

CAPÍTULO 1

ESCOAMENTO EM UMA CAVIDADE

1.1 Definição do problema

O problema físico estudado neste tutorial consiste no escoamento de um fluido Newtoniano, isotérmico, incompressível em uma cavidade, onde a fronteira superior desloca-se com velocidade conhecida e as demais fronteiras são estacionárias. A Figura 1.1 mostra a geometria bidimensional do problema.

A solução do escoamento em uma cavidade é um problema clássico adotado na validação e/ou verificação de métodos numéricos [1, 2, 4, 3]. A solução analítica deste tipo de escoamento pode ser determinada apenas para fluidos invíscidos [5]. Este tipo de escoamento gera linhas de corrente fechadas onde a natureza do vórtice depende da razão de aspecto (razão da altura pela largura) da cavidade e do número de Reynolds [2]. O número de Reynolds para esse escoamento é definido por,

$$Re = \frac{U_x A}{\nu} \quad (1.1)$$

onde U_x é a velocidade da fronteira superior, A é a largura da cavidade e ν é a viscosidade cinemática [2].

O estudo experimental deste tipo de escoamento é reportado para diferentes valores de Reynolds e razão de aspecto da cavidade [5, 6, 7, 8]. A Figura 1.2 reproduz os resultados da visualização do escoamento para os casos de Reynolds 1.070 e 4.030 [5, 7].

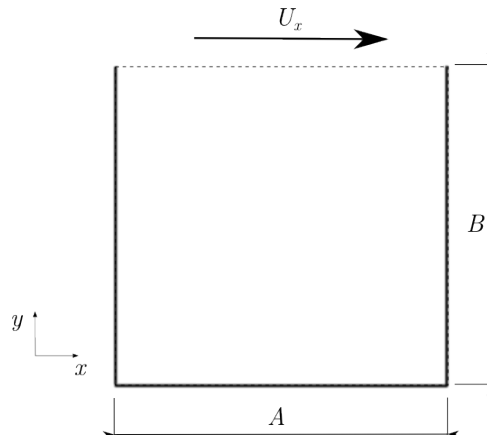
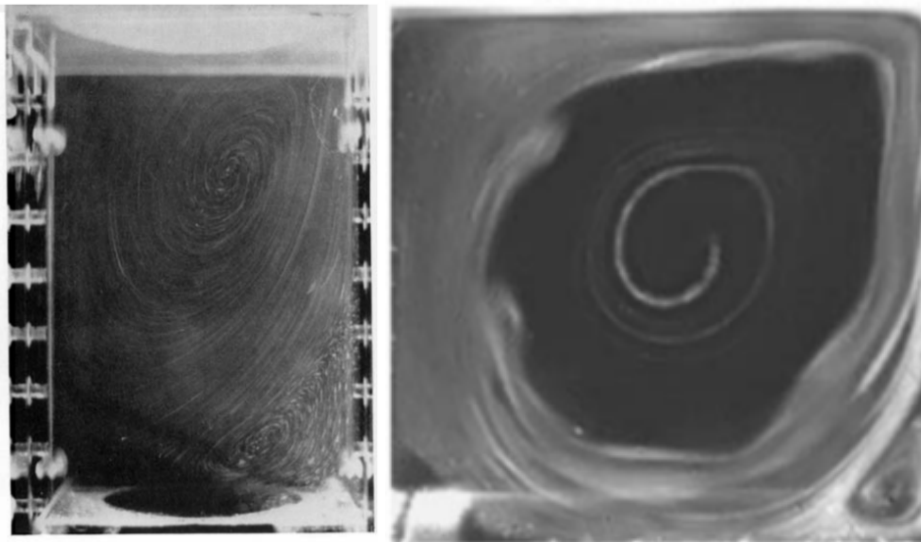


Figura 1.1 Geometria do problema do escoamento em uma cavidade.



(a) $Re = 1.070$ [5]

(b) $Re = 4.030$ [7]

Figura 1.2 Resultado experimental da visualização do escoamento em uma cavidade.

1.2 Modelagem matemática

A equação que governa este problema é a equação de Navier-Stokes para um fluido com massa específica e viscosidade constantes, onde o termo devido a ação de forças de campo foi considerado desprezível.

$$\frac{\partial \mathbf{U}}{\partial t} + \mathbf{U} \cdot \nabla \mathbf{U} - \nu \nabla^2 \mathbf{U} = -\nabla \left(\frac{P}{\rho} \right) \quad (1.2)$$

A condição de contorno para a velocidade é de primeiro tipo, ou Dirichlet, onde a velocidade nas fronteiras estacionárias é nula (condição de não deslizamento) e a velocidade na fronteira superior é conhecida (U_x).

