

Pedro Paulo de Carvalho Brito

**MÉTODO LATTICE-BOLTZMANN APLICADO A  
AERODINÂMICA EXTERNA**

Universidade Federal de Uberlândia  
Curso de Graduação em Engenharia Aeronáutica  
Uberlândia-MG

2017

Pedro Paulo de Carvalho Brito

# **MÉTODO LATTICE-BOLTZMANN APLICADO A AERODINÂMICA EXTERNA**

Trabalho de Conclusão de Curso apresentado ao  
Departamento de Engenharia Mecânica da  
Universidade Federal de Uberlândia, como parte  
das exigências para conclusão do curso de  
graduação em Engenharia Aeronáutica.

Orientador: Prof. Dr. Francisco José de Souza

Universidade Federal de Uberlândia  
Curso de Graduação em Engenharia Aeronáutica  
Uberlândia-MG

2017

Pedro Paulo de Carvalho Brito

## **MÉTODO LATTICE-BOLTZMANN APLICADO A AERODINÂMICA EXTERNA**

Trabalho de Conclusão de Curso apresentado ao  
Departamento de Engenharia Mecânica da  
Universidade Federal de Uberlândia, como parte  
das exigências para conclusão do curso de  
graduação em Engenharia Aeronáutica.

Orientador: Prof. Dr. Francisco José de Souza

Aprovado em: 22 de dezembro de 2017.

Prof. Dr. Carlos Antonio Ribeiro Duarte

Membro da Banca Examinadora

Ms. João Rodrigo Andrade

Membro da Banca Examinadora

Prof. Dr. Francisco José de Souza

Orientador



## RESUMO

A necessidade de métodos voltados para a Fluido Dinâmica Computacional (CFD) que sejam numericamente estáveis mesmo quando aplicados a geometrias complexas, sem a necessidade de realizar grandes simplificações, e sem introduzir grande taxa de dissipação para melhorar a estabilidade, fez com que nos últimos anos o método Lattice-Boltzmann (LB) ganhasse bastante notoriedade tanto no meio acadêmico quando nas indústrias automotiva e aeroespacial.

Com base no exposto acima, o presente trabalho visa analisar a capacidade do método LB em prever as características aerodinâmicas de escoamentos sobre cilindros para números de Reynolds intermediários. Primeiramente foi realizada uma revisão bibliográfica dos avanços do método nos últimos anos com relação à capacidade de simular e prever características fluidodinâmicas em geometrias complexas para números de Reynolds intermediários e no regime turbulento. Em um segundo momento, avaliou-se e comparou o código *open source* OpenLB com o código desenvolvido na UFU pelo autor em Fortran 90 quanto a capacidade de calcular o coeficiente de arrasto e a frequência de emissão de vórtices sobre cilindros para Reynolds igual a 200. Comparou-se também os resultados fornecidos com o método Lattice-Boltzmann e com o método de volumes finitos utilizando o software Ansys Fluent®. Por fim, analisou-se a interferência do tamanho da malha e do número de Mach para a simulação realizada.

Palavras-chave: CFD, Lattice-Boltzmann, Reynolds intermediário, Fortran 90, OpenLB.

## **ABSTRACT**

The continuous need for methods dedicated to Computational Fluid Dynamics (CFD) which are numerically stable even when applied to complex geometries, without the need of performing large simplifications and without introducing a large dissipation rate to improve stability, made the Lattice-Boltzmann Method gain prominence in the last years both in the academia and in the aerospace and automotive industries.

The present work aims to analyze the capacity of the Lattice-Boltzmann Method to predict the aerodynamics characteristics of flows around circular cylinders for intermediate Reynolds numbers. First, a bibliographic review of the latest studies about the Lattice-Boltzmann method related to the capabilities of simulating and predicting flow characteristics over complex geometries in the turbulent regime and for intermediate Reynolds numbers was performed. In a second moment, the open source code, OpenLB, and the code developed at UFU by the author using Fortran 90 were compared with respect to the capacity of calculating the drag coefficient and the vortex shedding frequency over cylinders for Reynolds 200. The results obtained with the Lattice-Boltzmann method also were compared with the finite volume method, using the software Ansys Fluent®. Finally, both the grid size and the Mach number interference were analyzed.

Key-words: CFD, Lattice-Boltzmann, intermediate Reynolds, Fortran 90, OpenLB.

## LISTA DE FIGURAS

Figura 1 - Esquema da metodologia MRT .....	23
Figura 2 - Arranjo D1Q3 .....	25
Figura 3 - Arranjo D2Q9 .....	26
Figura 4 – Arranjos Tridimensionais .....	27
Figura 5 - Vorticidade (esquerda) e coeficientes de pressão para NACA 0012.....	33
Figura 6 - Esquema do aerofólio simulado .....	34
Figura 7 - Malha aplicada no estudo. Slat (esquerda) e Flap (direita) .....	34
Figura 8 - Comparação entre LBM-VLES, RANS-2D e dados experimentais.....	35
Figura 9 - Comparação entre LBM-VLES e dados experimentais para vorticidade (superior) e energia cinética turbulenta (inferior).....	36
Figura 10 - Geometria ensaiada no túnel de vento AMES (NASA Langley).....	36
Figura 11 - Corte na malha ao longo da envergadura para o setup 2 (superior) e setup 1 (inferior) .....	37
Figura 12 – Comparação para CL entre a simulação (PowerFLOW) e os dados experimentais	38
Figura 13 - Comparação para CD entre a simulação (PowerFLOW) e os dados experimentais	38
Figura 14 - Coeficiente de arrasto para um cilindro quadrado em função do número de Reynolds .....	40
Figura 15 - Comparação do perfil de velocidade de três métodos numéricos com resultado experimental.....	41
Figura 16 - Comparação das linhas e corrente do perfil de velocidade com Ghia (1982). .....	44
Figura 17 - Representação da geometria e do domínio empregados na simulação.....	45
Figura 18 - Detalhe da malha para a simulação utilizando volumes finitos.....	46
Figura 19 - Coeficiente de arrasto utilizando LB com a condição de Bouzidi e bounce-back simples.....	47
Figura 20 - Coeficiente de arrasto com o código em Fortran 90 e OpenLB .....	48
Figura 21 - Campo de velocidade e isolinhas de velocidade U.....	49
Figura 22 - Tendência do coeficiente de arrasto no início da simulação para o método de volumes finitos .....	50
Figura 23 - Comparação dos três métodos de solução para coeficiente de arrasto .....	51
Figura 24 - Coeficiente de arrasto para as três simulações .....	53

## LISTA DE TABELAS

Tabela 1 - Pesos e velocidades para os modelos tridimensionais: D3Q15, D3Q19 e D3Q27 ....	26
Tabela 2 - Coeficiente de arrasto e número de Strouhal para diferentes autores .....	40
Tabela 3 - Coeficientes de arrasto para Reynolds entre 10 e 40 .....	49
Tabela 4 - Comparação dos resultados.....	51
Tabela 5 - Tempo de simulação .....	52
Tabela 6 - Parâmetros das simulações realizadas.....	53
Tabela 7 - Resultados para diferentes parâmetros.....	54



## LISTA DE SÍMBOLOS

LB	Lattice-Boltzmann
PDF	Função de distribuição de probabilidade
BGK	Operador de Bhatnagar-Gross-Krook
MRT	Tempo de relaxação múltiplo
TRT	Tempo de relaxação duplo
Ma	Número de Mach
SA	Modelo de turbulência de Spalart Allmaras
K- $\epsilon$ RG	Modelo de turbulência K- $\epsilon$ ( <i>Renormalization Group</i> )
$\tau$	Fator de relaxação (tempo de relaxação)
$\tau_{eff}$	Fator de relaxação efetivo
$\omega$	Taxa de relaxação
$f$	Função de distribuição
$f^{eq}$	Função de distribuição de equilíbrio
RANS	<i>Reynolds Average Navier-Stokes Equations</i>
URANS	<i>Unsteady Reynolds Average Navier-Stokes Equations</i>
DNS	<i>Direct Numerical Simulations</i>
LES	<i>Large Eddy Simulaton</i>
LBM-VLES	<i>Lattice-Boltzmann Very Large Eddy Simulations</i>
LBM-LES	<i>Lattice-Boltzmann Large Eddy Simulations</i>
LBM-DNS	<i>Lattice-Boltzmann Direct Numerical Simulation</i>
RSE	<i>Reynolds Stress Equations</i>

