

A STUDY OF CONTINUUM MECHANICS AND PHASE CHANGE PROBLEMS AND IT'S NUMERICAL MODELLING USING OPENFOAM

Maicon William Niebus Odone

Ophir Valiante Duarte Neto

Elson Magalhães Toledo

Luis Paulo S. Barra

maicon.odone@gmail.com

ophir.neto@engenharia.ufjf.br

emtc@lncc.br

luispbarra@gmail.com

Universidade Federal de Juiz de Fora - UFJF

Rua José Lourenço Kelmer - São Pedro - Juiz de Fora, 36036-900, Minas Gerais, Brazil

Aline Cruz Soares

alinecruz@ime.eb.br

Instituto Militar de Engenharia - IME

Praça General Tibúrcio, 80 - Praia Vermelha - Urca - Rio de Janeiro, 22290-270, Rio de Janeiro, Brazil

Abstract. Several applications in continuum mechanics involve processes such as solidification and melting of materials, natural convection, stresses and deformations in solids, among other related phenomena. Given the complexity of the phenomena related to these processes, the numerical and computational modeling are powerful and of low costs tools suitable to understand and simulate problems which may involve one or more of these complex phenomena. In this paper, a set of mathematical models representing some of these processes is introduced, discretized and solved by the Finite Volume Method, using open-source package OpenFOAM, a based C++ toolbox for the development of customized numerical solvers, and pre-/post-processing utilities for the solution of continuum mechanics problems from computational fluid dynamics, including also static solid elastic problems. Application examples are shown including the gallium melting problem, a process commonly needed in computer chips manufacturing; the lid-driven cavity problem, widely used for testing, verification and validation of numerical procedures and a simple linear elasticity problem. We show some of the basic features of OpenFOAM as well as explore some of its potential pointing on some possible disadvantages of this code.

Keywords: Continuum mechanics; Computational fluid dynamics; Finite volume method; OpenFOAM.

1 Introdução

Métodos computacionais têm se mostrado bastante poderosos como ferramenta para resolver os mais variados tipos de problema. Há décadas, cientistas e engenheiros recorrem a procedimentos numéricos como forma de entender o mundo e ajudar em seus projetos. Assim, vários *softwares* tem sido construídos, com o objetivo de permitir o uso do cálculo numérico como ferramenta, bem como também fornecer plataformas versáteis e confiáveis para estes profissionais. Códigos como o AnSYS® e o Abaqus® consagraram-se por este motivo e recentemente temos visto os **softwares livres** (ou *open source software*) ganharem confiança e reconhecimento.

A essência de um *software livre* é a disponibilização do livre acesso a seu código fonte por seus usuários permitindo, com isto, que seus algoritmos possam ser analisados, evitando seu funcionamento como uma caixa preta, como seus concorrentes convencionais. Ademais, por ser plenamente modificável, é possível realizar ajustes de desempenho, corrigir defeitos e até mesmo implementar novas funcionalidades, mesmo as mais complexas, sem depender da ajuda de terceiros. Além disto, aplicações livres, em particular as licenciadas pela GPL (General Public License), costumam ser totalmente gratuitas (o que constitui o conceito de *Free Open Source Software* (FOSS)). Isto se põe em forte contraste com as programas mais populares, que costumam ser fechados em sua essência, resultando que ninguém além do desenvolvedor saiba com clareza como funcionam; não permitindo modificações e fazendo com que seus usuários dependam do apoio do desenvolvedor para que sejam corrigidos *bugs* e cobrando periodicamente por seu uso. Portanto, tanto para fins acadêmicos quanto para fins comerciais, a utilização de ferramentas numéricas de código aberto apresenta um grande potencial, ainda pouco explorado.

Dos pacotes de cálculo numérico com código aberto disponíveis atualmente, um dos mais famosos é o OpenFOAM®, que dedica-se principalmente à resolução de problemas de Dinâmica dos Fluidos Computacional (*Computational Fluid Dynamics* ou CFD). Neste trabalho, o OpenFOAM® é utilizado na resolução de problemas relacionados à dinâmica dos fluidos e dos sólidos, visando ilustrar alguns de seus recursos e potencialidades nativas, além de também indicar algumas de suas fragilidades. A modelagem matemática dos problemas ilustrados são apresentadas e os resultados obtidos são comparados com a literatura.

2 O software OpenFOAM

OpenFOAM (Open Field Operation And Manipulation) é um *toolkit* escrito em C++ voltado para a construção de ferramentas para a resolução de problemas de mecânica do contínuo (Maric et al. [1]) e, em especial, para problemas de dinâmica dos fluidos. Possui licença GPLv3 que é aquela que permite que seu usuário o utilize e modifique sem incorrer em qualquer infração de direito autoral.

Sua robusta base de código contém módulos dedicados a atividades de pré-processamento, processamento e pós-processamento aplicáveis a diversos tipos de problemas computacionais. Também possui interfaces com outros programas, como o gerador de malhas **gmsh** e com o **ParaView** para pós-processamento dos resultados.

Historicamente, o OpenFOAM foi criado por Henry Weller no *Imperial College London* em 2004, juntamente com Chris Greenshields e Mattijs Janssens que fundaram a empresa OpenCFD Ltd., e tornaram este código aberto. Nos anos seguintes, a empresa dedicou-se a distribuir e realizar a manutenção do software. Em 2011, a OpenCFD foi adquirida pela Silicon Graphics Group (SGI), e esta foi adquirida pelo grupo ESI em 2012 (Chen et al. [2]). Na época da aquisição da OpenCFD pela SGI, foi criada a OpenFOAM Foundation, com o propósito de manter o código do software aberto e distribuí-lo ao público em geral. As versões produzidas pela ESI podem ser encontradas no site *openfoam.com*, enquanto que as versões lançadas pela OpenFOAM Foundation podem ser encontradas em *openfoam.org*. [3]

Para a resolução de um problema através do OpenFOAM®, é preciso criar um diretório (chamado de diretório de caso, ou *case directory*), com os seguintes sub-diretórios: *0*, *constant* e *system*. Em *0*, encontram-se as descrições das condições iniciais e das condições de contorno do problema. Em *constant*, encontram-se informações da discretização do problema, como sua malha (quando esta é fixa, no

diretório *polymesh*) e propriedades de materiais (viscosidade, condutividade térmica, etc.). Em *system*, encontram-se configurações inerentes aos métodos numéricos, como o passo de tempo da simulação, o número de iterações, os métodos de discretização e os métodos de resoluções dos sistemas lineares, e arquivos com informações de pré e pós-processamento, como a geração de malha ou geração de gráficos. Durante a etapa de pré-processamento, todo este diretório de caso deve ser montado cuidadosamente, de modo a descrever corretamente a situação a ser interpretada pelo *solver* escolhido. Durante a etapa de processamento, o solver preencherá este diretório com outros diretórios, numerados de acordo com o passo de tempo para problemas transientes

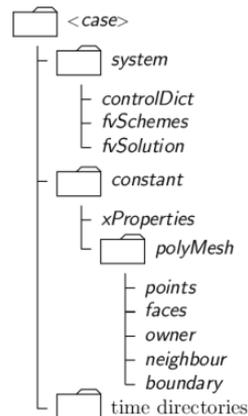


Figura 1. Exemplo de estrutura de um diretório de caso do OpenFOAM® [4].

No OpenFOAM®, *solvers* nada mais são que aplicações construídas com base nas suas bibliotecas cuja finalidade é receber como entrada os dados do diretório de caso e resolver um problema de acordo com sua descrição. Podem ser nativos, sendo distribuídos na instalação do pacote, ou serem criados e compartilhados entre seus usuários. Por ser uma plataforma de código aberto incentiva-se que os usuários criem e compartilhem seus próprios *solvers*. Toda instalação do OpenFOAM® conta com um grande acervo de *solvers* nativos, capazes de resolver diversos casos de problemas da mecânica do contínuo: escoamentos incompressíveis, escoamentos turbulentos supersônicos, tensões e deformações em materiais elásticos, combustão em motores alternativos, escoamentos provocados por rompimento de barragens (*dam breaking*), e até mesmo problemas financeiros, como a Equação de Black-Scholes. Esta multiplicidade de aplicações decorre da flexibilidade deste código em permitir ao usuário a construção de *solvers* para a resolução de problemas diversos através da "montagem" adequada de seus módulos.

