



UNIVERSIDADE FEDERAL DE PERNAMBUCO
CENTRO DE TECNOLOGIA E GEOCIÊNCIAS
DEPARTAMENTO DE ENGENHARIA CIVIL
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA CIVIL



LUCÍOLO VICTOR MAGALHÃES E SILVA

MÉTODO DOS ELEMENTOS FINITOS APLICADO À SOLUÇÃO DE PROBLEMAS
POROELÁSTICOS

Recife

2018

LUCÍOLO VICTOR MAGALHÃES E SILVA

MÉTODO DOS ELEMENTOS FINITOS APLICADO À SOLUÇÃO DE PROBLEMAS
POROELÁSTICOS

Dissertação submetida ao corpo docente do Programa de Pós-Graduação em Engenharia Civil da Universidade Federal de Pernambuco como parte integrante dos requisitos necessários à obtenção do grau de Mestre em Engenharia Civil.

Área de concentração: Simulação e Gerenciamento de reservatórios de Petróleo.

Orientador: Prof. Dr. José Leonardo do Nascimento Guimarães

Recife

2018

S886m Silva, Lucíolo Victor Magalhães e.
Método dos elementos finitos aplicado à solução de problemas poroelásticos /
Lucíolo Victor Magalhães e Silva. – Recife, 2018.
227f.: il., fig. tab.

Orientador: Prof. Dr. José Leonardo do Nascimento Guimarães.

Dissertação (Mestrado) – Universidade Federal de Pernambuco. CTG. Programa de Pós-graduação em Engenharia Civil, 2018.

Inclui Referências e Apêndices.

Engenharia Civil. 2. Elementos Finitos. 3. Poroelásticidade. 4. Geomecânica.
I. Guimarães, José Leonardo do Nascimento (Orientador). II. Título.



UNIVERSIDADE FEDERAL DE PERNAMBUCO
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA CIVIL

A comissão examinadora da Defesa de Dissertação de Mestrado

**MÉTODO DOS ELEMENTOS FINITOS APLICADO À SOLUÇÃO DE PROBLEMAS
POROELÁSTICOS**

defendida por

Lucíolo Victor Magalhães e Siva

Considera o candidato APROVADO

Recife, 28 de fevereiro de 2018

Orientador - Prof. Dr. Leonardo José do Nascimento Guimarães – UFPE

Banca Examinadora:

Dr. Jonathan da Cunha Teixeira – UFPE
(examinador externo)

Prof.^a Dr.^a Lícia Mouta da Costa – UFPE
(examinadora externa)

Prof. Dr.^a Igor Fernandes Gomes – UFPE
(examinador externo)

Esse trabalho é dedicado à memória do Professor Doutor Ivaldo Dário da Silva Pontes Filho.

AGRADECIMENTOS

Agradeço especialmente a minha família pelo apoio em todos os momentos, agradeço à meus irmãos Igor e Hugo pelo apoio e aos meus pais Luciene e José Carlos por todo amor e educação que me deram ao longo da vida.

Agradeço aos meus amigos do LMCG e da vida, pela presença, apoio e paciência durante esse período. Alguns dos amigos mais importantes são Pedro Romero, Jean Baptiste, Isabel, Ricardo, Bruno, Tarcísio, Nina, Vitor, Oscar, Nayara, Jeferson, Aura, Ana, Laura, Laís, Marinésio, Thiago, Daniel, Vitor cada um deles contribuíram de certa forma para minha formação. Agradeço ao meu companheiro Jonathan Teixeira pelos ensinamentos e discussões.

Agradeço aos funcionários e funcionárias do LMCG e do PPGEC, em especial a Rose e Priscila, assim como Andréa, que além da amizade sempre estiveram à disposição de todos.

Agradeço aos professores do LMCG pelos ensinamentos e pela forma que recebem seus alunos no grupo. Agradeço ao professor Leonardo Guimarães, meu atual orientador, à Professora Lícia e ao professor Igor. Ao professor Ivaldo, que no presente momento não se encontra mais entre nós, eu dedico esse trabalho que tenho como sendo uma homenagem à sua lembrança, a partir dele foi que me dediquei ao mestrado na área que escolhi e ao grupo.

Agradeço a todos que contribuíram para minha formação acadêmica desde a graduação, os amigos do LABCOM, os professores Paulo Lyra, Darlan Karlo, os alunos de pós graduação Eduardo e Márcio, por me introduzirem na área numérica durante a iniciação científica. Alberto, Fernando, Gustavo, Felipe, Marcelo, pelo acolhimento no começo da vida na pesquisa científica. Ao professor Paulo Marcelo, que contribuiu de forma significativa para minha formação.

RESUMO

Os problemas poromecânicos são caracterizados por situações onde há o interesse de determinar o comportamento mecânico de meios formados por uma matriz porosa saturada ou não por fluidos ou gases. Esses são encontrados em diversas áreas do conhecimento e são objeto de interesse de institutos de pesquisa e da indústria e, assim como outros problemas apresentados na ciência, podem ser definidos em termos matemáticos por equações diferenciais. Tais equações, devido ao advento da evolução da computação e pela natureza dos problemas poromecânicos: geometria, condições de contorno, características físicas dos materiais; são, atualmente, resolvidos em grande parte das aplicações utilizando formulações numéricas implementadas em computadores digitais. Sendo assim, esse trabalho consiste no desenvolvimento de um simulador numérico, desenvolvido em ambiente MATLAB®, para a solução de uma das vertentes da poromecânica, a poroelasticidade linear. O pré-processamento, no que diz respeito à geração de malha e modelagem da geometria dos problemas, é realizado no Gmsh. Todo o processamento e pós-processamento foi implementado no código desenvolvido. As condições a que estão sujeitas a formulação utilizada são as seguintes: 1. Meio sólido sujeito ao regime elástico-linear; 2. Meio sujeito ao regime do estado plano de deformações; 3. Todas as deformações ocorridas são consideradas como pequenas deformações; 4. O escoamento é laminar e pode ser descrito pela Lei de Darcy; 5. O escoamento e as deformações ocorrem de forma isotérmica. As formulações numéricas utilizadas para a discretização foram uma formulação mista do Método dos Elementos Finitos em termos da discretização espacial e uma formulação do Método das Diferenças Finitas para a discretização temporal. As variáveis mecânicas são interpoladas com uma função quadrática, definida em um elemento com seis nós e as funções interpoladoras das variáveis hidráulicas são construídas com os valores nos vértices de um triângulo, sendo assim uma função linear. Da forma que foi implementado o problema a discretização temporal é função de um parâmetro que define e aproximação como uma ponderação entre o passo de tempo em que as variáveis serão avaliadas. Para a validação e comparação do código foram resolvidos cinco problemas apresentados na literatura. Os problemas validaram o código, obtendo resultados satisfatórios. Também foi analisada a influência das condições de contorno em problemas com solução analítica considerando o domínio infinito e o adensamento de um meio anisotrópico e heterogêneo.

Palavras-chave: Elementos Finitos. Poroelasticidade. Geomecânica.

ABSTRACT

Poromechanical problems are characterized by situations where it is of interest to determine the mechanical behavior of media formed by a porous matrix saturated or not by fluids or gases. These are found in several areas of knowledge and are of interest to research institutes and industry and, like other problems presented in science, can be defined in mathematical terms by differential equations. These equations, due to the advent of computational evolution and the nature of the poromechanical problems: geometry, contour conditions, physical characteristics of materials; are currently solved in most applications using numerical formulations implemented in digital computers. Thus, this work consists in the development of a numerical simulator, developed in MATLAB® environment, for the solution of one of the slopes of the poromechanics, the linear poroelasticity. Preprocessing, with respect to mesh generation and modeling of problem geometry, is reported in Gmsh. All processing and post-processing was implemented in the developed code. The conditions to which the formulation is subjected are the following: 1. Solid medium subject to the linear-elastic regime; 2. Medium subject to the flat state of deformations; 3. All deformations occurring are considered as small deformations; 4. The flow is laminar and can be described by Darcy's Law; 5. The flow and deformation occur isothermal. The numerical formulations used for the discretization were a mixed formulation of the Finite Element Method in terms of spatial discretization and a formulation of the Finite Differences Method for time discretization. The mechanical variables are interpolated with a quadratic function, defined in a six-node element, and the interpolating functions of the hydraulic variables are constructed with the values at the vertices of a triangle, thus being a linear function. In the way that the problem was implemented the temporal discretization is a function of a parameter that defines an approximation as a point of reference between the time step in which the variables will be evaluated. For the validation and comparison of the code, five problems were solved in the literature. The problems validated the code, obtaining satisfactory results. We also analyzed the influence of boundary conditions on problems with analytical solution considering the infinite domain and the densification of an anisotropic and heterogeneous medium.

Keywords: Finite elements. Poroelasticity. Geomechanics

LISTA DE ILUSTRAÇÕES

Figura 1.1. Meio poroso, esquema de rocha reservatório.....	14
Figura 2.1. Professor Maurice Anthony Biot.	17
Figura 2.2. General Theory of Three-Dimensional Consolidation, artigo de Biot.....	18
Figura 4.1. Corpo submetido a carregamentos externos.	26
Figura 4.2. Seção no corpo.	27
Figura 4.3. Detalhe do ponto P.	28
Figura 4.4. Tensões em um elemento infinitesimal.....	29
Figura 4.5. Tipos de análise: (a) Geometria do problema; (b) Estado Plano de Tensões e Estado Plano de Deformações; (c) Análise 3-D.	33
Figura 4.6. Velocidade em um elemento infinitesimal.....	36
Figura 5.1. Deslocamentos nodais utilizados na função interpoladora dos deslocamentos.	43
Figura 5.2. Pressões nodais utilizados na função interpoladora das pressões.	45
Figura 7.1. Procedimento de simulação - criação da geometria do Gmsh.	56
Figura 7.2. Procedimento de simulação - geração da malha computacional no Gmsh.	57
Figura 7.3. Procedimento de simulação - arquivo .geo e .msh obtidos no Gmsh.	57
Figura 7.4. Procedimento de simulação - definição, no MATLAB, dos arquivos que serão lidos.	58
Figura 7.5. Procedimento de simulação - definição dos parâmetros para a simulação na rotina do MATLAB.	58
Figura 7.6. Procedimento de simulação - processamento do MATLAB.....	59
Figura 7.7. Procedimento de simulação - criação de diretórios com os resultados.....	59
Figura 7.8. Procedimento de simulação - resultados salvos.....	60
Figura 8.1. Carreamento em função do tempo.	61
Figura 8.2. Problema de Terzaghi - Geometria e condições de contorno.	63
Figura 8.3. Problema de Terzaghi: (a) Malha 1; (b) Malha 2.....	64
Figura 8.4. Problema de Terzaghi - Solução analítica e numérica das pressões ao longo do eixo $x=0$ para a Malha 1.	66
Figura 8.5. Problema de Terzaghi - Solução analítica e numérica das pressões ao longo do eixo $x=0$ para a Malha 2.	66
Figura 8.6. Problema de Terzaghi - Campo de pressões, (a) Malha1, (b) Malha2.....	67
Figura 8.7. Problema de Terzaghi - Campo de tensões normais na direção x , (a) Malha1, (b) Malha2.....	68
Figura 8.8. Problema de Terzaghi - Campo de tensões normais na direção y , (a) Malha1, (b) Malha2.....	69
Figura 8.9. Problema de Terzaghi Modificado – Geometria e Condições de Contorno.	70
Figura 8.10. Problema de Terzaghi Modificado – Malhas computacionais utilizadas: (a) Malha1; (b) Malha2.	72
Figura 8.11. Problema de Terzaghi Modificado – soluções analíticas e numéricas para a Malha1.	73

Figura 8.12. Problema de Terzaghi Modificado – soluções analíticas e numéricas para a Malha2.....	74
Figura 8.13. Problema de Terzaghi modificado - Campo de pressões, (a) Malha1, (b) Malha2.	75
Figura 8.14. Problema de Terzaghi modificado - Campo de tensões normais na direção x, (a) Malha1, (b) Malha2.	76
Figura 8.15. Problema de Terzaghi modificado - Campo de tensões normais na direção y, (a) Malha1, (b) Malha2.	77
Figura 8.16. Problema de Mandel - Geometria e condições de contorno.	78
Figura 8.17. Problema de Mandel - Simetria utilizada para a modelagem.	80
Figura 8.18. Problema de Mandel – Malhas computacionais utilizadas: (a) Malha1; (b) Malha2.....	81
Figura 8.19. Problema de Mandel – pressões ao longo de $y=0$ para a Malha1.	83
Figura 8.20. Problema de Mandel – pressões ao longo de $y=0$ para a Malha2.	84
Figura 8.21. Problema de Mandel – malha inicial (azul) e deformada (vermelho) para a Malha1 ($T_v = 1,00$).	84
Figura 8.22. Problema de Mandel – malha inicial (azul) e deformada (vermelho) para a Malha2 ($T_v = 1,00$)	85
Figura 8.23. Problema de Mandel - Campo de pressões, (a) Malha1, (b) Malha2.....	85
Figura 8.24. Problema de Mandel - Campo de tensões normais na direção x, (a) Malha1, (b) Malha2.	86
Figura 8.25. Problema de Mandel - Campo de tensões normais na direção y, (a) Malha1, (b) Malha2.....	87
Figura 8.26. Problema de Schiffman - Geometria e condições de contorno.	88
Figura 8.27. Problema de Schiffman – Malha computacional.	91
Figura 8.28. Problema de Schiffman – solução analítica e soluções numéricas das pressões de fluido ao longo de $x=0$ ($T_v = 0,10$).	92
Figura 8.29. Problema de Schiffman – malha inicial (azul) e deformada (vermelho) para a situação1 ($T_v = 0,10$).....	94
Figura 8.30. Problema de Schiffman – malha inicial (azul) e deformada (vermelho) para a situação2 ($T_v = 0,10$).....	94
Figura 8.31. Problema de Schiffman – detalhe da malha inicial (azul) e deformada (vermelho) para a situação1 ($T_v = 0,10$).	95
Figura 8.32. Problema de Schiffman – detalhes da malha inicial (azul) e da malha deformada (vermelho) para a situação2 ($T_v = 0,10$)	95
Figura 8.33. Problema de Schiffman - Campo de pressões, (a) Situação1, (b) Situação2.	96
Figura 8.34. Problema de Schiffman - Campo de tensões normais na direção x, (a) Situação1, (b) Situação2.....	97
Figura 8.35. Problema de Schiffman - Campo de tensões cisalhantes, (a) Malha1, (b) Malha2.	98
Figura 8.36. Problema de Schiffman - Campo de tensões normais na direção y, (a) Situação1, (b) Situação2.....	99

Figura 8.37. Problema Heterogêneo e Anisotrópico – Geometria, contornos e domínios.....	103
Figura 8.38. Problema Heterogêneo e Anisotrópico – Malha computacional.....	103
Figura 8.39. Problema Heterogêneo e Anisotrópico – solução analítica e soluções numéricas das pressões de fluido ao longo de $x=0$ para $r=1$ ($T_v = 0,10$) .	104
Figura 8.40. Problema Heterogêneo e Anisotrópico – solução analítica e soluções numéricas das pressões de fluido ao longo de $x=0$ para $r=5$ ($T_v = 0,10$)	105
Figura 8.41. Problema Heterogêneo e Anisotrópico – solução analítica e soluções numéricas das pressões de fluido ao longo de $x=0$ para $r=50$ ($T_v = 0,10$)	106
Figura 8.42. Problema Heterogêneo e Anisotrópico – detalhes da malha inicial (azul) e da malha deformada (vermelho) para $r=1$ ($T_v = 0,10$)	106
Figura 8.43. Problema Heterogêneo e Anisotrópico – detalhes da malha inicial (azul) e da malha deformada (vermelho) para $r=5$ ($T_v = 0,10$)	107
Figura 8.44. Problema Heterogêneo e Anisotrópico – detalhes da malha inicial (azul) e da malha deformada (vermelho) para $r=5$ ($T_v = 0,10$)	107
Figura 8.45. Adensamento bidimensional heterogêneo e anisotrópico - Campo de pressões, (a) $r=1$, (b) $r=5$, (c), $r=50$.	108
Figura 8.46. Adensamento bidimensional heterogêneo e anisotrópico - Campo de tensões normais na direção x, (a) $r=1$, (b) $r=5$, (c), $r=50$.	109
Figura 8.47. Adensamento bidimensional heterogêneo e anisotrópico - Campo de tensões cisalhantes, (a) $r=1$, (b) $r=5$, (c), $r=50$.	110
Figura 8.48. Adensamento bidimensional heterogêneo e anisotrópico - Campo de tensões normais na direção y, (a) $r=1$, (b) $r=5$, (c), $r=50$.	111

SUMÁRIO

1	INTRODUÇÃO	13
2	REVISÃO DA LITERATURA	16
3	OBJETIVO DO TRABALHO	25
4	TEORIA DE BIOT	26
4.1	EQUAÇÃO DE EQUILÍBRIO MECÂNICO.....	26
4.2	ESTADO PLANO DE DEFORMAÇÕES.....	31
4.3	EQUAÇÃO DE BALANÇO DE MASSA DE FLUIDO.....	35
4.4	LEI DE DARCY.....	37
4.5	PRINCÍPIO DAS TENSÕES EFETIVAS.....	38
4.6	SISTEMA DE EQUAÇÕES DA TEORIA GERAL DO ADENSAMENTO DE BIOT.....	39
5	FORMULAÇÃO DO MÉTODO DOS ELEMENTOS FINITOS APLICADA À TEORIA DE BIOT	41
6	FORMULAÇÃO DO MÉTODO DAS DIFERENÇAS FINITAS PARA A DISCRETIZAÇÃO TEMPORAL DA TEORIA DE BIOT	52
7	ASPECTOS COMPUTACIONAIS	53
8	ANÁLISES REALIZADAS	61
8.1	ADENSAMENTO UNIDIMENSIONAL DE MEIO HOMOGÊNEO E ISOTRÓPICO – PROBLEMA DE TERZAGHI.....	62
8.2	ADENSAMENTO UNIDIMENSIONAL DE MEIO HETEROGÊNEO E ISOTRÓPICO – PROBLEMA DE TERZAGHI MODIFICADO.....	69
8.3	ADENSAMENTO BIDIMENSIONAL DE MEIO HOMOGÊNEO E ISOTRÓPICO – PROBLEMA DE MANDEL.....	77
8.4	ADENSAMENTO BIDIMENSIONAL DE MEIO HOMOGÊNEO E ISOTRÓPICO – PROBLEMA DE SCHIFFMAN.....	87
8.5	ADENSAMENTO BIDIMENSIONAL DE MEIO HETEROGÊNEO E ANISOTRÓPICO.....	100
9	CONCLUSÕES	113

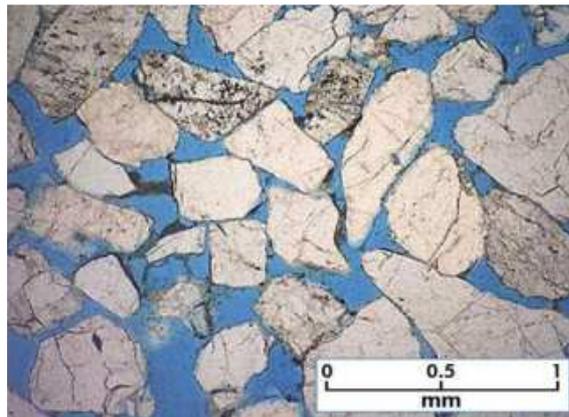
10 SUJESTÕES PARA TRABALHOS FUTUROS.....	114
REFERÊNCIAS.....	115
APÊNDICE A.....	121
APÊNDICE B.....	209
SOLUÇÃO ANALÍTICA PARA O PROBLEMA DE TERZAGHI.....	209
SOLUÇÃO ANALÍTICA PARA O PROBLEMA DE TERZAGHI MODIFICADO.	212
SOLUÇÃO ANALÍTICA PARA O PROBLEMA DE MANDEL.....	220

1 INTRODUÇÃO

Os problemas poromecânicos são encontrados com frequência em aplicações da Engenharia e ciências. Usualmente constituem casos com complexidade considerável, envolvendo conhecimentos multidisciplinares para serem entendidos e resolvidos. Esses ocorrem quando há, em determinado fenômeno, a interação entre um meio sólido poroso onde seus poros são preenchidos, total ou parcialmente, por um ou mais fluidos, onde as deformações ocorridas no esqueleto sólido podem gerar modificação na distribuição de pressões no fluido e, da mesma forma, alterações nas pressões no fluido podem gerar modificação no estado de tensões no meio sólido. A Figura 1.1 apresenta uma imagem de um meio poroso numa escala milimétrica, onde é possível a verificação da ocorrência dos poros em um meio e o preenchimento dos seus vazios por um fluido. É válido salientar que nas aplicações usuais as dimensões dos poros ocorrem em escalas microscópicas e a consideração da porosidade, nas formulações físicas que regem esses problemas, são associadas ao meio contínuo, onde as variáveis do problema são avaliadas de forma contínua ao longo do domínio. Sendo assim, a poromecânica é, em termos gerais, o estudo do comportamento mecânico do meio poroso, que pode ser submetido a diversos processos sujeitos a diversas condições (COUSSY, 2004). Um exemplo desse processo pode ser a consideração de carregamentos gerados por condições térmicas impostas ao meio poroso, outros exemplos podem ser os processos químicos e até mesmo biológicos. Esses carregamentos podem gerar, por exemplo, acelerações inerciais no meio, devido à periodicidade do carregamento ou a ocorrência de um impacto, sendo necessária a avaliação da resposta dinâmica do meio a tal carregamento.

Outros exemplos são as considerações das relações constitutivas que regem o comportamento tanto do fluido quando do esqueleto sólido, que determinam classes de problemas específicos, como a consideração de meios plásticos, elásticos ou viscoplásticos, fluido newtoniano ou não-newtoniano (CARMO; MARQUES; THOMPSON, 2010). Outra classe de problemas que podem se manifestar em um meio poroso é a ocorrência de um ou mais fluidos ou até mesmo a presença de gás. Portanto, várias são as vertentes da poromecânica e esta é objeto de interesse de diversos centros de pesquisas, instituições de desenvolvimento e indústria.

Figura 1.1. Meio poroso, esquema de rocha reservatório.



Fonte: http://www.pucrs.br/cepac/download/Eng_Reservatorio_CCS.pdf

A poromecânica rege situações em áreas como a Medicina e Odontologia (LÓPEZ, 2012), onde as aplicações são as mais variadas possíveis, como a avaliação do comportamento de uma massa cancerígena, a avaliação das tensões provocadas pela mecânica do movimento em ossos e articulações e até mesmo o comportamento mecânico dos dentes na mastigação. A Geomecânica, ciência interessada no comportamento mecânico das rochas e solos, compreende uma área de grande aplicação da poromecânica. Exemplos de problemas geomecânicos podem ser encontrados na Mecânica dos Solos, na Engenharia Civil; Mecânica das Rochas, na Engenharia de Minas; Hidrogeologia, na Geologia; Engenharia de Reservatórios de Petróleo, na Engenharia de Petróleo e outras áreas que compreendem fenômenos que ocorrerem em meios porosos.

Assim como diversas outras áreas do conhecimento, a poromecânica é regida por um conjunto de equações diferenciais. A solução dessas equações diferenciais é objeto de diversos trabalhos de pesquisadores, apresentados ao longo do texto, que se dedicaram tanto às soluções analíticas, em casos menos complexos e representativos, quanto ao desenvolvimento de ferramentas de cálculos automáticos utilizados para problemas casos complexos (TERZAGHI, 1943). Em geral, as soluções analíticas podem ser utilizadas em situações particulares que se adequam à suas geometrias, condições de contorno e parâmetros físicos, ou para que se estabeleça uma aproximação quando se quer, por exemplo, uma estimativa de determinado parâmetro. Isso pode ser visto, por exemplo, quando há a aplicação de carga de uma sapata em um solo totalmente homogêneo e saturado, condição onde a solução analítica se adequa totalmente ao caso real (SCHIFFMAN; CHEN; JORDAN, 1969). Porém, quando se tem um

cenário associado à geomecânica de formações geológicas como aquíferos e reservatórios de petróleo que apresentam geralmente uma geometria irregular, os parâmetros físicos são, em geral, anisotrópicos e as formações heterogêneas. Essas características tornam o desenvolvimento das soluções analíticas de tais problemas demasiadamente tediosas ou até não solucionáveis.

Com o desenvolvimento de técnicas numéricas, em paralelo ao desenvolvimento da computação, em especial a partir da segunda metade do século XX, a solução das equações diferenciais utilizando formulações numéricas se tornou bastante atrativa. Sendo assim, atualmente, a simulação numérica se tornou a forma mais natural de solução das equações que regem determinadas áreas do conhecimento. Essas técnicas, juntamente com a disponibilidade de várias ferramentas disponíveis para o tratamento de dados, visualização de resultados e o desenvolvimento de linguagens computacionais específicas para problemas técnicos-científicos, proporcionam ao cientista e engenheiro da atualidade a oportunidade de desenvolvimento de seus próprios simuladores numéricos.

Sendo assim, esse trabalho consiste no desenvolvimento de um simulador poromecânico, para as condições de um meio sujeito ao regime de escoamento linear, pequenas deformações, ocorrendo sob o Estado Plano de Deformações com escoamento laminar e todo o processo ocorrendo de forma isotérmica. Para a solução das equações foram utilizados o Método dos Elementos Finitos (MEF) e Método das Diferenças Finitas (MDF), para a discretização espacial e temporal, respectivamente. A formulação do MEF utilizada é uma formulação mista com elementos triangulares quadráticos para as variáveis mecânicas e lineares para as variáveis hidráulicas.

2 REVISÃO DA LITERATURA

A Teoria da Poroelasticidade, que pode surgir com outras denominações, como a Teoria do Adensamento ou Consolidação, em trabalhos desenvolvidos nas áreas da Engenharia, por exemplo, tem seu desenvolvimento contínuo e progressivo em termos de teorias e formulações que, cada vez mais, tendem a levar em considerações as condições reais do meio em estudo. O primeiro grande trabalho que se tem reconhecimento por parte da comunidade científica relacionado à poroelasticidade foi desenvolvido por Terzagui no ano de 1923 (TERZAGHI, 1943). Esse trabalho apresenta a solução de um problema poroelástico unidimensional, homogêneo e isotrópico onde se admite a linearidade entre as tensões efetivas e deformações do meio, que tem comportamento elástico. Também é admitido que as tensões totais permanecem constantes ao longo do desenvolvimento do adensamento. Porém, apesar das simplificações adotadas por Terzagui e a consideração de condições que, em muitas das situações não se aplicam à realidade de determinada situação, essa teoria é, talvez, a mais difundida nas escolas de graduação em Engenharia e certamente utilizada de forma considerável nos escritórios de projetos geotécnicos e de fundações. Aplicações como recalques de estruturas ou de aterros assentados sobre argilas são exemplos práticos onde pode se aplicar essa teoria para ser usada no âmbito da prática da Engenharia.

Na Poroelasticidade, assim como em outras áreas do conhecimento, a busca por Teorias mais gerais é uma tendência natural e, sendo assim, o próprio Terzagui se dedicou ao desenvolvimento de uma teoria mais generalista, considerando o adensamento se desenvolvendo ao longo das três direções, chegando a publicar esse trabalho que pode ser visto em Terzaghi e Frohlich (1936) e Terzaghi (1943). Porém, essa teoria, desenvolvida por Terzaghi, possui uma consideração que se alastra desde a Teoria do adensamento unidimensional, nela é levada em consideração a constância nas tensões totais ao longo do adensamento, fato que não condiz com a realidade. Uma das verificações que pode ser feita é o fato de que no adensamento pode haver a ocorrência de valores das pressões de fluido maiores que os iniciais. Esse aumento nas pressões de fluido pode ser dado tanto devido à redistribuição dessas pressões ao longo do domínio como pela variação nas tensões totais em determinadas regiões. Rendulic (1936) também apresenta uma formulação para processos poroelásticos mais abrangente que a desenvolvida por Terzaghi, apesar de não ser considerada ainda uma formulação acoplada.

O aumento nas pressões de fluidos discutido anteriormente, quando ocorrido devido à variação nas tensões totais, foram formulados teoricamente nos trabalhos de Mandel (1953) e de Cryer (1963), e recebe o nome de Efeito Mandel-Cryer em suas homenagens. Outro aspecto importante apresentado no trabalho de Terzaghi (1923) é o Princípio das Tensões Efetivas de Terzagui, que propõe uma relação entre as tensões mecânicas do meio e a pressão de fluido. Tal princípio foi fundamental para o desenvolvimento das formulações poroelásticas.

Em 1941, Maurice A. Biot, apresentado na Figura 2.1, desenvolveu, em seu trabalho a Teoria do Adensamento Tridimensional (BIOT, 1941), ou como expressão no próprio título do seu trabalho, que pode ser visto na Figura 2.2, a Teoria Geral da Consolidação Tridimensional, também conhecida como Teoria de Biot. Essa teoria, apesar da denominação de “Teoria Geral”, possui condições específicas para ser aplicadas. Então, quando se trata do termo “Geral”, este é associado ao desenvolvimento do adensamento ao longo das três direções coordenadas e não geral no sentido estrito da palavra. Porém, pelo fato de seu trabalho ser o primeiro a descrever corretamente o desenvolvimento tridimensional do adensamento, tal denominação é amplamente aceita pela comunidade científica.

Figura 2.1. Professor Maurice Anthony Biot.



Fonte: <https://shellbuckling.com/cv/biot.pdf>

Na Teoria de Biot (BIOT, 1941), há considerações acerca dos seguintes aspectos: do material, que é homogêneo e isotrópico; há a reversibilidade na relação tensão-deformação, que é considerada linear; não há ocorrência de grandes deformações; as partículas que com-

põem o esqueleto sólido são indeformáveis; o líquido é incompressível e seu fluxo é regido pela Lei de Darcy e obedece ao Princípio das Tensões efetivas. A Lei de Darcy consiste em uma teoria, fruto do trabalho de Henry Philibert Gaspar Darcy no ano de 1856, que rege o comportamento de um fluido escoando, de forma laminar em um meio poroso. Tal lei estabelece a relação entre o gradiente de pressões de fluido e a permeabilidade do meio com a velocidade do escoamento.

Figura 2.2. General Theory of Three-Dimensional Consolidation, artigo de Biot.

General Theory of Three-Dimensional Consolidation*

MAURICE A. BIOT
Columbia University, New York, New York
(Received October 25, 1940)

The settlement of soils under load is caused by a phenomenon called consolidation, whose mechanism is known to be in many cases identical with the process of squeezing water out of an elastic porous medium. The mathematical physical consequences of this viewpoint are established in the present paper. The number of physical constants necessary to determine the properties of the soil is derived along with the general equations for the prediction of settlements and stresses in three-dimensional problems. Simple applications are treated as examples. The operational calculus is shown to be a powerful method of solution of consolidation problems.

INTRODUCTION

IT is well known to engineering practice that a soil under load does not assume an instantaneous deflection under that load, but settles gradually at a variable rate. Such settlement is very apparent in clays and sands saturated with water. The settlement is caused by a gradual adaptation of the soil to the load variation. This process is known as *soil consolidation*. A simple

possible: the extension to the three-dimensional case, and the establishment of equations valid for any arbitrary load variable with time. The theory was first presented by the author in rather abstract form in a previous publication.² The present paper gives a more rigorous and complete treatment of the theory which leads to results more general than those obtained in the previous paper.

Fonte: Elaborada pelo autor.

O próprio Biot, ao longo de uma sequência de publicações e trabalhos desenvolvidos tornou a sua teoria inicial mais generalista. Em seu trabalho (BIOT, 1941), o mesmo introduz a possibilidade da ocorrência de um fluido compressível em sua teoria inicial, no seu trabalho (BIOT, 1955), Biot introduz a anisotropia em termos de permeabilidade. No trabalho (BIOT, 1956) Biot introduz à teoria anisotrópica a possibilidade de uma relação constitutiva viscoelástica em termos mecânicos. Biot e Willis (1957) apresentam os coeficientes elásticos utilizados na sua teoria e em seu trabalho (BIOT, 1973) introduz os conceitos da reologia à sua Teoria. A partir do desenvolvimento da Teoria Tridimensional do Adensamento, vários pesquisadores concentraram seus trabalhos no desenvolvimento de maneiras para solucionar as equações que a regem.

Biot, além de produzir trabalhos relacionados à poroelasticidade, também foi responsável por contribuições em áreas como aerodinâmica, termodinâmica irreversível e transferência de calor, viscoelasticidade e também a termoelasticidade. A poroelasticidade e termoelasticidade são, assim como vários outros fenômenos naturais, regidos por um conjunto de equações diferenciais análogos. Em sua trajetória, Biot trabalhou em diversas agências governamentais e universidades como a Universidade de Columbia, Harvard, Lovaina e Brown. Biot recebeu o grau de bacharel em filosofia, se formou em engenharia e minas e elétrica, e doutorado em ciências, na Universidade de Lovaina. Em 1932 terminou seu Ph.D em ciências aeronáuticas no Instituto de Tecnologia da Califórnia.

Além do trabalho desenvolvido por Terzaghi (1923) para o adensamento unidimensional, Mandel (1953) apresenta uma solução analítica para uma situação de adensamento bidimensional que consiste em um meio poroso retangular, sujeito a um carregamento em seu topo e base, que se mantém impermeáveis e com suas laterais com capacidade drenante e com carregamento nulo, esse problema é frequentemente denominado na literatura como “Problema de Mandel”. Cryer (1963) desenvolve uma solução analítica para o adensamento de um meio poroso esférico submetido à um carregamento constante em seus contornos que permanecem, ao longo do adensamento, com pressões nulas. Essas soluções analíticas clássicas são bastante utilizadas para a validação de alternativas desenvolvidas para a solução de problemas poroelásticos.

Algumas soluções analíticas para as equações de Biot considerando variações no adensamento de uma camada de solo submetida a um carregamento específico são apresentadas nos trabalhos: (GIBSON; MCNAMEE, 1960.a), (GIBSON; MCNAMEE, 1960.b) e (GIBSON; MCNAMEE, 1957) onde em cada um desses trabalhos um parâmetro do problema é modificado, seja ele físico, geométrico ou associado às condições de contorno. Outros estudos com soluções analíticas para esse problema são apresentados em (BOOKER, 1973), (SINGH; RANI; KUMAR, 2013) e (AMIRI; HAJALI; ESMAEILY, 2014). Outra solução analítica desenvolvida em um problema axissimétrico de adensamento é apresentado por Wang e Fang (2003).

Os avanços tecnológicos, em particular na segunda metade do século XX, no ramo da computação científica, permitiram aos engenheiros e cientistas o desenvolvimento e aplicação de procedimentos numéricos de forma automática e eficiente, tendo em vista que os cálculos tediosos e repetitivos realizados pelos antigos analistas numéricos envolviam esforços manu-

ais consideráveis, principalmente associados ao cálculo matricial. É válido salientar que introdução dos computadores modernos na ciência não deu início ao investimento dos cientistas no desenvolvimento de ferramentas numéricas, alguns relatos apontam que há mais de 2.000 anos já se utilizavam procedimentos de interpolação lineares. Mais recentemente, cientistas como Newton, Gauss, Euler e Lagrange já desenvolviam técnicas numéricas que eram utilizadas sem o advento dos computadores modernos (LOTTI et al., 2006). Porém, certamente o acesso à computação na formatação atual atraiu uma quantidade significativa de desenvolvedores e usuários de ferramentas numéricas.

No que se trata de métodos numéricos para a solução de equações diferenciais temos como um dos principais o Método das Diferenças Finitas (MDF). Esse método é bastante utilizado e talvez um dos mais antigos em uso. Para se obter a forma discreta da equação diferencial pelo MDF é necessária a substituição do operador diferencial por um operador discreto obtido através de manipulações numéricas na série de Taylor. Apesar de haverem, na literatura corrente, formulações mais generalistas desse método que resolvam essas questões, em sua essência esse método é utilizado em situações onde a geometria e a malha do problema sejam regulares (PENA, 2003). Além disso, a imposição da condição de contorno, em particular de Neumann, envolve certos inconvenientes para se implementar de forma geral. É fato que as características apresentadas pelo MDF podem ser observadas principalmente quando há a discretização em termos das coordenadas espaciais, porém esse método é amplamente utilizado quando se trata da discretização da equação em termos da variável temporal da equação avaliada.

O desenvolvimento da matemática, computação e a necessidade de desenvolvimento de ferramentas mais potentes para a solução de equações diferenciais deram origem ao surgimento de novos métodos numéricos. No final do século XVIII Gauss propôs a utilização de funções de aproximação para a solução de problemas matemáticos (LOTTI et al., 2006), onde a aproximação era estabelecida em todo o domínio e necessitava, em muitos casos, o uso do bom senso do analista. No século XIX trabalhos de cientistas como Bernoulli, Lagrange e Stevin formataram uma série de trabalhos que deram origem à utilização de métodos energéticos na solução de problemas de engenharia. Courant (1943) introduziu o conceito de Gauss em elementos do domínio, que é discretizado, e aplica a condição de continuidade entre os trechos. Esses desenvolvimentos, juntamente com a combinação dos conceitos do cálculo variacional e de trabalhos como o de Ritz (FARENZENA; HAUSER, 2009) serviram como base para que na década de 1960 em seu trabalho, apresentasse o Método dos Elementos Fi-

nitos (MEF) como é conhecido atualmente (CLOUGH, 1960). Esse trabalho foi desenvolvido com a influência e necessidade da indústria aeronáutica e aeroespacial que se desenvolvia no período pós-guerra.

O MEF, em comparação com o MDF, possui um embasamento matemático mais robusto, que precisa ser entendido pelos usuários do método. Segundo Campos e Silva (2006) o MEF possui três características que o definem sendo a primeira delas a consideração de um campo contínuo, subdividido em células, que são denominados elementos e que constituem uma malha. A segunda característica importante é que a solução, a priori, possui uma forma que será regida pelas funções interpoladoras, ou também chamadas de funções de forma, do elemento. A terceira característica do MEF é a forma integral da solução, de forma que a formulação do problema é obtida a partir da aplicação do Método dos Resíduos Ponderados. Em geral o MEF não possui imposições restritivas em função da geometria do modelo, porém as formas como os elementos da malha computacional se apresentam pode afetar significativamente a solução. Uma área de grandes aplicações do MEF é a mecânica dos materiais, seja na Engenharia Estrutural, Engenharia Geotécnica ou Geomecânica, por exemplo, devido às suas considerações matemáticas e físicas, que o tornam um método natural pra esse tipo de aplicação.

Outros dois métodos bastante utilizados para a solução de equações diferenciais são o Método dos Volumes Finitos (MVF) e o Método dos Elementos de Contorno (MEC). O MVF tem normalmente aplicação onde precisam ser respeitadas rigorosamente as leis de conservação, como transferência de calor e de massa. O MEC possui uma vantagem, considerando a maioria das suas formulações, em relação aos outros métodos em domínios infinitos ou semi-infinitos.

Na literatura são encontradas aplicações dos os métodos citados na poroelasticidade. Brown (2002), em sua tese de doutoramento, apresenta uma formulação do MEC aplicado à poroelasticidade. Ribeiro (2016) emprega uma formulação do MVF denominada de Método dos Volumes Finitos Baseada em Elementos, assim como Pizzol (2014). Os trabalhos: (FERREIRA, 2010), (SLOAN; ABBO, 1999.a), (SLOAN; ABBO, 1999.b) , (JIA-DONG et al., 1982), (ZHANG; ZARROUK; ARCHER, 2015), (PHILLIPS, 2005) e (SIQUEIRA, 1995) apresentam formulações do MEF para a solução de problemas poroelásticos a partir de desenvolvimentos de códigos computacionais. Esses trabalhos apresentam o MEF para a discre-

tização das equações em termos das variáveis espaciais, a discretização em termos da variável temporal é feita, em todos os casos, utilizando o MDF.

Alguns trabalhos, e até mesmo simuladores comerciais são desenvolvidos utilizando métodos específicos para a parte mecânica e hidráulica. Em geral é utilizado o MEF para a solução da parte mecânica e o MVF ou o MDF para a solução do problema hidráulico. Outro aspecto onde é apresentado investimento significativo em pesquisas é o tipo de acoplamento que será considerado para determinada aplicação. Em geral os métodos de acoplamento são divididos em quatro classes: pseudo-acoplamento, acoplamento explícito, acoplamento iterativo e totalmente acoplado. Cada um deles apresenta vantagens em termos de considerações sobre simplificações adotadas e avaliação dos custos computacionais envolvidos.

As formulações mistas do MEF surgem como uma alternativa para a solução de problemas onde ocorre o “travamento” volumétrico, denominado locking, que numericamente ocorre quando os resultados obtidos pela formulação aproximada apresentam valores de deslocamentos menores que os reais. Esses resultados são inerentes às formulações tradicionais do MEF, gerando resultados inconsistentes em situações de corpos incompressíveis ou quase incompressíveis (SANTOS, 2011). Diversas formulações mistas são apresentadas na literatura com aproximações combinadas de diversos campos do problema como as variáveis primais e secundárias, como exemplo temos as formulações deslocamento/velocidade – tensão. Nesse trabalho a formulação mista apresentada será em termos de velocidade – pressão.

Em relação aos tipos de acoplamento, o pseudo-acoplamento não considera a resolução do modelo geomecânico durante a simulação. Os parâmetros hidráulicos e mecânicos são correlacionados a partir de leis específicas. Tais leis podem ser obtidas a partir de considerações físicas ou até mesmo de forma empírica. No acoplamento explícito a mecânica e hidráulica são resolvidas separadamente, para cada passo de tempo, onde as informações hidráulicas são passadas para o simulador mecânico que os considera. A partir da deformação volumétrica ocorre a atualização da porosidade e conseqüentemente da permeabilidade que alimenta o simulador de fluxo. No acoplamento iterativo as informações hidráulicas e mecânicas se relacionam no mesmo passo de tempo até a observância de um determinado critério, como critérios de convergência de métodos não-lineares, por exemplo. No modelo totalmente acoplado a mecânica e hidráulica são resolvidas simultaneamente a partir de um sistema de equações acoplado, considerando as duas físicas, para cada passo de tempo. A escolha do tipo de aco-

plamento contempla as características de cada método levando em conta o grau de dependência das duas físicas (PIZZOL, 2014).

Pizzol (2014) e Ribeiro (2016) utilizam em seus trabalhos um esquema iterativo de acoplamento geomecânico. No trabalho de Ferreira (2010) é apresentado um estudo sobre um método com acoplamento total e acoplamento iterativo. Sloan e Abbo (1999.a), Sloan e Abbo (1999.b) e Siqueira (1995) utilizam, em seus trabalhos, formulações totalmente acoplada.

No mercado atual há a disponibilidade de vários simuladores poroelásticos. Grande parte dessas ferramentas são utilizadas para o contexto de geomecânica de reservatórios de petróleo de petróleo como o STARS® (COMPUTER MODELLING GROUP LTD, 2013), o GEM® (COMPUTER MODELLING GROUP LTD, 2013) e o ECLIPSE® (SCHLUMBERGER, 2007). Além disso, vários grupos de pesquisa desenvolvem seus próprios códigos, onde normalmente são incorporadas várias formulações e técnicas de soluções de problemas poro-mecânicos, como é exemplo o CODE_BRIGHT, desenvolvido na Universidade Politécnica da Catalunha e utilizado e aperfeiçoado pelo grupo de Geomecânica do LMCG da Universidade Federal de Pernambuco (PASTALLE, 1995).

Desde o surgimento da computação os seus desenvolvedores fornecem à comunidade científica várias de linguagens computacionais para a elaboração de códigos e softwares. No cenário atual há uma disponibilidade de um grande acervo dessas linguagens que, cada vez mais, fornecem ao desenvolvedor maneiras de otimizar a utilização da máquina e até facilitar a implementação com ferramentas direcionadas para situações específicas. É notório que nem sempre a elaboração de um código com uma linguagem mais moderna trará um resultado mais satisfatório para determinada situação, por isso tais linguagens precisam ser escolhidas levando em consideração o conhecimento e bom senso em relação ao que se pretende desenvolver.

Nesse contexto, linguagens como FORTRAN (FERRER et al., 1992), C (THE MATH INC, 1992), C++ (ECKEL, 1991), Python® (BORGES, 2010) e MATLAB® : (THE MATH INC, 1992) são exemplos dessas linguagens. A programação em ambiente MATLAB®, por exemplo, em comparação com o FORTRAN, permite ao programador da área numérica uma economia em termos de linhas de código, pois essa linguagem apresenta uma biblioteca de funções específicas para o tratamento de matrizes. Em contrapartida o FORTRAN, em geral, permite ao programador a possibilidade de desenvolver códigos compilados com uma eficiên-

cia computacional consideravelmente melhor, porém a construção do código envolve um número maior de procedimentos por se uma linguagem de baixo nível.

O pré-processamento da simulação, que consiste na construção do modelo computacional e a geração da malha, e o pós-processamento, que consiste na interpretação e análise dos resultados, são operações que exigem do usuário, em geral, a utilização de outras ferramentas que se comunicam com o código desenvolvido. Há situações em que essas operações estão incorporadas ao próprio código desenvolvido ou até mesmo faz parte do mesmo programa, como é o caso da maioria dos softwares comerciais disponíveis no mercado. Alguns exemplos de software de pré-processamento são o GiD (CMNE, 2016), Gmsh (GEUZAINÉ; REMACLE, 2017) e o Gocad (BERLIOUX, 2001). O próprio GiD também possui o módulo de pós processamento e visualização, assim como o Visit (LLNL, 2005) e o Surfer (LANDIM; MONTEIRO; CORSI, 2002).

3 OBJETIVO DO TRABALHO

O principal objetivo do presente trabalho é a construção de um programa numérico para a solução numérica de problemas poroelásticos, com emprego de malhas não estruturadas, envolvendo anisotropia de tensor de permeabilidade e de tensões. Tal programa será desenvolvido em ambiente MATLAB integrando o pré-processador Gmsh.

Com isso, os objetivos específicos são:

- Validar o código desenvolvido comparando soluções numéricas com soluções analíticas de problemas consolidados na literatura;
- Verificar a convergência do erro associado à solução numérica quanto diferentes tipos de malhas utilizadas;
- Comparar a solução numérica com programas comerciais envolvendo problemas anisotrópicos e heterogêneos;
- Desenvolver um programa em Elementos Finitos Mistos;
- Avaliar a solução numérica com relação ao comportamento físico, resolvendo problemas em que o comportamento transiente das variáveis são distintos, como a utilização de materiais com propriedades heterogêneas.

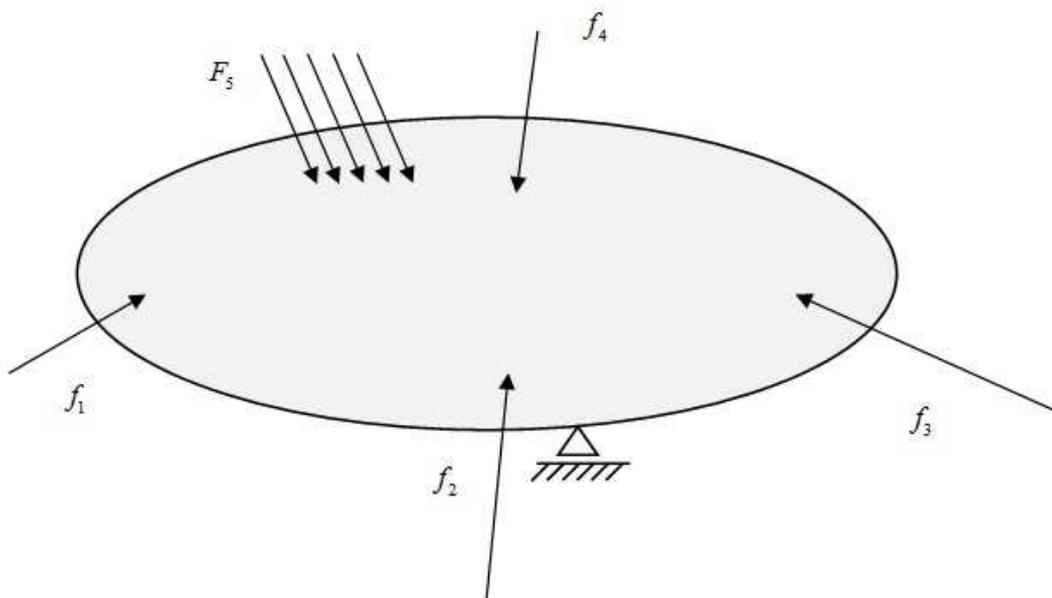
4 TEORIA DE BIOT

Nesse capítulo serão introduzidos os conceitos utilizados por Biot para a formulação da sua teoria.

4.1 EQUAÇÃO DE EQUILÍBRIO MECÂNICO

A equação de equilíbrio mecânico é utilizada, juntamente com a relação constitutiva do material em questão, para a determinação do comportamento mecânico do meio. Para a apresentação da equação de equilíbrio é necessária a introdução do conceito de esforços internos. Para isso, é necessário que se considere um corpo, submetido a condições de carregamentos e deslocamentos, como apresentado na Figura 4.1, em equilíbrio.

Figura 4.1. Corpo submetido a carregamentos externos.

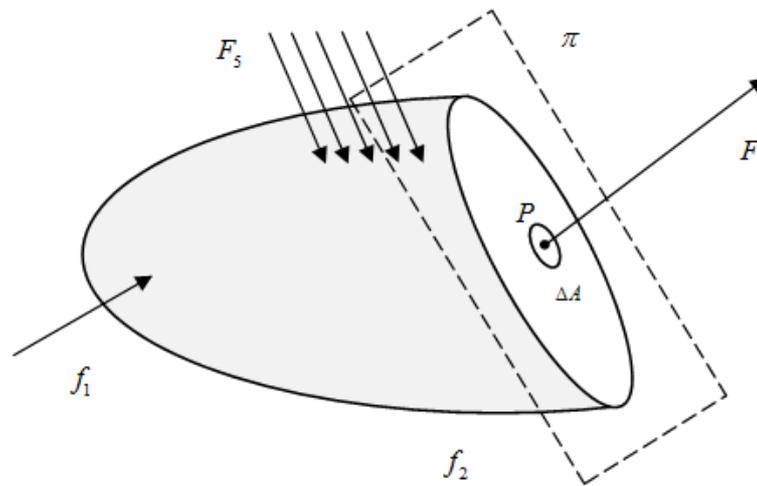


Fonte: Elaborada pelo autor.

Os esforços internos podem ser compreendidos como as forças reativas que aparecem, em determinado ponto, quando este serve de apoio para a parte da estrutura adjacente. Sendo

assim, usualmente se considera um plano, que divide o corpo em duas partes e um ponto, situado nesse plano, onde serão avaliados os esforços externos. Na Figura 4.2 o plano considerado é o plano π e o ponto P . A força F é a força equilibrante, correspondente às ações que são transmitidas a esse ponto pela parte não considerada do corpo.

Figura 4.2. Seção no corpo.



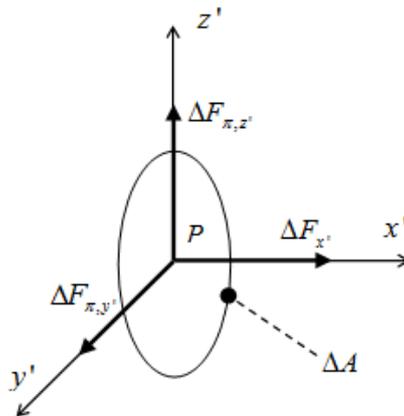
Fonte: Elaborada pelo autor.

A partir de um detalhe na área ΔA , que pode ser vista na Figura 4.3. Pode-se estabelecer um sistema de eixos coordenados a partir do ponto P , sendo esses os eixos x' , y' e z' , onde o eixo x' é normal ao plano π e os eixos y' e z' paralelos, como apresentado na Figura 4.3.

As forças $\Delta F_{x'}$, $\Delta F_{\pi, y'}$ e $\Delta F_{\pi, z'}$ são as componentes da força equilibradora F . A partir dessas definições, as tensões correspondentes às tais forças no ponto P são obtidas da seguinte maneira.

$$\begin{aligned}
 \sigma_{x'} &= \lim_{\Delta A \rightarrow 0} \frac{\Delta F_{x'}}{\Delta A} \\
 \tau_{\pi y'} &= \lim_{\Delta A \rightarrow 0} \frac{\Delta F_{\pi, y'}}{\Delta A} \\
 \tau_{\pi z'} &= \lim_{\Delta A \rightarrow 0} \frac{\Delta F_{\pi, z'}}{\Delta A}
 \end{aligned}
 \tag{4.1}$$

Figura 4.3. Detalhe do ponto P.



Fonte: Elaborada pelo autor.

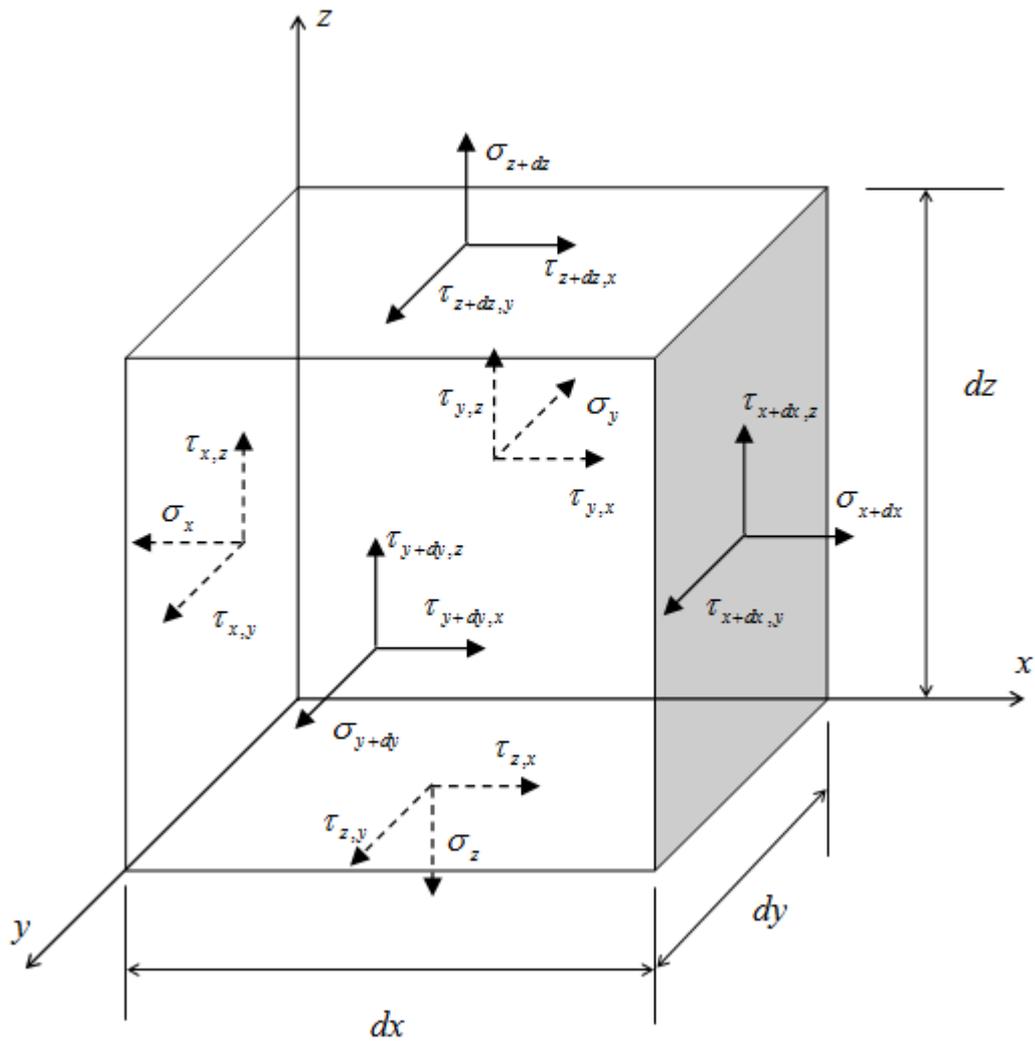
onde $\sigma_{x'}$, $\tau_{\pi, y'}$ e $\tau_{\pi, z'}$ são, respectivamente, a tensão normal, a tensão cisalhante na direção y e a tensão cisalhante na direção z em relação ao plano π .

