4,4
0,15625	-0,142	-0,139	-0,1480	2,1	4,1
0,1875	-0,108	-0,114	-0,1130	5,3	4,4
0,21875	-0,0587	-0,06	-0,0614	2,2	4,4
0,25	2,79E-10	0,0044	0,0000	-	-
0,28125	0,0587	0,055	0,0614	6,3	4,4
0,3125	0,108	0,12	0,1130	10,0	4,4
0,34375	0,142	0,137	0,1480	3,5	4,1
0,375	0,153	0,165	0,1600	7,3	4,4
0,40625	0,142	0,139	0,1480	2,1	4,1
0,4375	0,108	0,114	0,1130	5,3	4,4
0,46875	0,0587	0,058	0,0614	1,2	4,4
0,5	2,79E-10	-0,003033	0,0000	-	-
0,53125	-0,0587	-0,058	-0,0614	1,2	4,4
0,5625	-0,108	-0,118	-0,1130	8,5	4,4
0,59375	-0,142	-0,139	-0,1480	2,1	4,1
0,625	-0,153	-0,165	-0,1600	7,3	4,4
0,65625	-0,142	-0,136	-0,1480	4,2	4,1
0,6875	-0,108	-0,117	-0,1130	7,7	4,4
0,71875	-0,0587	-0,051	-0,0614	13,1	4,4
0,75	2,79E-10	-0,003805	0,0000	-	-
0,78125	0,0587	0,067	0,0614	12,4	4,4

Instante	FF++	SAP2000	Solução Teórica	Erro	Erro
	Newmark			Newmark	Newmark
	[mm]	[mm]	[mm]	vs.	vs.
				SAP2000	S. Teórica
0,8125	0,108	0,109	0,1130	0,9	4,4
0,84375	0,142	0,148	0,1480	4,1	4,1
0,875	0,153	0,156	0,1600	1,9	4,4
0,90625	0,142	0,143	0,1480	0,7	4,1
0,9375	0,108	0,111	0,1130	2,7	4,4
0,96875	0,0587	0,054	0,0614	8,0	4,4
1	2,79E-10	0,001097	0,0000	-	-

Tabela 6.4: Tabela Comparativa de resultados do deslocamento vertical do ponto central da laje simplesmente apoiada

Nota: (-) comparação de diferentes gamas de zeros

No caso da laje simplesmente apoiada, da comparação entre os resultados de deslocamentos do FreeFEM++ e do SAP2000, nota-se uma maior variação no valores dos erros obtidos em cada instante comparativamente ao primeiro caso da laje totalmente encastrada. Variação esta que no método de Euler vai dos 0,1 % aos 17,3 % nos instantes $t = 0,125 s$ e $0,28125 s$ respectivamente e que no método de Newmark vai dos 0,7 % aos 13,1 % nos instantes $t = 0,90625 s$ e $0,71875 s$ respectivamente. De um modo geral a maior parte dos erros situa-se abaixo da gama dos 10 % sendo que este é ultrapassado com uma maior frequência do que acontece na laje totalmente encastrada. Relativamente à comparação entre os resultados do FreeFEM++ com a Solução Teórica de Navier, à excepção do primeiro instante ($t = 0,03125 s$) nos restantes instantes o valor do erro situa-se entre os 2 % e os 3 % no método de Euler e na ordem dos 4 no método de Newmark. Portanto há uma maior aproximação entre os resultados do FreeFEM++ e da Solução Teórica de Navier, em que o método de Euler é o mais próximo.

Para observar a resposta em deslocamentos do ponto central da laje ao longo do tempo é apresentado o seguinte gráfico:

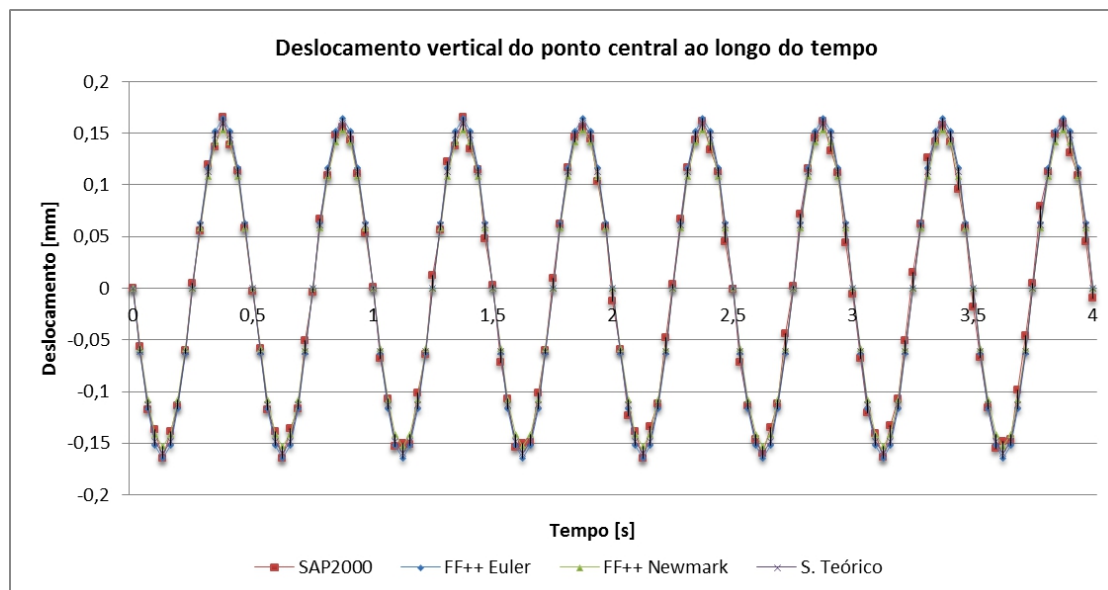


Figura 6.18: Gráfico da resposta em deslocamento do ponto central da laje simplesmente apoiada

O gráfico da resposta em deslocamentos do ponto central da laje demonstra a grande aproximação dos resultados obtidos a partir das três análises realçando-se uma maior aproximação da Solução Teórica de Navier com a nossa implementação em FreeFEM++.

6.4 Laje Vigada apoiada sobre Pilares Rígidos

O terceiro caso corresponde a uma laje vigada quadrada com vãos de 4 m em ambas as direcções horizontais e com uma espessura de 20 cm . As vigas têm uma altura de $0,5\text{ m}$ e uma largura de $0,2\text{ m}$ que por sua vez apoiam-se sobre um conjunto de 4 pilares indeformáveis (elevada rigidez) com secções transversais de $0,2 \times 0,2\text{ m}$, localizados junto aos cantos da laje vigada. (cf. Anexo A.3)

6.4.1 Malha do GMSH e Resultados do FreeFEM++

Neste caso de estudo o modelo físico tridimensional, gerado a partir do programa GMSH, corresponde a um conjunto de paralelepípedos, com as respectivas dimensões,

ligados entre si formando um único volume que engloba a laje e as vigas. A discretização é feita através de elementos finitos tetraédricos obtendo um total de 4608 elementos. Visto que foram considerados pilares indeformáveis, e impedindo todos os deslocamentos na ligação viga-pilar, junto aos cantos da laje, ($u_1 = 0, u_2 = 0$ e $u_3 = 0$) consegue-se simular a ligação entre estes elementos, sem ter a necessidade de considerar os pilares.

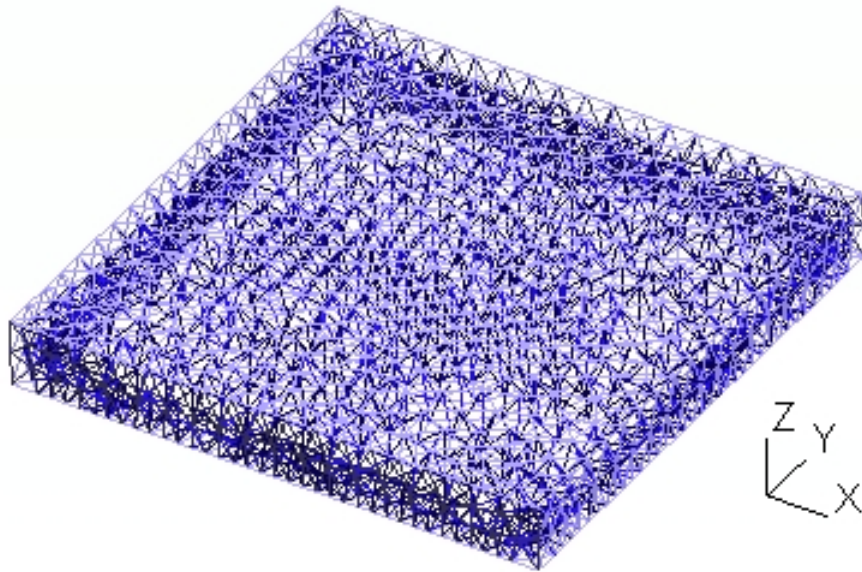


Figura 6.19: Malha do GMSH para a laje vigada

Resolvido o problema com o programa FreeFEM++, obtemos o campo de deslocamentos da malha que nos permite visualizar a malha deformada, em GMSH, com o respectivo valor do deslocamento vertical máximo para um determinado instante. Como exemplo, nas figuras 6.20, 6.21, 6.22 e 6.23 temos as malhas deformadas da laje para os instantes $t = 0,125 s$ e $t = 0,375 s$ respectivamente:

- Pelo Método de Euler

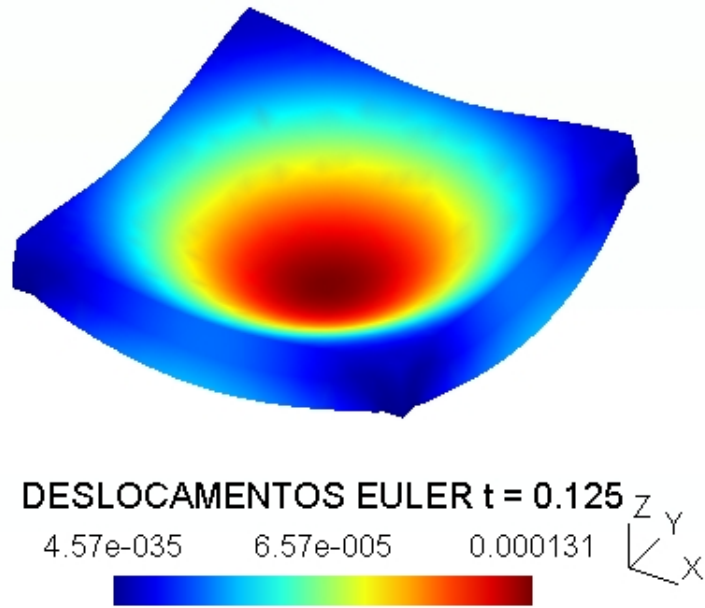


Figura 6.20: Malha Deformada pelo Método de Euler no instante $t = 0,125$ s

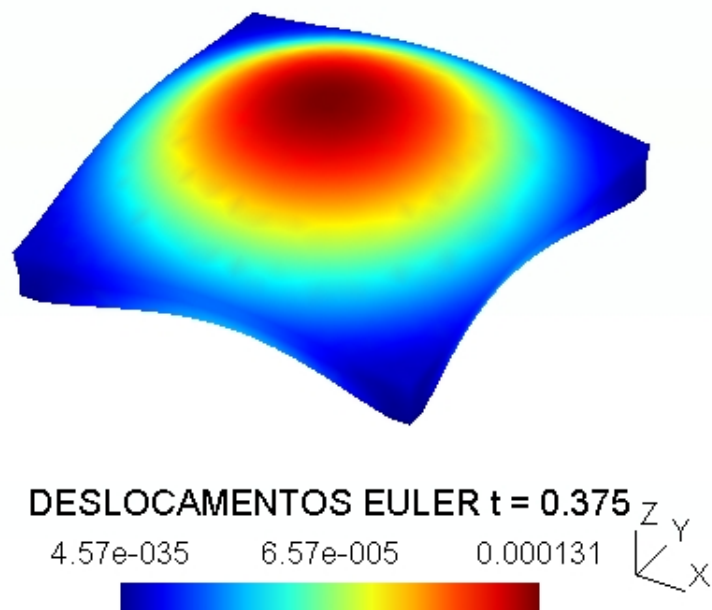


Figura 6.21: Malha Deformada pelo método de Euler no instante $t = 0,375$ s

- Pelo Método de Newmark

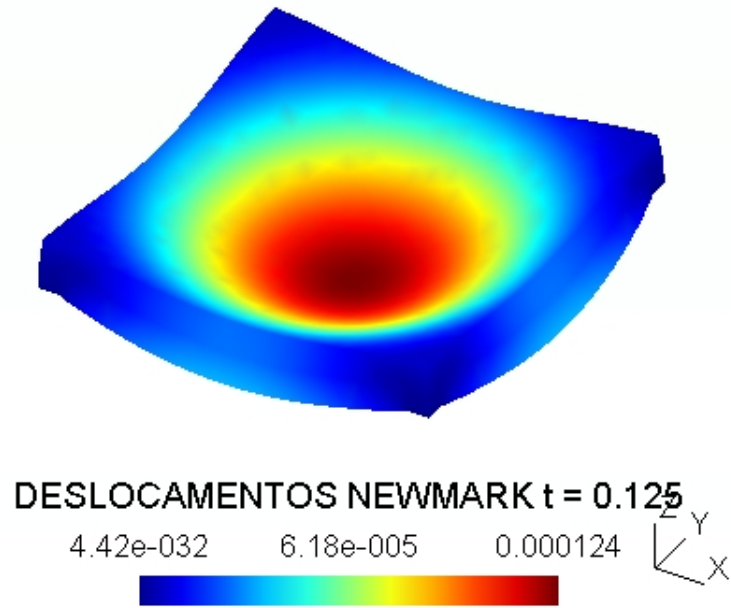


Figura 6.22: Malha Deformada pelo Método de Newmark no instante $t = 0,125$ s

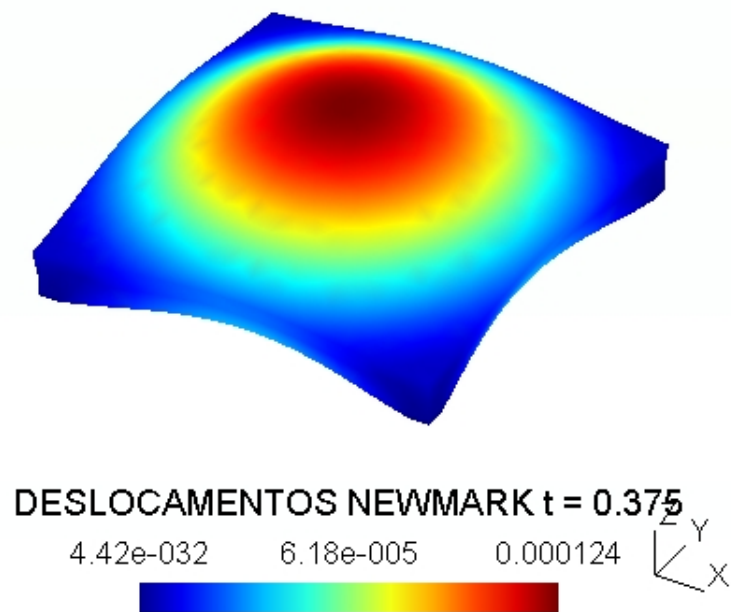


Figura 6.23: Malha Deformada pelo método de Newmark no instante $t = 0,375$ s

6.4.2 Malha e Resultados do SAP2000

No SAP2000, o modelo físico da laje é dado pelo modelo bidimensional de Kirchoff que corresponde a um quadrado com vãos de dimensões $3,8 \times 3,8 m$ e discretizada em 1444 elementos finitos quadrados ($0,1 \times 0,1 m$). Nos bordos são colocadas vigas com secções transversais de $0,5 \times 0,2 m$ estando estas encastradas nas suas extremidades de modo a simular o encastramento nos pilares.