Geralmente, por serem específicos para os problemas que pretendem resolver, uma das etapas importantes de pré-processamento no OpenFOAM® é justamente a escolha do *solver* mais adequado para o trabalho, conforme as considerações feitas em sua modelagem. Sua escolha também influencia diretamente na estrutura do diretório caso, visto que o *solver* espera ler este de uma maneira bem específica.

No geral, os *solvers* são construídos de maneira a permitir que haja flexibilidade em seu uso. Por exemplo, a escolha dos métodos de discretização a serem utilizados num problema é feita inteiramente pelo usuário em *system/fvSchemes*, um arquivo com formato específico chamado **dicionário** (dictionary).

Todas informações a serem lidas pelas aplicações do OpenFOAM® encontram-se nestes arquivos de texto chamados **dicionários**. Possuem sintaxe bem-estruturada, que depende da informação que disponibilizam. Por exemplo, o arquivo que descreve as condições iniciais e de contorno do campo de velocidade tem uma estrutura específica, enquanto que os arquivos que descrevem a malha devem ter outra sintaxe. Internamente, a plataforma possui bibliotecas com a função de ler e escrever dicionários, realizando todas as operações de *input* e *output* necessárias de forma automatizada.

Da mesma maneira, a escolha dos métodos de resolução dos sistemas lineares do problema é também feita pelo usuário no dicionário em *system/fvSolutions*. Há ainda outras bibliotecas frequentemente presentes nos *solvers* da plataforma, como a *fvOptions*, a *thermophysicalProperties*, e bibliotecas dedicadas à modelagem de turbulência, cujo objetivo é justamente ampliar a variedade de problemas que

um *solver* pode resolver. Por exemplo, um *solver* que trate problemas com turbulência normalmente inclui tratamentos do tipo *Reynolds-averaged simulation*, *large-eddy simulation*, $k - \epsilon$, $k - \omega$.

O OpenFOAM® possui duas aplicações nativas dedicadas à geração de malhas. São elas o **blockMesh** e o **snappyHexMesh**. O primeiro possibilita a criação de malhas estruturadas com elementos hexaédricos enquanto que o *snappyHexMesh* é adequado para a criação de malha em torno de corpos, como na geração de malhas para o estudo de escoamento de ar em aerofólios ou no estudo do desempenho aerodinâmico de veículos. Além disto, possui, em sua instalação, programas para importar malhas geradas em outros *softwares* para o diretório de caso, como o *gmsh* e o *Fluent* (*gmshToFoam* e *fluentMeshToFoam*, respectivamente). Em problemas de malha fixa, todas as informações pertinentes à malha encontram-se no diretório *constant/polyMesh*, organizada em uma série de dicionários, como mostrado na Fig. 1.

No caso da análise dos problemas de mudança de fase com convecção natural descrito neste artigo, utilizamos o *solver buoyantPimpleFoam* que trata-se de um *solver* para resolução de um problema de escoamento transiente, incompressível com transferência de calor que permite incluir, se desejável, modelos de turbulências e ser estendido através do *fvOptions*. O *fvOptions* é uma forma do usuário adicionar novos fenômenos ao modelo de seu problema, através da inclusão de parcelas no termo fonte das equações diferenciais, além de impor a estas restrições e correções. Por exemplo, ao configurar um termo fonte do tipo **solidificationMeltingSource** no diretório de um caso a ser resolvido com o *buoyantPimpleFoam* (ESI [5]), este passa a incluir mudança de fase descrita através do modelo da entalpia-porosidade. Isto torna o *solver*, que em sua essência resolve problemas de convecção natural capaz de resolver problemas de mudança de fase. Esse termo fonte é incluído através de um arquivo chamado *fvOptions* (ESI [6]), presente ou na pasta *constant* ou na pasta *system*.

3 Problemas Resolvidos com o OpenFOAM

3.1 Problema de advecção-difusão

Sendo ϕ uma quantidade escalar qualquer, tal como temperatura ou alguma concentração a equação, para uma região bidimensional, que descreve o fenômeno de transporte com difusão de ϕ nesta região pode ser descrita como:

$$\frac{\partial}{\partial t}(\rho\phi) + \nabla \cdot (\rho\vec{V}\phi) - \nabla \cdot (\Gamma_\phi \nabla \phi) - S_\phi = 0 \quad (1)$$

com condições inicial e de contorno descritas por Versteeg [7] e Voller [8]:

$$\begin{aligned} \phi(x, y, 0) &= \phi(x, y) \quad \text{em } \Omega \quad \text{para } t = 0 \\ \phi(x, y, t) &= \phi_D(x, y, t) \quad \text{em } \partial\Omega_{dir} \quad \text{para } t > 0 \\ (\Gamma_\phi \nabla \phi) \cdot \vec{n} &= \phi_N(x, y, t) \quad \text{em } \partial\Omega_{neu} \quad \text{para } t > 0 \end{aligned} \quad (2)$$

onde ρ é a densidade, Γ_ϕ o coeficiente de difusão, $\vec{V} = (u, v)$ o vetor velocidade, \vec{n} um vetor normal ao contorno $\partial\Omega$ de Ω e S_ϕ são eventuais termos fonte. As notações $\partial\Omega_{dir}$ e $\partial\Omega_{neu}$ indicam, respectivamente, regiões do contorno com condições de Dirichlet e Neumann prescritas.

Tratamos aqui de problema de advecção-difusão baseado em um dos exemplos presentes em Brooks and Hughes [9], cuja geometria e condições de contorno encontram-se resumidos na Fig. 2. O ângulo θ assumiu os valores $22,5^\circ$, 45° e $67,5^\circ$, correspondendo a uma velocidade constante com $\|\vec{V}\| = 1$. A discretização da geometria corresponde a uma malha de elementos triangulares com 2.056 elementos.

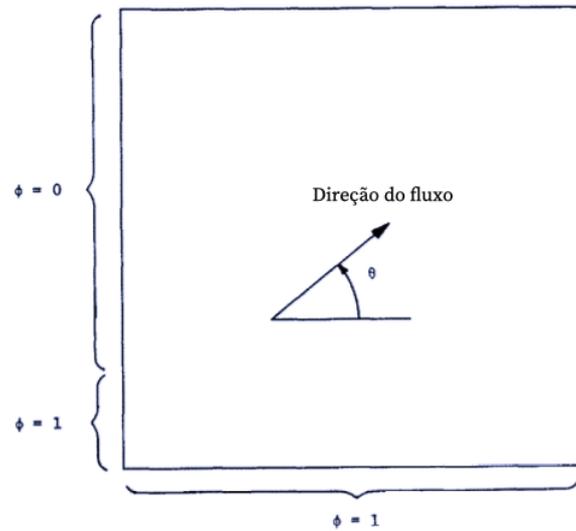


Figura 2. Geometria do problema advectivo-difusivo.

O *solver* utilizado para a solução deste problema foi o *scalarTransportFoam* que pertence a categoria *basic*, que inclui *solvers* voltados a problemas mais "básicos" de dinâmica dos fluidos computacional. Foram resolvidos problemas com três esquemas de discretização diferentes para o termo advectivo: *linear*, *upwind* e *linearUpwind grad(U)*. O número de Peclet assumiu valores variados dentro destas condições, de 100 a 10^6 .

As Figs. 3, 4 e 5 apresentam os resultados do problema para $Pe = 125$ obtidos com o OpenFOAM®.