Para que se obtenha o estado de tensões no ponto P é necessário que se definam as tensões normais e cisalhantes em relação aos eixos globais, que são dados por x , y e z . Essas tensões são apresentadas na Figura 4.4 e são definidas como σ_i é a tensão normal ao plano i e $\tau_{i, j}$ é a tensão cisalhante na direção do plano i e no sentido do eixo j , com i e j assumindo os valores de x , y e z .

Para que seja obtida a equação de equilíbrio mecânico é necessária a aplicação das condições de equilíbrio para o elemento diferencial apresentado na Figura 4.4, então.

$$\begin{aligned}
 \sum F_x = 0 \quad \sum M_x = 0 \\
 \sum F_y = 0 \quad \sum M_y = 0 \\
 \sum F_z = 0 \quad \sum M_z = 0
 \end{aligned}
 \tag{4.2}$$

Figura 4.4. Tensões em um elemento infinitesimal.



Fonte: Elaborada pelo autor.

onde F_i representa as forças na direção i e M_i os momentos na direção i .

Escrevendo a equação de equilíbrio de força na direção x e definindo b_x , b_y e b_z como sendo as forças de corpo respectivas às direções x , y e z , tem-se

$$\begin{aligned} \sum F_x = & \left(\sigma_x + \frac{\partial \sigma_x}{\partial x} dx \right) dydz - \sigma_x dydz + \left(\tau_{y,x} + \frac{\partial \tau_{y,x}}{\partial y} dy \right) dx dz - \tau_{y,x} dx dz \\ & + \left(\tau_{z,x} + \frac{\partial \tau_{z,x}}{\partial z} dz \right) dx dy - \left(\tau_{z,x} + \frac{\partial \tau_{z,x}}{\partial z} dz \right) dx dy - \tau_{z,x} dx dy + b_x dx dy dz = 0 \end{aligned} \quad (4.3)$$

Simplificando a Equação 4.3

$$\frac{\partial \sigma_x}{\partial x} + \frac{\partial \tau_{y,x}}{\partial y} + \frac{\partial \tau_{z,x}}{\partial z} + b_x = 0 \quad (4.4)$$

de forma análoga para as outras direções coordenadas, obtém-se.

$$\begin{aligned} \frac{\partial \tau_{x,y}}{\partial x} + \frac{\partial \sigma_y}{\partial y} + \frac{\partial \tau_{z,y}}{\partial z} + b_y &= 0 \\ \frac{\partial \tau_{z,x}}{\partial x} + \frac{\partial \tau_{y,z}}{\partial y} + \frac{\partial \sigma_z}{\partial z} + b_z &= 0 \end{aligned} \quad (4.5)$$

Escrevendo as equações de equilíbrio de momentos em relação ao eixo z , tem-se

$$\begin{aligned} \sum M_z = & \left(\tau_{y,x} + \frac{\partial \tau_{y,x}}{\partial y} dy \right) dx dz \frac{dy}{2} + \tau_{y,x} dx dz \frac{dy}{2} - \left(\tau_{x,y} + \frac{\partial \tau_{x,y}}{\partial x} dx \right) dy dz \frac{dx}{2} \\ & + \tau_{x,y} dy dz \frac{dx}{2} = 0 \end{aligned} \quad (4.6)$$

A simplificação da Equação 4.6 resulta em

$$\tau_{x,y} = \tau_{y,x} \quad (4.7)$$

de forma análoga, para o equilíbrio de momento em torno dos eixos coordenados das outras direções tem-se

$$\begin{aligned} \tau_{z,x} &= \tau_{x,z} \\ \tau_{z,y} &= \tau_{y,z} \end{aligned} \quad (4.8)$$

A partir das relações apresentadas pode-se observar que para a definição do estado de tensões de um corpo em um ponto determinado é necessário o conhecimento das seguintes variáveis: $\sigma_x, \sigma_y, \sigma_z, \tau_{x,y}, \tau_{x,z}$ e $\tau_{y,z}$.

O conjunto de equações de equilíbrio dado pela Equação 4.4 e pela Equação 4.5 é, usualmente, apresentado na forma matricial como.

$$\bar{\nabla} \cdot \underline{\underline{\sigma}} + \underline{\underline{b}}^T = \underline{\underline{0}} \quad (4.9)$$

onde $\underline{\underline{0}} = [0 \ 0 \ 0]$ e

$$\underline{\underline{\sigma}} = \begin{bmatrix} \sigma_x & \tau_{xy} & \tau_{xz} \\ \tau_{yx} & \sigma_y & \tau_{yz} \\ \tau_{zx} & \tau_{zy} & \sigma_z \end{bmatrix} \quad (4.10)$$

$$\underline{\underline{b}} = [b_x \ b_y \ b_z] \quad (4.11)$$

$$\bar{\nabla} = \begin{bmatrix} \frac{\partial}{\partial x} & \frac{\partial}{\partial y} & \frac{\partial}{\partial z} \end{bmatrix} \quad (4.12)$$

4.2 ESTADO PLANO DE DEFORMAÇÕES

Para determinado material, sujeito a um Estado de Tensões e ao equilíbrio mecânico, são definidas, de acordo com parâmetros obtidos a partir de experimentos ou leis empíricas, as leis que relacionam as tensões e deformações do meio. Para o presente trabalho, por se tratar de um meio elástico, essas relações são dadas pela Lei de Hooke. Essa lei foi descoberta pelo físico inglês Robert Hooke no ano de 1660 ao estudar o comportamento de um corpo elástico, porém só veio a ser publicada no ano de 1676. A Lei de Hooke possui sua forma geral, onde é definida a relação constitutiva elástica para o caso tridimensional e anisotrópico, porém, no presente trabalho, serão aplicadas à forma geral da Lei de Hooke algumas simplificações a fim de deixá-la com uma apresentação mais elegante para a aplicação considerada. Sendo assim, será considerado o meio sujeito ao regime do Estado Plano de Deformações e composto por um material isotrópico.

Quando se analisa um corpo submetido a um Estado de Tensões, em determinado ponto, pode-se definir, para tal ponto, variáveis básicas para o problema mecânico. Essas variáveis são os deslocamentos em cada direção coordenada, definidos como u_x , u_y e u_z e as ten-

sões já definidas anteriormente. A partir dos deslocamentos, são definidas as deformações ocorridas no ponto avaliado, que são, de fato, o fenômeno físico que gera o aparecimento de tensão em tal ponto, que são dadas por:

$$\begin{aligned}\varepsilon_x &= \frac{\partial u_x}{\partial x} \\ \varepsilon_y &= \frac{\partial u_y}{\partial x} \\ \varepsilon_z &= \frac{\partial u_z}{\partial x}\end{aligned}\tag{4.13}$$

onde ε_x , ε_y e ε_z são, respectivamente, as deformações nas direções x , y e z . Essas deformações se relacionam com as tensões através do Módulo de Elasticidade E e o coeficiente de Poisson do material de acordo com as expressões:

$$\begin{aligned}\varepsilon_x &= \frac{1}{E}[\sigma_x - \nu(\sigma_y + \sigma_z)] \\ \varepsilon_y &= \frac{1}{E}[\sigma_y - \nu(\sigma_x + \sigma_z)] \\ \varepsilon_z &= \frac{1}{E}[\sigma_z - \nu(\sigma_x + \sigma_y)]\end{aligned}\tag{4.14}$$

Também são definidas, para cada ponto do domínio, as deformações causadas por distorções no material. Essas distorções são definidas por:

$$\begin{aligned}\gamma_{xy} &= \frac{\partial u_x}{\partial y} + \frac{\partial u_y}{\partial x} \\ \gamma_{xz} &= \frac{\partial u_x}{\partial z} + \frac{\partial u_z}{\partial x} \\ \gamma_{yz} &= \frac{\partial u_y}{\partial z} + \frac{\partial u_z}{\partial y}\end{aligned}\tag{4.15}$$

que se relacionam com as tensões cisalhantes de acordo com as expressões:

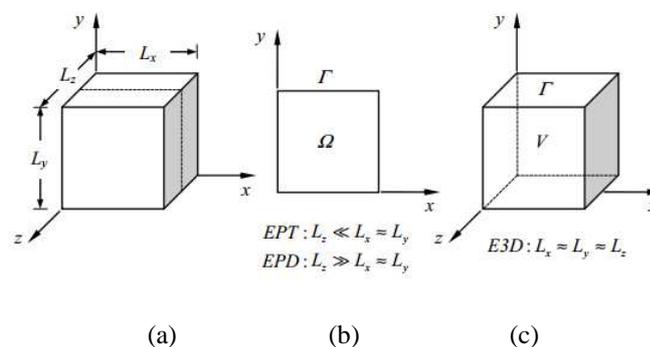
$$\begin{aligned}\gamma_{xy} &= \frac{\tau_{xy}}{G} \\ \gamma_{xz} &= \frac{\tau_{xz}}{G} \\ \gamma_{yz} &= \frac{\tau_{yz}}{G}\end{aligned}\tag{4.16}$$

onde G é o módulo de cisalhamento, definido como:

$$G = \frac{E}{2(1+\nu)} \quad (4.17)$$

Tendo em vista a natureza dos problemas geomecânicos, esses foram tratados, nesse trabalho, como sendo submetido ao estado plano de deformações. As hipóteses consideradas para a modelagem desses problemas são dependentes das relações entre as dimensões e características do problema a ser resolvido. Uma das considerações que podem ser atribuídas aos problemas são as considerações acerca das relações geométricas entre as dimensões do domínio. Quando o problema considerado se desenvolve com dimensões com a mesma ordem de grandeza nas três direções, esses problemas precisam ser modelados como sendo tridimensionais, sem considerações de simplificações em termos de tensões ou deslocamentos em alguma dessas direções. Quando uma das dimensões possui ordem de grandeza menor do que outras duas, as tensões desenvolvidas ao longo da menor direção podem ser desprezadas, quando comparadas com as tensões nas direções ao longo da seção transversal, sendo assim $\sigma_z = \tau_{xz} = \tau_{yz} = 0$. Quando uma das direções é ordens de grandeza maior que as outras duas as deformações desenvolvidas ao longo da maior direção são desprezadas, pois apresentam valores, na prática, bem menor que as demais, sendo assim $\gamma_{xz} = \gamma_{yz} = \varepsilon_z = 0$. A Figura 4.5 apresenta um esquema ilustrativo das situações citadas.

Figura 4.5. Tipos de análise: (a) Geometria do problema; (b) Estado Plano de Tensões e Estado Plano de Deformações; (c) Análise 3-D.



Fonte: Elaborada pelo autor.

A consideração acima será atribuída à Lei de Hooke Generalizada para que esta seja escrita na forma final, na qual será aplicada a formulação do Método dos Elementos Finitos,

com isso podemos reescrever a equação que rege a Lei de Hooke para o caso de um corpo submetido a um Estado Plano de Deformações como segue.

$$\underline{\underline{\sigma}} = \underline{\underline{\mathbf{D}}}\underline{\underline{\varepsilon}} \quad (4.18)$$

onde os termos dessa equação podem ser escritos na forma reduzida como:

$$\underline{\underline{\sigma}} = [\sigma_x \quad \sigma_y \quad \tau_{xy}]^T \quad (4.19)$$

$$\underline{\underline{\varepsilon}} = [\varepsilon_x \quad \varepsilon_y \quad \gamma_{xy}]^T \quad (4.20)$$

e

$$\underline{\underline{\mathbf{D}}} = \begin{bmatrix} D_{11} & D_{12} & 0 \\ D_{21} & D_{22} & 0 \\ 0 & 0 & D_{32} \end{bmatrix} \quad (4.21)$$

onde $\underline{\underline{\varepsilon}}$ é o vetor de deformações. A matriz $\underline{\underline{\mathbf{D}}}$ pode ser obtida a partir das seguintes manipulações.

Para o caso de Estado Plano de Deformações, tem-se:

$$\gamma_{xz} = \gamma_{yz} = \varepsilon_z = 0 \quad (4.22)$$

então, pode-se escrever:

$$\varepsilon_z = \frac{1}{E} [\sigma_z - \nu(\sigma_x + \sigma_y)] = 0 \quad (4.23)$$

obtendo assim:

$$\sigma_z = \nu(\sigma_x + \sigma_y) \quad (4.24)$$

Pode-se notar que apesar de $\varepsilon_z = 0$, a tensão σ_z não necessariamente será nula. Substituindo a Equação 4.23 na Equação 4.13 e manipulando algebricamente o conjunto de equações obtido, tem-se:

$$\begin{aligned}
\varepsilon_x &= \frac{1}{E} \left[\sigma_x - \nu \sigma_y - \nu^2 (\sigma_x + \sigma_y) \right] \\
\varepsilon_y &= \frac{1}{E} \left[\sigma_y - \nu \sigma_x - \nu^2 (\sigma_x + \sigma_y) \right] \\
\varepsilon_z &= 0 \\
\gamma_{xy} &= \frac{\tau_{xy}}{G}
\end{aligned} \tag{4.25}$$

Através de manipulações no conjunto dado na Equação 4.25 pode-se definir a matriz constitutiva para o Estado Plano de Deformações como

$$\mathbf{D} = \frac{E(1-\nu)}{(1+\nu)(1-2\nu)} \begin{bmatrix} 1 & \frac{\nu}{1-\nu} & 0 \\ \frac{\nu}{1-\nu} & 1 & 0 \\ 0 & 0 & \frac{1-2\nu}{2(1-\nu)} \end{bmatrix} \tag{4.26}$$

4.3 EQUAÇÃO DE BALANÇO DE MASSA DE FLUIDO

A equação de balanço de massa de fluido é uma equação diferencial parcial que rege o comportamento do fluxo de fluido em determinado meio. O desenvolvimento da equação é dado a partir da análise de um elemento infinitesimal de dimensões dx , dy e dz correspondentes às direções x , y e z respectivamente, que, sem perda de generalidade, possui os três lados paralelos aos eixos coordenados, como apresentado na Figura 4.6.

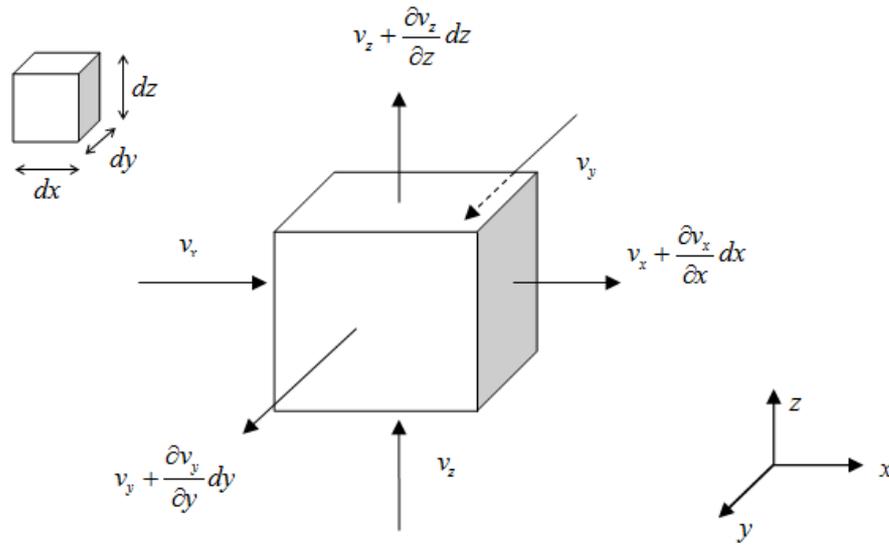
As componentes do vetor velocidade (\mathbf{v}) do fluxo do fluido são dadas por v_x , v_y e v_z dadas nas direções x , y e z respectivamente.

Pode-se de definir o balanço de massa de fluido no interior do elemento definido, para determinado momento, como sendo a soma das contribuições de massa que entra e sai do elemento em tal instante. Tais quantidades são definidas em cada face considerada, também há a inserção, na equação do problema, de um termo de fonte ou sumido que pode ocorrer no ponto avaliado.

Avaliando o balanço de massa na direção x tem-se:

$$\text{balanço de massa em } x \rightarrow v_x dydz - v_x dydz + \frac{\partial v_x}{\partial x} dx dy dz \quad (4.27)$$

Figura 4.6. Velocidade em um elemento infinitesimal.



Fonte: Elaborada pelo autor.

Avaliando as contribuições nas três direções e o termo de fonte ou sumidoro, definido por q , tem-se:

$$\begin{aligned} v_x dydz - v_x dydz + \frac{\partial v_x}{\partial x} dx dy dz + v_y dx dz - v_y dx dz + \frac{\partial v_y}{\partial y} dy dx dz + v_z dx dy - v_z dx dy \\ + \frac{\partial v_z}{\partial z} dz dx dy + q dx dy dz = 0 \end{aligned} \quad (4.28)$$

Simplificando a equação do balanço de massa, tem-se:

$$\frac{\partial v_x}{\partial x} + \frac{\partial v_y}{\partial y} + \frac{\partial v_z}{\partial z} + q = 0 \quad (4.29)$$

que pode ser escrita de maneira simplificada como:

$$\bar{\nabla} \cdot \underline{\mathbf{v}} + q = 0 \quad (4.30)$$

4.4 LEI DE DARCY

A Lei de Darcy, assim como a Lei de Hooke, é uma lei constitutiva que relaciona a velocidade do fluido, em um meio poroso, com o gradiente de pressões de fluido em determinado ponto do domínio, a partir de propriedades associadas ao meio e ao fluido considerado. Essa lei foi proposta por Henry Philibert Gaspard Darcy no ano de 1856, a partir de experimentos realizados em estruturas contendo areia a submetidas a gradientes de pressões.

Desprezando os efeitos de capilaridade e gravidade, a equação de rege a Lei de Darcy pode ser escrita como

$$\underline{\mathbf{v}} = -\underline{\underline{\mathbf{K}}} \left(\frac{\bar{\nabla} p}{\gamma_f} \right) \quad (4.31)$$

onde

$$\underline{\mathbf{v}} = [v_x \quad v_y \quad v_z] \quad (4.32)$$

$$\underline{\underline{\mathbf{K}}} = \begin{bmatrix} k_x & k_{xy} & k_{xz} \\ k_{yx} & k_y & k_{yz} \\ k_{zx} & k_{zy} & k_z \end{bmatrix} \quad (4.33)$$

O vetor de velocidade do fluido $\underline{\mathbf{v}}$ possui as componentes v_x , v_y e v_z que são as velocidades nas direções x , y e z respectivamente. $\underline{\underline{\mathbf{K}}}$ é o tensor de permeabilidades do problema, γ_w o peso específico do fluido e p a pressão de fluido no ponto avaliado. As componentes do tensor de permeabilidades são mostradas na Equação 4.33

4.5 PRINCÍPIO DAS TENSÕES EFETIVAS

O Princípio das tensões efetivas foi desenvolvido por Terzaghi (CAPUTO, 1998), e consiste na relação entre as variáveis mecânicas e hidráulicas do problema, que pode ser escrito como:

$$\underline{\underline{\sigma}} = \underline{\underline{\sigma}}' + \underline{\underline{I}}p \quad (4.34)$$

onde $\underline{\underline{\sigma}}$ é o tensor de tensões totais dado por:

$$\underline{\underline{\sigma}} = \begin{bmatrix} \sigma_x & \tau_{xy} & \tau_{xz} \\ \tau_{yx} & \sigma_y & \tau_{yz} \\ \tau_{zx} & \tau_{zy} & \sigma_z \end{bmatrix} \quad (4.35)$$

e $\underline{\underline{\sigma}}'$ é o tensor de tensões efetivas dado por:

$$\underline{\underline{\sigma}}' = \begin{bmatrix} \sigma'_x & \tau'_{xy} & \tau'_{xz} \\ \tau'_{yx} & \sigma'_y & \tau'_{yz} \\ \tau'_{zx} & \tau'_{zy} & \sigma'_z \end{bmatrix} \quad (4.36)$$

onde σ_i / σ'_i são as tensões totais/efetivas que se desenvolvem nas direções normais à direção coordenada i e τ_{ij} / τ'_{ij} são as tensões cisalhantes totais/efetivas que se desenvolver na direção i ao longo do plano normal à direção j .

onde $\underline{\underline{I}}$ é a matriz identidade dada por:

$$\underline{\underline{I}} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (4.37)$$

e p é a pressão de fluido.

4.6 SISTEMA DE EQUAÇÕES DA TEORIA GERAL DO ADENSAMENTO DE BIOT

As equações diferenciais que regem o problema poroelástico são obtidas utilizando os conceitos de Equação de Equilíbrio Mecânico, Estado Plano de Deformações, Equação de Balanço de Massa de Fluido, Lei de Darcy e o Princípio das Tensões Efetivas.

Utilizando o *Princípio das Tensões Efetivas* podemos escrever a Equação de Equilíbrio Mecânico como:

$$\nabla \cdot \underline{\underline{\boldsymbol{\sigma}'}} + \underline{\underline{\mathbf{b}}} = \underline{\underline{\mathbf{0}}} \quad (4.38)$$

O balanço de massa do fluido é igual à variação volumétrica do meio, esse fenômeno é decorrente da hipótese do fluido ser incompressível e está preenchendo todos os poros do meio, ou seja, está completamente saturado, sendo assim a variação espacial do volume de fluido no meio é dada por:

$$\bar{\nabla} \cdot \underline{\underline{\mathbf{v}}} = \bar{\nabla} \cdot \left(\begin{array}{c} \bar{\nabla} p \\ \underline{\underline{\boldsymbol{\kappa}}} \\ \gamma_f \end{array} \right) \quad (4.39)$$

A taxa de variação volumétrica do elemento é dada por:

$$\frac{\partial}{\partial t} (\varepsilon_x + \varepsilon_y + \varepsilon_z) \quad (4.40)$$

onde t é a variável temporal do problema.

Igualando a Equação 4.39 e a Equação 4.40, tem-se

$$\bar{\nabla} \cdot \left(\begin{array}{c} \bar{\nabla} p \\ \underline{\underline{\boldsymbol{\kappa}}} \\ \gamma_f \end{array} \right) = \frac{\partial}{\partial t} (\varepsilon_x + \varepsilon_y + \varepsilon_z) \quad (4.41)$$

e escrevendo essa relação na forma:

$$\bar{\nabla} \cdot (\underline{\underline{\mathbf{v}}}) - \frac{\partial}{\partial t} (\varepsilon_x + \varepsilon_y + \varepsilon_z) = 0 \quad (4.42)$$

Então, o sistema de equações que rege o problema poroelástico é dado por.

$$\begin{cases} \overline{\nabla} \cdot \underline{\underline{\boldsymbol{\sigma}}}' + \underline{\mathbf{b}} = 0, \text{ onde } \underline{\underline{\boldsymbol{\sigma}}}' = \underline{\underline{\boldsymbol{\sigma}}} - Ip, \text{ onde } \underline{\underline{\boldsymbol{\sigma}}} = \underline{\underline{\mathbf{D}}}\underline{\underline{\boldsymbol{\varepsilon}}} \\ \overline{\nabla} \cdot (\underline{\mathbf{v}}) - \frac{\partial}{\partial t}(\varepsilon_x + \varepsilon_y + \varepsilon_z) = 0, \text{ onde } \underline{\mathbf{v}} = -\underline{\underline{\boldsymbol{\kappa}}}\frac{\overline{\nabla} p}{\gamma_f} \end{cases} \quad (4.43)$$

O sistema de equações acima apresenta as equações que regem a Teoria Geral do A-
densamento de Biot.

5 FORMULAÇÃO DO MÉTODO DOS ELEMENTOS FINITOS APLICADA À TEORIA DE BIOT

Para a discretização do sistema de equações que rege o problema poroelástico, será utilizado, nesse trabalho, uma formulação mista do Método dos Elementos Finitos. Essa formulação é utilizada para a solução do problema totalmente acoplado, onde as duas físicas, mecânica e hidráulica, são resolvidas em uma única matriz global do problema.

Para o desenvolvimento da aproximação numérica do problema, o Princípio das Tensões Efetivas será reescrito da seguinte maneira:

$$\underline{\sigma} = \underline{\sigma}' + \underline{\mathbf{m}}p \quad (5.1)$$

onde $\underline{\mathbf{m}} = [1 \quad 1 \quad 0]^T$.

Diferenciando a equação acima em relação ao tempo, tem-se:

$$\frac{\partial}{\partial t}(\underline{\sigma}) = \frac{\partial}{\partial t}(\underline{\sigma}') + \underline{\mathbf{m}} \frac{\partial}{\partial t}(p) \quad (5.2)$$

Aplicando o Método dos Resíduos ponderados à Equação de Equilíbrio Mecânico, temos:

$$\int_V \underline{\mathbf{w}}^T (\nabla^T \underline{\sigma} + \underline{\mathbf{b}}) dV = 0 \quad (5.3)$$

onde $\underline{\mathbf{w}} = [w_x \quad w_y]^T$ é o vetor com as funções de formas correspondentes às direções x e y .

Integrando por partes essa equação e usando o Teorema de Gauss, é obtida a forma fraca da Equação de Equilíbrio Mecânico, dada por:

$$\int_V (\nabla \underline{\mathbf{w}})^T \underline{\sigma} dV - \int_V \underline{\mathbf{w}}^T \underline{\mathbf{b}} dV - \int_S \underline{\mathbf{w}}^T \underline{\mathbf{t}} dS = 0 \quad (5.4)$$

onde $\underline{\mathbf{t}} = [t_x \quad t_y]^T$ é o vetor de forças de superfície aplicados na superfície S . As forças superficiais devem satisfazer as condições de contorno:

$$\begin{aligned} t_x &= \sigma_x n_x + \tau_{xy} n_y \\ t_y &= \tau_{xy} n_x + \sigma_y n_y \end{aligned} \quad (5.5)$$

sendo n_x e n_y são os cossenos diretores normais à superfície S .

Diferenciando no tempo a Equação 5.4 e aplicando o Princípio das Tensões efetivas, obtém-se:

$$\int_V (\nabla \mathbf{w})^T \left(\frac{\partial \underline{\underline{\boldsymbol{\sigma}}}}{\partial t} + \mathbf{m} \frac{\partial p}{\partial t} \right) dV - \int_V \mathbf{w}^T \frac{\partial \mathbf{b}}{\partial t} dV - \int_S \mathbf{w}^T \frac{\partial \mathbf{t}}{\partial t} dS = 0 \quad (5.6)$$

Será aplicado, à Equação 5.6 a discretização pelo Método dos Elementos Finitos. Essa aproximação consiste em dividir o domínio em elementos, onde as variáveis do problema que, para o presente caso, é o deslocamento, serão interpoladas ao longo do domínio por funções chamadas de funções interpoladoras, construídas em termos de pontos estabelecidos no elemento de interesse. Para isso é necessário que sejam definidas essas funções para as variáveis analisadas. A interpolação dos deslocamentos a nível de elemento é dada de acordo com a expressão abaixo:

$$\underline{\underline{\mathbf{d}}} = \underline{\underline{\mathbf{N}}}_u \underline{\underline{\mathbf{u}}} \quad (5.7)$$

onde $\underline{\underline{\mathbf{d}}} = [u \ v]$ sendo u o deslocamento na direção x e v o deslocamento na direção y . A matriz $\underline{\underline{\mathbf{N}}}_u$ corresponde à matriz contendo as funções de forma associadas ao elemento e é dada por:

$$\underline{\underline{\mathbf{N}}}_u = \begin{bmatrix} N_{u1} & 0 & N_{u2} & 0 & \dots & N_{u6} & 0 \\ 0 & N_{u1} & 0 & N_{u2} & \dots & 0 & N_{u6} \end{bmatrix} \quad (5.8)$$

O vetor $\underline{\underline{\mathbf{u}}}$ é o vetor que contém os deslocamentos nodais, que são os graus de liberdade do elemento, e é dado por:

$$\underline{\underline{\mathbf{u}}} = [u_1 \ v_1 \ u_2 \ v_2 \ \dots \ u_6 \ v_6]^T \quad (5.9)$$

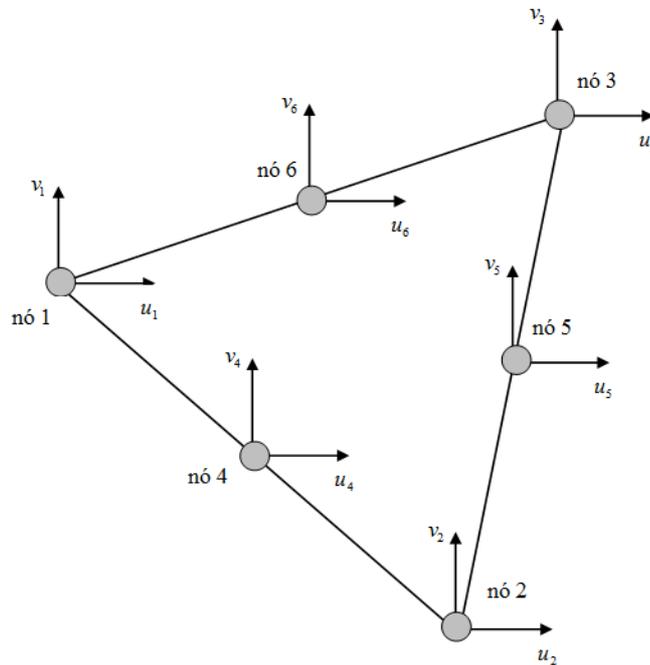
onde os deslocamentos são dados em cada nó, onde u_i corresponde ao deslocamento na direção x do nó com numeração i e os deslocamentos v_i são os deslocamentos na direção y dados de maneira análoga. A Figura 5.1 apresenta a localização dos deslocamentos nodais.

Para o problema mecânico, como apresentado, cada nó possui dois graus de liberdade, relativos ao deslocamento na direção x e ao deslocamento na direção y . Esses deslocamentos podem ser aproximados de forma polinomial como segue:

$$\begin{aligned} u &= b_1 + b_2x + b_3y + b_4x^2 + b_5xy + b_6y^2 \\ v &= b_7 + b_8x + b_9y + b_{10}x^2 + b_{11}xy + b_{12}y^2 \end{aligned} \quad (5.10)$$

sendo os coeficientes $b_i = 1, \dots, 12$ são constantes.

Figura 5.1. Deslocamentos nodais utilizados na função interpoladora dos deslocamentos.



Fonte: Elaborada pelo autor.

A Equação 5.10 pode ser escrita de forma matricial como:

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} 1 & x & y & x^2 & xy & y^2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & x & y & x^2 & xy & y^2 \end{bmatrix} \begin{bmatrix} b_1 \\ \vdots \\ b_{12} \end{bmatrix} \quad (5.11)$$

que pode ser escrita de maneira mais compacta como:

$$\begin{bmatrix} u \\ v \end{bmatrix} = \underline{\underline{\mathbf{M}}}\underline{\underline{\mathbf{b}}} \quad (5.12)$$

Os deslocamentos nodais, para um determinado elemento, satisfazem o sistema de equações abaixo:

$$\begin{cases} u_1 = b_1 + b_2 x_1 + b_3 y_1 + b_4 x_1^2 + b_5 x_1 y_1 + b_6 y_1^2 \\ \vdots \\ u_6 = b_1 + b_2 x_6 + b_3 y_6 + b_4 x_6^2 + b_5 x_6 y_6 + b_6 y_6^2 \\ v_1 = b_7 + b_8 x_1 + b_9 y_1 + b_{10} x_1^2 + b_{11} x_1 y_1 + b_{12} y_1^2 \\ \vdots \\ v_6 = b_7 + b_8 x_6 + b_9 y_6 + b_{10} x_6^2 + b_{11} x_6 y_6 + b_{12} y_6^2 \end{cases} \quad (5.13)$$

onde x_i e y_i com $i=1, \dots, 6$ são, respectivamente, os valores das coordenadas dos nós de numeração i do elemento avaliado, dado de acordo com a Figura 5.1.

Esse sistema de equações pode ser manipulado algebricamente e escrito, de maneira matricial, como:

$$\begin{bmatrix} b_1 \\ \vdots \\ b_1 \\ b_1 \\ \vdots \\ b_1 \end{bmatrix} = \begin{bmatrix} 1 & x_1 & y_1 & x_1^2 & x_1 y_1 & y_1^2 & 0 & 0 & 0 & 0 & 0 & 0 \\ \vdots & \vdots \\ 1 & x_6 & y_6 & x_6^2 & x_6 y_6 & y_6^2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & x_1 & y_1 & x_1^2 & x_1 y_1 & y_1^2 \\ \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & x_6 & y_6 & x_6^2 & x_6 y_6 & y_6^2 \end{bmatrix}^{-1} \begin{bmatrix} u_1 \\ \vdots \\ u_6 \\ v_1 \\ \vdots \\ v_6 \end{bmatrix} \quad (5.14)$$

Escrevendo a Equação 5.14 de forma compacta temos:

$$\underline{\underline{\mathbf{b}}} = \underline{\underline{\mathbf{X}}}^{-1} \underline{\underline{\mathbf{u}}} \quad (5.15)$$

Manipulando adequadamente as expressões anteriores obtém-se a seguinte relação:

$$\underline{\underline{\mathbf{N}}}_u = \underline{\underline{\mathbf{M}}}\underline{\underline{\mathbf{X}}}^{-1} \quad (5.16)$$

que define a matriz contendo as funções de forma para o elemento triangular quadrático, utilizado para a definição das variáveis mecânicas do problema.

De forma análoga a interpolação da pressão é dada pela Equação 5.17 (

$$p = \underline{\underline{\mathbf{N}}}_p \underline{\underline{\mathbf{p}}} \quad (5.17)$$

sendo:

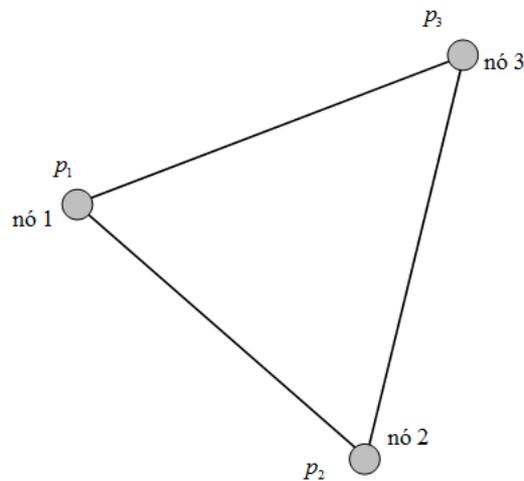
$$\underline{\mathbf{N}}_p = [N_{p1} \quad N_{p2} \quad N_{p3}] \quad (5.18)$$

que é a matriz contendo as funções interpoladoras da pressão e a o vetor $\underline{\mathbf{p}}$ é dado como:

$$\underline{\mathbf{p}} = [p_1 \quad p_2 \quad p_3] \quad (5.19)$$

é o vetor que contém as pressões nodais, definidas de acordo com a Figura 5.2:

Figura 5.2. Pressões nodais utilizados na função interpoladora das pressões.



Fonte: Elaborada pelo autor.

De forma análoga ao problema mecânico, podem ser obtidas as funções interpoladoras do problema hidráulico, e são dadas por:

$$\begin{aligned} N_{1p} &= \frac{1}{2A} \{ (x_2 y_3 - x_3 y_2) + (y_2 - y_3)x + (x_3 - x_2)y \} \\ N_{2p} &= \frac{1}{2A} \{ (x_3 y_1 - x_1 y_3) + (y_3 - y_1)x + (x_1 - x_3)y \} \\ N_{3p} &= \frac{1}{2A} \{ (x_1 y_2 - x_2 y_1) + (y_1 - y_2)x + (x_2 - x_1)y \} \end{aligned} \quad (5.20)$$

onde A é a área do triângulo que define o elemento.

Para esse caso, as coordenadas x_i e y_i são associadas aos nós $i=1,2,3$ de acordo com a Figura 5.2.

Desta forma estão definidas todas as funções de forma necessárias para as interpolações das variáveis hidráulicas e mecânicas.

Definidas as funções de forma, pode-se obter a velocidade aplicando a derivada temporal ao vetor $\underline{\mathbf{d}}$.

$$\underline{\mathbf{v}} = \frac{\partial \underline{\mathbf{d}}}{\partial t} = \underline{\mathbf{N}}_u \frac{\partial \underline{\mathbf{u}}}{\partial t} \quad (5.21)$$

As funções peso para a aplicação do Método de Galerkin à Equação 5.6 podem ser tomadas como a velocidade, que são dadas por:

$$\underline{\mathbf{w}} = \delta \frac{\partial \underline{\mathbf{d}}}{\partial t} = \underline{\mathbf{N}}_u \delta \frac{\partial \underline{\mathbf{u}}}{\partial t} \quad (5.22)$$

Substituindo a função peso na Equação 5.6 (e a integrando no volume elementar e área elementar, dados respectivamente por V^e e S^e , obtém-se:

$$\delta \frac{\partial \underline{\mathbf{u}}}{\partial t} \left[\int_{V^e} (\bar{\mathbf{V}} \underline{\mathbf{N}}_u)^T \frac{\partial \underline{\boldsymbol{\sigma}}'}{\partial t} dV + \int_{V^e} (\bar{\mathbf{V}} \underline{\mathbf{N}}_u)^T \underline{\mathbf{m}} \frac{\partial p}{\partial t} dV - \int_{S^e} \underline{\mathbf{N}}_u^T \frac{\partial \underline{\mathbf{t}}}{\partial t} dS - \int_{V^e} \underline{\mathbf{N}}_u^T \frac{\partial \underline{\mathbf{b}}}{\partial t} dV \right] = 0 \quad (5.23)$$

Como as funções peso podem ser arbitrárias, o termo entre colchetes da Equação 5.23 assume o valor nulo.

$$\int_{V^e} \underline{\mathbf{B}}_u^T \frac{\partial \underline{\boldsymbol{\sigma}}'}{\partial t} dV + \int_{V^e} \underline{\mathbf{B}}_u^T \underline{\mathbf{m}} \frac{\partial p}{\partial t} dV - \int_{S^e} \underline{\mathbf{N}}_u^T \frac{\partial \underline{\mathbf{t}}}{\partial t} dS - \int_{V^e} \underline{\mathbf{N}}_u^T \frac{\partial \underline{\mathbf{b}}}{\partial t} dV = 0 \quad (5.24)$$

onde a matriz $\underline{\mathbf{B}}_u$ é dada por:

$$\underline{\mathbf{B}}_u = \bar{\mathbf{V}} \underline{\mathbf{N}}_u = \begin{bmatrix} \frac{\partial N_{u1}}{\partial x} & 0 & \dots & \frac{\partial N_{u6}}{\partial x} & 0 \\ 0 & \frac{\partial N_{u1}}{\partial y} & \dots & 0 & \frac{\partial N_{u6}}{\partial y} \\ \frac{\partial N_{u1}}{\partial y} & \frac{\partial N_{u1}}{\partial x} & \dots & \frac{\partial N_{u6}}{\partial y} & \frac{\partial N_{u6}}{\partial x} \end{bmatrix} \quad (5.25)$$

A matriz $\underline{\mathbf{B}}_u$ associa os deslocamentos às deformações no elemento. As deformações são dadas pelo vetor $\underline{\boldsymbol{\varepsilon}} = [\varepsilon_x \quad \varepsilon_y \quad \gamma_{xy}]^T$ de acordo com a relação:

$$\underline{\underline{\boldsymbol{\varepsilon}}} = \overline{\nabla} \underline{\underline{\mathbf{d}}} = \overline{\nabla} \underline{\underline{\mathbf{N}_u}} \underline{\underline{\mathbf{u}}} = \underline{\underline{\mathbf{B}_u}} \underline{\underline{\mathbf{u}}} \quad (5.26)$$

Na Teoria da Elasticidade, as tensões são dadas por:

$$\underline{\underline{\boldsymbol{\sigma}}} = \underline{\underline{\mathbf{D}}} \underline{\underline{\boldsymbol{\varepsilon}}} \quad (5.27)$$

ou ainda:

$$\underline{\underline{\boldsymbol{\sigma}}} = \underline{\underline{\mathbf{D}}} \underline{\underline{\mathbf{B}_u}} \underline{\underline{\mathbf{u}}} \quad (5.28)$$

Substituindo a Equação 5.28 na (5. Equação 5.24 e escrevendo de uma forma mais apresentável, obtém-se a equação de equilíbrio para um elemento em termos da derivada temporal, dada por:

$$\underline{\underline{\mathbf{k}}} \frac{\partial \underline{\underline{\mathbf{u}}}}{\partial t} + \underline{\underline{\mathbf{l}}} \frac{\partial p}{\partial t} = \frac{\partial \underline{\underline{\mathbf{f}^{ext}}}}{\partial t} \quad (5.9)$$

onde:

$$\begin{aligned} \underline{\underline{\mathbf{k}}} &= \int_{V^e} \underline{\underline{\mathbf{B}_u}}^T \underline{\underline{\mathbf{D}}} \underline{\underline{\mathbf{B}_u}} dV \\ \underline{\underline{\mathbf{l}}} &= \int_{V^e} \underline{\underline{\mathbf{B}_u}}^T \underline{\underline{\mathbf{m}}} \underline{\underline{\mathbf{N}_p}} dV \end{aligned} \quad (5.30)$$

As matrizes definidas constituem as matrizes elementares, ou seja, são definidas para um determinado elemento da malha computacional avaliada. O vetor $\underline{\underline{\mathbf{f}^{ext}}}$ é definido como:

$$\underline{\underline{\mathbf{f}^{ext}}} = \int_{S^e} \underline{\underline{\mathbf{N}_u}}^T \frac{\partial \underline{\underline{\mathbf{t}}}}{\partial t} dS - \int_{V^e} \underline{\underline{\mathbf{N}_u}}^T \frac{\partial \underline{\underline{\mathbf{b}}}}{\partial t} dV \quad (5.31)$$

Para a descrição completa do problema de adensamento, é necessária a avaliação da equação de Conservação de Massa para o meio poroso. Tal equação é dada, para o caso bidimensional, por:

$$\frac{\partial v_x}{\partial x} + \frac{\partial v_y}{\partial y} = - \left(\frac{\partial \varepsilon_x}{\partial t} + \frac{\partial \varepsilon_y}{\partial t} \right) \quad (5.32)$$

Essa equação expressa à igualdade entre a variação de fluido no meio poroso e a taxa de deformação volumétrica do esqueleto sólido do meio poroso. Nesse processo é considerado

a indeformabilidade dos grãos que formam o esqueleto sólido. Essa equação pode ser apresentada, de forma compacta como:

$$\bar{\nabla} \cdot \underline{\mathbf{v}} + \underline{\mathbf{m}}^T \frac{\partial \underline{\boldsymbol{\varepsilon}}}{\partial t} = 0 \quad (5.33)$$

onde $\underline{\mathbf{v}} = [v_x \ v_y]^T$ dado por:

$$\underline{\mathbf{v}} = \frac{\underline{\mathbf{K}} \bar{\nabla} p}{\gamma_w} \quad (5.34)$$

e o tensor de permeabilidades $\underline{\mathbf{K}}$, para o caso bidimensional é dado por:

$$\underline{\mathbf{K}} = \begin{bmatrix} k_x & k_{xy} \\ k_{yx} & k_y \end{bmatrix} \quad (5.35)$$

Na Equação 5.34 pode ser introduzida a componente de fluxo gerada pelo efeito gravitacional, atribuído ao peso próprio do fluido. Dessa forma, a Equação 5.34 será reescrita como:

$$\underline{\mathbf{v}} = \frac{\underline{\mathbf{K}} (\bar{\nabla} p - \underline{\mathbf{b}}_w)}{\gamma_w} \quad (5.36)$$

onde $\underline{\mathbf{b}}_w = [0 \ \gamma_w]$ é o vetor de forças, que considera os efeitos gravitacionais.

Aplicando a equação que representa a Lei de Darcy na equação que apresenta a Conservação de Massa de fluido, pode-se obter:

$$\nabla \cdot \left[\frac{\underline{\mathbf{K}} (\bar{\nabla} p - \underline{\mathbf{b}}_w)}{\gamma_w} \right] + \underline{\mathbf{m}}^T \frac{\partial \underline{\boldsymbol{\varepsilon}}}{\partial t} = 0 \quad (5.37)$$

A forma fraca da Equação 5.37 é obtida aplicando o Método dos Resíduos Ponderados de Galerkin com as funções peso dadas por ω .

$$\int_V \underline{\omega} \nabla \cdot \left[\frac{\underline{\mathbf{K}} (\bar{\nabla} p - \underline{\mathbf{b}}_w)}{\gamma_w} \right] dV + \int_V \underline{\omega} \underline{\mathbf{m}}^T \frac{\partial \underline{\boldsymbol{\varepsilon}}}{\partial t} dV = 0 \quad (5.38)$$

Utilizando, no primeiro termo da equação o Teorema de Gauss, obtém-se:

$$\int_S \underline{\omega} (\bar{\nabla} p)^T \frac{\underline{\mathbf{K}}}{\gamma_w} \underline{\mathbf{n}} dS - \int_V (\bar{\nabla} \underline{\omega})^T \frac{\underline{\mathbf{K}}}{\gamma_w} (\underline{\mathbf{b}}_w) dV + \int_V \underline{\omega} \underline{\mathbf{m}}^T \frac{\partial \underline{\boldsymbol{\varepsilon}}}{\partial t} dV = 0 \quad (5.39)$$

que também pode ser apresentada da forma:

$$\int_V (\bar{\nabla} \underline{\omega})^T \frac{\underline{\mathbf{K}}}{\gamma_w} \bar{\nabla} p dV - \int_V \underline{\omega} \underline{\mathbf{m}}^T \frac{\partial \underline{\boldsymbol{\varepsilon}}}{\partial t} dV - \int_S \underline{\omega} q dS - \int_V (\bar{\nabla} \underline{\omega})^T \frac{\underline{\mathbf{K}}}{\gamma_w} \underline{\mathbf{b}}_w dV = 0 \quad (5.40)$$

sendo:

$$q = (\bar{\nabla} p - \underline{\mathbf{b}}_w)^T \frac{\underline{\mathbf{K}}}{\gamma_w} \underline{\mathbf{n}} = \underline{\mathbf{v}}^T \underline{\mathbf{n}} \quad (5.41)$$

O vetor $\underline{\mathbf{n}} = [n_x \quad n_y]^T$ indica os cossenos diretores normais à superfície S avaliada nas integrais apresentadas.

Definidas as funções interpoladoras para as variáveis hidráulicas, pode-se definir como função peso para o Método de Galerkin a função $\underline{\omega}$, dada por:

$$\underline{\omega} = \delta p = \underline{\underline{\mathbf{N}}}_p \delta \underline{\mathbf{p}} \quad (5.42)$$

onde $\delta \underline{\mathbf{p}}$ assume valores arbitrários ao longo do domínio. Esse procedimento é análogo ao procedimento utilizado para a Equação de Equilíbrio Mecânico. Diferenciando a função peso ao longo das variáveis espaciais obtém-se:

$$\bar{\nabla} \underline{\omega} = \bar{\nabla} \underline{\underline{\mathbf{N}}}_p \delta \underline{\mathbf{p}} = \underline{\underline{\mathbf{B}}}_p \delta \underline{\mathbf{p}} \quad (5.43)$$

sendo:

$$\underline{\underline{\mathbf{B}}}_p = \bar{\nabla} \underline{\underline{\mathbf{N}}}_p = \begin{bmatrix} \frac{\partial N_{p1}}{\partial x} & \frac{\partial N_{p2}}{\partial x} & \frac{\partial N_{p3}}{\partial x} \\ \frac{\partial N_{p1}}{\partial y} & \frac{\partial N_{p2}}{\partial y} & \frac{\partial N_{p3}}{\partial y} \end{bmatrix} \quad (5.44)$$

A Equação 5.40 combinada com as definições apresentadas pode ser escrita da forma:

$$\delta \underline{\mathbf{p}}^T \left(\int_{V^e} \underline{\mathbf{B}}_u^T \underline{\mathbf{m}}^T \underline{\mathbf{N}}_p dV \frac{\partial \underline{\mathbf{u}}}{\partial t} - \int_{V^e} \underline{\mathbf{B}}_p^T \frac{\underline{\mathbf{K}}}{\gamma_w} \underline{\mathbf{B}}_p dV \underline{\mathbf{p}} + \int_{S^e} \underline{\mathbf{N}}_p^T \underline{\mathbf{q}} dS + \int_{V^e} \underline{\mathbf{B}}_p^T \frac{\underline{\mathbf{K}}}{\gamma_w} \underline{\mathbf{B}}_p \underline{\mathbf{b}}_w dV \right) = 0 \quad (5.45)$$

como $\delta \underline{\mathbf{p}}^T$ assume valores arbitrários, temos:

$$\int_{V^e} \underline{\mathbf{B}}_u^T \underline{\mathbf{m}}^T \underline{\mathbf{N}}_p dV \frac{\partial \underline{\mathbf{u}}}{\partial t} - \int_{V^e} \underline{\mathbf{B}}_p^T \frac{\underline{\mathbf{K}}}{\gamma_w} \underline{\mathbf{B}}_p dV \underline{\mathbf{p}} + \int_{S^e} \underline{\mathbf{N}}_p^T \underline{\mathbf{q}} dS + \int_{V^e} \underline{\mathbf{B}}_p^T \frac{\underline{\mathbf{K}}}{\gamma_w} \underline{\mathbf{B}}_p \underline{\mathbf{b}}_w dV = 0 \quad (5.46)$$

Essa equação expressa a forma fraca da equação que representa a Conservação de Massa de fluido, e pode ser apresentada, de maneira simplificada, como:

$$\underline{\mathbf{l}}^T \frac{\partial \underline{\mathbf{u}}}{\partial t} + \underline{\mathbf{h}} \underline{\mathbf{p}} = \underline{\mathbf{q}} \quad (5.47)$$

onde:

$$\underline{\mathbf{h}} = - \int_{V^e} \underline{\mathbf{B}}_p^T \frac{\underline{\mathbf{K}}}{\gamma_w} \underline{\mathbf{B}}_p dV \quad (5.48)$$

e

$$\underline{\mathbf{q}} = - \int_{S^e} \underline{\mathbf{N}}_p^T \underline{\mathbf{q}} dS + \int_{V^e} \underline{\mathbf{B}}_p^T \frac{\underline{\mathbf{K}}}{\gamma_w} \underline{\mathbf{B}}_p \underline{\mathbf{b}}_w dV \quad (5.49)$$

Sendo assim, as Equação 5.24 e Equação 5.46 são as equações que regem o problema poroelástico. É evidente que essas equações são tomadas para cada elemento em particular. Para a solução do problema em todo o domínio da malha computacional definida é necessário a definição das matrizes globais do problema, utilizando os princípios do MEF, tais matrizes são definidas como:

$$\begin{aligned} \underline{\underline{\mathbf{K}}} &= \sum_{\text{elementos}} \underline{\underline{\mathbf{k}}} = \sum_{\text{elementos } V^e} \int_{V^e} \underline{\mathbf{B}}_u^T \underline{\underline{\mathbf{D}}} \underline{\mathbf{B}}_u dV \\ \underline{\underline{\mathbf{L}}} &= \sum_{\text{elementos}} \underline{\underline{\mathbf{l}}} = \sum_{\text{elementos } V^e} \int_{V^e} \underline{\mathbf{B}}_u \underline{\mathbf{m}} \underline{\mathbf{N}}_p^T dV \\ \underline{\underline{\mathbf{H}}} &= \sum_{\text{elementos}} \underline{\underline{\mathbf{h}}} = - \sum_{\text{elementos } V^e} \int_{V^e} \underline{\mathbf{B}}_p^T \frac{\underline{\mathbf{K}}}{\gamma_w} \underline{\mathbf{B}}_p dV \end{aligned} \quad (5.50)$$

Da mesma forma, os vetores globais do problema são definidos como:

$$\begin{aligned}
\underline{\mathbf{Q}} &= \sum_{\text{elementos}} \underline{\mathbf{q}} = \sum_{\text{elementos}} - \int_{S^e} \underline{\mathbf{N}}_p^T q dS + \int_{V^e} \underline{\mathbf{B}}_p^T \frac{\underline{\mathbf{K}}}{\gamma_w} \underline{\mathbf{B}}_p \underline{\mathbf{b}}_w dV \\
\underline{\mathbf{F}}^{ext} &= \sum_{\text{elementos}} \underline{\mathbf{f}}^{ext} = \sum_{\text{elementos}} \int_{S^e} \underline{\mathbf{N}}_u^T \frac{\partial \underline{\mathbf{t}}}{\partial t} dS - \int_{V^e} \underline{\mathbf{N}}_u^T \frac{\partial \underline{\mathbf{b}}}{\partial t} dV \\
\underline{\mathbf{f}}^{ext} &= \int_{S^e} \underline{\mathbf{N}}_u^T \frac{\partial \underline{\mathbf{t}}}{\partial t} dS - \int_{V^e} \underline{\mathbf{N}}_u^T \frac{\partial \underline{\mathbf{b}}}{\partial t} dV
\end{aligned} \tag{5.51}$$

Sendo assim, está definida a discretização espacial, a partir de uma formulação mista do MEF, para o problema poroelástico que é representado pela equação:

$$\begin{bmatrix} \underline{\mathbf{K}} & \underline{\mathbf{L}} \\ \underline{\mathbf{L}}^T & \underline{\mathbf{0}} \end{bmatrix} \begin{bmatrix} \frac{\partial \underline{\mathbf{U}}}{\partial t} \\ \frac{\partial \underline{\mathbf{P}}}{\partial t} \end{bmatrix} + \begin{bmatrix} \underline{\mathbf{0}} & \underline{\mathbf{0}} \\ \underline{\mathbf{0}} & \underline{\mathbf{H}} \end{bmatrix} \begin{bmatrix} \underline{\mathbf{U}} \\ \underline{\mathbf{P}} \end{bmatrix} = \begin{bmatrix} \underline{\mathbf{F}}^{ext} \\ \underline{\mathbf{Q}} \end{bmatrix} \tag{5.52}$$

onde $\underline{\mathbf{U}}$ e $\underline{\mathbf{P}}$ são, respectivamente, os vetores de incógnitas mecânicas (deslocamentos) e hidráulicas (pressões) do problema.