## SUMÁRIO

<b>1. INTRODUÇÃO .....</b>	<b>13</b>
<b>2. OBJETIVOS .....</b>	<b>15</b>
<b>2.1. OBJETIVO GERAL .....</b>	<b>15</b>
<b>2.2. OBJETIVOS ESPECÍFICOS .....</b>	<b>15</b>
<b>3. FUNDAMENTAÇÃO TEÓRICA .....</b>	<b>16</b>
<b>3.1. TEORIA CINÉTICA DOS GASES .....</b>	<b>16</b>
<b>3.1.1. Introdução à Teoria Cinética .....</b>	<b>16</b>
<b>3.1.2. Teoria Cinética – Dinâmica da Partícula .....</b>	<b>17</b>
<b>3.2. FUNÇÃO DE DISTRIBUIÇÃO .....</b>	<b>18</b>
<b>3.2.1. Equação do Transporte de Boltzmann .....</b>	<b>19</b>
<b>3.2.2. Aproximação de Bhatnagar-Gross-Krook (BGK) .....</b>	<b>20</b>
<b>3.2.3. Limitações da Metodologia BGK .....</b>	<b>21</b>
<b>3.2.4. Modelo MRT .....</b>	<b>22</b>
<b>3.2.5. Modelo TRT .....</b>	<b>23</b>
<b>3.3. ARRANJOS DE REDE .....</b>	<b>24</b>
<b>3.3.1. Modelos Unidimensionais .....</b>	<b>24</b>
<b>3.3.2. Modelos Bidimensionais .....</b>	<b>25</b>
<b>3.3.3. Modelos Tridimensionais .....</b>	<b>26</b>
<b>3.4. SIMULAÇÃO EM FLUIDOS INCOMPRESÍVEIS .....</b>	<b>27</b>
<b>3.4.1. Função Distribuição de Equilíbrio .....</b>	<b>27</b>
<b>3.4.2. Número de Reynolds e Mach .....</b>	<b>28</b>
<b>3.5. CONDIÇÕES DE CONTORNO .....</b>	<b>29</b>
<b>3.5.1. Bounce Back .....</b>	<b>30</b>
<b>3.5.2. Condições de Contorno com Velocidade Conhecida .....</b>	<b>30</b>
<b>3.5.3. Condição de Contorno Aberta .....</b>	<b>31</b>
<b>3.5.4. Condição de Contorno Periódica .....</b>	<b>31</b>

<b>4. REVISÃO BIBLIOGRÁFICA.....</b>	<b>32</b>
<b>4.1. Modelos LBM-URANS .....</b>	<b>32</b>
<b>4.2. Modelos LBM-LES .....</b>	<b>39</b>
<b>5. SIMULAÇÃO DE ESCOAMENTO SOBRE CILINDROS.....</b>	<b>42</b>
<b>5.1. OpenLB .....</b>	<b>42</b>
<b>5.2. Código Lattice-Boltzmann Desenvolvido .....</b>	<b>43</b>
<b>5.3. Simulação Sobre Cilindros .....</b>	<b>44</b>
<b>5.3.1. Parâmetros da Simulação .....</b>	<b>44</b>
<b>5.3.2. Condições de Contorno .....</b>	<b>46</b>
<b>5.3.3. Resultados Obtidos.....</b>	<b>47</b>
<b>6. INFLUÊNCIA DO TAMANHO DA MALHA E DO NÚMERO DE MACH .....</b>	<b>53</b>
<b>7. CONCLUSÃO .....</b>	<b>55</b>
<b>8. REFERÊNCIAS BIBLIOGRÁFICAS .....</b>	<b>57</b>
<b>9. APÊNDICE.....</b>	<b>59</b>



## 1. INTRODUÇÃO

Ao longo dos últimos anos, a Fluido Dinâmica Computacional (CFD) tem tido um papel fundamental no desenvolvimento de produtos mais eficientes, tais como aeronaves, automóveis e trens de alta velocidade. Além de permitir a otimização do projeto, há também a vantagem de reduzir consideravelmente o custo e o ciclo de desenvolvimento, já que muitos experimentos dispendiosos e complexos não precisam ser realizados.

Do ponto de vista científico, a Dinâmica dos Fluidos Computacional também revela-se como ferramenta essencial à compreensão e análise de vários fenômenos cuja investigação experimental é difícil ou mesmo impossível em alguns casos.

Apesar de sua extensiva aplicação no mundo moderno, os métodos comumente empregados em CFD não têm uma base matemática sólida, e foram derivados nas décadas de 80 a 90 com base nas equações de Navier-Stokes (FERZIGER e PERIC, 2013). É interessante notar que tais métodos foram desenvolvidos principalmente por engenheiros que procuravam soluções práticas para problemas envolvendo escoamentos. De forma bastante empírica, os métodos de discretização das equações de Navier-Stokes foram propostos e testados, buscando-se evitar os problemas comuns de instabilidade numérica devidos à precisão finita dos processadores digitais. Infelizmente, uma consequência do desenvolvimento destes métodos foi o aumento da complexidade dos algoritmos envolvidos, caracterizados pelo alto grau de não linearidade das equações diferenciais parciais e pelo alto número de graus de liberdade existentes. Mesmo para problemas simples como o escoamento laminar de um fluido em um canal, vários sistemas lineares envolvendo milhares de incógnitas devem ser resolvidos, resultando em um custo computacional considerável. Considerando escoamentos em regime turbulento, o tempo de CPU cresce conforme uma potência do número de Reynolds, e a solução precisa em tempo hábil exige clusters de processadores. Mesmo os algoritmos mais popularmente utilizados, da família SIMPLE (FERZIGER e PERIC, 2013), são inerentemente ineficientes em função do chamado acoplamento pressão-velocidade, exemplificando que grande parte do problema está nas técnicas numéricas aplicadas para providenciar soluções aproximadas para as equações de Navier-Stokes.

Outra desvantagem das equações de Navier-Stokes é que sua aplicabilidade é limitada pela hipótese do contínuo, utilizada em sua derivação. Embora esta hipótese seja válida para diversos problemas de interesse prático, há várias situações na engenharia que

demandam uma abordagem mais adequada, como nanotecnologia, voo hipersônico e dinâmica molecular. Nestes casos, a aplicação da hipótese do contínuo pode gerar resultados irrealistas, já que a escala de interesse é da ordem da distância média entre partículas que compõem o fluido.

Como uma alternativa interessante à simulação das equações de Navier-Stokes, surge a abordagem do método Lattice-Boltzmann (SUCCI, 2001. WOLF-GLADROW, 2005).

O método Lattice-Boltzmann consiste em resolver a equação de Boltzmann (CHAPMAN e COWLING, 1970), que por sua vez pode ser compreendida como análoga das equações de Navier-Stokes em nível molecular (descrição cinética). Essencialmente, a equação de Boltzmann descreve a dinâmica espacial-temporal de uma grandeza estatística chamada de função de distribuição de probabilidade (PDF), portanto é um método inerentemente transiente. A principal vantagem em relação às equações de Navier-Stokes é não estar limitada pela hipótese do contínuo, o que permite sua aplicação a uma gama de problemas mais vasta que as primeiras. É possível, portanto, descrever o movimento de fluidos que não se estejam no chamado regime hidrodinâmico.

Outra grande vantagem do método Lattice-Boltzmann é sua eficiência. O algoritmo é consideravelmente mais simples que aquele necessário para a solução das equações do contínuo, já que sistemas lineares não precisam ser resolvidos. Também não há o chamado acoplamento pressão-velocidade, pois a pressão é obtida de uma equação de estado algébrica. De forma sintética, apenas partículas com velocidade discreta e determinados pesos descrevem a PDF em cada ponto. O avanço no tempo de um escoamento é descrito por duas operações explícitas: advecção das partículas e colisão, as quais serão vistas a seguir. As propriedades macroscópicas de interesse como velocidade são simplesmente calculadas com base em médias em cada ponto. Conclui-se assim que é possível descrever escoamentos complexos com este método com recursos mais modestos que aqueles normalmente necessários para a solução das equações de Navier-Stokes. Em função de sua simplicidade, a extensão do método para aplicação em computação de alto desempenho (paralelização) é uma tarefa bastante realizável, diferentemente das equações do contínuo.

## **2. OBJETIVOS**

### **2.1. OBJETIVO GERAL**

Avaliar o método LB quanto à capacidade de prever as características fluido dinâmicas em escoamentos com número de Reynolds intermediários e ampliar os conhecimentos adquiridos ao longo da graduação em engenharia aeronáutica.

### **2.2. OBJETIVOS ESPECÍFICOS**

Estipulou-se os seguintes objetivos específicos:

- Comparar a capacidade do código desenvolvido com o código *open source* OpenLB.
- Avaliar as limitações do código desenvolvido.
- Comparar a capacidade do método LB em prever o coeficiente de arrasto sobre cilindros com o método de volumes finitos.

