

EDUCATIONAL CODE TO COMPUTE PERMEABILITY OF POROUS MATERIALS USING HOMOGENIZATION, MICRO-CT AND FEM

Rafael S. Vianna

Alexsander M. Cunha

rafaelvianna@id.uff.br

amarciano@id.uff.br

Escola de Engenharia, Universidade Federal Fluminense (UFF)

Rua Passo da Pátria 156, 24210-240, Niterói, Rio de Janeiro, Brasil

Rodrigo R. B. V. Azeredo

rbagueira@id.uff.br

Instituto de Química, Universidade Federal Fluminense (UFF)

Rua Outeiro de São João Batista s/n, 24020-141, Niterói, Rio de Janeiro, Brasil

Ricardo Leiderman

Andre M. B. Pereira

leider@ic.uff.br

andremaues@id.uff.br

Instituto de Computação, Universidade Federal Fluminense (UFF)

Rua Passo da Pátria 156, 24210-240, Niterói, Rio de Janeiro, Brasil

Abstract. In this work, we present a description of the method and an educational program to determine permeability of a porous media considering the microstructure of pores provided by micro computed tomography (micro-CT) images using numerical homogenization based on finite element (FE) implementation. Permeability is a very important parameter for transport analysis that is commonly determined in laboratorial tests using the Darcy's law. However, laboratorial tests can be expensive, time-consuming and do not offer a deep understanding of the internal structure of the material. An alternative and efficient strategy to determine permeability of porous materials considering their microstructures is by means of numerical homogenization technique. In this strategy, determining permeability requires dealing with a multiscale problem, where the determination of the macro parameter relies on the simulation of a fluid flowing through canals created by connected pores presented in the material's internal microstructure. This paper aims to explain how to model a Stokes flow through the porous media using FE code implemented in MatLab/Octave to compute materials' permeability. The boundary value problem is modeled with periodic boundary conditions and a unit gradient pressure applied in the desired direction to obtain the macro parameter. The velocity field caused by the prescribed pressure gradient is determined solving the Stokes equation. Lastly, the micro and macro scales are linked together with the average volume flow rate and the unit pressure gradient using Darcy's law. The accuracy of the method depends on reliable computational models, therefore micro-CT has an important role in numerical homogenization generating representative virtual models for numerical simulations.

Keywords: Permeability, Finite elements, Numerical homogenization, Multiscale, Micro-CT

1 Introdução

A permeabilidade é um importante macroparâmetro que é geralmente determinado no processo de caracterização de meios porosos. Este parâmetro permite quantificar a capacidade que um material possui de permitir o escoamento de fluidos. Por isso, sua grande relevância em diversas áreas, tais como no estudo de aquíferos, na mecânica dos solos, na exploração de petróleo, petrofísica, etc. A permeabilidade é usualmente obtida experimentalmente, por meio de ensaios convencionais em laboratórios, aplicando-se a lei de Darcy. Contudo, esses ensaios laboratoriais podem ser custosos, além de não oferecerem um entendimento aprofundado da microestrutura interna do material.

As simulações numéricas vêm sendo cada vez mais utilizadas como metodologias alternativas aos ensaios laboratoriais, visando contornar diversas limitações que comumente ocorrem em experimentos físicos realizados em laboratórios. Simulações numéricas baseadas em métodos computacionais passaram a ser estratégias muito atrativas para avaliar a permeabilidade de materiais porosos, pois a partir dessas técnicas é possível considerar a microestrutura do material, com a geração de modelos computacionais fiéis ao meio poroso investigado. Os trabalhos de Andreassen e Andreassen [1], Aarnes et al. [2] e Sheng e Zhi [3] são alguns exemplos que ilustram o uso dos métodos dos elementos finitos, dos volumes finitos e de lattice-Boltzmann no cálculo da permeabilidade por meio de homogeneização computacional.

O desenvolvimento de um código computacional para determinar a permeabilidade de um material poroso possui várias peculiaridades. Encontrar na literatura documentos que abordem as nuances da programação desse problema não é uma tarefa fácil. Além disso, grande parte dos códigos encontrados na literatura, que se propuseram a apresentar suas metodologias de forma educacional, tem em comum o fato de priorizarem a eficiência computacional, o que os leva, em grande parte das vezes, a omitir etapas importantes para o processo de cognição de um leitor que não tem vasta experiência com todos os diversos assuntos envolvidos no processo de determinação da permeabilidade. Apesar de se enquadrar neste contexto, o trabalho de Andreassen e Andreassen [1] é ao mesmo tempo um dos mais relevantes, pois os autores descrevem um programa educacional compacto, que adota um conjunto de estratégias interessantes, o qual foi desenvolvido em MatLab, por ser uma plataforma de programação simples e eficiente. Por tais razões, utilizou-se o trabalho de Andreassen e Andreassen [1] como ponto de partida e referência básica para os desenvolvimentos propostos neste artigo.

Neste trabalho, tem-se como objetivo principal apresentar o passo a passo do desenvolvimento e implementação de um programa educacional para determinar a permeabilidade de materiais porosos, baseando-se nos conceitos de homogeneização numérica e no Método dos Elementos Finitos (MEF) considerando a microestrutura porosa desses materiais adquirida a partir de micro-tomografia computadorizada (micro-CT). O programa se propõe basicamente a determinar o valor médio do campo de velocidades em um fluido escoando através dos poros conectados de um determinado material. Assume-se que esse escoamento é governado pela equação de Stokes que é resolvida com o MEF. O programa também consta com a implementação de condições de contorno periódicas, em que o pressuposto básico por trás do uso de tais condições de contorno é que o modelo numérico é um volume representativo de um meio (estatisticamente homogêneo) não contornado, o que permite acelerar a convergência das propriedades avaliadas em direção às propriedades efetivas do material.

É importante mencionar que apesar deste trabalho estar inserido no contexto de petrofísica digital, no qual os materiais de interesse apresentam arranjos dos poros intrinsecamente com características tridimensionais, com as propriedades efetivas devendo ser determinadas preferencialmente por modelos 3-D, optou-se, por questões didáticas, apresentar uma implementação para modelos 2-D. Porém, ressalta-se que todos os desenvolvimentos aqui descritos são simples e diretamente extensíveis para análises com modelos 3-D.

2 Micro-tomografia computadorizada de raios-X

Uma vez que este trabalho se propõe a utilizar uma metodologia numérica que considera a microestrutura do material, a convergência dos resultados numéricos com os resultados experimentais tem dependência direta na fidelidade da representação do modelo computacional. Uma das formas de criar modelos computacionais fidedignos é pelo uso de micro-tomografia computadorizada de raios-X (micro-CT). A micro-CT é uma técnica utilizada para a visualização tridimensional interna de qualquer material na escala de micrometros. Esta técnica consiste na geração de múltiplas projeções radiográficas tiradas de diversos ângulos para a produção de uma imagem 3-D da estrutura interna de um objeto. A geração de uma micro-CT envolve diversas etapas: preparação e montagem da amostra, aquisição das projeções, reconstrução, processamento de imagem e segmentação, geração da malha do modelo e simulações computacionais, como ilustrado na Figura 1.