A equação para pressão em um escoamento incompressível, é dada por [9],

$$\nabla^2 \left(\frac{P}{\rho} \right) = -\nabla \cdot [\mathbf{U} \cdot \nabla \mathbf{U}] \quad (1.3)$$

onde a condição de contorno para pressão é valor constante para qualquer fronteira. Isto significa atribuir o valor da pressão nas fronteiras igual ao valor do seu respectivo volume vizinho, ou seja, uma condição de contorno de primeiro tipo. Porém, como o valor do volume vizinho será calculado em tempo de simulação, a forma de lidar com esta condição de contorno significa impor valor da pressão nas fronteiras constante e, portanto, o gradiente da pressão é nulo.

O acoplamento pressão-velocidade pode ser realizado através dos métodos PISO, para um solução transiente, ou SIMPLE, para uma solução estacionária. Por fim, se o escoamento for turbulento, é necessário ainda escolher um modelo de turbulência. No OpenFOAM, as equações que governam esse problema podem ser resolvidas pelos seguintes *solvers*:

1. **pisoFoam** : Escoamento incompressível, transiente (PISO), laminar ou turbulento, fluido Newtoniano ou não-Newtoniano.
2. **pimpleFoam** : Escoamento incompressível, estacionário (PIMPLE), laminar ou turbulento, fluido Newtoniano ou não-Newtoniano.
3. **simpleFoam** : Escoamento incompressível, estacionário (SIMPLE), laminar ou turbulento, fluido Newtoniano ou não-Newtoniano.
4. **icoFoam** : Escoamento incompressível, transiente, laminar e fluido Newtoniano.

O **pisoFoam** foi escolhido para resolver este tutorial pois trata-se do *solver* mais geral.

1.3 Geometria, malha e préprocessamento

A geometria e a malha deste tutorial serão construídas simultaneamente usando o utilitário **blockMesh** do OpenFOAM. O dicionário `blockMeshDict` , localizado no diretório `constant` do caso, contém as instruções necessárias para criar a geometria e malha.

A partir da versão do OpenFOAM-3.0, o dicionário `blockMeshDict` encontra-se no diretório `system` .

A primeira instrução no `blockMeshDict` determina a unidade de comprimento da geometria em relação ao metro. Neste caso, a geometria está em decímetros, conforme mostra o Código 1.1:

Código 1.1 Definição do comprimento da geometria no `blockMeshDict`

```
convertToMeters 0.1;
```

A próxima etapa é definir as coordenadas dos vértices. A Figura 1.3 mostra a geometria da cavidade e seus respectivos vértices. O Código 1.2 é o trecho do `blockMeshDict` onde a declaração dos vértices é feita.

Código 1.2 Definição dos vértices no `blockMeshDict`

```
vertices
(
  (0 0 0)      //Vertice 0
  (1 0 0)      //Vertice 1
  (1 1 0)      //Vertice 2
  (0 1 0)      //Vertice 3
  (0 0 0.1)    //Vertice 4
  (1 0 0.1)    //Vertice 5
  (1 1 0.1)    //Vertice 6
  (0 1 0.1)    //Vertice 7
);
```

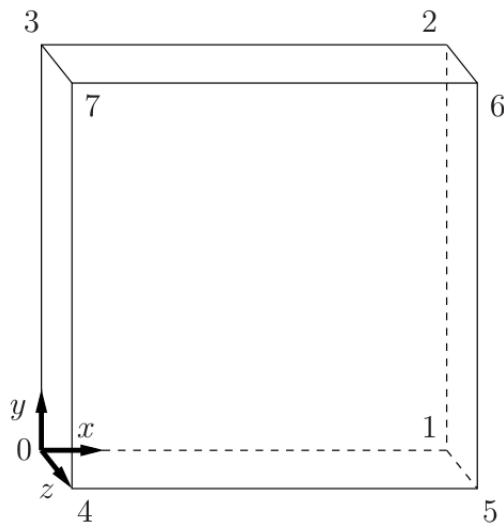


Figura 1.3 Construção da geometria [10].

A construção da malha é dada a partir da definição de blocos dentro da geometria. Neste caso, foi construído um único bloco hexaédrico com 20 divisões na direção x , 20 na direção y , 1 na direção z e sem refino de malha nas fronteiras. O Código 1.3 mostra a estrutura do OpenFOAM para criar um bloco:

Código 1.3 Definição de um bloco no `blockMeshDict`

```
blocks
(
  hex (0 1 2 3 4 5 6 7) (20 20 1) simpleGrading (1 1 1)
);
```

onde primeiro é definido o tipo da malha (`hex` = hexaédrica), depois os vértices que constroem o bloco são declarados, seguidos do número de divisões em cada direção (x, y, z) e, por fim, é declarado uma opção de refino de malha em cada direção.