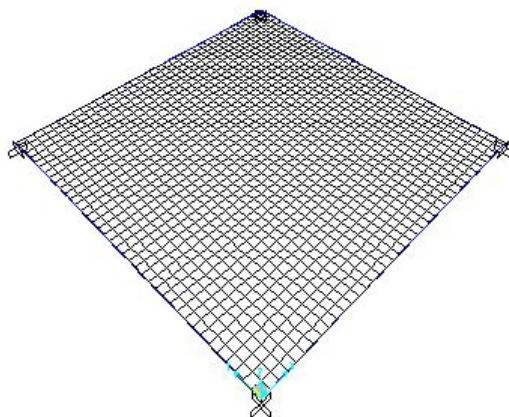
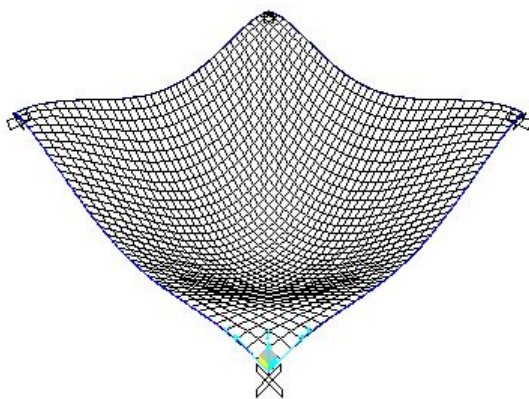


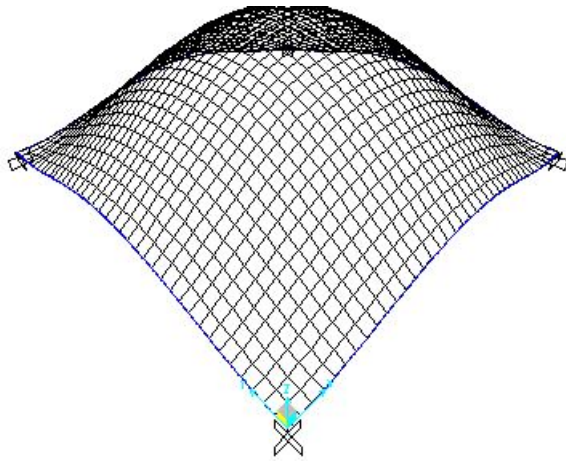
Figura 6.24: Malha do SAP2000 para a laje vigada

Resolvido o problema é obtido o campo de deslocamentos cuja malha deformada poderá ser visualizada directamente no programa com o respectivo valor do deslocamento vertical máximo para um determinado instante. Como exemplo, nas figura 6.25 e 6.26 temos as malhas deformadas da laje para os instantes $t = 0,125 s$ e $t = 0,375 s$ respectivamente:



Factor de Escala: 10000

Figura 6.25: Malha Deformada do SAP2000 no instante $t = 0,125 s$



Factor de Escala: 10000

Figura 6.26: Malha Deformada do SAP2000 no instante $t = 0,375 s$

6.4.3 Comparações

Vejam os agora a tabela comparativa dos resultados obtidos para o deslocamento vertical do ponto central da laje para um tempo total de cálculo igual a 1 s, discretizada em intervalos de tempo $\Delta t = 0,03125 s$:

Instante	FF++ Newmark [mm]	FF++ Euler [mm]	SAP2000 [mm]	Erro Newmark vs. SAP2000	Erro Euler vs. SAP2000
0,03125	-0,0473	-0,0494	-0,043	9,1	13,0
0,0625	-0,0875	-0,0928	-0,088	0,6	5,2
0,09375	-0,114	-0,121	-0,104	8,8	14,0
0,125	-0,124	-0,131	-0,124	0,0	5,3
0,15625	-0,114	-0,121	-0,104	8,8	14,0
0,1875	-0,0875	-0,093	-0,087	0,6	6,5
0,21875	-0,0473	-0,0504	-0,043	9,1	14,7
0,25	7,22E-10	1,16E-04	0,0001271	-	-
0,28125	-0,0473	0,0502	0,045	4,9	10,4
0,3125	-0,0875	0,0928	0,087	0,6	6,3
0,34375	-0,114	0,121	0,106	7,0	12,4

Instante	FF++ Newmark [mm]	FF++ Euler [mm]	SAP2000 [mm]	Erro Newmark vs. SAP2000	Erro Euler vs. SAP2000
0,375	-0,124	0,131	0,123	0,8	6,1
0,40625	-0,114	0,121	0,104	8,8	14,0
0,4375	-0,0875	0,093	0,088	0,6	5,4
0,46875	-0,0473	0,0504	0,039	17,5	22,6
0,5	7,22E-10	1,16E-04	0,0035	-	-
0,53125	-0,0473	-0,0502	-0,05	5,4	0,4
0,5625	-0,0875	-0,0928	-0,081	7,4	12,7
0,59375	-0,114	-0,121	-0,112	1,8	7,4
0,625	-0,124	-0,131	-0,117	5,6	10,7
0,65625	-0,114	-0,121	-0,109	4,4	9,9
0,6875	-0,0875	-0,093	-0,083	5,1	10,8
0,71875	-0,0473	-0,0504	-0,042	11,2	16,7
0,75	7,22E-10	-1,16E-04	-0,00001107	-	-
0,78125	-0,0473	0,0502	0,049	3,5	2,4
0,8125	-0,0875	0,0928	0,082	6,3	11,6
0,84375	-0,114	0,121	0,113	0,9	6,6
0,875	-0,124	0,131	0,115	7,3	12,2
0,90625	-0,114	0,121	0,112	1,8	7,4
0,9375	-0,0875	0,093	0,078	10,9	16,1
0,96875	-0,0473	0,0504	0,047	0,6	6,7
1	7,22E-10	-1,16E-04	-0,006513	-	-

Tabela 6.5: Tabela Comparativa de resultados do deslocamento vertical do ponto central da laje vigada

Nota: (-) comparação de diferentes gamas de zeros

A partir da análise da tabela 6.5, em termos de deslocamentos, é possível constatar-se que no método de Newmark há uma forte presença de erros entre 5 % a 9 % para os diferentes instantes, estando o restante bem abaixo desses valores à exceção dos instantes $t = 0,46875 s$, $t = 0,71875 s$ e $t = 0,9375 s$ em que ocorreram erros de 17,5 %, 11,2 % e 10,9 % respectivamente. Ao que parece os erros no método de Euler sofrem todos um agravamento na ordem dos 5 %. Este caso tem a particularidade de envolver mais do que um elemento estrutural (laje e vigas) o que justifica o aumento dos erros em relação aos casos apresentados anteriormente, uma vez que há um acréscimo de dificuldade na definição dos modelos, principalmente em termos das dimensões destes, para que possam apresentar o mesmo comportamento.

Para observar a resposta em deslocamentos do ponto central da laje ao longo do tempo é apresentado o seguinte gráfico:

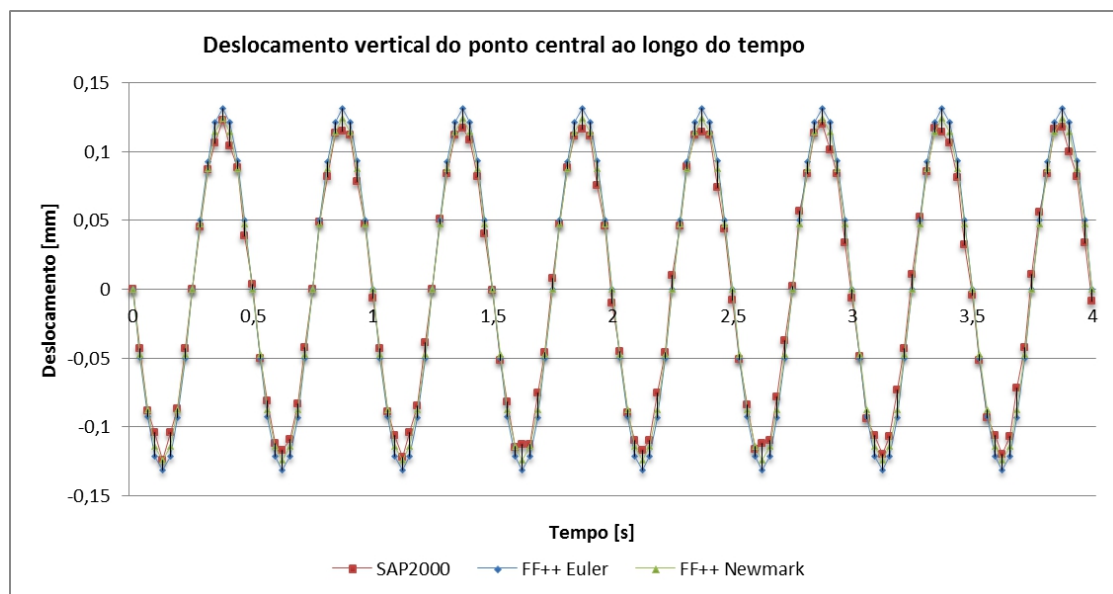


Figura 6.27: Gráfico da resposta em deslocamento do ponto central da laje vigada

Pela observação do gráfico já se nota alguma dispersão entre os resultados do método de Euler do FreeFEM++ e os resultados do SAP2000, comparativamente com os outros casos apresentados anteriormente. Embora os resultados do método de Newmark do FreeFEM++ ainda apresentem uma boa aproximação relativamente aos resultados do SAP2000.

6.5 Laje Fungiforme

O quarto e último caso estudado corresponde a uma laje quadrada com vãos de 4 m em ambas as direcções horizontais e com uma espessura de 20 cm , estando os quatro cantos da laje apoiados directamente sobre pilares. Os pilares apresentam uma secção transversal de dimensões $20 \times 20\text{ cm}$, uma altura de $2,8\text{ m}$ e encontram-se encastrados na sua base. (cf. Anexo A.4)

6.5.1 Malha do GMSH e Resultados do FreeFEM++

Neste caso de estudo o modelo físico tridimensional laje-pilares, gerado a partir do programa GMSH, corresponde a um conjunto de paralelepípedos, com as respectivas dimensões, ligados entre si formando um único volume. A discretização é feita através de elementos finitos tetraédricos obtendo um total de 5339 elementos. O encastramento da base dos pilares é garantido impedindo o deslocamento das mesmas em todas as direcções: $u_1 = 0$, $u_2 = 0$ e $u_3 = 0$.

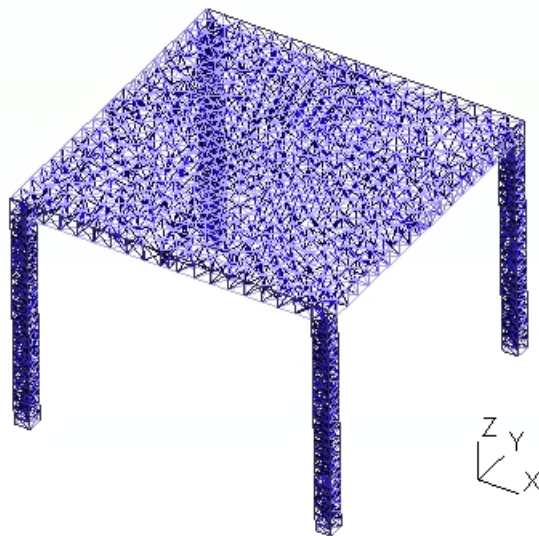


Figura 6.28: Malha do GMSH para a laje fungiforme

Após a resolução numérica obtemos o campo de deslocamentos da malha, sendo que a partir do GMSH é possível visualizar a malha deformada com o respectivo valor do deslocamento vertical máximo para um determinado instante. Como exemplo, nas figuras 6.29, 6.30, 6.31 e 6.32 temos as malhas deformadas da laje para os instantes $t = 0,125\text{ s}$ e $t = 0,375\text{ s}$ respectivamente:

- Pelo Método de Euler

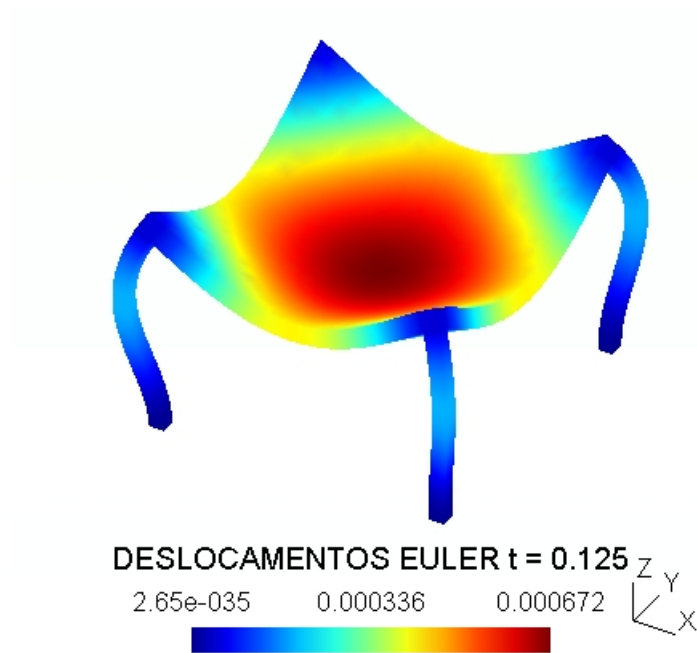


Figura 6.29: Malha Deformada pelo Método de Euler no instante $t = 0,125 s$

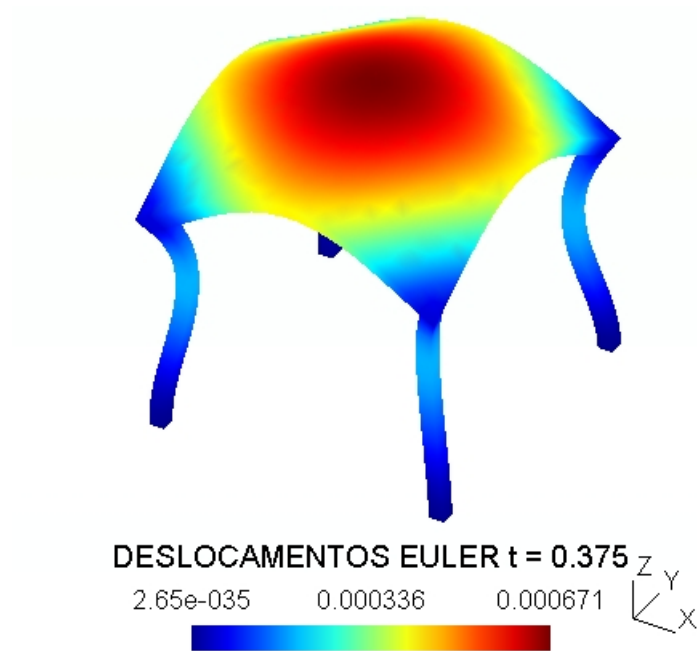


Figura 6.30: Malha Deformada pelo método de Euler no instante $t = 0,375 s$

- Pelo Método de Newmark

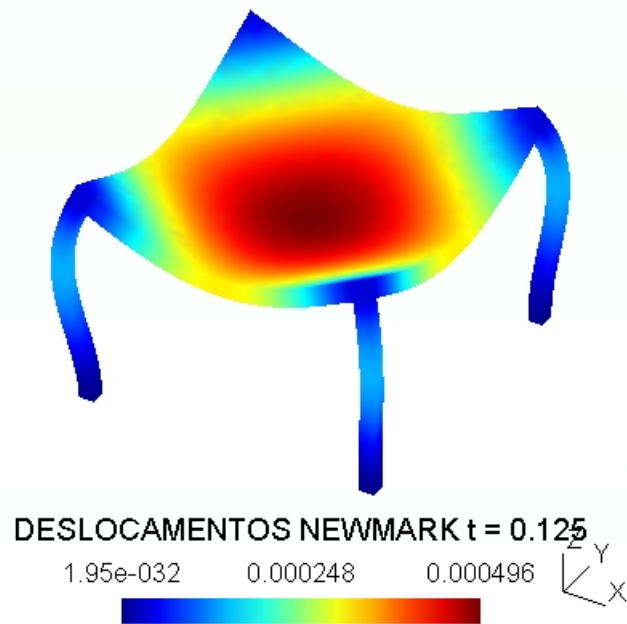


Figura 6.31: Malha Deformada pelo Método de Newmark no instante $t = 0,125$ s

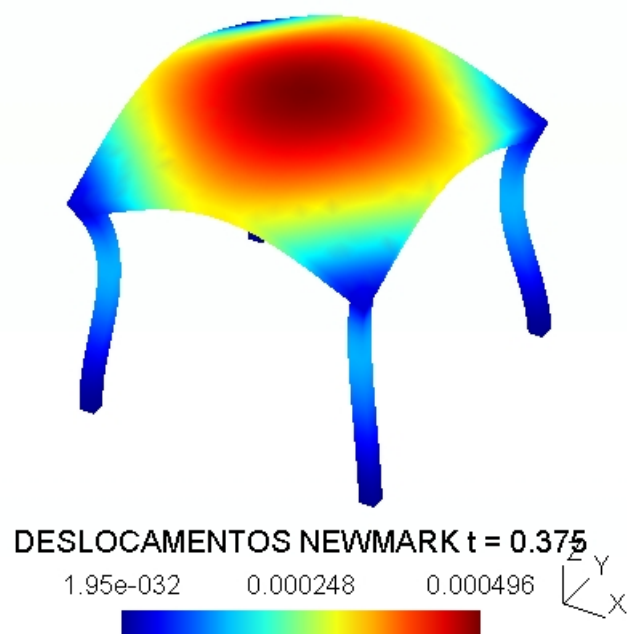


Figura 6.32: Malha Deformada pelo método de Newmark no instante $t = 0,375$ s

6.5.2 Malha e Resultados do SAP2000

No SAP2000, o modelo físico da laje é dado pelo modelo bidimensional de Kirchhoff que corresponde a um quadrado com vãos iguais ao modelo tridimensional do FreeFEM++ ($4 \times 4 m$) e discretizada em 484 elementos finitos quadrados e rectangulares ($0,2 \times 0,2 m$, $0,1 \times 0,1 m$ e $0,2 \times 0,1 m$). A densidade da discretização foi aumentada junto aos pilares e foram atribuídos "Constraints" aos nós dos elementos $0,1 \times 0,1 m$ de modo a simular melhor a ligação laje-pilar. As condições de fronteira são atribuídas colocando encastramentos na base dos pilares tal como é ilustrado na figura 6.33:

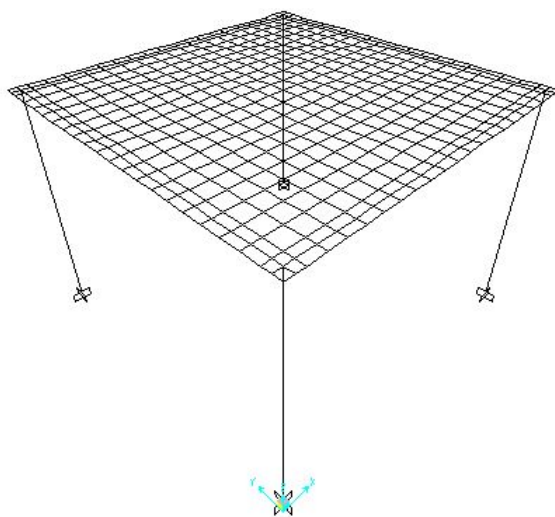
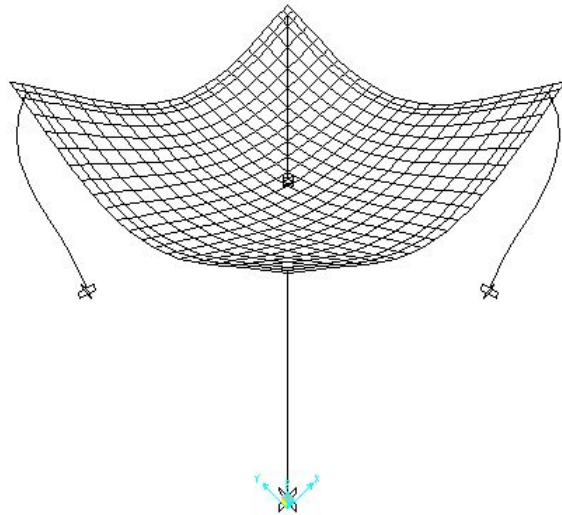


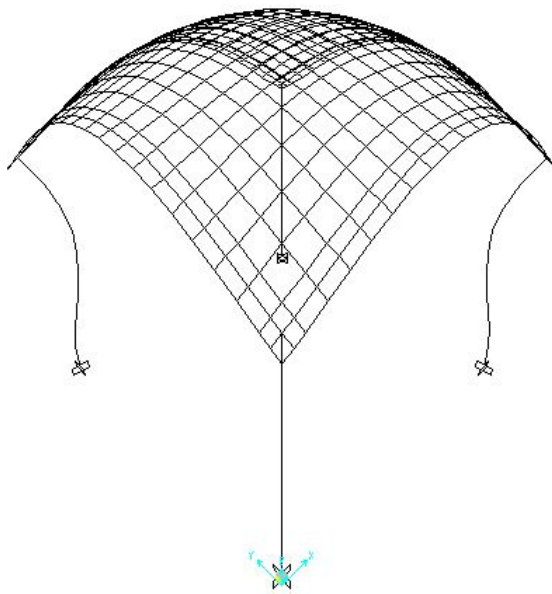
Figura 6.33: Malha do SAP2000 para a laje fungiforme

Resolvido o problema, utilizando o método de Newmark, é obtido o campo de deslocamentos cuja malha deformada poderá ser visualizada directamente no programa com o respectivo valor do deslocamento vertical máximo para um determinado instante. Como exemplo, nas figura 6.34 e 6.35 temos as malhas deformadas da laje para os instantes $t = 0,125 s$ e $t = 0,375 s$ respectivamente:



Factor de Escala: 10000

Figura 6.34: Malha Deformada do SAP2000 no instante $t = 0,125 s$



Factor de Escala: 10000

Figura 6.35: Malha Deformada do SAP2000 no instante $t = 0,375 s$

6.5.3 Comparações

A seguir exibimos a tabela comparativa dos resultados obtidos para o deslocamento vertical do ponto central da laje para um tempo total de cálculo igual a 1 s, discretizada em intervalos de tempo $\Delta t = 0,03125 s$:

Instante	FF++ Euler [mm]	FF++ Newmark [mm]	SAP2000 [mm]	Erro Euler vs. SAP2000	Erro Newmark vs. SAP2000
0,03125	-0,234	-0,19	-0,19	18,8	0,0
0,0625	-0,469	-0,35	-0,541	13,3	35,3
0,09375	-0,62	-0,458	-0,614	1,0	25,4
0,125	-0,672	-0,496	-0,605	10,0	18,0
0,15625	-0,622	-0,458	-0,688	9,6	33,4
0,1875	-0,477	-0,35	-0,46	3,6	23,9
0,21875	-0,26	-0,19	-0,195	25,0	2,6
0,25	-3,34E-03	2,19E-09	-0,07	-	-
0,28125	0,254	0,19	0,28	9,3	32,1
0,3125	0,472	0,35	0,527	10,4	33,6
0,34375	0,619	0,458	0,543	12,3	15,7
0,375	0,671	0,496	0,702	4,4	29,3
0,40625	0,622	0,458	0,655	5,0	30,1
0,4375	0,477	0,35	0,391	18,0	10,5
0,46875	0,26	0,19	0,297	12,5	36,0
0,5	3,34E-03	2,19E-09	0,017	-	-
0,53125	-0,254	-0,19	-0,336	24,4	43,5
0,5625	-0,472	-0,35	-0,418	11,4	16,3
0,59375	-0,619	-0,458	-0,615	0,6	25,5
0,625	-0,671	-0,496	-0,734	8,6	32,4
0,65625	-0,622	-0,458	-0,543	12,7	15,7
0,6875	-0,477	-0,35	-0,483	1,2	27,5
0,71875	-0,26	-0,19	-0,297	12,5	36,0
0,75	3,34E-03	2,19E-09	0,087	-	-
0,78125	0,254	0,19	0,224	11,8	15,2
0,8125	0,472	0,35	0,452	4,2	22,6

Instante	FF++ Euler [mm]	FF++ Newmark [mm]	SAP2000 [mm]	Erro Euler vs. SAP2000	Erro Newmark vs. SAP2000
0,84375	0,619	0,458	0,695	10,9	34,1
0,875	0,671	0,496	0,606	9,7	18,2
0,90625	0,622	0,458	0,613	1,4	25,3
0,9375	0,477	0,35	0,524	9,0	33,2
0,96875	0,26	0,19	0,213	18,1	10,8
1	3,34E-03	2,19E-09	0,018	-	-

Tabela 6.6: Tabela Comparativa de resultados do deslocamento vertical do ponto central da laje fungiforme

Nota: (-) comparação de diferentes gamas de zeros

Com base na tabela de resultados apresentado pode-se dizer que há uma maior discrepância entre os resultados obtidos pelo FreeFEM++ e pelo SAP2000 comparativamente aos outros casos estudados anteriormente. Neste caso os valores de alguns dos erros obtidos com o método de Euler chegam a valores próximos dos 25 % como é o caso dos instantes $t = 0,21875 s$, $t = 0,53125 s$. Contudo, ainda podemos encontrar instantes com erros bastante baixos, inferiores a 1 %, como acontece com os instante $t = 0,59375 s$ e $t = 0,90625 s$. Nos restantes instantes temos maioritariamente erros abaixo dos 10 % embora possamos encontra erros situados entre os 10 % e os 20 %. No método de Newmark o caso desfigura-se obtendo valores bastante elevados comparativamente aos restantes casos. Para este método temos um valor máximo de erro de 43,5 % no instante $t = 0,53125 s$ e um valor mínimo de 0 % no instante inicial. Como foi dito anteriormente, a complexidade da estrutura é directamente proporcional à dificuldade em aproximar os modelos, aumentando a probabilidade de ocorrerem erros maiores, que é o que se verifica com este caso.

Para observar a resposta em deslocamentos do ponto central da laje ao longo do tempo é apresentado o seguinte gráfico:

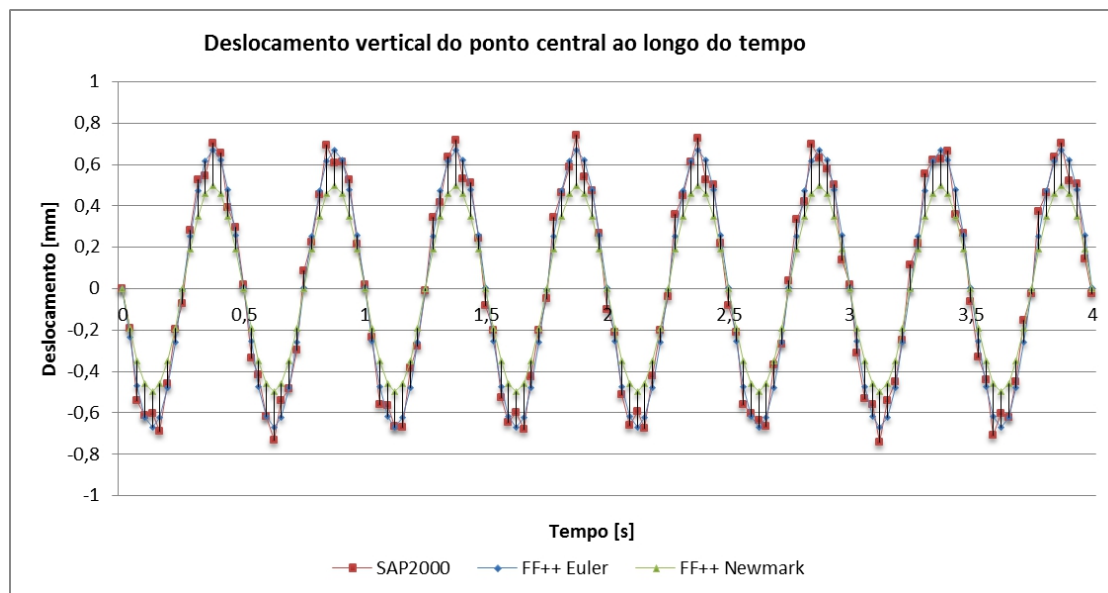


Figura 6.36: Gráfico da resposta em deslocamento do ponto central da laje fungiforme

Neste caso da laje fungiforme, já se começa a observar no gráfico das respostas em deslocamentos do ponto central da laje alguns desvios entre as curvas, principalmente junto aos picos das respostas. Para este caso em particular o método de Newmark em FreeFEM++ é o que apresenta maior dispersão em relação aos resultados do SAP2000, o que já era expectável tendo em conta o parâmetro de discretização utilizado que é igual a 0,5, superior ao valor utilizado nos outros casos de 0,2. A razão desta escolha prende-se com o tempo necessário para a execução dos cálculos. Com este valor de parâmetro cada simulação de 4 s representa mais de 8 horas de cálculo num PC com um processador *Intel(R) Core(TM) i5 - 2410M CPU 2,30GHz*.

6.6 Análise de Tensões

A partir das soluções do problema dinâmico, mais precisamente o campo de deslocamentos da laje, torna-se possível obter as restantes incógnitas do problema,

nomeadamente o campo de deformação e de tensão da laje, recorrendo às equações de compatibilidade, Eq. (3.3) e às relações constitutivas, Eq. (3.7), respectivamente.