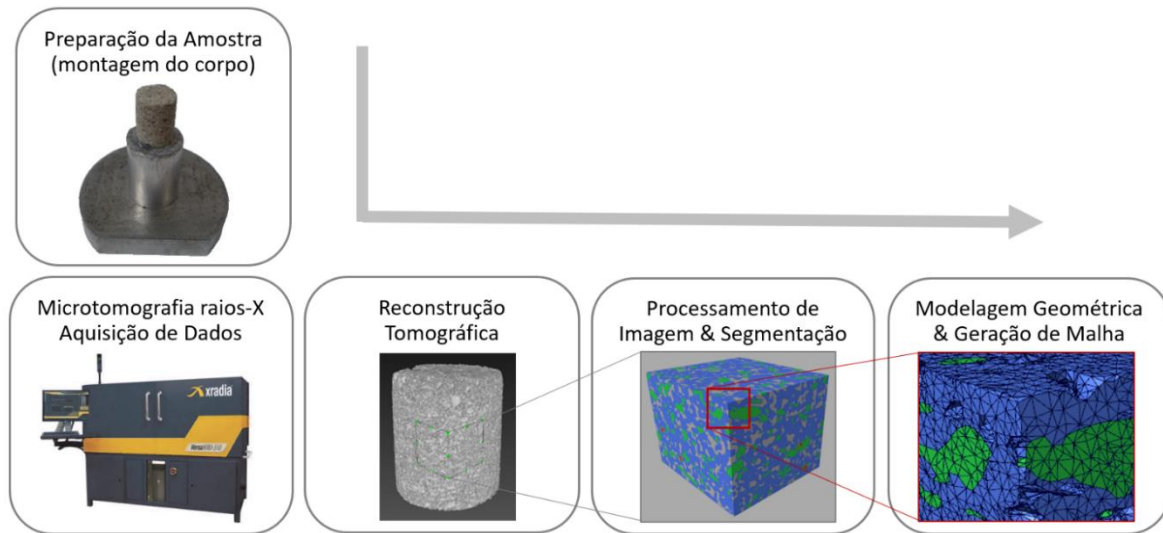


Figura 1. Fluxo de trabalho em petrofísica digital, desde a preparação do material até obtenção dos resultados numéricos, passando pela montagem da amostra, aquisição de dados, reconstrução, segmentação, geração da malha do modelo e simulações computacionais.

2.1 Aquisição das projeções e reconstrução da micro-CT

No processo de aquisição de uma micro-CT, a amostra de material a ser micro-tomografada é posicionada em uma base giratória dentro do micro-tomógrafo que rotaciona em torno de seu próprio eixo a cada radiografia realizada. As projeções radiográficas são então utilizadas por um algoritmo de reconstrução para a criação de uma imagem 3-D do objeto. A reconstrução é feita por uma análise inversa que determina o nível de absorção de raios-X de um ponto espacial do material baseando-se nos níveis de absorção da projeção deste ponto nas imagens radiografadas. O resultado de uma micro-CT é uma imagem digital matricial 3-D de um objeto. Uma imagem digital matricial é uma representação visual de um objeto formada por pequenos elementos chamados de voxel. Cada voxel recebe um valor específico que será responsável por atribuir a ele uma cor. Na Figura 2, ilustra-se o resultado de uma micro-CT realizada pelos autores de uma amostra de arenito.

2.2 Processamento de imagens e segmentação

Após digitalização e reconstrução tomográfica, um objeto virtual 3-D é gerado. A visualização da microestrutura interna do material pode ser realizada por planos de corte, como ilustrado na Figura 2. Todavia, para análises mais refinadas, como a determinação de porosidade, permeabilidade e outras propriedades efetivas, as imagens devem ser processadas. O processamento de imagens inclui: seleção da região de interesse, aplicação de filtros para tratamento de ruídos e segmentação da imagem. Para determinar a permeabilidade do arenito a partir de modelos 2-D, procurou-se escolher uma fatia

(imagem 2-D) que pudesse ser representativa. Após a escolha da imagem, foram utilizados filtros de imagem para retirada de ruído. Os filtros são ferramentas que reduzem as interferências, artefatos e aprimoram determinadas características das imagens. Eles são baseados em algoritmos sofisticados que facilitam a extração de determinadas informações de uma imagem, pela modificação do histograma de cinzas da imagem.

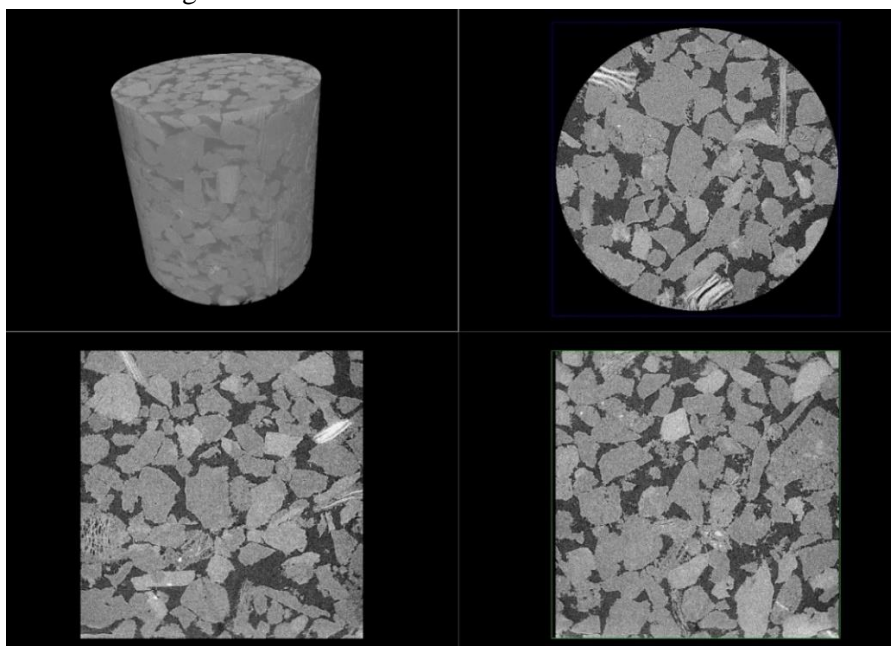


Figura 2. Micro-CT de uma amostra de arenito.

Como última etapa, foi realizada a segmentação da imagem. A segmentação consiste na determinação das fases do material de acordo com os tons cinza dos voxels da micro-CT. Este processo pode ser realizado por meio do histograma da imagem. Existem diferentes estratégias que podem ser utilizadas no processo de segmentação de imagens. Apesar de dispor de recursos de fácil aplicação, o processo de segmentação pode exigir muito tempo e experiência para uma boa segmentação. Após a segmentação, utilizamos ainda uma ferramenta para fazer a separação de grãos. A utilização da ferramenta de separação dos grãos permite realizar uma análise mais rica do fluxo de fluido através dos grãos, que é essencial para a determinação de permeabilidade. A partir da Figura 3 é possível fazer uma comparação entre o antes e o depois das etapas de processamento e segmentação da imagem da seção escolhida, onde o azul representa a região sólida e o preto a região conectada porosa em que o fluido escoar.

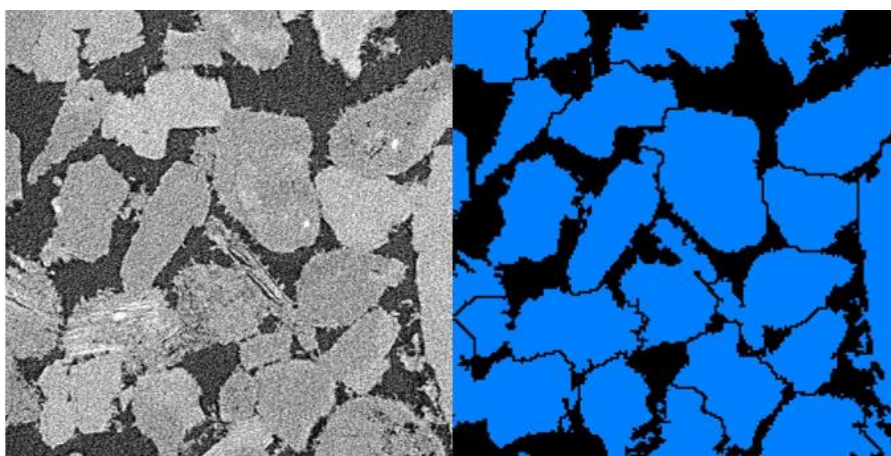


Figura 3. Fatia selecionada: antes e depois das etapas de processamento e segmentação da imagem.