6 FORMULAÇÃO DO MÉTODO DAS DIFERENÇAS FINITAS PARA A DISCRETIZAÇÃO TEMPORAL DA TEORIA DE BIOT

Para a discretização temporal da equação que rege o problema poroelástico, dada pela Equação 5.50, será utilizado uma formulação do Método das Diferenças Finitas. O Método é conhecido como Método θ (SLOAN; ABBO, 1999.a), onde a calibração do termo θ dita como será realizada a aproximação, seja ela implícita, explícita ou intermediária.

A Equação 5.50 será escrita, de maneira compacta como:

$$\underline{\underline{\mathbf{M}}}_1 \frac{\partial \underline{\underline{\mathbf{X}}}}{\partial t} + \underline{\underline{\mathbf{M}}}_2 \underline{\underline{\mathbf{X}}} + \underline{\underline{\mathbf{F}}}(t) \quad (6.1)$$

Sendo:

$$\underline{\underline{\mathbf{M}}}_1 = \begin{bmatrix} \underline{\underline{\mathbf{K}}} & \underline{\underline{\mathbf{L}}} \\ \underline{\underline{\mathbf{L}}}^T & \underline{\underline{\mathbf{0}}} \end{bmatrix} \quad (6.2)$$

$$\underline{\underline{\mathbf{M}}}_2 = \begin{bmatrix} \underline{\underline{\mathbf{0}}} & \underline{\underline{\mathbf{0}}} \\ \underline{\underline{\mathbf{0}}} & \underline{\underline{\mathbf{H}}} \end{bmatrix} \quad (6.3)$$

O vetor $\underline{\underline{\mathbf{X}}}$ é dado por $\underline{\underline{\mathbf{X}}} = [\underline{\underline{\mathbf{U}}} \quad \underline{\underline{\mathbf{P}}}]^T$ e $\underline{\underline{\mathbf{F}}}(t) = [\underline{\underline{\mathbf{F}}}^{ext} \quad \underline{\underline{\mathbf{Q}}}]^T$

Para dado instante de tempo a forma discreta da Equação 6.1 é dada por

$$\left(\underline{\underline{\mathbf{M}}}_1 + \theta \Delta t \underline{\underline{\mathbf{M}}}_2 \right) \underline{\underline{\mathbf{X}}}^n = \left[\underline{\underline{\mathbf{M}}}_1 - (1-\theta) \Delta t \underline{\underline{\mathbf{M}}}_2 \right] \underline{\underline{\mathbf{X}}}^{n-1} + \Delta t \left[(1-\theta) \underline{\underline{\mathbf{F}}}^{n-1} + \underline{\underline{\mathbf{F}}}^n \right] \quad (6.4)$$

onde θ é o parâmetro que define o tipo de aproximação que será utilizada e pode variar no intervalo $0 \leq \theta \leq 1$ e os sobrescritos n e $n+1$ correspondem aos valores dos termos avaliados nos tempos t_n e t_{n+1} respectivamente, onde $t_{n+1} = t_n + \Delta t$, sendo Δt o intervalo de tempo entre esses dois instantes de tempo consecutivos.

7 ASPECTOS COMPUTACIONAIS

A implementação da formulação numérica descrita ao longo desse trabalho foi feita em ambiente MATLAB®. A linguagem MATLAB® foi escolhida pela sua capacidade e versatilidade no processo de implementação de códigos feitos para operações de cálculo matricial. O código desenvolvido contempla as fases de processamento e pós processamento dos problemas, sendo a parte de pré-processamento realizada no Gmsh®. Para o presente trabalho, é compreendido como pré-processamento a definição da geometria e a geração da malha computacional, retornando ao usuário arquivos de saída em um formato que possa ser lido de forma natural pelo código desenvolvido. No que se trata de entrada de dados de simulação, tais como parâmetros e características do problema como tempo de simulação, valores de parâmetros físicos, tipo de análise, esses são incluídos em uma rotina MATLAB® construída para tal. A visualização de resultados e organização em arquivos também faz parte da rotina em MATLAB® que compõe o código criado.

Para a discretização da variável espacial foi utilizada uma formulação mista do MEF. Tal formulação considera a aproximação para o campo de pressões (problema hidráulico) como sendo linear dentro de cada elemento e bilinear para o campo de velocidade (problema mecânico), onde os elementos possuem geometria triangular. O acoplamento é total, onde em cada passo de tempo uma única matriz é resolvida. A discretização temporal é feita em uma formulação do MDF que será apresentada ao longo do trabalho.

A validação do código é realizada com a comparação dos resultados numéricos com soluções analíticas encontradas na literatura. O primeiro problema consiste no problema de adensamento unidimensional e isotrópico. No segundo será considerado o adensamento unidimensional com um meio heterogêneo, dividido em duas camadas. O terceiro problema é o problema conhecido como problema de Mandel. No quarto problema é realizada uma comparação com o problema de adensamento que simula o adensamento de uma camada de solo gerada por o carregamento de uma sapata. Também é avaliado um problema com a aplicação de cargas, por uma sapata, em uma massa de solo dividida em três camadas com permeabilidades anisotrópicas.

O esquema abaixo é um resumo das atividades que são desenvolvidas em cada etapa do processo de simulação de um problema:

- PRÉ-PROCESSAMENTO

- Gmsh®

- ✓ Construção da geometria do modelo;
- ✓ Geração da malha computacional;
- ✓ Obtenção dos arquivos .geo e .msh para leitura pelo código;

- MATLAB® (Código desenvolvido)

- ✓ Leitura do arquivo .geo e .msh gerado no Gmsh®;
- ✓ Entrada de parâmetros da simulação
 - Nome do arquivo .geo;
 - Nome do arquivo .msh;
 - Tipo de simulação: mecânica, hidráulica, acoplada;
 - Valor de θ ;
 - Número máximo de iterações;
 - Valor de Δt ;
 - Iterações que serão pós-processadas;
- ✓ Entrada de parâmetros relativos ao pós-processamento
 - Fator de escala;
 - Definição das variáveis que serão avaliadas;
 - Valores relativos às avaliações em seções definidas.
- ✓ Definição dos parâmetros físicos
 - Módulo de elasticidade;
 - Coeficiente de Poisson;
 - Permeabilidades;
 - Peso específico do fluido.
- ✓ Definição das condições de contorno do problema
- ✓ Transformação dos valores obtidos nos arquivos .geo e .msh em matrizes capazes de serem operadas ao longo do processo.

- PROCESSAMENTO

- MATLAB®

- ✓ Conjunto de rotinas que calculam, de acordo com a formulação numérica descrita nesse trabalho, as variáveis do problema poroelástico avaliado.

- PÓS-PROCESSAMENTO

- MATLAB®

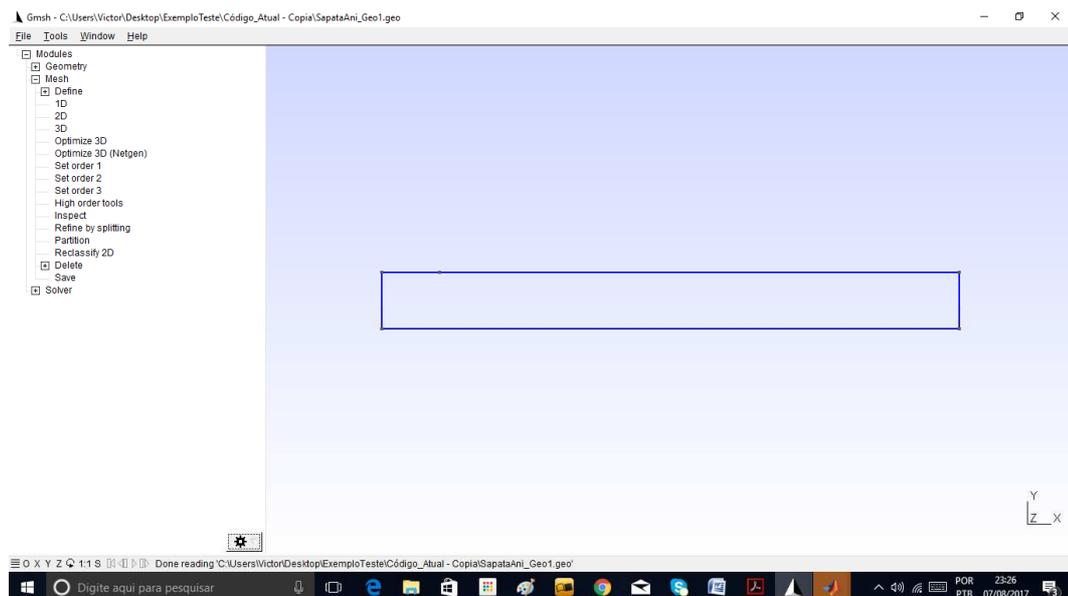
- ✓ Rotinas criadas com as saídas definidas no pré-processamento, analisadas nos instantes de tempo definidos, que podem ser
 - Campo de tensões em todo o domínio apresentado em escala de cores;
 - Campo de pressões em todo o domínio apresentado em escala de cores;
 - Tensões ao longo do eixo y presentes em um intervalo definido de x ;
 - Pressões ao longo do eixo y presentes em um intervalo definido de x ;
 - Deslocamentos ao longo do eixo y presentes em um intervalo definido de x ;
 - Obs: as Tensões, Pressões e Deslocamentos também podem ser processadas, de forma análoga, ao longo de um eixo na direção x ;
 - Malha deformada;
 - Valores máximos e mínimos das variáveis avaliadas.
 - ✓ Criação de diretórios com as saídas de forma automatizada.

A sequência de figuras da Figura 7.1 à Figura 7.8 apresenta o procedimento realizado para a simulação de um determinado problema utilizando o código desenvolvido. Basicamente todos os problemas seguem a seguinte sequência:

- A geometria do problema é criada no Gmsh® (Figura 7.1);
- São definidas as configurações sobre a malha que também é gerada no Gmsh® (Figura 7.2);
- O Gmsh® fornece o arquivo de geometria, com formato `.geo` e um arquivo de malha com formato `.msh`, transformando a informação visual em informações que possam ser manipuladas pelo código (Figura 7.3);
- Após esse procedimento os arquivos `.geo` e `.msh` são salvos na pasta que contém os códigos do programa;

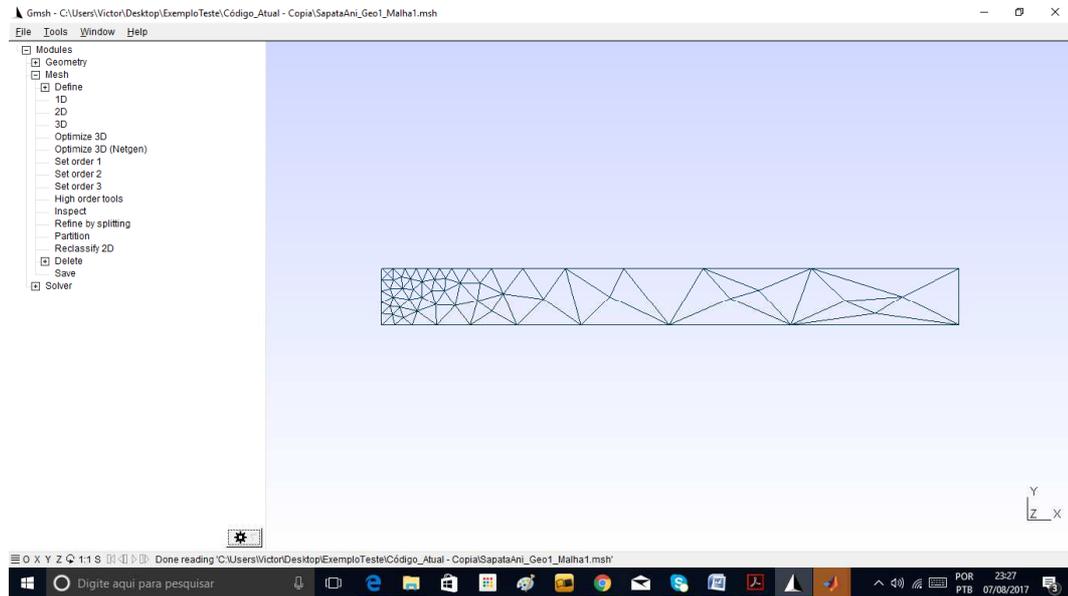
- Os arquivos .msh e .geo são definidos em uma rotina do MATLAB® denominada ‘Main_MefM_Biot.m’ (Figura 7.4);
- As entradas dos parâmetros de simulação são definidas em ‘Main_MefM_Biot.m’ (Figura 7.5);
- A função ‘Main_MefM_Biot.m’ é processada (Figura 7.6);
- Ao término do processamento são criados diretórios com os resultados obtidos com um arquivo contendo alguns valores sobre as variáveis e documentos de imagem (Figura 7.7);
- Os resultados obtidos possuem a formatação apresentada na Figura 7.8.

Figura 7.1. Procedimento de simulação - criação da geometria do Gmsh.



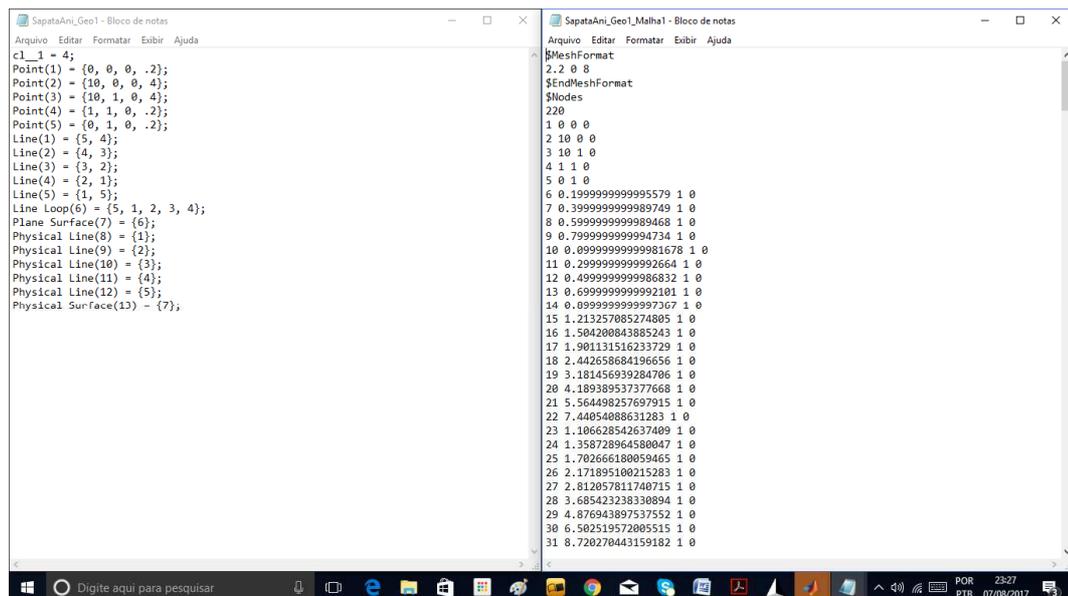
Fonte: Elaborada pelo autor.

Figura 7.2. Procedimento de simulação - geração da malha computacional no Gmsh.



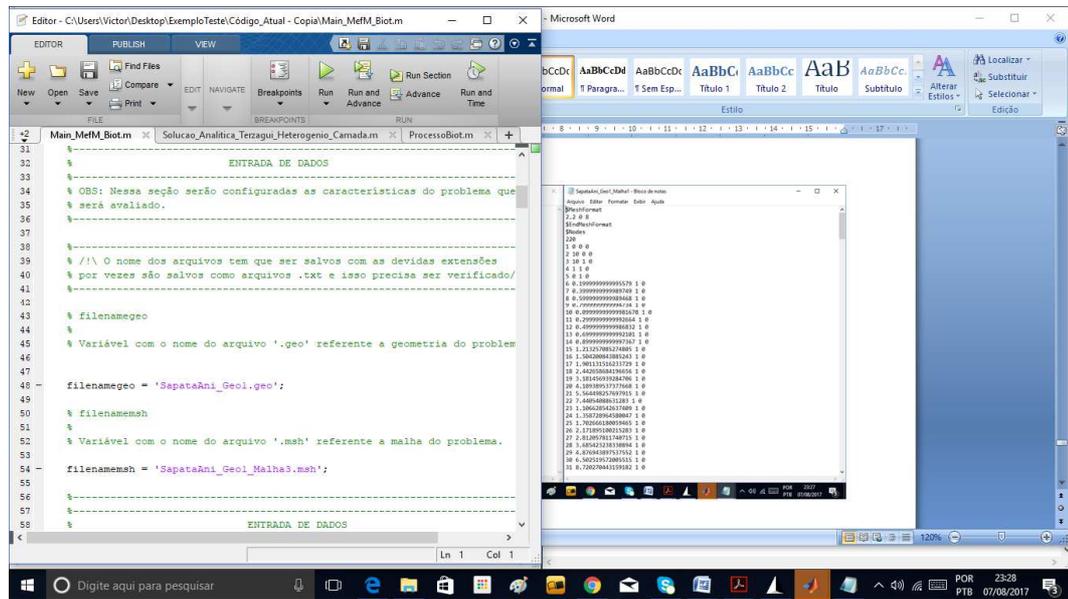
Fonte: Elaborada pelo autor.

Figura 7.3. Procedimento de simulação - arquivo .geo e .msh obtidos no Gmsh.



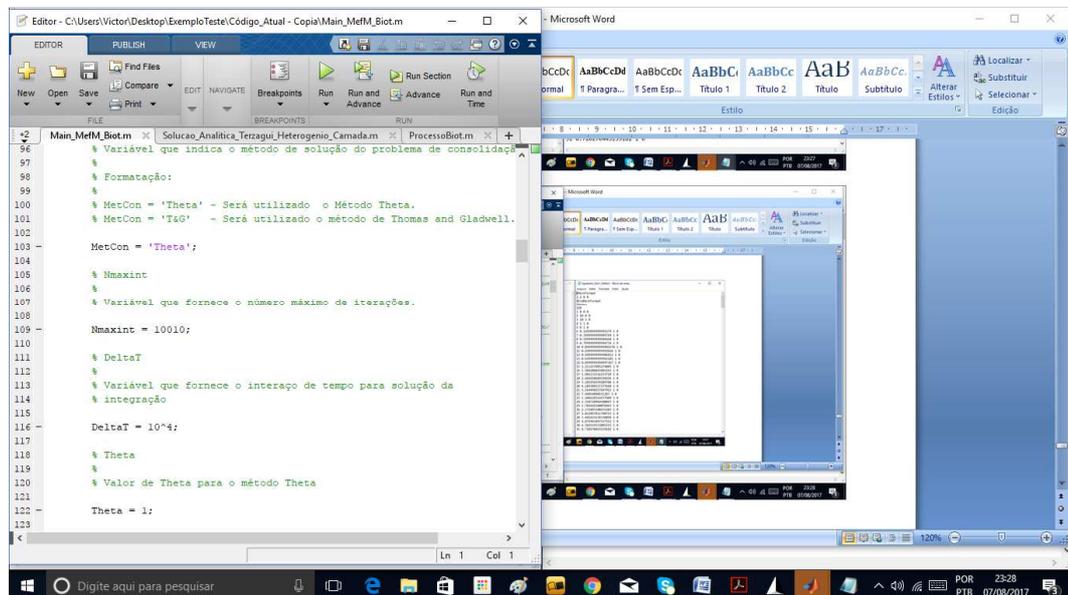
Fonte: Elaborada pelo autor.

Figura 7.4. Procedimento de simulação - definição, no MATLAB, dos arquivos que serão lidos.



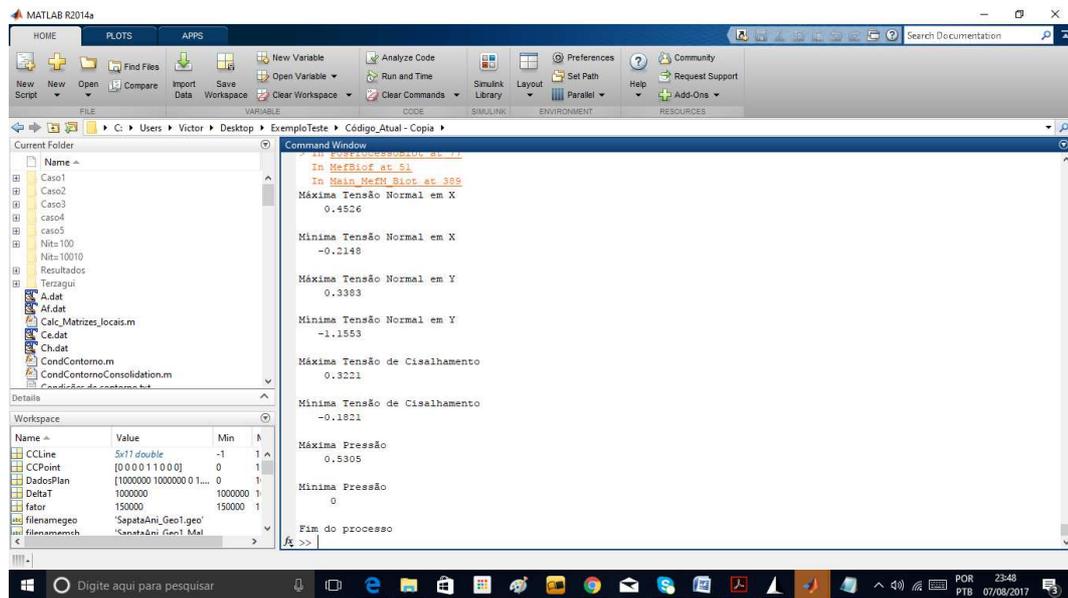
Fonte: Elaborada pelo autor.

Figura 7.5. Procedimento de simulação - definição dos parâmetros para a simulação na rotina do MATLAB.



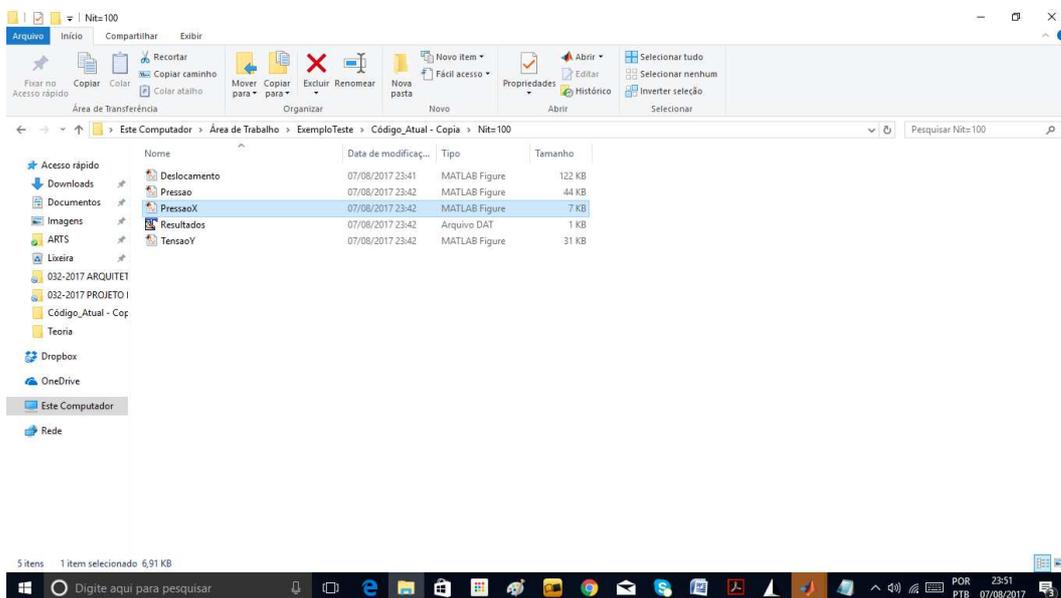
Fonte: Elaborada pelo autor.

Figura 7.6. Procedimento de simulação - processamento do MATLAB.



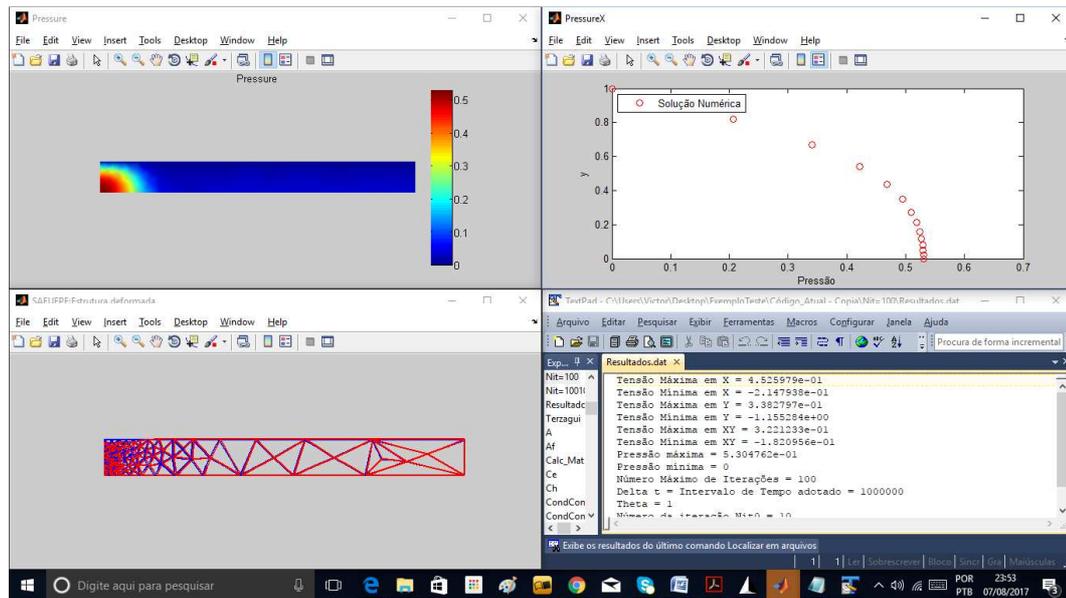
Fonte: Elaborada pelo autor.

Figura 7.7. Procedimento de simulação - criação de diretórios com os resultados.



Fonte: Elaborada pelo autor.

Figura 7.8. Procedimento de simulação - resultados salvos.



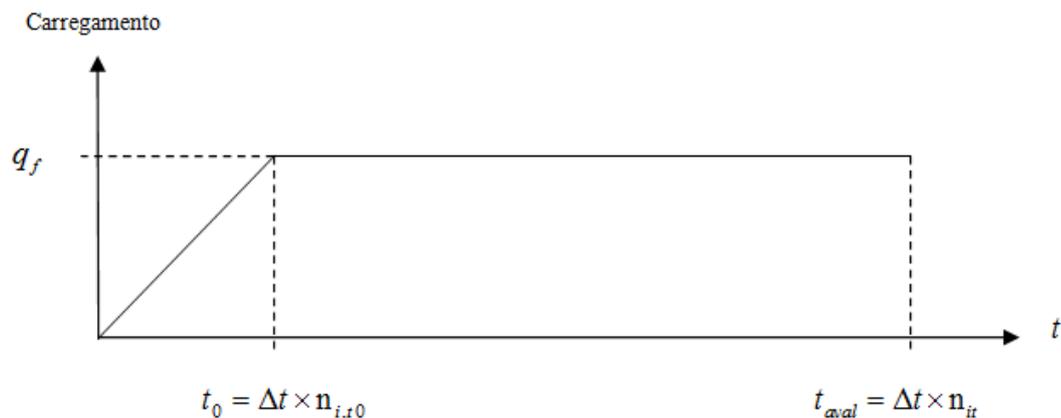
Fonte: Elaborada pelo autor.

No APÊNDICE A são apresentados os detalhes de cada função específica, um modelo do código e os códigos propriamente ditos.

8 ANÁLISES REALIZADAS

Essa seção é dedicada à apresentação dos resultados obtidos com o código computacional desenvolvido. São avaliados cinco problemas, todos eles apresentando soluções analíticas encontradas na literatura. Três deles são clássicos, homogêneos e isotrópicos, sendo esse o Problema de Terzaghi, o Problema de Mandel e o Problema de Schiffman. Além desses são apresentados o Problema de Terzaghi modificado para um meio heterogêneo e um adensamento, que se desenvolve nas duas direções, que contempla anisotropia e heterogeneidade. O objetivo de tais comparação é a validação do código desenvolvido e a comparações entre os resultados obtidos intrínsecos a cada tipo de solução das equações diferenciais, tais como soluções analíticas, numéricas e contendo condições de simplificação ou generalizações sejam das propriedades, geometrias ou qualquer outro aspecto. As tensões avaliadas são as tensões efetivas.

Figura 8.1. Carreamento em função do tempo.



Fonte: (SLOAN; ABBO, 1999.b).

Para os problemas avaliados temos que, a partir do instante $t = 0,00$, o carregamento é aplicado, de forma linear, até que seja obtido seu valor final, q_f , no instante t_0 , como apresentado na Figura 8.1. Os resultados são avaliados no tempo t_{aval} , e temos que $t_0 = \Delta t \times n_{i,t_0}$ e $t_{aval} = \Delta t \times n_{it}$, onde Δt é o passo de tempo considerado, n_{it} é o número de iterações avaliadas para o tempo final e n_{i,t_0} é o número de iterações avaliadas para o tempo t_0 . Temos esse fato devido à formulação do problema que é apresentada em termos de derivadas, sendo necessária a existência da parte variável do carregamento.

8.1 ADENSAMENTO UNIDIMENSIONAL DE MEIO HOMOGÊNIO E ISOTRÓPICO – PROBLEMA DE TERZAGHI

O problema de Terzaghi, como é conhecido, é a base para a Teoria do Adensamento unidimensional. É rotineiramente utilizado na validação de códigos computacionais. Apesar das simplificações impostas aos modelos e à própria solução analítica desenvolvida por Terzaghi, este é largamente utilizado em escritórios de engenharia geotécnica e de grupos de reservatórios de petróleo, quando há a necessidade de prever o comportamento de adensamento de solos como medição de recalques causados por fundações ou a subsidência gerada pelas operações do reservatório. É evidente que suas simplificações só contemplam situações específicas de adensamento desses meios, porém na engenharia, por muitas vezes, a ordem de grandeza da variável analisada é suficiente para a tomada de decisões importantes.

Em se tratando do problema propriamente dito, ele consiste na aplicação de um carregamento sobre uma coluna poroelástica homogênea e isotrópica, totalmente saturada por um fluido monofásico. A coluna possui capacidade drenante no seu topo e se mantém impermeável em sua base, que permanece fixa, ou seja, com deslocamentos nulos. Devido ao seu caráter ele se desenvolve apenas ao longo da direção y .

A Figura 8.2 (a) apresenta a geometria utilizada para a modelagem desse problema e a Figura 8.2 (b) a definição do seu domínio e contornos.

As condições iniciais e de contorno para esse problema são dadas abaixo.

Condições iniciais ($t = 0,00s$)

$$\Gamma_1, \Gamma_2, \Gamma_3, \Gamma_4 \text{ e } \Omega \rightarrow \begin{cases} u_x = u_y = 0,00MPa \\ p = 0,00MPa \end{cases}$$

Condições de Contorno ($0,00s < t \leq 2,25s$)

$$\Gamma_1 \rightarrow \begin{cases} \sigma_x = 0,00MPa \\ \frac{\partial \sigma_y}{\partial t} = \frac{-1.000,00MPa}{2,25s} \\ p = 0,00MPa \end{cases}$$

$$\Gamma_2, \Gamma_4 \rightarrow \begin{cases} u_x = 0,00m \\ q = 0,00m/s \end{cases}$$

$$\Gamma_3 \rightarrow \begin{cases} u_y = 0,00m \\ q = 0,00m/s \end{cases}$$

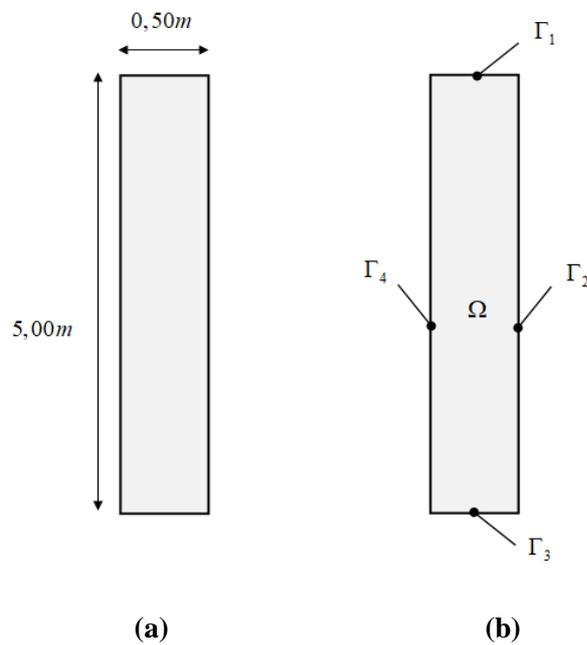
Condições de Contorno ($t > 2,25s$)

$$\Gamma_1 \rightarrow \begin{cases} \sigma_x = 0,00MPa \\ \sigma_y = -1.000,00MPa \\ p = 0,00MPa \end{cases}$$

$$\Gamma_2, \Gamma_4 \rightarrow \begin{cases} u_x = 0,00m \\ q = 0,00m/s \end{cases}$$

$$\Gamma_3 \rightarrow \begin{cases} u_y = 0,00m \\ q = 0,00m/s \end{cases}$$

Figura 8.2. Problema de Terzaghi - Geometria e condições de contorno.



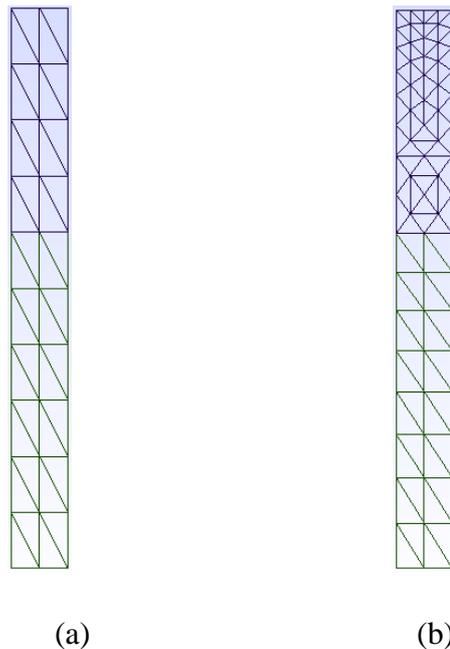
Fonte: Elaborada pelo autor.

Os parâmetros físicos utilizados na simulação são:

- $E = 1,00 \times 10^7 MPa$
- $\nu = 0,20$
- $\mathbf{\kappa} = \begin{bmatrix} 10^{-6} & 0 \\ 0 & 10^{-6} \end{bmatrix} cm/s$
- $\gamma_w = 1000,00 kgf/m^3$

Foram propostas duas malhas computacionais tendo em vista a observância de resultados em trabalhos da mesma natureza e pelos próprios resultados obtidos ao longo do trabalho. As duas malhas utilizadas são apresentadas na Figura 8.3(a) e Figura 8.3(b) e serão denominadas respectivamente e Malha1 e Malha2.

Figura 8.3. Problema de Terzaghi: (a) Malha 1; (b) Malha 2.



Fonte: Elaborada pelo autor.

Para as análises foram definidos os fatores tempos, valores usuais utilizados na poroelasticidade, que são definidos de acordo com as características físicas e geométricas de cada caso, com o intuito de adimensionalizar as simulações facilitando analogias entre situações de magnitudes e características diferentes, ou seja, o fator tempo correlaciona o tempo de adensamento com as propriedades do meio poroso. O fator tempo também é uma medida que fornece informações sobre o estágio de adensamento em que o problema se encontra e é definido como:

$$T_v = \frac{tc_v}{H^2} \quad (8.1)$$

onde H é o comprimento de drenagem da coluna e c_v o coeficiente de adensamento, dado por:

$$c_v = \frac{k_y E (1 - \nu)}{\gamma_w (1 + \nu) (1 - 2\nu)} \quad (8.2)$$

Definido o fator tempo, os parâmetros da simulação estão apresentados abaixo:

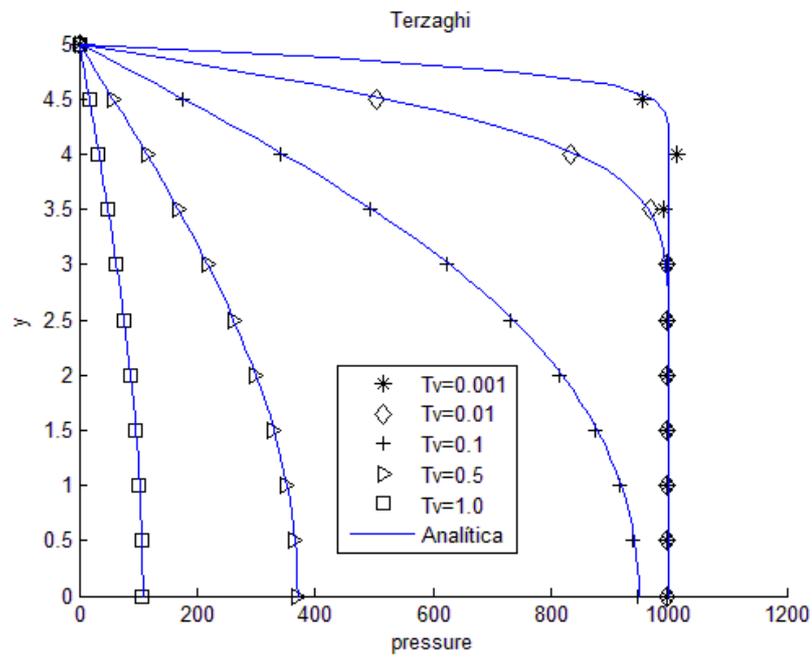
- Fatores tempo avaliados: $T_v = 0,001$; $T_v = 0,01$; $T_v = 0,1$; $T_v = 0,50$ e $T_v = 1,00$
- Fator tempo correspondente ao t_0 : $T_{v0} = 0,001$
- Número de iterações correspondentes ao t_0 : 225,00
- Passo de tempo da simulação: $\Delta t = 0,01s$
- Número máximo de iterações: 225.225,00

A Figura 8.4 e a Figura 8.5 apresentam os resultados das pressões ao longo do eixo y para a reta $x = 0,00$. Essas Figuras apresentam os resultados respectivos à Malha1 e à Malha2.

Como pode ser observado na Figura 8.4 e como já era esperado, na região correspondente ao $T_v = 0,001$ onde há uma variação brusca no gradiente de pressões, a solução numérica, naturalmente, oscila, tomando valores maiores do que os obtidos na solução analítica. Esse fato é decorrente de primordialmente dois aspectos. O primeiro é que as funções interpoladoras elementares para as variáveis hidráulicas são lineares, e é natural que haja esse comportamento quando se utiliza esse tipo de função para situações com modificações bruscas nos gradientes das variáveis analisadas. O outro aspecto que contribui para esse resultado é o refinamento da malha computacional, que influencia na captura desses valores. Dessa forma, refinando a malha computacional na região superior do domínio o resultado apresentou melhoras significativas, como pode ser observado na Figura 8.5.

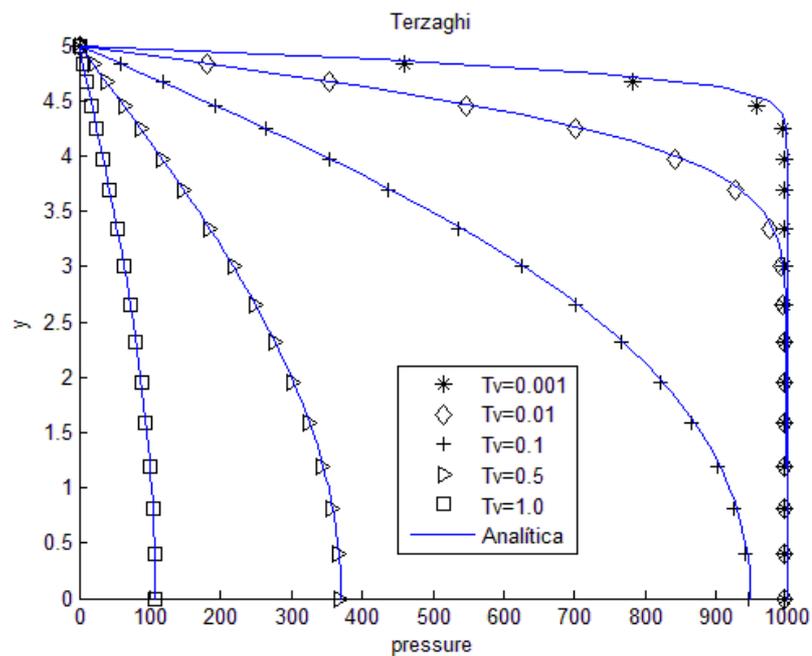
Observa-se que o refinamento da malha foi realizado apenas nas regiões onde já se esperava a manifestação de um determinado evento. Esse tipo de sensibilidade por parte do analista numérico que trata com simulações é de total importância, tendo em vista que o refinamento da malha computacional implica no aumento significativo do custo computacional e até mesmo o aumento de número de graus de liberdade para serem analisados. Caso não haja, principalmente em problemas mais complexos, a identificação preliminar dessas zonas é importante que se construam modelos básicos, quando viável, para que haja esse tipo de informação. A análise com a Malha1 e com a Malha2 poderia ser um exemplo desse tipo de avaliação.

Figura 8.4. Problema de Terzaghi - Solução analítica e numérica das pressões ao longo do eixo $x=0$ para a Malha 1.



Fonte: Elaborada pelo autor.

Figura 8.5. Problema de Terzaghi - Solução analítica e numérica das pressões ao longo do eixo $x=0$ para a Malha 2.

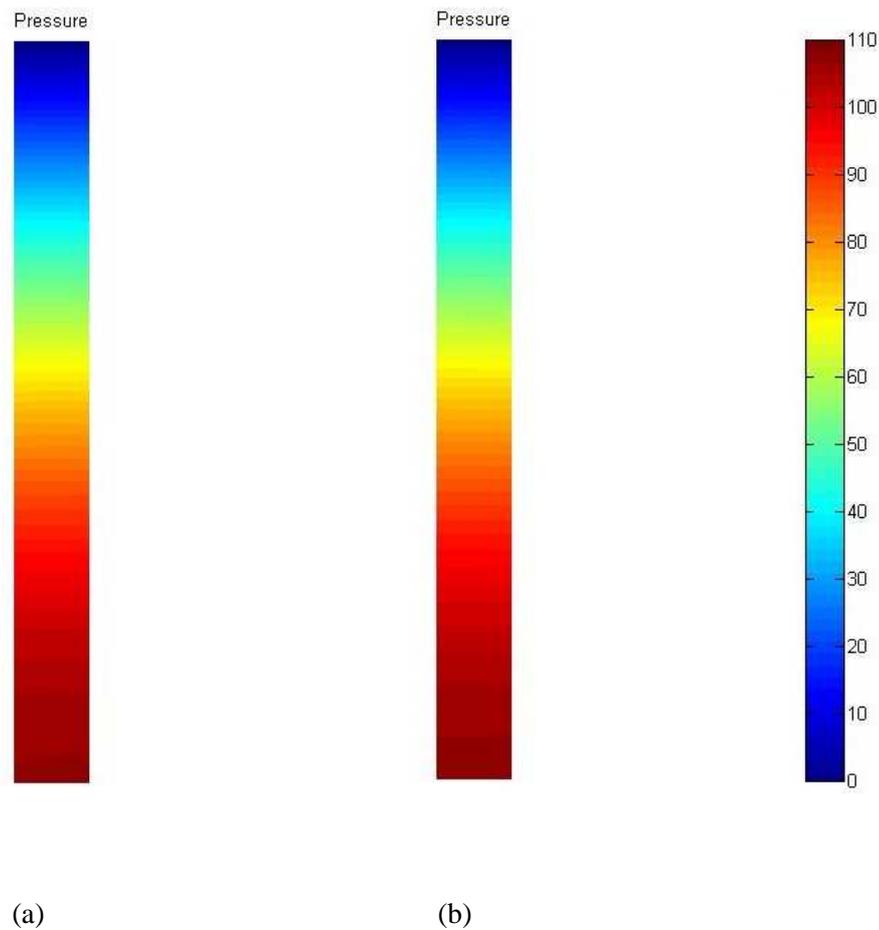


Fonte: Elaborada pelo autor.

Serão apresentados os resultados em escala de cores correspondentes ao $T_v = 1,00$ para as duas malhas computacionais. Os resultados para as pressões são apresentados na Figura 8.6(a) e Figura 8.6(b) para cada malha respectiva. Os resultados para as tensões são apresentados na Figura 8.7 e na Figura 8.8. Como pode ser observado, as tensões normais correspondentes às duas malhas possuem concordância. Devido às características do problema as tensões cisalhantes apresentam valores nulos.

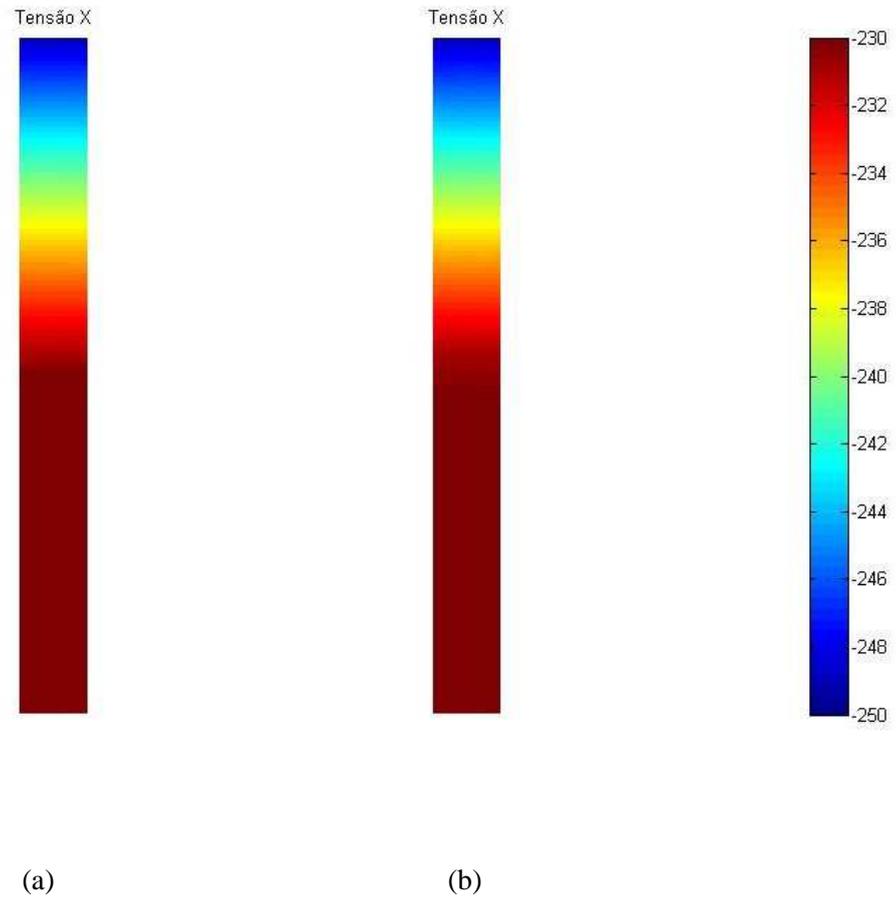
De acordo com os resultados, o código se mostrou satisfatório para a solução desse problema. As peculiaridades apresentadas que divergem da solução analítica já eram esperadas e foram obtidas dentro da normalidade.

Figura 8.6. Problema de Terzaghi - Campo de pressões, (a) Malha1, (b) Malha2.



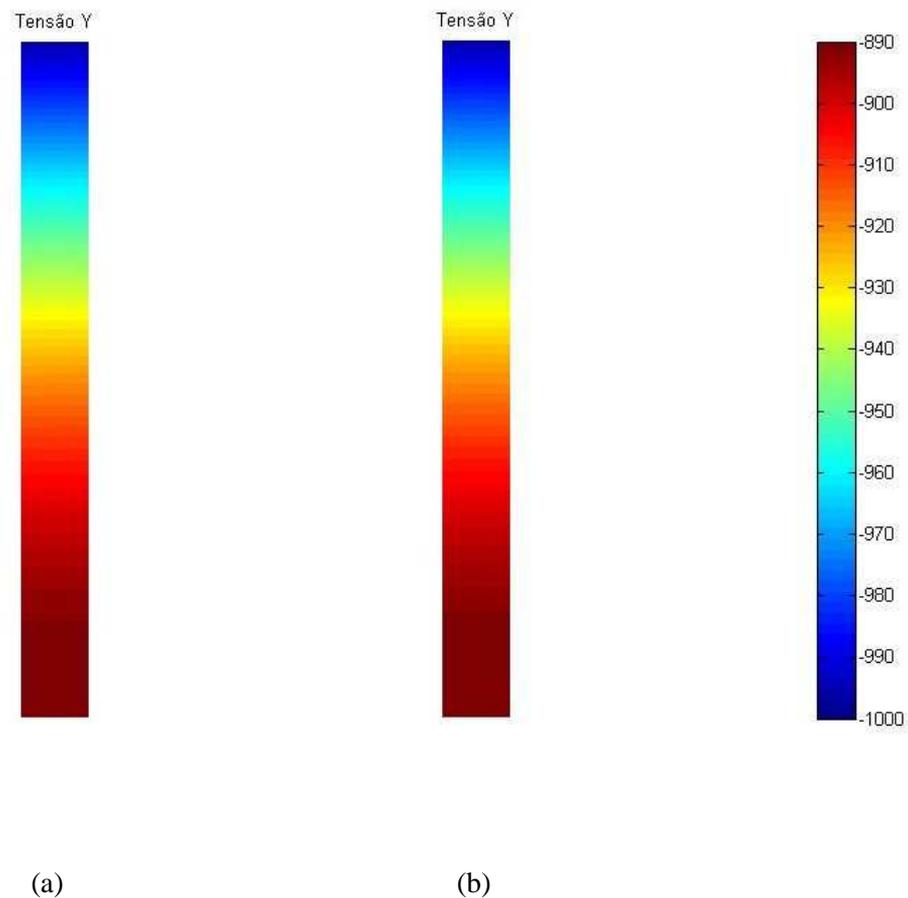
Fonte: Elaborada pelo autor.

Figura 8.7. Problema de Terzaghi - Campo de tensões normais na direção x, (a) Malha1, (b) Malha2.



Fonte: Elaborada pelo autor.

Figura 8.8. Problema de Terzaghi - Campo de tensões normais na direção y, (a) Malha1, (b) Malha2.



Fonte: Elaborada pelo autor.

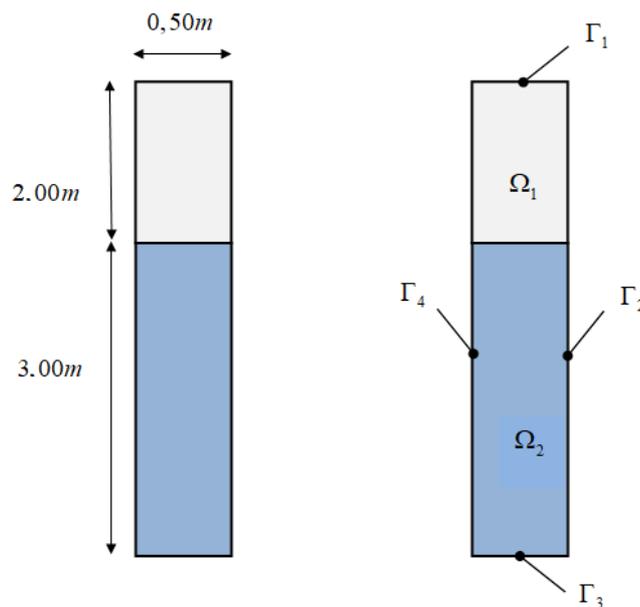
8.2 ADENSAMENTO UNIDIMENSIONAL DE MEIO HETEROGÊNIO E ISOTRÓPICO – PROBLEMA DE TERZAGHI MODIFICADO

Esse problema consiste em uma generalização do de Terzaghi para meios formados por dois tipos de materiais poroelástico distribuídos em duas camadas verticais. A sua solução analítica foi proposta por (CARSLAW; JAEGER., 1948) Dessa forma a geometria também consiste em uma coluna poroelástica, porém heterogênea, dividida em dois materiais, onde o

topo é drenante e submetido a um carregamento que se mantém constante ao longo do processo. Sua base é impermeável e com deslocamentos restritos. As condições de contorno nas laterais são impostas de forma que seja considerada a simetria existente, que se desenvolve apenas em uma direção. Sendo assim são impostas, nas laterais, uma condição de impermeabilidade e os deslocamentos na direção horizontal são nulos. De forma geral a diferença entre o Problema de Terzaghi é apenas a heterogeneidade.

Para a análise no presente caso a geometria e condições iniciais e de contorno são apresentadas na Figura 8.9.

Figura 8.9. Problema de Terzaghi Modificado – Geometria e Condições de Contorno.



Fonte: Elaborada pelo autor.