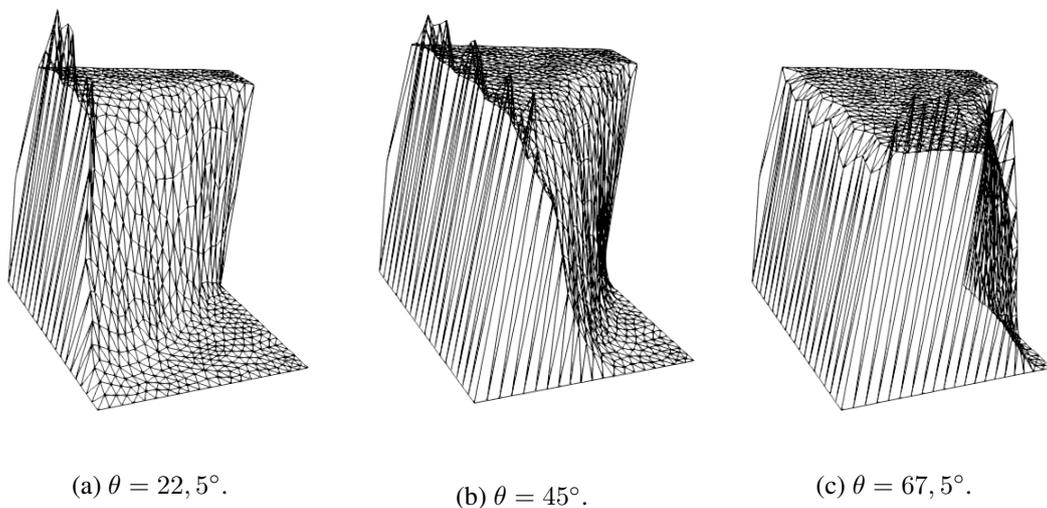


Figura 3. Esquema *linearUpwind grad(U)* para $Pe = 125$

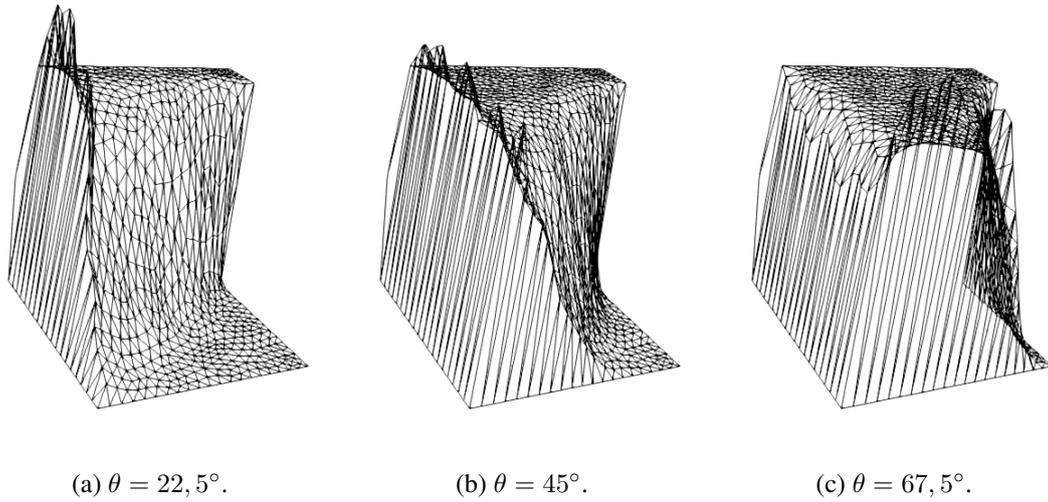


Figura 4. Esquema *upwind* para $Pe = 125$

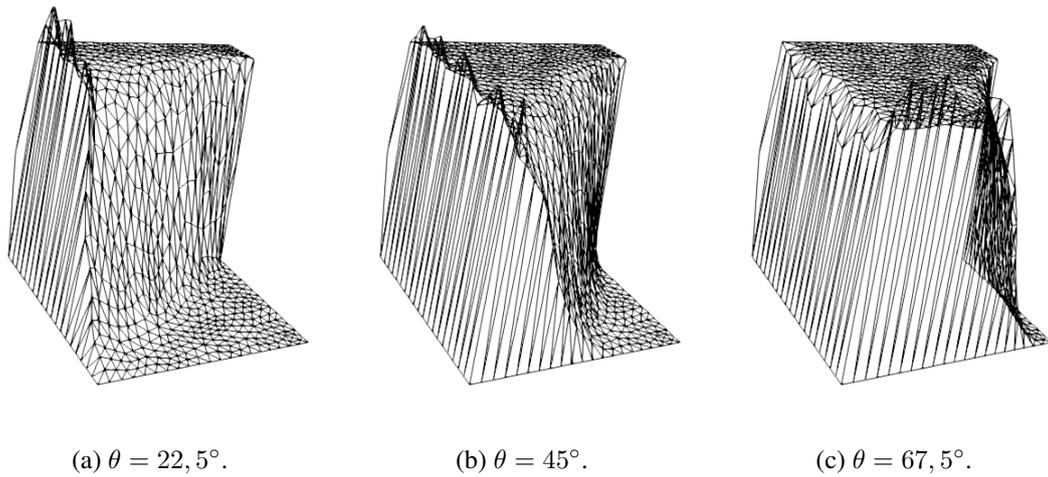


Figura 5. Esquema *linear* para $Pe = 125$

A Fig. 6 exibe a solução analítica do problema, enquanto as Figs. 7, 8 e 9 mostram os resultados encontrados em Brooks and Hughes [9], para diversas formulações baseadas no método dos elementos finitos, com número de Peclet $Pe = 10^6$.

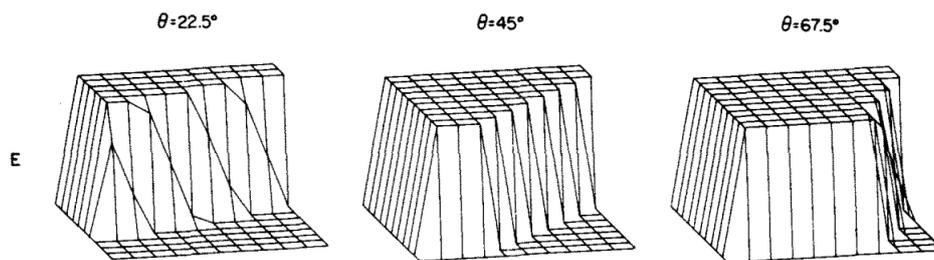


Figura 6. Solução exata (E)

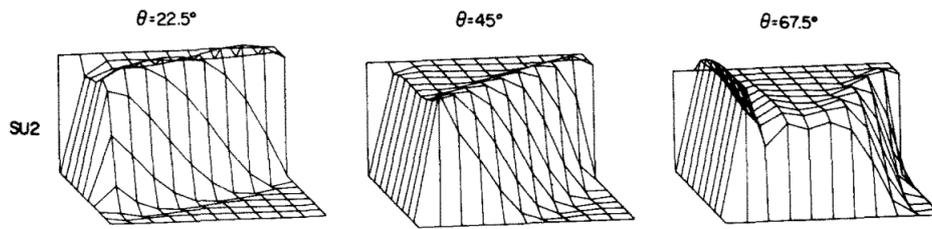


Figura 7. Streamline upwind Petrov-Galerkin, dois pontos de quadratura gaussiana (SU2)

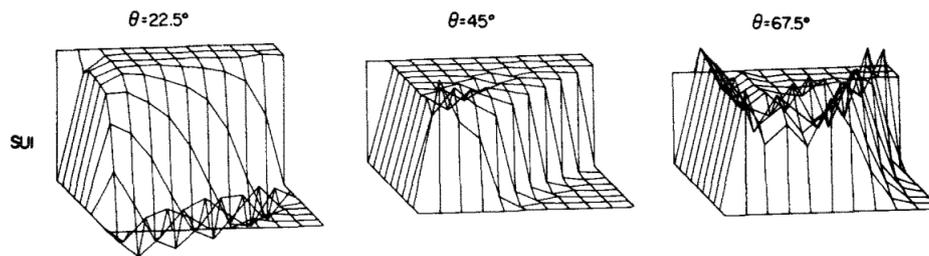


Figura 8. Streamline upwind Petrov-Galerkin, um ponto de quadratura gaussiana (SU1)