A última etapa de construção da malha consiste na declaração das fronteiras do domínio. O Código 1.4 é o trecho do `blockMeshDict` onde é feita a declaração das fronteiras. A declaração de uma fronteira começa pela escolha de um nome (qualquer) para identificá-la. Este nome será usado para definir as condições de contorno para velocidade e pressão. Nesta malha, três fronteiras são criadas: `movingWall`, `fixedWalls` e `frontAndBack`. Em seguida, cada fronteira deverá ser identificada através de uma característica topológica (`type`) e dos vértices das faces que a compõem.

Código 1.4 Definição das fronteiras no `blockMeshDict`

```
boundary
(
  movingWall
  {
    type wall;
    faces
    (
      (3 7 6 2)
    );
  }
  fixedWalls
  {
    type wall;
    faces
    (
      (0 4 7 3)
      (2 6 5 1)
      (1 5 4 0)
    );
  }
  frontAndBack
  {
    type empty;
    faces
    (
      (0 3 2 1)
      (4 5 6 7)
    );
  }
);
```

As faces que compõem as fronteiras são identificadas através dos vértices que a constrói. Assim, através da Figura 1.3, é possível identificar que a fronteira superior é formada pela face de vértices (3 7 6 2), que recebeu o nome de `movingWall`. A escolha da característica topológica da fronteira (`type`) depende das características do problema. Neste caso, a fronteira superior, laterais e inferior são todas do tipo `wall`, ou seja, representam uma parede. A diferença entre a face superior e as demais será feita através da condição de contorno para a velocidade. Note que, apesar do problema ser bidimensional, todas as geometrias no OpenFOAM são tridimensionais. Desta forma, para definir o problema com aproximação bidimensional é necessário adotar uma característica topológica particular para as fronteiras que não serão discretizadas (eixo z com 1 divisão). Essas fronteiras foram identificadas como `frontAndBack` e são do tipo `empty`.

Uma vez editado o arquivo `blockMeshDict`, a geometria e malha podem ser construídas através da execução do comando `blockMesh` no diretório do caso. O OpenFOAM tem ainda um outro utilitário, `checkMesh`, que pode ser executado para obter maiores informações sobre a malha.

A próxima etapa de pré-processamento consiste na definição das propriedades do fluido, do regime de escoamento (laminar ou turbulento) e das condições de contorno para velocidade e pressão.

As propriedades do fluido são especificadas no dicionário `transportProperties` no diretório `constant`. É necessário especificar o modelo do fluido (Newtonian) e a viscosidade cinemática ($\nu = \text{nu}$).

Código 1.5 Definição das propriedades do fluido no `transportProperties`

```
transportModel  Newtonian;
nu              nu [ 0 2 -1 0 0 0 0 ] 0.01;
```

O regime do escoamento é definido no dicionário `turbulenceProperties` no diretório `constant`. O Código 1.6 mostra como é feita a escolha do regime de escoamento, através da definição da variável `simulationType`, exemplificada para uma simulação laminar.

Código 1.6 Definição do regime do escoamento no `turbulenceProperties`

```
simulationType  laminar;
```

Existem duas opções no caso de escoamento turbulento: `RASModel` ou `LESModel`. A opção `RASModel` refere-se aos modelos de turbulência baseados nas equações de Navier-Stokes médias no tempo (*Reynolds Average Simulation*) e `LESModel` são os modelos de grande escala conhecidos como *Large Eddy Simulation*. Os parâmetros de cada modelo de turbulência são especificados através do dicionário `RASProperties` ou `LESProperties`, respectivamente, no diretório `constant`.

Na versão OpenFOAM-3.0 a especificação do tipo de modelo de turbulência é dada por apenas RAS ou LES (a palavra "Model" foi retirada). Além disso, os arquivos com os dicionários `RASProperties` e `LESProperties` foram substituídos por subdicionários declarados dentro do arquivo `turbulenceProperties`. Os subdicionários são chamados RAS ou LES, para cada caso.

Existem diversos modelos de turbulência do tipo `RASModel` no OpenFOAM porém, neste tutorial os seguintes serão investigados: `kEpsilon` e `kOmegaSST`. O Código 1.7 é um exemplo do arquivo `RASProperties` onde o modelo `kEpsilon` foi escolhido. Os modelos do tipo `LESModel` não serão investigados neste tutorial.

Código 1.7 Escolha do modelo de turbulência `RASModel` no `RASProperties`

```
RASModel      kEpsilon;
turbulence    on;
printCoeffs   on;
```

A condição de contorno e inicial dos campos de velocidade (U) e pressão (p) são especificadas em um arquivo para cada variável no diretório 0 do caso. Se o escoamento for turbulento, é necessário especificar a condição de contorno/inicial para a energia cinética turbulenta (k), viscosidade turbulenta (nut) e a variável ϵ (`epsilon`) para o modelo `kEpsilon` ou ω (`omega`) para o modelo `kOmegaSST`.