As condições iniciais e de contorno para esse problema são dadas abaixo:

Condições iniciais ($t = 0,00s$)

$$\Gamma_1, \Gamma_2, \Gamma_3, \Gamma_4 \text{ e } \Omega \rightarrow \begin{cases} u_x = u_y = 0,00MPa \\ p = 0,00MPa \end{cases}$$

Condições de Contorno ($0,00s < t \leq 2,25s$)

$$\Gamma_1 \rightarrow \begin{cases} \sigma_x = 0,00MPa \\ \frac{\partial \sigma_y}{\partial t} = \frac{-1.000,00MPa}{2,25s} \\ p = 0,00MPa \end{cases}$$

$$\Gamma_2, \Gamma_4 \rightarrow \begin{cases} u_x = 0,00m \\ q = 0,00m/s \end{cases}$$

$$\Gamma_3 \rightarrow \begin{cases} u_y = 0,00m \\ q = 0,00m/s \end{cases}$$

Condições de Contorno ($t > 2,25s$)

$$\Gamma_1 \rightarrow \begin{cases} \sigma_x = 0,00MPa \\ \sigma_y = -1.000,00MPa \\ p = 0,00MPa \end{cases}$$

$$\Gamma_2, \Gamma_4 \rightarrow \begin{cases} u_x = 0,00m \\ q = 0,00m/s \end{cases}$$

$$\Gamma_3 \rightarrow \begin{cases} u_y = 0,00m \\ q = 0,00m/s \end{cases}$$

Os parâmetros físicos do problema são definidos abaixo:

- $E = 1,00 \times 10^7 MPa$
- $\nu = 0,20$
- $\underline{\underline{\kappa_1}} = \begin{bmatrix} 3 \times 10^{-6} & 0 \\ 0 & 3 \times 10^{-6} \end{bmatrix} cm/s$ e $\underline{\underline{\kappa_2}} = \begin{bmatrix} 10^{-6} & 0 \\ 0 & 10^{-6} \end{bmatrix} cm/s$
- $\gamma_w = 1000,00 kgf/m^3$

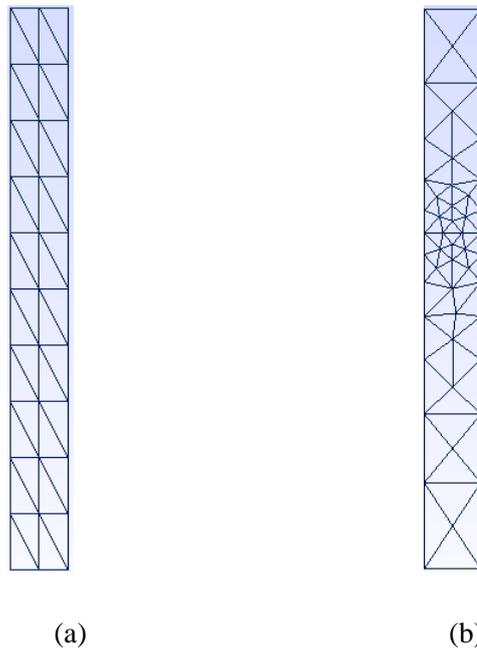
A avaliação foi realizada utilizando duas malhas computacionais distintas. A primeira, denominada de Malha1 está apresentada na Figura 8.10 (a) e a Malha2 na Figura 8.10 (b). A Malha2 foi proposta para avaliar a influência da discretização na região de descontinuidade.

O fator tempo para esse problema será dado da mesma forma do fator tempo para o problema de Terzaghi, porém os parâmetros do meio serão os do domínio Ω_1 . Sendo assim os parâmetros de simulação são:

- Fatores tempo avaliados: $T_v = 0,001$; $T_v = 0,01$; $T_v = 0,1$; $T_v = 0,50$ e $T_v = 1,00$
- Fator tempo correspondente ao t_0 : $T_{v0} = 0,001$

- Número de iterações correspondentes ao t_0 : 225,00
- Passo de tempo da simulação: $\Delta t = 0,01s$
- Número máximo de iterações: 225.225,00

Figura 8.10. Problema de Terzaghi Modificado – Malhas computacionais utilizadas: (a) Malha1; (b) Malha2.

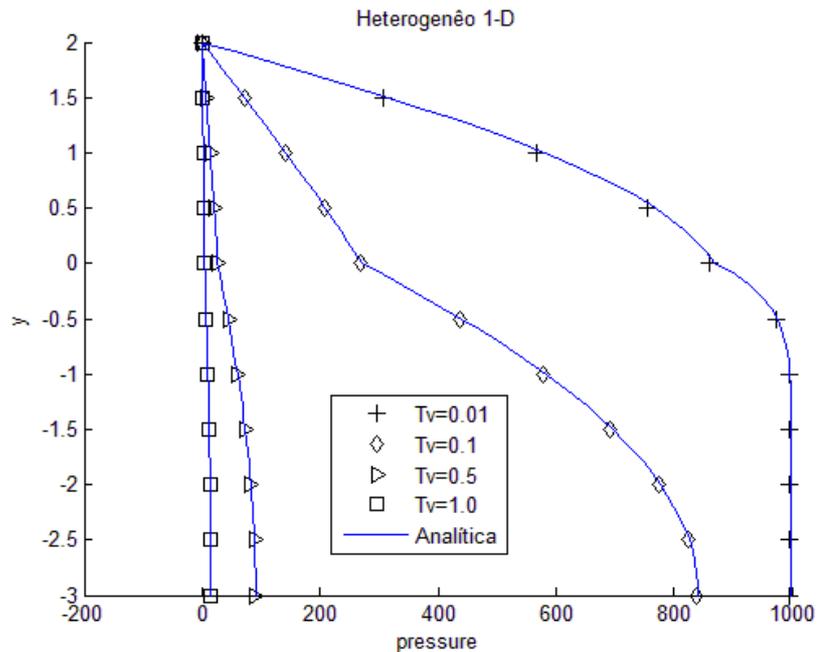


Fonte: Elaborada pelo autor.

O presente caso é uma variação do problema de Terzaghi, sendo assim será avaliado o adensamento do meio para os mesmos instantes de tempo, considerando a camada superior três vezes mais permeável que a inferior, sendo possível, com a observância dos resultados de ambos, verificar a influência da mudança na permeabilidade no meio. Como pode ser observado na Figura 8.11, que apresenta o resultado para o problema obtido com a Malha1, os valores de pressões sofrem uma redução, em particular na camada superior do domínio. Esse fato é esperado, como pode ser visto pela própria solução analítica do problema, e explicado devido à maior facilidade drenante da camada, que possui valor maior de permeabilidade. Quando se observa o $T_v = 1,00$ para os dois casos é possível verificar que para o problema de Terzaghi Modificado as pressões são praticamente dissipadas integralmente, ao contrário do observado no problema de Terzaghi.

A Figura 8.12 (b) apresenta o resultado das pressões para a Malha2. Como pode ser observado há uma concentração no resultado das variáveis onde há a interface entre as camadas.

Figura 8.11. Problema de Terzaghi Modificado – soluções analíticas e numéricas para a Malha1.

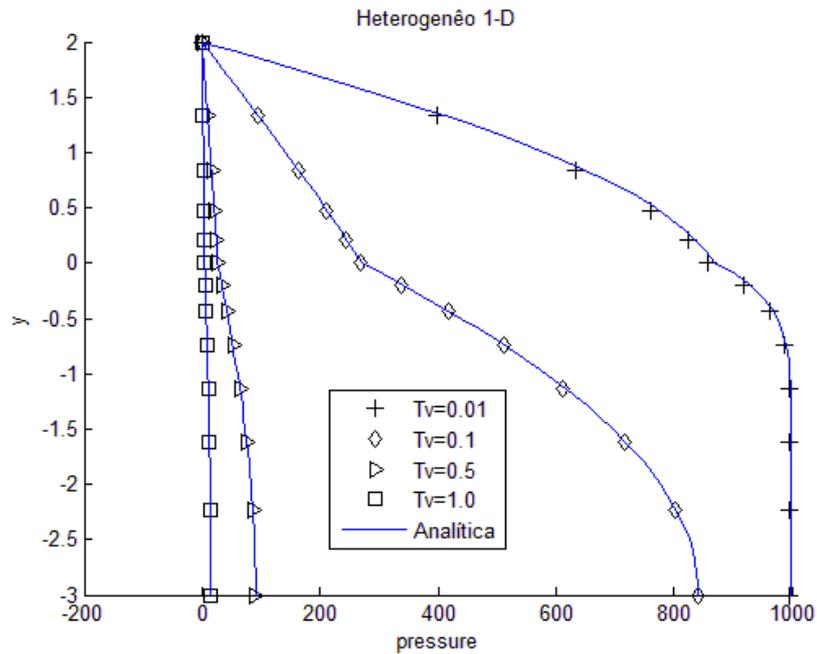


Fonte: Elaborada pelo autor.

Como podem ser observados, de acordo com os resultados na Figura 8.11 e Figura 8.12, os dois resultados foram compatíveis, de maneira satisfatória, com a solução analítica. Para o problema em particular a diferença de malhas computacionais não gerou modificações consideráveis na distribuição de pressões, tendo as duas valores condizentes com a solução analítica. Sendo assim, é possível verificar que a heterogeneidade aplicada no domínio não gerou uma diferença de gradientes de pressões nas interfaces capaz de gerar oscilações no resultado. Tal fato mostra que, dependendo do grau de heterogeneidade não há a necessidade de um refino da malha computacional na região de interface.

A diferença entre os valores das pressões obtidas nesse exemplo e no exemplo de Terzaghi são explicados pelas diferenças nos valores de permeabilidades dos meios. O fato das soluções para o problema heterogêneo apresentarem pressões menores que o de Terzaghi é um reflexo do fato da drenagem ser mais fácil na camada superior quando a permeabilidade é maior, possibilitando assim a dissipação mais rápida das pressões.

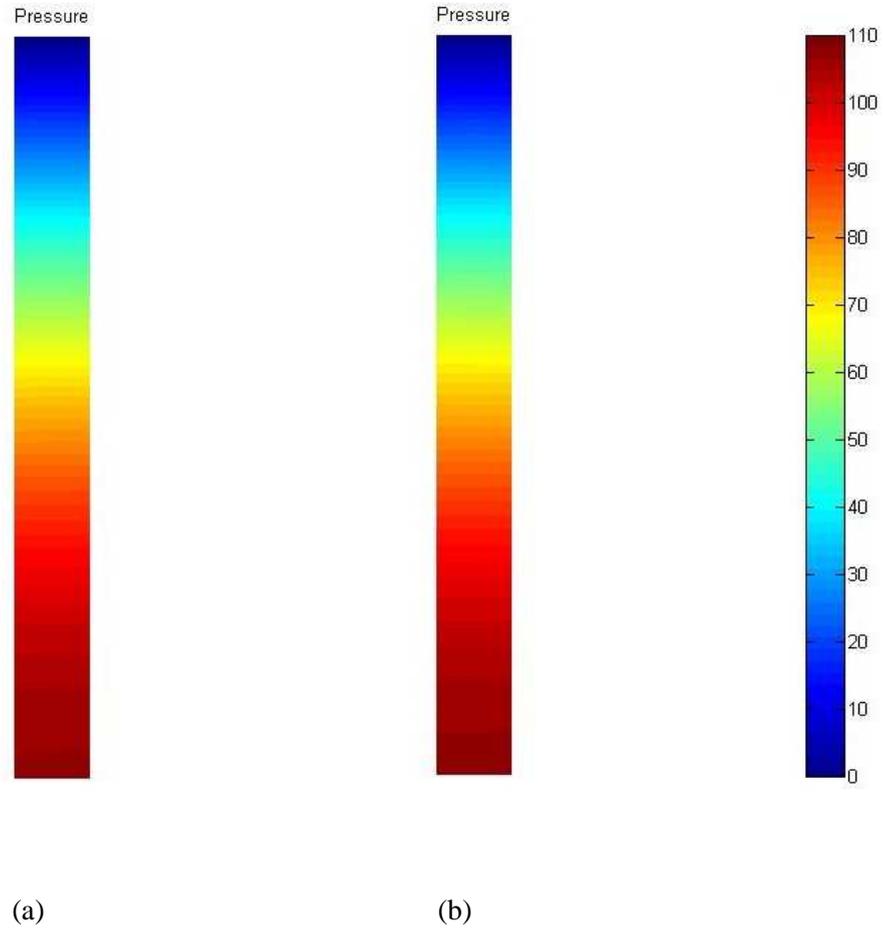
Figura 8.12. Problema de Terzaghi Modificado – soluções analíticas e numéricas para a Malha2.



Fonte: Elaborada pelo autor.

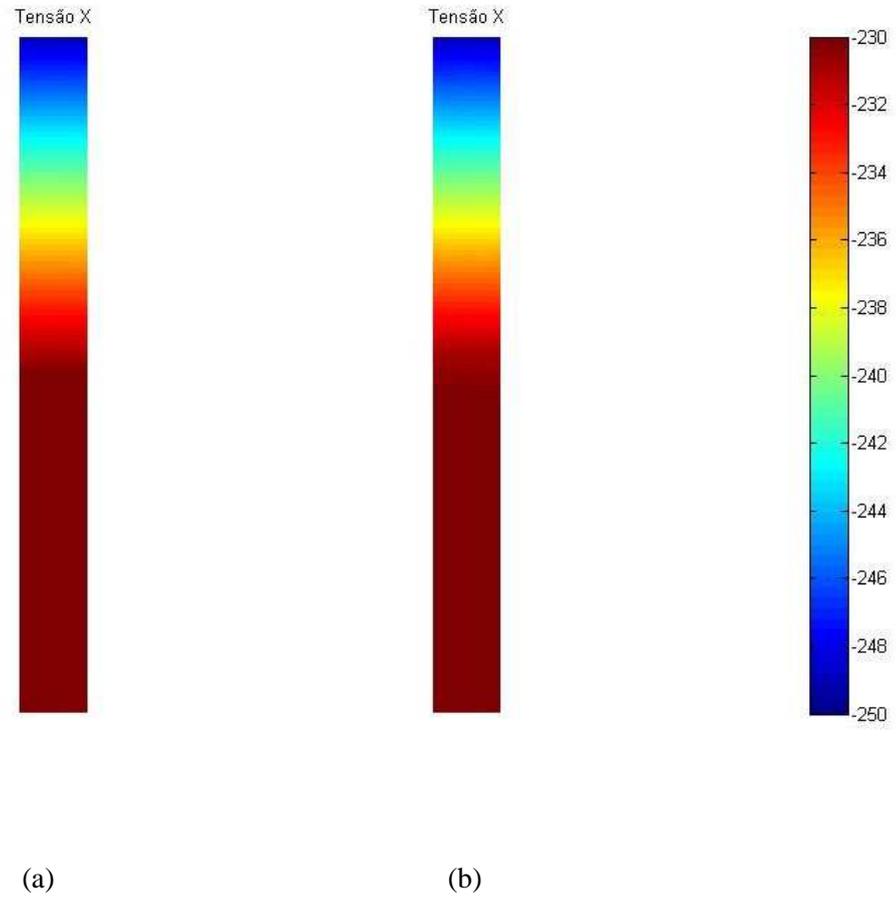
O conjunto de figuras que vai da Figura 8.13 à Figura 8.15 apresentam os resultados de pressões e tensões em escala de cores para as duas malhas computacionais analisados em um fator tempo $T_v = 0,1$. Observados esses resultados é possível a verificação da concordância entre as duas simulações. Em relação às tensões cisalhantes são válidas as mesmas observações acerca do Problema de Terzaghi. A partir dos resultados foi possível observar uma capacidade do código em solucionar problemas com o tipo de heterogeneidade avaliada nesse problema.

Figura 8.13. Problema de Terzaghi modificado - Campo de pressões, (a) Malha1, (b) Malha2.



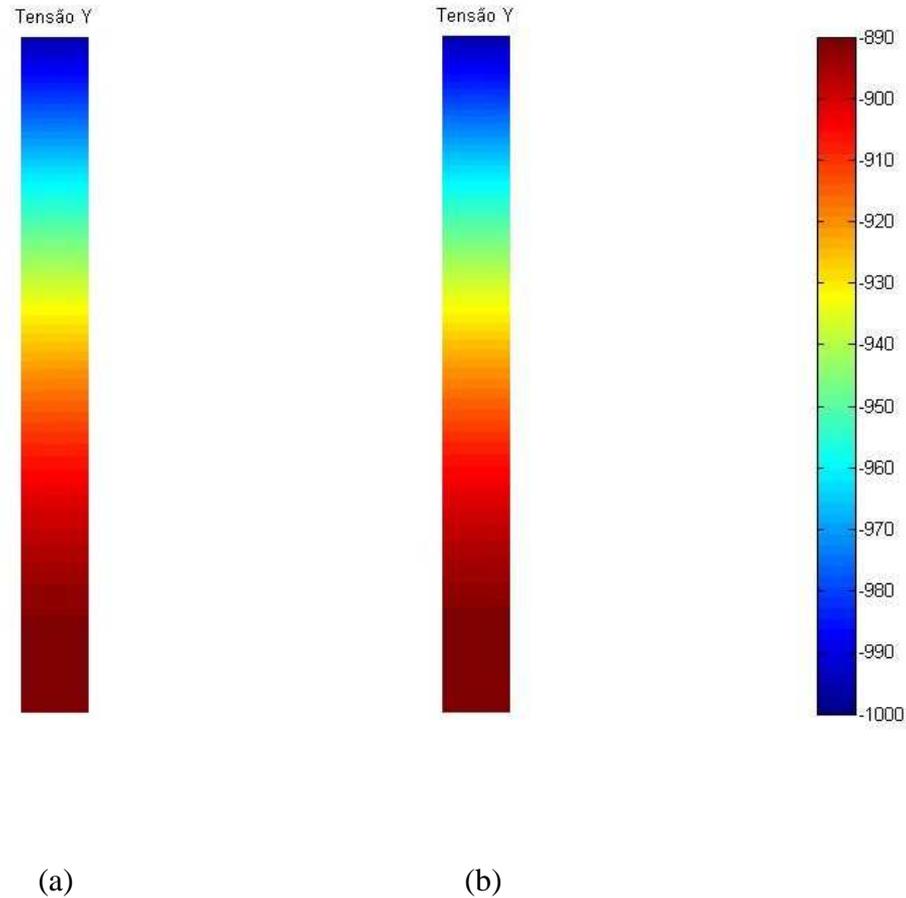
Fonte: Elaborada pelo autor.

Figura 8.14. Problema de Terzaghi modificado - Campo de tensões normais na direção x, (a) Malha1, (b) Malha2.



Fonte: Elaborada pelo autor.

Figura 8.15. Problema de Terzaghi modificado - Campo de tensões normais na direção y, (a) Malha1, (b) Malha2.



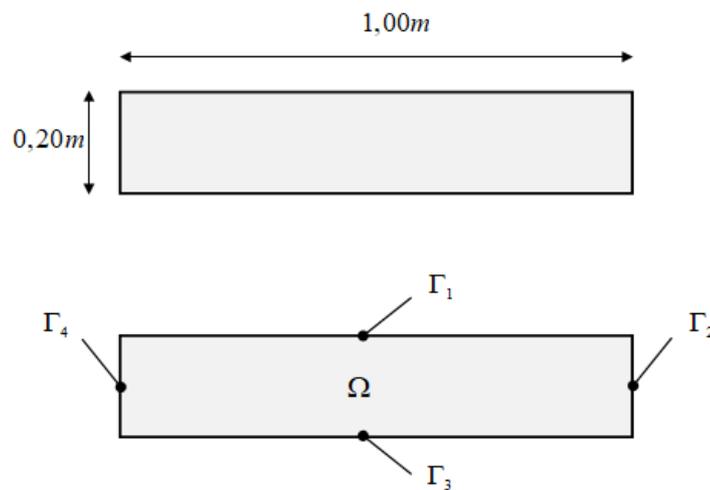
Fonte: Elaborada pelo autor.

8.3 ADENSAMENTO BIDIMENSIONAL DE MEIO HOMOGÊNIO E ISOTRÓPICO – PROBLEMA DE MANDEL

O Problema de Mandel consiste em no adensamento que se desenvolve nas duas direções, ou seja, tem caráter bidimensional, em um meio homogêneo e isotrópico. Juntamente com o problema de Terzaghi, compõe o conjunto de exemplos clássicos da poromecânica que possuem soluções analíticas e são utilizados em grande parte dos trabalhos que visam avaliar a capacidade dos simuladores e formulações numéricas desenvolvidos para a poroelasticidade.

Esse caso, definido de forma prática, consiste em um meio poroso retangular, que se estende infinitamente em uma de suas direções. Seu topo e base são comprimidos por placas rígidas e impermeáveis. Suas laterais são livres e drenantes. A geometria e condições de contorno são apresentadas na Figura 8.16 de forma detalhada.

Figura 8.16. Problema de Mandel - Geometria e condições de contorno.



Fonte: Elaborada pelo autor.

A geometria consiste em um retângulo com dimensões $1,00m \times 0,20m$ e as condições iniciais e de contorno são dadas por:

Condições iniciais ($t = 0,00s$)

$$\Gamma_1, \Gamma_2, \Gamma_3, \Gamma_4 \text{ e } \Omega \rightarrow \begin{cases} u_x = u_y = 0,00MPa \\ p = 0,00MPa \end{cases}$$

Condições de Contorno ($0,00s < t \leq 4,66 \times 10^{-3} s$)

$$\Gamma_1 \rightarrow \begin{cases} \sigma_x = 0,00MPa \\ \frac{\partial \sigma_y}{\partial t} = \frac{-1.000,00MPa}{4,66 \times 10^{-3} s} \\ q = 0,00m/s \end{cases}$$

$$\Gamma_2 \rightarrow \begin{cases} \sigma_x = \sigma_y = 0,00MPa \\ p = 0,00MPa \end{cases}$$

$$\Gamma_3 \rightarrow \begin{cases} u_y = 0,00m \\ q = 0,00m/s \end{cases}$$

$$\Gamma_4 \rightarrow \begin{cases} u_x = 0,00m \\ q = 0,00m/s \end{cases}$$

Condições de Contorno ($t > 4,66 \times 10^{-3} s$)

$$\Gamma_1 \rightarrow \begin{cases} \sigma_x = 0,00MPa \\ \sigma_y = -1.000,00MPa \\ q = 0,00m/s \end{cases}$$

$$\Gamma_2 \rightarrow \begin{cases} \sigma_x = \sigma_y = 0,00MPa \\ p = 0,00MPa \end{cases}$$

$$\Gamma_3 \rightarrow \begin{cases} u_y = 0,00m \\ q = 0,00m/s \end{cases}$$

$$\Gamma_4 \rightarrow \begin{cases} u_x = 0,00m \\ q = 0,00m/s \end{cases}$$

A partir das condições de contorno é possível perceber foi analisando considerada a simetria que ocorre em relação aos eixos horizontal e vertical, como apresentado na Figura 8.17.

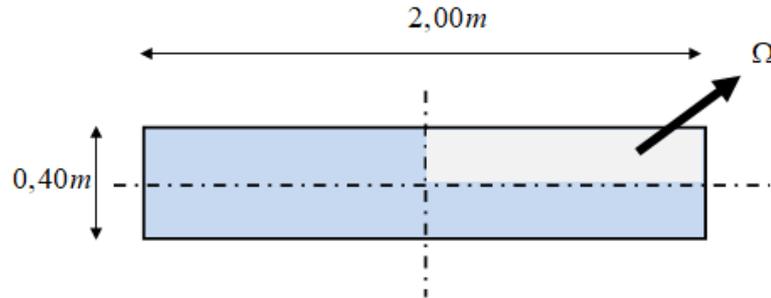
A modelagem de problemas utilizando a simetria possibilita que o analista crie malhas mais refinadas para análise sem um custo computacional tão alto quando comparado à modelagem do domínio todo. Pode haver, também, em situações mais complexas, a redução em termos de trabalho para a construção do modelo. Dessa forma, a consideração da simetria se torna, em grande parte dos casos de simulação numérica, uma forma eficiente de realizar tais simulações.

Os parâmetros físicos do problema são definidos abaixo

- $E = 1,00 \times 10^8 MPa$
- $\nu = 0,40$

- $\mathbf{\kappa} = \begin{bmatrix} 10^{-6} & 0 \\ 0 & 10^{-6} \end{bmatrix} cm/s$
- $\gamma_w = 1000,00 kgf/m^3$

Figura 8.17. Problema de Mandel - Simetria utilizada para a modelagem.



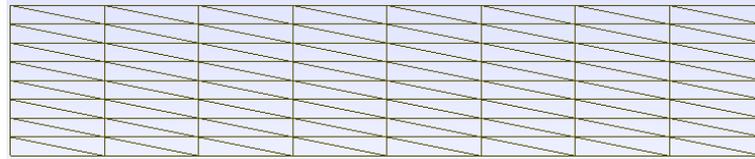
Fonte: Elaborada pelo autor.

Esse caso foi avaliado utilizando duas malhas computacionais distintas, apresentadas abaixo. O uso das malhas foi motivado pela apresentação na variação brusca dos gradientes das variáveis gerados nas proximidades do extremo direito do domínio que não é capturado de forma adequada em determinadas malhas. Tais malhas, denominadas de Malha 1 e Malha 2, são apresentadas respectivamente na Figura 8.18 (a) e (b).

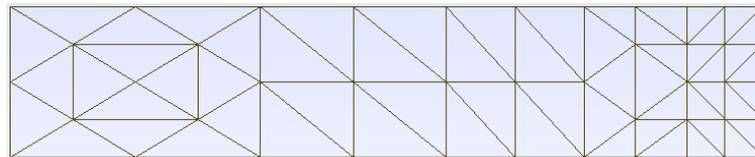
Como pode ser verificado, a Malha 2 apresenta um refinamento não uniforme ao longo do domínio, sendo o refinamento crescente do extremo esquerdo para a extremo direito do domínio. Esse refino é utilizado levando em consideração os resultados obtidos na Malha 1.

Para a determinação dos tempos de simulação é definido um fator tempo adimensional. Esse fator tempo relaciona os parâmetros geométricos e físicos e representa a evolução do processo de adensamento.

Figura 8.18. Problema de Mandel – Malhas computacionais utilizadas: (a) Malha1; (b) Malha2.



(a)



(b)

Fonte: Elaborada pelo autor.

É necessária a definição do coeficiente de adensamento c_v para a determinação do fator tempo do presente caso dado por:

$$c_v = \frac{k \left[K + \left(\frac{4}{3} \right) G \right]}{\gamma_w} \quad (8.3)$$

Como o problema é isotrópico, a permeabilidade do meio é dada por k . Os termos K e G são, respectivamente o Módulo Volumétrico e o Módulo Cisalhante, ambos tidos como Parâmetros de Lammé do meio poroso. São calculados da seguinte forma:

$$K = \frac{E}{[3(1-2\nu)]} \quad (8.4)$$

$$G = \frac{E}{[2(1+\nu)]}$$

Definidos tais parâmetros, o fator tempo é dado por:

$$T = \frac{tL^2}{c_v} \quad (8.5)$$

onde L é metade do comprimento da camada na direção x .

Os parâmetros utilizados na simulação são\:

- Fatores tempo avaliados: $T_v = 0,001$; $T_v = 0,01$; $T_v = 0,1$; $T_v = 0,50$ e $T_v = 1,00$
- Fator tempo correspondente ao t_0 : $T_{v0} = 0,001$
- Número de iterações correspondentes ao t_0 : 200,00
- Passo de tempo da simulação: $\Delta t = 2,33 \times 10^{-5}$
- Número máximo de iterações: 20.200,00

A Figura 8.19 e a Figura 8.20 apresentam, respectivamente, o resultado das pressões de fluido obtidas ao longo da reta $y=0$ para a Malha1 e para a Malha2. Como já esperado, o refinamento realizado na Malha2 suaviza o efeito da variação brusca que ocorre nos gradientes de pressões, que pode ser percebido de maneira mais efetiva nos resultados da Malha1. Esse efeito se materializa como a oscilação do resultado quando o $T_v = 0,001$. Esse tipo de oscilação na pressão pode comprometer, de maneira significativa, resultados obtidos em casos onde se precisa, por exemplo, da pressão máxima ou mínima para um certo dimensionamento. Como pode ser observado, na Malha1 a pressão numérica máxima atingiu um valor com cerca de 20% o valor da pressão analítica.

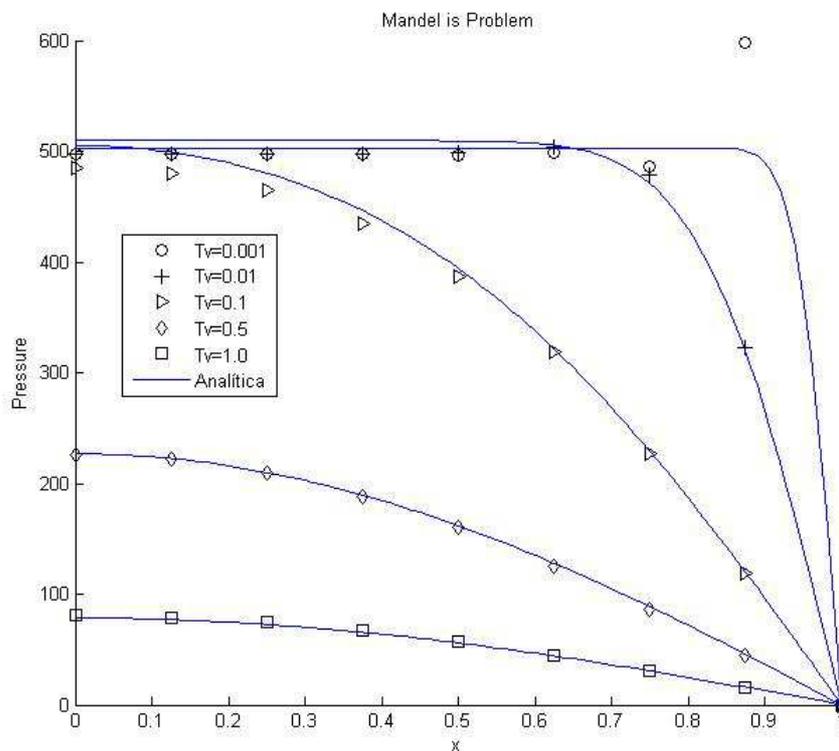
Assim como no problema de Terzaghi, pode-se perceber que para os tempos iniciais do adensamento é necessário que se faça uma análise dos resultados nas proximidades das faces drenantes, tendo em vista que nessas regiões há uma tendência à ocorrência de oscilações por conta das variações de gradientes de pressões. Outro fator que se pode observar é que a função interpoladora das variáveis hidráulicas é linear, não sendo capaz de capturar, de maneira satisfatória, a solução do problema.

É evidente, como pode-se observar na Figura 8.19 e na Figura 8.20 que, para graus mais altos de adensamento, o refinamento da malha não afeta significativamente a solução e, além disso, a oscilação gerada nos graus iniciais do adensamento não se propagam de forma a afetar a resposta em outros instante. Tal fato pode ser observado quando se compara os resultados para a Malha1 e a Malha2 onde apesar da diferença no $T_v = 0,001$, quando se compara os outros tempos apresentados os resultados são compatíveis.

A Figura 8.21 e a Figura 8.22 apresentam, respectivamente, o comportamento, em termos de deslocamentos, para o $T_v = 1,00$ para a Malha1 e para a Malha2. Nessas figuras a configuração inicial do domínio é apresentada em azul e a configuração deformada em vermelho. Os deslocamentos foram ampliados 10^4 vezes.

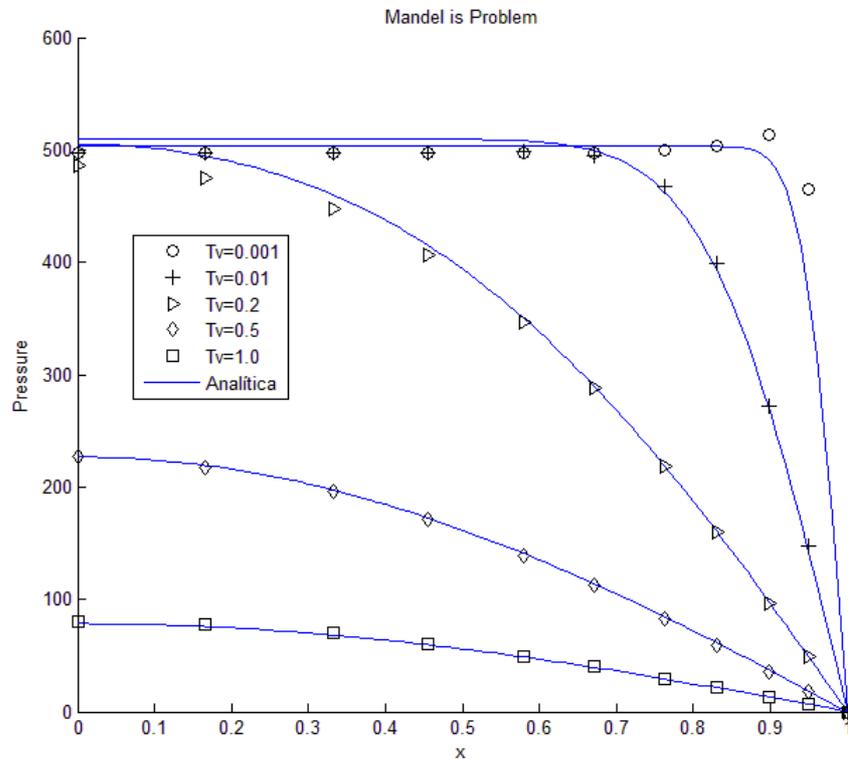
A sequência da Figura 8.23 à Figura 8.35 apresenta os resultados de pressões e tensões para o $T_v = 1,00$ obtidos utilizando as duas malhas computacionais. Os resultados se mostram coerentes.

Figura 8.19. Problema de Mandel – pressões ao longo de $y=0$ para a Malha1.



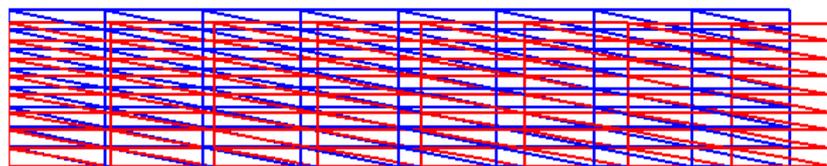
Fonte: Elaborada pelo autor.

Figura 8.20. Problema de Mandel – pressões ao longo de $y=0$ para a Malha2.



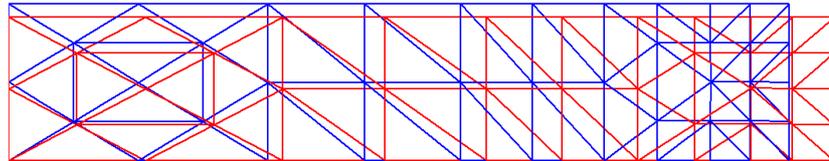
Fonte: Elaborada pelo autor.

Figura 8.21. Problema de Mandel – malha inicial (azul) e deformada (vermelho) para a Malha1 ($T_v = 1.00$).



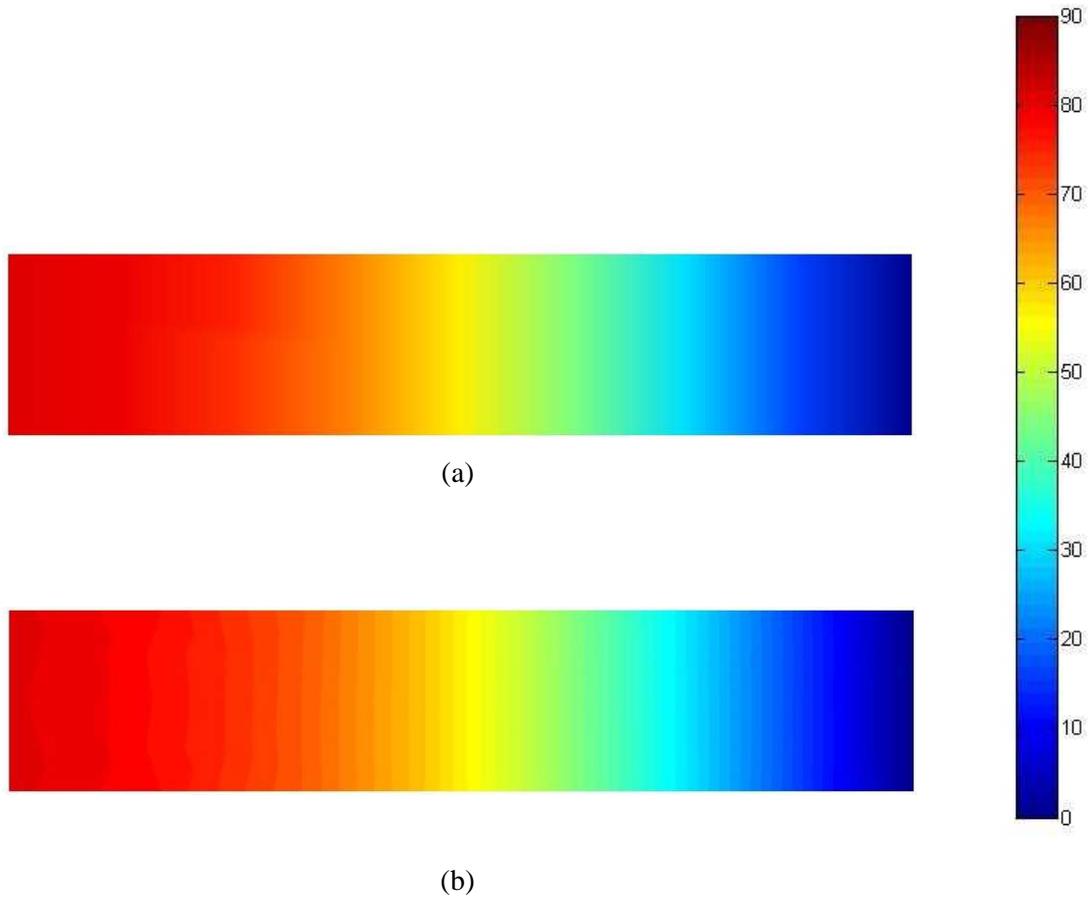
Fonte: Elaborada pelo autor.

Figura 8.22. Problema de Mandel – malha inicial (azul) e deformada (vermelho) para a Malha2
($T_v = 1,00$) .



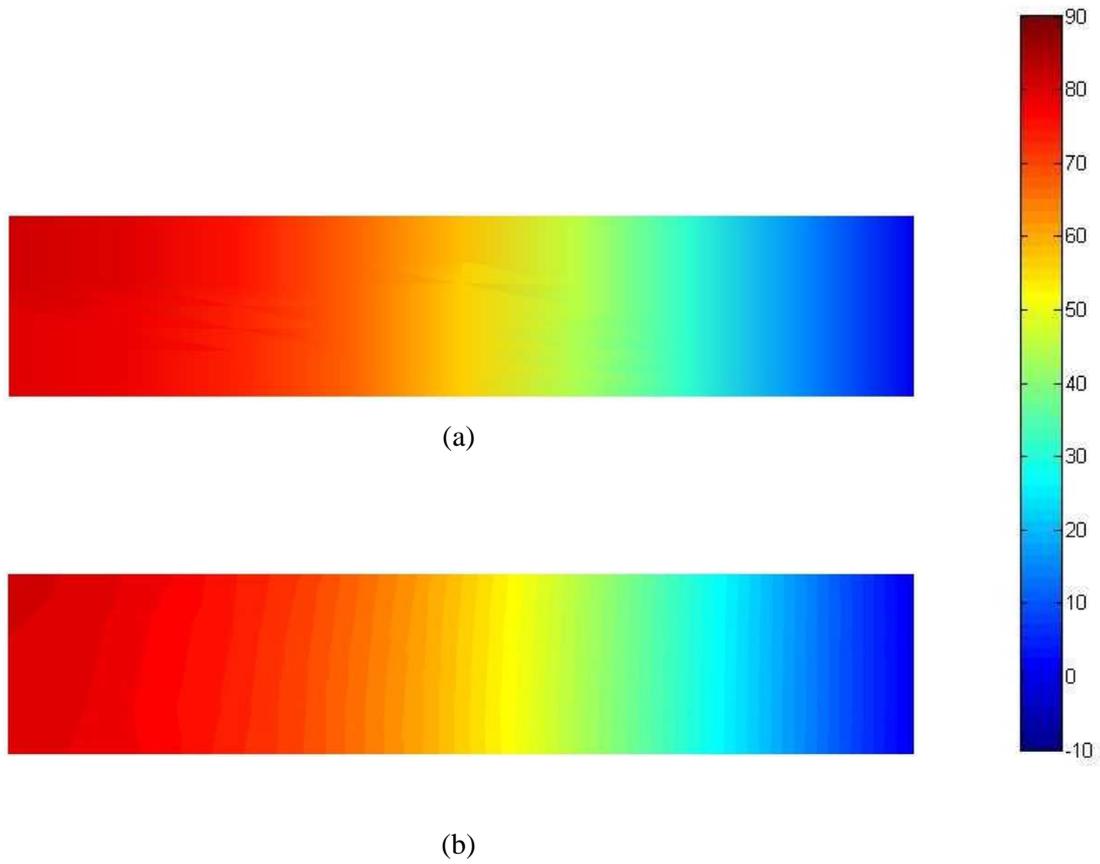
Fonte: Elaborada pelo autor.

Figura 8.23. Problema de Mandel - Campo de pressões, (a) Malha1, (b) Malha2.



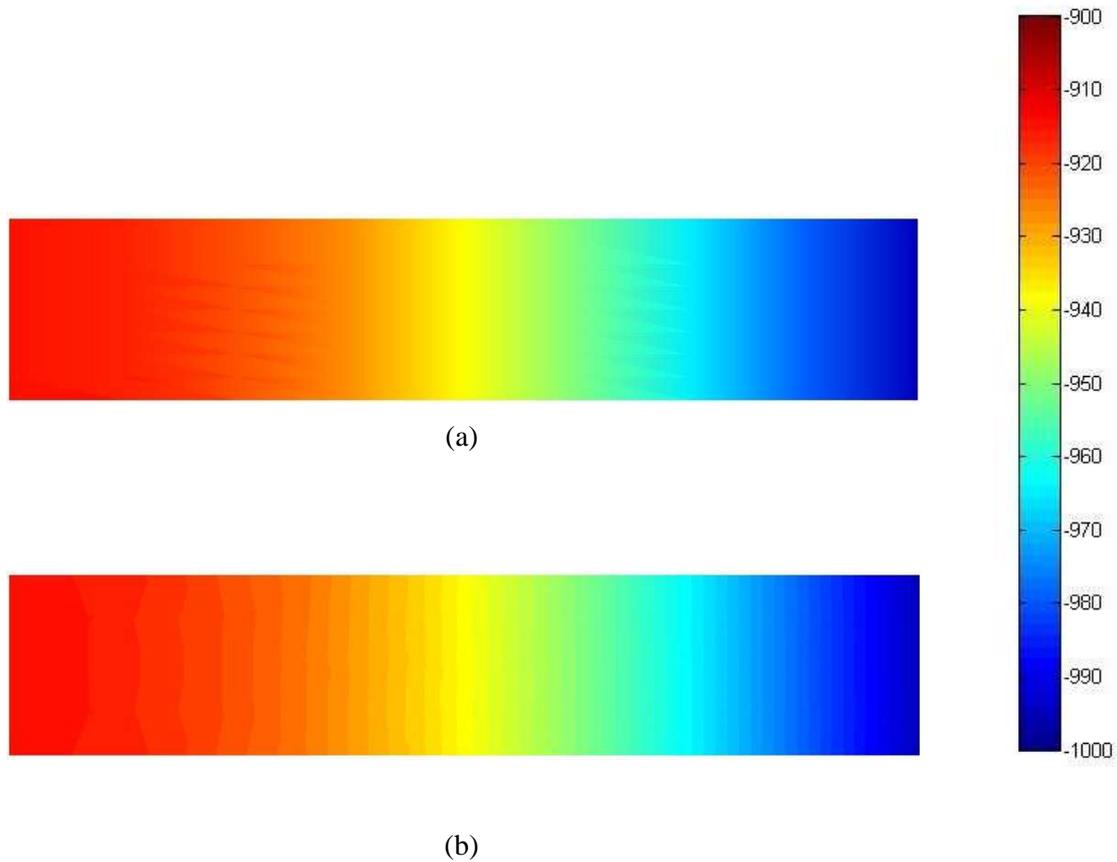
Fonte: Elaborada pelo autor.

Figura 8.24. Problema de Mandel - Campo de tensões normais na direção x, (a) Malha1, (b) Malha2.



Fonte: Elaborada pelo autor.

Figura 8.25. Problema de Mandel - Campo de tensões normais na direção y, (a) Malha1, (b) Malha2.



Fonte: Elaborada pelo autor.

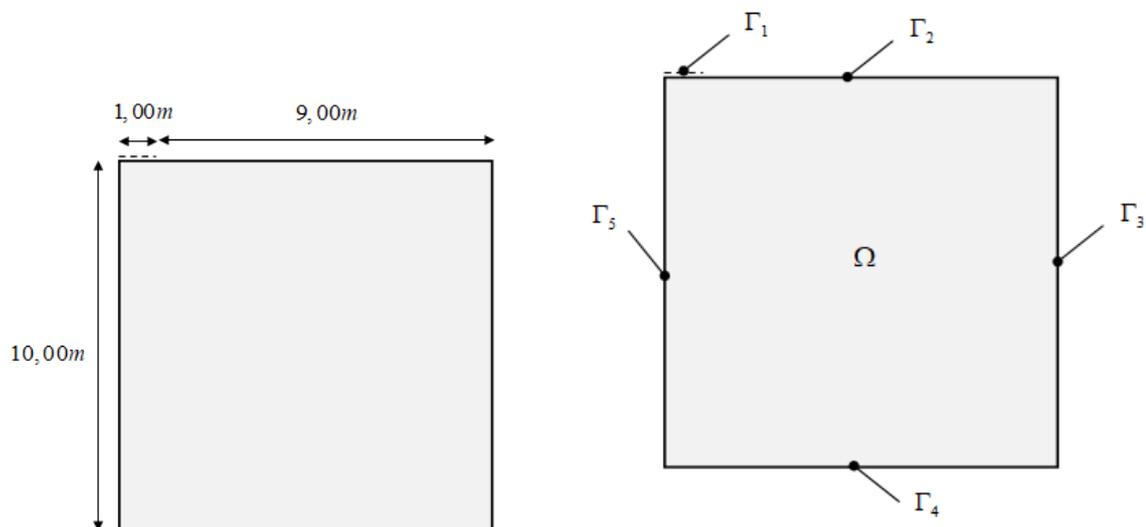
8.4 ADENSAMENTO BIDIMENSIONAL DE MEIO HOMOGENEO E ISOTRÓPICO – PROBLEMA DE SCHIFFMAN

Esse problema foi proposto por Schiffman, Chen e Jordan (1969), a fim de apresentar uma solução analítica para o comportamento do adensamento causado por uma viga de fundação contínua apoiada sobre uma camada de solo semi-infinita. Em termos de modelagem computacional a consideração de um meio semi-infinito será materializada com um domínio onde a influência das condições de contorno seja mínima no resultado. Também é uma condi-

ção do modelo que a necessidade de novos elementos para preencher tal domínio não aumente significativamente o custo computacional devido ao aumento no número de graus de liberdade analisados. Sendo assim, para o presente caso, a geometria foi tomada como um quadrado de lado dez vezes maior que a faixa de aplicação da carga que representa a sapata.

A Figura 8.26 apresenta a geometria e a definição dos contornos e do domínio. As condições de contorno serão avaliadas para duas situações, com o intuito de analisar a influência da modificação de tais condições de contorno na resposta obtida na região de interesse, que é na proximidade da viga de fundação. As duas situações possuem as condições de contorno iguais no topo, base e na face esquerda do domínio. No topo será aplicado um carregamento na faixa correspondente à área de contato da sapata com a camada de solo, e onde não houver esse contato o carregamento será nulo. As pressões no topo do domínio serão mantidas nulas ao longo da evolução do processo, permitindo a drenagem de fluido. A base será mantida impermeável e com deslocamentos verticais nulos e serão atribuídas à face esquerda as condições de simetria do problema.

Figura 8.26. Problema de Schiffman - Geometria e condições de contorno.



Fonte: Elaborada pelo autor.

Definidos a geometria e contornos, as condições iniciais e as condições de contorno para cada situações específica serão apresentadas abaixo.

Condições iniciais ($t = 0,00s$)

$$\Gamma_1, \Gamma_2, \Gamma_3, \Gamma_4, \Gamma_5 \text{ e } \Omega \rightarrow \begin{cases} u_x = u_y = 0,00MPa \\ p = 0,00MPa \end{cases}$$

Condições de Contorno (Situação 1) ($0,00s < t \leq 100,00s$)

$$\Gamma_1 \rightarrow \begin{cases} \sigma_x = 0,00MPa \\ \frac{\partial \sigma_y}{\partial t} = \frac{-1.000,00MPa}{100,00s} \\ q = 0,00m/s \end{cases}$$

$$\Gamma_2 \rightarrow \begin{cases} \sigma_x = \sigma_y = 0,00MPa \\ p = 0,00MPa \end{cases}$$

$$\Gamma_3 \rightarrow \begin{cases} u_x = 0,00m \\ p = 0,00MPa \end{cases}$$

$$\Gamma_4 \rightarrow \begin{cases} u_y = 0,00m \\ q = 0,00m/s \end{cases}$$

$$\Gamma_5 \rightarrow \begin{cases} u_x = 0,00m \\ q = 0,00m/s \end{cases}$$

Condições de Contorno (Situação 1) ($100,00s < t$)

$$\Gamma_1 \rightarrow \begin{cases} \sigma_x = 0,00MPa \\ \sigma_y = -1.000,00MPa \\ q = 0,00m/s \end{cases}$$

$$\Gamma_2 \rightarrow \begin{cases} \sigma_x = \sigma_y = 0,00MPa \\ p = 0,00MPa \end{cases}$$

$$\Gamma_3 \rightarrow \begin{cases} u_x = 0,00m \\ p = 0,00MPa \end{cases}$$

$$\Gamma_4 \rightarrow \begin{cases} u_y = 0,00m \\ q = 0,00m/s \end{cases}$$

$$\Gamma_5 \rightarrow \begin{cases} u_x = 0,00m \\ q = 0,00m/s \end{cases}$$

Condições de Contorno (Situação 2) ($0,00s < t \leq 100,00s$)

$$\Gamma_1 \rightarrow \begin{cases} \sigma_x = 0,00MPa \\ \frac{\partial \sigma_y}{\partial t} = \frac{-1.000,00MPa}{100,00s} \\ q = 0,00m/s \end{cases}$$

$$\Gamma_2 \rightarrow \begin{cases} \sigma_x = \sigma_y = 0,00MPa \\ p = 0,00MPa \end{cases}$$

$$\Gamma_3 \rightarrow \begin{cases} \sigma_x = \sigma_y = 0,00MPa \\ q = 0,00m/s \end{cases}$$

$$\Gamma_4 \rightarrow \begin{cases} u_y = 0,00m \\ q = 0,00m/s \end{cases}$$

$$\Gamma_5 \rightarrow \begin{cases} u_x = 0,00m \\ q = 0,00m/s \end{cases}$$

Condições de Contorno (Situação 2) ($100,00s < t$)

$$\Gamma_1 \rightarrow \begin{cases} \sigma_x = 0,00MPa \\ \sigma_y = -1.000,00MPa \\ q = 0,00m/s \end{cases}$$

$$\Gamma_2 \rightarrow \begin{cases} \sigma_x = \sigma_y = 0,00MPa \\ p = 0,00MPa \end{cases}$$

$$\Gamma_3 \rightarrow \begin{cases} \sigma_x = \sigma_y = 0,00MPa \\ q = 0,00m/s \end{cases}$$

$$\Gamma_4 \rightarrow \begin{cases} u_y = 0,00m \\ q = 0,00m/s \end{cases}$$

$$\Gamma_5 \rightarrow \begin{cases} u_x = 0,00m \\ q = 0,00m/s \end{cases}$$

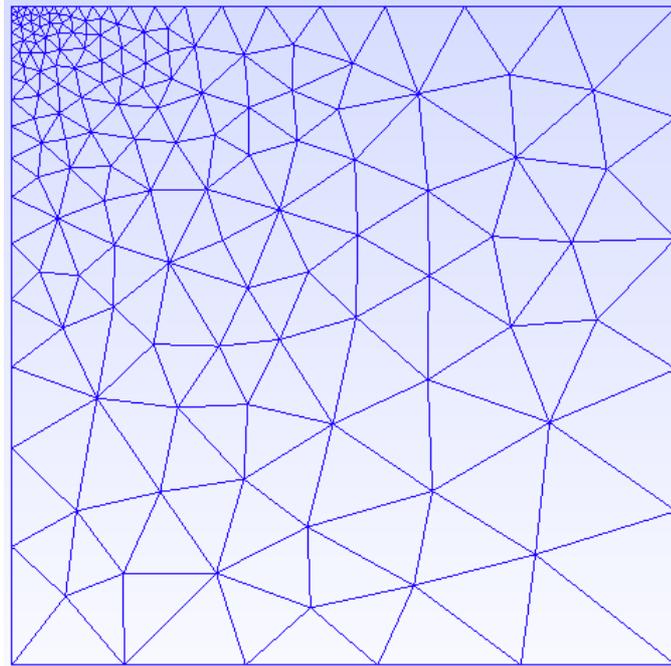
Os parâmetros físicos do problema são definidos abaixo

- $E = 1,00 \times 10^8 MPa$
- $\nu = 0,00$
- $\mathbf{\kappa} = \begin{bmatrix} 10^{-8} & 0 \\ 0 & 10^{-8} \end{bmatrix} cm/s$
- $\gamma_w = 1000,00 kgf/m^3$

A malha computacional utilizada na simulação será apresentada na Figura 8.27. O refinamento da malha foi feito visando otimizar o processo, para isso o maior refinamento ocorre na região de interesse, que compreende a região nas proximidades da aplicação da carga. Para a

simulação realizada os elementos nas proximidades dos contornos mais distantes se mostraram suficientes para transmitir os efeitos das condições de contorno à região de interesse.

Figura 8.27. Problema de Schiffman – Malha computacional.



Fonte: Elaborada pelo autor.

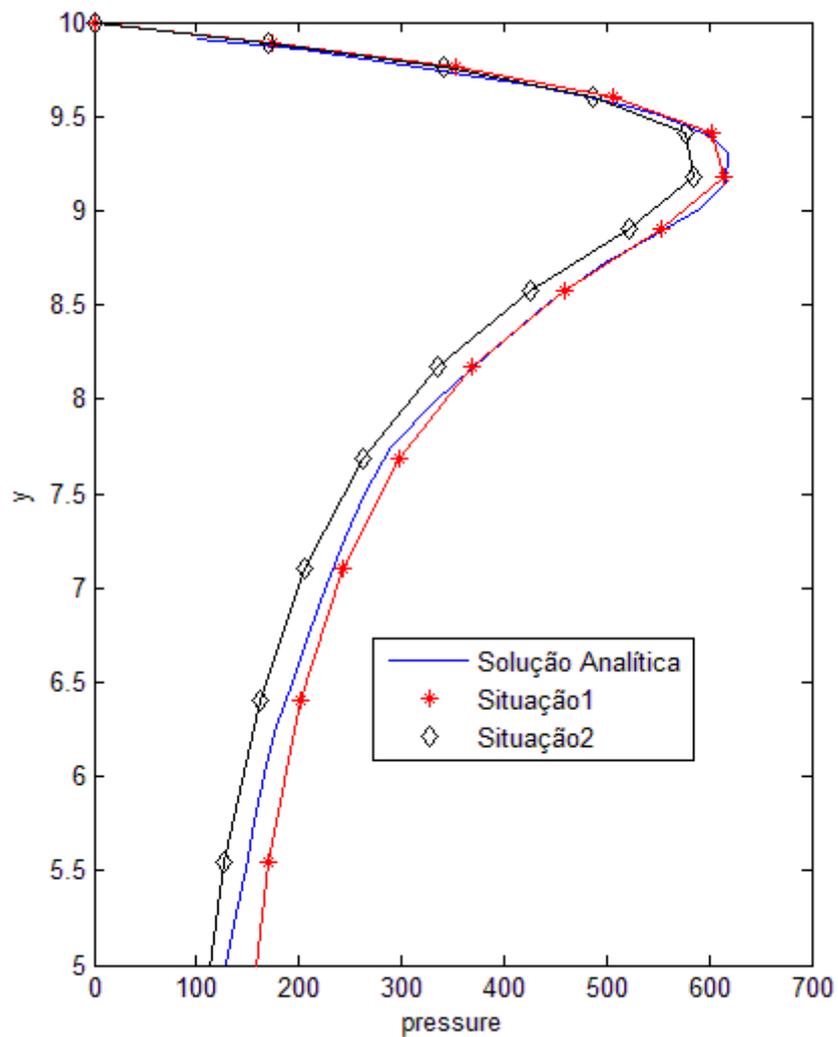
Os parâmetros utilizados na simulação são

- Fatores tempo avaliados: $T_v = 0,10$;
- Fator tempo correspondente ao $t_0 : T_{v,0} = 0,001$
- Número de iterações correspondentes ao $t_0 : 100,00$
- Passo de tempo da simulação: $\Delta t = 1,00$
- Número máximo de iterações: 10.100,00

Os resultados obtidos foram comparados com os resultados da solução analítica de (Schiffman, Chen, & Jordan, 1969) apresentados no trabalho de (Cavalcanti, 2002). Nesse

trabalho é apresentada a expressão utilizada para a obtenção do fator tempo. A solução analítica consiste na distribuição das pressões de fluido ao longo de $x=0$, e nesse trabalho foi analisada para um $T_v = 0,10$. A Figura 8.28 apresenta os resultados das pressões obtidas pelas duas situações, diferenciadas pelas condições de contorno, e a solução analítica.