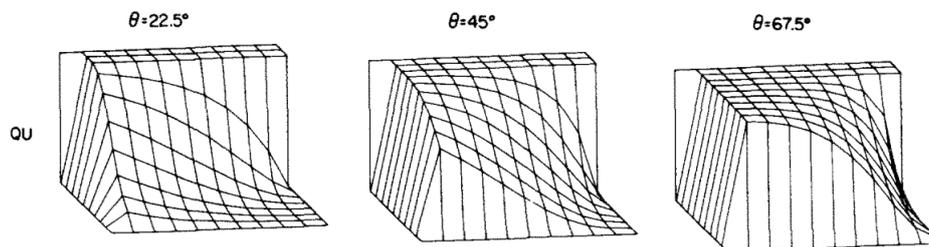


Figura 9. Quadrature upwind (QU)

No geral, observou-se uma deficiência do OpenFOAM®, em comparação com os resultados de Brooks and Hughes [9], que apresentam boa estabilidade mesmo para número de Peclet alto (da ordem de 10^6). De fato, verificou-se que a partir de valores relativamente baixos do número de Peclet, como $Pe = 125$, os resultados obtidos no OpenFOAM® tornam-se rapidamente inadequados. Por isto, são ilustrados apenas os resultados para $Pe = 125$, que apresentam resultados satisfatórios.

3.2 Problema de circulação em uma cavidade fechada (*Lid-Driven Cavity*)

Este problema é amplamente empregado para verificação e validação de algoritmos numéricos. Nele, um fluido encontra-se inicialmente em repouso no interior de uma cavidade. Ao se impor uma velocidade tangencial a uma das paredes, enquanto as demais permanecem estáticas, o fluido entra em movimento de caráter circulatório no interior da cavidade.

Para a formulação do modelo computacional, assume-se que o fluido é newtoniano incompressível em escoamento laminar viscoso, com propriedades termofísicas constantes em meio isotérmico. Ainda, assume-se o regime do escoamento como estacionário.

Sob estas hipóteses, o modelo pode ser descrito pelas equações de conservação da massa e da

conservação da quantidade de movimento linear como para um problema bidimensional (Versteeg [7]):

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0 \quad (3)$$

$$\frac{\partial(uu)}{\partial x} + \frac{\partial(vu)}{\partial y} = \frac{\mu}{\rho} \left[\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right] - \frac{1}{\rho} \frac{\partial p}{\partial x} \quad (4)$$

$$\frac{\partial(uv)}{\partial x} + \frac{\partial(vv)}{\partial y} = \frac{\mu}{\rho} \left[\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right] - \frac{1}{\rho} \frac{\partial p}{\partial y} \quad (5)$$

onde, ρ é a densidade, μ a viscosidade, p a pressão e u e v são as componentes da velocidade nas direções x e y respectivamente.

As Eqs. (3), (4) e (5) podem ser adimensionalizadas através da definição dos seguintes parâmetros adimensionais:

$$x^* = \frac{x}{L}, \quad y^* = \frac{y}{L}, \quad u^* = \frac{u}{\mu/(\rho L^2)}, \quad v^* = \frac{v}{\mu/(\rho L^2)}, \quad p^* = \frac{p}{\mu^2/(\rho L^2)} \quad (6)$$

onde L é um comprimento característico. O emprego destes parâmetros permite escrever as Eqs. (3), (4) e (5) em formas adimensionais como:

$$\frac{\partial u^*}{\partial x^*} + \frac{\partial v^*}{\partial y^*} = 0 \quad (7)$$

$$\frac{\partial(u^*u^*)}{\partial x^*} + \frac{\partial(v^*u^*)}{\partial y^*} = \frac{1}{Re} \left[\frac{\partial^2 u^*}{\partial (x^*)^2} + \frac{\partial^2 u^*}{\partial (y^*)^2} \right] - \frac{\partial p^*}{\partial x^*} \quad (8)$$

$$\frac{\partial(u^*v^*)}{\partial x^*} + \frac{\partial(v^*v^*)}{\partial y^*} = \frac{1}{Re} \left[\frac{\partial^2 v^*}{\partial (x^*)^2} + \frac{\partial^2 v^*}{\partial (y^*)^2} \right] - \frac{\partial p^*}{\partial y^*} \quad (9)$$

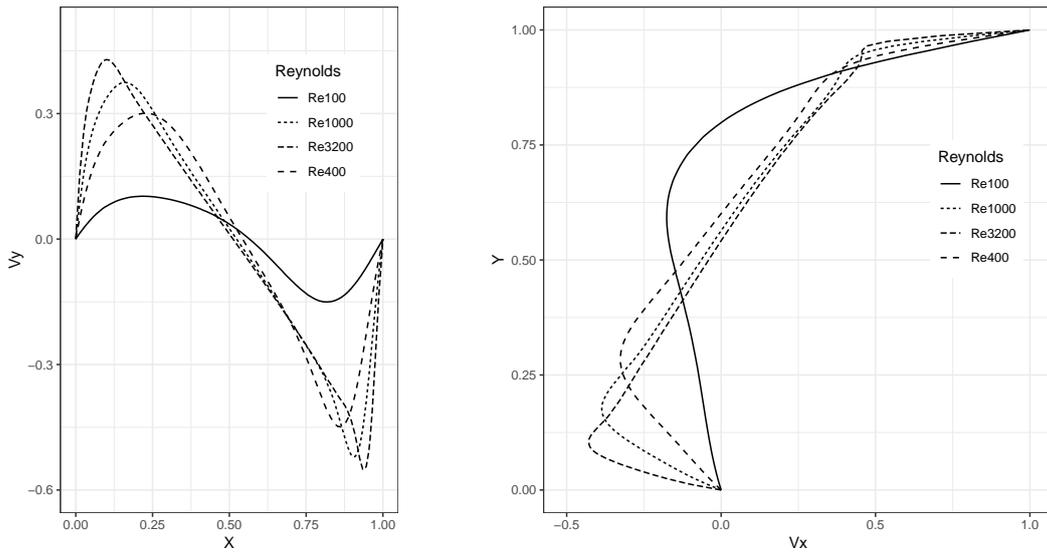
onde $Re = \rho V L / \mu$ é o número de Reynolds, que caracteriza o escoamento.

A geometria adotada na modelagem deste problema é a de um quadrado de lado unitário. Para as suas condições de contorno, assume-se que a velocidade do fluido é nula em todas as suas bordas, exceto a borda superior, que possui velocidade constante igual a 1 m/s na direção x .

O OpenFOAM® possui alguns *solvers* capazes de resolver o problema deste módulo, como o **simpleFoam**. Neste exemplo, o problema da cavidade foi resolvido utilizando-se este módulo que trata do escoamento de fluidos incompressíveis em regime estacionário empregando-se o algoritmo iterativo SIMPLE. Na categorização de *solvers* da plataforma, encontra-se no grupo *incompressible*, dedicado a problemas de dinâmica dos fluidos com escoamento incompressível.

A malha utilizada para este problema foi de 16.366 elementos triangulares, geradas através do gmsh baseada na malha 129×129 (16.641 elementos retangulares) de Ghia et al. [10], artigo com o qual nossos resultados foram comparados. Neste artigo são apresentados resultados computacionais obtidos através da formulação de vorticidade e função de corrente resolvidos com o método *multigrid*, além de resultados computacionais publicados por outros autores. Devido à particularidade do OpenFOAM® de aceitar apenas malhas tridimensionais, estes elementos triangulares na verdade são prismas triangulares, com a condição de contorno *empty* aplicadas às faces normais ao eixo z , o que diz ao *software* para que resolva o problema apenas nas direções x e y .