### **3. FUNDAMENTAÇÃO TEÓRICA**

#### **3.1. TEORIA CINÉTICA DOS GASES**

##### **3.1.1. Introdução à Teoria Cinética**

Existem duas abordagens principais quando se fala em simular equações de transporte (calor, massa e momento), contínua e discreta. Na primeira abordagem equações diferenciais parciais ou ordinárias podem ser alcançadas aplicando a conservação da energia, massa e momento para um volume de controle infinitesimal. No entanto, devido à dificuldade em resolver as equações governantes em função da não-linearidade, condições de contorno e geometrias complexas, esquemas de diferenças finitas, elementos finitos e volumes finitos são utilizados para converter as equações diferenciais com suas devidas condições de contorno e condições iniciais em sistemas de equações algébricas (MOHAMAD, 2011). Esses sistemas de equações algébricas podem ser resolvidos iterativamente até que a convergência seja garantida. Primeiramente as equações são identificadas, em sua maioria, equações diferenciais parciais. Em seguida, é realizada a discretização do domínio em volumes, nós ou elementos dependendo do método de solução. Pode-se interpretar esse passo como se um volume ou nó ou elemento contivesse uma coleção de partículas, assim a escala é macroscópica. A velocidade, pressão e temperatura de todas essas partículas são representadas por um valor nodal, ou pela média sobre um volume finito ou assumindo que varia linearmente de um nó para outro e as propriedades como viscosidade, condutividade térmica, calor específico, etc. são em geral valores conhecidos (parâmetros de entrada).

No outro extremo, o meio pode ser considerado por um conjunto de partículas (átomos, moléculas) e essas partículas colidem umas com as outras. Essa é a escala microscópica. Assim precisamos identificar as forças e resolver as equações diferenciais ordinárias referentes à segunda lei de Newton (conservação de momento). A cada passo de tempo precisamos identificar a posição e a velocidade de cada partícula, ou seja, a trajetória de cada uma. Nesse nível, a temperatura e a pressão estão relacionadas à energia cinética das partículas (massa e velocidade) e à frequência das mesmas (MOHAMAD, 2011). Esse método é chamado dinâmica molecular. O método da dinâmica molecular exige recursos computacionais que não estão disponíveis atualmente e provavelmente não



estarão nas próximas décadas para aplicação em problemas industriais de grande escala. No entanto, não é necessário conhecer o comportamento de cada partícula para cada instante de tempo se o interesse é resolver problemas de escala macroscópica, e sim o comportamento de um conjunto de partículas.

A ideia de considerar o comportamento de um conjunto de partículas como uma unidade é o que fundamenta o método Lattice-Boltzmann. A ideia do método é interessante, pois desfruta de um pouco das duas abordagens citadas, micro e macro, portanto trata-se de um método de mesoescala. As propriedades do conjunto de partículas mencionado podem ser representadas por uma função de distribuição.

### 3.1.2. Teoria Cinética – Dinâmica da Partícula

Sabemos que as moléculas, visualizadas de forma esférica movem-se de maneira randômica no espaço livre, e ainda, que esse movimento satisfaz a conservação de massa, momento e energia. Sendo assim, a segunda lei de Newton pode ser aplicada, implicando que a taxa de mudança de momento linear equivale a força líquida imposta, como mostra a Equação (3.1) a seguir.

$$\mathbf{F} = \frac{d(m\mathbf{c})}{dt} \quad (3.1)$$

onde  $\mathbf{F}$  representa as forças externas e intermoleculares,  $m$  é a massa da partícula,  $\mathbf{c}$  é o vetor velocidade da partícula e  $t$  é o tempo. Para uma massa constante, a equação anterior pode ser simplificada como mostra a Equação (3.2).

$$\mathbf{F} = m \frac{d\mathbf{c}}{dt} = m\mathbf{a} \quad (3.2)$$

onde  $\mathbf{a}$  é o vetor aceleração. A posição da partícula pode ser determinada pela definição de velocidade expressa pela Equação (3.3).

$$\mathbf{c} = \frac{d\mathbf{r}}{dt} \quad (3.3)$$

onde  $\mathbf{r}$  é o vetor posição da partícula com relação à origem.

Nas simulações da dinâmica molecular, as equações acima são solucionadas considerando  $\mathbf{F}$  como uma função conhecida. Assim, se uma força  $\mathbf{F}$  é aplicada em uma

partícula de massa  $m$ , a velocidade da partícula mudará de  $\mathbf{c}$  para  $\mathbf{c} + \mathbf{F}dt/m$  e a sua posição muda de  $\mathbf{r}$  para  $\mathbf{r} + \mathbf{c}dt$ . Na ausência de uma força externa, a partícula move-se livremente de uma posição para outra sem mudar sua direção e velocidade, assumindo que nenhuma colisão ocorre. Esse último fator mencionado (colisões) é influenciado pela pressão, temperatura e energia cinética do sistema, grandezas essas que podem alterar o estado das partículas (MOHAMAD, 2011).

### 3.2. FUNÇÃO DE DISTRIBUIÇÃO

Devido à grande dificuldade em formular todas as partículas que compõem a matéria, Maxwell propôs que o conhecimento da velocidade e posição de cada partícula em todos os espaços de tempo não é importante, ao invés disso sugere-se o uso da função de distribuição, um importante parâmetro para caracterizar o efeito das partículas. A ideia principal da função de distribuição consiste em determinar qual a porcentagem de moléculas, possuem velocidades dentro de certa faixa em um dado instante de tempo e em certo local.

Dessa maneira, para um gás com  $N$  partículas, o número de partículas na direção  $x$  com velocidades entre  $c_x$  e  $c_x + dc_x$  é  $Nf(c_x)dc_x$ . A função  $f(c_x)$  é justamente a fração das partículas que possuem velocidades no intervalo desejado e na direção  $x$ . De maneira similar, para as outras direções, a função de distribuição de probabilidade pode ser definida como anteriormente. Desse modo a probabilidade da velocidade estar entre determinada faixa para as três direções é  $Nf(c_x)f(c_y)f(c_z)dc_xdc_ydc_z$ . Logo a equação anterior pode ser integrada sobre todos os valores possíveis para velocidade, resultando em uma análise para todas as  $N$  partículas como expresso pela Equação (3.4) abaixo.

$$\iiint f(c_x)f(c_y)f(c_z)dc_xdc_ydc_z = 1 \quad (3.4)$$

Sabendo que qualquer direção pode ser  $x$  ou  $y$  ou  $z$ , a função de distribuição não deve depender da direção, mas somente da velocidade das partículas. Logo, podemos escrever a Equação (3.5).

$$f(c_x)f(c_y)f(c_z) = \Phi(c_x^2 + c_y^2 + c_z^2) \quad (3.5)$$

onde  $\Phi$  é outra função desconhecida que precisa ser determinada. A possível função que possui as propriedades da Equação (3.5) é a logarítmica ou a exponencial como mostram as Equações (3.6) e (3.7).

$$\log A + \log B = \log(AB) \quad (3.6)$$

$$e^A + e^B = e^{(A+B)}. \quad (3.7)$$

Pode ser mostrado que uma forma apropriada para a função de distribuição é,  $f(c_x) = Ae^{-Bc_x^2}$ , onde A e B são constantes. Esses e outros resultados podem ser encontrados em Succi (2011) e Mohamad (2011).

De maneira geral, Maxwell forneceu uma função de distribuição para um gás ideal no equilíbrio, porém não mencionou como esse equilíbrio é alcançado, o que mais tarde foi proposto por Boltzmann e é a base do método estudado nesse trabalho.