Figura 8.28. Problema de Schiffman – solução analítica e soluções numéricas das pressões de fluido ao longo de $x=0$ ($T_v = 0,10$).



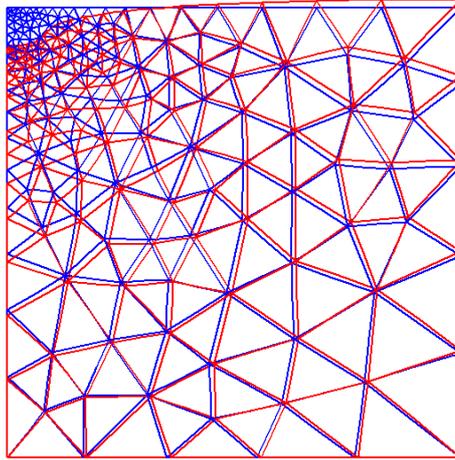
Fonte: Elaborada pelo autor.

Como se pode observar, as soluções numéricas apresentam uma concordância com a solução analítica. É evidente que a situação1 se mostra mais próxima à solução analítica do que a situação2 na região onde há os maiores valores de pressões, porém até mesmo a situação2 apresentou resultados satisfatórios.

A Figura 8.29 e a Figura 8.30 apresentam a configuração inicial e a configuração deformada do domínio para o $T_v = 0,10$ para a situação1 e a situação2, respectivamente. Os detalhes dessas figuras são apresentados na Figura 8.31 e na Figura 8.32 e, como pode ser observado, mostra a influência da condição de contorno no deslocamento da superfície. A Figura 8.31, em comparação com a Figura 8.32 apresenta, no bordo direito da face superior do domínio, um deslocamento vertical mais expressivo. Esse efeito ocorre pois na situação1 há menos possibilidades do meio se deformar em função das condições de contorno impostas. Em contrapartida a situação2, como apresentada na Figura 8.32, apresenta deslocamentos horizontais que podem não condizer com uma situação real, pois essa consideração despreza totalmente a rigidez de uma possível massa adjacente ao domínio. Para a situação de uma viga de fundação essa massa poderia ser a continuação da camada de solo. De forma análoga, a situação1 representa uma condição de rigidez infinita que na prática é impossível, porém se houver uma rocha de rigidez consideravelmente maior adjacente a essa aresta essa condição pode ser um boa simplificação. Os resultados dos deslocamentos foram ampliados em 5×10^4 vezes.

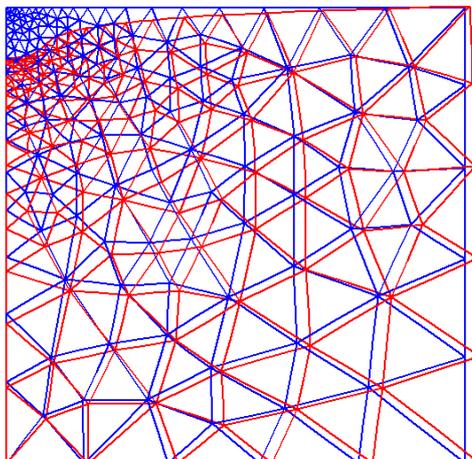
Os resultados desses deslocamentos mostram que o analista numérico precisa de além de experiência na colocação das condições de contorno dos problemas, ter bom senso em relação aos resultados sabendo que durante a interpretação os resultados foram obtidos a partir de métodos aproximados, onde são aplicadas simplificações de diversas naturezas, e as condições de contorno são representações aproximadas das manifestações reais.

Figura 8.29. Problema de Schiffman – malha inicial (azul) e deformada (vermelho) para a situação1
($T_v = 0,10$).



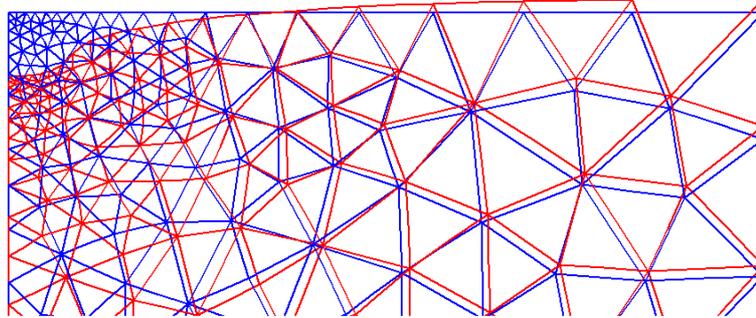
Fonte: Elaborada pelo autor.

Figura 8.30. Problema de Schiffman – malha inicial (azul) e deformada (vermelho) para a situação2
($T_v = 0,10$).



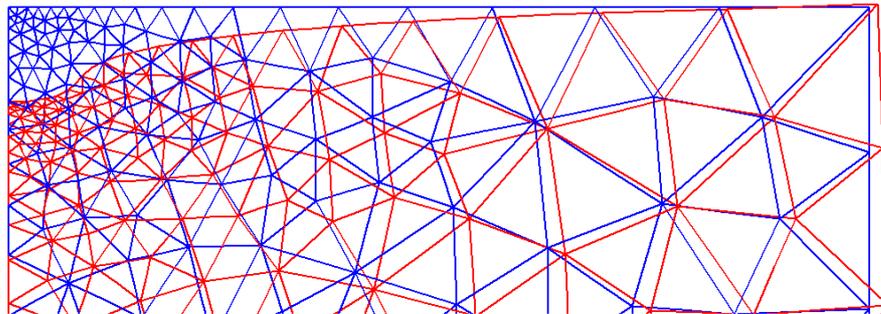
Fonte: Elaborada pelo autor.

Figura 8.31. Problema de Schiffman – detalhe da malha inicial (azul) e deformada (vermelho) para a situação1 ($T_v = 0,10$).



Fonte: Elaborada pelo autor.

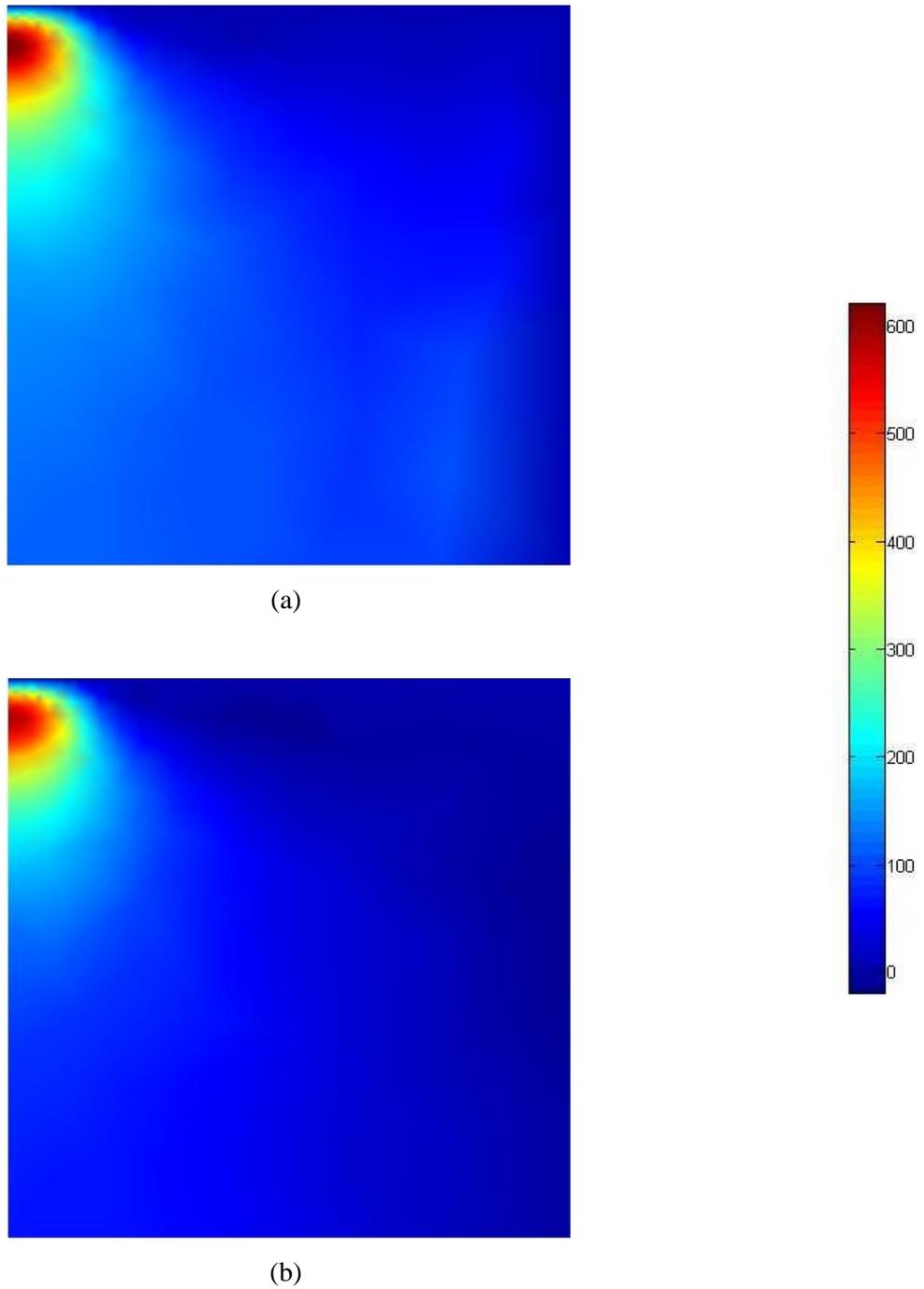
Figura 8.32. Problema de Schiffman – detalhes da malha inicial (azul) e da malha deformada (vermelho) para a situação2 ($T_v = 0,10$).



Fonte: Elaborada pelo autor.

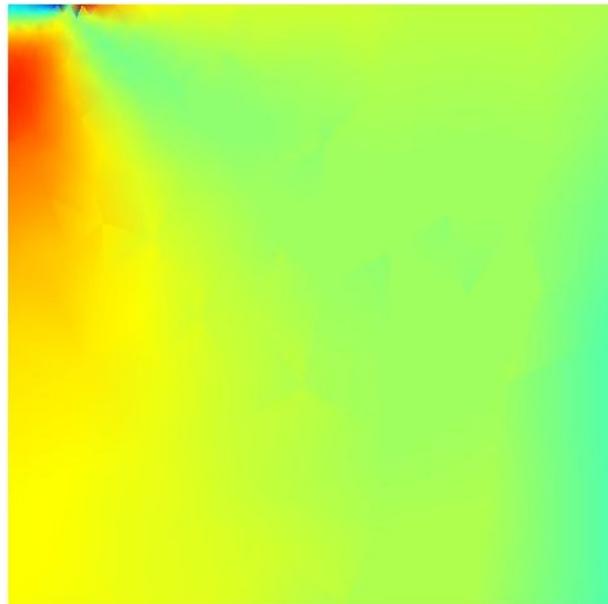
A sequência da Figura 8.33 à Figura 8.36 apresenta os resultados das pressões de fluidos e tensões. Como podem ser observados há diferenças entre as duas situações porém os resultados apresentam uma boa concordância, e em particular na região de interesse.

Figura 8.33. Problema de Schiffman - Campo de pressões, (a) Situação1, (b) Situação2.

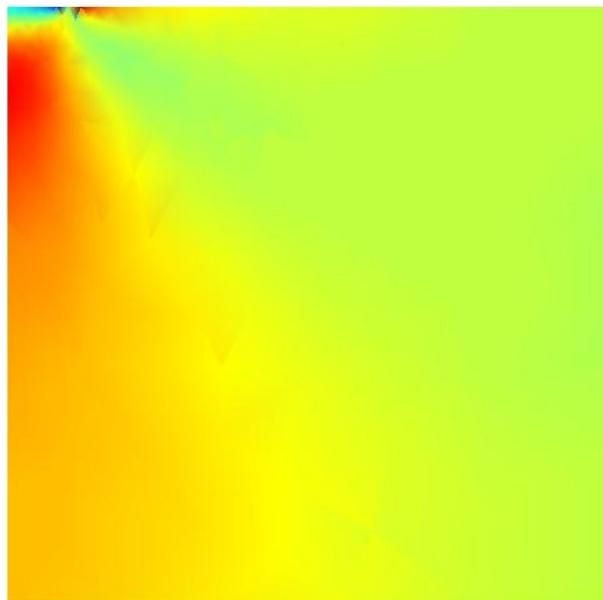


Fonte: Elaborada pelo autor.

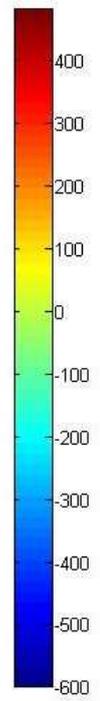
Figura 8.34. Problema de Schiffman - Campo de tensões normais na direção x , (a) Situação1, (b) Situação2.



(a)

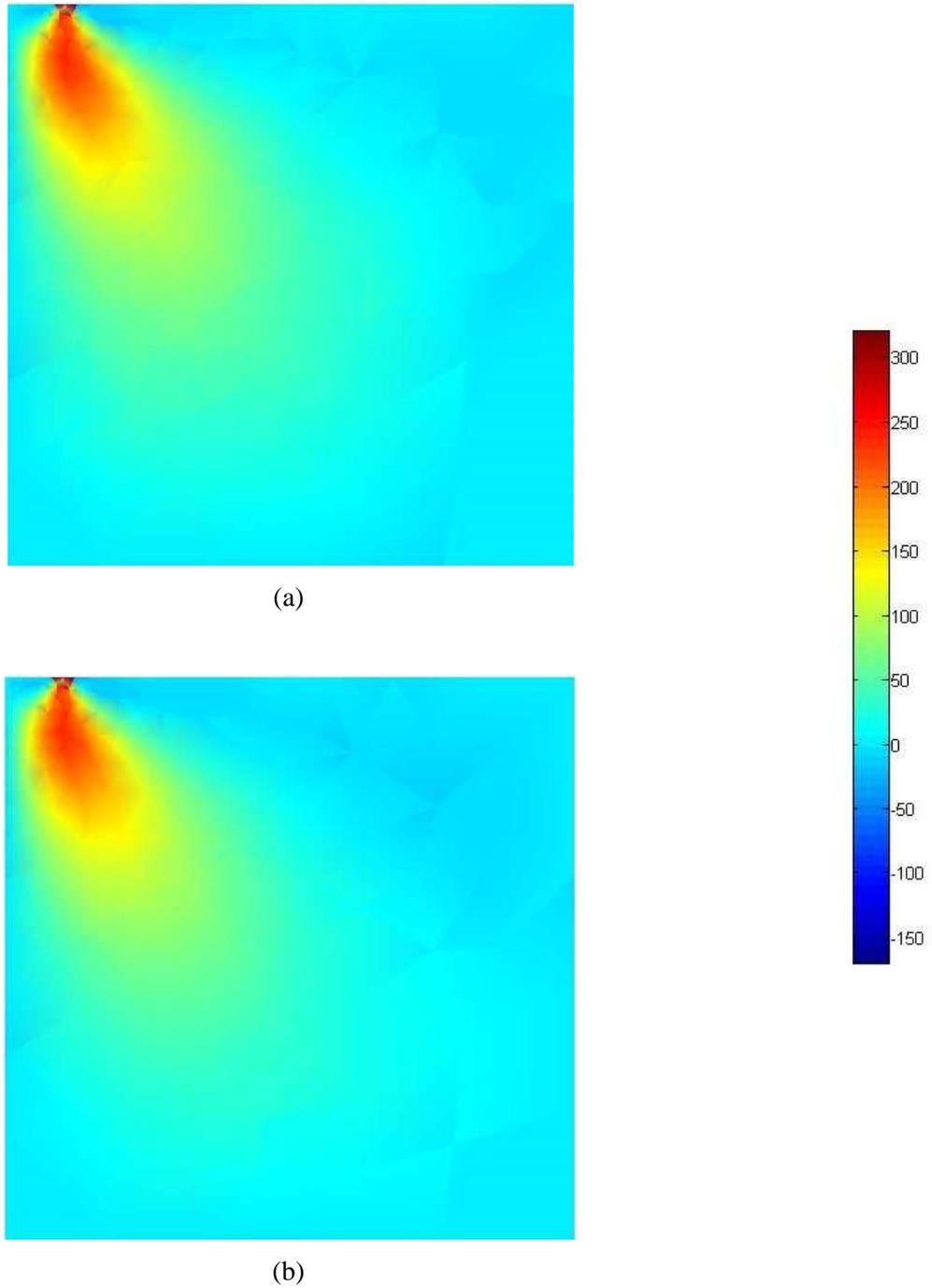


(b)



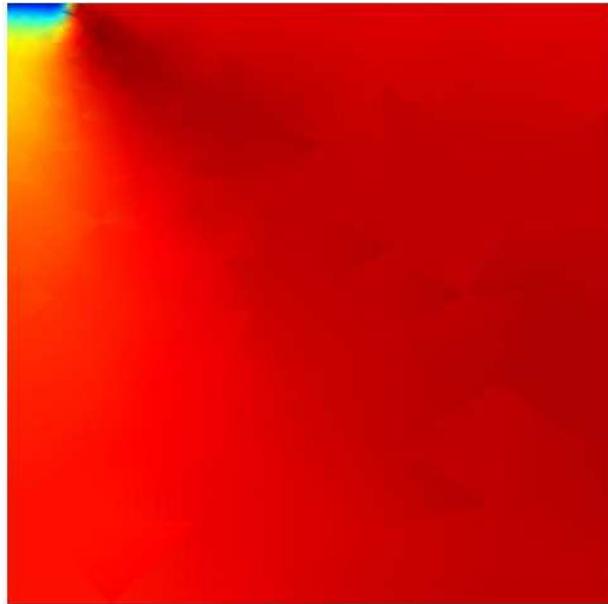
Fonte: Elaborada pelo autor.

Figura 8.35. Problema de Schiffman - Campo de tensões cisalhantes, (a) Malha1, (b) Malha2.

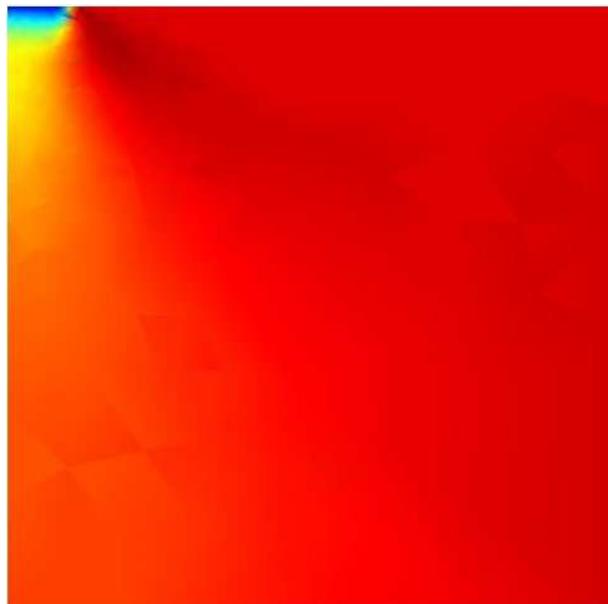


Fonte: Elaborada pelo autor.

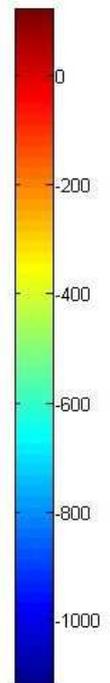
Figura 8.36. Problema de Schiffman - Campo de tensões normais na direção y, (a) Situação1, (b) Situação2.



(a)



(b)



Fonte: Elaborada pelo autor.

8.5 ADENSAMENTO BIDIMENSIONAL DE MEIO HETEROGÊNIO E ANISOTRÓPICO

Grande parte dos problemas poromecânicos são processos encontrados na natureza e se apresentam com características anisotrópicas e heterogêneas. Alguns desses exemplos são encontrados na engenharia geotécnica, na engenharia de reservatórios de petróleo e até na engenharia biomédica. Um dos grandes exemplos de meios anisotrópicos são os reservatórios de petróleo, que em geral possuem a permeabilidade vertical diferente da permeabilidade horizontal. Outro exemplo pode ser a diferença entre os tecidos do corpo que está sendo analisado.

Sendo assim, o desenvolvimento de simuladores que sejam capazes de simular problemas anisotrópicos e heterogêneos é de grande interesse para a indústria e centros de pesquisa. Com essa motivação Ai, Cao e Cheng (2010) apresentam a solução analítica para o adensamento de um solo estratificado em camadas horizontais, submetido ao carregamento de uma sapata de fundação contínua.

O caso propriamente dito consiste em um meio poroso, sujeito a um carregamento na face superior em uma faixa do domínio. O meio poroso é composto por três camadas horizontais que possuem valores diferentes de permeabilidades e módulo de elasticidade. O coeficiente de Poisson é igual para todo o meio. As permeabilidades nas direções x e y são diferentes para cada camada. A Figura 8.37 apresenta a geometria e a definição dos contornos e domínios.

As condições iniciais e de contorno do problema são dadas abaixo

Condições iniciais ($t = 0,00s$)

$$\Gamma_1, \Gamma_2, \Gamma_3, \Gamma_4, \Gamma_5 \text{ e } \Omega \rightarrow \begin{cases} u_x = u_y = 0,00MPa \\ p = 0,00MPa \end{cases}$$

Condições de Contorno ($0,00s < t \leq 7,50 \times 10^{-1}s$)

$$\Gamma_1 \rightarrow \begin{cases} \sigma_x = 0,00MPa \\ \frac{\partial \sigma_y}{\partial t} = \frac{-1.000,00MPa}{7,50 \times 10^{-1}s} \\ q = 0,00m/s \end{cases}$$

$$\Gamma_2 \rightarrow \begin{cases} \sigma_x = \sigma_y = 0,00MPa \\ p = 0,00MPa \end{cases}$$

$$\Gamma_3 \rightarrow \begin{cases} u_x = 0,00m \\ p = 0,00MPa \end{cases}$$

$$\Gamma_4 \rightarrow \begin{cases} u_y = 0,00m \\ q = 0,00m/s \end{cases}$$

$$\Gamma_5 \rightarrow \begin{cases} u_x = 0,00m \\ q = 0,00m/s \end{cases}$$

Condições de Contorno ($t > 7,50 \times 10^{-1}s$)

$$\Gamma_1 \rightarrow \begin{cases} \sigma_x = 0,00MPa \\ \sigma_y = -1.000,00MPa \\ q = 0,00m/s \end{cases}$$

$$\Gamma_2 \rightarrow \begin{cases} \sigma_x = \sigma_y = 0,00MPa \\ p = 0,00MPa \end{cases}$$

$$\Gamma_3 \rightarrow \begin{cases} u_x = 0,00m \\ p = 0,00MPa \end{cases}$$

$$\Gamma_4 \rightarrow \begin{cases} u_y = 0,00m \\ q = 0,00m/s \end{cases}$$

$$\Gamma_5 \rightarrow \begin{cases} u_x = 0,00m \\ q = 0,00m/s \end{cases}$$

Os parâmetros físicos são definidos abaixo

- $E_1 = 10^7 MPa$, $E_2 = 2,00 \times 10^7 MPa$ e $E_3 = 4,00 \times 10^7 MPa$
- $\nu = 0,00$
- $\gamma_w = 1000,00 kgf/m^3$
- Permeabilidades para a Situação 1, onde a razão entre as permeabilidades horizontais e verticais é igual a 1.

$$\underline{\underline{\mathbf{\kappa}_1}} = \begin{bmatrix} 4 \times 10^{-7} & 0 \\ 0 & 4 \times 10^{-7} \end{bmatrix} cm/s$$

$$\underline{\underline{\mathbf{\kappa}_2}} = \begin{bmatrix} 2 \times 10^{-7} & 0 \\ 0 & 2 \times 10^{-7} \end{bmatrix} cm/s$$

$$\underline{\underline{\kappa}}_3 = \begin{bmatrix} 10^{-7} & 0 \\ 0 & 10^{-7} \end{bmatrix} cm/s$$

- Permeabilidades para a Situação 2, onde a razão entre as permeabilidades horizontais e verticais é igual a 5.

$$\underline{\underline{\kappa}}_1 = \begin{bmatrix} 20 \times 10^{-7} & 0 \\ 0 & 4 \times 10^{-7} \end{bmatrix} cm/s$$

$$\underline{\underline{\kappa}}_2 = \begin{bmatrix} 10 \times 10^{-7} & 0 \\ 0 & 2 \times 10^{-7} \end{bmatrix} cm/s$$

$$\underline{\underline{\kappa}}_3 = \begin{bmatrix} 5 \times 10^{-7} & 0 \\ 0 & 10^{-7} \end{bmatrix} cm/s$$

- Permeabilidades para a Situação 3, onde a razão entre as permeabilidades horizontais e verticais é igual a 50.

$$\underline{\underline{\kappa}}_1 = \begin{bmatrix} 200 \times 10^{-7} & 0 \\ 0 & 4 \times 10^{-7} \end{bmatrix} cm/s$$

$$\underline{\underline{\kappa}}_2 = \begin{bmatrix} 100 \times 10^{-7} & 0 \\ 0 & 2 \times 10^{-7} \end{bmatrix} cm/s$$

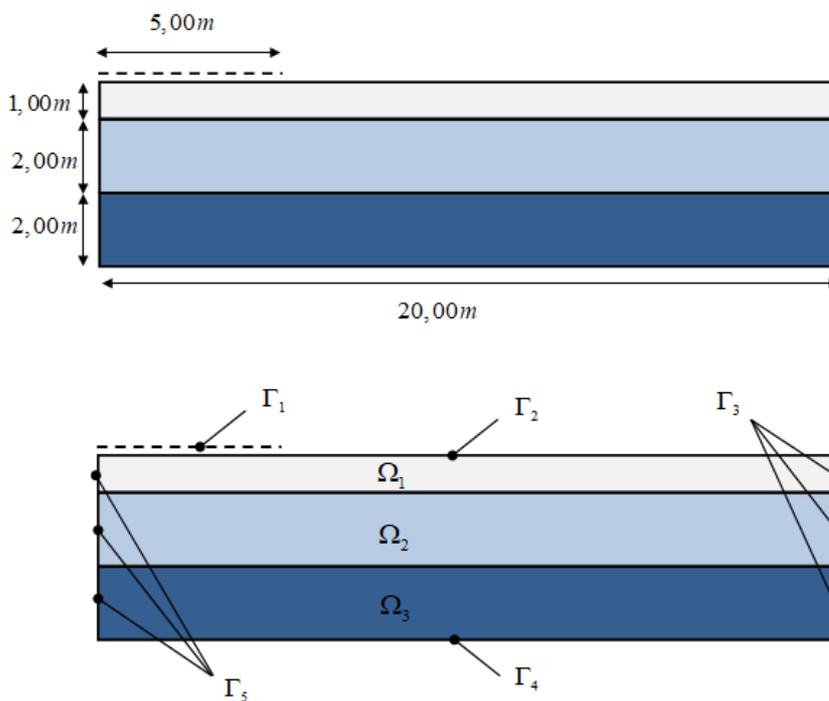
$$\underline{\underline{\kappa}}_3 = \begin{bmatrix} 50 \times 10^{-7} & 0 \\ 0 & 10^{-7} \end{bmatrix} cm/s$$

Como pode ser verificada a partir da definição das permeabilidades de cada região do domínio a anisotropia nas camadas vai se tornando cada vez mais expressiva. A forma quantitativa de tratar dessa anisotropia será definida como razão de anisotropia(r) que é definida como

$$r = \frac{k_{hi}}{k_{vi}} \tag{8.6}$$

onde k_{hi} e k_{vi} são, respectivamente, as permeabilidades horizontais e verticais da camada i

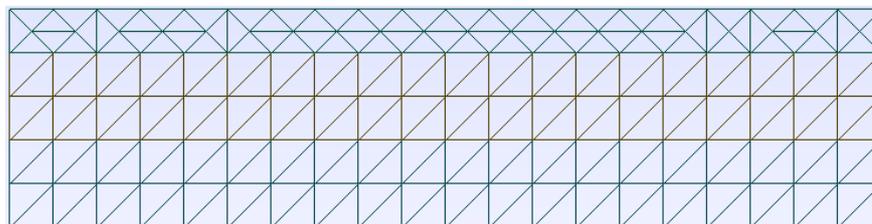
Figura 8.37. Problema Heterogêneo e Anisotrópico – Geometria, contornos e domínios.



Fonte: Elaborada pelo autor.

Para as análises foi utilizada a malha computacional apresentada na Figura 8.38.

Figura 8.38. Problema Heterogêneo e Anisotrópico – Malha computacional.

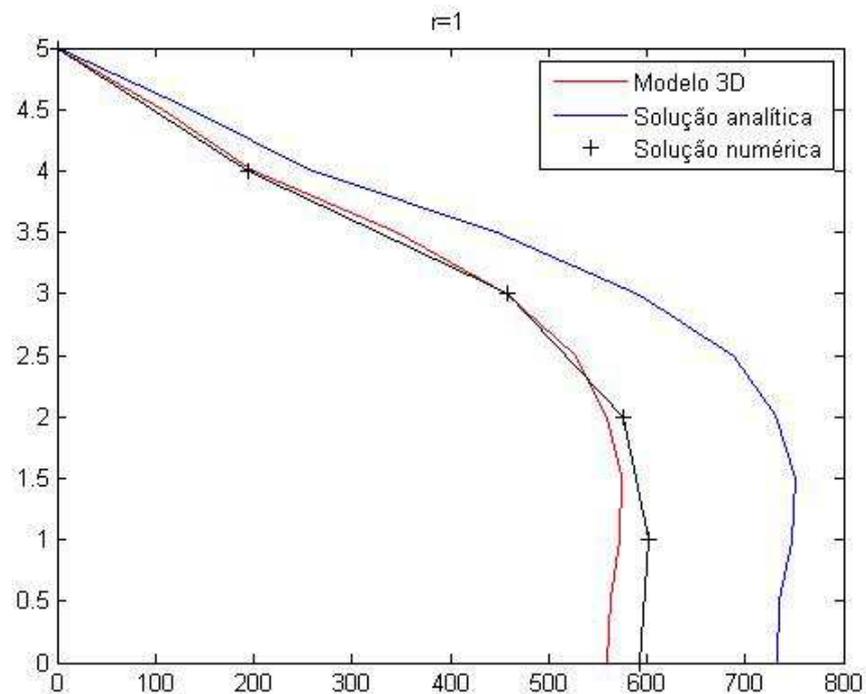


Fonte: Elaborada pelo autor.

A solução analítica assim como o fator tempo são apresentados no trabalho de Ai, Cao e Cheng (2010), onde é feito o desenvolvimento de uma solução analítica. Em seu trabalho Ai, Cao e Cheng (2010) também apresenta os resultados obtidos em um modelo tridimensional desenvolvido no ABAQUS®. Esses resultados serão comparados com os obtidos no código desenvolvido.

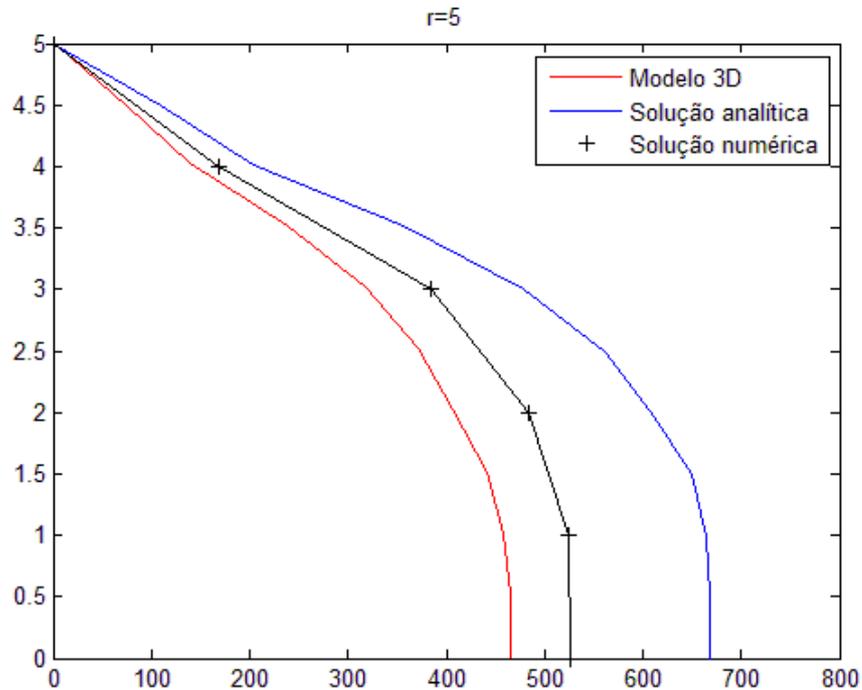
A Figura 8.39, Figura 8.40 e Figura 8.41 apresentam os resultados para as pressões ao longo da reta $x=0$ o fator tempo $T_v = 0,10$ para as razões de anisotropia $r=1$, $r=5$ e $r=50$ respectivamente.

Figura 8.39. Problema Heterogêneo e Anisotrópico – solução analítica e soluções numéricas das pressões de fluido ao longo de $x=0$ para $r=1$ ($T_v = 0,10$).



Fonte: Elaborada pelo autor.

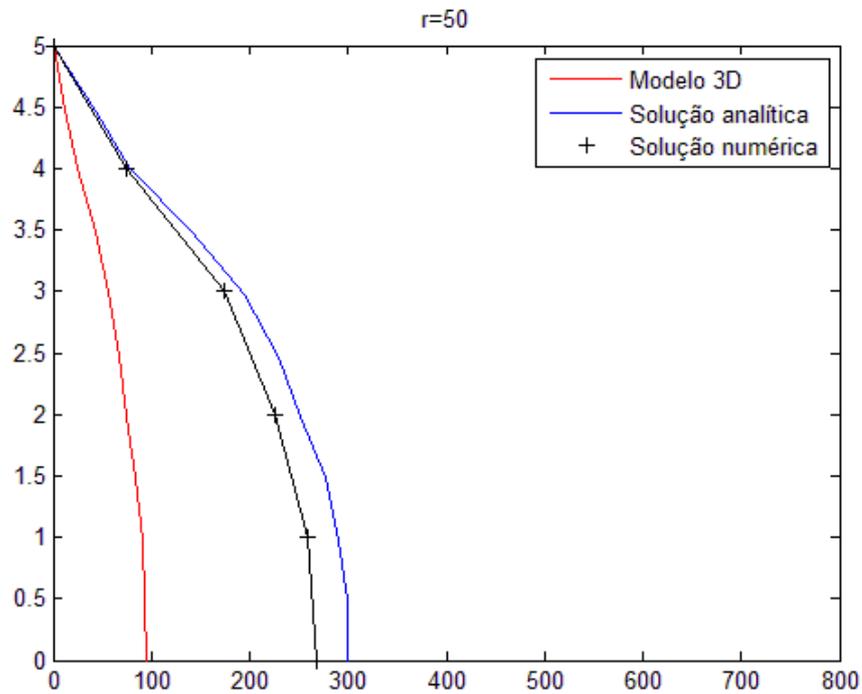
Figura 8.40. Problema Heterogêneo e Anisotrópico – solução analítica e soluções numéricas das pressões de fluido ao longo de $x=0$ para $r=5$ ($T_v = 0,10$).



Fonte: Elaborada pelo autor.

De acordo com os resultados apresentados pode-se perceber que há, de fato, uma diferença um tanto significativa entre as soluções apresentadas no próprio trabalho de Ai, Cao e Cheng (2010). Porém os resultados apresentados pelo código desenvolvido apresentam valores e comportamento intermediário entre os dois. Sendo assim, esse exemplo além de apresentar o comportamento do método serve para mostrar que o tipo de modelo utilizado para a aproximação de uma situação real pode afetar significativamente o resultado e tem que ser escolhido com cautela por parte de quem aplica tais métodos. Mesmo com as considerações sobre as diferenças verificadas nas soluções pode-se dizer que o resultado obtido foi satisfatório, comprovando a eficácia do método frente à problemas de natureza heterogênea e anisotrópica.

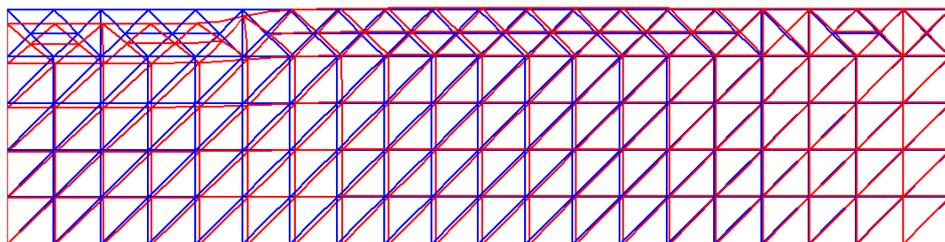
Figura 8.41. Problema Heterogêneo e Anisotrópico – solução analítica e soluções numéricas das pressões de fluido ao longo de $x=0$ para $r = 50$ ($T_v = 0,10$).



Fonte: Elaborada pelo autor.

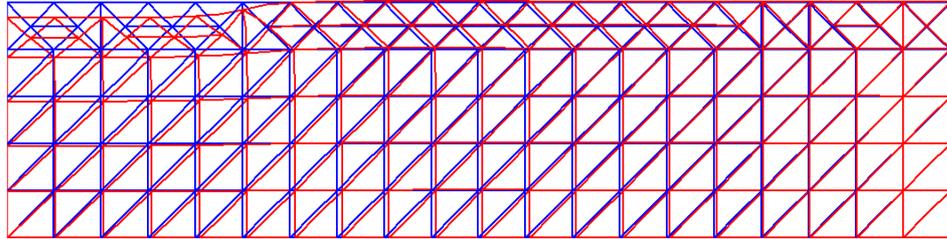
A sequência da Figura 8.42 à Figura 8.44 apresenta as deformações, em termos da malha computacional, para cada razão de anisotropia. Os deslocamentos foram ampliados em 100 vezes para a configuração deformada.

Figura 8.42. Problema Heterogêneo e Anisotrópico – detalhes da malha inicial (azul) e da malha deformada (vermelha) para $r = 1$ ($T_v = 0,10$).



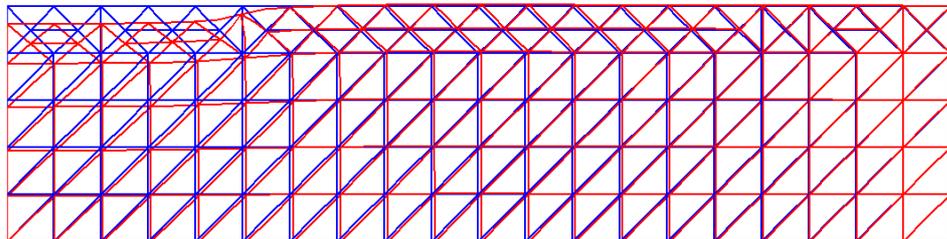
Fonte: Elaborada pelo autor.

Figura 8.43. Problema Heterogêneo e Anisotrópico – detalhes da malha inicial (azul) e da malha deformada (vermelho) para $r = 5$ ($T_v = 0,10$).



Fonte: Elaborada pelo autor.

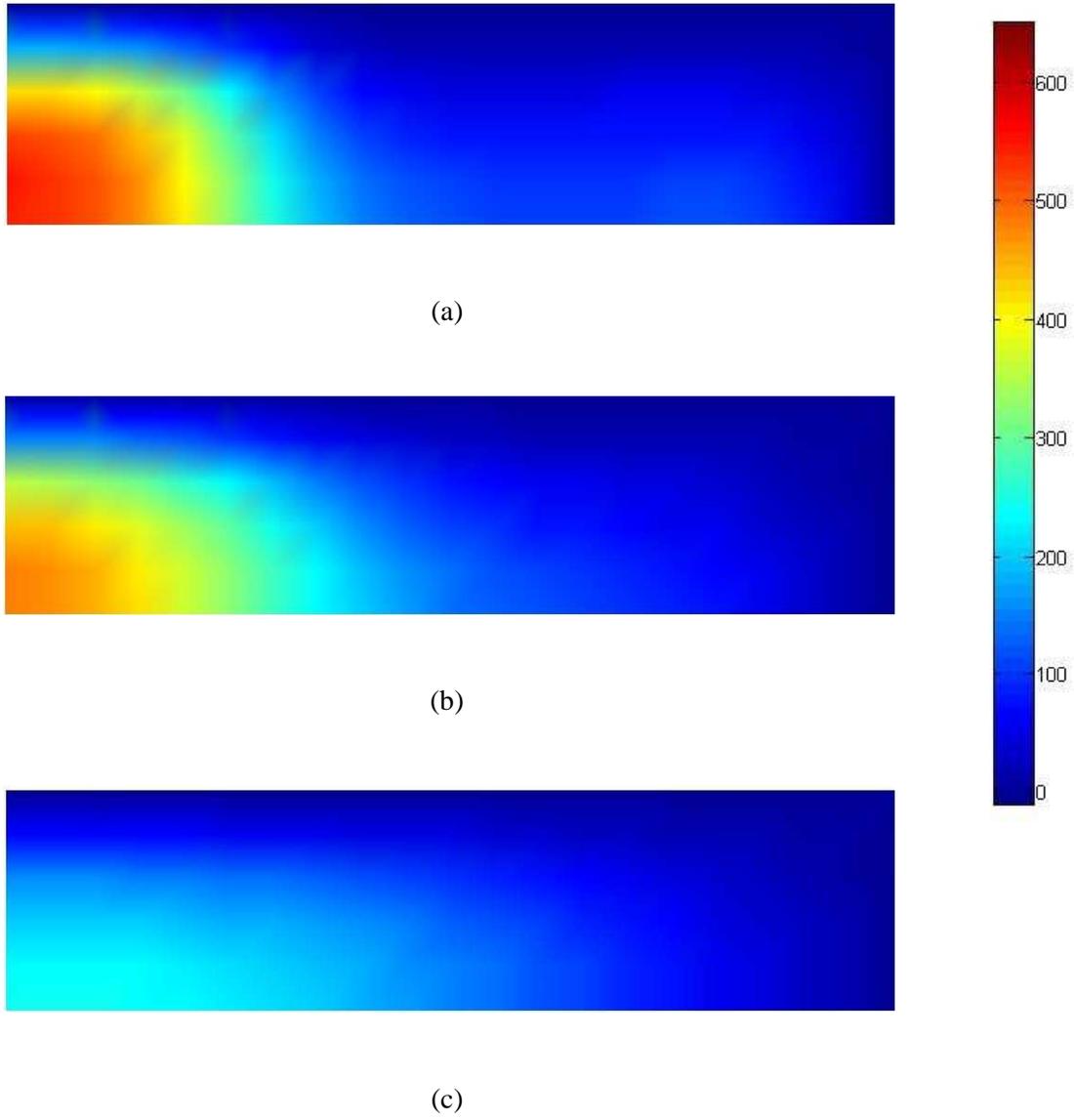
Figura 8.44. Problema Heterogêneo e Anisotrópico – detalhes da malha inicial (azul) e da malha deformada (vermelho) para $r = 5$ ($T_v = 0,10$).



Fonte: Elaborada pelo autor.

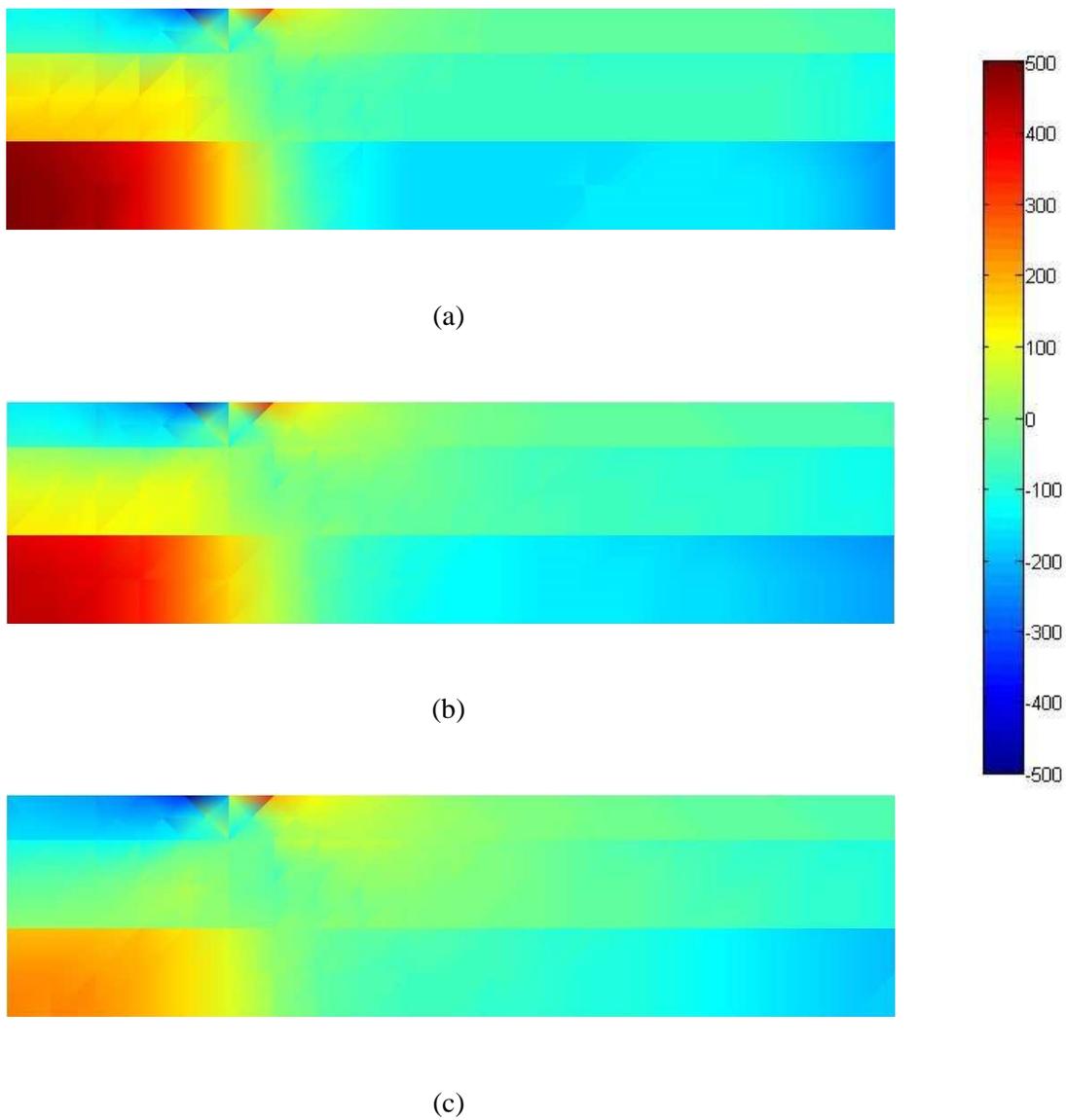
A Figura 8.45 apresenta o resultado das pressões de fluido ao longo do domínio para as razões de anisotropia utilizadas. Como pode ser observado, para um maior valor de razão de anisotropia maior a dissipação da pressão de fluido. Isso é esperado devido ao fato da permeabilidade assumir valores maiores nessa situação.

Figura 8.45. Adensamento bidimensional heterogêneo e anisotrópico - Campo de pressões, (a) $r=1$, (b) $r=5$, (c) $r=50$.



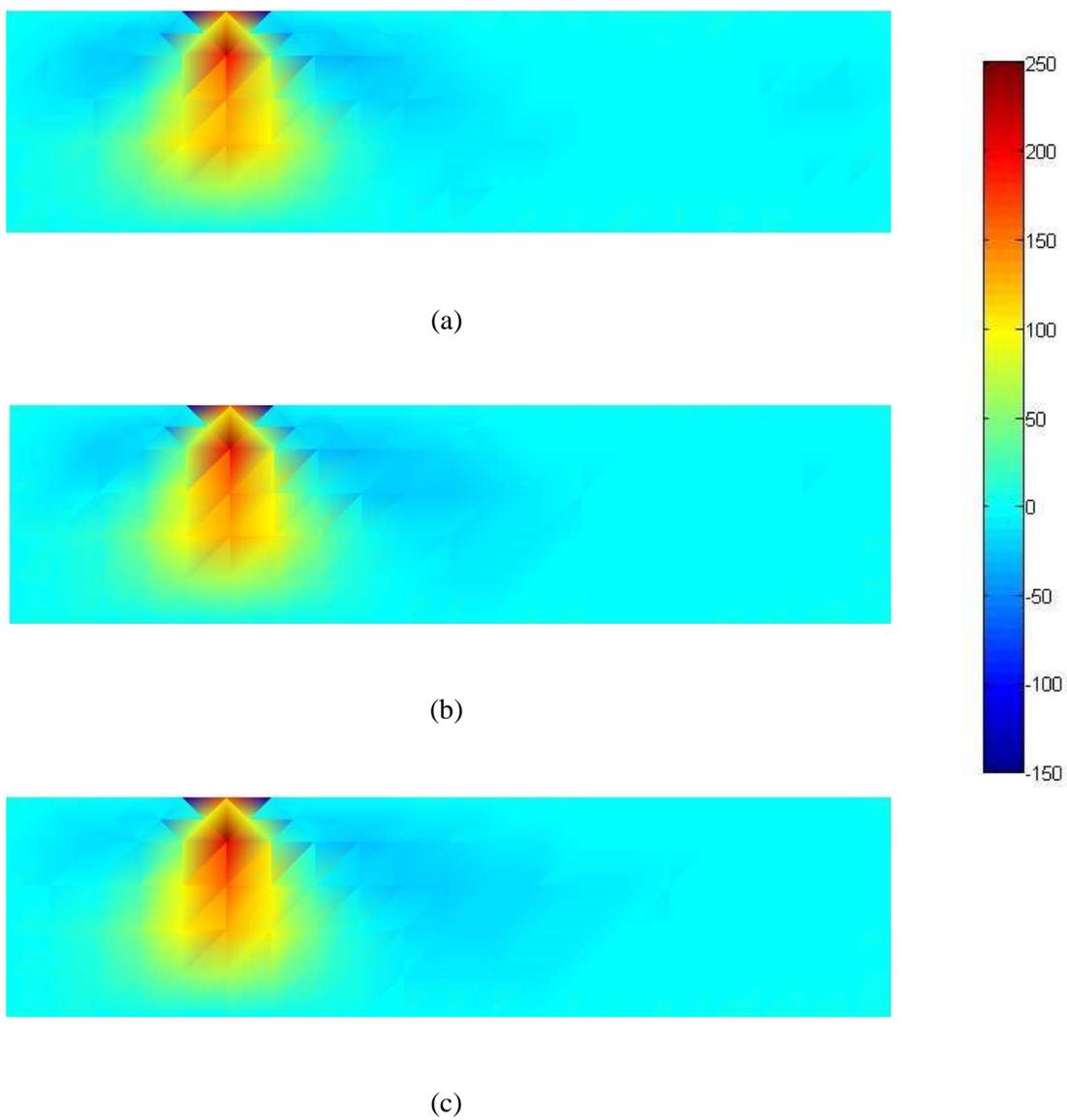
Fonte: Elaborada pelo autor.

Figura 8.46. Adensamento bidimensional heterogêneo e anisotrópico - Campo de tensões normais na direção x, (a) $r=1$, (b) $r=5$, (c) $r=50$.



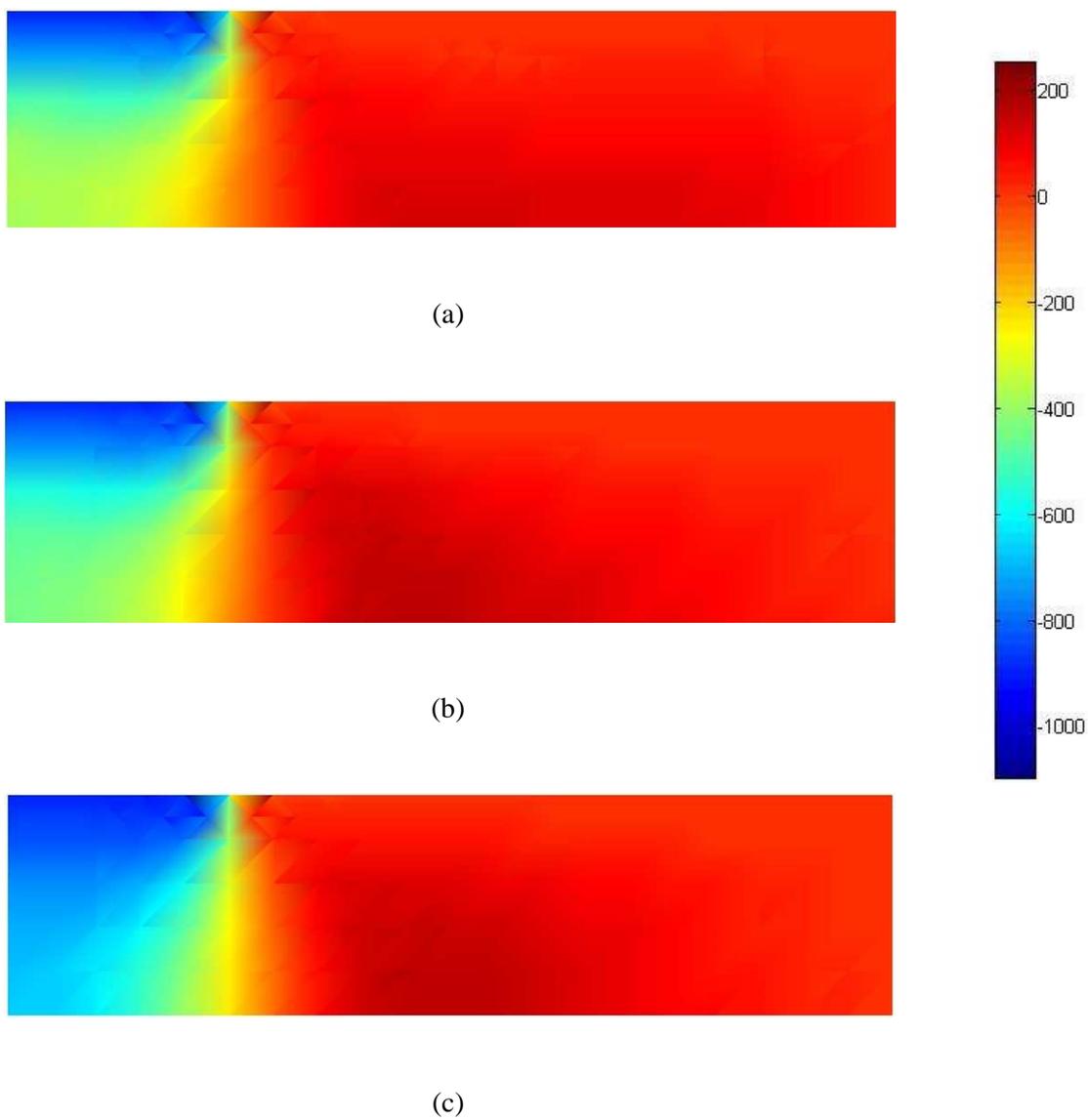
Fonte: Elaborada pelo autor.