A Fig. 10 exhibe o perfil das velocidades ao longo de retas que passam pelo centro geométrico da cavidade, tendo boa concordância com os resultados de Ghia et al. [10].

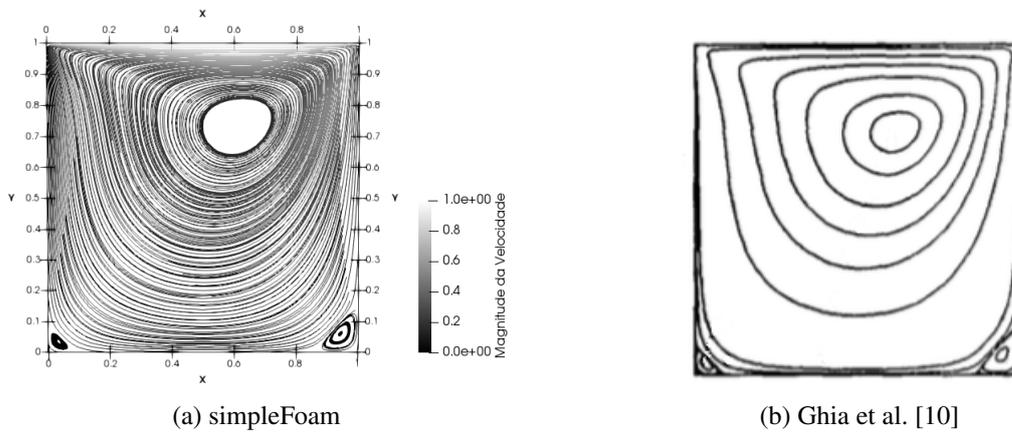


(a) Perfis ao longo de $Y = 0,5 m$.

(b) Perfis ao longo de $X = 0,5 m$.

Figura 10. Resultados da simulação com o simpleFoam

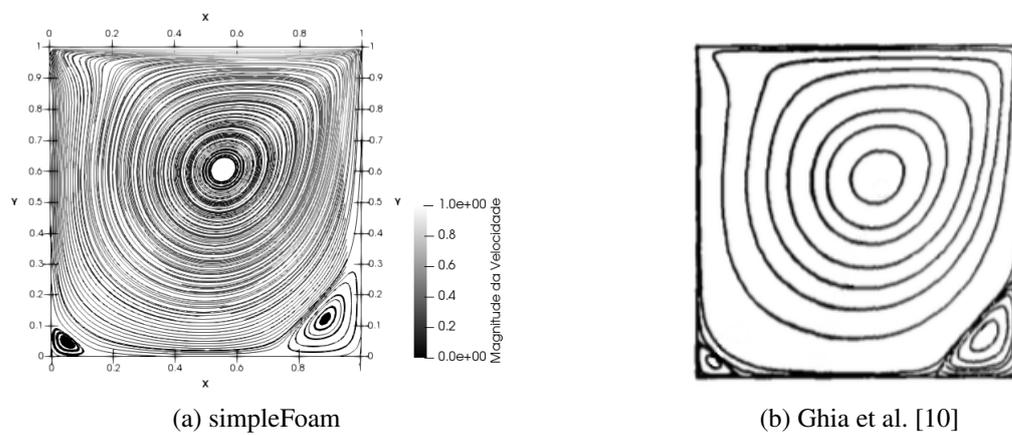
As Figs. 11, 12, 13 e 14 exibem as linhas de corrente para os números de Reynolds 100, 400, 1000 e 3200, respectivamente, comparadas com os resultados de Ghia et al. [10], observando-se concordância dos resultados obtidos em todos os casos.



(a) simpleFoam

(b) Ghia et al. [10]

Figura 11. Linhas de Corrente da Cavity para $Re = 100$.



(a) simpleFoam

(b) Ghia et al. [10]

Figura 12. Linhas de Corrente da Cavity para $Re = 400$.

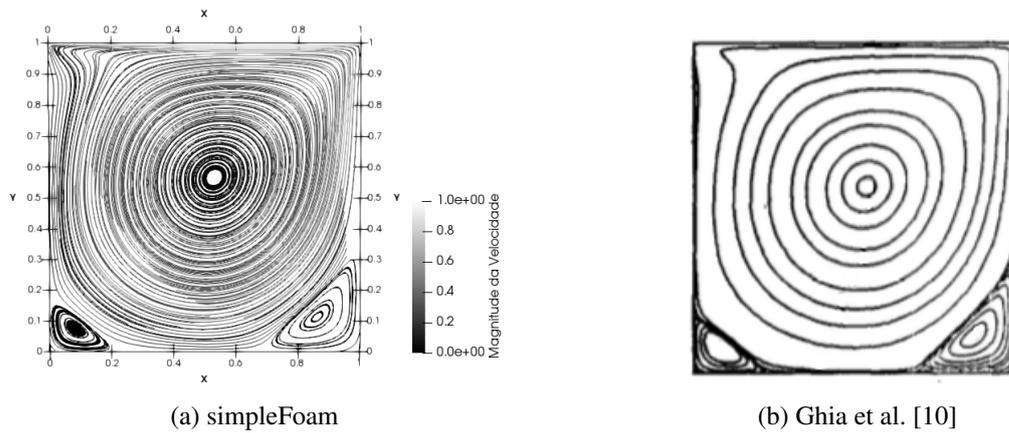


Figura 13. Linhas de Corrente da Cavity para $Re = 1000$.

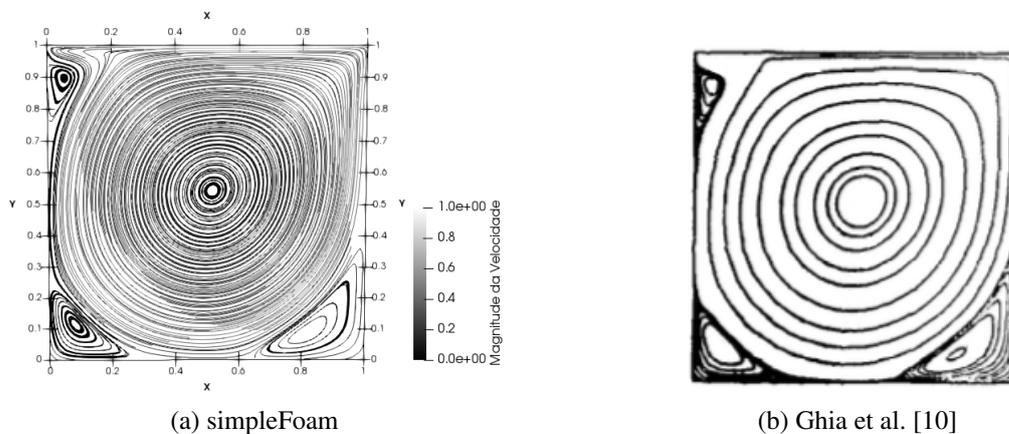


Figura 14. Linhas de Corrente da Cavity para $Re = 3200$.

3.3 O Método da Entalpia-Porosidade para Mudança de Fase

Consideraremos, em seguida, os fenômenos de mudança de fase que envolvem a transição de um material **sólido para líquido** (fusão) ou de **líquido para sólido** (solidificação). A compreensão de tais processos é importante em diversas áreas da engenharia como a metalurgia.