### 3.2.1. Equação do Transporte de Boltzmann

A função de distribuição  $f(r, c, t)$ , mencionada anteriormente, representa um número de moléculas, o qual não sofre variação antes e depois da aplicação de uma força caso nenhuma colisão ocorra entre estas. Assim podemos escrever a Equação (3.8).

$$f(r + cdt, c + Fdt, t + dt)drdc - f(r, c, t)drdc = 0 \quad (3.8)$$

No entanto, caso alguma colisão ocorra entre as moléculas, haverá uma diferença entre estas no intervalo  $drdt$ . A taxa de mudança entre a situação final e inicial da função de distribuição é chamada de operador de colisão,  $\Omega$ . Então a evolução do número de moléculas pode ser escrita como mostra a Equação (3.9).

$$f(r + cdt, c + Fdt, t + dt)drdc - f(r, c, t)drdc = \Omega(f)drdc dt \quad (3.9)$$

onde dividindo a equação acima por  $drdc dt$  e fazendo  $dt \rightarrow 0$ , temos a equação (3.10).

$$\frac{df}{dt} = \Omega(f) \quad (3.10)$$

A Equação (3.10) acima implica que a taxa total de mudança da função de distribuição equivale a taxa de colisões. Desde que  $f$  seja uma função de  $r, c, t$  e sabendo que o vetor  $r$  pode ser escrito em coordenadas cartesianas como  $r = xi + yj + zk$  a Equação (3.10) pode ser escrita na forma da Equação (3.11) mostrada abaixo.

$$\frac{df}{dt} = \frac{\partial f}{\partial r} c + \frac{\partial f}{\partial c} a + \frac{\partial f}{\partial t} \quad (3.11)$$

onde  $\mathbf{a}$  é igual a  $d\mathbf{c}/dt$ , a aceleração e pode ser relacionada com a força  $\mathbf{F}$  pela segunda lei de Newton,  $\mathbf{a} = \mathbf{F}/m$ .

Logo a equação do transporte de Boltzmann pode ser escrita como mostra a Equação (3.12).

$$\frac{\partial f}{\partial t} + \frac{\partial f}{\partial r} c + \frac{F}{m} \frac{\partial f}{\partial c} = \Omega \quad (3.12)$$

O operador  $\Omega$  é uma função de  $f$  e precisa ser determinado para resolver a equação de Boltzmann. Para sistemas sem uma força externa, a equação de Boltzmann pode ser escrita no formato da Equação (3.13).

$$\frac{\partial f}{\partial t} + \mathbf{c} \cdot \nabla f = \Omega \quad (3.13)$$

### 3.2.2. Aproximação de Bhatnagar-Gross-Krook (BGK)

A dificuldade existente na resolução da equação de Boltzmann encontra-se no termo de colisão, porém, é possível fazer uma aproximação para um operador mais simples sem introduzir um erro significativo, conhecida como aproximação de BGK. Fazendo uso disto o operador de colisão  $\Omega$  pode ser expresso pela Equação (3.14).

$$\Omega = \omega(f^{eq} - f) = \frac{1}{\tau} (f^{eq} - f) \quad (3.14)$$

onde o coeficiente  $\omega$ , ( $\omega = 1/\tau$ ) é denominado frequência de colisão e  $\tau$  é chamado fator de relaxação. Após introduzir a aproximação BGK, a equação de Boltzmann (sem forças externas) pode ser aproximada pela Equação (3.15).

$$\frac{\partial f}{\partial t} + \mathbf{c} \cdot \nabla f = \frac{1}{\tau} (f^{eq} - f) \quad (3.15)$$

No método de Lattice-Boltzmann, a Equação (3.15) pode ser aplicada em direções específicas. A discretização da equação de Boltzmann pode ser escrita ao longo dessas direções específicas como mostra a Equação (3.16).

$$\frac{\partial f_i}{\partial t} + c_i \nabla f_i = \frac{1}{\tau} (f_i^{eq} - f_i) \quad (3.16)$$

A relação acima é a principal equação no método Lattice-Boltzmann e substitui a equação de Navier-Stokes em simulações CFD.

Na forma discretizada a Equação (3.16) assume o modelo da Equação (3.17) mostrada abaixo.

$$f_i(r + c_i \Delta t, t + \Delta t) = f_i(r, t) + \frac{\Delta t}{\tau} [f_i^{eq}(r, t) - f_i(r, t)] \quad (3.17)$$

A Equação (3.17) pode ser decomposta em duas partes que são computadas em série: colisão e propagação (*streaming*).

1. Colisão, também denominada relaxação, é simplesmente uma operação algébrica local e pode ser representada pela Equação (3.18) representada abaixo.

$$f_i^*(r, t) = f_i(r, t) + \frac{\Delta t}{\tau} [f_i^{eq}(r, t) - f_i(r, t)] \quad (3.18)$$

2. Propagação para os nós vizinhos, descrito pela Equação (3.19).

$$f_i(r + c_i \Delta t, t + \Delta t) = f_i^*(r, t) \quad (3.19)$$

Uma vez que as duas operações acima são finalizadas, um passo de tempo terá sido computado. Como pode ser observado, a grande vantagem desse modelo reside na sua simplicidade e possibilidade de aplicação em diversos problemas diferentes, sendo necessário somente especificar uma função de distribuição de equilíbrio diferente e um termo fonte (força externa).

### 3.2.3. Limitações da Metodologia BGK

O modelo BGK é uma maneira simples e eficiente para o operador de colisão dentro do método Lattice-Boltzmann, no entanto essa simplicidade tem um custo: problemas de estabilidade para simulações com baixa viscosidade (altos números de Reynolds) e de acuracidade para altas viscosidades (KRUGER, 2017), fazendo com que seja necessário aumentar a resolução da malha utilizada e por consequência passa-se a requerer maiores recursos computacionais, principalmente quantidade de memória RAM.

Com intuito de evitar o aumento da resolução da malha para simular números de Reynolds maiores, dois modelos de colisão foram desenvolvidos de maneira a aumentar o número de graus de liberdade, como relação a metodologia BGK. Esses métodos são: MRT (*multiple-relaxation-time*) e TRT (*two-relaxation-time*). Uma das diferenças existentes entre esses dois métodos e o BGK dá-se pelo fato deste último considerar

somente uma taxa de relaxação,  $\omega = 1/\tau$ , para todas as populações, enquanto os outros dois introduzem diferentes taxas para cada população.

### 3.2.4. Modelo MRT

A principal diferença entre o MRT e o BGK dá-se pelo fato do primeiro ser realizado no espaço de momento, assim deve-se primeiro mapear as populações  $f_i$  para esse espaço, realizar o passo de colisão e posteriormente retornar as populações para o espaço inicial. De uma maneira simples, no MRT faz-se a relaxação de momentos e não de populações. Esse procedimento é realizado por meio da matriz de transformação  $\mathbf{M}$  para obter os momentos  $m_k$ , mostrados abaixo. Assim é possível afetar os termos: densidade, momento, tensores, etc. individualmente escolhendo-se diferentes taxas de relaxação. Pode-se entender então o BGK como um caso especial do MRT, onde todas as taxas de relaxação são iguais (KRUGER, 2017).

$$m_k = \mathbf{M}f_i \quad m_k = \begin{pmatrix} m_0 \\ \vdots \\ m_{q-1} \end{pmatrix} \quad \mathbf{M} = \begin{pmatrix} M_{0,0} & \cdots & M_{0,q-1} \\ \vdots & \ddots & \vdots \\ M_{q-1,0} & \cdots & M_{q-1,q-1} \end{pmatrix} \quad f = \begin{pmatrix} f_0 \\ \vdots \\ f_{q-1} \end{pmatrix} \quad (3.20)$$

Assim, a equação na forma discretizada para o modelo MRT é dada pela Equação (3.21), mostrada abaixo.

$$f_i(r + c_i \Delta t, t + \Delta t) - f_i(r, t) = -\mathbf{M}^{-1} \mathbf{S} [m(r, t) - m^{eq}(r, t)] \Delta t \quad (3.21)$$

que é equivalente à Equação (3.22),

$$f_i(r + c_i \Delta t, t + \Delta t) - f_i(r, t) = -\mathbf{M}^{-1} \mathbf{S} \mathbf{M} [f(r, t) - f^{eq}(r, t)] \Delta t \quad (3.22)$$

onde  $\mathbf{S}$ , mostrado pela Equação (3.23), é uma matriz diagonal,  $\mathbf{S} = \omega \mathbf{I} = \text{diag}(\omega, \dots, \omega)$  e os momentos de equilíbrio são dados por  $m^{eq} = \mathbf{M}f^{eq}$ .

$$\mathbf{S} = \begin{pmatrix} \omega_0 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \omega_{q-1} \end{pmatrix} \quad (3.23)$$

A forma geral para o algoritmo utilizando o MRT é mostrada abaixo.

1. Computar os momentos ( $\rho \mathbf{u}$ )

2. Realizar a transformação para o espaço de momento
3. Calcular os momentos de equilíbrio
4. Realizar o passo de colisão
5. Retornar para o espaço das populações
6. Realizar a propagação (*streaming*)

A Figura (1) abaixo ajuda a exemplificar os passos acima.