Figura 8.47. Adensamento bidimensional heterogêneo e anisotrópico - Campo de tensões cisalhantes,
(a) $r=1$, (b) $r=5$, (c) $r=50$.



Fonte: Elaborada pelo autor.

Figura 8.48. Adensamento bidimensional heterogêneo e anisotrópico - Campo de tensões normais na direção y , (a) $r=1$, (b) $r=5$, (c) $r=50$.



Fonte: Elaborada pelo autor.

A partir da análise dos resultados é possível verificar a capacidade do código desenvolvido em simular situações normalmente encontradas na prática da engenharia. Também é possível verificar que a definição da modelagem implica em um importante aspecto da simulação. Além das aproximações inerentes às formulações numéricas tem-se que levar em conta as características de cada consideração feita, como é o caso do próprio exemplo onde há uma modelagem tridimensional, uma modelagem numérica bidimensional com um domínio finito e uma modelagem analítica com um domínio infinito.

9 CONCLUSÕES

De acordo com os resultados apresentados é possível verificar que o código computacional desenvolvido foi validado, sendo capaz de solucionar problemas poroelásticos sujeitos a restrições apresentadas ao longo do trabalho. Esse fato é observado com a reprodução, de maneira satisfatória, das soluções analíticas avaliadas.

Com o problema de Schiffman foi possível verificar a influência nos resultados obtidos em situações onde ocorrem domínios infinitos. Foi observado que é necessária a avaliação das condições de contorno impostas quando uma pequena variação na solução é significativa porém, apesar das diferenças apresentadas, e por se tratar de uma formulação aproximada, os resultados foram satisfatórios, validando o código e a formulação para esse exemplo.

Também é verificado, principalmente em uma situação mais complexa, como é o caso heterogêneo e anisotrópico, que apesar da eficácia do código e da formulação, o uso do bom senso na análise de situações reais é imprescindível. As restrições e considerações impostas à formulação numérica implementada devem ser conhecidas no processo de modelagem pois vão afetar de forma significativa os resultados. Pode ser verificado que o próprio modelo utilizado, devido à suas considerações, afeta a solução. Esse fato foi observado com bastante proeminência no exemplo heterogêneo e anisotrópico, onde a solução analítica, dentro das suas limitações, teve uma diferença significativa com as soluções do ABAQUS® e do código desenvolvido. Tal diferença é aceitável devido à sua natureza, que se mostra um tanto complexo e das diferenças entre as considerações utilizadas em cada método.

10 SUJESTÕES PARA TRABALHOS FUTUROS

Podem ser tomados como sugestões para trabalhos futuros os seguintes tópicos:

- Implementação de elementos de ordem maiores que as consideradas para esse trabalho;
- Introdução de elementos quadriláteros e com outras geometrias;
- Incorporação da plasticidade no problema mecânico;
- Discretização temporal por uma formulação do MDF com controle de passo de tempo.

REFERÊNCIAS

- AI, Zhi Yong; CAO, Guo Jun; CHENG, Yi Chong. Analytical layer-element solutions of Biot's consolidation with anisotropic permeability and incompressible fluid and solid constituents. **Applied Mathematical Modelling**. p. 4817-4829. 2012.
- AMIRI, Shahin Nayyeri; HAJALI, Masood; ESMAEILI, Asad. Consolidation of a Layer under a Strip Loading and Resting on an Impermeable Foundation. **Journal Of Earth Sciences And Geotechnical Engineering**. p. 111-125. 2014.
- BERLIOUX, Arnaud. Building models with GOCAD. **Stanford Exploration Project**. p. 1-607. 15 maio 2001.
- BIOT, Maurice Anthony. General Theory of Three-Dimensional Consolidation. **Journal Of Applied Physics**. p. 155-164. 1941.
- BIOT, Maurice Anthony. Theory of Elasticity and Consolidation for a Porous Anisotropic Media. **Journal Of Applied Physics**. Recife, p. 182-185. jan. 1955.
- BIOT, Maurice Anthony. Nonlinear and Semilinear Rheology of Porous Solid. **Journal Of Geophysical Research**. Recife, p. 4924-4937. jan. 1973.
- BIOT, Maurice Anthony. Theory of Deformation of a Porous Viscoelastic Anisotropic Solid. **Journal Of Applied Physics**. Recife, p. 459-467. jan. 1956.
- BIOT, Maurice Anthony; WILLIS, D. G.. The Elastic Coefficients of the Theory of Consolidation. **Journal Of Applied Mechanics**. p. 594-601. 1957.
- BOOKER, J. R.. The consolidation of a finite layer subject to surface loading. **International Journal Of Solids And Structures**. p. 1053-1065. 10 dez. 1973.
- BORGES, Luiz Eduardo. **Phyton para desenvolvedores**. 2. ed. Rio de Janeiro: Edição do Autor, 2010.

BROWN, G. O.. Henry Darcy and the making of a law. **Water Resources Research**, Recife, v. 38, n. 7. 2002.

CAMPOS, Marco Donisete de; SILVA, João Batista Campos. O método dos elementos finitos aplicado à simulação numérica de escoamentos de fluidos. **III Bienal da Sbm**. 2006.

CAPUTO, Homero Pinto. **Mecânica dos solos e suas aplicações**. 6. ed. Rio de Janeiro: Ltc, 1998.

CARMO, L. V.; MARQUES, D. M.; THOMPSON, R. L.. Modelos para escoamentos de fluidos não newtonianos em meios porosos. **Vi Congresso Nacional de Engenharia Mecânica**, Campina Grande, p.1-2, ago. 2010.

CARSLAW, H. S.; JAEGER., J. C. **Operational Methods in Applied Mathematics**. London: Oxford University Press, 1948.

CAVALCANTI, Maria do Carmo Reis. **Análise por Elementos de Contorno de Meios Porro-elásticos Saturados pela Teoria de Biot**. 2006. 1 v. Tese (Doutorado) - Curso de Engenharia Civil, Coppe, Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2006.

CLOUGH, Ray W.. The finite element method in plane stress analysis. **2nd Conference On Electronic Computation**: sponsored by the Committee on Electronic Computation, Structural Division, Pittsburgh, p.345-378, set. 1960.

CMNE. **GiD**: Reference Manual. Recife, 2016.

COMPUTER MODELLING GROUP LTD. **Training Course Catalogue**. Cmg, 2013.

COUSSY, Olivier. **Poromechanics**. Marne-la-vallée: John Wiley & Sons, 2004.

CRYER, C. W.. A Comparison od the Three-Dimensional Consolidation Theories of Biot and Terzaghi. **The Quarterly Journal Of Mechanics And Applied Mathematics**, Recife, v. 16, n. 1, p.401-412, jan. 1963.

COURANT, Richard. Variational Methods for the Solution of Problems of Equilibrium and Vibrations. **Bulletin Of The American Mathematical Society**, v. 49, p.1-23, 1943.

ECKEL, Bruce. **Guia do Usuário em C++**. Recife: Makron Books, 1991.

FERRER, Harry et al. **Fortran Estruturado**. Recife: Guanabara Koogan, 1992.

FARENZENA, B. A.; HAUSER, E. B.. Método de Rayleigh-Ritz - Fundamentos Matemáticos e Aplicações. **X Salão de Iniciação Científica Pucrs**, Porto Alegre, v. 1, n. 1, p.1-1, jan. 2009.

FERREIRA, Francisco Henrique. **Uma Implementação Numérica para a Solução de Problemas de Poroelasticidade**. 2010. 1 v. Dissertação (Mestrado) - Curso de Engenharia Civil, Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, 2010.

GEUZAIN, Christophe; REMACLE, Jean-françois. **Gmsh Reference Manual: The documentation for Gmsh 3.0**. Recife: Copyright, 2017.

GIBSON, R. E.; MCNAMEE, J.. Plane strain and axially symmetric problems of the consolidation of a semi-infinite clastic stratum. **The Quarterly Journal Of Mechanics And Applied Mathematics**. p. 210-227. 1960.

GIBSON, R. E.; MCNAMEE, J.. The consolidation settlement of a half-space under a load uniformly distributed over a rectangular area. **Proc. 4th Icsmf**. p. 297-299. 1957.

GIBSON, R. E.; MCNAMEE, J.. Displacement Function and linear transforms applied to diffusion through porous elastic media. **The Quarterly Journal Of Mechanics And Applied Mathematics**. Recife, p. 98-111. 1 jan. 1960.

JIA-DONG, Zheng et al. The application of the finite element method to solving Biot's consolidation equation. **Applied Mathematics And Mechanics**. p. 857-871. 1982.

LANDIM, Paulo M. Barbosa; MONTEIRO, Rubens Caldeira; CORSI, Alessandra Cristina. **INTRODUÇÃO À CONFECCÃO DE MAPAS PELO SOFTWARE SURFER**. Rio Claro: Unesp/rio Claro, 2002.

Llnl. **VisIt User's Manual**. Lawrence: Department Of Energy By Lawrence Livermore National Laboratory, 2005.

LÓPEZ, Camilo Andrés Villabona. **Análise pelo método dos elementos finitos da distribuição de tensões em componentes protéticos posicionados em implantes com conexão cone-morse na região anterior da maxila.** 2012. 119 f. Dissertação (Mestrado) - Curso de Odontologia, Universidade de São Paulo, Ribeirão Preto, 2012.

LOTTI, Raquel S. et al. Aplicabilidade científica do método dos elementos finitos. **A Dental Press Ortodon Ortop Facil**, Maringá, v. 11, n. 2, p.35-43, abr. 2006.

MANDEL, J.. Consolidation des Sols (Etude Mathematique). **Geotechnique.** p. 287-299. 1953.

PASTALLE, SkbastÜ. Olivella. **NONISOTHERMAL MULTIPHASE FLOW OF BRINE AND GAS THROUGH SALINE MEDIA.** 1995. 123 f. Tese (Doutorado) - Curso de Enginyers de Camins, Universitat Politecnica de Catalunya, Barcelona, 1995.

PENA, Daniel Ricardo Izquierdo. **Método de diferenças finitas generalizadas por mínimos quadrados.** 2003. 1 v. Dissertação (Mestrado) - Curso de Ciências de Computação e Matemática Computacional, Universidade de São Paulo, São Carlos, 2003.

PHILLIPS, Phillip Joseph. **Finite Element Methods in Linear Poroeasticity: Theoretical and Computational Results.** 2005. 1 v. Dissertação (Mestrado), The University Of Texas At Austin, 2005.

PIZZOL, Alessandro dal. **Uma metodologia unificada empregando o método dos volumes finitos para a solução acoplada do escoamento e da geomecânica em meios porosos..** Dissertação (Mestrado) - Curso de Engenharia Mecânica, Universidade Federal de Santa Catarina, Florianópolis, 2014.

RIBEIRO, Gustavo Gondran. **Volumes finitos baseado em elementos para problemas de poroelasticidade.** 2016. 196 f. Tese (Doutorado) - Curso de Engenharia Mecânica, Universidade Federal de Santa Catarina, Florianópolis, 2016.

SANTOS, João Paulo Lima. **Estratégias adaptativas para formulações mistas em elementos finitos aplicadas a modelos reológicos viscoelásticos e modelos lineares incompressíveis.** 2011. 1 v. Tese (Doutorado) - Curso de Engenharia Civil, Coppe, Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2011.

SCHIFFMAN, L. R.; CHEN, A. T.; JORDAN, J. C.. An analyses of Consolidation Theories. **Journal Of The Soil Mechanics And Foundation Division Asce.** p. 285-612. 1969.

SCHLUMBERGER. **ECLIPSE 100.** Schlumberger Information Solutions, 2007.

SINGH, Sarva Jit; RANI, Sunita; KUMAR, Raman. Consolidation of a Poroelastic Half-Space. **Mathematical Journal Of Interdisciplinary Sciences.** p. 1-18. set. 2013.

SIQUEIRA, César Augusto Monteiro. **Um sistema orientado por objetos para análise numérica da poroelasticidade acoplada pela técnica dos elementos finitos.** 1995. 132 f. Dissertação (Mestrado) - Curso de Engenharia de Petróleo, Departamento de Engenharia de Petróleo, Universidade Estadual de Campinas, Campinas, 1995.

SLOAN, Scott W.; ABBO, Andrew J.. Biot consolidation analysis with automatic time stepping and error control part 1: theory and implementation. **International Journal For Numerical And Analytical Methods In Geomechanics.** p. 467-492. 1999.

SLOAN, Scott W.; ABBO, Andrew J.. Biot consolidation analysis with automatic time stepping and error control part 2: applications. **International Journal For Numerical And Analytical Methods In Geomechanics.** p. 493-529. 1999.

TERZAGHI, K.. Dir Berenchnung der Durchlassigkeitsziffer des Tones aus dem Verlaug Hydro-dynamischen spannugsercheinungen. **Mathematish Naturwissenschaftliche Klasse.** p. 125-138. 1923.

TERZAGHI, K.. Theoretical Soil Mechanics. **John Wiley & Sons.** Recife, p. 1-2. jan. 1943.

TERZAGHI, K.; FROHLICH, K. O.. Theorie der Setzung Ven Tonschiten. **Franz Deuticke.** Recife, p. 1-2. jan. 1936.

THE MATH INC. **MATLAB:** User's Guide. Recife, 1992.

WANG, Jianguo; FANG, Shisheng. State space solution of non-axisymmetric Biot consolidation problem for multilayered porous media. **International Journal Of Engineering Science,** v. 41, p.1799-1813. 2003.

ZHANG, Chao; ZARROUK, Sadiq J.; ARCHER, Rosalind. Development of a fully coupled flow-geomechanics simulator for flow in saturated porous media. **ICCM 2015**. Auckland, New Zealand, 2015.

APÊNDICE A

Esse apêndice apresenta um esquema do código computacional desenvolvido e suas funções, listadas abaixo.

As funções de cada rotina são apresentadas abaixo:

- Main_MefM_Biot.m

Script principal, onde ocorre a entrada dos dados das simulações e o início do processamento.

- MefBiot.m

Essa função organiza a relação entre os dados do pré-processamento, processamento e pós-processamento.

- PreprocessoBiot.m

Organiza a relação entre o pré-processamento da geometria e da malha.

- PreprocessorGeo.m

Faz a leitura do arquivo .geo e constrói matrizes com as informações desse arquivo que serão utilizadas no processamento.

- PreprocessorMsh.m

Faz a leitura do arquivo .msh e constrói matrizes com as informações desse arquivo que serão utilizadas no processamento.

- ProcessoBiot.m

Recebe os dados do pré-processamento e realiza todas as operações do processamento juntamente com suas funções auxiliares.

- GIPressure.m

Essa função organiza a numeração dos graus de liberdade mecânicos e hidráulicos.

- MontMatSist.m

Realiza a montagem das matrizes global do sistema, construída a partir das matrizes locais.

- Calc_Matrizes_locais.m

Constrói as matrizes locais para cada elemento e é acionada dentro de um loop na função MontMatSist.m

- Tens_Gauss_Point.m

Realiza a integração numérica para a construção de cada matriz local.

- ConstTermInd.m

Constrói o termo independente do sistema global de equações

- CondContornoConsolidation.m

São introduzidas as condições de contorno no vetor e matriz global do sistema

- ElasticConsolidation.m

Recebe os dados necessários para a solução temporal do sistema de equações

- ECTThetaMethod.m

Realiza o avanço no tempo do sistema de equações pelo Método θ

- PosprocessoBiot.m

Recebe os resultados do processamento e organiza para as funções específicas para o processamento das variáveis; salva arquivos de texto com as matrizes e vetores utilizadas no processamento.

- PosImprimir.m

Organiza os dados para o pós-processamento das variáveis mecânicas e hidráulicas

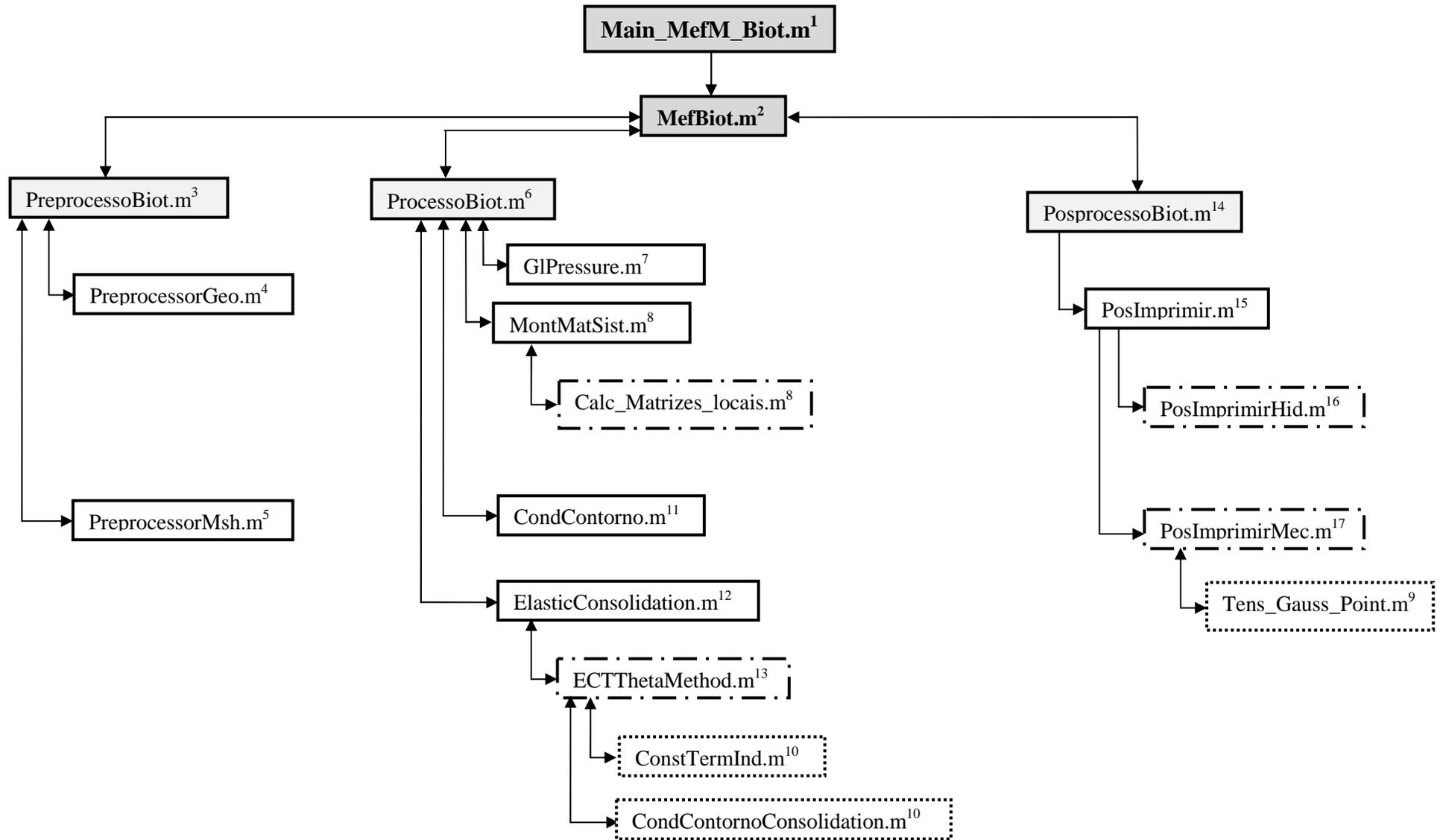
- PosImprimirHid.m

Realiza o pós processamento das variáveis hidráulicas (construção de imagens).

- PosImprimirMec.m

Realiza o pós processamento das variáveis mecânicas (construção de imagens).

Abaixo é apresentado um esquema dos rotinas e como elas se relacionam. O sobrescrito apresentado em cada rotina é referente à sequência de processamento que segue cada problema analisado.



As rotinas são apresentadas abaixo:

Main_MefM_Biot.m

```

%-----%
%-----%
%
%          PROGRAMA DE PÓS GRADUAÇÃO EM ENGENHARIA CIVIL - UFPE
%
%-----%
%-----%

%-----%
%
%          LMCG - UFPE
%
%-----%
%
%  ÁREA DE CONCENTRAÇÃO: GERENCIAMENTO E SIMULAÇÃO DE RESERVATÓRIOS DE
%  PETRÓLEO
%  ALUNO: LUCÍOLO VICTOR MAGALHÃES E SILVA
%  PROF. ORIENTADOR: IVALDO DÁRIO PONTES SILVA FILHO
%
%-----%

%-----%
%
%          ROTINA
%
%-----%
%
%  TÍTULO: MÉTODO DOS ELEMENTOS FINITOS MISTOS PARA A SOLUÇÃO DO PROBLEMA
%  DE ACOMPLAMENTO GEOMECÂNICO DADO PELA TEORIA DE BOIT
%  CRIAÇÃO: 20-08-2015
%  ÚLTIMA MODIFICAÇÃO: 16-10-2016
%  VERSÃO V1.1
%
%-----%

clc
clear all

%-----%
%
%          ENTRADA DE DADOS
%
%-----%
%
%  OBS: Nessa seção serão configuradas as características do problema que
%  será avaliado.
%
%-----%

%-----%
%
%  /\ O nome dos arquivos tem que ser salvos com as devidas extensões
%  por vezes são salvos como arquivos .txt e isso precisa ser verificado!/\
%
%-----%

% filenamegeo
%
% Variável com o nome do arquivo '.geo' referente a geometria do problema.

filenamegeo = 'Sapata_Geo4.geo';

% filenamemsh
%
% Variável com o nome do arquivo '.msh' referente a malha do problema.

filenamemsh = 'Sapata_Geo4_Malhal.msh';

```

```

%-----%
%-----%
%                               %
%                               %
%-----%
%-----%

%-----%
%                               %
%                               %
%-----%

% TipProb
%
% Flag para o tipo de problema a ser resolvido.
% A formatação dessa variável é a seguinte:
%
%
% TipProb = 1 - Problema Mecânico Elástico Linear.
% TipProb = 2 - Problema Hidráulico.
% TipProb = 3 - Problema Acoplado(Hidrogeomecânico) Elástico.

TipProb = 3;

% Units
%
% Esse flag é relacionado à utilização ou não das unidades sugeridas para
% rodar o caso.
%
% Formatação:
%
% Units = 1 - As unidade utilizadas são as seguintes:
%
% Módulo de Elasticidade - Pa (Pascal).
% Permeabilidade - cm/s (Centímetro por segundo)
% Tempo - s (Segundos).

% CONFIGURAÇÕES PARA O PROBLEMA DE ANDESAMENTO

% MetConc
%
% Variável que indica o método de solução do problema de consolidação.
%
% Formatação:
%
% MetCon = 'Theta' - Será utilizado o Método Theta.
% MetCon = 'T&G' - Será utilizado o método de Thomas and Gladwell.

MetCon = 'Theta';

% Nmaxint
%
% Variável que fornece o número máximo de iterações.

Nmaxint = 10000+100;

% DeltaT
%
% Variável que fornece o interaço de tempo para solução da
% integração

```

```

DeltaT = 1;

% Theta
%
% Valor de Theta para o método Theta

Theta = 1;

% iaval
%
% Iteração avaliada para pré-processamento ao longo do eixo y.

iaval = [10 100 1000]*10 + 100;

% Nint0
%
% Essa variável contém o número de iterações necessárias para que seja
% atingido o valor total da força aplicada.

Nint0 = 100;

%-----%
%                               %
%                               %
%-----%

% fator
%
% Essa variável define o fator relativo à impressão dos deslocamentos
% nodais

fator = 50000;

% Imprimir_Tens
%
% Essa variável indica as tensões que serão impressas:
% 1 - imprime
% 0 - não imprime
%
% Imprimir_Tens(1) - Tensões em X
% Imprimir_Tens(2) - Tensões em Y
% Imprimir_Tens(3) - Tensões em XY

Imprimir_Tens = [ 1 1 1 ];

% Imprimir_Desloc
%
% 1 - Imprime os deslocamentos
% 0 - Não imprime os deslocamentos

Imprimir_Desloc = 1;

% Imprimir_Press
%
% 1 - Imprime as pressões
% 0 - Não imprime as pressões

```

```

Imprimir_Pres = 1;

%-----%
% Impressão das pressões ao longo de uma superfície.          %
%-----%

% Imprimir_Px
%
% 1 - Imprime as pressões ao longo de x
% 0 - Não imprime as pressões ao longo de x

Imprimir_Px = 1;

% As superfícies que serão plotadas, associadas às pressões serão entre
% xpinf e xpsup

xpinf = 0-0.01;

xpsup = 0.0+0.01;

% Imprimir_Py
%
% 1 - Imprime as pressões ao longo de y
% 0 - Não imprime as pressões ao longo de y

Imprimir_Py = 1;

% As superfícies que serão plotadas, associadas às pressões serão entre
% ypinf e ypsup

ypinf = 0-0.01;

ypsup = +0.01;

%-----%
% Plotagem dos deslocamentos ao longo de uma superfície.      %
%-----%

% As variáveis abaixo definem a plotagem dos deslocamentos ao longo de
% superfícies definidas por xinf e xsup ou yinf e ysup

% Imprime_Desloc_x
%
% 1 - Imprime os deslocamentos ao longo das superfície x.
% 0 - Não mprime os deslocamentos ao longo das superfície x.

Imprimir_Desloc_x = 1;

% xinf, xsup
%
% A superfície associada ao Imprime_Desloc_x é definida como sendo os
% pontos ente xinf e xsup

xinf = 0;

xsup = 10;

% Imprime_Desloc_y

```

```

%
% 1 - Imprime os deslocamentos ao longo das superfícies y.
% 0 - Não imprime os deslocamentos ao longo das superfícies y.

Imprimir_Desloc_y = 1;

% yinf, ysup
%
% A superfície associada ao Imprime_Desloc_y é definida como sendo os
% pontos entre xinf e xsup

ysup = 0+0.01;

yinf = 0-0.01;

%-----%
%                               Características dos planos                               %
%-----%
%
% Entrada de dados que define as características associadas a regiões da %
% malha. Essas características são: %
% - Módulo de Elasticidade %
% - Coeficiente de Poisson %
% - Força de Corpo %
%
% DadosPlan - Variável que identifica as características associadas a %
% região do plano. %
%
% Formatação: %
%
% DadosPlan(i,:) = [ Ex Ey vxy kx ky kxy bx by p0 yw ], onde: %
%
% 'Ex' é o Módulo de Elasticidade para a direção x associado ao grupo %
% físico. %
% 'Ey' é o Módulo de Elasticidade para a direção y associado ao grupo %
% físico. %
% 'vxy' é o Coeficiente de Poisson associado ao grupo físico. %
% 'kxx', 'kyy' e 'kxy' são os valores de permeabilidades. %
% 'bx' e 'by' são, respectivamente, as componentes x e y associadas ao %
% grupo físico. %
% 'p0' é a poropressão inicial para o grupo físico avaliado. %
% 'yw' é o peso específico do material. %
%
% Obs: os grupos físicos precisam ser preenchidos de acordo com a ordem %
% que foram estabelecidos. %
%-----%
DadosPlan = [
%
% -----%
% | Ex | Ey | vxy | kx | ky | kxy | bx | by | p0 | yw |
% -----%
%
% 10^8 , 10^8 , 0.0 10^-10, 10^-10, 0 , 0 , 0 , 0 ,1000
%
];
%-----%
%                               Condições de contorno nos pontos                               %
%-----%
%
% Entrada das condições de contorno para os pontos da malha, essas %

```

```

% entradas são:
% - Força concentrada no nó
% - Deslocamento prescrito
% - Restrição no nó
%
% CCPoint - Variável que apresenta as condições de contorno associada ao
% ponto.
%
% Formatação:
%
% CCPoint(i,:) = [ Fx Fy ux uy Kx Ky v p y], onde:
%
% 'Fx' e 'Fy' são, respectivamente, as forças concentradas nas direções x
% e y do grupo físico.
% 'ux' e 'uy' são, respectivamente, os deslocamentos prescritos nas
% direções x e y do grupo físico.
% 'Kx' e 'Ky' são, respectivamente, os flags associados às restrições dos
% graus de liberdade do ponto. Se o flag for igual a 1 o grau de liberdade
% está restringido, se for outro valor ele está livre.
% 'v' é a vazão concentrada diretamente no nó
% 'p' é a pressão
% 'y' é o flag associado à condição de contorno de fluxo, se esse flag for
% igual a a 1 a pressão prescrita com valor igual a p
%-----%
% Obs: os grupos físicos precisam ser preenchidos de acordo com a ordem
% que foram estabelecidos.
%-----%
% /\ OBS: usualmente, para apoios rígidos, são introduzidos valores de
% constante de molas com valores elevados, ex: 10^5, 10^6, 10^7.
%-----%
% /\ OBS: para a imposição de condição de contorno de pressão é necessá-
% rio impor um valor de gama alto, ex: 10^5, 10^6, 10^7.
%-----%

CCPoint = [

% -----%
% | Fx | Fy | ux | uy | Kx | Ky | v | p | y |
% -----%

    0 , 0 , 0 , 0 , 1 , 1 , 0 , 0 , 0;

];

%-----%
%
%                               Condições de contorno nos linhas
%-----%
%
% Entrada das condições de contorno para as linhas da geometria, essas
% entradas são:
% - Força distribuída na aresta
% - Deslocamento prescrito
% - Restrição na linha
%
% CCLine - Variável que apresenta as condições de contorno associada à
% linha.
%
% Formatação:
%
% CCLine(i,:) = [ Fx Fy Fn Ft ux uy Kx Ky v p], onde:

```

```

%
% 'Fx' e 'Fy' são, respectivamente, as forças districuídas nas direções x %
% e y do grupo físico. %
% 'Fn' e 'Ft' são, respectivamente, as forças distribuídas nas direções %
% normal e tangencial a barra associada ao grupo físico avaliado. %
% 'ux' e 'uy' são, respectivamente, os deslocamentos prescritos nas %
% direções x e y do grupo físico. %
% 'Kx' e 'Ky' são flags que apresentam a restrição ou não do apoio. %
% Se Kx ou Ky = 1 o grau de liberdade é restringido, se não é livre. %
% 'v' é a vazão para esse grupo físico. %
% 'p' é a pressão para esse grupo físico. %
% 'y' é o flag associado à condição de contorno de fluxo onde: y = 1 a %
% pressão é prescrita. %
%-----%
% Obs: os grupos físicos precisam ser preenchidos de acordo com a ordem %
% que foram estabelecidos. %
%-----%
% /\ OBS: usualmente, para apoios rígidos, são introduzidos valores de %
% constante de molas com valores elevados, ex: 10^5, 10^6, 10^7. %
%-----%
% /\ OBS: para a imposição de condição de contorno de pressão é necessá- %
% rio impor um valor de gama alto, ex: 10^5, 10^6, 10^7. %
%-----%
% Para as arestas os valores são atribuídos aos nós que formam as linhas %
%-----%

CCLine = [

% -----%
% | Fx | Fy | Fn | Ft | ux | uy | Kx | Ky | v | p | y | %
% -----%

    0 , 1000 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 1
    0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 1
    0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0
    0 , 0 , 0 , 0 , 0 , 0 , 0 , 1 , 0 , 0 , 0 , 0
    0 , 0 , 0 , 0 , 0 , 0 , 1 , 0 , 0 , 0 , 0 , 0

];

MefBio( filenamegeo, filenameemsh, DadosPlan, CCPoint, CCLine, TipProb, MetCon...
, Nmaxint, DeltaT, Theta, iaval, Imprimir_Desloc, Imprimir_Tens, fator, ...
Imprimir_Pres, Nint0, Imprimir_Px, Imprimir_Py, Imprimir_Desloc_x, ...
Imprimir_Desloc_y, xsup, xinf, ysup, yinf, xpinf, xpsup, ypinf, ypsup)

%-----%
% REFERÊNCIAS BIBLIOGRÁFICAS %
%-----%

% >>REFERÊNCIA 1
% Matriz de rigidez do problema mecânico pelo Método Elementos Finitos
% Mistos.
% Pesquisador: João Paulo Lima Santos
% Título: Estratégias Adaptativas Para Formulações Mistas em Elementos
% Finitos.
% Aplicada a Modelos Reológicos Viscoelásticos e Modelos Lineares
% Incompressíveis.
% Tipo: Tese de Doutorado
% Ano: 2011
% Instituição: COPPE

```

```

% >>REFERÊNCIA 2
% Matriz de rigidez do problema mecânico LST.
% Pesquisador: Lucíolo Victor Magalhães e Silva
% Tipo: TCC
% Ano: 2015
% Instituição: UFPE

% >>REFERÊNCIA 3
% Matrizes de rigidez para o problema mecânico pelo Método dos Elementos.
% Finitos Mistos
% Pesquisador: Zienkiewicz e Taylor
% Título: The Finite Element Method
% Tipo: livro
% Ano: 2005
% Pág: 285

% >>REFERÊNCIA 4
% Funções de forma para a pressão, elemento triangular com três nós.
% Arquivo MEF >> Pasta MEF
% Nome arquivo: AE2_Elasticidade_2D

% >>REFERÊNCIA 5
% Arquivo de forças nodais equivalentes >> Pasta Referências Código.

% >>REFERÊNCIA 6
% Arquivo de Integração Numérica >> Pasta Referência Códigos.

```

MefBiot.m

```

%-----%
%
%                               ROTINA
%-----%
% TÍTULO: MÉTODO DOS ELEMENTOS FINITOS MISTOS PARA A SOLUÇÃO DO PROBLEMA
% DE ACOMPLAMENTO GEOMECÂNICO DADO PELA TEORIA DE BIOT
%-----%

%-----%
%                               SUB-ROTINA
%-----%
% TÍTULO: MefBiot
% DESCRIÇÃO: Rotina de processamento para a solução do problema mecânico
% utilizando a formulação do Método dos Elementos Finitos Mistos
%-----%
% Programador: Lucíolo Victor (DECIVIL-UFPE)
% Orientador: Prof. Dr. Ivaldo Dário (DECIVIL - UFPE)
% Data de criação: 11-09-2015
% Última modificação: 18-10-2016
%-----%

function MefBiof(filenamegeo,filenamemsh,DadosPlan,CCPoint,CCLine,...
    TipProb,MetCon,Nmaxint,DeltaT,Theta,iaval,Imprimir_Desloc,...
    Imprimir_Tens,fator,Imprimir_Pres,Nint0,Imprimir_Px,Imprimir_Py,...
    Imprimir_Desloc_x,Imprimir_Desloc_y,xsup,xinf,ysup,yinf,xpinf,xpsup....
    ,ypinf,ypsup)
tic
%-----%
% CÓDIGOS DE PREPROCESSAMENTO DE DADOS - ARQUIVO .geo E ARQUIVO .msh
%-----%

```

```

[ PropPoint, PropLines, PropElem, Coord, CaracPlan , CaracPoint,...
  CaracLine] = PreprocessoBiot( filenamegeo, filenamemsh, ...
  DadosPlan, CCPoint, CCLine );

%-----%
% CÓDIGOS DE PROGRESSAMENTO                                     %
%-----%

[X,Glp,K,Ce] = ProcessoBiot( PropPoint, PropLines, PropElem,...
  Coord, CaracPlan, CaracPoint, CaracLine,TipProb, MetCon, ...
  Nmaxint,DeltaT,Theta,Nint0);
toc

%-----%
% CÓDIGOS DE PÓS-PROGRESSAMENTO                                 %
%-----%

PosProcessoBiot(K,Ce,iaval,X,Coord,Glp,PropElem,CaracPlan,...
  Imprimir_Desloc,Imprimir_Tens,fator,Imprimir_Pres,Imprimir_Px,...
  Imprimir_Py,Imprimir_Desloc_x,Imprimir_Desloc_y,xsup,xinf,ysup,yinf...
  ,xpinf,xpsup,ypinf,ypsup,Nmaxint,DeltaT,Theta,Nint0)

% Fim do processo

disp('Fim do processo')

end

```

PreprocessoBiot.m

```

%-----%
%                                     ROTINA                                     %
%-----%
% TÍTULO: MÉTODO DOS ELEMENTOS FINITOS MISTOS PARA A SOLUÇÃO DO PROBLEMA %
% DE ACOMPLAMENTO GEOMECÂNICO DADO PELA TEORIA DE BIOT                       %
%-----%

%-----%
%                                     SUB-ROTINA                                 %
%-----%
% TÍTULO: PreprocessoBiot                                                         %
% DESCRIÇÃO: Preprocessamento de dados dos arquivos .msh,.geo e da          %
% entrada de dados para o problema mecânico.                                     %
%-----%
% Programador: Lucíolo Victor (DECIVIL-UFPE)                                     %
% Orientador: Prof. Dr. Ivaldo Dário (DECIVIL - UFPE)                         %
% Data de criação: 29-10-2015                                                    %
% Última modificação: 29-10-2015                                               %
%-----%

function [ PropPoint, PropLines, PropElem, Coord, CaracPlan...
  , CaracPoint, CaracLine] = PreprocessoBiot( filenamegeo,...
  filenamemsh,DadosPlan, CCPoint, CCLine )

```

```

[CoordBound, LineBound, PhysicalPoint, PhysicalLine, PhysicalSurface...
, Plans] = PreprocessorGeo(filenamegeo)

[PropPoint, PropLines, PropElem, Coord] = PreprocessorMsh(filenamemsh);

%-----%
% Status do código %
%-----%

disp(' |-----| ')
disp(' | ARQUIVOS DE ENTRADA DE DADOS | ')
disp(' |-----| ')

disp('Arquivo .geo =')
disp(' filenamegeo)
disp('%-----%')

disp('Arquivo .msh =')
disp(' filenamemsh)
disp('%-----%')

% Os comandos de repetição abaixo é realizado para que a indexação da
% identificação do grupo físico seja feito pelo índice e não há a
% necessidade de fazer uma busca do vetor de grupos físicos para atribuir a
% superfície seus parâmetros físicos.

% CaracPlan
%
% É uma variável que contém a mesma formatação que DadosPlan porém nas
% linhas associadas à numeração do grupo físico que está sendo avaliado.
% Obs: Esse comando só é ativado se existir grupos físicos para
% superfícies.

if PhysicalSurface ~= 0

for i=1:size(PhysicalSurface,2)

CaracPlan(PhysicalSurface(i,:),:) = DadosPlan(i,:);

end

else

CaracPlan = 0;

end

% CaracPoint
%
% É uma variável que contém a mesma formatação que DadosPoint porém nas
% linhas associadas à numeração do grupo físico que está sendo avaliado.
% Obs: Esse grupo físico só é ativado se eistir grupos físicos para pontos.

if PhysicalPoint ~= 0

for i=1:size(PhysicalPoint,2)

```

```

CaracPoint(PhysicalPoint(i),:) = CCPoint(i,:);

end

else

CaracPoint = 0;

end

% CaracLine
%
% é uma variável que contém a mesma formatação que DadosLine porém nas
% linhas associadas à numeração do grupo físico que está sendo avaliado.
% Obs: Esse grupo físico só é ativado se existir grupos físicos para
% linhas.

if PhysicalLine ~= 0

for i=1:size(PhysicalLine,2)

CaracLine(PhysicalLine(i),:) = CCLine(i,:);

end

else

CaracLine = 0;

end

end

```

PreprocessorGeo.m

```

%-----%
%
%                      ROTINA
%
%-----%
% TÍTULO: MÉTODO DOS ELEMENTOS FINITOS MISTOS PARA A SOLUÇÃO DO PROBLEMA
% DE ACOMPLAMENTO GEOMECÂNICO DADO PELA TEORIA DE BIOT
%
%-----%

%-----%
%
%                      SUB-ROTINA
%
%-----%
% TÍTULO: Preprocessorgeo
% DESCRIÇÃO: Rotina de preprocessamento utilizada para a leitura de dados
% para arquivos .geo de modelos feitos no Gmsh
%
%-----%
% Programador: Lucíolo Victor (DECIVIL-UFPE)
% Orientador: Prof. Dr. Ivaldo Dário (DECIVIL - UFPE)
% Data de criação: 28-08-2015
% Última modificação: 29-10-2015
%-----%

```

```

function [CoordBound, LineBound, PhysicalPoint, PhysicalLine, PhysicalSur-
face...
, Plans] = PreprocessorGeo(filename)

%-----%
%           Variáveis construídas nessa função           %
%-----%

% CoordBound
%
% Matriz contendo as coordenadas x e y dos elementos, essa matriz será
% preenchida abaixo.

% LineBound
%
% Essa matriz contém a indicação dos nós que estão representados em cada
% linha do contorno.

% PhysicalPoint
%
% Esse vetor consiste na identificação dos grupos físicos dos pontos na
% sequência em que eles foram atribuídos na modelagem.

% PhysicalLine
%
% Esse vetor consiste na identificação dos grupos físicos das linhas na
% sequência em que eles foram atribuídos na modelagem.

% PhysicalSurface
%
% Esse vetor consiste na identificação dos grupos físicos das superfícies
% na sequência em que eles foram atribuídos na modelagem.

% Plans
%
% Esse vetor consiste na identificação da numeração dos planos criados na
% modelagem

% fid
%
% Identificador do arquivo referente a 'fid'.

fid = fopen(filename);

% flagpsurface
%
% Flag relacionado à colocação ou não do grupo físico de superfície

flagpsurface = 0;

% flagppoint1
%
% Flag associado ao grupo físico de superfícies (existencia ou não desse
% tipo de grupo físico).

flagppoint1 = 0;

% flagplines1
%
```

```

% Flag associado ao grupo físico de linhas (existencia ou não desse tipo de
% grupo físico).

flagplines1 = 0;

% L1
%
% Variável que lê a primeira linha do arquivo .geo -> clc

L1 = fscanf(fid,'%s %s %s',3);

% N1
%
% Variável igual a primeira string das linhas, abaixo essa variável está
% sendo inicializada para a leitura das propriedades dos primeiros pontos
% da malha

N1 = fscanf(fid,'%s',1);

% AuxPoi
%
% Essa variável consiste em um vetor que determina as coordenadas dos
% pontos da malha e contém um flag associado ao respectivo ponto.
%
% Formatação AuxPoi = [ Coorx, Coordy, Coordz, Flag ]
% onde Coorx, Coordy e Coordz são os valores de coordenadas do nó e Flag
% é uma variável associada a este nó.

AuxPoi = fscanf(fid,' = {%f,%f,%f,%f};');

% Primeira linha de Coord

CoordBound(1,1:2) = AuxPoi(1:2);

% ic
%
% Índice da linha do CoordBound

ic = 1;

% Índice da linha de LineBoud

il = 0;

% ipp
%
% Índice do PhysicalPoint

ipp = 0;

% ipl
%
% Índice do PhysicalLine

ipl = 0;

% ips
%
```

```

% Índice do PhysicalSurface

ips = 0;

% iplans
%
% Índice dos planos modelados

iplans = 0;

% O comando de repetição abaixo consiste no comando principal do problema
% onde será lido todo o arquivo .geo.
%
% Funcionamento do comando: Para que sejam lidos os tipos de variáveis
% desejadas são primeiramente lidas as primeiras palavras da linha. Essas
% palavras são classificadas como pontos, linhas, grupos físicos e etc...
% a partir de comparações feitas entre as palavras lidas e os formatos
% desses elementos. Por exemplo: se é lido A = 'Point(1)' na linha i, esse
% valor será comparado da seguinte forma A(1:3) = 'Poi?', caso seja
% positivo será atribuído a uma variável auxiliar o valor das suas
% coordenadas.

while ~feof(fid) % O size atribuído nessa condição é
% necessário por conta da leitura de apenas um caracter no final desse
% comando.

% N1
%
% Essa variável foi definida acima e consiste na leitura do primeiro termo
% da linha, ela será utilizada para a identificação do tipo de elemento que
% está sendo avaliado (linha, ponto, grupo físico, et...)

N1 = fscanf(fid, '%s', 1)

if N1 ~= ' '

% O 'if' abaixo identifica as linhas onde são atribuídos os pontos da malha

if N1(1:3) == 'Poi'

    % AuxPoi
    %
    % Essa variável contém os valores das coordenadas dos nós da malha.
    %
    % formatação:
    %
    % AuxPoi = [Coordx, Coordy, Coordz, Flag]

    AuxPoi = fscanf(fid, ' = {%f,%f,%f,%f};');

    % Identificação da linha que será introduzida as novas coordenadas na
    % matriz Coord

    ic = ic + 1;

    % CoordBound
    %

```

```

% Construção da variável (Infor.Var.Acima).

CoordBound(ic,1:2) = AuxPoi(1:2);

end

% O 'if' abaixo identifica as linhas que estão sendo lidas no arquivo .geo
if N1(1:4) == 'Line'

    % O 'if' abaixo identifica o line loop

    if size(N1,2) == 4

        % AuxLineLoop
        %
        % Essa variável contém as informações acerca do Line Loop da geometria
        % avaliada.
        %
        % (Infor.Var.Acima)

        AxuLineLoop = fscanf(fid,' Loop(%d)');

        Descartar = 'a';

        while Descartar ~= ';'

            Descartar = fscanf(fid,'%c',1);

        end

        % O else abaixo serve para a leitura das linhas.

    else

        % AuxLine
        %
        % Essa variável contém os nós que formam uma linha
        %
        % formatação:
        %
        % AuxLine = [no1,no2]

        AuxLine = fscanf(fid,' = {%f, %f};');

        % Construção do índice associado a linha que forma a matriz LineBound

        il = il + 1;

        % LineBound
        %
        % Construção da variável (Infor.Var.Acima).

        LineBound(il,1:2) = AuxLine(1:2);

    end

```

```
end
```

```
% O 'if' abaixo consiste na identificação dos planos que serão lidos no
% arquivo .geo
```

```
if N1(1:3) == 'Pla'
```

```
    % AuxPla
    %
    % Essa variável contém os planos lidos no arquivo .geo
    %
    % formatação
    %
    % AuxPla = [NumPla], onde NumPla é a numeração do plano lido.
```

```
AuxPla = fscanf(fid, ' Surface(%d) = {%d};');
```

```
iplans = iplans + 1;
```

```
% Plans
%
% Construção da variável (Infor.Var.Acima).
```

```
Plans(iplans) = AuxPla(2);
```

```
end
```

```
% O comando de repetição abaixo editifica o Transfinite.
```

```
if N1(1:3) == 'Tra'
```

```
    % AuxTra
    %
    % Essa variável contém o valor indicado na variável Transfinite Surface
```

```
AuxTra = fscanf(fid, ' Surface {%d};');
```

```
end
```

```
% O 'if' abaixo consiste na leitura dos grupos físicos
```

```
if N1(1:3) == 'Phy'
```

```
    % AuxPhy
    %
    % A variável AuxPhy consiste na variável que indica se a linha lida é
    % referente a u grupo físico
```

```
AuxPhy = fscanf(fid, ' %c',1);
```

```
% Os 'if's' abaixo identifica o tipo de grupo físico que está sendo lido
% onde temos, onde é lido o primeiro caractere que está depois de da
% palavra Physical
%
% formatação
```

```

%
% Se AuxPhy igual a 'P' -> A linha consiste em um grupo físico de Ponto
% Se AuxPhy igual a 'L' -> A linha consiste em um grupo físico de Linha
% Se AuxPhy igual a 'S' -> A linha consiste em um grupo físico de Plano

if AuxPhy == 'P'

    % Atribuindo um flag ao grupo físico de pontos associado a
    % existência desse tipo de grupo físico.

    flagppoint1 = 1;

    % AuxPhyPoint
    %
    % Numeração dos grupos físicos de pontos

    AuxPhyPoint = fscanf(fid,'oint(%d)',1);

    Descartar = 'a';

    while Descartar ~= ';'

        Descartar = fscanf(fid,'%c',1);

    end

    % Incremento no 'ipp'

    ipp = ipp+1;

    % Construção de PhysicalPoint (Infor.Var.Acima).

    PhysicalPoint(ipp) = AuxPhyPoint;

end

if AuxPhy == 'L'

    % Atribuindo um flag ao grupo físico de linhas associado a
    % existência desse tipo de grupo físico.

    flagplines1 = 1;

    % AuxPhyLine
    %
    % Numeração dos grupos físicos das linhas

    AuxPhyLine = fscanf(fid,'ine(%d)',1);

    Descartar = 'a';

    while Descartar ~= ';'

        Descartar = fscanf(fid,'%c',1);

    end

```

```

    % Incremento no 'ipl'

    ipl = ipl+1;

    % Construção de PhysicalLine (Infor.Var.Acima).

    PhysicalLine(ipl) = AuxPhyLine;

end

if AuxPhy == 'S'

    % Atribuindo um valor diferente ao flag do grupo físico de
    % superfícies

    flagpsurface = 1;

    % AuxPhySurf
    %
    % Numeração dos grupos físicos dos planos

    AuxPhySurf = fscanf(fid,'urface(%d)',1);

    Descartar = 'a';

    while Descartar ~= ';'

        Descartar = fscanf(fid,'%c',1);

    end

    % Incremento no 'ips'

    ips = ips+1;

    % Construção de PhysicalSurface (Infor.Var.Acima).

    PhysicalSurface(ips) = AuxPhySurf;

end

end

end

end

% Os comandos abaixo atribuem o valor numo aos grupos físicos se esses não
% forem atribuídos.

if flagpsurface == 0

    PhysicalSurface = 0;

end

```

```

if flagppoint1 == 0
    PhysicalPoint = 0;
end

if flagplines1 == 0
    PhysicalLine = 0;
end

end

end

```

PreprocessorMsh.m

```

%-----%
%                               %
%                               %
%-----%
% TÍTULO: MÉTODO DOS ELEMENTOS FINITOS MISTOS PARA A SOLUÇÃO DO PROBLEMA %
% DE ACOMPLAMENTO GEOMECÂNICO DADO PELA TEORIA DE BIOT %
%-----%

%-----%
%                               %
%                               %
%-----%
% TÍTULO: Preprocessormsh %
% DESCRIÇÃO: Rotina de preprocessamento utilizada para a leitura de dados %
% para arquivos .msh de modelos feitos no Gmsh %
%-----%
% Programador: Lucíolo Victor (DECIVIL-UFPE) %
% Orientador: Prof. Dr. Ivaldo Dário (DECIVIL - UFPE) %
% Data de criação: 28-08-2015 %
% Última modificação: 29-10-2015 %
%-----%

function [PropPoint, PropLines, PropElem, Coord] = PreprocessorM-
sh(filename)

%-----%
%                               %
%                               %
%-----%
% Variáveis construídas nessa função %
%-----%

% PropPoint
%
% Essa variável contém as informações acerca dos pontos que formam a
% geometria da malha
%
% formatação
%
% PropPoint(i,:) = [flag,Gfp,no,no], onde Gfp é o grupo físico no qual o
% ponto está inserido e no é a numeração do ponto na malha do Gmsh.

% PropLines
%
% Essa variável contém as informações acerca das linhas que formam as

```

```

% arestas de contorno da malha computacional
%
% PropLines(i,:) = [flag,Gfl, Nlg, no1, no2, no3] onde Gfp o grupo físico
% associado a linha da geometria, Nlg é a numeração da linha na geometria
% que essa aresta está inserida e os outros valores são os nós que formam a
% aresta avaliada

% PropElem
%
% Essa variável contém as informações acerca dos elementos da malha.
%
% formatação
%
% PropElem(i,:) = [flag,Gfe, no1,no2,no3,no4,no5,no6], onde Gfe é o grupo
% físico associado ao elemento e os outros valores são associado aos nós
% que formam o elemento triangular de seis nós que formam a malha
% computacional, no nosso caso esse elemento é o LST.
%
%-----%
%-----%
%-----%

% flagppoint
%
% Esse flag é utilizado para identificar se há um grupo físico atribuído
% aos pontos ou esse não foi definido. Essa situação pode acontecer quando,
% por exemplo, só são introduzidos carregamentos linearmente distribuídos
% nas linhas da geometria e os apoios são atribuídos nas linhas.

flagppoint = 1;

% flagpline
%
% Esse flag é unitilizado para identificar se foi atribuído um grupo físico
% para os pontos e a condição é semelhante à flagpoint.

flagpline = 1;

% flagpsurf
%
% Esse flag é unitilizado para identificar se foi atribuído um grupo físico
% para as superfícies.

flagpsurf = 1;

% fid
%
% Identificador do arquivo referente a 'fid'.

fid = fopen(filename);

% ipropp
%
% Índice de PropPoint

ipropp = 0;

% ipropp

```

```

%
% Índice de PropLines

ipropl = 0;

% ipropp
%
% Índice de PropElem

iprope = 0;

% Nnodes
%
% Número de nós da malha computacional modelada no Gmsh

Nnodes = fscanf(fid, '%*s %*s %*s %*s %*s %*s %d', 1);

for i=1:Nnodes

    % Aux_Coord
    %
    % Variável auxiliar relativa a leitura das coordenadas de cada elementos

    Aux_Coord = fscanf(fid, '%d %f %f %f', 4);

    % Coord
    %
    % Coordenadas do arquivo .msh gerado no Gmsh
    %
    % formatação
    %
    % Coord(i,:) = (Coordx Coordy), onde Coordx e Coordy são as coordenadas x
    % e y do nó i, respectivamente.

    Coord(i,:) = Aux_Coord(2:3);

end

% NelemMsh
%
% Essa variável contém o número de elementos da malha, onde esse número de
% elementos é relativo a soma do número de arestas de contorno, número de
% pontos gerados na geometria e número de elementos da malha computacional.
% É dado no arquivo de malha e esses valores são gerados pelo próprio Gmsh.

NelemMsh = fscanf(fid, '%*s %*s %d', 1);

for i=1:NelemMsh

    % AuxElementsMsh
    %
    % Essa variável consiste na variável que contém o vetor lido nas linhas
    % relativos aos elementos no arquivo .msh.

    AuxElementsMsh = fscanf(fid, '%*d %d', 1);

    % O 'if' abaixo identifica na segunda coluna da linha equivalente aos

```

```

% elementos se o tipo de elemento que está sendo lido é um ponto da
% geometria.

if AuxElementsMsh == 15

    % AuxPoints
    %
    % Essa variável é utilizada como auxiliar e consiste em um vetor
    % relativo a leitura da linha realizada no arquivo .msh onde são
    % identificados os pontos que foramam a geometria.

    AuxPoints = fscanf(fid, '%d %d %d %d', 4);

    % Incremento no índice ipropp

    ipropp = ipropp + 1;

    % PropPoint
    %
    % A variável PropPoint está sendo preenchida abaixo (Infor.Var.Acima)

    PropPoint(ipropp,:) = AuxPoints';

    % Atribuindo o valor igua a zero para identificar que há um grupo
    % físico de pontos

    flagppoint = 0;

end

% O 'if' abaixo identifica na segunda coluna da linha equivalente aos
% elementos se o tipo de elemento que está sendo lido é uma aresta.

if AuxElementsMsh == 8

    % AuxLines
    %
    % Essa variável é utilizada como auxiliar e consiste em um vetor
    % relativo a leitura da linha realizada no arquivo .msh. onde são
    % identificadas as arestas de contorno

    AuxLines = fscanf(fid, '%d %d %d %d %d %d', 6);

    % Incremento no índice.

    ipropl = ipropl + 1;

    % PropLines
    %
    % A variável PropLines está sendo preenchida abaixo (Infor.Var.Acima)

    PropLines(ipropl,:) = AuxLines';

    % Atribuindo o valor para identificar que foram atribuídos grupos
    % físicos às linhas na geometria.

    flagpline = 0;

```

```
end
```

```
% O 'if' abaixo identifica na segunda coluna da linha equivalente aos
% elementos se o tipo de elemento que está sendo lido é um elemento da
% malha.
```

```
if AuxElementsMsh == 9
```

```
    % AuxElem
```

```
    %
```

```
    % Essa variável é utilizada como auxiliar e consiste em um vetor
    % relativo a leitura da linha realizada no arquivo .msh onde são
    % identificados os elementos da malha.
```

```
    AuxElem = fscanf(fid, '%d %d %d %d %d %d %d %d %d', 9);
```

```
    % Incremento no índice iprope
```

```
    iprope = iprope + 1;
```

```
    % PropElem
```

```
    %
```

```
    % A variável PropPoint está sendo preenchida abaixo (Infor.Var.Acima)
```

```
    PropElem(iprope,:) = AuxElem';
```

```
    % Flag para identificar se foi atribuído valores aos grupos físicos de
    % superfícies.
```

```
    flagpsurf = 0;
```

```
end
```

```
end
```

```
disp(' |-----|')
disp(' |          INFORMAÇÕES SOBRE OS GRUPOS FÍSICOS          |')
disp(' |-----|')
```

```
% Comando que atribui o valor das variáveis PropPoint e PropElem iguais a
% zero se eles não forem lidos no arquivo de malha, isso é utilizado para a
% introdução das condições de contorno.
```

```
if flagppoint == 0
```

```
    disp(' ')
```

```
    disp('Foi(ram) atribuído(s) grupo(s) físico(s) para o(s) ponto(s)')
```

```
    disp(' ')
```

```
elseif flagppoint == 1
```

```
    disp(' ')
```

```
    disp('Não foi(ram) atribuído(s) grupo(s) físico(s) para o(s) ponto(s)')
```

```
    disp(' ')
```

```
    PropPoint = 0;
```

```

end

if flagpline == 0

    disp(' ')
    disp('Foi(ram) atribuído(s) grupo(s) físico(s) para a(s) linha(s)')
    disp(' ')

elseif flagpline == 1

    disp(' ')
    disp('Não foi(ram) atribuído(s) grupo(s) físico(s) para a(s) linha(s)')
    disp(' ')

    PropLines = 0;

end

if flagpsurf == 0

    disp(' ')
    disp('Foi(ram) atribuído(s) grupo(s) físico(s) para a(s) superfí-
cie(s)')
    disp(' ')

elseif flagpsurf == 1

    disp(' ')
    disp('Não foi(ram) atribuído(s) grupo(s) físico(s) para a(s) superfí-
cie(s)')
    disp(' ')
    PropElem = 0;

end

% Fechando o arquivo de malha

fopen(fid);

end

```

ProcessoBiot.m

```

%-----%
%
%                               ROTINA                               %
%-----%
% TÍTULO: MÉTODO DOS ELEMENTOS FINITOS MISTOS PARA A SOLUÇÃO DO PROBLEMA %
% DE ACOMPLAMENTO GEOMECÂNICO DADO PELA TEORIA DE BIOT                %
%-----%

%-----%
%                               SUB-ROTINA                           %
%-----%

```

```

% TÍTULO: ProcessoBiot %
% DESCRIÇÃO: Processamento de dados dos arquivos .msh,.geo e da %
% entrada de dados para o problema mecânico. %
%-----%
% Programador: Lucíolo Victor (DECIVIL-UFPE) %
% Orientador: Prof. Dr. Ivaldo Dário (DECIVIL - UFPE) %
% Data de criação: 29-10-2015 %
% Última modificação: 23-03-2016 %
%-----%

function [X,Glp,K,Ce] = ProcessoBiot( PropPoint, PropLines, PropElem,...
    Coord, CaracPlan, CaracPoint, CaracLine, TipProb, MetCon,Nmaxint,...
    DeltaT, Theta,Nint0)

if PropElem ~= 0

% Glp
%
% Essa variável contém uma numeração associada aos graus de liberdade das
% tensões médias, essa será explicada, de maneira detalhada, na função
% GIPressure.

[ Glp, GlpCC ] = GIPressure(PropElem, Coord);

% A função abaixo calcula as matrizes Kup, Kpp e Kuu para o problema do
% método de Elementos Finitos Mistos.

[L,H,K] = MontMatSist(PropElem,Coord,Glp,CaracPlan);

save '-ascii' 'K.dat' 'K'
save '-ascii' 'H.dat' 'H'
save '-ascii' 'L.dat' 'L'

% A função abaixo introduz as condições de contorno nas matrizes acima e cria
% os vetores independentes do problema hidráulico e do problema mecânico.

[F,Kf,Hf,Q] = CondContorno(K,CaracPoint,CaracLine,Coord,PropPoint,...
    PropLines,Glp,H,GlpCC,CaracPlan,PropElem,L);

save '-ascii' 'Kf.dat' 'Kf'
save '-ascii' 'Hf.dat' 'Hf'

%%

%-----%
% SOLUÇÃO DO PROBLEMA MECÂNICO ELÁSTICO - INÍCIO %
%-----%

if TipProb == 1

% U V
%
% Vetor de deslocamentos
disp('Kf')
Kf
disp('F')
F
disp('U')
U = inv(Kf)*(-F);

```

```

disp(' ')
disp('Vetor de deslocamentos nodais')
disp(U)

% X
%
% Vetor de resultados do problema

X = U;

Ce = 1;
%-----%
%           SOLUÇÃO DO PROBLEMA MECÂNICO ELÁSTICO - FIM           %
%-----%

%-----%
%           SOLUÇÃO DO PROBLEMA HIDRÁULICO - INÍCIO             %
%-----%

elseif TipProb == 2

% P
%
% Valor de pressões.

P = inv(Hf)*Q;

% X
%
% Vetor de resultados do problema.

X = P;

disp(' ')
disp('Vetor de pressões nodais')

disp(P)

disp('Q')
Q

disp('Graus de liberdade da pressão')
disp(Glp)

Ce = 1;

%-----%
%           SOLUÇÃO DO PROBLEMA HIDRÁULICO - FIM           %
%-----%

%-----%
%           SOLUÇÃO DO PROBLEMA ACOPLADO (HIDROMECHANICO) - INÍCIO %
%-----%

elseif TipProb == 3

[X,Ce] = ElasticConsolidation(K,L,H,MetCon,Nmaxint,DeltaT,Theta,...