O método da entalpia-porosidade é um método de malha fixa, onde um único conjunto de equações de conservação é aplicado em todo o domínio do problema, tanto na fase líquida quanto na sólida. No modelo computacional, assume-se que o material em estado líquido é um fluido newtoniano incompressível em escoamento laminar viscoso. As propriedades termofísicas são assumidas constantes em cada fase, exceto pela densidade, que varia com a temperatura, empregando-se a aproximação de Boussinesq para contabilizar tal variação. A região de transição de fase, conhecida como **zona mista (mushy)** é modelada como um meio pseudo-poroso tratado numericamente com a inclusão de termos fonte baseados na lei de Darcy e na equação de Karman-Kozeny para escoamentos em meios porosos (Voller and Prakash [11]). Assim, o modelo descrevendo o método da entalpia-porosidade é descrito pelo conjunto de equações (Odone [12], Brent et al. [13]):

$$\nabla \cdot \vec{V} = 0 \tag{10}$$

$$\frac{\partial}{\partial t}(\rho u) + \nabla \cdot (\rho \vec{V} u) - \nabla \cdot (\mu \nabla u) = -\frac{\partial p}{\partial x} - Au \tag{11}$$

$$\frac{\partial}{\partial t}(\rho v) + \nabla \cdot (\rho \vec{V} v) - \nabla \cdot (\mu \nabla v) = -\frac{\partial p}{\partial y} - Av + S_b \tag{12}$$

$$\frac{\partial}{\partial t}(\rho h) + \nabla \cdot (\rho \vec{V} h) - \nabla \cdot (\alpha \nabla h) = -\frac{\partial}{\partial t}(\rho \Delta H) - \nabla \cdot (\rho \vec{V} \Delta H) \tag{13}$$

Na equação da energia, Eq. (13), ρ é a densidade, $\vec{V} = (u, v)$ é o vetor velocidade, k é a condutividade térmica, e $H = h + \Delta H$ é a entalpia total sendo $h = h_{ref} + \int_{T_{ref}}^T c(T)dT$ a entalpia sensível, ΔH o calor latente total, c o calor específico e T a temperatura. Tomando-se $T_{ref} = h_{ref} = 0$, a entalpia total pode ser escrita como $H = cT + \Delta H$.

Nas equações de conservação da quantidade de movimento, Eqs. (11) e (12), o valor de A é definido em termos da fração líquida da célula computacional, $f_l = \Delta H/L$, sendo L o calor latente de mudança de fase, e das equações de Karman-Kozeny e da lei de Darcy, calculada por:

$$A = -C \frac{(1 - f_l)^2}{f_l^3 + q} \quad (14)$$

onde C é uma constante relacionada à constituição da zona mista, e q uma constante de pequeno valor usada para impedir divisão por zero quando $f_l = 0$.

O termo S_b na Eq. (12) é empregado para induzir a convecção natural na parte líquida do domínio, baseado na aproximação de Boussinesq, na qual a densidade é assumida constante em todos os termos das equações da conservação de movimento, exceto para o termo de empuxo S_b . Para gradientes de temperatura relativamente baixos, a variação da densidade com a temperatura pode ser caracterizada de forma linear. Assim, a expressão para o termo S_b pode ser escrita como (Ferziger and Peric [14]):

$$S_b = \rho g \beta (h - h_{ref}) / c \quad (15)$$

onde g é a aceleração da gravidade e β o coeficiente de expansão térmica.

Simulou-se o experimento de derretimento de gálio numa cavidade como descrito por Gau and Visakanta [15] e Voller et al. [16], que apresentam, respectivamente, resultados experimentais e resultados obtidos com códigos computacionais próprios. A geometria do problema foi replicada com total fidelidade, e discretizada em malha triangular, com 1.342 elementos, utilizando-se o gmsh. O solver utilizado para a resolução deste problema foi o **buoyantPimpleFoam**, que resolve escoamento transiente com transferência de calor e convecção natural. Este solver pertence a categoria *heat transfer*, que contém solvers dedicados a escoamentos com transferência de calor e convecção natural (*buoyancy-driven flows*).

O OpenFOAM® possui nativamente métodos para implementar o método da entalpia-porosidade para mudança de fase, através da biblioteca **fvOptions**. A adoção desta biblioteca permite a inclusão de termos fonte nas equações diferenciais do problema, com suas respectivas restrições, a serem definidas pelo usuário no diretório de caso. Isto permite que diversos fenômenos físicos possam ser adicionados ao problema, desde que possam ser modelados como um termo fonte adicional. O método da entalpia-porosidade é descrito pelo **solidificationMeltingSource**, já presente na instalação do *software*. Para que seja reconhecido pelo buoyantPimpleFoam, basta descrever os parâmetros necessários no dicionário **fvOptions**, presente na pasta *system* ou na pasta *constant*, visto que este solver já possui a biblioteca implementada.

Na Fig. 15, é possível ver a frente de mudança de fase do gálio em diferentes instantes da simulação realizada.

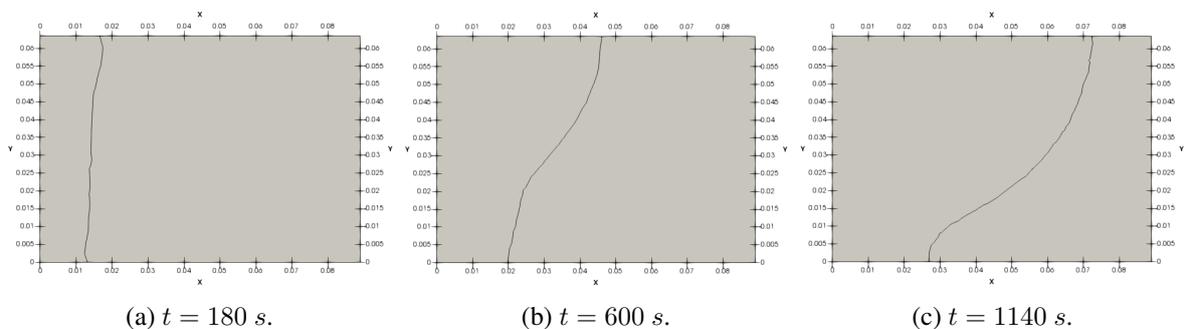


Figura 15. Frente de derretimento do gálio para fração líquida $f_l = 0, 5$.

Na Fig. 16, podemos ver as linhas de correntes da parte líquida.

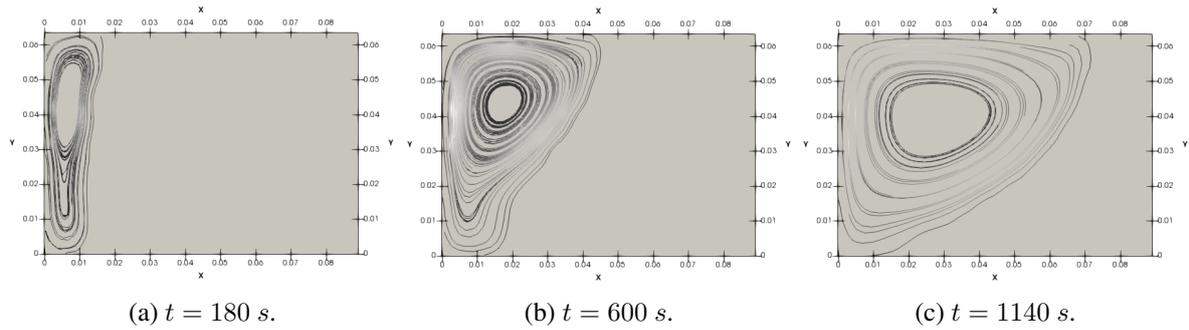


Figura 16. Linhas de corrente da fase líquida em diversos tempos.