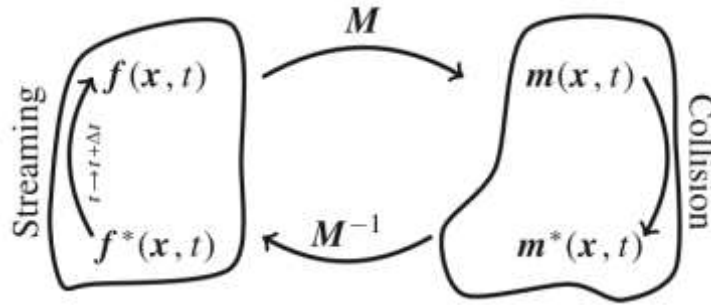


Figura 1 - Esquema da metodologia MRT. (Kruger et al, 2017)

### 3.2.5. Modelo TRT

O modelo TRT combina a simplicidade presente no BGK com a melhoria em acuracidade e estabilidade conseguidas no MRT. A grande vantagem desse modelo em comparação com o MRT consiste na redução do número de variáveis que devem ser inicializadas para realizar a relaxação. No MRT a matriz  $\mathbf{S}$ , não possui todos os seus termos conhecidos, assim, para simulações de fluidos no espaço bidimensional tem-se dois fatores de relaxação para serem inicializados, e no espaço tridimensional tem-se um número muito maior de parâmetros (taxas de relaxação) livres, fazendo com que o processo de estudo para a escolha desses parâmetros sejam quase inviáveis, principalmente para grandes sistemas ou aplicações industriais onde as soluções fluido dinâmicas tem curto prazo para serem alcançadas.

O modelo TRT necessita somente de dois parâmetros de relaxação, o primeiro é derivado da viscosidade dinâmica e o outro é um parâmetro livre. As duas taxas de relaxação empregadas são capazes de resolver as limitações do modelo BGK onde a precisão dos erros depende do parâmetro único de relaxação  $\omega$ . Assim a acuracidade do método depende de uma combinação de duas taxas de relaxação:  $\omega^+$  e  $\omega^-$ .

No TRT o parâmetro  $\Lambda$ , mostrado pela Equação (3.24), é utilizado para caracterizar o erro e as propriedades de estabilidade do método e é função das duas taxas de relaxação mencionadas.

$$\Lambda = \left( \frac{1}{\omega^+ \Delta t} - \frac{1}{2} \right) \left( \frac{1}{\omega^- \Delta t} - \frac{1}{2} \right) \quad (3.24)$$

Dessa maneira, pode-se manter  $\Lambda$  fixo enquanto mudamos a viscosidade através de  $\omega^+$  e contrabalanceamos o efeito dessa mudança com o parâmetro  $\omega^-$ .

É mostrado por Ginzburg (2010) que existem valores específicos de  $\Lambda$  que melhoram a estabilidade, enumerados abaixo.

- $\Lambda = \frac{1}{12} \rightarrow$  resultados ótimos para problemas de advecção pura;
- $\Lambda = \frac{1}{6} \rightarrow$  resultados mais precisos para equação da pura da difusão;
- $\Lambda = \frac{1}{4} \rightarrow$  soluções mais estáveis.

O modelo TRT pode ser implementado de maneira semelhante ao modelo BGK por meio das Equações (3.25) e (3.26), dadas abaixo.

$$f_i^* = f_i - \omega^+ \Delta t (f_i^+ - f_i^{eq}) - \omega^- \Delta t (f_i^- - f_i^{eq-}) \quad (3.25)$$

$$f_i(r + c_i \Delta t, t + \Delta t) = f_i^*(r, t) \quad (3.26)$$

### 3.3. ARRANJOS DE REDE

#### 3.3.1. Modelos Unidimensionais

A terminologia comumente usada no método LB é **DnQm**, onde **n** designa o número dimensões do problema e **m** o número do modelo de velocidade referente a quantidade de vínculos (lattices), logo podemos variar **n** de 1 a 3 e **m** depende do modelo dimensional empregado, mas de maneira geral esse último parâmetro varia entre 2 e 27, podendo chegar à 39 para aplicações soluções em regime transônico, como mostra Nie, Shan e Chen (2009).

De maneira geral três modelos de rede podem ser usados para o caso unidimensional, D1Q2, D1Q3 e D1Q5. O modelo D1Q3 possui três vetores de velocidade para  $f_0, f_1$  e  $f_2$ , assumindo valores iguais à 0, 1 e  $-1$ . Para esse arranjo o número total



de partículas a qualquer instante nunca é maior que três, sendo uma partícula estática na posição central e as outras duas movendo-se para direita ou esquerda durante o processo de propagação (*streaming*). O arranjo de rede D1Q5 é similar ao anterior sendo que o número de partículas não excede cinco em qualquer instante. De maneira análoga o arranjo D1Q2 segue o mesmo padrão exceto pelo fato de não possuir uma partícula central. A Figura (2) ilustra o modelo D1Q3.



Figura 2 - Arranjo D1Q3. (Kruger et al, 2017)

### 3.3.2. Modelos Bidimensionais

Três modelos de rede podem ser utilizados para análise bidimensional, D2Q4, D2Q5 e D2Q9, onde somente o último pode ser aplicado à simulação de fluidos. O modelo D2Q5 possui quatro vetores de velocidade a partir da posição central onde reside uma partícula estática (velocidade zero), cuja notação é  $c(0,0)$ , assim as funções de distribuição  $f_1$  e  $f_2$  movem-se com velocidades  $c(1,0)$  e  $c(-1,0)$  (direção horizontal), enquanto  $f_3$  e  $f_4$  movem-se com velocidades  $c(0,1)$  e  $c(0,-1)$  (direção vertical). O arranjo D2Q4 possui quatro vetores de velocidades e difere do anterior por não possuir uma partícula estática no nó central. O modelo D2Q9 (mais comum) possui um número maior de vetores de velocidades com a existência de uma partícula central com velocidade zero. Assim as velocidades são  $c(0,0)$ ,  $c(1,0)$ ,  $c(0,1)$ ,  $c(-1,0)$ ,  $c(0,-1)$ ,  $c(1,1)$ ,  $c(-1,1)$ ,  $c(-1,-1)$  e  $c(1,-1)$  para  $f_0$ ,  $f_1$ ,  $f_2$ ,  $f_3$ ,  $f_4$ ,  $f_5$ ,  $f_6$ ,  $f_7$  e  $f_8$  respectivamente. A Figura (3) ilustra o modelo D2Q9.

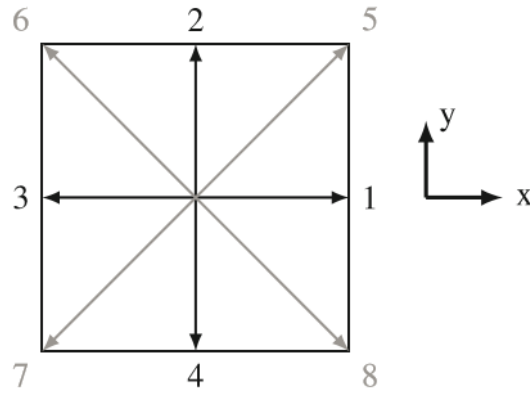


Figura 3 - Arranjo D2Q9 (Kruger et al, 2017)

### 3.3.3. Modelos Tridimensionais

De maneira geral três modelos são utilizados para simulações tridimensionais, são eles D3Q15, D3Q19 e D3Q27 sendo o último utilizado com menor frequência devido ao alto custo computacional associado ao grande número de direções. Todos os modelos anteriores possuem partícula residindo no nó central e os pesos e valores das velocidades podem ser vistos na Tabela (1), mostrada abaixo. A Figura (4) ilustra os três modelos descritos.