Deste modo também se elaborou uma análise em termos de tensões máximas e mínimas, na superfície superior e inferior da laje, a partir da implementação das referidas equações no FreeFEM++, comparando-as posteriormente com os resultados do SAP2000. Como exemplo tem-se o primeiro caso de estudo, correspondente à laje totalmente encastrada, no qual foram determinadas as tensões máximas e mínimas nos quatro primeiros instantes:

Instante	Tensão	FF++	FF++	SAP2000	Erro	Erro
		Newmark [kN/m ²]	Euler [kN/m ²]		vs. SAP2000	Euler vs. SAP2000
0,03125	Sxx	155	159	146	5,8	8,2
		-153	-158	-146	4,6	7,6
	Syy	154	157	146	5,2	7,0
		-155	-157	-146	5,8	7,0
		Sxy	27,2	27,6	27,5	1,1
-27,4	-27,6		-27,5	0,4	0,4	
0,0625	Sxx	285	296	278	2,5	6,1
		-286	-293	-278	2,8	5,1
	Syy	286	291	278	2,8	4,5
		-283	-292	-278	1,8	4,8
		Sxy	50,3	51,2	52,8	4,7
-50,6	-51,3		-52,8	4,2	2,8	
0,09375	Sxx	373	386	351	5,9	9,1
		-374	-383	-351	6,1	8,4
	Syy	374	380	351	6,1	7,6
		-370	-381	-351	5,1	7,9
		Sxy	65,8	66,9	66,2	0,6
-66,1	-67,1		-66,2	0,2	1,3	

Instante	Tensão	FF++	FF++	SAP2000	Erro	Erro
		Newmark	Euler		vs.	Euler
		[kN/m ²]	[kN/m ²]	[kN/m ²]	SAP2000	SAP2000
0,125	S _{xx}	405	418	393	3,0	6,0
		-400	-415	-393	1,8	5,3
	S _{yy}	404	411	393	2,7	4,4
		-405	-413	-393	3,0	4,8
	S _{xy}	71,2	72,4	74,7	4,7	3,1
		-71,5	-72,6	-74,7	4,3	2,8

Tabela 6.7: Tabela Comparativa de resultados de tensões máximas e mínimas da laje encastrada

Em termos de distribuição, tendo como exemplo o primeiro instante de cálculo, $t = 0,03125 s$, tem-se:

- Pelo Método de Newmark

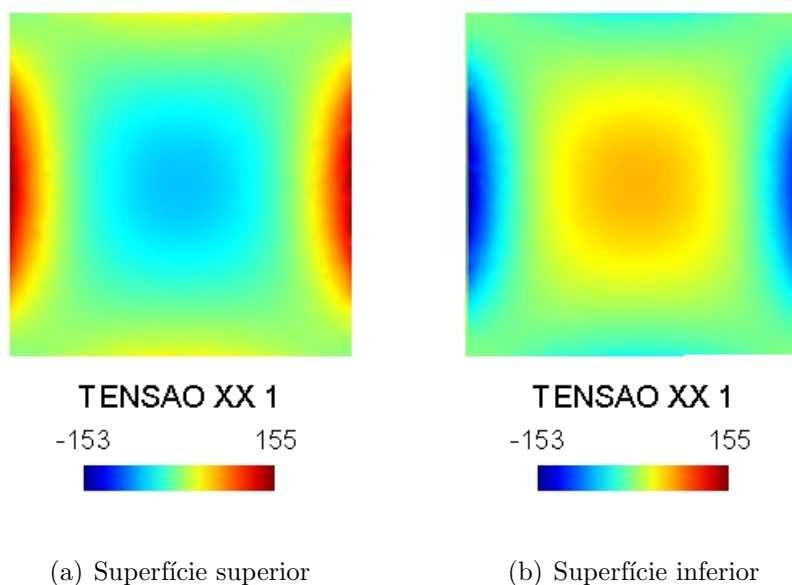


Figura 6.37: Distribuição de tensões ao longo da laje encastrada segundo a direcção xx para o instante $t = 0,03125 s$, pelo método de Newmark

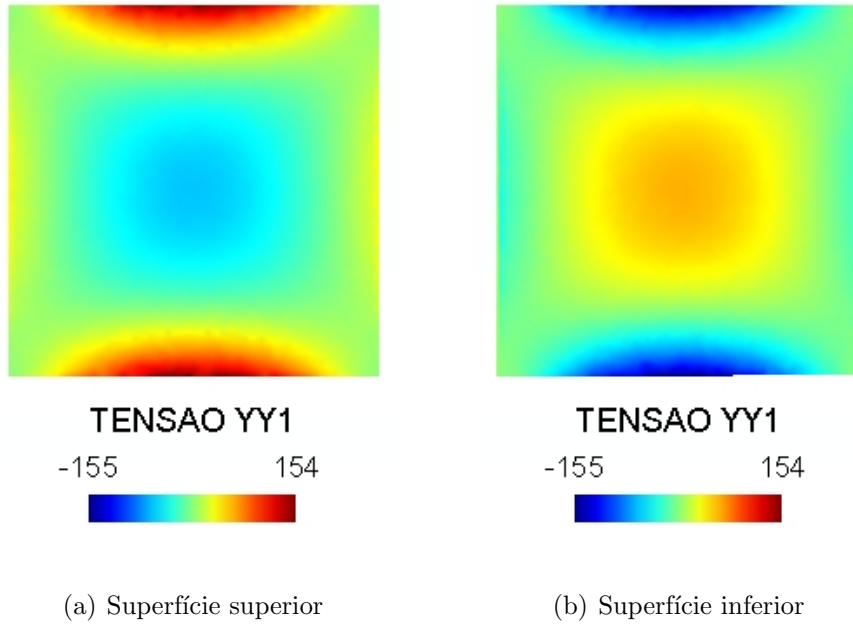


Figura 6.38: Distribuição de tensões ao longo da laje encastrada segundo a direcção yy para o instante $t = 0,03125$ s, pelo método de Newmark

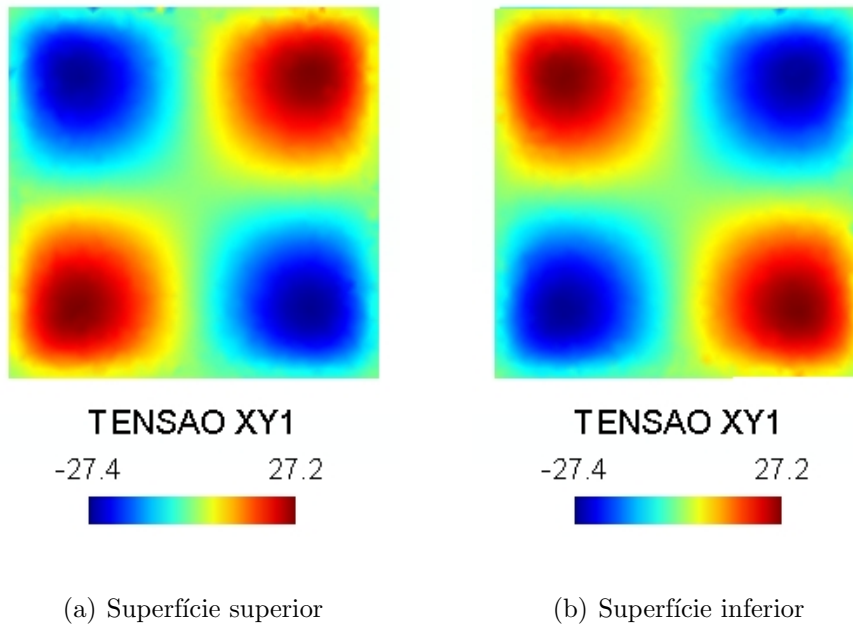


Figura 6.39: Distribuição de tensões tangenciais ao longo da laje encastrada para o instante $t = 0,03125$ s, pelo método de Newmark

- Pelo Método de Euler

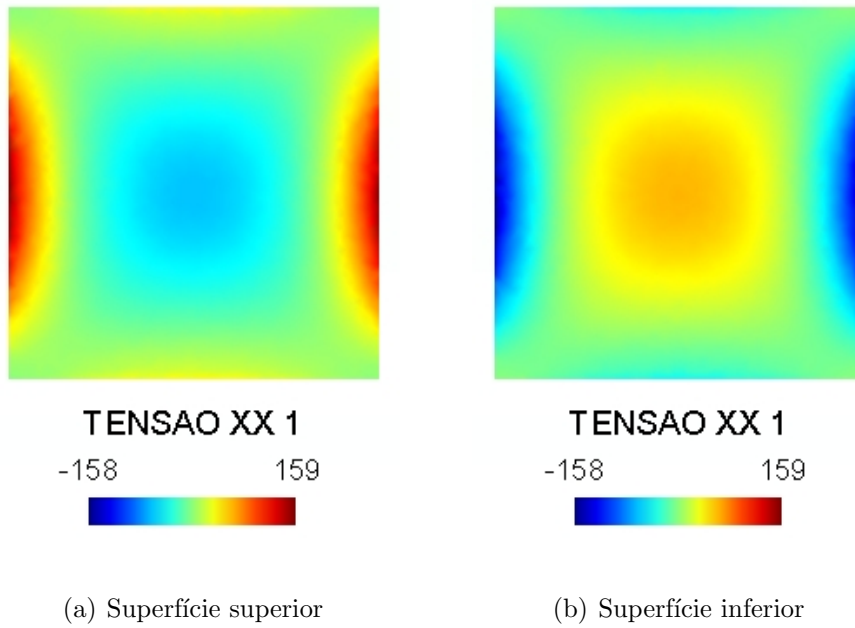


Figura 6.40: Distribuição de tensões ao longo da laje encastrada segundo a direcção xx para o instante $t = 0,03125 s$, pelo método de Euler

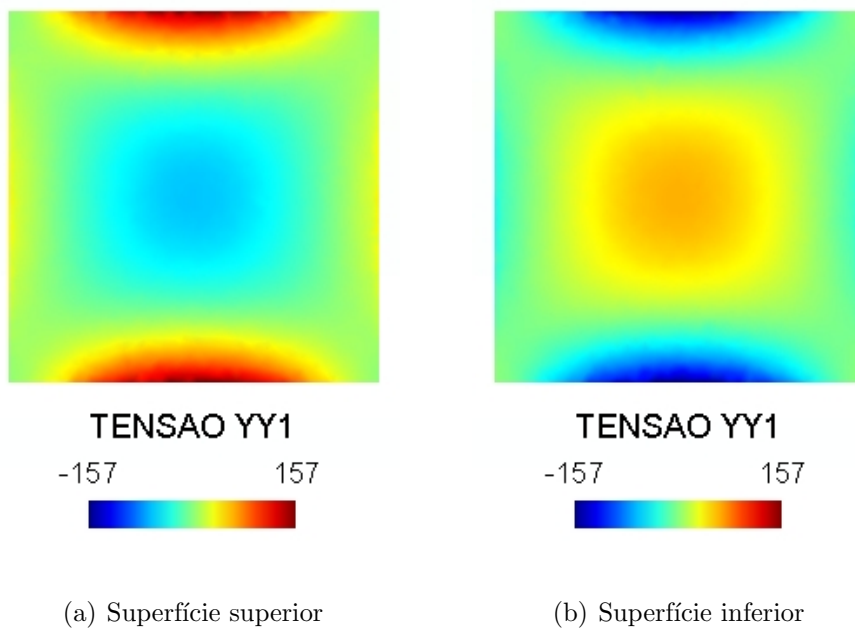


Figura 6.41: Distribuição de tensões ao longo da laje encastrada segundo a direcção yy para o instante $t = 0,03125 s$, pelo método de Euler

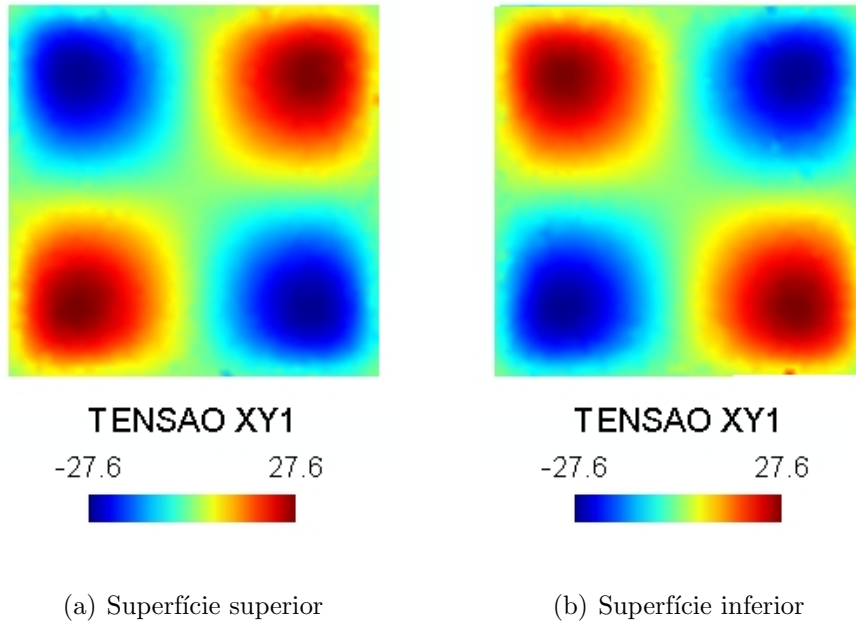


Figura 6.42: Distribuição de tensões tangenciais ao longo da laje encastrada para o instante $t = 0,03125 s$, pelo método de Euler

- Pelo SAP2000

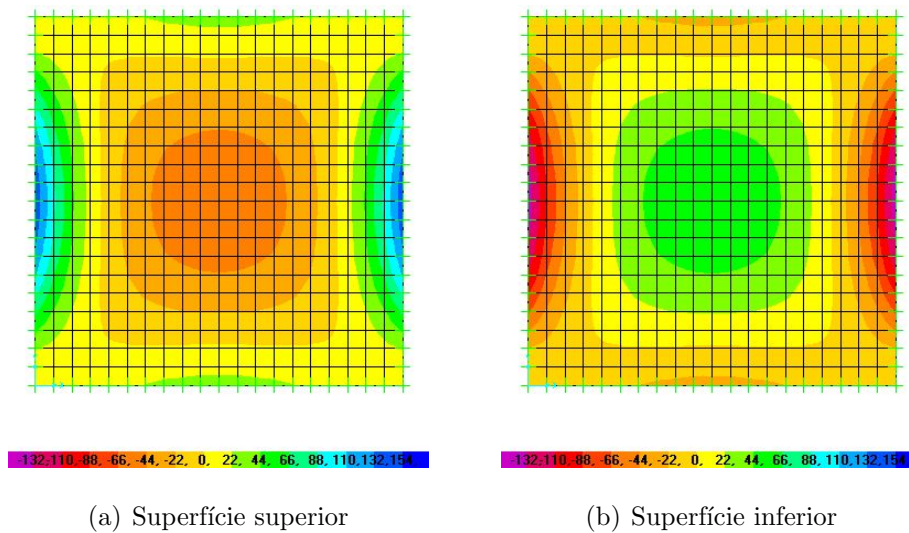


Figura 6.43: Distribuição de tensões ao longo da laje encastrada segundo a direcção xx para o instante $t = 0,03125 s$, pelo SAP2000