3 Permeabilidade via homogeneização computacional

A determinação da permeabilidade por meio da análise da microestrutura do material passa a ser um problema de multiescala, em que a determinação do macroparâmetro depende da determinação do campo de velocidades que é função da microestrutura do meio. Portanto, resolver o problema de permeabilidade de um meio está ligado à resolução das equações governantes do problema na microescala. Logo, é necessário utilizar uma metodologia que possibilite relacionar as informações da microescala com a macroescala. Por conseguinte, técnicas de homogeneização devem ser utilizadas.

As técnicas de homogeneização computacional buscam determinar relações entre a micro e a macroestrutura do material, que consistem na determinação de propriedades efetivas de volumes elementares representativos (REV) por meio das propriedades de elementos na escala microscópica. Para a determinação da permeabilidade de um meio poroso, aplica-se um gradiente de pressão conhecido no REV desse material. A aplicação desse gradiente de pressão induzirá um campo de velocidades no domínio. A propriedade efetiva homogeneizada do material $\bar{\mathbf{K}}$ é definida como a relação entre o gradiente de pressão aplicado no REV $\langle \nabla p \rangle_\Omega$ e a média das velocidades resultantes na escala microscópica $\langle \mathbf{u} \rangle_\Omega$, conhecida como lei de Darcy (Ω denota o volume representativo):

$$\langle \mathbf{u} \rangle_\Omega = \bar{\mathbf{K}} : \langle \nabla p \rangle_\Omega \quad (1)$$

em que,

$$\langle \mathbf{u} \rangle_\Omega = \frac{1}{\Omega} \int_\Omega \mathbf{u} \, d\Omega \quad (2)$$

Considerando um problema bidimensional, a matriz $\bar{\mathbf{K}}$ é obtida por meio de duas análises, nas quais se aplica um gradiente de pressão unitário por direção. A média do vetor de velocidades determinada em cada análise equivale às colunas da matriz $\bar{\mathbf{K}}$ relacionadas à direção do gradiente de pressão aplicado em cada caso. Para determinar o campo de velocidades no domínio dos problemas, consideram-se que os mesmos são governados pela equação de Stokes, utilizando-se o MEF, como descrito com mais detalhes na próxima seção.

4 Fluido-dinâmica computacional com o MEF

As equações que governam o problema de escoamento de fluidos na microescala são a equação da continuidade e a equação de Navier-Stokes. Este trabalho se propõe a avaliar fluidos viscosos de baixa velocidade. Como consequência do baixo número de Reynolds, os efeitos convectivos podem ser negligenciados e, então, a equação de Navier-Stokes é simplificada para a equação de Stokes. Entretanto, para a grande maioria das aplicações práticas, essas equações diferenciais que governam o problema não possuem resultados analíticos, e por isso são necessários métodos numéricos, como o método dos elementos finitos (MEF), para sua resolução.

O MEF se propõe a determinar a solução do problema de forma aproximada para um conjunto finito de elementos conectados por um número finito de pontos (comumente chamados de nós). O domínio contínuo do problema é representado por funções interpoladoras, chamadas de funções de forma, que interpolam os valores das variáveis calculados nos nós dos elementos. O MEF tem como um de seus requisitos básicos a transformação da forma diferencial (forma forte) das equações governantes do problema em equações integrais (forma fraca). O processo pode ser feito pela aplicação do método dos resíduos ponderados, que permite aproximar a solução exata do problema por aquela que tem os resíduos nulos das equações governantes, ponderados por funções contínuas e diferenciáveis, de forma integral no domínio de interesse. Dessa forma, o problema de valor de contorno do meio contínuo pode ser solucionado pela média de valores de um intervalo definido.

As equações governantes do problema da dinâmica do fluido de Stokes são definidas por:

$$\mu \nabla^2 \mathbf{u} = -\nabla p + \mathbf{b} \quad (3)$$

$$\nabla \cdot \mathbf{u} = 0 \quad (4)$$

Ao aplicar o método dos resíduos ponderados, multiplicam-se as equações governantes por funções de ponderação ψ_i . O somatório dos resíduos gerados pela aproximação é considerado nulo dentro de um intervalo definido:

$$\int_{\Omega_e} \psi_1 \left(\mu \frac{\partial^2 u}{\partial x^2} + \frac{\partial p}{\partial x} \right) dx dy = 0 \quad (5)$$

$$\int_{\Omega_e} \psi_2 \left(\mu \frac{\partial^2 v}{\partial y^2} + \frac{\partial p}{\partial y} \right) dx dy = 0 \quad (6)$$

$$\int_{\Omega_e} \psi_3 \left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right) dx dy = 0 \quad (7)$$

Utilizado o teorema da divergência no primeiro termo da integral das Eqs. (5) e (6), é possível reduzir a ordem da derivada e externalizar o gradiente da velocidade:

$$\psi_1 \mu \frac{\partial u}{\partial x} - \int_{\Omega_e} \mu \frac{\partial u}{\partial x} \frac{\partial \psi_1}{\partial x} dx dy = - \int_{\Omega_e} \psi_1 \frac{\partial p}{\partial x} dx dy \quad (8)$$

$$\psi_2 \mu \frac{\partial v}{\partial y} - \int_{\Omega_e} \mu \frac{\partial v}{\partial y} \frac{\partial \psi_2}{\partial y} dx dy = - \int_{\Omega_e} \psi_2 \frac{\partial p}{\partial y} dx dy \quad (9)$$

Diferentes funções de ponderação podem ser utilizadas no método dos resíduos ponderados. Uma das formas mais comuns de utilizar a função de ponderação é atribuir a ela as funções de aproximação dos valores nodais, as funções de forma. Esse método é conhecido como o método de Galerkin. A matriz de funções de forma é chamada de matriz \mathbf{N} . Nas formulações clássicas do MEF, surge uma matriz cujos elementos contêm derivadas das funções de forma, sendo conhecida como matriz \mathbf{B} . Essas matrizes estabelecem as relações:

$$u = \mathbf{N}_1 \bar{\mathbf{u}} \quad (10)$$

$$v = \mathbf{N}_2 \bar{\mathbf{v}} \quad (11)$$

$$p = \mathbf{N}_3 \bar{\mathbf{p}} \quad (12)$$

$$\mathbf{B} = d\mathbf{N} \quad (13)$$

em que os valores $\bar{\mathbf{u}}$, $\bar{\mathbf{v}}$ e $\bar{\mathbf{p}}$ representam os valores nodais de u , v e p .

De acordo com Reddy [4], para que seja feita uma analogia com o princípio das potências virtuais, as funções de ponderação ψ_1 e ψ_2 podem ser entendidas como velocidades virtuais tomadas respectivamente por u e v . Portanto, $\psi_1 = \mathbf{N}_1 \bar{\mathbf{u}}$ e $\psi_2 = \mathbf{N}_2 \bar{\mathbf{v}}$. A função de ponderação ψ_3 presente na Eq. (7) é associada à mudança de volume do elemento fluido, portanto ψ_3 deve ser entendida como uma pressão hidrostática virtual que causa essa variação de volume. Dessa forma, definimos $\psi_3 = \mathbf{N}_3 \bar{\mathbf{p}}$.