Por fim, na Fig. 17, estão presentes gráficos de isotermas, concluindo-se que todos os resultados aqui obtidos são similares aos valores encontrados nas referências citadas

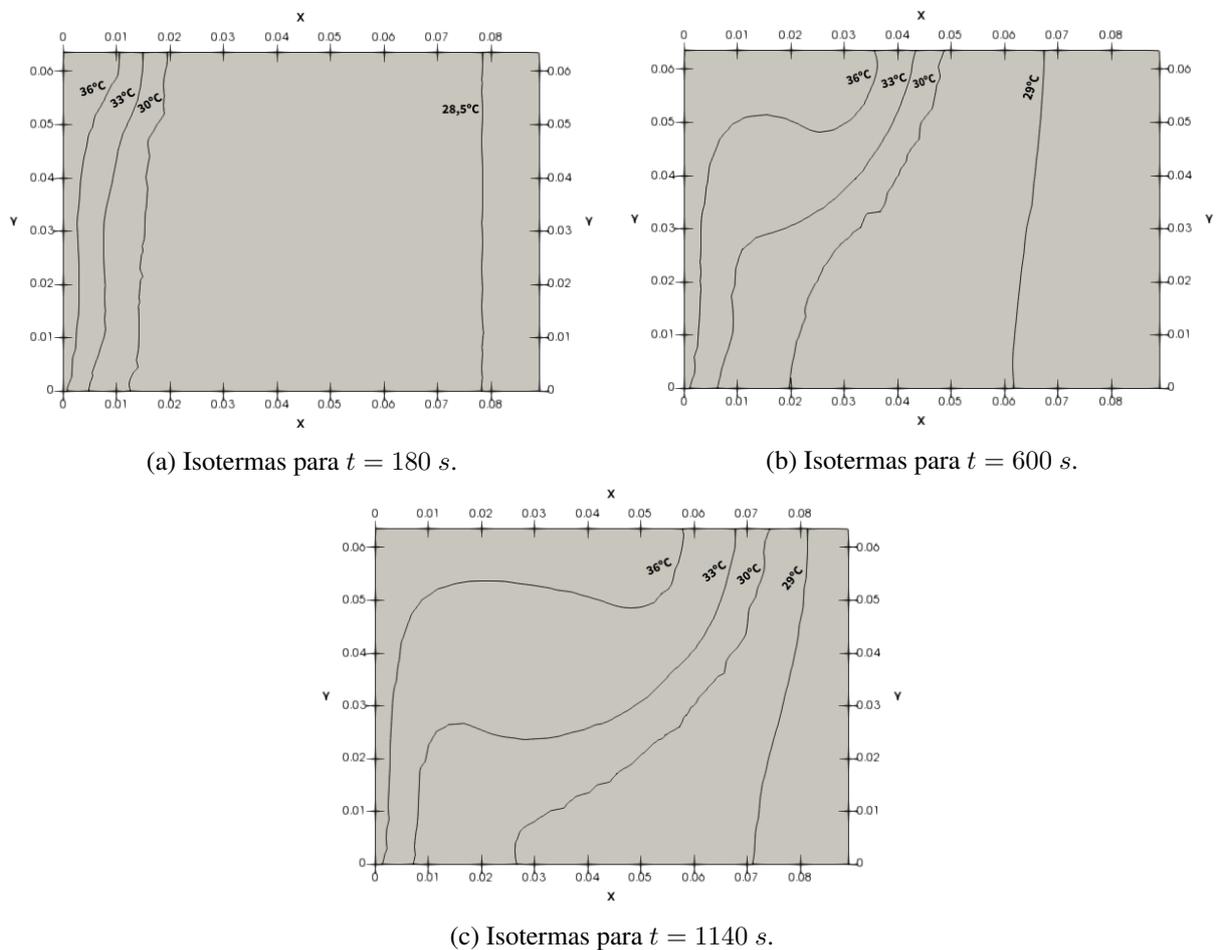


Figura 17. Gráficos das isotermas. As temperaturas decrescem da esquerda para a direita.

3.4 O problema de elasticidade linear

Na determinação do estado mecânico (tensões e deformações) de um sólido submetido a um carregamento, este pode ser tratado como elástico linear em equilíbrio estático, cujo balanço de forças pode ser escrito como (Voller [8], Jasak and Weller [17]):

$$\nabla \cdot \sigma - \rho \vec{f} = 0. \quad (16)$$

No caso em que o carregamento de um corpo resultar em pequenos deslocamentos e em pequenas deformações, as tensões e deformações podem ser relacionadas através da lei de Hooke:

$$\sigma = 2\mu\epsilon + \lambda tr(\epsilon)\mathbf{I} \tag{17}$$

e as deformações são relacionadas aos deslocamentos por:

$$\epsilon = \frac{1}{2} \left(\nabla \vec{u} + (\nabla \vec{u})^T \right) \tag{18}$$

onde \vec{u} são os deslocamentos, \mathbf{I} é o tensor unitário, e μ e λ são os coeficientes de Lamé, relacionando o módulo de elasticidade de Young E ao coeficiente de Poisson ν por:

$$\mu = \frac{E}{2(1 + \nu)} \quad \text{e} \quad \lambda = \frac{\nu E}{(1 + \nu)(1 - \nu)}. \tag{19}$$

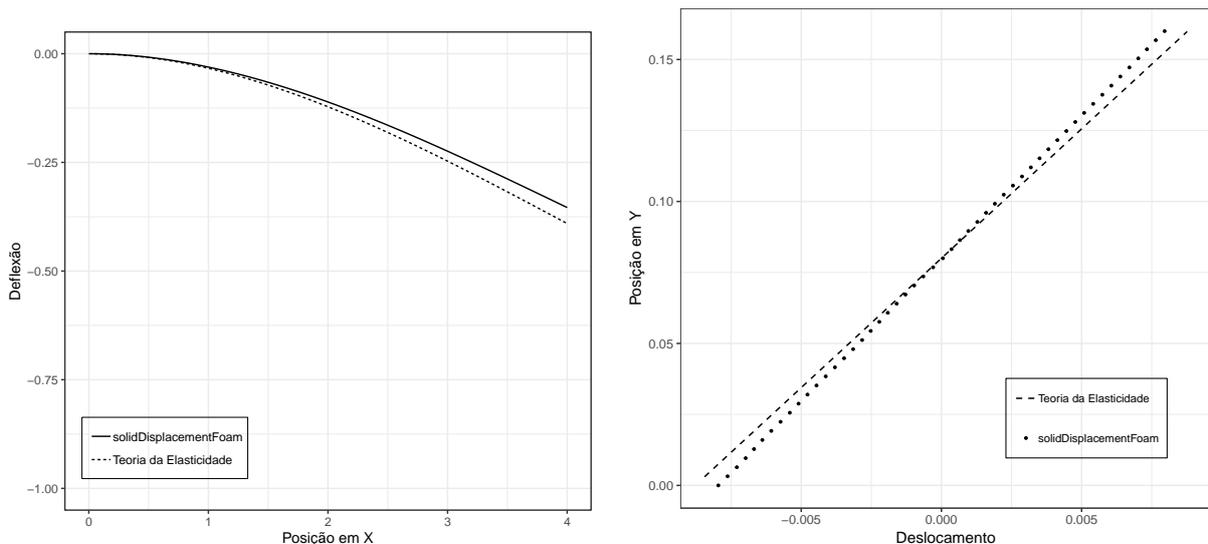
Dadas as relações anteriores, pode-se escrever a equação de equilíbrio em termos do campo de deslocamento \vec{u} como:

$$\nabla \cdot \left[\mu \nabla \vec{u} + \mu (\nabla \vec{u})^T + \lambda \mathbf{I} tr(\nabla \vec{u}) \right] - \rho \vec{f} = 0. \tag{20}$$

A viga em balanço (*cantilever beam*), modelada como um problema de estado plano de tensão, constitui um caso clássico no estudo da mecânica dos sólidos. Através da teoria da elasticidade, como descrita por Timoshenko and Goodier [18], é possível encontrar soluções analíticas que fornecem comparativo para testar os resultados aqui obtidos.

Neste problema, foi utilizado o solver **solidDisplacementFoam** para resolver o problema de uma viga com 100 *kN* aplicados à sua extremidade livre como tensão de cisalhamento uniformemente distribuída. Possui comprimento de 4 *m*, com seção transversal de 0,16 *m* de altura por 0,08 *m* de largura, módulo de elasticidade igual a 200 *GPa* e coeficiente de Poisson $\nu = 0,3$. A barra foi discretizada em 16.922 elementos triangulares, através do *gms*. Os resultados encontrados foram comparados com as soluções analíticas. O **solidDisplacementFoam** é aplicável a problemas de tensão em sólidos com elasticidade linear com pequenos deslocamentos e pequenas deformações.