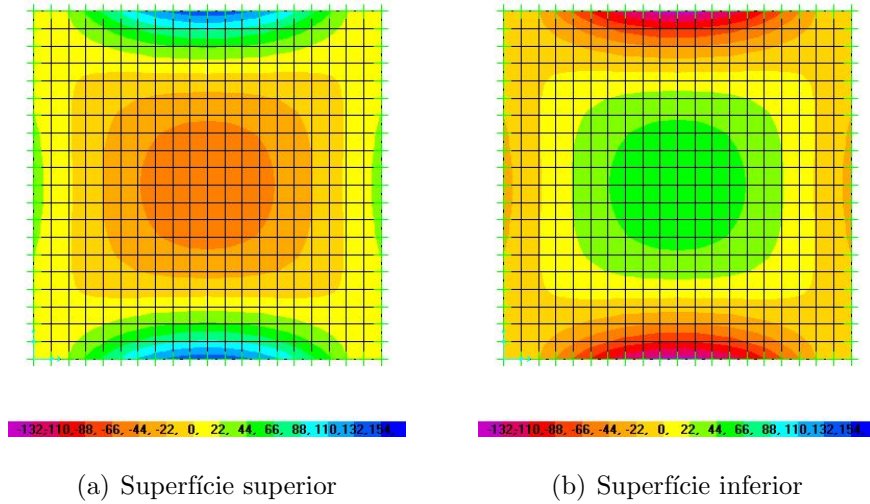


Figura 6.44: Distribuição de tensões ao longo da laje encastrada segundo a direcção yy para o instante $t = 0,03125$ s, pelo SAP2000

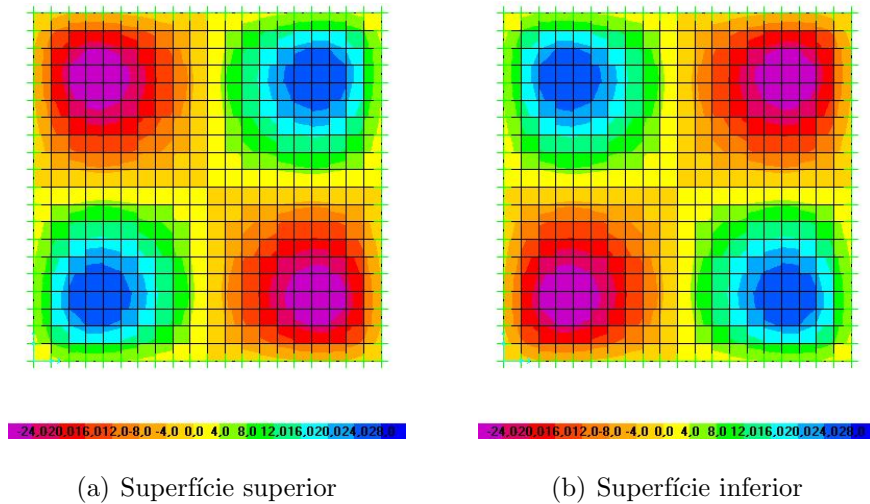


Figura 6.45: Distribuição de tensões tangenciais ao longo da laje encastrada para o instante $t = 0,03125$ s, pelo SAP2000

Analisando os resultados, constatamos, como já seria expectável, que os resultados para as tensões normais do método de Newmark (em FreeFEM++) aproximam-se mais dos resultados do SAP2000, já que o mesmo se verifica com os deslocamentos. No caso das tensões tangenciais os resultados obtidos com método de Euler (em FreeFEM++) mostram-se mais compatíveis com os resultados do SAP2000. Em termos de simetria dos resultados, quer a nível da superfície superior e inferior quer a nível das duas direcções ortogonais, podemos dizer que esta se verifica para os dois métodos embora o método de Newmark apareça como o mais simétrico.

Capítulo 7

Conclusão

7.1 Conclusões e Considerações Finais do TFM

Os objectivos definidos inicialmente consistiam em desenvolver um modelo matemático, a implementar no programa FreeFEM++, que permitisse resolver numericamente problemas dinâmicos de estruturas. Para tal, foi necessário encetar um estudo exaustivo dos princípios e das relações que governam a Teoria da Elasticidade Linear tridimensional e dos métodos de resolução numérica, designadamente o Método de Elementos Finitos e Métodos de Integração Directa, bem como a sua interacção. Assim sendo, as análises foram executadas utilizando como modelo físico, lajes sob diferentes condições de apoio e sujeitas a uma excitação harmónica, simulando a actividade humana sobre a mesma.

Os resultados obtidos em FreeFEM++ foram validados quer por comparação com os resultados obtidos pelo SAP2000, um programa já conceituado no mercado, quer por comparação com as Soluções Teóricas. Com base nas comparações, os resultados obtidos pelo FreeFEM++ são satisfatórios obtendo erros bastante reduzidos nos deslocamentos nos três primeiros casos, aumentando no quarto caso. Esta ocorrência justifica-se pelo aumento do número e o tipo de elementos estruturais o que aumenta a probabilidade de obter erros maiores devido sobretudo ao acréscimo de dificuldade em aproximar os modelos utilizados pelos dois programas, visto que o FreeFEM++ utiliza um modelo tridimensional e o SAP2000 recorre ao modelo simplificado de Kirchoff. Deste modo pode-se dizer que os objectivos foram al-

cançados, pois conseguiram-se implementar dois métodos numéricos distintos para a resolução de problemas dinâmicos: o Método de Euler e o Método de Newmark. Ambos os métodos forneceram bons resultados comparativamente quer aos obtidos em SAP2000, quer às Soluções Teóricas. Uma vez que o SAP2000 também utiliza o Método de Newmark para resolução de problemas dinâmicos já era expectável que houvesse ligeiramente uma maior aproximação com os resultados obtidos no FreeFEM++ utilizando o mesmo método.

Com a utilização do FreeFEM++ conseguiu-se adquirir novos conhecimentos quer a nível dos recursos do FreeFEM++, tendo este se revelado como uma ferramenta eficaz e capaz de simular o comportamento estrutural de lajes, quer a nível do SAP2000, aprendendo a executar análises dinâmicas com base no histórico da resposta de uma estrutura através do Método de Integração Directa. Além disso, com os fundamentos teóricos da presente dissertação conseguiu-se aprofundar e consolidar conhecimentos adquiridos ao longo do curso quer na Licenciatura como no Mestrado, nomeadamente em termos da Teoria da Elasticidade Linear e do Método de Elementos Finitos, e adquirir novos conhecimentos mais direccionados para o campo da dinâmica estrutural.

7.2 Propostas de Estudos Futuros

Alguns temas que se considera pertinentes e relevantes a serem desenvolvidos na sequência desta dissertação são:

- Desenvolvimento de uma estrutura de programação no FreeFEM++ que permita determinar as frequências e os modos de vibração da uma estrutura pela resolução do problema de valores próprios.
- Implementação no FreeFEM++ de modelos para resolução de problemas de não-linearidade.
- Melhoria em termos de imposição das condições de fronteira como por exemplo do tipo Neumann ou Robin.

Bibliografia

- [1] BIEZUNER Rodney Josué - Introdução às Equações Diferenciais Parciais. Minas Gerais: Departamento de Matemática do Instituto de Ciências Exactas da Universidade Federal de Minas Gerais, 2007. [consult. 7 Julho 2010]. Disponível em WWW:

<URL:http://www.mat.ufmg.br/~rodney/notas_de_aula/iedp.pdf>

- [2] CHOPRA, Anil K. - Dynamics of structures: theory and applications to earthquake engineering. 1st ed. Englewood Cliffs, New Jersey: Prentice Hall, 1995. ISBN 0-13855214-2.

- [3] CHRISPELL, John C.; HOWELL, Jason S. - SIAM Student Chapter Seminar. Finite Element Approximation of Partial Differential Equations Using FreeFem++. Clemson: Department of Mathematics Sciences Clemson University, 2007. [consult. 9 Abril 2010]. Disponível em WWW:

<Url:www.ccs.tulane.edu/~jchrispe/PDFdocuments/usc1.pdf>

- [4] CLOUGH, Ray W.; PENZIEN, Joseph - Dynamics of structures. 3rd ed. Berkeley, CA 94704 USA: Computers & Structures, Inc, 1995 University Ave.

- [5] COOK, Robert D.; MALKUS, David S.; PLESHA, Michael E. - Concepts and Applications of Finite Element Analysis. 3rd ed. John Wiley & Sons, Inc, Madison Wisconsin, 1989.

- [6] HECHT, Frédéric - Freefem++, version 3.5. 3.^a edition. Université Pierre et Marie Curie, Laboratoire Jacques-Louis Lions, Paris 2010. Disponível em WWW:

<URL:<http://www.freefem.org/ff++/>>

- [7] DINIS, Lúcia Maria de Jesus Simas - Leis Constitutivas. Porto: Secção de Mecânica Aplicada da Faculdade de Engenharia da Universidade do Porto, 2007. [consult. 15 Junho 2010]. Disponível em WWW:
- <URL:<http://paginas.fe.up.pt/~ldinis/indexport.htm>>
- [8] HUGHES, Thomas J. R. - The Finite Element Method: Linear Static and Dynamic Finite Element Analysis. 1st ed. Englewood Cliffs, New Jersey: Prentice Hall, 1987. ISBN 0-13855214-2.
- [9] JOHNSON Claes - Numerical solution of partial differential equations by the finite element method. 1st ed. Bromley: Cambridge University Press, 1987. ISBN 0-521-347-580.
- [10] LEITÃO, V. M. A. - Apontamentos sobre análise elástica de lajes. Análise de Estruturas I. Lisboa, IST -Instituto Superior Técnico, Departamento de Engenharia Civil, 1996 [consult. 3 Abril 2011]. Disponível em WWW:
- <URL:<http://pt.scribd.com/doc/64702217/Analise-Elastica-Linear-de-Lajes>>
- [11] LIMA, D. V. F. - Controle de Vibrações Induzidas em uma Laje de Academia de Ginástica com a Utilização de Amortecedores de Massa Sintonizados. Dissertação de Mestrado. Brasília: Departamento de Engenharia Civil e Ambiental da Universidade de Brasília, 2007.
- [12] MENDES, Paulo; OLIVEIRA, Sérgio - Análise Dinâmica de Estruturas, Utilização Integrada de Modelos de Identificação Modal e Modelos de Elementos Finitos. Lisboa: Laboratório Nacional de Engenharia Civil (LNEC), 2008.
- [13] MORAIS, Eduardo - Simulação Numérica do Comportamento de Lajes, Aplicação do Método dos Elementos Finitos a Problemas Tridimensionais de Elasticidade Linear usando o FreeFEM++. Dissertação de Mestrado. Lisboa: Departamento de Engenharia Civil do Instituto Superior de Engenharia de Lisboa, 2010.

- [14] MORAIS, Eduardo; GORGULHO, António; RODRIGUES, José Alberto - Avaliação Numérica da Convergência de Solução entre Modelos de Laje. Actas do X Congresso Ibero-americano em Engenharia Mecânica - CIBEM10 Eds. R.M. Natal Jorge et al. Faculdade de Engenharia da Universidade do Porto, Porto, Portugal ISBN: 978-989-96276-2-8, pgs.4011-4014
- [15] OLIVEIRA, Sérgio; CASTRO, A. Tavares de; GOMES, J. Pereira - Capítulo I: Equilíbrios Tridimensionais, Equações Fundamentais da Mecânica Estrutural. Capítulo II: O Método dos Elementos Finitos na Análise Estrutural, Formulação em Deslocamentos. Mecânica dos Sólidos III. Lisboa: Departamento de Engenharia Civil do Instituto Superior de Engenharia de Lisboa, 2009
- [16] SAP2000, Integrated Finite Elements Analysis and Design of Structures: Basic Analysis Reference. Version 7.0. Revised November 1998.
- [17] SAP2000, version 14 - Computers & Structures, Inc. 1995 University Ave. Suite 540, Berkeley, CA 94704 USA.
- [18] SILVA, W.; VASCONCELOS, R. - Desenvolvimento de uma Ferramenta Numérica em Linguagem Delphi Via Método dos Elementos Finitos para Análise de Estruturas de Placas. Mecânica Computacional Vol XXIX, Buenos Aires, Argentina, 2010 pgs. 1729-1746
- [19] TAVARES, João Manuel R. S. - Introdução ao Método de Elementos Finitos. Porto: Departamento de Engenharia Electrotécnica e de Computadores da Faculdade de Engenharia da Universidade do Porto, 1998 [consult. 28 Abril 2011]. Disponível em WWW:

<URL: <http://www.google.com/url?sa=t&rct=j&q=TAVARES>>
- [20] WAIDEMAM, L. - Análise dinâmica de placas delgadas utilizando elementos finitos triangulares e retangulares. 153 p. Dissertação (Mestrado em Engenharia Civil) - Faculdade de Engenharia de Ilha Solteira, Universidade Estadual Paulista, Ilha Solteira, 2004. Disponível em WWW:

<URL: http://www.ppgec.feis.unesp.br/teses/2004/waidemam_leandro_2004.pdf>

Anexo A

Programas para Caso da Laje Encastrada

A.1 Método de Euler

```
load "msh3"
load "medit"
// CONSTRUÇÃO DA MALHA-----
mesh3 Th("dominio1.mesh");
// CONSTANTES DO PROBLEMA-----
int n = 16;

real T = 0.5,
h = 0.03125,
sqrt2 = sqrt(2),
coef = 1.;
// CONSTANTES DO MATERIAL-----
real E = 30e6,
sigma = 0.2,
lambda = E*sigma/((1+sigma)*(1-2*sigma)),
mu = E/(2*(1+sigma)),
gama = 25.,
ro = gama/9.80665;
// MACROS-----
macro epsilon(u1,u2,u3) [dx(u1),dy(u2),dz(u3),(dz(u2)+ dy(u3))/sqrt2,
(dz(u1)+ dx(u3))/sqrt2,(dy(u1)+dx(u2))/sqrt2] // EOM
macro div(u1,u2,u3) ( dx(u1)+dy(u2)+dz(u3) ) // EOM
```

```

// DEFINIÇÃO DO ESPAÇO DE APROXIMAÇÃO_____
fespace Vh(Th,P2);
Vh u1, u2, u3, v1, v2, v3, ua1, ua2, ua3, ub1, ub2, ub3;

fespace Rh(Th,P1);
Rh Sxx, Syy, Sxy;
// RESOLUÇÃO DO PROBLEMA EVOLUTIVO_____
ua1[] = 0.;
ua2[] = 0.;
ua3[] = 0.;
ub1[] = 0.;
ub2[] = 0.;
ub3[] = 0.;
for (int i = 1; i <= n; i++)
{
func fd = -3.14*sin(2*pi*i*h/T);
solve Problema( u1,u2,u3,v1,v2,v3, solver = CG) =
    int3d(Th)( lambda*div(u1,u2,u3)*div(v1,v2,v3)*h*h
        + 2.*mu*(epsilon(u1,u2,u3)'*epsilon(v1,v2,v3))*h*h )
    - int2d(Th,1500)( fd*v3*h*h )
    + int3d(Th)( ([u1,u2,u3]'*[v1,v2,v3])*ro )
    - int3d(Th)( ([ua1,ua2,ua3]'*[v1,v2,v3])*2*ro )
    + int3d(Th)( ([ub1,ub2,ub3]'*[v1,v2,v3])*ro )
    + on(2500, u1 = 0, u2 = 0, u3 = 0.);

mesh3 Thm = movemesh3(Th, transfo = [x+u1*coef, y+u2*coef, z+u3*coef]);
plot (Thm, wait = 1);
// CÁLCULO DE TENSÕES_____
Sxx = lambda*(dx(u1)+dy(u2)+dz(u3)) + 2*mu*dx(u1);
Syy = lambda*(dx(u1)+dy(u2)+dz(u3)) + 2*mu*dy(u2);
Sxy = mu*(dy(u1) + dx(u2));
// GUARDAR RESULTADOS DINÂMICOS_____
string desl;
desl = "d"+i+".pos";
ofstream file1(desl);
file1 << "View \" \"<< "DESLOCAMENTOS EULER t = \" << i*h
    << \" \" {\" << endl;
// REGISTO DAS COORDENADAS DOS VÉRTICES DOS TETRAÉDRICOS_____
for (int k = 0; k < Th.nt; k++)