Após a discretização do domínio, a forma fraca das equações governantes do problema pode ser representada em função das matrizes \mathbf{N} e \mathbf{B} . As Eqs. (5)-(7) devem ser reescritas como:

$$\int_{\Omega_e} \mathbf{B}_1^T \mu \mathbf{B}_1 \mathbf{u} dx dy + \int_{\Omega_e} \mathbf{N}_1^T \mathbf{B}_3 \mathbf{p} dx dy = -\mathbf{N}^T \mu \mathbf{B} \mathbf{u} \quad (14)$$

$$\int_{\Omega_e} \mathbf{B}_2^T \mu \mathbf{B}_2 \mathbf{u} dx dy + \int_{\Omega_e} \mathbf{N}_2^T \mathbf{B}_3 \mathbf{p} dx dy = -\mathbf{N}^T \mu \mathbf{B} \mathbf{u} \quad (15)$$

$$\int_{\Omega_e} \mathbf{N}_3 \mathbf{B} \mathbf{u} \, dx dy = \mathbf{0} \quad (16)$$

Os termos do lado direito das equações de conservação de momentum são resultado do efeito das condições de contorno de cada elemento. Portanto, para os elementos internos à malha, o termo do lado direito da equação equivale a 0. Para elementos com condições de contorno, o termo do lado direito equivale à potência virtual aplicada no contorno do elemento e será chamado de \mathbf{f} .

Resolver o sistema de equações diferenciais parciais pelo MEF se resume a resolver o sistema de equações algébricas:

$$\begin{bmatrix} \mathbf{K} & \mathbf{G} \\ \mathbf{D} & \mathbf{0} \end{bmatrix} \begin{Bmatrix} \mathbf{u} \\ \mathbf{p} \end{Bmatrix} = \begin{Bmatrix} \mathbf{f} \\ \mathbf{0} \end{Bmatrix} \quad (17)$$

em que \mathbf{K} é a matriz de viscosidade, dada por:

$$\mathbf{K} = \int_{\Omega_e} \mathbf{B}^T \mu \mathbf{B} \, dx dy \quad (18)$$

\mathbf{G} representa a matriz dos gradientes, dada por:

$$\mathbf{G} = \int_{\Omega_e} \mathbf{B}^T \mathbf{N}_3 \, dx dy \quad (19)$$

e \mathbf{D} é a matriz do divergente das velocidades nodais, dada por:

$$\mathbf{D} = \int_{\Omega_e} \mathbf{N}_3^T \mathbf{B} \, dx dy \quad (20)$$

O modelo acima é conhecido como formulação mista pois possui como variáveis os campos de pressão e velocidade. Considerando que, usualmente, o problema de dinâmica dos fluidos possui condições de contorno em termos de velocidade e pressão, esta acaba se tornando a formulação mais intuitiva para determinar a solução do problema. Entretanto, o acoplamento das variáveis de pressão com as variáveis de velocidade traz novas dificuldades à resolução do sistema.

Uma forma de interpretar o acoplamento entre velocidade e pressão é entender a equação da continuidade como uma restrição de incompressibilidade às equações de conservação de momentum. Devido ao fato de que a restrição de incompressibilidade não depende das variáveis de pressão, a construção de um modelo de elementos finitos acaba se tornando complicada. A solução do sistema de forma direta não pode ser realizada uma vez que a matriz dos coeficientes das variáveis possui uma submatriz com coeficientes nulos na diagonal, o que a torna singular. Dessa forma, torna-se necessário o uso de outras estratégias de resolução do sistema de equações.

Outra dificuldade comum para esse modelo é a escolha das funções de aproximação dos domínios. Escolher uma aproximação para o campo de pressão igual ao do campo de velocidades pode provocar instabilidade. Portanto, a confiabilidade do modelo computacional está diretamente ligada à escolha da aproximação dos domínios do problema.

4.1 Aproximação Q1-Q1 com estabilização

A relação entre a segunda derivada do campo de velocidades e a primeira derivada do campo de pressões explicitada na equação de Stokes nos mostra que o campo de velocidades deve ser um grau maior que o de pressões. A utilização de funções de aproximação de mesma ordem para o campo de pressões e velocidades gera instabilidade na resolução do problema. Essa instabilidade pode ser explicada pela análise de solvabilidade do sistema de equações. A solvabilidade do sistema de equações é explicada por diversos autores como a garantia da condição inf-sup, também conhecida como condição LBB, graças aos trabalhos de Ladyzhenskaya [5], Babuska [6] e Brezzi [7].

Utilizar funções de forma de mesma ordem não garante que a condição inf-sup seja atingida. Para que o sistema possua solução única, outros métodos de aproximação dos campos de velocidade e pressão devem ser utilizados. Na literatura são encontrados outros métodos de aproximação que são recomendados para a formulação mista. Bathe [8] e Elman et al. [9] citam as seguintes aproximações

como satisfatórias da condição inf-sup: aproximação Q2-Q1 (aproximação bi quadrática para velocidade e bi linear para pressão), também conhecida como método de Taylor-Hood; aproximação Q2-P1 (aproximação bi quadrática para velocidade e linear para pressão em cada elemento sem garantir continuidade); e a aproximação Q2-P0 (aproximação bi quadrática para velocidade e constante para pressão em cada elemento).

Entretanto, utilizar funções de forma de mesma ordem para interpolar os campos de pressão e velocidades possui diversas vantagens do ponto de vista da implementação computacional. A fim de utilizar funções de forma de mesma ordem para os campos de velocidade e pressão, e eliminar a instabilidade, técnicas de estabilização podem ser adotadas. As técnicas de estabilização têm como ideia básica relaxar a restrição de incompressibilidade, garantindo a condição inf-sup. Alguns autores propuseram diferentes metodologias que se mostraram bastante eficientes na estabilização de diferentes modelos de aproximação.

Uma das metodologias de discretização mais simples é a utilização de funções de forma lineares para os domínios de velocidades e pressão. Esse tipo de aproximação é conhecido como aproximação Q1-Q1. A aproximação Q1-Q1, além de ser de fácil implementação, produz menor custo computacional para resolução do sistema de equações. Por esse motivo, muitos autores buscam utilizar técnicas de estabilização para fazer uso desse tipo de discretização. Como exemplo, podemos citar os trabalhos de Andreassen e Andreasen [1], Karim et al. [10], Braack e Schieweck [11], Codina e Blasco [12] e Becker e Hansbo [13]. Para construção de um modelo computacional estável de discretização Q1-Q1, este trabalho seguirá a mesma estabilização adotada por Andreassen e Andreasen [1]. Segundo Andreassen e Andreasen [1], a estabilização do modelo Q1-Q1 pode ser obtida adicionado um novo termo com um coeficiente de estabilização na equação da continuidade. A forma fraca da equação da continuidade já com o termo de estabilização é definida por:

$$\int_{\Omega} \mathbf{N}^T \mathbf{B} \mathbf{u} \, dx dy - \int_{\Omega} \tau \mathbf{N}^T \mathbf{N} \mathbf{p} \, dx dy = \mathbf{0} \quad (21)$$

em que τ é o parâmetro de estabilização da pressão definido como $\tau = h^2/12$ para elementos quadriláteros, sendo h o tamanho da maior diagonal do elemento finito.

A integral com o termo de estabilização da pressão resulta numa matriz \mathbf{P} que deve ser adicionada no sistema de equações da Eq. (17), resultando em:

$$\begin{bmatrix} \mathbf{K} & \mathbf{G} \\ \mathbf{D} & \mathbf{P} \end{bmatrix} \begin{Bmatrix} \mathbf{u} \\ \mathbf{p} \end{Bmatrix} = \begin{Bmatrix} \mathbf{f} \\ \mathbf{0} \end{Bmatrix} \quad (22)$$

A utilização de funções de forma iguais para os campos de velocidade e pressão faz com que a matriz dos divergentes seja igual à inversa da matriz dos gradientes, portanto o sistema de equações pode ser reescrito como:

$$\begin{bmatrix} \mathbf{K} & \mathbf{G} \\ \mathbf{G}^T & \mathbf{P} \end{bmatrix} \begin{Bmatrix} \mathbf{u} \\ \mathbf{p} \end{Bmatrix} = \begin{Bmatrix} \mathbf{f} \\ \mathbf{0} \end{Bmatrix} \quad (23)$$

A introdução do novo termo na equação da continuidade faz com que a matriz de coeficientes se torne positiva definida. Isso torna possível a solução direta do problema.