Nas Figs. 18a e 18b apresentamos, respectivamente, a comparação com a solução analítica da deflexão da viga e do deslocamento horizontal ao longo da altura. Em seguida nas Figs. 19a e 19b comparam-se as tensões normais e tangenciais. Estes três últimos valores foram avaliados no ponto médio da viga ($x = 2m$).



(a) Deflexão vertical da viga em balanço, ao longo de seu eixo principal. (b) Deslocamento horizontal da viga em balanço, em $X = 2m$.

Figura 18. Deslocamentos da viga.

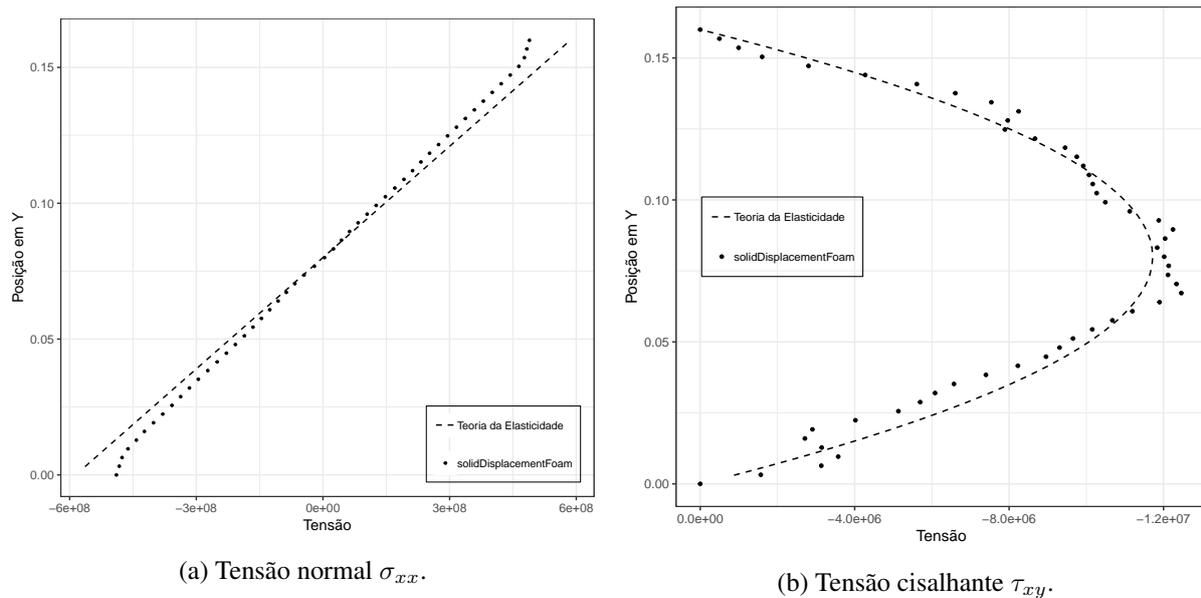


Figura 19. Tensões ao longo da altura da viga para $X = 2 \text{ m}$, em Pascal.

4 Conclusões

Neste trabalho descreveu-se e empregou-se OpenFOAM®, para a resolução de diversos problemas relacionados à mecânica do contínuo, com ênfase em mecânica dos fluidos computacional. De acordo com os modelos propostos e os resultados obtidos, esta ferramenta se mostrou bastante eficiente e versátil, ainda que apresente algumas limitações.

No caso de problemas de mudança de fase em certos processos industriais os materiais a serem solidificados podem passar por um processo de resfriamento envolvendo convecção forçada. Numericamente, este processo é implementado por uma condição de contorno conhecida como condição de Robin. O estudo e a implementação da condição de Robin é uma proposta de continuação deste trabalho para a determinação de condições mais adequadas em processos de solidificação usualmente empregados em diversas indústrias. Outra possibilidade interessante é trabalhar com uma biblioteca em Python conhecida como PyFoam, que permite manipulação e controle dos casos a serem executados pelo OpenFOAM®. Por fim, a necessidade de execução de problemas e casos com grande volumes de dados pode ser melhor otimizada usando-se os recursos de execução em paralelo a serem acoplados com a execução do OpenFOAM®.

Referências

- [1] Maric, T., Hopken, J., & Mooney, K., 2014. *The OpenFOAM Technology Primer*. Sourceflux.
- [2] Chen, G., Xiong, Q., Morris, P. J., Paterson, E. G., Sergeev, A., & Wang, Y., 2014. Openfoam for computational fluid dynamics. *Not. AMS*, vol. 61, n. 4, pp. 354–363.
- [3] OpenFOAM, 2019a. About OpenFOAM | CFD Direct. [Online; acessado em 20 de Julho de 2019].
- [4] OpenFOAM, 2019b. OpenFOAM v7 User Guide: 4.1 OpenFOAM case directory. [Online; acessado em 2 de Agosto de 2019].
- [5] ESI, 2019a. OpenFOAM: User Guide: buoyantPimpleFoam. [Online; acessado em 24 de Julho de 2019].
- [6] ESI, 2019b. OpenFOAM: User Guide: Finite volume options. [Online; acessado em 24 de Julho de 2019].

- [7] Versteeg, Henk Kaarle & Malalasekera, W., 2007. *An introduction to computational fluid dynamics: the finite volume method*. Pearson education.
- [8] Voller, V. R., 2009. *Basic control volume finite element methods for fluids and solids*, volume 1. World Scientific.
- [9] Brooks, A. N. & Hughes, T. J., 1982. Streamline upwind/petrov-galerkin formulations for convection dominated flows with particular emphasis on the incompressible navier-stokes equations. *Computer methods in applied mechanics and engineering*, vol. 32, n. 1-3, pp. 199–259.
- [10] Ghia, U., Ghia, K. N., & Shin, C., 1982. High-re solutions for incompressible flow using the navier-stokes equations and a multigrid method. *Journal of computational physics*, vol. 48, n. 3, pp. 387–411.
- [11] Voller, V. R. & Prakash, C., 1987. A fixed grid numerical modelling methodology for convection-diffusion mushy region phase-change problems. *International Journal of Heat and Mass Transfer*, vol. 30, n. 8, pp. 1709–1719.
- [12] Odone, M. W. N., 2014. Estudos numéricos na solução de problemas de mudança de fase. Master's thesis, Universidade Federal de Juiz de Fora - Programa de Pós-Graduação em Modelagem Computacional, Juiz de Fora, Brasil.
- [13] Brent, A., Voller, V. R., & Reid, K., 1988. Enthalpy-porosity technique for modeling convection-diffusion phase change: application to the melting of a pure metal. *Numerical Heat Transfer, Part A Applications*, vol. 13, n. 3, pp. 297–318.
- [14] Ferziger, J. H. & Peric, M., 2012. *Computational methods for fluid dynamics*. Springer Science & Business Media.
- [15] Gau, C. & Viskanta, R., 1986. Melting and solidification of a pure metal on a vertical wall. *Journal of Heat Transfer*, vol. 108, n. 1, pp. 174–181.
- [16] Voller, V. R., Brent, A., & Prakash, C., 1989. The modelling of heat, mass and solute transport in solidification systems. *International Journal of Heat and Mass Transfer*, vol. 32, n. 9, pp. 1719–1731.
- [17] Jasak, H. & Weller, H., 2000. Application of the finite volume method and unstructured meshes to linear elasticity. *International journal for numerical methods in engineering*, vol. 48, n. 2, pp. 267–287.
- [18] Timoshenko, S. & Goodier, J., 1970. *Theory of Elasticity*. McGraw-Hill.