```

```

{
file1 << "VS(";
    for (int j = 0; j < 3; j++)
        file1 << Th[k][j].x << ","
            << Th[k][j].y << ","
            << Th[k][j].z << ",";
        file1 << Th[k][3].x << ","
            << Th[k][3].y << ","
            << Th[k][3].z;
        file1 << "){" << endl;
// REGISTO DOS DESLOCAMENTOS EM CADA VÉRTICE-----
    for (int j=0; j<3; j++)
        file1 << u1[] [Vh(k,j)] << ","
            << u2[] [Vh(k,j)] << ","
            << u3[] [Vh(k,j)] << "," << endl;
        file1 << u1[] [Vh(k,3)] << ","
            << u2[] [Vh(k,3)] << ","
            << u3[] [Vh(k,3)] << "};" << endl;
}

    file1 << "};" << endl;
// GUARDAR RESULTADOS DINÂMICOS-----
string tensaoxx;
tensaoxx = "txx"+i+".pos";
ofstream file3(tensaoxx);
file3 << "View \" << "TENSAO XX" << i << " \" {" << endl;
// REGISTO DAS COORDENADAS DOS VÉRTICES DOS TETRAÉDRICOS-----
    for (int k = 0; k < Th.nt; k++) // k elemento finito
    {
file3 << "SS(";
        for (int j = 0; j < 3; j++) // j vértice do tetraédrico
            file3 << Th[k][j].x << ","
                << Th[k][j].y << ","
                << Th[k][j].z << ",";
            file3 << Th[k][3].x << ","
                << Th[k][3].y << ","
                << Th[k][3].z;
            file3 << "){" << endl;
// REGISTO DAS TENSÕES XX EM CADA VÉRTICE-----
        for (int j=0; j<3; j++)

```

```

        file3 << Sxx[] [Rh(k,j)]<< "," << endl;
        file3 << Sxx[] [Rh(k,3)] << "};" << endl;
    }
    file3 << "};" << endl;
// GUARDAR RESULTADOS DINÂMICOS_____
string tensaoyy;
tensaoyy = "tyy"+i+".pos";
ofstream file4(tensaoyy);
file4 << "View \" "<< "TENSAO YY" << i << " \" {" << endl;
// REGISTO DAS COORDENADAS DOS VÉRTICES DOS TETRAÉDRICOS_____
    for (int k = 0; k < Th.nt; k++)
    {
        file4 << "SS(";
            for (int j = 0; j < 3; j++)
                file4 << Th[k][j].x << ","
                    << Th[k][j].y << ","
                    << Th[k][j].z << ",";
            file4 << Th[k][3].x << ","
                << Th[k][3].y << ","
                << Th[k][3].z;
            file4 << "){" << endl;
// REGISTO DAS TENSÕES YY EM CADA VÉRTICE_____
        for (int j=0; j<3; j++)
            file4 << Syy[] [Rh(k,j)]<< "," << endl;
            file4 << Syy[] [Rh(k,3)] << "};" << endl;
    }
    file4 << "};" << endl;
// GUARDAR RESULTADOS DINÂMICOS_____
string tensaoxy;
tensaoxy = "txy"+i+".pos";
ofstream file5(tensaoxy);
file5 << "View \" "<< "TENSAO XY" << i << " \" {" << endl;

// REGISTO DAS COORDENADAS DOS VÉRTICES DOS TETRAÉDRICOS_____
    for (int k = 0; k < Th.nt; k++)
    {
        file5 << "SS(";
            for (int j = 0; j < 3; j++)
                file5 << Th[k][j].x << ","

```

```

        << Th[k][j].y << ","
        << Th[k][j].z << ",";
file5 << Th[k][3].x << ","
        << Th[k][3].y << ","
        << Th[k][3].z;
file5 << "){" << endl;
// REGISTO DAS TENSÕES XY EM CADA VÉRTICE-----
    for (int j=0; j<3; j++)
        file5 << Sxy[][Rh(k,j)]<< "," << endl;
        file5 << Sxy[][Rh(k,3)] << "};" << endl;
    }
file5 << "};" << endl;
// ACTUALIZAÇÃO DA SOLUÇÃO-----
ub1[] = ua1[];
ub2[] = ua2[];
ub3[] = ua3[];
ua1[] = u1[];
ua2[] = u2[];
ua3[] = u3[];
}

```

A.2 Método de Newmark

```

load "msh3"
load "medit"
// CONSTRUÇÃO DA MALHA-----
mesh3 Th("dominio1.mesh");
// plot (Th, wait = 1);
// CONSTANTES DO PROBLEMA-----
int    n = 12;

real T = 0.5, h = 0.03125, sqrt2 = sqrt(2),
      gama = 0.5, beta = 0.25,
      b1 = 1./(beta*h*h),    b2 = 1./(beta*h),    b3 = 1/(2*beta)-1,
      b4 = gama*h*b1,       b5 = gama/beta-1,    b6 = (gama*h)/(2*beta)-h;
// CONSTANTES DO MATERIAL-----
real    E = 30.e6,
      sigma = 0.2,

```

```

        lambda = E*sigma/((1+sigma)*(1-2*sigma)),
        mu = E/(2*(1+sigma)),
        rho = 25./9.80665;

// MACROS_-----
macro epsilon(u1,u2,u3) [dx(u1),dy(u2),dz(u3),(dz(u2)+ dy(u3))/sqrt2,
                        (dz(u1)+ dx(u3))/sqrt2,(dy(u1)+dx(u2))/sqrt2] // EOM
macro div(u1,u2,u3) ( dx(u1)+dy(u2)+dz(u3) ) // EOM
// ESPAÇO DE APROXIMAÇÃO_-----
fespace Vh(Th,P2);
        Vh u1,u2,u3,v1,v2,v3,w1,w2,w3,
        pdu1,pdu2,pdu3,
        du1,du2,du3,ddu1,ddu2,ddu3;

// INICIALIZAÇÃO_-----
        u1[]=0.;u2[]=0.;u3[]=0.;
        du1[]=0.; du2[]=0.; du3[]=0.;
        ddu1[]=0.; ddu2[]=0.; ddu3[]=0.;

// INICIO DO CICLO EM i_-----
for (int i = 1; i<=n; i++)
{
solve problema(w1,w2,w3,v1,v2,v3, solver = CG) =
        int3d(Th)( lambda*div(w1,w2,w3)*div(v1,v2,v3)
        + 2.*mu*(epsilon(w1,w2,w3)'*epsilon(v1,v2,v3)))
        + int3d(Th)( b1*rho*(w1*v1 + w2*v2 + w3*v3) )
        - int2d(Th,1500)( (-3.14*sin(2*pi*i*h/T))*v3 )
        - int3d(Th)( b1*rho*(u1*v1 + u2*v2 + u3*v3) )
        + int3d(Th)( b3*rho*(ddu1*v1 + ddu2*v2 + ddu3*v3) )
        + on(2500, w1 = 0., w2 = 0., w3 = 0.);

// CORREÇÃO
for (int n = 0; n < Vh.ndof; n++)
{
pdu1[] [n] = b4*(w1[] [n]-u1[] [n]) + b5*du1[] [n] + b6*ddu1[] [n];
pdu2[] [n] = b4*(w2[] [n]-u2[] [n]) + b5*du2[] [n] + b6*ddu2[] [n];
pdu3[] [n] = b4*(w3[] [n]-u3[] [n]) + b5*du3[] [n] + b6*ddu3[] [n];

ddu1[] [n] = b1*(w1[] [n]-u1[] [n]) + b3*ddu1[] [n];
ddu2[] [n] = b1*(w2[] [n]-u2[] [n]) + b3*ddu2[] [n];
ddu3[] [n] = b1*(w3[] [n]-u3[] [n]) + b3*ddu3[] [n];
}
du1[] = pdu1[]; du2[] = pdu2[]; du3[] = pdu3[];

```

```

u1[] = w1[]; u2[] = w2[]; u3[] = w3[];

// GUARDAR RESULTADOS DINÂMICOS_____
string resul;
resul = "d"+i+".pos";
ofstream file(resul);
file << "View \n " << "DESLOCAMENTOS NEWMARK t = "
    << i*h << " \n {" << endl;
// REGISTO DAS COORDENADAS DOS VÉRTICES DOS TETRAEDRICOS_____

    for (int k=0; k<Th.nt; k++)
    {
        file << "VS(";

        for (int j=0; j<3; j++)
            file << Th[k][j].x << ", "
                << Th[k][j].y << ", "
                << Th[k][j].z << ", ";
file << Th[k][3].x << ", "
    << Th[k][3].y << ", "
    << Th[k][3].z;
file << "){" << endl;

// REGISTO DOS DESLOCAMENTOS EM CADA VÉRTICE_____

        for (int j=0; j<3; j++)
file << u1[][Vh(k,j)] << ", "
    << u2[][Vh(k,j)] << ", "
    << u3[][Vh(k,j)] << ", " << endl;
        file << u1[][Vh(k,3)] << ", "
            << u2[][Vh(k,3)] << ", "
            << u3[][Vh(k,3)] << "};" << endl;
    }

file << "};" << endl;
}

```

Anexo B

Programas para Caso da Laje Simplesmente Apoiada

B.1 Método de Euler

```
load "msh3"
load "medit"
// CONSTRUÇÃO DA MALHA-----
mesh3 Th("dominio2.mesh");
// CONSTANTES DO PROBLEMA-----
int n = 16;

real T = 0.5,
h = 0.03125,
sqrt2 = sqrt(2),
coef = 1.;
// CONSTANTES DO MATERIAL-----
real E = 30e6,
sigma = 0.2,
lambda = E*sigma/((1+sigma)*(1-2*sigma)),
mu = E/(2*(1+sigma)),
gama = 25.,
ro = gama/9.80665;
// MACROS-----
macro epsilon(u1,u2,u3) [dx(u1),dy(u2),dz(u3),(dz(u2)+ dy(u3))/sqrt2,
(dz(u1)+ dx(u3))/sqrt2,(dy(u1)+dx(u2))/sqrt2] // EOM
macro div(u1,u2,u3) ( dx(u1)+dy(u2)+dz(u3) ) // EOM
```

```

// DEFINIÇÃO DO ESPAÇO DE APROXIMAÇÃO_____
fespace Vh(Th,P2);
Vh u1, u2, u3, v1, v2, v3, ua1, ua2, ua3, ub1, ub2, ub3;

fespace Rh(Th,P1);
Rh Sxx, Syy, Sxy;
// RESOLUÇÃO DO PROBLEMA EVOLUTIVO_____
ua1[] = 0.;
ua2[] = 0.;
ua3[] = 0.;
ub1[] = 0.;
ub2[] = 0.;
ub3[] = 0.;
for (int i = 1; i <= n; i++)
{
func fd = -3.14*sin(2*pi*i*h/T);
solve Problema( u1,u2,u3,v1,v2,v3, solver = CG) =
    int3d(Th)( lambda*div(u1,u2,u3)*div(v1,v2,v3)*h*h
        + 2.*mu*(epsilon(u1,u2,u3)'*epsilon(v1,v2,v3))*h*h )
    - int2d(Th,1500)( fd*v3*h*h )
    + int3d(Th)( ([u1,u2,u3]'*[v1,v2,v3])*ro )
    - int3d(Th)( ([ua1,ua2,ua3]'*[v1,v2,v3])*2*ro )
    + int3d(Th)( ([ub1,ub2,ub3]'*[v1,v2,v3])*ro )
    + on(2500, u3 = 0.);

mesh3 Thm = movemesh3(Th, transfo = [x+u1*coef, y+u2*coef, z+u3*coef]);
plot (Thm, wait = 1);
// CÁLCULO DE TENSÕES_____
Sxx = lambda*(dx(u1)+dy(u2)+dz(u3)) + 2*mu*dx(u1);
Syy = lambda*(dx(u1)+dy(u2)+dz(u3)) + 2*mu*dy(u2);
Sxy = mu*(dy(u1) + dx(u2));
// GUARDAR RESULTADOS DINÂMICOS_____
string desl;
desl = "d"+i+".pos";
ofstream file1(desl);
file1 << "View \" \"<< "DESLOCAMENTOS EULER t = \" << i*h
    << \" \" {\" << endl;
// REGISTO DAS COORDENADAS DOS VÉRTICES DOS TETRAÉDRICOS_____
for (int k = 0; k < Th.nt; k++)

```

```

{
file1 << "VS(";
    for (int j = 0; j < 3; j++)
        file1 << Th[k][j].x << ","
            << Th[k][j].y << ","
            << Th[k][j].z << ",";
        file1 << Th[k][3].x << ","
            << Th[k][3].y << ","
            << Th[k][3].z;
        file1 << "){" << endl;
// REGISTO DOS DESLOCAMENTOS EM CADA VÉRTICE-----
    for (int j=0; j<3; j++)
        file1 << u1[] [Vh(k,j)] << ","
            << u2[] [Vh(k,j)] << ","
            << u3[] [Vh(k,j)] << "," << endl;
        file1 << u1[] [Vh(k,3)] << ","
            << u2[] [Vh(k,3)] << ","
            << u3[] [Vh(k,3)] << "};" << endl;
}

    file1 << "};" << endl;
// GUARDAR RESULTADOS DINÂMICOS-----
string tensaoxx;
tensaoxx = "txx"+i+".pos";
ofstream file3(tensaoxx);
file3 << "View \" << "TENSAO XX" << i << " \" {" << endl;
// REGISTO DAS COORDENADAS DOS VÉRTICES DOS TETRAÉDRICOS-----
    for (int k = 0; k < Th.nt; k++) // k elemento finito
    {
file3 << "SS(";
        for (int j = 0; j < 3; j++) // j vértice do tetraédrico
            file3 << Th[k][j].x << ","
                << Th[k][j].y << ","
                << Th[k][j].z << ",";
            file3 << Th[k][3].x << ","
                << Th[k][3].y << ","
                << Th[k][3].z;
            file3 << "){" << endl;
// REGISTO DAS TENSÕES XX EM CADA VÉRTICE-----
        for (int j=0; j<3; j++)

```

```

        file3 << Sxx[] [Rh(k,j)]<< "," << endl;
        file3 << Sxx[] [Rh(k,3)] << "};" << endl;
    }
    file3 << "};" << endl;
// GUARDAR RESULTADOS DINÂMICOS_____
string tensaoyy;
tensaoyy = "tyy"+i+".pos";
ofstream file4(tensaoyy);
file4 << "View \" "<< "TENSAO YY" << i << " \" {" << endl;
// REGISTO DAS COORDENADAS DOS VÉRTICES DOS TETRAÉDRICOS_____
    for (int k = 0; k < Th.nt; k++)
    {
        file4 << "SS(";
            for (int j = 0; j < 3; j++)
                file4 << Th[k][j].x << ","
                    << Th[k][j].y << ","
                    << Th[k][j].z << ",";
            file4 << Th[k][3].x << ","
                << Th[k][3].y << ","
                << Th[k][3].z;
            file4 << "){" << endl;
// REGISTO DAS TENSÕES YY EM CADA VÉRTICE_____
        for (int j=0; j<3; j++)
            file4 << Syy[] [Rh(k,j)]<< "," << endl;
            file4 << Syy[] [Rh(k,3)] << "};" << endl;
    }
    file4 << "};" << endl;
// GUARDAR RESULTADOS DINÂMICOS_____
string tensaoxy;
tensaoxy = "txy"+i+".pos";
ofstream file5(tensaoxy);
file5 << "View \" "<< "TENSAO XY" << i << " \" {" << endl;

// REGISTO DAS COORDENADAS DOS VÉRTICES DOS TETRAÉDRICOS_____
    for (int k = 0; k < Th.nt; k++)
    {
        file5 << "SS(";
            for (int j = 0; j < 3; j++)
                file5 << Th[k][j].x << ","