4.2 Condições de contorno do problema

Ao passo que se aumenta o volume a ser analisado, a propriedade medida para esse volume se aproxima da propriedade efetiva do material até que se obtenha um volume que pode ser considerado representativo. Para que as propriedades medidas converjam para a propriedade efetiva mais rapidamente, isto é, com volumes menores, é interessante utilizar condições de contorno periódicas. Esse tipo de condição de contorno simula o volume de interesse dentro de uma região periódica, ou seja, o volume de interesse se repete em todas as direções, como mostrado na Figura 4. Dessa forma, é possível obter resultados de meios homogêneos e periódicos pela modelagem de volumes menores,

reduzindo assim o esforço computacional. A implementação computacional de condições de contorno periódicas se dá de forma que os graus de liberdade das bordas opostas possuam a mesma numeração.

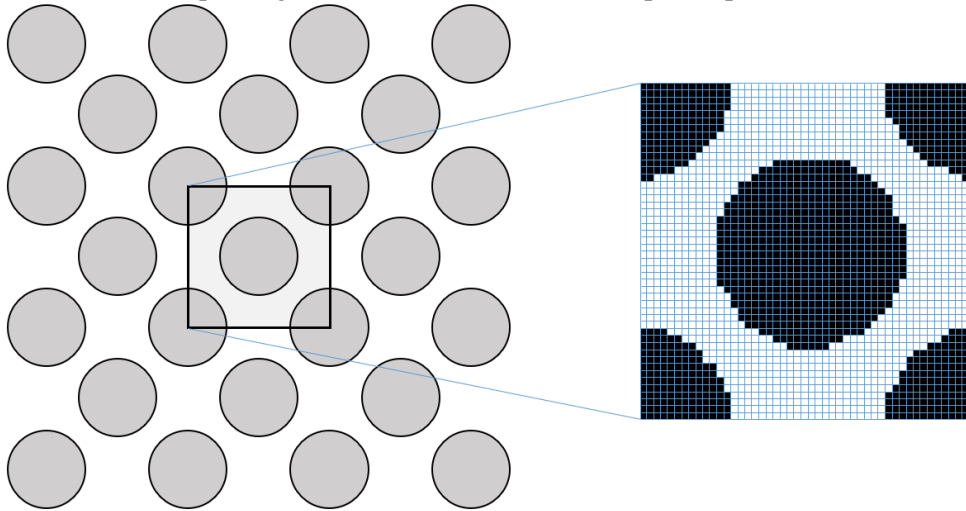


Figura 4. Representação de um meio periódico por uma imagem.

A simulação de escoamentos em um meio poroso impõe desafios à modelagem do problema. A modelagem do meio poroso realizada neste trabalho se baseia na modelagem de regiões de fluido, ignorando os elementos sólidos representados na imagem. Isso faz com que seja necessária a imposição da seguinte condição nas regiões de interface sólido-fluido: os graus de liberdade relativos às componentes de velocidades nas direções x e y nos nós da interface recebem valor nulo.

5 Implementação em Octave

5.1 Entrada de dados e inicialização do programa

O programa desenvolvido realiza simulação de escoamento de fluidos de Stokes a partir de imagens binárias. Como dados iniciais, o usuário deve entrar com uma imagem matricial binária (linha 02 do código) representando o meio periódico do material (como na Figura 4) e com as dimensões (linhas 12 e 13 do código) dessa imagem (do modelo), ou seja:

```

01 %% Leitura da imagem
02 z = imread('imagem.jpg');
03 %% Transformar em imagem binária 0-1
04 z = im2bw(z,0.5);
05 %% Aumento da resolução da imagem em n vezes
06 n = 1;
07 z = repelem(z,n,n);
08 %% Desenha a imagem na tela
09 figure(1)
10 imshow(z)
11 %% Determina a permeabilidade:
12 lx = 2; % dimensão do modelo na horizontal
13 ly = 2; % dimensão do modelo na vertical
14 % com o programa proposto pelos autores:
15 CH = calcula_permeabilidade(lx,ly,z);
16 % com o programa proposto por Andreassen e Andreassen [1]:
17 z = z+1;
18 zeta1 = 0;
19 zeta2 = 10000000; % coeficiente de penalização de Brinkman
20 CH = homogenize_fluid(lx,ly,[zeta1,zeta2],90,z);

```

Observe que na etapa de inicialização, a imagem matricial é transformada em um vetor de indicação dos elementos que informa a natureza dos elementos da malha (se sólido ou fluido, pela variável “z” no código).

É conveniente que o trecho de código apresentado, entre as linhas 01 e 20, seja armazenado em um arquivo próprio de entrada de dados, desacoplando assim o programa principal (que calcula a permeabilidade) das entradas de dados (modelo: imagem e dimensões), pois dessa forma é possível salvar os dados de diversos problemas, assim como utilizar o mesmo modelo com diferentes programas de análise (ver linhas 15 e 20 do código, como exemplo). Portanto, o programa principal é definido em uma função própria, que será apresentada a partir da linha 21 do código.

Para determinar a permeabilidade, chama-se a função “calcula_permeabilidade”, a qual recebe como parâmetros as dimensões do modelo e a informação de cada elemento se é sólido ou fluido, ou seja:

```
21 function X = calcula_permeabilidade(lx,ly,z)
22 %% Inicialização
23 [nely, nelx] = size(z);
24 dx = lx/nelx;
25 dy = ly/nely;
26 nel = nelx*nely;
```

Também é necessária a criação de uma matriz de conectividade dos elementos que indica as incidências nodais de cada elemento finito. A matriz de conectividade é montada de forma que a imagem esteja inserida em um meio periódico, ou seja, os nós das bordas opostas da imagem possuem a mesma numeração. Ilustra-se na Figura 5 como é realizada a numeração dos nós da imagem com o uso de condições de contorno periódicas.

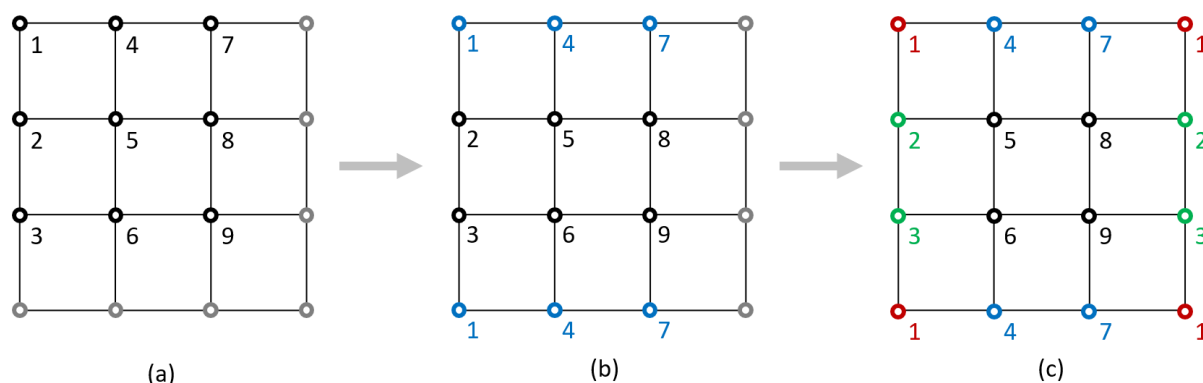


Figura 5. Numeração dos nós da malha para consideração de condições de contorno periódicas.