Tabela 1 - Pesos e velocidades para os modelos tridimensionais: D3Q15, D3Q19 e D3Q27

Modelo	Velocidades $c_i$	Número	Comprimento $ c_i $	Pesos $w_i$
<b>D3Q15</b>	(0,0,0)	1	0	2/9
	$(\pm 1, 0, 0), (0, \pm 1, 0), (0, 0, \pm 1)$	6	1	1/9
	$(\pm 1, \pm 1, \pm 1)$	8	$\sqrt{3}$	1/72
<b>D3Q19</b>	(0,0,0)	1	0	1/3
	$(\pm 1, 0, 0), (0, \pm 1, 0), (0, 0, \pm 1)$	6	1	1/18
	$(\pm 1, \pm 1, 0), (\pm 1, 0, \pm 1), (0, \pm 1, \pm 1)$	12	$\sqrt{2}$	1/36
<b>D3Q27</b>	(0,0,0)	1	0	8/27
	$(\pm 1, 0, 0), (0, \pm 1, 0), (0, 0, \pm 1)$	6	1	2/27
	$(\pm 1, \pm 1, 0), (\pm 1, 0, \pm 1), (0, \pm 1, \pm 1)$	12	$\sqrt{2}$	1/54
	$(\pm 1, \pm 1, \pm 1)$	8	$\sqrt{3}$	1/216

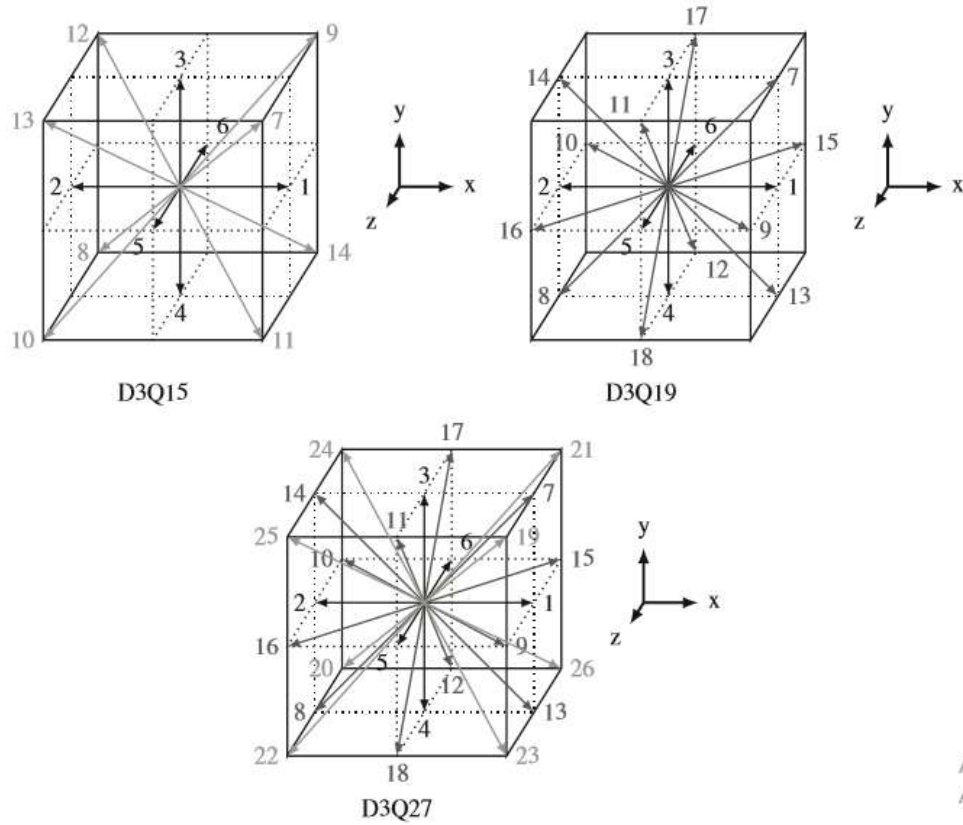


Figura 4 – Arranjos Tridimensionais (Kruger et al, 2017)

### 3.4. SIMULAÇÃO EM FLUIDOS INCOMPRESÍVEIS

#### 3.4.1. Função Distribuição de Equilíbrio

O elemento chave da aplicação do método de Lattice-Boltzmann em diferentes problemas reside na função de distribuição de equilíbrio,  $f_i^{eq}$ . A forma geral da função de distribuição de equilíbrio pode ser vista na Equação (3.27).

$$f_i^{eq} = \Phi \omega_i [A + B c_i \cdot u + C (c_i \cdot u)^2 + D u^2] \quad (3.27)$$

onde  $\mathbf{u}$  é o vetor velocidade do fluido, A, B, C e D são constantes que precisam ser determinadas com base nos princípios de conservação (massa, momento e energia) e  $\Phi$  representa um parâmetro escalar, como densidade, temperatura ou concentração de uma espécie.

Para simulações de fluidos incompressíveis a função distribuição de equilíbrio é escrita como mostra a Equação (3.28).

$$f_k^{eq} = w_k \rho(x, y) \left[ 1 + \frac{c_k \cdot \mathbf{u}}{c_s^2} + \frac{1}{2} \frac{(c_k \cdot \mathbf{u})^2}{c_s^4} - \frac{1}{2} \frac{\mathbf{u}^2}{c_s^2} \right] \quad (3.28)$$

Onde os termos  $c_s$ ,  $c_k$  e  $\mathbf{u}$  são dados pelas Equações (3.29), (3.30) e (3.31).

$$c_s = \frac{c_k}{\sqrt{3}} \quad (3.29)$$

$$c_k = \frac{\Delta x}{\Delta t} i + \frac{\Delta y}{\Delta t} j \quad (3.30)$$

e

$$\mathbf{u} = ui + vj \quad (3.31)$$

No método LB os parâmetros que controlam escoamentos incompressíveis são o número de Reynolds e as restrições geométricas. O número de Mach não é parâmetro de similaridade, e como será visto posteriormente, este pode ser utilizado para acelerar o tempo de simulação por meio da determinação de um passo de tempo maior, exceto quando problemas de estabilidade numérica forem limitantes.

### 3.4.2. Número de Reynolds e Mach

É possível mostrar que o erro associado ao método LB é da ordem de  $Ma^2$ . A viscosidade do fluido pode ser relacionada com a frequência de relaxação para o modelo BGK por meio da Equação (3.32).

$$\nu = \frac{\Delta x^2}{3\Delta t} (\omega - 0.5) \quad (3.32)$$

Para o modelo TRT emprega-se a Equação (3.33).

$$\nu = c_s^2 \left( \frac{1}{\omega + \Delta t} - \frac{1}{2} \right) \quad (3.33)$$

O número de Reynolds é dado pela Equação (3.34), onde  $U$  e  $L$  são a velocidade e o comprimento característico, respectivamente.

$$Re = \frac{UL}{\nu} \quad (3.34)$$

A Equação (3.32) pode ser manipulada, de modo que podemos obter a expressão para o número de Mach, onde notamos que o fator que mais influência esse parâmetro e, portanto, o erro associado à simulação, é a velocidade macroscópica  $U$  definida.

$$Ma = \frac{\Delta x}{L\sqrt{3}}(\omega - 0.5)Re \quad (3.35)$$

Assim, para manter um número de Mach pequeno, Mohamad (2011) recomenda que  $U$  seja da ordem de 0.1 ou 0.2, no entanto, para simulações com números de Reynolds mais elevados utilizando a metodologia BGK, as vezes é necessário manter  $U$  abaixo de 0.05.

O número de Mach e, portanto, a velocidade característica estão diretamente ligados ao tempo de simulação. A unidade utilizada para  $U$  é lattices por passo de tempo. Assim se definimos  $U_1 = 0.2$  e  $U_2 = 0.05$ ,  $U_1$  levará 5 passos de tempo para percorrer a distância entre dois lattices, enquanto  $U_2$  levará 20 passos de tempo. O tempo físico de um passo de tempo pode ser determinado pela Equação (3.36).

$$\Delta t = V_{lattice} \frac{\Delta x}{V_\infty} = Ma \, a_{lattice} \frac{\Delta x}{V_\infty} \quad (3.36)$$

onde  $a_{lattice}$  é a velocidade do som em unidades de lattice. Deve-se sempre procurar adotar números e Mach elevados, mesmo que estes não correspondam as condições reais da simulação, respeitando o limite de  $Ma = 0.3$  e estabelecendo similaridade para a geometria e para o número de Reynolds. Faz-se isso a fim de aumentar o passo de tempo da simulação. Em alguns casos isso não é possível devido a problemas de instabilidade numérica, nesses casos os métodos MRT e TRT se tornam diferenciais no que se refere ao tempo e custo computacional.

### 3.5. CONDIÇÕES DE CONTORNO

Uma das principais diferenças existentes entre o método LB e os métodos fundamentados nas equações de Navier-Stokes reside nas condições de contorno. No método estudado aqui, é necessário determinar equações apropriadas para determinar as

diferentes funções de distribuição nos contornos. A seguir as principais condições de contorno utilizadas nas simulações serão detalhadas.

### 3.5.1. Bounce Back

A condição de contorno denominada bounce back é utilizada para modelar principalmente paredes móveis ou estacionárias. As principais vantagens do método consistem na sua simplicidade e na contribuição para a conservação de massa devido à ausência de mecanismos de perda ou incerteza nos contornos, já que essa abordagem consiste em um simples rebatimento das partículas que chegam à parede.

Três versões para o bounce back são apresentados na literatura: à meia distância da parede, à meia distância dentro da parede e exatamente na parede, sendo este último o mais simples e o utilizado neste trabalho. O bounce back pode ser efetuado igualando (rebatendo) as funções de distribuição  $f_5 = f_7$ ,  $f_2 = f_4$  e  $f_6 = f_8$  na parede.