```

```

        << Th[k][j].y << ","
        << Th[k][j].z << ",";
file5 << Th[k][3].x << ","
        << Th[k][3].y << ","
        << Th[k][3].z;
file5 << "}" << endl;
// REGISTO DAS TENSÕES XY EM CADA VÉRTICE-----
for (int j=0; j<3; j++)
    file5 << Sxy[][Rh(k,j)] << "," << endl;
    file5 << Sxy[][Rh(k,3)] << "};" << endl;
}
file5 << "};" << endl;
// ACTUALIZAÇÃO DA SOLUÇÃO-----
ub1[] = ua1[];
ub2[] = ua2[];
ub3[] = ua3[];
ua1[] = u1[];
ua2[] = u2[];
ua3[] = u3[];
}

```

B.2 Método de Newmark

```

load "msh3"
load "medit"
// CONSTRUÇÃO DA MALHA-----
mesh3 Th("dominio2.mesh");
// plot (Th, wait = 1);
// CONSTANTES DO PROBLEMA-----
int    n = 12;

real T = 0.5, h = 0.03125, sqrt2 = sqrt(2),
      gama = 0.5, beta = 0.25,
      b1 = 1./(beta*h*h),   b2 = 1./(beta*h),   b3 = 1/(2*beta)-1,
      b4 = gama*h*b1,      b5 = gama/beta-1,   b6 = (gama*h)/(2*beta)-h;
// CONSTANTES DO MATERIAL-----
real    E = 30.e6,
      sigma = 0.2,
      lambda = E*sigma/((1+sigma)*(1-2*sigma)),

```

```

        mu = E/(2*(1+sigma)),
        rho = 25./9.80665;
// MACROS-----
macro epsilon(u1,u2,u3) [dx(u1),dy(u2),dz(u3),(dz(u2)+ dy(u3))/sqrt2,
                        (dz(u1)+ dx(u3))/sqrt2,(dy(u1)+dx(u2))/sqrt2] // EOM
macro div(u1,u2,u3) ( dx(u1)+dy(u2)+dz(u3) ) // EOM
// ESPAÇO DE APROXIMAÇÃO-----
fespace Vh(Th,P2);
        Vh u1,u2,u3,v1,v2,v3,w1,w2,w3,
        pdu1,pdu2,pdu3,
        du1,du2,du3,ddu1,ddu2,ddu3;
// INICIALIZAÇÃO-----
        u1[]=0.;u2[]=0.;u3[]=0.;
        du1[]=0.; du2[]=0.; du3[]=0.;
        ddu1[]=0.; ddu2[]=0.; ddu3[]=0.;
// INICIO DO CICLO EM i-----
for (int i = 1; i<=n; i++)
{
solve problema(w1,w2,w3,v1,v2,v3, solver = CG) =
        int3d(Th)( lambda*div(w1,w2,w3)*div(v1,v2,v3)
        + 2.*mu*(epsilon(w1,w2,w3)*epsilon(v1,v2,v3))
        + int3d(Th)( b1*rho*(w1*v1 + w2*v2 + w3*v3) )
        - int2d(Th,1500)( (-3.14*sin(2*pi*i*h/T))*v3 )
        - int3d(Th)( b1*rho*(u1*v1 + u2*v2 + u3*v3) )
        + int3d(Th)( b3*rho*(ddu1*v1 + ddu2*v2 + ddu3*v3) )
        + on(2500, w3 = 0.);
// CORREÇÃO
for (int n = 0; n < Vh.ndof; n++)
{
pdu1[] [n] = b4*(w1[] [n]-u1[] [n]) + b5*du1[] [n] + b6*ddu1[] [n];
pdu2[] [n] = b4*(w2[] [n]-u2[] [n]) + b5*du2[] [n] + b6*ddu2[] [n];
pdu3[] [n] = b4*(w3[] [n]-u3[] [n]) + b5*du3[] [n] + b6*ddu3[] [n];

ddu1[] [n] = b1*(w1[] [n]-u1[] [n]) + b3*ddu1[] [n];
ddu2[] [n] = b1*(w2[] [n]-u2[] [n]) + b3*ddu2[] [n];
ddu3[] [n] = b1*(w3[] [n]-u3[] [n]) + b3*ddu3[] [n];
}
du1[] = pdu1[]; du2[] = pdu2[]; du3[] = pdu3[];
u1[] = w1[]; u2[] = w2[]; u3[] = w3[];

```

```

// GUARDAR RESULTADOS DINÂMICOS_____
string  resul;
resul = "d"+i+".pos";
ofstream file(resul);
file << "View \n " << "DESLOCAMENTOS NEWMARK t = "
    << i*h << " \n {" << endl;
// REGISTO DAS COORDENADAS DOS VÉRTICES DOS TETRAÉDRICOS_____

    for (int k=0; k<Th.nt; k++)
    {
        file << "VS(";

        for (int j=0; j<3; j++)
            file << Th[k][j].x << ", "
                << Th[k][j].y << ", "
                << Th[k][j].z << ", ";
file << Th[k][3].x << ", "
    << Th[k][3].y << ", "
    << Th[k][3].z;
file << "){" << endl;

// REGISTO DOS DESLOCAMENTOS EM CADA VÉRTICE_____

        for (int j=0; j<3; j++)
file << u1[][Vh(k,j)] << ", "
    << u2[][Vh(k,j)] << ", "
    << u3[][Vh(k,j)] << ", " << endl;
        file << u1[][Vh(k,3)] << ", "
            << u2[][Vh(k,3)] << ", "
            << u3[][Vh(k,3)] << "};" << endl;
    }

file << "};" << endl;
}

```

Anexo C

Programas para Caso da Laje Vigada sobre Pilares Rígidos

C.1 Método de Euler

```
load "msh3"
load "medit"
// CONSTRUÇÃO DA MALHA-----
mesh3 Th("dominio3.mesh");
// CONSTANTES DO PROBLEMA-----
int n = 16;

real T = 0.5,
h = 0.03125,
sqrt2 = sqrt(2),
coef = 1.;
// CONSTANTES DO MATERIAL-----
real E = 30e6,
sigma = 0.2,
lambda = E*sigma/((1+sigma)*(1-2*sigma)),
mu = E/(2*(1+sigma)),
gama = 25.,
ro = gama/9.80665;
// MACROS-----
macro epsilon(u1,u2,u3) [dx(u1),dy(u2),dz(u3),(dz(u2)+ dy(u3))/sqrt2,
(dz(u1)+ dx(u3))/sqrt2,(dy(u1)+dx(u2))/sqrt2] // EOM
macro div(u1,u2,u3) ( dx(u1)+dy(u2)+dz(u3) ) // EOM
```

```

// DEFINIÇÃO DO ESPAÇO DE APROXIMAÇÃO_____
fespace Vh(Th,P2);
Vh u1, u2, u3, v1, v2, v3, ua1, ua2, ua3, ub1, ub2, ub3;

fespace Rh(Th,P1);
Rh Sxx, Syy, Sxy;
// RESOLUÇÃO DO PROBLEMA EVOLUTIVO_____
ua1[] = 0.;
ua2[] = 0.;
ua3[] = 0.;
ub1[] = 0.;
ub2[] = 0.;
ub3[] = 0.;
for (int i = 1; i <= n; i++)
{
func fd = -3.14*sin(2*pi*i*h/T);
solve Problema( u1,u2,u3,v1,v2,v3, solver = CG) =
    int3d(Th)( lambda*div(u1,u2,u3)*div(v1,v2,v3)*h*h
        + 2.*mu*(epsilon(u1,u2,u3)'*epsilon(v1,v2,v3))*h*h )
    - int2d(Th,1500)( fd*v3*h*h )
    + int3d(Th)( ([u1,u2,u3]'*[v1,v2,v3])*ro )
    - int3d(Th)( ([ua1,ua2,ua3]'*[v1,v2,v3])*2*ro )
    + int3d(Th)( ([ub1,ub2,ub3]'*[v1,v2,v3])*ro )
    + on(2500, u1 = 0., u2 = 0., u3 = 0.);

mesh3 Thm = movemesh3(Th, transfo = [x+u1*coef, y+u2*coef, z+u3*coef]);
plot (Thm, wait = 1);
// CÁLCULO DE TENSÕES_____
Sxx = lambda*(dx(u1)+dy(u2)+dz(u3)) + 2*mu*dx(u1);
Syy = lambda*(dx(u1)+dy(u2)+dz(u3)) + 2*mu*dy(u2);
Sxy = mu*(dy(u1) + dx(u2));
// GUARDAR RESULTADOS DINÂMICOS_____
string desl;
desl = "d"+i+".pos";
ofstream file1(desl);
file1 << "View \" \"<< "DESLOCAMENTOS EULER t = \" << i*h
    << \" \" {\" << endl;
// REGISTO DAS COORDENADAS DOS VÉRTICES DOS TETRAÉDRICOS_____
for (int k = 0; k < Th.nt; k++)

```

```

{
file1 << "VS(";
    for (int j = 0; j < 3; j++)
        file1 << Th[k][j].x << ","
            << Th[k][j].y << ","
            << Th[k][j].z << ",";
        file1 << Th[k][3].x << ","
            << Th[k][3].y << ","
            << Th[k][3].z;
        file1 << "){" << endl;
// REGISTO DOS DESLOCAMENTOS EM CADA VÉRTICE-----
    for (int j=0; j<3; j++)
        file1 << u1[] [Vh(k,j)] << ","
            << u2[] [Vh(k,j)] << ","
            << u3[] [Vh(k,j)] << "," << endl;
        file1 << u1[] [Vh(k,3)] << ","
            << u2[] [Vh(k,3)] << ","
            << u3[] [Vh(k,3)] << "};" << endl;
}

    file1 << "};" << endl;
// GUARDAR RESULTADOS DINÂMICOS-----
string tensaoxx;
tensaoxx = "txx"+i+".pos";
ofstream file3(tensaoxx);
file3 << "View \" << "TENSAO XX" << i << " \" {" << endl;
// REGISTO DAS COORDENADAS DOS VÉRTICES DOS TETRAÉDRICOS-----
    for (int k = 0; k < Th.nt; k++) // k elemento finito
    {
file3 << "SS(";
        for (int j = 0; j < 3; j++) // j vértice do tetraédrico
            file3 << Th[k][j].x << ","
                << Th[k][j].y << ","
                << Th[k][j].z << ",";
            file3 << Th[k][3].x << ","
                << Th[k][3].y << ","
                << Th[k][3].z;
            file3 << "){" << endl;
// REGISTO DAS TENSÕES XX EM CADA VÉRTICE-----
        for (int j=0; j<3; j++)

```

```

        file3 << Sxx[] [Rh(k,j)]<< "," << endl;
        file3 << Sxx[] [Rh(k,3)] << "};" << endl;
    }
    file3 << "};" << endl;
// GUARDAR RESULTADOS DINÂMICOS_____
string tensaoyy;
tensaoyy = "tyy"+i+".pos";
ofstream file4(tensaoyy);
file4 << "View \" "<< "TENSAO YY" << i << " \" {" << endl;
// REGISTO DAS COORDENADAS DOS VÉRTICES DOS TETRAÉDRICOS_____
    for (int k = 0; k < Th.nt; k++)
    {
        file4 << "SS(";
            for (int j = 0; j < 3; j++)
                file4 << Th[k][j].x << ","
                    << Th[k][j].y << ","
                    << Th[k][j].z << ",";
            file4 << Th[k][3].x << ","
                << Th[k][3].y << ","
                << Th[k][3].z;
            file4 << "){" << endl;
// REGISTO DAS TENSÕES YY EM CADA VÉRTICE_____
        for (int j=0; j<3; j++)
            file4 << Syy[] [Rh(k,j)]<< "," << endl;
            file4 << Syy[] [Rh(k,3)] << "};" << endl;
    }
    file4 << "};" << endl;
// GUARDAR RESULTADOS DINÂMICOS_____
string tensaoxy;
tensaoxy = "txy"+i+".pos";
ofstream file5(tensaoxy);
file5 << "View \" "<< "TENSAO XY" << i << " \" {" << endl;

// REGISTO DAS COORDENADAS DOS VÉRTICES DOS TETRAÉDRICOS_____
    for (int k = 0; k < Th.nt; k++)
    {
        file5 << "SS(";
            for (int j = 0; j < 3; j++)
                file5 << Th[k][j].x << ","

```

```

        << Th[k][j].y << ","
        << Th[k][j].z << ",";
file5 << Th[k][3].x << ","
        << Th[k][3].y << ","
        << Th[k][3].z;
file5 << "){ " << endl;
// REGISTO DAS TENSÕES XY EM CADA VÉRTICE-----
    for (int j=0; j<3; j++)
        file5 << Sxy[][Rh(k,j)] << "," << endl;
        file5 << Sxy[][Rh(k,3)] << "};" << endl;
    }
file5 << "};" << endl;
// ACTUALIZAÇÃO DA SOLUÇÃO-----
ub1[] = ua1[];
ub2[] = ua2[];
ub3[] = ua3[];
ua1[] = u1[];
ua2[] = u2[];
ua3[] = u3[];
}

```

C.2 Método de Newmark

```

load "msh3"
load "medit"
// CONSTRUÇÃO DA MALHA-----
mesh3 Th("dominio3.mesh");
// plot (Th, wait = 1);
// CONSTANTES DO PROBLEMA-----
int    n = 12;

real T = 0.5, h = 0.03125, sqrt2 = sqrt(2),
      gama = 0.5, beta = 0.25,
      b1 = 1./(beta*h*h),    b2 = 1./(beta*h),    b3 = 1/(2*beta)-1,
      b4 = gama*h*b1,       b5 = gama/beta-1,    b6 = (gama*h)/(2*beta)-h;
// CONSTANTES DO MATERIAL-----
real    E = 30.e6,
      sigma = 0.2,