Para implementar essa estratégia de numeração dos nós da malha (ver linhas 28 a 32) e geração das informações de conectividade (incidência nodais) dos elementos (ver linhas 33 a 37), que considera condições de contorno periódicas, utiliza-se o seguinte trecho de código:

```
27 %% Condições de contorno periódicas (P)
28 nnP = nel; %Número de nós únicos
29 ngdlsP = 3*nnP; %Número de graus de liberdade únicos mIdNosP =
30 reshape(1:nnP, nelx, nely); %Matriz dos nós não repetidos
31 mIdNosP(end+1, :) = mIdNosP(1, :); %Inclusão dos nós da borda inferior
32 mIdNosP(:, end+1) = mIdNosP(:, 1); %Inclusão dos nós da borda direita
33 mConectP = zeros(nel, 4); %Matriz de conectividade nodal dos elementos
34 mConectP = [reshape(mIdNosP(2:end, 1:end-1), nel, 1) ...
35             reshape(mIdNosP(2:end, 2:end), nel, 1) ...
36             reshape(mIdNosP(1:end-1, 2:end), nel, 1) ...
37             reshape(mIdNosP(1:end-1, 1:end-1), nel, 1)];
```

5.2 Cálculo das matrizes de coeficientes e do lado direito dos elementos

Como o programa se propõe a fazer simulações a partir de imagem, e tem seus elementos baseados em voxels, todos os elementos são iguais, portanto, as matrizes de viscosidade, gradiente, e de estabilização de pressão podem ser calculadas apenas uma vez, reduzindo consideravelmente o tempo de simulação. Essas matrizes são incorporadas no programa principal por meio do seguinte comando:

```
38 %% Cálculo das matrizes únicas dos elementos
39 [ke,ge,pe,fe] = matElementoFinito(dx, dy);
```

em que o cálculo das matrizes pode ser feito pelo seguinte algoritmo:

```
100 %% Matrizes de Viscosidade, Gradiente, Pressão e de Carga
101 function [k,g,pe,f] = matElementoFinito(dx, dy)
102 coordsElem = [0 0; dx 0; dx dy; 0 dy];
103 rr = [-1/sqrt(3) 1/sqrt(3)];
104 ww = [1 1];
105 ss = rr;
106 k = zeros(8);
107 C = [2 0 0; 0 2 0; 0 0 1];
108 g = zeros(8,4);
109 pe = zeros(4);
110 h2 = ((dx)^2)+((dy)^2);
111 stab = h2/12;
112 f = zeros(4,1);
113 for i = 1:2
114     r = rr(1,i);
115     for j = 1:2
116         s = ss(1,j);
117         N=(1/4)*[(1-s)*(1-r) 0 (1-s)*(1+r) 0 (1+s)*(1+r) 0 (1-r)*(1+s) 0;
118                 0 (1-s)*(1-r) 0 (1-s)*(1+r) 0 (1+s)*(1+r) 0 (1-r)*(1+s)];
119         dN1dr = -1/4*(1-s); dN1ds = -1/4*(1-r);
120         dN2dr = +1/4*(1-s); dN2ds = -1/4*(1+r);
121         dN3dr = +1/4*(1+s); dN3ds = +1/4*(1+r);
122         dN4dr = -1/4*(1+s); dN4ds = +1/4*(1-r);
123         DN = [dN1dr dN2dr dN3dr dN4dr;
124              dN1ds dN2ds dN3ds dN4ds];
125         J = DN*coordsElem;
126         invJ = 1.0/det(J)*[J(2,2) -J(1,2); -J(2,1) J(1,1)];
127         dN = invJ*DN;
128         B = [dN(1,1) 0 dN(1,2) 0 dN(1,3) 0 dN(1,4) 0;
129              0 dN(2,1) 0 dN(2,2) 0 dN(2,3) 0 dN(2,4);
130              dN(2,1) dN(1,1) dN(2,2) dN(1,2) dN(2,3) dN(1,3) dN(2,4) dN(1,4)];
131         k = k + (B')*C*B*det(J)*ww(1,i)*ww(1,j);
132         g = g + (B')*[1;1;0]*N(1:4:end)*(det(J))*ww(1,i)*ww(1,j);
133         Bp = inv(J)*[DN(1,:);DN(2,:)];
134         pe = pe + stab*(Bp')*Bp*(det(J))*ww(1,i)*ww(1,j);
135         f = f + N(1,1:2:end)'\*det(J)*ww(1,i)*ww(1,j);
136     end
137 end
138 end
```

O código anterior, apesar de estar apresentado de uma forma didática, não ficam explicitados os elementos necessários para o cálculo de cada matriz de forma independente. Entretanto, como as funções de forma utilizadas neste trabalho para interpolar os campos de velocidade e pressão são as mesmas, as matrizes de viscosidade, gradiente, estabilização de pressão e os vetores do lado direito podem ser calculados dentro de uma mesma função, tornando o algoritmo mais compacto e elegante.

5.3 Montagem do Sistema de Equações

A montagem da matriz dos coeficientes é realizada de forma que a contribuição dos graus de liberdade de cada elemento seja alocada nos graus de liberdade correspondentes da malha da imagem. O mapeamento dos graus de liberdade dos elementos é feito por meio de um vetor que indica os índices da matriz de coeficientes onde serão alocados os termos das matrizes \mathbf{K} , \mathbf{G} , \mathbf{G}^T e \mathbf{P} para cada elemento. Uma forma intuitiva e didática de realizar essa montagem é por meio do seguinte algoritmo:

```

40 %% Montagem do sistema de equações
41 velF = find(~z); %Vetor com elementos somente da região do fluido
42 for elem = 1:size(velF,1)
43     e= velF(elem);
44     vgd1F = [mConectP(e,1)*2-1; mConectP(e,1)*2; %gdl 1,2(v) nó n
45             mConectP(e,2)*2-1; mConectP(e,2)*2;
46             mConectP(e,3)*2-1; mConectP(e,3)*2;
47             mConectP(e,4)*2-1; mConectP(e,4)*2;
48             2*nel+mConectP(e,1); %gdl 3(p) nó n
49             2*nel+mConectP(e,2);
50             2*nel+mConectP(e,3);
51             2*nel+mConectP(e,4)];
52     for i=1:8
53         for j=1:8 %% Alocação Matriz de Viscosidade
54             A(vgd1F(i), vgd1F(j)) = A(vgd1F(i), vgd1F(j)) + ke(i,j);
55         end
56         for j=1:4 %% Alocação Matriz Gradiente
57             A(vgd1F(i), vgd1F(8+j)) = A(vgd1F(i), vgd1F(8+j)) + ge(i,j);
58         end
59     end
60     for i=1:4 %% Vetor RHS
61         F(vgd1F(i*2-1),1) = F(vgd1F(i*2-1),1) + f(i);
62         F(vgd1F(i*2),2) = F(vgd1F(i*2),2) + f(i);
63         for j=1:4 %% Alocação Matriz de Pressão
64             A(vgd1F(8+i),vgd1F(8+j)) = A(vgd1F(8+i),vgd1F(8+j)) - pe(i,j);
65         end
66     end
67 end
68 G = A(1:nel*2,nel*2+1:nel *3);
69 A(nel*2+1:nel*3,1:nel*2) = G';