```

```
CaracPoint,CaracLine,Coord,PropPoint,PropLines,GlpCC,CaracPlan,...
PropElem,Nint0);
```

```
end
```

```
%-----%
%           SOLUÇÃO DO PROBLEMA ACOPLADO (HIDROMECÂNICO) - FIM           %
%-----%
```

```
%%
```

```
else
```

```
% Erro para que o programa só rode se forem inseridos os valores referentes
% aos materiais.
```

```
disp(' |-----| ')
disp(' |           ERRO           | ')
disp(' |-----| ')
disp(' | As propriedades dos materiais da análise pressão ser inseridas | ')
disp(' |-----| ')
disp(' ')
disp(' ')
disp(' ')
```

```
end
```

```
end
```

GIPressure.m

```
%-----%
%           ROTINA           %
%-----%
% TÍTULO: MÉTODO DOS ELEMENTOS FINITOS MISTOS PARA A SOLUÇÃO DO PROBLEMA %
% DE ACOMPLAMENTO GEOMECÂNICO DADO PELA TEORIA DE BIOT           %
%-----%
```

```
%-----%
%           SUB-ROTINA           %
%-----%
% TÍTULO: GIPressure %
% DESCRIÇÃO: Rotina para a montagem do vetor contendo os graus de liberda-%
% de pressões no elemento %
%-----%
% Programador: Lucíolo Victor (DECIVIL-UFPE) %
% Orientador: Prof. Dr. Ivaldo Dário (DECIVIL - UFPE) %
% Data de criação: 16-09-2015 %
% Última modificação: 16-11-2015 %
%-----%
```

```
% Glp
%
% Vetor contendo a numeração dos graus de liberdade contendo as pressões
% calculadas
%
% Formatação: Glp(1,i) = nói
%             Glp(2,i) = graude liberdade correspondente ao nói
```

```

function [ Glp,GlpCC ] = GlPressure(PropElem, Coord)

% AuxCoord
%
% Vetor que contem as numerações dos nós da malha

AuxCoord = 1:size(Coord,1);

% Glp
%
% Esse vetor contém os graus de liberdade onde as pressões são calculadas.
% A primeira linha consiste no sistema local e a segunda linha consiste na
% numeração global. o procedimento de tirar coluna por coluna foi feito
% desta maneira por conta da versão do MATLAB 2009 que só trabalha dessa
% forma.

Glp1 = setdiff(AuxCoord, PropElem(:,7));
Glp2 = setdiff(Glp1, PropElem(:,8));
Glp = setdiff(Glp2, PropElem(:,9));

% Preenchimento da numeração dos graus de liberdade para a as matrizes L
% e H.
%
% Obs: os graus de liberdade para a pressão são numerados da seguinte
% maneira:
% A partir dos graus de liberdade globais que, neste caso, são iguais a
% numeração dos nós, são retirados os nós onde são calculados
% exclusivamente deslocamentos. Sendo assim, a numeração local (referente a
% numeração das matrizes L e H), é estelecida da maneira como se segue
% no exemplo abaixo
%
% Nós da malha: 1 2 3 4 5 6 7 8 9 10
%
% Nós onde são calculados apenas deslocamentos: 2 3 7 8
%
% Nós restantes: 1 4 5 6 9 10
% Grau de liberdade local equivalente para os nós acima: 1(1) 4(2) 5(3)
% 6(4) 9(5) 10(6)
%
% O GlpCC indexa os valores nodais no coluna com seu próprio índice, nesse
% caso teremos
%
% GlpCC = [ 1 0 0 4 5 6 0 0 9 10 ;
%           1 0 0 2 3 4 0 0 5 6 ];

Glp(2,:) = 1:size(Glp,2);

% GlpCC
%
% Gaus de liberdade utilizados par a aintrodução das condições de contorno.

for i=1:size(Glp,2)

    GlpCC(:,Glp(1,i)) = Glp(:,i);

end

end

```

MontMatSist.m

```

%-----%
%                               %
%                               %
%-----%
% TÍTULO: MÉTODO DOS ELEMENTOS FINITOS MISTOS PARA A SOLUÇÃO DO PROBLEMA %
% DE ACOMPLAMENTO GEOMECÂNICO DADO PELA TEORIA DE BIOT %
%-----%

%-----%
%                               %
%                               %
%-----%
% TÍTULO: MontMatSist %
% DESCRIÇÃO: Rotina de montagem das matrizes do problema mecânico para a %
% solução pelo Método dos Elementos Finitos Mistos %
%-----%
% Programador: Lucíolo Victor (DECIVIL-UFPE) %
% Orientador: Prof. Dr. Ivaldo Dário (DECIVIL - UFPE) %
% Data de criação: 22-10-2015 %
% Última modificação: 13-01-2016 %
%-----%

function [L,H,K] = MontMatSist(PropElem,Coord,Glp,CaracPlan)

% nump
%
% Número de pressões calculadas

nump = size(Glp,2);

% Nnode
%
% Número de nós da malha

Nnode = size(Coord,1);

% Inicialização de L

L = zeros(Nnode*2,nump);

% Inicialização de H

H = zeros(nump,nump);

% K
%
% Inicialização de K

K = zeros(Nnode*2);

% Nelem
%
% Número de elementos da malha

Nelem = size(PropElem,1);

for k=1:Nelem

```

```

% Gfelem
%
% Essa variável infica o grupo físico que o elemento faz parte

Gfelem = PropElem(k,2);

% Ex
%
% Módulo de elasticidade na direção x.

Ex = CaracPlan(Gfelem,1);

% Ex
%
% Módulo de elasticidade na direção y.

Ey = CaracPlan(Gfelem,2);

% vxy
%
% Coeficiente de Poisson associado às direções x e y.

vxy = CaracPlan(Gfelem,3);

% kxx
%
% Permeabilidade do meio na direção x.

kxx = CaracPlan(Gfelem,4);

% kyy
%
% Permeabilidade do meio na direção y.

kyy = CaracPlan(Gfelem,5);

% kxy
%
% Permeabilidade do meio na direção xy.

kxy = CaracPlan(Gfelem,6);

% yw
%
% Peso específico do material.

yw = CaracPlan(Gfelem,10);

% Kmat
%
% Matriz de permeabilidades do material.

Kmat = [ kxx kxy ; kxy kyy ];

```

```

% C1,C2,C3
%
% Coeficientes para a matriz constitutiva.

C1 = Ex*(1-vxy)/((1+vxy)*(1-2*vxy));

C2 = Ey*vxy/((1+vxy)*(1-2*vxy));

C3 = ((Ex+Ey)/2)/(2*(1+vxy));

% D
%
% matriz constitutiva referente ao estado plano de tensões

D = [ C1 C2 0 ; C2 C1 0 ; 0 0 C3 ];

% dg
%
% dg consiste nos ponteiros para o elemento avaliado para a construção da
% matriz de rigidez global

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

[L_local, H_local, K_local] = Calc_Matrizes_locais(k, Coord, ...
    PropElem ,D, Kmat, yw);

% dgp
%
% dgp consiste nos ponteiros relativos a pressão para o elemento avaliado
% na para a construção das matrizes Kup e Kpp

for i=1:size(Glp,2)

    if PropElem(k,4) == Glp(1,i)

        dgp(1) = Glp(2,i);

    end

    if PropElem(k,5) == Glp(1,i)

        dgp(2) = Glp(2,i);

    end

    if PropElem(k,6) == Glp(1,i)

```

```

    dgp(3) = Glp(2,i);

    end

end

%%

for ii=1:12

    for jj=1:3

        % L
        %
        % Montagem da Matriz L

        L(dg(ii),dgp(jj)) = L(dg(ii),dgp(jj)) + L_local(ii,jj);

    end

end

%%

for ii=1:3

    for jj=1:3

        % H
        %
        % Montagem da H

        H(dgp(ii),dgp(jj)) = H(dgp(ii),dgp(jj)) + H_local(ii,jj);

    end

end

%%

for ii=1:12

    for jj=1:12

        % K
        %
        % Montagem da Matriz de rigidez Global

        K(dg(ii),dg(jj)) = K(dg(ii),dg(jj)) + K_local(ii,jj);

    end

end

```

```

%%
end

disp(' |-----| ')
disp(' |           MONTAGEM DAS MATRIZES - STATUS           | ')
disp(' |-----| ')
disp(' ')
disp('/!\ Montagem de L OK! /!\')
disp('/!\ Montagem de H OK! /!\')
disp('/!\ Montagem de K OK! /!\')

end

```

Calc_Matrizes_locais.m

```

%-----%
%                               %
%                               ROTINA                               %
%-----%
% TÍTULO: MÉTODO DOS ELEMENTOS FINITOS MISTOS PARA A SOLUÇÃO DO PROBLEMA %
% DE ACOMPLAMENTO GEOMECÂNICO DADO PELA TEORIA DE BIOT                %
%-----%

%-----%
%                               SUB-ROTINA                           %
%-----%
% TÍTULO: Calc_Matrizes_locais                                       %
% DESCRIÇÃO: Rotina para a montagem da matrizes locais para a solução do %
% problema mecânico pelo Método dos Elementos Finitos Mistos        %
%-----%
% Programador: Lucíolo Victor (DECIVIL-UFPE)                          %
% Orientador: Prof. Dr. Ivaldo Dário (DECIVIL - UFPE)                %
% Data de criação: 22-10-2015                                         %
% Última modificação: 13-01-2016                                      %
%-----%

function [L_local, H_local, K_local] = Calc_Matrizes_locais(m, Coord, ...
    PropElem ,D ,Kmat, yw)

% nen
%
% Número de nós por elemento

nen = 6;

% ndof
%
% Número de graus de liberdade por nó

ndof = 2;

% L_local
%

```

```

% Inicialização da L_local

L_local = zeros(12,3);

% K_local
%
% Matriz da K_local

K_local = zeros(nen*ndof,nen*ndof);

% Nós que formam o elemento i
% Coordenadas de cada nó

x1 = Coord(PropElem(m,4),1);
x2 = Coord(PropElem(m,5),1);
x3 = Coord(PropElem(m,6),1);
x4 = Coord(PropElem(m,7),1);
x5 = Coord(PropElem(m,8),1);
x6 = Coord(PropElem(m,9),1);

y1 = Coord(PropElem(m,4),2);
y2 = Coord(PropElem(m,5),2);
y3 = Coord(PropElem(m,6),2);
y4 = Coord(PropElem(m,7),2);
y5 = Coord(PropElem(m,8),2);
y6 = Coord(PropElem(m,9),2);

% CC
%
% Matriz de Coordenadas nodais

CC=[x1 x2 x3 x4 x5 x6;
    y1 y2 y3 y4 y5 y6]';

% Essa matriz são os valores dos pontos de Gauss e seus respectivos pesos

Int_Gauss_Aux = [ 1/6 1/6 1/6
                 1/6 2/3 1/6
                 1/6 1/6 2/3];

% npg
%
% Número de pontos de Gauss por elementos

npg = size(Int_Gauss_Aux,1);

% Comandos de repetição
%
% Os comandos de repetição abaixo estão fazendo a integração numérica
% para a construção da matriz de rigidez.

for i=1:npg;

    epsilon = Int_Gauss_Aux(i,2);
    neta = Int_Gauss_Aux(i,3);
    gama = 1 - epsilon - neta ;

```

```

w = Int_Gauss_Aux(i,1);

% N
%
% Função de forma mapeada

%N1 = -gama*(1-2*gama);
%N2 = -epsilon*(1-2*epsilon);
%N3 = -neta*(1-2*neta);
%N4 = 4*epsilon*gama;
%N5 = 4*epsilon*neta;
%N6 = 4*neta*gama;

% GN
%
% Gradiente de N

GN = [ 1-4*gama    -1+4*epsilon    0    4*(gama - epsilon)
4*neta          -4*neta;
      1-4*gama    0    -1+4*neta    -4*epsilon
4*epsilon    4*(gama - neta) ];
% J
%
% Matriz jacobiana

J = GN*CC;

% detJ
%
% Determinante do jacobiano

detJ = det(J);

% BB
%
% Derivada da matriz de forma

BB = J\GN;

B1x = BB(1,1);
B2x = BB(1,2);
B3x = BB(1,3);
B4x = BB(1,4);
B5x = BB(1,5);
B6x = BB(1,6);
B1y = BB(2,1);
B2y = BB(2,2);
B3y = BB(2,3);
B4y = BB(2,4);
B5y = BB(2,5);
B6y = BB(2,6);

```

```

    B = [ B1x      0      B2x      0      B3x      0      B4x      0      B5x
0        B6x      0;
          0      B1y      0      B2y      0      B3y      0      B4y      0
B5y      0      B6y;
          B1y      B1x      B2y      B2x      B3y      B3x      B4y      B4x      B5y
B5x      B6y      B6x ];

% Np
%
% Matriz contendo as funções de forma para a pressão
%
% Obs: Elemento triangular com 3 nós.
% REFERÊNCIA 4.

Np = [ 1-epsilon-neta epsilon neta];

% Dup
%
% Vetor 'm'

Dup = [ 1 1 0 ];

% Construção das matrizes locais pelo Método da Integração Numérica de
% Gauss

L_local = L_local - w*B'*Dup'*Np*detJ;

K_local = K_local + w*B'*D*B*detJ;

end

% Matriz dos coeficientes.

A = [ 1 x1 y1 ;
      1 x2 y2 ;
      1 x3 y3 ];

% Gradiente das funções de forma das poropressões.

Bp = [ 0 1 0 ; 0 0 1 ]*inv(A);

% Ae
%
% Área do elemento.

Ae = det(A)/2;

% Construção da matriz H local, observe que essa matriz não é calculada a
% partir de um processo iterativo, seu cálculo é explícito.

H_local = -Bp'*Kmat*Bp*Ae/yw;

end

```

CondContorno.m

```

%-----%
%
%                               ROTINA                               %
%-----%
% TÍTULO: MÉTODO DOS ELEMENTOS FINITOS MISTOS PARA A SOLUÇÃO DO PROBLEMA %
% DE ACOMPLAMENTO GEOMECÂNICO DADO PELA TEORIA DE BIOT                %
%-----%

%-----%
%                               SUB-ROTINA                           %
%-----%
% TÍTULO: CondContorno                                               %
% DESCRIÇÃO: Rotina para introdução das condições de contorno na matriz %
% de rigidez e para a construção do vetor de forças.                %
%-----%
% Programador: Lucíolo Victor (DECIVIL-UFPE)                         %
% Orientador: Prof. Dr. Ivaldo Dário (DECIVIL - UFPE)               %
% Data de criação: 24-10-2015                                         %
% Última modificação: 01-12-2015                                       %
%-----%

function [F,Kf,Hf,Q] = CondContor-
no(K,CaracPoint,CaracLine,Coord,PropPoint,...
    PropLines,Glp,H,GlpCC,CaracPlan,PropElem,L)

%%

%-----%
% Inicialização das matrizes - Início                                %
%-----%

% Kf
%
% Matríz K final.

Kf = K;

% Hf
%
% Matríz hidráulica final

Hf = H;

% Matríz de acoplamento final

Lf = L;

% Inicialização do vetor de forças associado ao Kuu.

F = zeros(size(Coord,1)*2,1);

% Q
%
% Inicialização do vetor independente para os fluxos.

Q = zeros(size(Glp,2),1);

```

```

%-----%
% Inicialização das matrizes - Fim
%-----%

%%

%-----%
% Introdução das condições de contorno das arestas - Início
%-----%

% O comando de repetição abaixo introduz os condições de contorno
% associadas às linhas da geometria

if PropLines ~= 0

for i=1:size(PropLines,1)

% inicialização do vetor de forças locais do elemento

felem = zeros(6,1);

% no1, no2, no3
%
% Nós que formam a aresta avaliada.

no1 = PropLines(i,4);
no2 = PropLines(i,6);
no3 = PropLines(i,5);

% x1, y1, x2, y2, x3, y3
%
% Coordenadas x e y dos três nós que formam a aresta avaliada.

x1 = Coord(no1,1);
y1 = Coord(no1,2);
x2 = Coord(no2,1);
y2 = Coord(no2,2);
x3 = Coord(no3,1);
y3 = Coord(no3,2);

% pg
%
% Matriz que contém os valores da integração de Gauss. A primeira coluna
% contém as coordenadas mapeadas e a segunda os pesos associadas a cada
% coordenada.

pg = [ -0.7746 0.55556;
        0.0000 0.88889;
        0.7746 0.55556 ];

% npg
%
% Número de pontos de Gauss utilizados na integração numérica.

npg = size(pg,1);

```

```

% Numgf
%
% Número do grupo físico que a aresta pertence.

Numgf = PropLines(i,2);
%%

%-----%
% Valores das condições de contorno mecânica nas arestas - Início
%-----%

% fx,fy
%
% Forças distribuídas nas direções x e y.

fx = CaracLine(Numgf,1);
fy = CaracLine(Numgf,2);

% fn,ft
%
% Forças distribuídas nas direções normal e tangencial à aresta avaliada.

fn = CaracLine(Numgf,3);
ft = CaracLine(Numgf,4);

% ux,uy
%
% Deslocamentos prescritos nas direções x e y na aresta avaliada.

ux = CaracLine(Numgf,5);
uy = CaracLine(Numgf,6);

% kx,ky
%
% São os flags relativos às condições de contorno na aresta avaliada.

kx = CaracLine(Numgf,7);
ky = CaracLine(Numgf,8);

%-----%
% Valores das condições de contorno mecânicas nas arestas - Fim
%-----%

%%

%-----%
% Valores das condições de contorno hidráulicas nas arestas - Início
%-----%
% v
%
% Fluxo imposto na aresta.

v = CaracLine(Numgf,9);

% p
%
% Pressão imposta na aresta.

```

```

p = CaracLine(Numgf,10);

% y
%
% Valor de gama para a condição de contorno hidráulica.

y = CaracLine(Numgf,11);

%-----%
% Valores das condições de contorno hidráulicas nas arestas - Fim
%-----%

%%

%-----%
% Introdução das condições de contornos mecânicas nas arestas - Início
%-----%

% O comando de repetição abaixo é relativo à introdução das condições de
% contorno associadas as arestas de contorno.
for j=1:npg

    % S1
    %
    % Valor da coordenada mapeada.

    S1 = pg(j,1);

    % w
    %
    % Peso da integração de Gauss.

    w = pg(j,2);

    % N1,N2,N3
    %
    % Valores das funções de forma na coordenada S1.

    N1 = ((S1^2) - S1)/2;
    N2 = 1 - (S1^2);
    N3 = ((S1^2) + S1)/2;

    % dN1dS1,dN2dS1,dN3dS1
    %
    % Derivadas das funções de forma em relação à coordenada do mapeamento

    dN1dS1 = S1 - 1/2;
    dN2dS1 = -2*S1;
    dN3dS1 = S1 + 1/2;

    % dxdS1,dydS1
    %
    % derivadas das direções coordenadas em relação à coordenada do
    % mapeamento.

    dxdS1 = dN1dS1*x1 + dN2dS1*x2 + dN3dS1*x3;

```

```

dydS1 = dN1dS1*y1 + dN2dS1*y2 + dN3dS1*y3;

% PrimNMatriz
%
% Primeira matriz associada às funções de forma presente na expressão
% para a integral.

PrimNMatriz = [ N1 0 ;
                0 N1 ;
                N2 0 ;
                0 N2 ;
                N3 0 ;
                0 N3 ];

% MatrizDeriv
%
% Matriz que contém as derivadas, essa matriz é utilizada na integração
% das cargas

MatrizDeriv = [ dxdS1 -dydS1 ;
                dydS1  dxdS1 ];

% Matrizp
%
% Essa matriz é associada as cargas aplicadas na linhas, ela será
% utilizada na integração numérica.

Matrizp = [ ft ft ft ;
            fn fn fn ];

% dLdS1
%
% Jacobiano das transformadas

dLdS1 = sqrt( dxdS1^2 + dydS1^2 );

% SegNMatriz
%
% Segunda matriz associada às funções de forma presente na expressão
% para a integral.

SegNMatriz = [ N1 N2 N3 ]';

% Determinação do vetor de forças nodais equivalentes

felem = felem + PrimNMatriz*MatrizDeriv*Matrizp*SegNMatriz*w + ...
        PrimNMatriz*[ fx fy ]'*dLdS1*w ;

end

% auxK
%
% Matriz auxiliar para a introdução dos apoios.

auxK = zeros(size(K,1),1);

% auxL

```

```

%
% Matriz auxiliar para introdução dos apoios na matriz de complemento.

auxL = zeros(size(H,1),1);

% glno1x,glno1y,glno2x,glno2y,glno3x,glno3y
%
% Graus de liberdade para os nós da aresta avaliado onde serão
% introduzidas as forças nodais equivalentes.

glno1x = no1*2-1;
glno1y = no1*2;

glno2x = no2*2-1;
glno2y = no2*2;

glno3x = no3*2-1;
glno3y = no3*2;

% Introdução das condições de contorno no vetor de forças

F(glno1x) = F(glno1x) + felem(1);
F(glno1y) = F(glno1y) + felem(2);
F(glno2x) = F(glno2x) + felem(3);
F(glno2y) = F(glno2y) + felem(4);
F(glno3x) = F(glno3x) + felem(5);
F(glno3y) = F(glno3y) + felem(6);

% Introdução das condições de contorno na matriz de rigidez global.

if kx == 1

Kf(glno1x,:) = auxK;
Kf(glno2x,:) = auxK;
Kf(glno3x,:) = auxK;
Kf(glno1x,glno1x) = 1;
Kf(glno2x,glno2x) = 1;
Kf(glno3x,glno3x) = 1;
Lf(glno1x,:) = auxL;
Lf(glno2x,:) = auxL;
Lf(glno3x,:) = auxL;
F = F - K(:,glno1x)*ux;
F = F - K(:,glno2x)*ux;
F = F - K(:,glno3x)*ux;

end

if ky == 1

Kf(glno1y,:) = auxK;
Kf(glno2y,:) = auxK;
Kf(glno3y,:) = auxK;
Kf(glno1y,glno1y) = 1;
Kf(glno2y,glno2y) = 1;
Kf(glno3y,glno3y) = 1;
Lf(glno1y,:) = auxL;
Lf(glno2y,:) = auxL;
Lf(glno3y,:) = auxL;
F = F - K(:,glno1y)*uy;

```

```
F = F - K(:,glno2y)*uy;
F = F - K(:,glno3y)*uy;
```

```
end
```

```
%-----%
% Introdução das condições de contornos mecânicas nas arestas - Fim
%-----%
```

```
%%
```

```
%-----%
% Introdução das condições de contornos hidráulicas nas arestas - Início
%-----%
```

```
% noh1, noh2
%
% Primeiro e segundo nó da aresta considerando apenas o problema
% hidráulico.
```

```
noh1 = no1;
noh2 = no3;
```

```
% xh1, yh1, xh2, yh2
%
% Coordenadas x e y dos nós noh1 e noh2.
```

```
xh1 = Coord(noh1,1);
yh1 = Coord(noh1,2);
xh2 = Coord(noh2,1);
yh2 = Coord(noh2,2);
```

```
% L
%
% Comprimento da aresta avaliada.
```

```
L = sqrt( (xh1 - xh2)^2 + (yh1 - yh2)^2);
```

```
% vce
%
% Vazão concentrada equivalente.
```

```
vce = v*L/2;
```

```
% Introdução da vazão concentrada equivalente no vetor independente de
% fluxos.
```

```
Q(GlpCC(2,noh1)) = vce + y*p;
Q(GlpCC(2,noh2)) = vce + y*p;
```

```
% Introdução da condição de contorno na matriz H.
```

```
Hf(GlpCC(2,noh1),GlpCC(2,noh1)) = Hf(GlpCC(2,noh1),GlpCC(2,noh1)) + y;
Hf(GlpCC(2,noh2),GlpCC(2,noh2)) = Hf(GlpCC(2,noh2),GlpCC(2,noh2)) + y;
```

```
%-----%
```

```

% Introdução das condições de contornos hidráulicas nas arestas - Fim
%-----%

end

end

%-----%
% Introdução das condições de contorno das arestas - Fim
%-----%

%%

%-----%
% Introdução das condições de contorno dos pontos - Início
%-----%

% O comando de repetição abaixo introduz as condições de contorno
% associadas aos pontos nodais.

if PropPoint ~= 0

for i=1:size(PropPoint,1)

% NumGfPoint
%
% Número do grupo físico dos pontos.

NumGfPoint = PropPoint(i,2);

% no
%
% Nó avaliado.

no = PropPoint(i,3);

%-----%
% Valores e introdução das condições de contorno mecânicas nos pontos
% - Início
%-----%

% fx,fy
%
% Forças nodais concentradas.

fx = CaracPoint(NumGfPoint,1);
fy = CaracPoint(NumGfPoint,2);

% ux,uy
%
% Deslocamentos prescritos nos nós.

```

```

ux = CaracPoint(NumGfPoint,3);
uy = CaracPoint(NumGfPoint,4);

% kx,ky
%
% Flag associado à restrição de apoio do nó.

kx = CaracPoint(NumGfPoint,5);
ky = CaracPoint(NumGfPoint,6);

% glnox,glnoy
%
% Gruas de liberdade em x e y do nó avaliado.

glnox = no*2 - 1;
glnoy = no*2;

% AuxKp
%
% Essa variável é uma variável auxiliar utilizada para a introdução dos
% apoios.

AuxKp = zeros(size(K,1),1);

% Contribuição na matriz de rigidez global das condições de contorno
% associadas aos nós.

if kx == 1

Kf(glnox,:) = AuxKp;
Kf(glnox,glnox) = 1;
% Lf(glnox,:) = AuxLp;
F = F - K(:,glnox)*ux;

end

if ky == 1

Kf(glnoy,:) = AuxKp;
Kf(:,glnoy) = AuxKp';
Kf(glnoy,glnoy) = 1;
% Lf(glnoy,:) = AuxLp;
F = F - K(:,glnoy)*uy;

end

% Introdução das forças concentradas.

F(glnox) = F(glnox) + fx;
F(glnoy) = F(glnoy) + fy;

%-----%
% Valores e introdução das condições de contorno mecânicas nos pontos
% - Fim
%-----%

```

```

%%

%-----%
% Valores e introdução das condições de contorno hidráulicas nos pontos
% - Início
%-----%

% v
%
% Valor de fluxo prescrito.

v = CaracPoint(NumGfPoint,7);

% p
%
% Valor de pressão prescrita no ponto.

p = CaracPoint(NumGfPoint,8);

% y
%
% Valor de gama para o ponto.

y = CaracPoint(NumGfPoint,9);

Q(GlpCC(2,no)) = v + y*p;

% Introdução da condição de contorno na matriz H.

Hf(GlpCC(2,no),GlpCC(2,no)) = Hf(GlpCC(2,no),GlpCC(2,no)) + y;

%-----%
% Valores e introdução das condições de contorno hidráulicas nos pontos
% - Fim
%-----%

%%
end

end

%-----%
% Introdução das condições de contorno dos pontos - Fim
%-----%

%%

%-----%
% Introdução das condições de contorno dos elementos - Início
%-----%

for k=1:size(PropElem,1)

% F_local é o vetor de cargas nodais equivalentes para as forças de corpo

```

```

% para cada elemento avaliado.

F_local = zeros(12,1);

% Nós que formam o elemento i
% Coordenadas de cada nó

x1 = Coord(PropElem(k,4),1);
x2 = Coord(PropElem(k,5),1);
x3 = Coord(PropElem(k,6),1);
x4 = Coord(PropElem(k,7),1);
x5 = Coord(PropElem(k,8),1);
x6 = Coord(PropElem(k,9),1);

y1 = Coord(PropElem(k,4),2);
y2 = Coord(PropElem(k,5),2);
y3 = Coord(PropElem(k,6),2);
y4 = Coord(PropElem(k,7),2);
y5 = Coord(PropElem(k,8),2);
y6 = Coord(PropElem(k,9),2);

% gfelem
%
% Grupo correspondente ao elemento.

gfelem = PropElem(k,2);

% bx,by
%
% Componentes do vetor de forças de corpo.

bx = CaracPlan(gfelem,7);
by = CaracPlan(gfelem,8);

% Vetorb
%
% Vetor de forças de corpo.

Vetorb = [ bx ; by ];

% CC
%
% Matriz de Coordenadas nodais

CC=[x1 x2 x3 x4 x5 x6;
    y1 y2 y3 y4 y5 y6]';

% Essa matriz são os valores dos pontos de Gauss e seus respectivos pesos

Int_Gauss_Aux = [ 1/6 1/6 1/6
                  1/6 2/3 1/6
                  1/6 1/6 2/3];

% npg
%
% Número de pontos de Gauss por elementos

npg = size(Int_Gauss_Aux,1);

```

```

for i=1:npg;

    epsilon = Int_Gauss_Aux(i,2) ;
    neta = Int_Gauss_Aux(i,3) ;
    gama = 1 - epsilon - neta;

    w = Int_Gauss_Aux(i,1);

    % N
    %
    % Função de forma mapeada

    N1 = -gama*(1-2*gama);
    N2 = -epsilon*(1-2*epsilon);
    N3 = -neta*(1-2*neta);
    N4 = 4*epsilon*gama;
    N5 = 4*epsilon*neta;
    N6 = 4*neta*gama;

    % MatrizN
    %
    % Matriz das funções de forma

    MatrizN = [ N1  0  N2  0  N3  0  N4  0  N5  0  N6  0;
                0  N1  0  N2  0  N3  0  N4  0  N5  0  N6];

    % GN
    %
    % Gradiente de N

    GN = [ 1-4*gama  -1+4*epsilon  0  4*(gama - epsilon)
4*neta  -4*neta;
          1-4*gama  0  -1+4*neta  -4*epsilon
4*epsilon  4*(gama - neta) ];

    % J
    %
    % Matriz jacobiana

    J = GN*CC ;

    % detJ
    %
    % Determinante do jacobiano

    detJ = det(J);

    % Np
    %
    % Matriz contendo as funções de forma para a pressão
    %
    % OBS: Elemento triangular com 3 nós.
    % REFERÊNCIA 4.

```

```

%Np = [ 1-epsilon-neta epsilon neta];

% Construção das matrizes locais pelo Método da Integração Numérica de
% Gauss

F_local = F_local + w*MatrizN'*Vetorb*detJ;

% dg(i)
%
% Contadores para a indexação do vetor local de forças nodais
% equivalentes.

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

for r=1:12

F(dg(r)) = F(dg(r)) + F_local(r);

end

end

end

%-----%
% Introdução das condições de contorno dos elementos - Fim %
%-----%

end

```

ElasticConsolidation.m

```

%-----%
%
% ROTINA %
%-----%
% TÍTULO: MÉTODO DOS ELEMENTOS FINITOS MISTOS PARA A SOLUÇÃO DO PROBLEMA %
% DE ACOMPLAMENTO GEOMECÂNICO DADO PELA TEORIA DE BIOT %
%-----%

%-----%
%
% SUB-ROTINA %
%-----%
% TÍTULO: ElasticConsolidation %
% DESCRIÇÃO: Rotina para a solução do problema de adensamento elástico . %
%-----%

```

```

% Programador: Lucíolo Victor (DECIVIL-UFPE) %
% Orientador: Prof. Dr. Ivaldo Dário (DECIVIL - UFPE) %
% Data de criação: 17-11-2015 %
% Última modificação: 23-03-2016 %
%-----%

% O problema de adensamento é dado pelo seguinte sistema de equações
% lineares
%
%  $Ce \cdot dX + Ch = F(t)$ . onde:
%
%  $Ce = \begin{bmatrix} K_f & L \\ L' & 0 \end{bmatrix}$   $Ch = \begin{bmatrix} 0 & 0 \\ 0 & H \end{bmatrix}$ 
%

function [X,Ce] = ElasticConsolidati-
on(K,L,H,MetCon,Nmaxint,DeltaT,Theta,...
CaracPoint,CaracLine,Coord,PropPoint,PropLines,GlpCC,CaracPlan,...
PropElem,Nint0)

% Aux1 e Aux2 são variáveis auxiliares para a construção das matrizes Ch e
% Ce

Aux1 = zeros(size(K));
Aux2 = zeros(size(L));
Aux3 = zeros(size(H));

% Ce
%
% Essa matriz foi definida anteriormente.

Ce = [ K L ;
      L' Aux3 ];

% Ch
%
% Essa matriz foi definida anteriormente.

Ch = [ Aux1 Aux2;
      Aux2' H];

save '-ascii' 'Ce.dat' 'Ce'
save '-ascii' 'Ch.dat' 'Ch'

% O comando a seguir apresenta as soluções para a integração no tempo da
% equação de consolidação.

if strcmp(MetCon,'Theta') == 1

    % Solução da integração direta pelo Método Theta.

    X = ECThetaMethod(Ce,Ch,Nmaxint,DeltaT,Theta,CaracPoint,...
CaracLine,Coord,PropPoint,PropLines,GlpCC,CaracPlan,PropElem,K,H,Nint0);

save '-ascii' 'X.dat' 'X'

elseif strcmp(MetCon,'T&G') == 1

```

```

% Solução da integração direta pelo Método de Thomas e Gladwell

X = ECThomasAndGladwell(Ce,Ch,Fg,Nmaxint,DeltaT,Theta);

end

end

```

ECthetaMethod.m

```

%-----%
%
%                               ROTINA
%-----%
% TÍTULO: MÉTODO DOS ELEMENTOS FINITOS MISTOS PARA A SOLUÇÃO DO PROBLEMA
% DE ACOMPLAMENTO GEOMECÂNICO DADO PELA TEORIA DE BIOT
%-----%

%-----%
%                               SUB-ROTINA
%-----%
% TÍTULO: ECthetaMethod
% DESCRIÇÃO: Rotina para a solução do problema de adensamento elástico
% pelo Método Theta
%-----%
% Programador: Lucíolo Victor (DECIVIL-UFPE)
% Orientador: Prof. Dr. Ivaldo Dário (DECIVIL - UFPE)
% Data de criação: 17-11-2015
% Última modificação: 17-11-2015
%-----%

% Função que soluciona a integração numérica da matriz pelo Método Theta.
% Esse método será dada da seguinte maneira
%
%  $A \cdot X_n = B \cdot X_{n-1} + C$ , onde o termo de C depende de Fg. E são defnidas
% abaixo.

function X = ECthetaMethod(Ce,Ch,Nmaxint,DeltaT,Theta,CaracPoint,...
    CaracLine,Coord,PropPoint,PropLines,GlpCC,CaracPlan,PropElem,K,H,Nint0)

% X
%
% Matriz que fornece os resultados dos graus de liberdade ao longo do
% tempo.

X = zeros(size(Ce,1),Nmaxint);

for i=2:Nmaxint

    A = (Ce + Theta*DeltaT*Ch);

    B = (Ce-(1- Theta)*DeltaT*Ch)*X(:,i-1);

    TermInd = ConstrTermInd(CaracPoint,CaracLine,Coord,PropPoint...
        ,PropLines,GlpCC,CaracPlan,PropElem,K,Ce,i,Nint0,DeltaT);

    %TermInd'

```

```

C = DeltaT*((1-Theta)*TermInd + Theta*TermInd);

Ml = (B + C);

% Introdução das condições de contorno nas matrizes do sistema.

[Af,Mlf] = CondContornoConsolidation(CaracPoint,CaracLine,...
PropPoint,PropLines,GlpCC,A,Ml,K,H);

% save '-ascii' 'Ml.dat' 'Ml'
% save '-ascii' 'A.dat' 'A'
% save '-ascii' 'Mlf.dat' 'Mlf'
% save '-ascii' 'Af.dat' 'Af'
% save '-ascii' 'Ti.dat' 'TermInd'

X(:,i) = inv(Af)*Mlf;

disp(i)
end

disp('t0')
t0 = Nint0*DeltaT;
disp(t0)
disp('tf')
tf = Nmaxint*DeltaT;
disp(tf)
disp('Passo de tempo')
disp(DeltaT)

end

```

ConstrTemInd.m

```

function [TermInd] = ConstrTermInd(CaracPoint,CaracLine,Coord,PropPoint...
,PropLines,GlpCC,CaracPlan,PropElem,K,Ce,intatual,Nint0,DeltaT)

%%

%-----%
% Inicialização das matrizes - Início
%-----%

% TermInd
%
% Inicialização do termo independente.

TermInd = zeros(size(Ce,1),1);

% nglm
%
% Número de graus de liberdades mecânicos.

```

```

nglm = size(K,1);

%-----%
% Inicialização das matrizes - Fim
%-----%

%%

%-----%
% Introdução das condições de contorno das arestas - Início
%-----%

% O comando de repetição abaixo introduz os condições de contorno
% associadas às linhas da geometria

if PropLines ~= 0

for i=1:size(PropLines,1)

% inicialização do vetor de forças locais do elemento

felem = zeros(6,1);

% no1, no2, no3
%
% Nós que formam a aresta avaliada.

no1 = PropLines(i,4);
no2 = PropLines(i,6);
no3 = PropLines(i,5);

% x1, y1, x2, y2, x3, y3
%
% Coordenadas x e y dos três nós que formam a aresta avaliada.

x1 = Coord(no1,1);
y1 = Coord(no1,2);
x2 = Coord(no2,1);
y2 = Coord(no2,2);
x3 = Coord(no3,1);
y3 = Coord(no3,2);

% pg
%
% Matriz que contém os valores da integração de Gauss. A primeira coluna
% contém as coordenadas mapeadas e a segunda os pesos associadas a cada
% coordenada.

pg = [ -0.7746 0.55556;
        0.0000 0.88889;
        0.7746 0.55556 ];

% npg
%
% Número de pontos de Gauss utilizados na integração numérica.

```

```

npg = size(pg,1);

% Numgf
%
% Número do grupo físico que a aresta pertence.

Numgf = PropLines(i,2);
%%

%-----%
% Valores das condições de contorno mecânica nas arestas - Início
%-----%

% fx,fy
%
% Forças distribuídas nas direções x e y.

fx = CaracLine(Numgf,1);
fy = CaracLine(Numgf,2);

% fn,ft
%
% Forças distribuídas nas direções normal e tangencial à aresta avaliada.

fn = CaracLine(Numgf,3);
ft = CaracLine(Numgf,4);

%-----%
% Valores das condições de contorno mecânicas nas arestas - Fim
%-----%

%%

%-----%
% Valores das condições de contorno hidráulicas nas arestas - Início
%-----%
% v
%
% Fluxo imposto na aresta.

v = CaracLine(Numgf,9);

%-----%
% Valores das condições de contorno hidráulicas nas arestas - Fim
%-----%

%%

%-----%
% Introdução das condições de contornos mecânicas nas arestas - Início
%-----%

% O comando de repetição abaixo é relativo à introdução das condições de
% contorno associadas as arestas de contorno.
for j=1:npg

    % S1
    %

```

```

% Valor da coordenada mapeada.

S1 = pg(j,1);

% w
%
% Peso da integração de Gauss.

w = pg(j,2);

% N1,N2,N3
%
% Valores das funções de forma na coordenada S1.

N1 = ((S1^2) - S1)/2;
N2 = 1 - (S1^2);
N3 = ((S1^2) + S1)/2;

% dN1dS1,dN2dS1,dN3dS1
%
% Derivadas das funções de forma em relação à coordenada do mapeamento

dN1dS1 = S1 - 1/2;
dN2dS1 = -2*S1;
dN3dS1 = S1 + 1/2;

% dxdS1,dydS1
%
% derivadas das direções coordenadas em relação à coordenada do
% mapeamento.

dxdS1 = dN1dS1*x1 + dN2dS1*x2 + dN3dS1*x3;
dydS1 = dN1dS1*y1 + dN2dS1*y2 + dN3dS1*y3;

% PrimNMatriz
%
% Primeira matriz associada às funções de forma presente na expressão
% para a integral.

PrimNMatriz = [ N1 0 ;
                0 N1 ;
                N2 0 ;
                0 N2 ;
                N3 0 ;
                0 N3 ];

% MatrizDeriv
%
% Matriz que contém as derivadas, essa matriz é utilizada na integração
% das cargas

MatrizDeriv = [ dxdS1 -dydS1 ;
                dydS1  dxdS1 ];

% Matrizp
%
% Essa matriz é associada as cargas aplicadas na linhas, ela será
% utilizada na integração numérica.

```

```

Matrizp = [ ft ft ft ;
           fn fn fn ];

% dLdS1
%
% Jacobiano das transformadas

dLdS1 = sqrt( dxds1^2 + dyds1^2 );

% SegNMatriz
%
% Segunda matriz associada às funções de forma presente na expressão
% para a integral.

SegNMatriz = [ N1 N2 N3 ]';

% Determinação do vetor de forças nodais equivalentes

felem = felem + PrimNMatriz*MatrizDeriv*Matrizp*SegNMatriz*w + ...
        PrimNMatriz*[ fx fy ]'*dLdS1*w ;

end

% glno1x,glno1y,glno2x,glno2y,glno3x,glno3y
%
% Graus de liberdade para os nós da aresta avaliado onde serão
% introduzidas as forças nodais equivalentes.

glno1x = no1*2-1;
glno1y = no1*2;

glno2x = no2*2-1;
glno2y = no2*2;

glno3x = no3*2-1;
glno3y = no3*2;

% Introdução das condições de contorno no vetor de forças

% dfelem
%
% Esse valor é a derivada das forças aplicadas.

if intatual <= Nint0

    dfelem = felem/(Nint0*DeltaT);

elseif intatual > Nint0

    dfelem = zeros(6,1);

end

TermInd(glno1x) = TermInd(glno1x) + dfelem(1);

```

```

TermInd(glno1y) = TermInd(glno1y) + dfelem(2);
TermInd(glno2x) = TermInd(glno2x) + dfelem(3);
TermInd(glno2y) = TermInd(glno2y) + dfelem(4);
TermInd(glno3x) = TermInd(glno3x) + dfelem(5);
TermInd(glno3y) = TermInd(glno3y) + dfelem(6);

% Obs: é importante salientar que a imposição
% condição de contorno de pressão precisa ser colocada após a condição de
% contorno de vazão pois caso isso não ocorra, é possível que a vazão
% modifique o valor da pressão, se não houver cuidado nas imposições das
% condições de contorno.

%-----%
% Introdução das condições de contornos mecânicas nas arestas - Fim
%-----%

%%

%-----%
% Introdução das condições de contornos hidráulicas nas arestas - Início
%-----%

% noh1, noh2
%
% Primeiro e segundo nó da aresta considerando apenas o problema
% hidráulico.

noh1 = no1;
noh2 = no3;

% xh1, yh1, xh2, yh2
%
% Coordenadas x e y dos nós noh1 e noh2.

xh1 = Coord(noh1,1);
yh1 = Coord(noh1,2);
xh2 = Coord(noh2,1);
yh2 = Coord(noh2,2);

% L
%
% Comprimento da aresta avaliada.

L = sqrt((xh1 - xh2)^2 + (yh1 - yh2)^2);

% vce
%
% Vazão concentrada equivalente.

vce = v*L/2;

% Introdução da vazão concentrada equivalente no vetor independente de
% fluxos.

TermInd(nglm+GlpCC(2,noh1)) = TermInd(nglm+GlpCC(2,noh1)) + vce;
TermInd(nglm+GlpCC(2,noh2)) = TermInd(nglm+GlpCC(2,noh2)) + vce;

%-----%

```

```

% Introdução das condições de contornos hidráulicas nas arestas - Fim
%-----%

end

end

%-----%
% Introdução das condições de contorno das arestas - Fim
%-----%

%%

%-----%
% Introdução das condições de contorno dos pontos - Início
%-----%

% O comando de repetição abaixo introduz as condições de contorno
% associadas aos pontos nodais.

if PropPoint ~= 0

for i=1:size(PropPoint,1)

% NumGfPoint
%
% Número do grupo físico dos pontos.

NumGfPoint = PropPoint(i,2);

% no
%
% Nó avaliado.

no = PropPoint(i,3);

%-----%
% Valores e introdução das condições de contorno mecânicas nos pontos
% - Início
%-----%

% fx,fy
%
% Forças nodais concentradas.

fx = CaracPoint(NumGfPoint,1);
fy = CaracPoint(NumGfPoint,2);

% glnox,glnoy
%
% Graus de liberdade em x e y do nó avaliado.

glnox = no*2 - 1;
glnoy = no*2;

% dfx,dfy
%
% Esse valor é a derivada das forças aplicadas.