```

```

        lambda = E*sigma/((1+sigma)*(1-2*sigma)),
        mu = E/(2*(1+sigma)),
        rho = 25./9.80665;

// MACROS_-----
macro epsilon(u1,u2,u3) [dx(u1),dy(u2),dz(u3),(dz(u2)+ dy(u3))/sqrt2,
                        (dz(u1)+ dx(u3))/sqrt2,(dy(u1)+dx(u2))/sqrt2] // EOM
macro div(u1,u2,u3) ( dx(u1)+dy(u2)+dz(u3) ) // EOM
// ESPAÇO DE APROXIMAÇÃO_-----
fespace Vh(Th,P2);
        Vh u1,u2,u3,v1,v2,v3,w1,w2,w3,
        pdu1,pdu2,pdu3,
        du1,du2,du3,ddu1,ddu2,ddu3;

// INICIALIZAÇÃO_-----
        u1[]=0.;u2[]=0.;u3[]=0.;
        du1[]=0.; du2[]=0.; du3[]=0.;
        ddu1[]=0.; ddu2[]=0.; ddu3[]=0.;

// INICIO DO CICLO EM i_-----
for (int i = 1; i<=n; i++)
{
solve problema(w1,w2,w3,v1,v2,v3, solver = CG) =
        int3d(Th)( lambda*div(w1,w2,w3)*div(v1,v2,v3)
        + 2.*mu*(epsilon(w1,w2,w3)'*epsilon(v1,v2,v3)))
        + int3d(Th)( b1*rho*(w1*v1 + w2*v2 + w3*v3) )
        - int2d(Th,1500)( (-3.14*sin(2*pi*i*h/T))*v3 )
        - int3d(Th)( b1*rho*(u1*v1 + u2*v2 + u3*v3) )
        + int3d(Th)( b3*rho*(ddu1*v1 + ddu2*v2 + ddu3*v3) )
        + on(2500, w1 = 0., w2 = 0., w3 = 0.);

// CORREÇÃO
for (int n = 0; n < Vh.ndof; n++)
{
pdu1[] [n] = b4*(w1[] [n]-u1[] [n]) + b5*du1[] [n] + b6*ddu1[] [n];
pdu2[] [n] = b4*(w2[] [n]-u2[] [n]) + b5*du2[] [n] + b6*ddu2[] [n];
pdu3[] [n] = b4*(w3[] [n]-u3[] [n]) + b5*du3[] [n] + b6*ddu3[] [n];

ddu1[] [n] = b1*(w1[] [n]-u1[] [n]) + b3*ddu1[] [n];
ddu2[] [n] = b1*(w2[] [n]-u2[] [n]) + b3*ddu2[] [n];
ddu3[] [n] = b1*(w3[] [n]-u3[] [n]) + b3*ddu3[] [n];
}
du1[] = pdu1[]; du2[] = pdu2[]; du3[] = pdu3[];

```

```

u1[] = w1[]; u2[] = w2[]; u3[] = w3[];

// GUARDAR RESULTADOS DINÂMICOS_____
string resul;
resul = "d"+i+".pos";
ofstream file(resul);
file << "View \n " << "DESLOCAMENTOS NEWMARK t = "
    << i*h << " \n {" << endl;
// REGISTO DAS COORDENADAS DOS VÉRTICES DOS TETRAÉDRICOS_____

    for (int k=0; k<Th.nt; k++)
    {
        file << "VS(";

        for (int j=0; j<3; j++)
            file << Th[k][j].x << ", "
                << Th[k][j].y << ", "
                << Th[k][j].z << ", ";
file << Th[k][3].x << ", "
    << Th[k][3].y << ", "
    << Th[k][3].z;
file << "){" << endl;

// REGISTO DOS DESLOCAMENTOS EM CADA VÉRTICE_____

        for (int j=0; j<3; j++)
file << u1[][Vh(k,j)] << ", "
    << u2[][Vh(k,j)] << ", "
    << u3[][Vh(k,j)] << ", " << endl;
        file << u1[][Vh(k,3)] << ", "
            << u2[][Vh(k,3)] << ", "
            << u3[][Vh(k,3)] << "};" << endl;
    }

file << "};" << endl;
}

```

Anexo D

Programas para Caso da Laje Fungiforme

D.1 Método de Euler

```
load "msh3"
load "medit"
// CONSTRUÇÃO DA MALHA-----
mesh3 Th("dominio4.mesh");
// CONSTANTES DO PROBLEMA-----
int n = 16;

real T = 0.5,
h = 0.03125,
sqrt2 = sqrt(2),
coef = 1.;
// CONSTANTES DO MATERIAL-----
real E = 30e6,
sigma = 0.2,
lambda = E*sigma/((1+sigma)*(1-2*sigma)),
mu = E/(2*(1+sigma)),
gama = 25.,
ro = gama/9.80665;
// MACROS-----
macro epsilon(u1,u2,u3) [dx(u1),dy(u2),dz(u3),(dz(u2)+ dy(u3))/sqrt2,
(dz(u1)+ dx(u3))/sqrt2,(dy(u1)+dx(u2))/sqrt2] // EOM
macro div(u1,u2,u3) ( dx(u1)+dy(u2)+dz(u3) ) // EOM
```

```

// DEFINIÇÃO DO ESPAÇO DE APROXIMAÇÃO-----
fespace Vh(Th,P2);
Vh u1, u2, u3, v1, v2, v3, ua1, ua2, ua3, ub1, ub2, ub3;

fespace Rh(Th,P1);
Rh Sxx, Syy, Sxy;
// RESOLUÇÃO DO PROBLEMA EVOLUTIVO-----
ua1[] = 0.;
ua2[] = 0.;
ua3[] = 0.;
ub1[] = 0.;
ub2[] = 0.;
ub3[] = 0.;
for (int i = 1; i <= n; i++)
{
func fd = -3.14*sin(2*pi*i*h/T);
solve Problema( u1,u2,u3,v1,v2,v3, solver = CG) =
    int3d(Th)( lambda*div(u1,u2,u3)*div(v1,v2,v3)*h*h
        + 2.*mu*(epsilon(u1,u2,u3)'*epsilon(v1,v2,v3))*h*h )
    - int2d(Th,1500)( fd*v3*h*h )
    + int3d(Th)( ([u1,u2,u3]'*[v1,v2,v3])*ro )
    - int3d(Th)( ([ua1,ua2,ua3]'*[v1,v2,v3])*2*ro )
    + int3d(Th)( ([ub1,ub2,ub3]'*[v1,v2,v3])*ro )
    + on(2500, u1 = 0, u2 = 0, u3 = 0.);

mesh3 Thm = movemesh3(Th, transfo = [x+u1*coef, y+u2*coef, z+u3*coef]);
plot (Thm, wait = 1);
// CÁLCULO DE TENSÕES-----
Sxx = lambda*(dx(u1)+dy(u2)+dz(u3)) + 2*mu*dx(u1);
Syy = lambda*(dx(u1)+dy(u2)+dz(u3)) + 2*mu*dy(u2);
Sxy = mu*(dy(u1) + dx(u2));
// GUARDAR RESULTADOS DINÂMICOS-----
string desl;
desl = "d"+i+".pos";
ofstream file1(desl);
file1 << "View \" \"<< "DESLOCAMENTOS EULER t = \" \"<< i*h
    << \" \" {\" << endl;
// REGISTO DAS COORDENADAS DOS VÉRTICES DOS TETRAÉDRICOS-----
for (int k = 0; k < Th.nt; k++)

```

```

{
file1 << "VS(";
    for (int j = 0; j < 3; j++)
        file1 << Th[k][j].x << ","
            << Th[k][j].y << ","
            << Th[k][j].z << ",";
        file1 << Th[k][3].x << ","
            << Th[k][3].y << ","
            << Th[k][3].z;
        file1 << "){" << endl;
// REGISTO DOS DESLOCAMENTOS EM CADA VÉRTICE-----
    for (int j=0; j<3; j++)
        file1 << u1[] [Vh(k,j)] << ","
            << u2[] [Vh(k,j)] << ","
            << u3[] [Vh(k,j)] << "," << endl;
        file1 << u1[] [Vh(k,3)] << ","
            << u2[] [Vh(k,3)] << ","
            << u3[] [Vh(k,3)] << "};" << endl;
}

    file1 << "};" << endl;
// GUARDAR RESULTADOS DINÂMICOS-----
string tensaoxx;
tensaoxx = "txx"+i+".pos";
ofstream file3(tensaoxx);
file3 << "View \" << "TENSAO XX" << i << " \" {" << endl;
// REGISTO DAS COORDENADAS DOS VÉRTICES DOS TETRAÉDRICOS-----
    for (int k = 0; k < Th.nt; k++) // k elemento finito
    {
file3 << "SS(";
        for (int j = 0; j < 3; j++) // j vértice do tetraédrico
            file3 << Th[k][j].x << ","
                << Th[k][j].y << ","
                << Th[k][j].z << ",";
            file3 << Th[k][3].x << ","
                << Th[k][3].y << ","
                << Th[k][3].z;
            file3 << "){" << endl;
// REGISTO DAS TENSÕES XX EM CADA VÉRTICE-----
        for (int j=0; j<3; j++)

```

```

        file3 << Sxx[] [Rh(k,j)]<< "," << endl;
        file3 << Sxx[] [Rh(k,3)] << "};" << endl;
    }
    file3 << "};" << endl;
// GUARDAR RESULTADOS DINÂMICOS_____
string tensaoyy;
tensaoyy = "tyy"+i+".pos";
ofstream file4(tensaoyy);
file4 << "View \" "<< "TENSAO YY" << i << " \" {" << endl;
// REGISTO DAS COORDENADAS DOS VÉRTICES DOS TETRAÉDRICOS_____
    for (int k = 0; k < Th.nt; k++)
    {
        file4 << "SS(";
            for (int j = 0; j < 3; j++)
                file4 << Th[k][j].x << ","
                    << Th[k][j].y << ","
                    << Th[k][j].z << ",";
            file4 << Th[k][3].x << ","
                << Th[k][3].y << ","
                << Th[k][3].z;
            file4 << "){" << endl;
// REGISTO DAS TENSÕES YY EM CADA VÉRTICE_____
        for (int j=0; j<3; j++)
            file4 << Syy[] [Rh(k,j)]<< "," << endl;
            file4 << Syy[] [Rh(k,3)] << "};" << endl;
    }
    file4 << "};" << endl;
// GUARDAR RESULTADOS DINÂMICOS_____
string tensaoxy;
tensaoxy = "txy"+i+".pos";
ofstream file5(tensaoxy);
file5 << "View \" "<< "TENSAO XY" << i << " \" {" << endl;

// REGISTO DAS COORDENADAS DOS VÉRTICES DOS TETRAÉDRICOS_____
    for (int k = 0; k < Th.nt; k++)
    {
        file5 << "SS(";
            for (int j = 0; j < 3; j++)
                file5 << Th[k][j].x << ","

```

```

        << Th[k][j].y << ","
        << Th[k][j].z << ",";
file5 << Th[k][3].x << ","
        << Th[k][3].y << ","
        << Th[k][3].z;
file5 << "){" << endl;
// REGISTO DAS TENSÕES XY EM CADA VÉRTICE-----
    for (int j=0; j<3; j++)
        file5 << Sxy[][Rh(k,j)]<< "," << endl;
        file5 << Sxy[][Rh(k,3)] << "};" << endl;
    }
file5 << "};" << endl;
// ACTUALIZAÇÃO DA SOLUÇÃO-----
ub1[] = ua1[];
ub2[] = ua2[];
ub3[] = ua3[];
ua1[] = u1[];
ua2[] = u2[];
ua3[] = u3[];
}

```

D.2 Método de Newmark

```

load "msh3"
load "medit"
// CONSTRUÇÃO DA MALHA-----
mesh3 Th("dominio4.mesh");
// plot (Th, wait = 1);
// CONSTANTES DO PROBLEMA-----
int    n = 12;

real T = 0.5, h = 0.03125, sqrt2 = sqrt(2),
      gama = 0.5, beta = 0.25,
      b1 = 1./(beta*h*h),    b2 = 1./(beta*h),    b3 = 1/(2*beta)-1,
      b4 = gama*h*b1,       b5 = gama/beta-1,    b6 = (gama*h)/(2*beta)-h;
// CONSTANTES DO MATERIAL-----

```

```

real    E = 30.e6,
        sigma = 0.2,
        lambda = E*sigma/((1+sigma)*(1-2*sigma)),
        mu = E/(2*(1+sigma)),
        rho = 25./9.80665;

// MACROS-----
macro  epsilon(u1,u2,u3)  [dx(u1),dy(u2),dz(u3),(dz(u2)+ dy(u3))/sqrt2,
                          (dz(u1)+ dx(u3))/sqrt2,(dy(u1)+dx(u2))/sqrt2] // EOM
macro  div(u1,u2,u3)  ( dx(u1)+dy(u2)+dz(u3) ) // EOM
// ESPAÇO DE APROXIMAÇÃO-----
fespace Vh(Th,P2);
      Vh u1,u2,u3,v1,v2,v3,w1,w2,w3,
        pdu1,pdu2,pdu3,
        du1,du2,du3,ddu1,ddu2,ddu3;

// INICIALIZAÇÃO-----
u1[]=0.;u2[]=0.;u3[]=0.;
du1[]=0.; du2[]=0.; du3[]=0.;
ddu1[]=0.; ddu2[]=0.; ddu3[]=0.;

// INICIO DO CICLO EM i-----
for (int i = 1; i<=n; i++)
{
solve  problema(w1,w2,w3,v1,v2,v3, solver = CG) =
      int3d(Th)( lambda*div(w1,w2,w3)*div(v1,v2,v3)
      + 2.*mu*(epsilon(w1,w2,w3)'*epsilon(v1,v2,v3)))
      + int3d(Th)( b1*rho*(w1*v1 + w2*v2 + w3*v3) )
      - int2d(Th,1500)( (-3.14*sin(2*pi*i*h/T))*v3 )
      - int3d(Th)( b1*rho*(u1*v1 + u2*v2 + u3*v3) )
      + int3d(Th)( b3*rho*(ddu1*v1 + ddu2*v2 + ddu3*v3) )
      + on(2500, w1 = 0., w2 = 0., w3 = 0.);

// CORREÇÃO
for (int n = 0; n < Vh.ndof; n++)
{

```

```

pdu1[] [n] = b4*(w1[] [n]-u1[] [n]) + b5*du1[] [n] + b6*ddu1[] [n];
pdu2[] [n] = b4*(w2[] [n]-u2[] [n]) + b5*du2[] [n] + b6*ddu2[] [n];
pdu3[] [n] = b4*(w3[] [n]-u3[] [n]) + b5*du3[] [n] + b6*ddu3[] [n];

ddu1[] [n] = b1*(w1[] [n]-u1[] [n]) + b3*ddu1[] [n];
ddu2[] [n] = b1*(w2[] [n]-u2[] [n]) + b3*ddu2[] [n];
ddu3[] [n] = b1*(w3[] [n]-u3[] [n]) + b3*ddu3[] [n];
}
du1[] = pdu1[]; du2[] = pdu2[]; du3[] = pdu3[];
u1[] = w1[]; u2[] = w2[]; u3[] = w3[];

// GUARDAR RESULTADOS DINÂMICOS_____
string resul;
resul = "d"+i+".pos";
ofstream file(resul);
file << "View \" \"<< "DESLOCAMENTOS NEWMARK t = "
    << i*h << " \" {" << endl;
// REGISTO DAS COORDENADAS DOS VÉRTICES DOS TETRAEDRICOS_____

    for (int k=0; k<Th.nt; k++)
    {
        file << "VS(";

        for (int j=0; j<3; j++)
            file << Th[k][j].x << ", "
                << Th[k][j].y << ", "
                << Th[k][j].z << ", ";
file << Th[k][3].x << ", "
    << Th[k][3].y << ", "
    << Th[k][3].z;
file << "){" << endl;

```

```

// REGISTO DOS DESLOCAMENTOS EM CADA VÉRTICE_-----

        for (int j=0; j<3; j++)
file << u1[] [Vh(k,j)] << ","
        << u2[] [Vh(k,j)] << ","
        << u3[] [Vh(k,j)] << "," << endl;
file << u1[] [Vh(k,3)] << ","
        << u2[] [Vh(k,3)] << ","
        << u3[] [Vh(k,3)] << "};" << endl;
}

file << "};" << endl;
}

```