```

Entretanto, apesar ser uma forma mais intuitiva, o uso de diversos laços faz com que o tempo de processamento aumente significativamente. Uma abordagem mais elegante e otimizada de realizar a mesma operação é pelo uso de *triplets*. Dessa forma, o algoritmo anterior pode ser substituído por:

```

40 %% Montagem do sistema de equações
41 velF = find(~z); %Vetor com elementos somente da região do fluido
42 nelF = size(velF,1);
43 mgdlF = [mConectP(velF,1) '*2-1; mConectP(velF,1) '*2; %gdl 1,2(v) nó n
44          mConectP(velF,2) '*2-1; mConectP(velF,2) '*2;
45          mConectP(velF,3) '*2-1; mConectP(velF,3) '*2;
46          mConectP(velF,4) '*2-1; mConectP(velF,4) '*2;
47          2*nel+mConectP(velF,1)'; %gdl 3(p) nó n
48          2*nel+mConectP(velF,2)';
49          2*nel+mConectP(velF,3)';
50          2*nel+mConectP(velF,4)'];
51 iK = repelem(reshape(mgdlF(1:8,1:end), nelF*8,1), 8,1);
52 jK = reshape(repmat(mgdlF(1:8,1:end), 8,1), nelF*64,1);
53 iG = repelem(reshape(mgdlF(1:8,1:end), nelF*8,1), 4,1);
54 jG = reshape(repmat(mgdlF(9:12,1:end), 8,1), nelF*32,1);
55 iP = repelem(reshape(mgdlF(9:12,1:end), nelF*4,1), 4,1);
56 jP = reshape(repmat(mgdlF(9:12,1:end), 4,1), nelF*16,1);

```

```

57  iA = [iK;iG;jG;iP];
58  jA = [jK;jG;iG;jP];
59  clear iK iG iP jK jG jP
60  coef = [repmat(ke,nelF,1); repmat(ge,nelF,1);
61         repmat(ge,nelF,1);-repmat(pe,nelF,1)];
62  A = sparse(iA,jA,coef);
63  clear iA jA coef
64  iF = [reshape(mgdlF(1:2:8,1:end),nelF*4,1);
65        reshape(mgdlF(2:2:8,1:end),nelF*4,1)];
66  jF = [ones(nelF*4,1);2*ones(nelF*4,1)];
67  sF = repmat(fe,nelF*2,1);
68  F = sparse(iF,jF,sF,ngdlsP,2);
69  clear iF jF sF mgdlF

```

5.4 Condições de contorno na interface dos grãos e resolução do sistema de equações

A implementação das condições de velocidade zero nos nós relativos à interface fluido-sólido é feita pela redução da matriz de coeficientes para solucionar somente as variáveis não-nulas. Esse processo é realizado nas linhas 71 a 79, em que a solução do sistema é realizado na linha 79.

```

70  %% Resolução do sistema de equações
71  resolveF_u = 1:nel*2;
72  nosnulos = [unique(mConectP(find(z),1:4))];
73  gdlnulos = [nosnulos*2-1 nosnulos*2];
74  resolveF_u(gdlnulos) = []; %Elimina linhas e colunas relativas aos
                             gdl nulos
75  resolveF_p = (nel*2+unique(mConectP(find(~z),1:4)))';
76  resolveF = [resolveF_u resolveF_p];
77  clear resolveF_u resolveF_p nosnulos gdlnulos mConectP
78  X = zeros(ngdlsP,2);
79  X(resolveF,:) = A(resolveF,resolveF)\F(resolveF,:);

```

5.5 Cálculo da Permeabilidade

Após a montagem do vetor de variáveis nodais, a permeabilidade é calculada pela média das velocidades e do gradiente de pressão. Este processo é realizado nas linhas 80 a 88.

```

80  %% Homogeneização
81  CH = zeros(2);
82  vIdNosP = reshape(mIdNosP,size(mIdNosP,1)*size(mIdNosP,2),1);
83  CH(1,1) = sum(X(vIdNosP(1:end,1)*2-1,1));
84  CH(1,2) = sum(X(vIdNosP(1:end,1)*2,1));
85  CH(2,1) = sum(X(vIdNosP(1:end,1)*2-1,2));
86  CH(2,2) = sum(X(vIdNosP(1:end,1)*2,2));
87  CH = CH/nel;
88  disp(CH)

```

6 Resultados

6.1 Teste de validação

Para validar o programa apresentado neste trabalho, foi realizada uma simulação de escoamento com baixo valor de Reynolds no meio poroso representado na imagem da Figura 6. A permeabilidade do meio adquirida pelo código proposto neste trabalho foi comparada com o resultado obtido pelo programa de Andreassen e Andreassen [1], e com a solução analítica proposta por Drummond e Tahir

[14]. Segundo esses últimos autores, a permeabilidade do meio apresentado na Figura 6 pode ser determinada pela equação $k = r^2(-\ln(c) - 1,476 + 2c - 1,774c^2) / (8c)$, em que a permeabilidade (k) é determinada em função do raio dos grãos circulares (r) e da porcentagem volumétrica de sólido (c).



Figura 6. Meio poroso utilizado para validação do programa.

De acordo com Drummond e Tahir [14], a equação analítica proposta é válida para pequenos valores de c . Portanto, para validação do programa, o REV do meio foi modelado com dimensões unitárias e grãos de raio igual a 0,1. Para esses valores adotados, pode-se registrar o c correspondente e obter a permeabilidade $k = 0,0281$ via solução analítica. A Tabela 1 e a Figura 7 apresentam os resultados obtidos por meio das modelagens numéricas.

Tabela 1. Comparação entre os resultados obtidos do problema de validação por diferentes autores.

Malha	Este Trabalho	Andreassen e Andreassen [1]
10x10	0,0253	0,0267
20x20	0,0288	0,0295
30x30	0,0255	0,0260
40x40	0,0272	0,0275
50x50	0,0271	0,0273
60x60	0,0278	0,0280
70x70	0,0275	0,0276
80x80	0,0271	0,0273
90x90	0,0277	0,0278
100x100	0,0276	0,0277
110x110	0,0276	0,0277
120x120	0,0280	0,0281
130x130	0,0275	0,0276
140x140	0,0279	0,0279
150x150	0,0276	0,0277

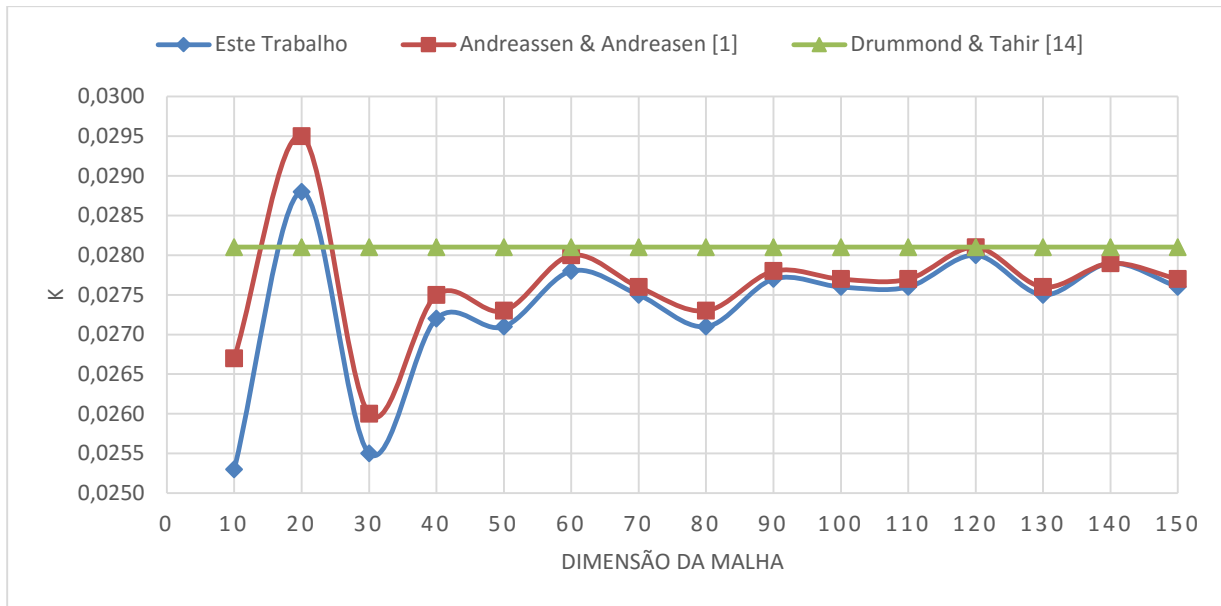


Figura 7. Convergência do resultado do modelo de validação.