```

```

if intatual <= Nint0

    dfx = fx/(Nint0*DeltaT);
    dfy = fy/(Nint0*DeltaT);

elseif intatual > Nint0

    dfx = 0;
    dfy = 0;

end

% Introdução das forças concentradas.

TermInd(glnox) = TermInd(glnox) + dfx;
TermInd(glnoy) = TermInd(glnoy) + dfy;

%-----%
% Valores e introdução das condições de contorno mecânicas nos pontos
% - Fim
%-----%

%%

%-----%
% Valores e introdução das condições de contorno hidráulicas nos pontos
% - Início
%-----%

% v
%
% Valor de fluxo prescrito.

v = CaracPoint(NumGfPoint,7);

% Introdução da condição de contorno na matriz H.

TermInd(nglm + GlpCC(2,no)) = TermInd(nglm + GlpCC(2,no)) + v;

%-----%
% Valores e introdução das condições de contorno hidráulicas nos pontos
% - Fim
%-----%

%%

end

end

%-----%
% Introdução das condições de contorno dos pontos - Fim
%-----%

%%

```

```

%-----%
% Introdução das condições de contorno dos elementos - Início      %
%-----%

for k=1:size(PropElem,1)

    % F_local é o vetor de cargas nodais equivalentes para as forças de corpo
    % para cada elemento avaliado.

    F_local = zeros(12,1);

    % Nós que formam o elemento i
    % Coordenadas de cada nó

    x1 = Coord(PropElem(k,4),1);
    x2 = Coord(PropElem(k,5),1);
    x3 = Coord(PropElem(k,6),1);
    x4 = Coord(PropElem(k,7),1);
    x5 = Coord(PropElem(k,8),1);
    x6 = Coord(PropElem(k,9),1);

    y1 = Coord(PropElem(k,4),2);
    y2 = Coord(PropElem(k,5),2);
    y3 = Coord(PropElem(k,6),2);
    y4 = Coord(PropElem(k,7),2);
    y5 = Coord(PropElem(k,8),2);
    y6 = Coord(PropElem(k,9),2);

    % gfelem
    %
    % Grupo correspondente ao elemento.

    gfelem = PropElem(k,2);

    % bx,by
    %
    % Componentes do vetor de forças de corpo.

    bx = CaracPlan(gfelem,7);
    by = CaracPlan(gfelem,8);

    % Vetorb
    %
    % Vetor de forças de corpo.

    Vetorb = [ bx ; by ];

    % CC
    %
    % Matriz de Coordenadas nodais

    CC=[x1 x2 x3 x4 x5 x6;
        y1 y2 y3 y4 y5 y6]';

    % Essa matriz são os valores dos pontos de Gauss e seus respectivos pesos

    Int_Gauss_Aux = [ 1/6 1/6 1/6
                     1/6 2/3 1/6

```

```

1/6 1/6 2/3];

% npg
%
% Número de pontos de Gauss por elementos

npg = size(Int_Gauss_Aux,1);

for i=1:npg;

    epsilon = Int_Gauss_Aux(i,2) ;
    neta = Int_Gauss_Aux(i,3) ;
    gama = 1 - epsilon - neta;

    w = Int_Gauss_Aux(i,1);

    % N
    %
    % Função de forma mapeada

    N1 = -gama*(1-2*gama);
    N2 = -epsilon*(1-2*epsilon);
    N3 = -neta*(1-2*neta);
    N4 = 4*epsilon*gama;
    N5 = 4*epsilon*neta;
    N6 = 4*neta*gama;

    % MatrizN
    %
    % Matriz das funções de forma

    MatrizN = [ N1 0 N2 0 N3 0 N4 0 N5 0 N6 0;
                0 N1 0 N2 0 N3 0 N4 0 N5 0 N6];

    % GN
    %
    % Gradiente de N

    GN = [ 1-4*gama -1+4*epsilon 0 4*(gama - epsilon)
            4*neta -4*neta;
            1-4*gama 0 -1+4*neta -4*epsilon
            4*epsilon 4*(gama - neta) ];

    % J
    %
    % Matriz jacobiana

    J = GN*CC ;

    % detJ
    %
    % Determinante do jacobiano

    detJ = det(J);

    % Np
    %

```

```

% Matriz contendo as funções de forma para a pressão
%
% OBS: Elemento triangular com 3 nós.
% REFERÊNCIA 4.

%Np = [ 1-epsilon-neta epsilon neta];

% Construção das matrizes locais pelo Método da Integração Numérica de
% Gauss

F_local = F_local + w*MatrizN'*Vetorb*detJ;

% dg(i)
%
% Contadores para a indexação do vetor local de forças nodais
% equivalentes.

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

% dF_local
%
% Esse valor é a derivada das forças aplicadas.

if intatual <= Nint0

    dF_local = F_local/(Nint0*DeltaT);

elseif intatual > Nint0

    dF_local = zeros(size(F_local));

end

for r=1:12

    TermInd(dg(r)) = TermInd(dg(r)) + dF_local(r);

end

end

end

%-----%
% Introdução das condições de contorno dos elementos - Fim %
%-----%

```

```

% qglt
%
% Quantidade de graus de liberdade totais.

qglt = size(Ce,1);

% nglm
%
% Número de graus de liberdades mecânicos.

nglm = size(K,1);

% Correção do termo independente.
TermInd(1:nglm) = -TermInd(1:nglm);
TermInd';
%TermInd(nglm+1:qglt) = -TermInd(nglm+1:qglt);

```

End

CondContornoConsolidation.m

```

%-----%
%                               ROTINA                               %
%-----%
% TÍTULO: MÉTODO DOS ELEMENTOS FINITOS MISTOS PARA A SOLUÇÃO DO PROBLEMA %
% DE ACOMPLAMENTO GEOMECÂNICO DADO PELA TEORIA DE BIOT                %
%-----%

%-----%
%                               SUB-ROTINA                             %
%-----%
% TÍTULO: CondContornoConsolidation                                     %
% DESCRIÇÃO: Rotina para introdução das condições de contorno na matriz %
% de rigidez e para a construção do vetor de forças.                  %
%-----%
% Programador: Lucíolo Victor (DECIVIL-UFPE)                          %
% Orientador: Prof. Dr. Ivaldo Dário (DECIVIL - UFPE)                 %
% Data de criação: 24-02-2016                                           %
% Última modificação: 24-02-2016                                        %
%-----%

function [Af,Mlf] = CondContornoConsolidation(CaracPoint,CaracLine,...
    PropPoint,PropLines,GlpCC,A,Ml,K,H)

%%

%-----%
% Inicialização das matrizes - Início                                  %
%-----%

% Af
%
% Matriz A final.

Af = A;

```

```

% Mlf
%
% Vetor Ml final.

Mlf = Ml;

% nglm,nglh
%
% Número de graus de liberdades mecânicos, número de graus de liberdade
% hidráulicos.

nglm = size(K,1);
nglh = size(H,1);

%-----%
% Inicialização das matrizes - Fim
%-----%

%%

%-----%
% Introdução das condições de contorno das arestas - Início
%-----%

% O comando de repetição abaixo introduz os condições de contorno
% associadas às linhas da geometria.

if PropLines ~= 0

for i=1:size(PropLines,1)

% no1, no2, no3
%
% Nós que formama a aresta avaliada.

no1 = PropLines(i,4);
no2 = PropLines(i,6);
no3 = PropLines(i,5);

% Numgf
%
% Número do grupo físico que a aresta pertence.

Numgf = PropLines(i,2);
%%

%-----%
% Valores das condições de contorno mecânica nas arestas - Início
%-----%

% ux,uy
%
% Deslocamentos prescritos nas direções x e y na aresta avaliada.

ux = CaracLine(Numgf,5);
uy = CaracLine(Numgf,6);

% kx,ky

```

```

%
% São os flags relativos às condições de contorno na aresta avaliada.

kx = CaracLine(Numgf,7);
ky = CaracLine(Numgf,8);

%-----%
% Valores das condições de contorno mecânicas nas arestas - Fim
%-----%

%%

%-----%
% Valores das condições de contorno hidráulicas nas arestas - Início
%-----%

% p
%
% Pressão imposta na aresta.

p = CaracLine(Numgf,10);

% y
%
% Valor de gama para a condição de contorno hidráulica.

y = CaracLine(Numgf,11);

%-----%
% Valores das condições de contorno hidráulicas nas arestas - Fim
%-----%

%%

%-----%
% Introdução das condições de contornos mecânicos nas arestas - Início
%-----%

% auxK
%
% Matriz auxiliar para a introdução dos apoios.

auxA = zeros(nglm+ngh,1);

% glno1x,glno1y,glno2x,glno2y,glno3x,glno3y
%
% Graus de liberdade para os nós da aresta avaliado onde serão
% introduzidas as forças nodais equivalentes.

glno1x = no1*2-1;
glno1y = no1*2;

glno2x = no2*2-1;
glno2y = no2*2;

glno3x = no3*2-1;
glno3y = no3*2;

```

```

% Introdução das condições de contorno mecânica na matriz A. Essa condição
% de contorno é associada à condição de contorno de pressão prescrita.

if kx == 1

Af(glno1x,:) = auxA;
Af(glno2x,:) = auxA;
Af(glno3x,:) = auxA;
Af(:,glno1x) = auxA;
Af(:,glno2x) = auxA;
Af(:,glno3x) = auxA;
Af(glno1x,glno1x) = 1;
Af(glno2x,glno2x) = 1;
Af(glno3x,glno3x) = 1;
Mlf(glno1x) = ux;
Mlf(glno2x) = ux;
Mlf(glno3x) = ux;

end

if ky == 1

Af(glno1y,:) = auxA;
Af(glno2y,:) = auxA;
Af(glno3y,:) = auxA;
Af(:,glno1y) = auxA;
Af(:,glno2y) = auxA;
Af(:,glno3y) = auxA;
Af(glno1y,glno1y) = 1;
Af(glno2y,glno2y) = 1;
Af(glno3y,glno3y) = 1;
Mlf(glno1y) = uy;
Mlf(glno2y) = uy;
Mlf(glno3y) = uy;

end

%-----%
% Introdução das condições de contornos mecânicas nas arestas - Fim
%-----%

%%

%-----%
% Introdução das condições de contornos hidráulicas nas arestas - Início
%-----%

% noh1, noh2
%
% Primeiro e segundo nó da aresta considerando apenas o problema
% hidráulico.

noh1 = no1;
noh2 = no3;

% Introdução da condição de contorno hidráulica na matriz A. Essa condição
% de contorno é associada à condição de contorno de pressão prescrita.

if y == 1

```

```

Af(nglm+GlpCC(2,noh1),:) = auxA;
Af(nglm+GlpCC(2,noh2),:) = auxA;
Af(:,nglm+GlpCC(2,noh1)) = auxA;
Af(:,nglm+GlpCC(2,noh2)) = auxA;
Af(nglm+GlpCC(2,noh1),nglm+GlpCC(2,noh1)) = 1;
Af(nglm+GlpCC(2,noh2),nglm+GlpCC(2,noh2)) = 1;
Mlf(nglm+GlpCC(2,noh1)) = p;
Mlf(nglm+GlpCC(2,noh2)) = p;

end

%-----%
% Introdução das condições de contornos hidráulicas nas arestas - Fim
%-----%

end

end

%-----%
% Introdução das condições de contorno das arestas - Fim
%-----%

%%

%-----%
% Introdução das condições de contorno dos pontos - Início
%-----%

% O comando de repetição abaixo introduz as condições de contorno
% associadas aos pontos nodais.

if PropPoint ~= 0

for i=1:size(PropPoint,1)

% NumGfPoint
%
% Número do grupo físico dos pontos.

NumGfPoint = PropPoint(i,2);

% no
%
% Nó avaliado.

no = PropPoint(i,3);

%-----%
% Valores e introdução das condições de contorno mecânicas nos pontos
% - Início
%-----%

% ux,uy
%
```

```

% Deslocamentos prescritos nos nós.

ux = CaracPoint(NumGfPoint,3);
uy = CaracPoint(NumGfPoint,4);

% kx,ky
%
% Flag associado à restrição de apoio do nó.

kx = CaracPoint(NumGfPoint,5);
ky = CaracPoint(NumGfPoint,6);

% glnox,glnoy
%
% Gruas de liberdade em x e y do nó avaliado.

glnox = no*2 - 1;
glnoy = no*2;

% AuxAp
%
% Essa variável é uma variável auxiliar utilizada para a introdução dos
% apoios para os pontos.

AuxAp = zeros(size(K,1)+size(H,1),1);

% Contribuição na matriz de rigidez global das condições de contorno
% associadas aos nós.

if kx == 1

Af(glnox,:) = AuxAp;
Af(:,glnox) = AuxAp;
Af(glnox,glnox) = 1;
Mlf(glnox) = ux;

end

if ky == 1

Af(glnoy,:) = AuxAp;
Af(:,glnoy) = AuxAp;
Af(glnoy,glnoy) = 1;
Mlf(glnoy) = uy;

end

%-----%
% Valores e introdução das condições de contorno mecânicas nos pontos
% - Fim
%-----%

%%

%-----%
% Valores e introdução das condições de contorno hidráulicas nos pontos
% - Início
%-----%

```

```

% p
%
% Valor de pressãao prescrita no ponto.

p = CaracPoint(NumGfPoint,8);

% y
%
% Valor de gama para o ponto.

y = CaracPoint(NumGfPoint,9);

% Introdução da condição de contorno de fluxo na matriz A. Esse
% procedimento está associado à introdução das pressões prescritas.

if y == 1

Af(nglm + GlpCC(2,no),:) = AuxAp;
Af(:,nglm + GlpCC(2,no)) = AuxAp;
Af(nglm + GlpCC(2,no),nglm + GlpCC(2,no)) = 1;
Mlf(nglm + GlpCC(2,no)) = p;

end

%-----%
% Valores e introdução das condições de contorno hidráulicas nos pontos
% - Fim
%-----%

%%

end

end

%-----%
% Introdução das condições de contorno dos pontos - Fim
%-----%

end

```

PosProcessoBiot.m

```

%-----%
%
% ROTINA
%-----%
% TÍTULO: MÉTODO DOS ELEMENTOS FINITOS MISTOS PARA A SOLUÇÃO DO PROBLEMA
% DE ACOMPLAMENTO GEOMECAÂNICO DADO PELA TEORIA DE BIOT
%-----%

%-----%
%
% SUB-ROTINA
%-----%
% TÍTULO: ProcessoMecMefMistos
% DESCRIÇÃO: Processamento de dados dos arquivos .msh,.geo e da
% entrada de dados para o problema mecânico.
%-----%

```

```

% Programador: Lucíolo Victor (DECIVIL-UFPE) %
% Orientador: Prof. Dr. Ivaldo Dário (DECIVIL - UFPE) %
% Data de criação: 29-10-2015 %
% Última modificação: 18-10-2016 %
%-----%

function PosProcessoBiot(K,Ce,iavall,X,Coord,Glp,PropElem,CaracPlan,...
Imprimir_Desloc,Imprimir_Tens,fator,Imprimir_Pres,Imprimir_Px,...
Imprimir_Py,Imprimir_Desloc_x,Imprimir_Desloc_y,xsup,xinf,ysup,yinf...
,xpinf,xpsup,ypinf,ypsup,Nmaxint,DeltaT,Theta,Nint0)

% Nelem
%
% Número de elementos.

Nelem = size(PropElem,1);

% Nnodes
%
% Números de nós da malha

Nnodes = size(Coord,1);

% Qgld
%
% Quantidade de graus de liberdade de deslocamento

Qgld = size(K,1);

% Qgl
%
% Quantidade de graus de liberdade

Qgl = size(Ce,1);

disp('Número de elementos da malha')
disp(Nelem)
disp('Número de nós da malha')
disp(Nnodes)

for i=1:size(iavall,2)

% iaval
%
% Iteração avaliada

iaval = iavall(i);

disp('ITERAÇÃO AVALIADA')
disp(iaval)

% niaval
%
% Strig correspondente a iaval.

niavalaux = num2str(iaval);

```

```

niaval = strcat('Nit=',niavalaux);

% Criando o diretório correspondente a iaval.

mkdir(niaval)

[TxMax,TxMin,TyMax,TyMin,TxyMax,TxyMin] = PosImprimirMec(Coord,Qgld,...
Nelem,fator,Imprimir_Tens,Imprimir_Desloc,Imprimir_Desloc_x,...
Imprimir_Desloc_y,xsup,xinf,ysup,yinf,PropElem,iaval,X,CaracPlan,niaval);

[MaxP,MinP] = PosImprimirHid(X,Coord,Qgld,Qgl,iaval,Imprimir_Px,xpsup,...
xpinf,Imprimir_Py,ypsup,ypinf,Imprimir_Pres,Glp,Nelem,PropElem,niaval);

% Salvando os resultados do pós processamento

local = strcat('/',niaval,/, 'Resultados.dat');
fileID = fopen( [pwd '/' local], 'wt' );

txmaxima = 'Tensão Máxima em X';
txminima = 'Tensão Mínima em X';
tymaxima = 'Tensão Máxima em Y';
tyminima = 'Tensão Mínima em Y';
txymaxima = 'Tensão Máxima em XY';
txyminima = 'Tensão Mínima em XY';
pmaxima = 'Pressão máxima';
pminima = 'Pressão mínima';
nmi = 'Número Máximo de Iterações';
dtaval = 'Delta t = Intervalo de Tempo adotado';
thetaaval = 'Theta';
n0 = 'Número da iteração Nit0';

fprintf(fileID, '%s = %d\n', txmaxima, TxMax, txminima, TxMin, ...
tymaxima, TyMax, tyminima, TyMin, txymaxima, TxyMax, txyminima...
, TxyMin, pmaxima, MaxP, pminima, MinP, nmi, Nmaxint', dtaval...
, DeltaT, thetaaval, Theta, n0, Nint0);

fclose(fileID);

end

end

```

PosImprimirMec.m

```

%-----%
%
%                      ROTINA
%
%-----%
% TÍTULO: MÉTODO DOS ELEMENTOS FINITOS MISTOS PARA A SOLUÇÃO DO PROBLEMA
% DE ACOMPLAMENTO GEOMECÂNICO DADO PELA TEORIA DE BIOT
%-----%

%-----%
%                      SUB-ROTINA
%
%-----%
% TÍTULO: PosImprimirMec
% DESCRIÇÃO: Pós-processamento de dados para o problema mecânico.
%-----%

```

```

% Programador: Lucíolo Victor (DECIVIL-UFPE) %
% Orientador: Prof. Dr. Ivaldo Dário (DECIVIL - UFPE) %
% Data de criação: 16-08-2016 %
% Última modificação: 16-08-2016 %
%-----

function [TxMax,TxMin,TyMax,TyMin,TxyMax,TxyMin] = PostImprimirMec(Coord...
,Qgld,Nelem,fator,Imprimir_Tens,Imprimir_Desloc,Imprimir_Desloc_x,...
Imprimir_Desloc_y,xsup,xinf,ysup,yinf,PropElem,iaval,X,CaracPlan,niaval)

% Post_Processor
%
% Esta função imprime os resultados da análise

%%
%-----%
%                               Cálculo das Tensões Totais                               %
%-----%

% Inc
%
% Matriz de incidência.

Inc(1,:) = PropElem(:,4)';
Inc(2,:) = PropElem(:,7)';
Inc(3,:) = PropElem(:,5)';
Inc(4,:) = PropElem(:,8)';
Inc(5,:) = PropElem(:,6)';
Inc(6,:) = PropElem(:,9)';

% Construção de dg

for m=1:Nelem

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

end

% Deslocamento
%
% Deslocamentos na iteração i

Deslocamento = X(1:Qgld,iaval);

```

```

for m=1:Nelem

    for i=1:12

        dnelem(i) = Deslocamento(dg(i,m));

    end

    % A matriz DNE guarda os valores dos deslocamentos de cada elemento.

    DNE(m,:) = dnelem;

end

% Comando de repetição com cálculo das tensões nos pontos de Gauss.

for m=1:Nelem

    % Gfelem
    %
    % Essa variável infica o grupo físico que o elemento faz parte

    Gfelem = PropElem(m,2);

    % Ex
    %
    % Módulo de elasticidade na direção x.

    Ex = CaracPlan(Gfelem,1);

    % Ex
    %
    % Módulo de elasticidade na direção y.

    Ey = CaracPlan(Gfelem,2);

    % vxy
    %
    % Coeficiente de Poisson associado às direções x e y.

    vxy = CaracPlan(Gfelem,3);

    % C1,C2,C3
    %
    % Coeficientes para a matriz constitutiva.

    C1 = Ex*(1-vxy)/((1+vxy)*(1-2*vxy));

    C2 = Ey*vxy/((1+vxy)*(1-2*vxy));

    C3 = ((Ex+Ey)/2)/(2*(1+vxy));

    % D
    %
    % matriz constitutiva referente ao estado plano de tensões

    D = [ C1 C2 0 ; C2 C1 0 ; 0 0 C3 ];

```

```

% DeslocElement
%
% Deslocamento no elemento avaliado

DeslocElement = DNE(m,:)';

% Tens_Gauss_Point
%
% Cálculo das tensões nos pontos de Gauss

[X1,Y1,TensX,TensY,TensXY] = Tens_Gauss_Point(m,Coord,Inc,DeslocElement,D);

% XX,YY
%
% Coordenadas.

XX(m,:) = X1';
YY(m,:) = Y1';

% TensX_Elements, TensY_Elements, TensXY_Elements.
%
% Valores de tensões em x, y e xy para os pontos de gauss nos elementos
% avaliadas.

TensX_Elements(m,:) = TensX;
TensY_Elements(m,:) = TensY;
TensXY_Elements(m,:) = TensXY;

end

%-----%
% Impressão a deformada                                     %
%-----%

% Imprimir_Desloc
%
% Essa variável é um flag para a impressão dos deslocamentos, se
% Imprimir_Desloc = 1 os deslocamentos são imprimidos

if Imprimir_Desloc == 1

f1 = figure('Position', [100 100 752 350]);
set(f1,'name','SAEUFPE:Estrutura deformada','numbertitle','off')
clf;

for i=1:Nelem,

    xd1=Coord(Inc(1,i),1)+fator*DNE(i,1);
    xd2=Coord(Inc(2,i),1)+fator*DNE(i,3);
    xd3=Coord(Inc(3,i),1)+fator*DNE(i,5);
    xd4=Coord(Inc(4,i),1)+fator*DNE(i,7);
    xd5=Coord(Inc(5,i),1)+fator*DNE(i,9);
    xd6=Coord(Inc(6,i),1)+fator*DNE(i,11);

    yd1=Coord(Inc(1,i),2)+fator*DNE(i,2);

```

```

yd2=Coord(Inc(2,i),2)+fator*DNE(i,4);
yd3=Coord(Inc(3,i),2)+fator*DNE(i,6);
yd4=Coord(Inc(4,i),2)+fator*DNE(i,8);
yd5=Coord(Inc(5,i),2)+fator*DNE(i,10);
yd6=Coord(Inc(6,i),2)+fator*DNE(i,12);

x1 = Coord(Inc(1,i),1);
x2 = Coord(Inc(2,i),1);
x3 = Coord(Inc(3,i),1);
x4 = Coord(Inc(4,i),1);
x5 = Coord(Inc(5,i),1);
x6 = Coord(Inc(6,i),1);

y1 = Coord(Inc(1,i),2);
y2 = Coord(Inc(2,i),2);
y3 = Coord(Inc(3,i),2);
y4 = Coord(Inc(4,i),2);
y5 = Coord(Inc(5,i),2);
y6 = Coord(Inc(6,i),2);

plot([x1 x2 x3 x4 x5 x6 x1],[y1 y2 y3 y4 y5 y6 y1],'b','LineWidth',2);
hold on
plot([xd1 xd2 xd3 xd4 xd5 xd6 xd1],[yd1 yd2 yd3 yd4 yd5 yd6
yd1],'r','LineWidth',2);hold on;

axis equal;
axis off
end

% Variável avaliada.

caracaval = 'Deslocamento';

% local
%
% Local onde será salva a imagem.

local = strcat('/',niaval,'/',caracaval);

% figaval
%
% Figura que será salva.

figaval = f1;

% salvando a Figura no diretório associado.

saveas(figaval,[pwd local]);

% fechando a figura

close(f1)

end

```

```

%-----%
% Imprimir deslocamentos de uma superfície definida entre xinf e xsup      %
%-----%

% Imprimir_Desloc_x
%
% Essa variável é um flag para a impressão dos deslocamentos ao longo de
% uma superfície x, se Imprimir_Desloc_x = 1 os deslocamentos são
% imprimidos

if Imprimir_Desloc_x == 1

f1 = figure('Position', [100 100 752 350]);
set(f1, 'name', '\dx', 'numbertitle', 'off')

cont = 0;

for i=1:Nelem,

    for j=1:6

        xaux=Coord(Inc(j,i),1);

        if xaux > xinf && xaux < xsup
            cont = cont + 1;
            yp2(cont)=Coord(Inc(j,i),2);
            xp2(cont)=DNE(i,j*2-1);

        end

    end

end

axis equal;
axis off

end

title('\dx')
plot(xp2,yp2,'o')

% Variável avaliada.

caracaval = 'DeslocamentoX';

% local
%
% Local onde será salva a imagem.

local = strcat('/',niaval,'/',caracaval);

% figaval
%
% Figura que será salva.

figaval = f1;

```

```

% salvando a Figura no diretório associado.

saveas(figaval,[pwd local]);

% fechando a figura

close(f1)

end

%-----%
% Imprimir deslocamentos el uma superfície definida entre yinf e ysup      %
%-----%

% Imprimir_Desloc_y
%
% Essa variável é um flag para a impressão dos deslocamentos ao longo de
% uma superfície y, se Imprimir_Desloc_y = 1 os deslocamentos são
% imprimidos

if Imprimir_Desloc_y == 1

f1 = figure('Position', [100 100 752 350]);
set(f1,'name','\dy','numbertitle','off')

cont = 0;

for i=1:Nelem,

    for j=1:6

        yaux=Coord(Inc(j,i),2);

        if yaux > yinf && yaux < ysup
            cont = cont + 1;
            xp1(cont)=Coord(Inc(j,i),1);
            yp1(cont)=DNE(i,j*2);

        end

    end

end

axis equal;
axis off
end

plot(xp1,yp1,'o')
% Variável avaliada.

caracaval = 'DeslocamentoY';

% local
%
% Local onde será salva a imagem.

```

```

local = strcat('/',niaval, '/',caracaval);

% figaval
%
% Figura que será salva.

figaval = f1;

% salvando a Figura no diretório associado.

saveas(figaval,[pwd local]);

% fechando a figura

close(f1)

end

%------%
% Impressão das tensões                                     %
%------%

% Imprimir_Tens
%
% Essa variável é um flag para a impressão das tensões, se
% Imprimir_Tens(1) = 1 as tensões em x serão impressas

if Imprimir_Tens(1) == 1

    %Tensões em direção x

    f1 = figure('Position', [100 100 752 350]);
    set(f1, 'name', '\sigma_x_x contours', 'numbertitle', 'off')

    for e=1:Nelem

        XX1 = [XX(e,1) XX(e,2) XX(e,3) XX(e,1)];
        YY1 = [YY(e,1) YY(e,2) YY(e,3) YY(e,1)];
        ddl = [TensX_Elements(e,1) TensX_Elements(e,2) TensX_Elements(e,3)
TensX_Elements(e,1)];
        patch(XX1,YY1,ddl, 'EdgeColor', 'none');

    end

    axis equal;
    axis off
    title('Tensão X'); xlabel('X'); ylabel('Y'); colorbar

    % Variável avaliada.

caracaval = 'TensaoX';

% local
%
% Local onde será salva a imagem.

```

```

local = strcat('/',niaval, '/',caracaval);

% figaval
%
% Figura que será salva.

figaval = f1;

% salvando a Figura no diretório associado.

saveas(figaval,[pwd local]);

% fechando a figura

close(f1)

end

% Imprimir_Tens
%
% Essa variável é um flag para a impressão das tensões, se
% Imprimir_Tens(2) = 1 as tensões em y serão impressas

if Imprimir_Tens(2) == 1

%Tensões em direção y
f1 = figure('Position', [100 100 752 350]);
set(f1,'name','\sigma_y_y contours','numbertitle','off')

    for e=1:Nelem

        XX1 = [XX(e,1) XX(e,2) XX(e,3) XX(e,1)];
        YY1 = [YY(e,1) YY(e,2) YY(e,3) YY(e,1)];
        dd2 = [TensY_Elements(e,1) TensY_Elements(e,2) TensY_Elements(e,3)
TensY_Elements(e,1)];

        patch(XX1,YY1,dd2,'EdgeColor','none');

    end

axis equal;
axis off
title('Tensão Y'); xlabel('X'); ylabel('Y'); colorbar

% Variável avaliada.

caracaval = 'TensaoY';

% local
%
% Local onde será salva a imagem.

local = strcat('/',niaval, '/',caracaval);

% figaval
%
```

```

% Figura que será salva.

figaval = f1;

% salvando a Figura no diretório associado.

saveas(figaval,[pwd local]);

% fechando a figura

close(f1)

end

% Imprimir_Tens
%
% Essa variável é um flag para a impressão das tensões, se
% Imprimir_Tens(3) = 1 as tensões em xy serão impressas

if Imprimir_Tens(3) == 1
    %Tensão em direção xy
    f1 = figure('Position', [100 100 752 350]);
    set(f1,'name','\sigma_x_y contours','numbertitle','off')

    for e=1:Nelem

        XX1 = [XX(e,1) XX(e,2) XX(e,3) XX(e,1)];
        YY1 = [YY(e,1) YY(e,2) YY(e,3) YY(e,1)];
        dd3 = [TensXY_Elements(e,1) TensXY_Elements(e,2) TensXY_Elements(e,3)
TensXY_Elements(e,1)];

        patch(XX1,YY1,dd3,'EdgeColor','none');
    end

    axis equal;
    axis off
    title('Tensao XY'); xlabel('X'); ylabel('Y'); colorbar

% Variável avaliada.

caracaval = 'TensaoXY';

% local
%
% Local onde será salva a imagem.

local = strcat('/',niaval,'/',caracaval);

% figaval
%
% Figura que será salva.

figaval = f1;

% salvando a Figura no diretório associado.

```

```

saveas(figaval,[pwd local]);

% fechando a figura

close(f1)

end

TxMax = max(max(TensX_Elements));
TxMin = min(min(TensX_Elements));
TyMax = max(max(TensY_Elements));
TyMin = min(min(TensY_Elements));
TxyMax = max(max(TensXY_Elements));
TxyMin = min(min(TensXY_Elements));

disp('Máxima Tensão Normal em X')
disp(TxMax)
disp('Mínima Tensão Normal em X')
disp(TxMin)
disp('Máxima Tensão Normal em Y')
disp(TyMax)
disp('Mínima Tensão Normal em Y')
disp(TyMin)
disp('Máxima Tensão de Cisalhamento')
disp(TxyMax)
disp('Mínima Tensão de Cisalhamento')
disp(TxyMin)

end

```

PosImprimirHid.m

```

%-----%
%                                     %
%                                     ROTINA                                     %
%-----%
% TÍTULO: MÉTODO DOS ELEMENTOS FINITOS MISTOS PARA A SOLUÇÃO DO PROBLEMA %
% DE ACOMPLAMENTO GEOMECÂNICO DADO PELA TEORIA DE BIOT                       %
%-----%

%-----%
%                                     SUB-ROTINA                               %
%-----%
% TÍTULO: PosImprimirHid                                                         %
% DESCRIÇÃO: Pós-processamento de dados para o problema hidráulico.           %
%-----%
% Programador: Lucíolo Victor (DECIVIL-UFPE)                                     %
% Orientador: Prof. Dr. Ivaldo Dário (DECIVIL - UFPE)                           %
% Data de criação: 16-08-2016                                                    %
% Última modificação: 16-08-2016                                                %
%-----%

function [MaxP,MinP] = PosImprimirHid(X,Coord,Qgld,Qgl,iaval,Imprimir_Px...
,xpsup,xpinf,Imprimir_Py,ypsup,ypinf,Imprimir_Pres,Glp,Nelem,PropElem...
,niaval)

%-----%
% Pressões ao longo de uma seção definida por um intervalo em x                %
%-----%

```

```

% Pressões na iteração i

Pi = X((Qgld+1):Qgl,iaval);

if Imprimir_Px == 1

    contx = 0;

    for i=1:size(Pi,1)

        no = Glp(1,i);
        paux = Pi(Glp(2,i));
        cxaux = Coord(no,1);
        cyaux = Coord(no,2);

        if cxaux < xpsup && cxaux > xpinf

            contx = contx + 1;

            pressurey(contx) = paux;
            cy(contx) = cyaux;

        end

    end

    % Plotagem dos valores das pressões ao longo da coordenada x.
    f1 = figure('Position', [100 100 752 350]);
    set(f1,'name','PressureX','numbertitle','off')
    clf;
    plot(pressurey,cy,'o','color','r')

    xlabel('Pressão')
    ylabel('y')
    legend('Solução Numérica','Location','northwest')
    save '-ascii' 'PressureY.dat' 'pressurey'
    save '-ascii' 'PressureYCoord.dat' 'cy'

    % Variável avaliada.

caracaval = 'PressaoX';

% local
%
% Local onde será salva a imagem.

local = strcat('/',niaval,'/',caracaval);

% figaval
%
% Figura que será salva.

figaval = f1;

% salvando a Figura no diretório associado.

```

```

saveas(figaval,[pwd local]);

% fechando a figura

close(f1)

end

%-----%
%   Pressões ao longo de uma seção definida por um intervalo em y   %
%-----%

if Imprimir_Py == 1

    conty = 0;

    for i=1:size(Pi,1)

        no = Glp(1,i);
        cxaux = Coord(no,1);
        cyaux = Coord(no,2);
        paux = Pi(Glp(2,i));

        if cyaux < ypsup && cyaux > ypinf

            conty = conty + 1;

            pressurex(conty) = paux;
            cx(conty) = cxaux;

        end

    end

    % Plotagem dos valores das pressões ao longo da coordenada y.
    f1 = figure('Position', [100 100 752 350]);
    set(f1,'name','PressureY','numbertitle','off')
    clf;
    plot(cx,pressurex,'o','color','r')
    xlabel('x')
    ylabel('Pressão')
    legend('Solução Numérica','Location','northwest')
    save '-ascii' 'PressureX.dat' 'pressurex'
    save '-ascii' 'PressureXCoord.dat' 'cx'
% Variável avaliada.

caracaval = 'PressaoY';

% local
%
% Local onde será salva a imagem.

local = strcat('/',niaval,'/',caracaval);

% figaval
%
% Figura que será salva.

```

```

figaval = f1;

% salvando a Figura no diretório associado.

saveas(figaval,[pwd local]);

% fechando a figura

close(f1)

end

if Imprimir_Pres == 1

f1 = figure('Position', [100 100 752 350]);
set(f1,'name','Pressure','numbertitle','off')
clf;

for k=1:Nelem

% dgp
%
% dgp consiste nos ponteiros relativos a pressão para o elemento avaliado
% na para a construção das matrizes Kup e Kpp

for i=1:size(Glp,2)

if PropElem(k,4) == Glp(1,i)

dgp(1) = Glp(2,i);

end

if PropElem(k,5) == Glp(1,i)

dgp(2) = Glp(2,i);

end

if PropElem(k,6) == Glp(1,i)

dgp(3) = Glp(2,i);

end

end

p1 = Pi(dgp(1));
p2 = Pi(dgp(2));
p3 = Pi(dgp(3));
x1 = Coord(PropElem(k,4),1);
x2 = Coord(PropElem(k,5),1);
x3 = Coord(PropElem(k,6),1);
y1 = Coord(PropElem(k,4),2);
y2 = Coord(PropElem(k,5),2);
y3 = Coord(PropElem(k,6),2);

```

```

% Plotar malha
% plot([x1 x2 x3 x1],[y1 y2 y3 y1],'b','LineWidth',2); hold on

XX1 = [x1 x2 x3 x1];
YY1 = [y1 y2 y3 y1];
P = [p1 p2 p3 p1];

patch(XX1,YY1,P,'EdgeColor','none');
title('Pressure'); colorbar

axis equal;
axis off

end

% Impressão dos valores extremos da pressão

% Variável avaliada.

caracaval = 'Pressao';

% local
%
% Local onde será salva a imagem.

local = strcat('/',niaval, '/',caracaval);

% figaval
%
% Figura que será salva.

figaval = f1;

% salvando a Figura no diretório associado.

saveas(figaval,[pwd local]);

% fechando a figura

close(f1)

end

MaxP = max(Pi);
MinP = min(Pi);
disp('Máxima Pressão')
disp(MaxP)
disp('Mínima Pressão')
disp(MinP)

end

```

APÊNDICE B

Esse apêndice apresenta as soluções analíticas dos problemas apresentados

SOLUÇÃO ANALÍTICA PARA O PROBLEMA DE TERZAGHI

O Problema de Terzaghi consiste em um problema de um meio homogêneo e isotrópico, como permeabilidade E , coeficiente de Poisson ν , permeabilidade k_y , densidade do fluido γ_w , espessura H , sujeito, na sua face superior, por um carregamento f . O problema desenvolve em uma das direções coordenadas. A base da camada permanece impermeável e indelocável ao longo do adensamento e o seu topo, onde é aplicada a carga, se mantém drenante.

As pressões são dadas pela expressão

$$p(y,t) = f \frac{4\gamma}{\pi} \sum_{j=0}^{\infty} \text{sen} \left[\frac{(2j-1)\frac{\pi y}{2H}}{2H} \right] \exp \left[-\frac{(2j-1)^2 \pi^2 c_v t}{4H^2} \right] \quad (\text{A.1})$$

Onde o coeficiente de consolidação é dado por

$$c_v = \frac{k_y E (1-\nu)}{\gamma_w (1+\nu)(1-2\nu)} \quad (\text{A.2})$$

e o fator tempo por

$$T_v = \frac{tc_y E}{H^2} \quad (\text{A.3})$$

A solução da equação foi implementada no código computacional abaixo.

```

%-----%
%-----%
%
%          PROGRAMA DE PÓS GRADUAÇÃO EM ENGENHARIA CIVIL - UFPE
%
%-----%
%-----%

%
%          LMCG - UFPE
%
%-----%
%
%  ÁREA DE CONCENTRAÇÃO: GERENCIAMENTO E SIMULAÇÃO DE RESERVATÓRIOS DE
%  PETRÓLEO
%  ALUNO: LUCÍOLO VICTOR MAGALHÃES E SILVA
%  PROF. ORIENTADOR: IVALDO DÁRIO PONTES SILVA FILHO
%-----%

clc
clear all

```

```

%-----%
%                                     %
%                                     %
%-----%
% TÍTULO: SOLUÇÃO DA EQUAÇÃO DE TERZAGUI PARA O PROBLEMA DE ADENSAMENTO %
% UNIDIMENSIONAL. %
% CRIAÇÃO: 30-11-2015 %
% ÚLTIMA MODIFICAÇÃO: 30-11-2015 %
%-----%

%-----%
%                                     %
%                                     %
%-----%

% E
%
% Módulo de Elasticidade do solo.
% UNIDADE: MPa.

E = 10^7;

% v
%
% Coeficiente de Poisson do solo.

v = 0.20;

% Kv
%
% Permeabilidade vertical do solo.
% UNIDADE: cm/s

%Kv = 10^-6;
Kv = 10^-6;

% H
%
% Altura da camada.
% UNIDADE: m

H = 5;

% yw
%
% Densidade da água.
% UNIDADE: kgf/m^3 ( Ex: água 1000kgf/m^3 )

yw = 1000;

% p0
%
% Poro pressão inicial.
% UNIDADE: Pa.

%p0 = 1000
p0 = 10^3;

```

```

% T
%
% Fator tempo

T = 1;

% nd
%
% Número de divisões da camada.

nd = 40;

%
%%
%-----%
%                                PROCESSAMENTO                                %
%-----%

% Tratamento dos dados para mudança de unidade.

% Módulo de elasticidade de Pa para N/cm^2.

%E = E*0.0001;
%E = 10^7
% Densidade do fluido de kgf/m^3 para N/cm^3.

%yw = yw*(10/(100^3));
%yw= 1500

% Altura da camada e profundidade analisada de m para cm.

%H = 100*H;
%H=5
% cv
%
% Coeficiente de consolidação.

cv = Kv*E*(1-v)/(yw*(1+v)*(1-2*v));

% tempo da análise em função do fator tempo;

t = T*(H^2)/cv

% Dz
%
% Variação na profundidade.

Dz = H/nd;

z = 0:Dz:H;

for i=1:size(z,2)

    pn = 0;

    for n=1:2000

```

```

    pn = pn + (sin((2*n-1)*(pi/2)*((H-z(i))/H))/(2*n-1))*exp(-((2*n-
1)^2)*((pi^2)/4)*cv*(t/(H^2)));

    end

    u(i) = p0*(4/pi)*pn;

end

% Poropressão adimensional

%u = u/p0;

%z = z/H;

hold on

%plot(z(2:size(z,2)),u(2:size(u,2)));

plot(u(2:size(u,2)),z(2:size(z,2)),'-');
legend('Solução Analítica')

```

SOLUÇÃO ANALÍTICA PARA O PROBLEMA DE TERZAGHI MODIFICADO

Esse problema consiste na consolidação de um meio de solo estratificado composto por duas camadas verticias. A primeira camada é definida pelos pontos $y > 0$ e a segunda $y < 0$. A primeira camada possui espessura h_1 , a segunda h_2 . A permeabilidade da camada 1 é dada por k_{y1} e da camada 2 por k_{y2} . Os parâmetros de Lammé da camada 1 são dados por K_1 e G_1 e da camada 2 por K_2 e G_2 . Todo o domínio é saturado por um fluido homogêneo com densidade γ_w . O topo da camada é drenante e submetido a um carregament de valor f e assim como o Problema de Terzaghi a base da camada é impermeável e indeslocável.

A solução analítica das pressões é dada por

$$\begin{aligned}
 y > 0 : \frac{p(y,t)}{f} &= 2 \sum_{j=1}^{\infty} \frac{\cos(x_j) \cos\left(\frac{\beta x_j y}{h_1}\right) - \alpha \operatorname{sen}(x_j) \operatorname{sen}\left(\frac{\beta x_j y}{h_1}\right)}{(1 - \alpha \beta) \cos(\beta x_j) \operatorname{sen}(x_j) + (\alpha + \beta) \operatorname{sen}(\beta x_j) \cos(x_j)} \frac{\exp\left(\frac{-x_j^2 t}{t_2}\right)}{x_j} \\
 y < 0 : \frac{p(y,t)}{f} &= 2 \sum_{j=1}^{\infty} \frac{\cos(x_j) \cos\left(\frac{x_j y}{h_2}\right) - \operatorname{sen}(x_j) \operatorname{sen}\left(\frac{x_j y}{h_2}\right)}{(1 + \alpha \beta) \cos(\beta x_j) \operatorname{sen}(x_j) + (\alpha + \beta) \operatorname{sen}(\beta x_j) \cos(x_j)} \frac{\exp\left(\frac{-x_j^2 t}{t_2}\right)}{x_j}
 \end{aligned} \quad (44)$$

onde o meio é dividido em duas camadas. A camada correspondendo a $y > 0$ é a camada 1, a camada correspondendo a $y < 0$ é a camada 2. Senso assim, os subscritos 1 e 2 nas propriedades apresentadas são relativos às camadas avaliadas.

Os tempos x_j são as raízes da equação

$$-\alpha \operatorname{sen}(\beta x) \operatorname{sen}(x) + \cos(\beta x) \cos(x) = 0 \quad (\text{A.4})$$

onde

$$\beta = \frac{t_1}{t_2} \quad (\text{A.5})$$

que é a relação entre os tempos de consolidação calculados em cada camada, dados por

$$t_1 = \frac{t_1^2}{c_1} \quad (\text{A.6})$$

$$t_2 = \frac{t_2^2}{c_2} \quad (\text{A.7})$$

Os coeficientes de consolidação de cada camada são dados por

$$c_1 = \frac{k_{y1}}{\gamma_w m_1} \quad (\text{A.8})$$

$$c_2 = \frac{k_{y2}}{\gamma_w m_2} \quad (\text{A.9})$$

onde são definidos.

$$\alpha = \sqrt{\frac{k_{y2} m_2}{k_{y1} m_1}} \quad (\text{A.10})$$

$$m_1 = \frac{1}{K_1 + \frac{4}{3} G_1} \quad (\text{A.11})$$

$$m_2 = \frac{1}{K_2 + \frac{4}{3} G_2} \quad (\text{A.12})$$

A seguir é apresentada a implementação dessa solução analítica

```

%-----%
%-----%
%
%          PROGRAMA DE PÓS GRADUAÇÃO EM ENGENHARIA CIVIL - UFPE
%
%-----%
%-----%

%-----%
%          LMCG - UFPE
%-----%
%
%  ÁREA DE CONCENTRAÇÃO: GERENCIAMENTO E SIMULAÇÃO DE RESERVATÓRIOS DE
%  PETRÓLEO
%-----%

```

```

% ALUNO: LUCÍOLO VICTOR MAGALHÃES E SILVA %
% PROF. ORIENTADOR: IVALDO DÁRIO PONTES SILVA FILHO & %
% LEONARDO GUIMARÃES %
%-----%

clc
clear all

%-----%
%                               ROTINA                               %
%-----%
% TÍTULO: SOLUÇÃO DA EQUAÇÃO DE TERZAGUI PARA O PROBLEMA DE ADENSAMENTO %
% UNIDIMENSIONAL PARA MEIOS HETEROGÊNEOSS. %
%
% CRIAÇÃO: 30-11-2015 %
% ÚLTIMA MODIFICAÇÃO: 30-11-2015 %
%-----%

%-----%
%                               ENTRADA DE DADOS DO PROBLEMA          %
%-----%

% k1
%
% Permeabilidade do meio 1, permeabilidade do meio 2.

k1 = 3*10^-6;

k2 = 10^-6;

% y1,y2
%
% Peso específico do meio 1, peso específico do meio 1.

y1 = 1000;

y2 = 1000;

% E1,E2
%
% Módulo de elasticidade do meio 1, módulo de elasticidade do meio 2.

E1 = 10^7;

E2 = 10^7;

% v1,v2
%
% Coeficiente de Poisson do meio 1, Coeficiente de Poisson do meio 2.

v1 = 0.2;

v2 = 0.2;

% h1,h2
%
% Espessura da camada 1, espessura da camada 2.

```

```

h1 = 2;

h2 = 3;

% t
%
% Tempo

t = 22.5;

%2250
%1800
%1350
%900
%450

% p0
%
% Carga aplicada no topo

p0 = 100;

%-----%
%                                PROCESSAMENTO                                %
%-----%

% K1, K2
%
% Módulo volumétrico do meio 1, módulo volumétrico do meio 2.

K1 = E1/(3*(1-2*v1));

K2 = E2/(3*(1-2*v2));

% G1, G2
%
% Módulo de cisalhamento do meio 1, módulo de cisalhamento do meio 2.

G1 = E1/(2*(1+v1));

G2 = E2/(2*(1+v2));

% m1,m2
%
% Compressibilidade do meio 1, compressibilidade do meio 2.

m1 = 1/(K1 + (4/3*G1));

m2 = 1/(K2 + (4/3*G2));

% alpha
%
% Variável auxiliar para a determinação da solução analítica.

```

```

alpha = sqrt(k2*m2)/sqrt(k1*m1);

% c1,c2
%
% Coeficiente de consolidação do meio 1, coeficiente de consolidação do
% meio 2.

c1 = k1/(y1*m1);
c2 = k2/(y2*m2);

% t1,t2
%
% Variáveis auxiliares associadas ao tempo, necessárias para a construção
% da solução analítica.

t1 = (h1^2)/c1;
t2 = (h2^2)/c2;

% beta
%
% Variável auxiliar para a determinação da solução analítica.

beta = sqrt(t1/t2);

% Procedimento para a solução das raízes necessárias para a obtenção das
% soluções analíticas do problema.

% X
%
% Vetor contendo as raízes.

X = zeros(400,1);

x1 = 0;

% Deltax
%
% Incremento em x para avaliação da função.

Deltax = 0.001;

% Inicializando o contador, onde i é o índice associado à raiz.

i = 0;

% Método da bisseção

while i<400

    x2 = x1 + Deltax;

    f1 = -alpha*sin(beta*x1)*sin(x1) + cos(beta*x1)*cos(x1);

    f2 = -alpha*sin(beta*x2)*sin(x2) + cos(beta*x2)*cos(x2);

```

```

if f1*f2 < 0
    i = i+1;

   iaux1 = x1;
iaux2 = x2;

iauxm = iaux1/2+iaux2/2;

fauxm = -alpha*sin(beta*iauxm)*sin(xauxm) +
cos(beta*iauxm)*cos(xauxm);

while norm(fauxm) > 0.001

   iaux1 = -alpha*sin(beta*iaux1)*sin(xaux1) +
cos(beta*iaux1)*cos(xaux1);

   iaux2 = -alpha*sin(beta*iaux2)*sin(xaux2) +
cos(beta*iaux2)*cos(xaux2);

iauxm = iaux1/2+iaux2/2;

fauxm = -alpha*sin(beta*iauxm)*sin(xauxm) +
cos(beta*iauxm)*cos(xauxm);

if faux1*fauxm < 0
   iaux2 = iauxm;

elseif faux2*fauxm < 0
   iaux1 = iauxm;
end

end

X(i) = iauxm;

end

x1 = x2;

end

% As solução são apresentadas em duas expressões para x < 0 e x > 0.

Z = (-h2:0.1:h1);

% Procedimento iterativo para a obtenção da solução analítica, que é dada
% por um somatório.

```

```

for k=1:size(Z,2)

    % z
    %
    % Valor da coordenada vertical avaliada.

    z = Z(k);

    % Obtenção da solução para z >= 0.

    if z >= 0

        Soma = 0;

        for n=1:size(X,1)

            % xx
            %
            % Valor da raiz de número n.

            xx = X(n);

            % Num, Den, Mult
            %
            % Numerador, denominador e multiplicador dos termos
            % correspondentes ao somatório.

            Num = cos(xx)*cos(beta*xx*z/h1) - al-
pha*sin(xx)*sin(beta*xx*z/h1);

            Den = (1+alpha*beta)*cos(beta*xx)*sin(xx) + (al-
pha+beta)*sin(beta*xx)*cos(xx);

            Mult = (1/xx)*exp(-xx*xx*t/t2);

            % Soma
            %
            % Somatório.

            Soma = Soma + Mult*Num/Den;

        end

        % p
        %
        % Pressão associada a z.

        p = 2*p0*Soma;

    elseif z < 0

        % Obtenção da solução para z < 0.

        Soma = 0;

```

```

for n=1:size(X,1)

    % xx
    %
    % Valor da raiz de número n.

    xx = X(n);

    % Num, Den, Mult
    %
    % Numerador, denominador e multiplicador dos termos
    % correnpondentes ao somatório.

    Num = cos(xx)*cos(xx*z/h2) - sin(xx)*sin(xx*z/h2);

    Den = (1+alpha*beta)*cos(beta*xx)*sin(xx) + (al-
pha+beta)*sin(beta*xx)*cos(xx);

    Mult = (1/xx)*exp(-xx*xx*t/t2);

    % Soma
    %
    % Somatório.

    Soma = Soma + Mult*Num/Den;

end

% p
%
% Pressão associada a z.

p = 2*p0*Soma;

end

% P
%
% Vetor contendo as pressões.

P(k) = p;

end

% Plogatem da solução

hold on

plot(P,Z)

disp('fim')

```

SOLUÇÃO ANALÍTICA PARA O PROBLEMA DE MANDEL

O Problema de Mandel consiste em um meio poroelástico sujeito a um carregamento f em seu topo e sua base por uma chapa rígida. O topo e base são impermeáveis. As laterais do domínio são drenantes e o carregamento é nulo. A solução analítica é dada ao longo da linha central do domínio, coincidente com o eixo x considerando a direção horizontal. O domínio possui comprimento em x igual a a , os Parâmetros de Lamé são dados por K e G . O fluido tem peso específico γ_w e k é a permeabilidade do meio

Sendo assim a solução analítica do problema é dada por

$$p(x,t) = f 2\eta \sum_{j=1}^{\infty} \frac{\cos(\xi_j) \left[\cos\left(\frac{\xi_j x}{a}\right) - \cos(\xi_j) \right]}{1 - 2\eta \cos^2(\xi_j)} \exp\left(\frac{-\xi_j^2 c_v t}{a^2}\right) \quad (\text{A.13})$$

O coeficiente de consolidação é dado por

$$c_v = \frac{k \left(K + \frac{4}{3} G \right)}{\gamma_w} \quad (\text{A.14})$$

sendo

$$\eta = \frac{\left(K + \frac{4}{3} G \right)}{2G} \quad (\text{A.15})$$

Os valores de ξ_j são dados pelas raízes da equação não-linear abaixo

$$\frac{\text{sen}(\xi)}{\xi} - 2\eta \xi \cos(\xi) = 0 \quad (45)$$

Abaixo está apresentado o código computacional implementado para a solução do problema

```

%-----%
%-----%
%
%          PROGRAMA DE PÓS GRADUAÇÃO EM ENGENHARIA CIVIL - UFPE
%
%-----%
%-----%
%
%          LMCG - UFPE
%
%-----%
%
%   ÁREA DE CONCENTRAÇÃO: GERENCIAMENTO E SIMULAÇÃO DE RESERVATÓRIOS DE
%   PETRÓLEO
%   ALUNO: LUCÍOLO VICTOR MAGALHÃES E SILVA
%   PROF. ORIENTADOR: IVALDO DÁRIO PONTES SILVA FILHO
%-----%

```

clc

```
clear all
```

```

%-----%
%                               ROTINA                               %
%-----%
% TÍTULO: SOLUÇÃO DA EQUAÇÃO QUE REGE O PROBLEMA DE MANDEL DE ANDESAMENTO %
% BIDIMENSIONAL.                                                    %
% CRIAÇÃO: 19-04-2016                                                %
% ÚLTIMA MODIFICAÇÃO: 19-04-2016                                     %
%-----%

%-----%
%                               ENTRADA DE DADOS DO PROBLEMA        %
%-----%

% E
%
% Módulo de Elasticidade do solo.

E = 10^8;

% v
%
% Coeficiente de Poisson.

v = 0.4;

% yw
%
% Peso específico do fluido.

yw = 1000;

% Kp
%
% Coeficiente de permeabilidades do solo.

Kp = 10^-6;

% t
%
% Fator tempo.

t = 1.0;

% p0
%
% Pressão aplicada no topo da camada.

p0 = 1000;

% comp
%
% Comprimento da metade da camada em y.

comp = 1;

%-----%

```

```

%
%-----PROCESSAMENTO-----%
%
% K
%
% Módulo volumétrico

K = E/(3*(1-2*v));

% G
%
% Módulo de cisalhamento

G = E/(2*(1+v));

% cv
%
% Coeficiente de consolidação

cv = Kp*(K+(4/3)*G)/yw;

% Tempo
%
% Tempo associado ao fator tempo

Tempo = t*(comp^2)/cv;

% eta
%
% Variável utilizada para a solução analítica

eta=(1-v)/(1-2*v);

% Inicialização da variável x(coordenada)

x = 0;

% N
%
% Número de iterações para a construção da solução analítica

N=200;

% nd
%
% Número de divisões para a obtenção da solução analítica

nd = 100;

% dx
%
% Acréscimo de coordenada para a obtenção da solução analítica em vários
% pontos

dx = 1/nd;

% O comando de repetição abaixo é para a construção da solução analítica

```

```

for mm=1:nd

% xx
%
% Variável contendo as coordenadas que serão plotadas

xx(mm) = x;

% eps
%
% Variável auxiliar para a determinação das raízes utilizadas na construção
% da solução analítica

eps=0.000001;

% O comando de repetição abaixo estabelece os valores das raízes utilizadas
% na construção da solução numérica.

for i=1:N

    a1=(i-1)*pi+pi/4;
    a2=a1+pi/2-eps;

    for j=1:40

        y1=tan(a1)-2*eta*a1;
        y2=tan(a2)-2*eta*a2;
        am=(a1+a2)/2;
        ym=tan(am)-2*eta*am;

        if ym*y1>0

            a1=am;

        else

            a2=am;

        end

        if abs(y2)<eps

            am=a2;

        end

        % ZERO
        %
        % Variável que contém as raízes da solução do método não
        % linear.

        ZERO(i)=am;

    end

end

end

```

```

% Inicialização da pressão que é calculada como sendo uma série.

p=0;

for i=1:N

s=sin(ZERO(i));
c=cos(ZERO(i));
cx=cos(x*ZERO(i));
a=2*eta*ZERO(i)*s-(2*eta-1)*c;
b=ZERO(i)*ZERO(i)*t;

%if b<20

    p=p+2*eta*(cx-c)*exp(-b)/a;
    %
    %p=p+2*((cx-c)/(ZERO(i)-s*c))*exp(-((ZERO(i))^2)*t)

%else

    i=N+1;

%end

end

% pp
%
% Vetor que contém as pressões associadas à cada coordenada do vetor xx.

pp(mm) = p;

x = x + dx;

end

% Estabelecimento do valor da condição de contorno em x = comp.

pp(1,nd) = 0;

% A seguir há a desnormalização das pressões e das coordenadas.

xx = xx*comp;
pp = pp*p0;
xx(1,nd) = comp;

% Pós processamento.

hold on

plot(xx,pp)
disp('Tempo equivalente ao Tv')
xlabel('x')
ylabel('Pressure')
title('Mandel is Problem')

```

```
disp(Tempo)
```