### 3.5.2. Condições de Contorno com Velocidade Conhecida

Para casos onde temos velocidade conhecida, tais como entradas, as equações de Zhu & He são utilizadas para calcular as três funções de distribuição desconhecidas. Para o caso simulado, onde a entrada é situada à esquerda do domínio, as três funções de distribuição desconhecidas são  $f_1$ ,  $f_5$  e  $f_8$ . As Equações (3.37) (3.38) e (3.39) representam a condição de contorno para as funções de distribuição desconhecidas, utilizando as equações de Zhu & He.

$$f_1 = f_3 + \frac{2}{3}\rho_{esq}u_{esq} \quad (3.37)$$

$$f_5 = f_7 - \frac{1}{2}(f_2 - f_4) + \frac{1}{6}\rho_{esq}u_{esq} + \frac{1}{2}\rho_{esq}v_{esq} \quad (3.38)$$

$$f_8 = f_6 + \frac{1}{2}(f_2 - f_4) + \frac{1}{6}\rho_{esq}u_{esq} - \frac{1}{2}\rho_{esq}v_{esq} \quad (3.39)$$

Onde  $u_{esq}$ ,  $v_{esq}$  e  $\rho_{esq}$  representam a velocidade  $u_x$  na estrada esquerda, velocidade  $u_y$  na entrada esquerda e a densidade adicional na mesma entrada, respectivamente. Esse último parâmetro,  $\rho_{esq}$ , é uma variável adicional criada para que as equações de Zhu & He possam ser resolvidas e pode ser escrito como mostra a Equação (3.40).

$$\rho_{esq} = \frac{1}{1 - u_w} [f_0 + f_2 + f_4 + 2(f_3 + f_6 + f_7)] \quad (3.40)$$

### 3.5.3. Condição de Contorno Aberta

Em vários problemas a velocidade na saída não é conhecida. Como prática normal, utiliza-se extrapolação para as funções de distribuição desconhecidas (MOHAMAD, 2011). No caso da simulação realizada, a velocidade na saída, localizada à direita do domínio é desconhecida, logo, não sabemos as funções  $f_1$ ,  $f_5$  e  $f_8$ . Assim podemos utilizar interpolação de segunda ordem para determinar tais funções, como mostram as Equações (3.41), (3.42) e (3.43).

$$f_{1,n} = 2f_{1,n-1} - f_{1,n-2} \quad (3.41)$$

$$f_{5,n} = 2f_{5,n-1} - f_{5,n-2} \quad (3.42)$$

$$f_{8,n} = 2f_{8,n-1} - f_{8,n-2} \quad (3.43)$$

### 3.5.4. Condição de Contorno Periódica

Condição de contorno periódica pode ser aplicada fazendo com que as funções de distribuição que deixam uma certa linha ou plano sejam reinseridas em outra região do domínio. Se as funções de distribuição  $f_4$ ,  $f_7$  e  $f_8$  são desconhecidas na parede superior do domínio e as funções  $f_2$ ,  $f_5$  e  $f_6$  na parede inferior do mesmo, podemos aplicar a condição de periodicidade fazendo:  $f_{4,superior} = f_{4,inferior}$ ,  $f_{7,superior} = f_{7,inferior}$  e  $f_{8,superior} = f_{8,inferior}$  para a parede superior e  $f_{2,inferior} = f_{2,superior}$ ,  $f_{5,inferior} = f_{5,superior}$  e  $f_{6,inferior} = f_{6,superior}$  para a parede inferior.

## **4. REVISÃO BIBLIOGRÁFICA**

O método LB já é bem difundido quando se trata de simulações em regime laminar. Alguns softwares, como a plataforma de simulação PowerFLOW®, atestam que até um certo limite para o número de Reynolds ( $<10000$ ) é possível realizar simulação direta, abordagem que permite representar todas as escalas de movimento relevantes, algo relativamente difícil para as metodologias empregadas em códigos de CFD convencionais devido a elevada dissipação numérica.

Para altos números de Reynolds, independentemente do método de simulação utilizado, simulação direta se torna impraticável para problemas de média e larga escala. Assim, de maneira similar à códigos convencionais de CFD, também se incorpora modelos de turbulência no método LB para levar em consideração as estruturas turbulentas não resolvidas. Alguns institutos acadêmicos e empresas já desenvolveram códigos capazes de simular tais condições, alguns utilizando a metodologia URANS e, outros a metodologia LES. Na maioria dos casos essa última abordagem é utilizada mais no âmbito acadêmico, visto que os esforços computacionais ainda são elevados para problemas de aplicação prática. Nesta seção, mostra-se alguns resultados já obtidos para os dois tipos de abordagem citados acima.

### **4.1. Modelos LBM-URANS**

Chen (2012) foi capaz de simular um aerofólio NACA 0012 para ângulo de ataque de  $4^\circ$  e Reynolds igual a  $10^5$ . O estudo foi realizado acoplando-se o modelo de turbulência SA ao método LB com tempo de relaxação múltiplo (MRT).

Os resultados para vorticidade e pressão obtidos pelo autor podem ser vistos na Figura (5), mostrada abaixo.



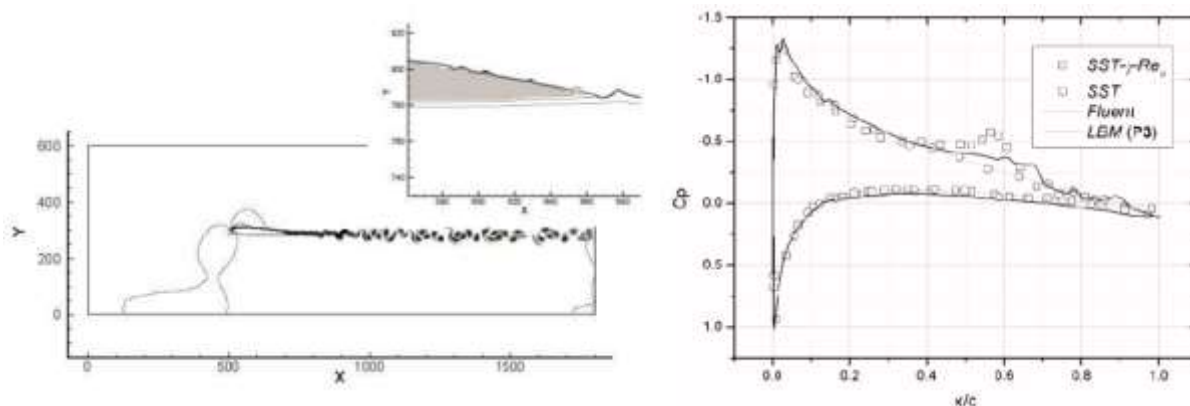


Figura 5 - Vorticidade (esquerda) e coeficientes de pressão para NACA 0012 (Chen, 2012)

Nesse estudo, é possível observar que o método LB foi capaz de prever o descolamento existente na região próxima ao bordo de fuga, notado pela distribuição do coeficiente de pressão e das linhas de corrente mostradas em detalhe, em conformidade com os resultados obtidos com o modelo SST- $\gamma$ -Re por Counsiul e Boulama (2011) que foi coincidente com resultados de simulação direta.

As análises de Counsiul e Boulama (2011) e Rumsey e Spalart (2009) mostraram que tanto o modelo de turbulência SST quanto o SA falham ao tentar prever condições de relaminarização. Observando a distribuição de pressão obtida com o método LB, nota-se pequenas diferenças em  $C_p$  na região de separação. Chen (2012) justifica a discrepância entre o modelo SA (Fluent®) e os resultados do presente método em função do tipo de malha utilizada no LB. Nesse método usam-se estruturas quadradas de mesma resolução tanto na direção da parede-normal quanto na direção das linhas de corrente, enquanto que na maioria das plataformas convencionais para simulação CFD, utilizam-se malhas com elementos com alta razão de estiramento na região da camada limite.

Rajani et al. (2008), utilizando a plataforma de simulação PowerFLOW®, simularam um aerofólio composto por três elementos e compararam os dados obtidos com os resultados experimentais realizados no *Basic Aerodynamics Reserch Center (BART)* na *NASA*. A simulação utilizava como ar fluido a uma velocidade de 56 m/s, o que corresponde a Mach 0.17 e número de Reynolds igual a  $1.7 \cdot 10^6$ . A Figura (6) representa o aerofólio simulado.