O Código 1.8 apresenta a especificação da condição de contorno e inicial do campo de velocidade. A primeira definição do arquivo é o vetor que representa a unidade da grandeza (m/s), em seguida o valor da velocidade dentro do domínio é inicializado como

uniforme e nulo. As condições de contorno são especificadas para cada fronteira a partir do nome escolhido na etapa de construção de malha. As três fronteiras criadas neste caso foram: `movingWall`, `fixedWalls` e `frontAndBack`. Cada fronteira será especificada através de um tipo (`type`) e um valor (`value`). O tipo `fixedValue` é uma condição de contorno de primeiro tipo onde o valor da variável é prescrita. O valor da velocidade na fronteira superior (`movingWall`) é $U_x = 1 \text{ m/s}$ e nas laterais (`fixedWalls`) é igual a zero. Por fim, o problema é bidimensional e portanto a fronteira `frontAndBack` é do tipo `empty`.

Código 1.8 Arquivo de condição de contorno/inicial para velocidade (U) no diretório 0.

```

dimensions      [0 1 -1 0 0 0 0];
internalField   uniform (0 0 0);
boundaryField
{
    movingWall
    {
        type      fixedValue;
        value     uniform (1 0 0);
    }

    fixedWalls
    {
        type      fixedValue;
        value     uniform (0 0 0);
    }

    frontAndBack
    {
        type      empty;
    }
}

```

O Código 1.9 apresenta a especificação da condição de contorno e inicial do campo de pressão. Note que o campo de pressão especificado na solução do escoamento incompressível no OpenFOAM é um campo de pressão relativa (manométrica) dividido pela massa específica do fluido, ou seja, a pressão é na verdade P/ρ . Por este motivo, a unidade do campo de pressão é m^2/s^2 . A condição de contorno adotada para a pressão nas fronteiras `movingWall` e `fixedWalls` é do tipo `zeroGradient`. Este tipo de condição de contorno atribui o valor da variável do volume vizinho para a face da fronteira. Isto significa que o gradiente da pressão normal à face da fronteira é nulo. A fronteira `frontAndBack` é do tipo `empty` pois o problema é bidimensional.

Código 1.9 Arquivo de condição de contorno/inicial para a pressão relativa (p) no diretório 0.

```

dimensions      [0 2 -2 0 0 0 0];
internalField   uniform 0;
boundaryField
{
    movingWall
    {
        type      zeroGradient;
    }

    fixedWalls
    {
        type      zeroGradient;
    }
}

```

```
frontAndBack
{
    type          empty;
}
}
```

Referências Bibliográficas

- [1] Odus R. Burggraf, Analytical and numerical studies of the structure of steady separated flows, *Journal of Fluid Mechanics*, Volume 24, Issue 01, pp 113 151, 1966.
- [2] James D. Bozeman, Numerical Study of Viscous Flow in a Cavity, *Journal Of Computational Physics*, V. 12, P. 348-363, 1973
- [3] Vivian O'Brien, Closed Streamlines Associated with Channel Flow over a Cavity, *Physics of Fluids* 15, 2089 (1972)
- [4] Ghia, U; Ghia, Kn; Shin, Ct, High-Re Solutions For Incompressible-Flow Using The Navier Stokes Equations And A Multigrid Method, *Journal Of Computational Physics* Volume: 48 Issue: 3 Pages: 387-411 Published: 1982
- [5] Frank Pan and Andreas Acrivos, Steady flows in rectangular cavities, *Journal of Fluid Mechanics*, Volume 28, Issue 04, June 1967, pp 643 655
- [6] Leo F. Donovan, A Numerical Solution Of Unsteady Flow In A Two-Dimensional Square Cavity, *Aiaa Journal*, 1969
- [7] Thierry M. Faure, Panayotis Adrianos Francois Lusseyran, Luc Pastur, Visualizations of the flow inside an open cavity at medium range Reynolds numbers, *Exp Fluids* (2007) 42:169–184
- [8] Tanja Siegmann-Hegerfeld • Stefan Albensoeder Hendrik C. Kuhlmann, Three-dimensional flow in a lid-driven cavity with width-to-height ratio of 1.6, *Exp Fluids* (2013) 54:1526
- [9] Ferziger, J. H. & Peric, M., *NComputational Methods for Fluid Dynamics*, 3rd. Ed, Springer, 2002.
- [10] Christopher J. Greenshields, *OpenFOAM, The Open Source CFD Toolbox, User Guide*, 2015