As flutuações observadas nos resultados dos modelos computacionais podem ser explicadas pelo fato de que a entrada de dados dos programas são imagens, e para que o tamanho da malha seja modificado, a resolução da imagem também é modificada. Isso faz com que o contorno dos grãos varie de uma imagem para a outra conforme se altera a resolução, fazendo com que o resultado oscile, pois a geometria do problema não é respeitada. Entretanto, apesar de existir tal oscilação, a permeabilidade obtida pelo código apresentado por este trabalho apresentou uma diferença de apenas 1,78% do resultado analítico (malha 150x150).

6.2 Permeabilidade de amostra de arenito

Após validação, o programa foi utilizado para medir a permeabilidade de uma amostra de arenito representada por uma imagem 2-D. A imagem bidimensional do arenito a ser analisada é exatamente a mesma imagem apresentada na Figura 3, que tem dimensões de 1 mm x 1 mm. A Tabela 2 apresenta os valores obtidos e o tempo de processamento para diferentes tamanhos de malha. Os tempos de processamento obtidos são referentes à simulação do problema no software Octave versão 5.1.0 em um Dell Inspiron 14 7460 com processador i7-7500, 16Gb de memória RAM e sistema operacional Windows 10. A Figura 8 mostra a convergência da permeabilidade do arenito conforme refina-se a malha de elementos finitos.

Tabela 2. Convergência do resultado de permeabilidade do arenito.

Malha	Kxx (μm^2)	Kyy (μm^2)	Tempo de Processamento
600 x 600	0,2772	0,2954	0,5 min
800 x 800	0,2019	0,2344	1,5 min
1000 x 1000	0,1638	0,2051	3,5 min
1200 x 1200	0,1420	0,1889	7,5 min
1400 x 1400	0,1283	0,1790	15,0 min
1600 x 1600	0,1192	0,1724	27,0 min

A permeabilidade encontrada para a imagem 2-D da amostra de arenito analisada foi $K_{xx} = 0,1192 \mu\text{m}^2$ e $K_{yy} = 0,1724 \mu\text{m}^2$. Isso significa que a amostra é anisotrópica em relação à permeabilidade.



Figura 8. Convergência do resultado da permeabilidade 2-D de uma amostra de arenito.

7 Conclusões

Este trabalho apresentou um programa computacional educativo capaz de medir a permeabilidade de materiais porosos por meio da simulação de escoamentos governados pela equação de Stokes. A fundamentação teórica necessária para modelagem do problema foi desenvolvida, e as peculiaridades do MEF aplicado à fluidodinâmica computacional foram discutidas. Foram explicitadas diferentes formas de realizar a montagem do sistema de equações a ser resolvido, sendo uma das formulações capaz de reduzir significativamente o tempo de processamento do programa.

Por estar inserido no âmbito da petrofísica digital, o programa aqui proposto utiliza imagens binárias como entrada de dados. Essa característica permite que a permeabilidade de materiais porosos seja determinada pelo uso de micro-CT, que é capaz de representar a estrutura interna do material por meio de pixels que constituem o fluido e pixels que constituem o sólido. Neste trabalho, a modelagem do meio poroso foi realizada de forma que apenas os pixels do domínio fluido fossem considerados. Essa metodologia é extremamente eficiente para materiais que possuem baixas porosidades, pois permite que a permeabilidade seja obtida utilizando menor esforço computacional se comparada com metodologias que modelam o domínio sólido e fluido.

O programa desenvolvido pelos autores foi validado por meio de um problema simplificado que possui solução analítica conhecida. Doravante, a permeabilidade de uma amostra de arenito foi calculada por meio de uma análise 2-D. Embora a validação do código confira credibilidade ao programa computacional proposto, o valor da permeabilidade do arenito calculada em duas dimensões pode não ser representativo para uma amostra de arenito em três dimensões devido às características geométricas tridimensionais de seus poros. Entretanto, o desenvolvimento do programa de análise 2-D apresentado confere ao leitor o conhecimento necessário para que a extensão do programa para 3-D seja realizada de forma simples e intuitiva.

Agradecimentos

Essa pesquisa foi parcialmente financiada pela Shell e pelo CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico), projeto número 404593/2016-0.

Referências

- [1] E. Andreassen and C. S. Andreasen. How to determine composite material properties using numerical homogenization. *Computational Materials Science*, vol. 83, pp. 488-495, 2014. <https://doi.org/10.1016/j.commatsci.2013.09.006>.
- [2] J. E. Aarnes, T. Gimse and K.-A. Lie. An Introduction to the Numerics of Flow in Porous Media using Matlab. *Geometric Modelling, Numerical Simulation, and Optimization*. Springer, Berlin, Heidelberg, 2007.
- [3] X. Y. Sheng and X. X. Zhi. A new numerical method for the analysis of the permeability anisotropy ratio. *Chinese Physics*, vol. 11, pp. 1009-1963, 2002.
- [4] J. N. Reddy. *An Introduction to The Finite Element Method*. McGraw-Hill, New York, 2005.
- [5] O. A. Ladyzhenskaya. *The mathematical theory of viscous incompressible flow*. New York: Gordon and Breach, 1969.
- [6] I. Babuška. Error-Bounds for Finite Element Method. *Numer. Math*, vol. 16, pp. 322-333, 1971.
- [7] F. Brezi. On the Existence, Uniqueness and Approximation of Saddle-Point Problems Arising from Lagrangian Multipliers. *Revue française d'automatique informatique recherche operationnelle. Analyse Numérique*, vol. 8, pp. 129-51, 1974.
- [8] K.-J. Bathe. *Finite element procedures*, Prentice Hall, 1996.
- [9] H. C. Elman, D. J. Silvester and A. J. Wathen. *Finite elements and fast iterative solvers: with applications in incompressible fluid dynamics*. Oxford University Press, 2006
- [10] M. R. Karim, K. Krabbenhoft and A. V. Lyamin. Permeability determination of porous media using large-scale finite elements and iterative solver. *International Journal for Numerical and Analytical Methods in Geomechanics*, vol. 38, pp. 991–1012, 2014.
- [11] M. Braack and F. Schieweck. Equal-order finite elements with local projection stabilization for the Darcy–Brinkman equations. *Computer Methods in Applied Mechanics and Engineering*, vol. 200, pp. 1126–1136, 2011.
- [12] R. Codina and J. Blasco. A finite element formulation for the Stokes problem allowing equal velocity-pressure interpolation. *Computer Methods in Applied Mechanics and Engineering*, vol. 143, pp. 373-391, 1997.
- [13] R. Becker and P. Hansbo. A simple pressure stabilization method for the Stokes equation. *International Journal for Numerical Methods in Biomedical Engineering*, vol. 24, pp. 1421-1430, 2008.
- [14] J. E. Drummond and M. I. Tahir. Laminar viscous flow through regular arrays of parallel solid cylinders. *International Journal of Multiphase Flow*, vol. 10, No. 5, pp. 515-540, 1984.