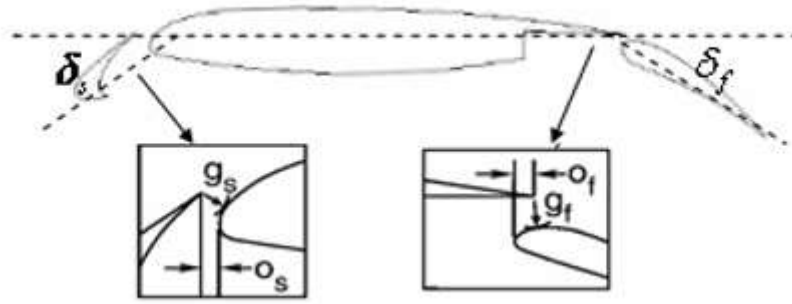


Figura 6 - Esquema do aerofólio simulado (Rajani et al, 2008)

O modelo de 19 direções (D3Q19) e tempo de relaxação simples (BGK) foi utilizado na plataforma PowerFLOW®, juntamente com uma malha cartesiana de alta resolução com a menor distância entre lattices igual a 0.07mm, totalizando 20 milhões de elementos. A Figura (7) mostra detalhes da malha aplicada.

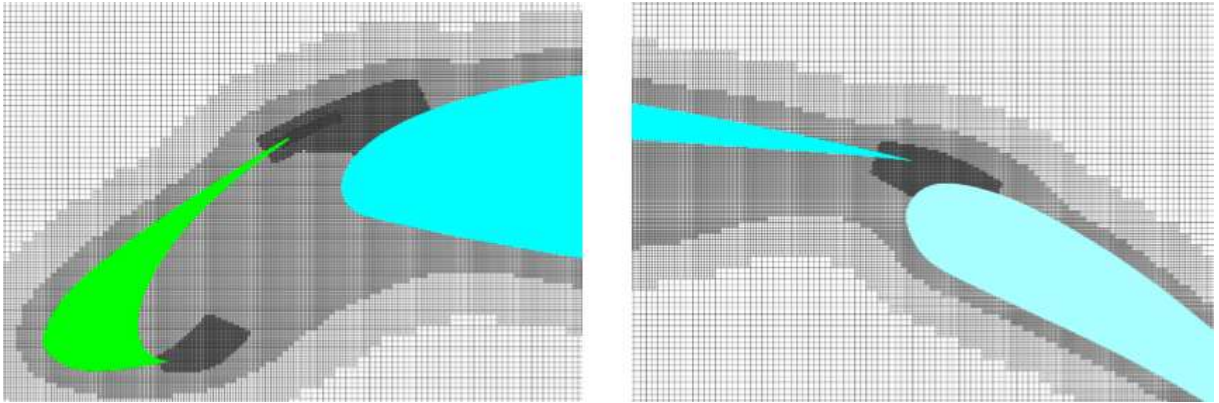


Figura 7 - Malha aplicada no estudo. Slat (esquerda) e Flap (direita)

A função de distribuição de equilíbrio utilizada pelo PowerFLOW® é diferente da apresentada anteriormente e é dada pela Equação (4.1).

$$f_i^{eq} = \rho w_i \left[ 1 + \frac{c_i \cdot u}{T} + \frac{(c_i \cdot u)^2}{2T^2} - \frac{u^2}{2T} + \frac{(c_i \cdot u)^3}{6T^2} - \frac{c_i \cdot u}{2T^2} u^2 \right] \quad (4.1)$$

onde  $w_i$  são os pesos, mostrados anteriormente na Tabela (1) e  $T$  vale 1/3 para simulações isotérmicas.

Para modelar as flutuações causadas pela turbulência, a plataforma de simulação utiliza o modelo  $k-\epsilon$  RG acoplado ao método LB e substituiu-se o tempo de relaxação molecular por um tempo de relaxação efetivo ( $\tau \rightarrow \tau_{eff}$ ).

A Figura (8) mostra a comparação entre resultado da simulação (LBM-VLES) com os dados experimentais e os dados obtidos utilizando RANS-2D para os ângulos de ataque  $4^\circ$ ,  $6^\circ$  e  $8^\circ$ . Apesar da plataforma PowerFLOW® utilizar a metodologia URANS, os autores denominam por VLES (*Very Large Eddy Simulations*) o modelo utilizado.

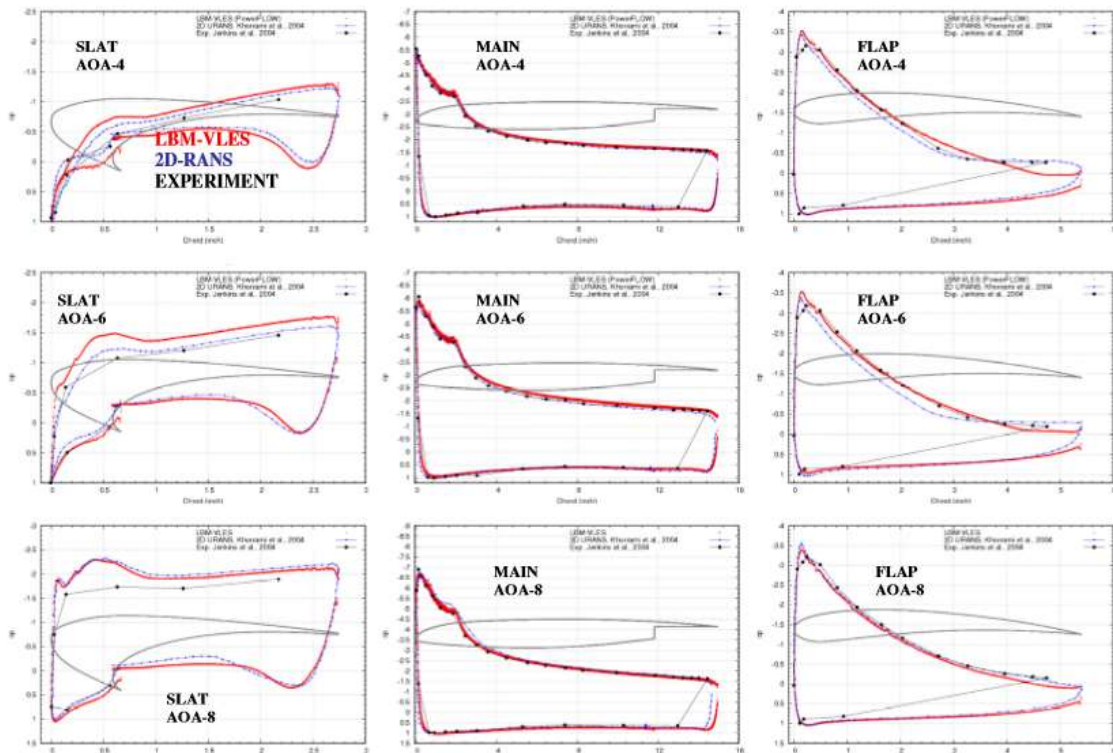


Figura 8 - Comparação entre LBM-VLES, RANS-2D e dados experimentais (Rajani et al., 2008)

Os estudos de Rajani et al. (2008) mostram que os resultados obtidos com LBM-VLES estão em bastante concordância com os resultados experimentais. Em comparação com os resultados obtidos pela metodologia RANS-2D, a maior diferença é notada no elemento que compõe o flap. Na Figura (9) é possível ver a comparação para a vorticidade e para a energia cinética turbulenta para ângulo de ataque igual a  $6^\circ$ .

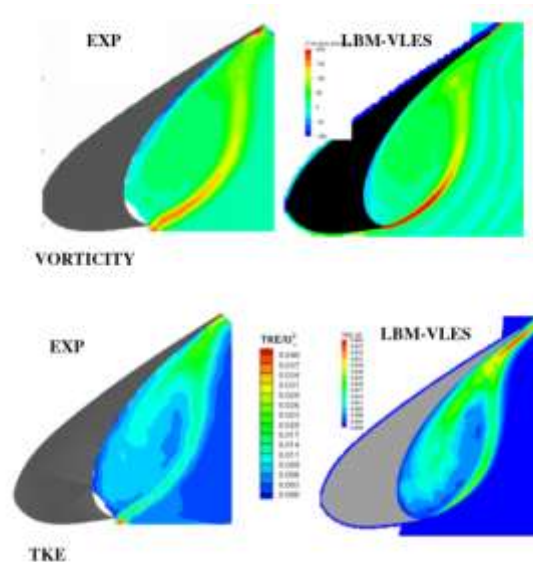


Figura 9 - Comparação entre LBM-VLES e dados experimentais para vorticidade (superior) e energia cinética turbulenta (inferior) (Rajani et al., 2008)

Ehab e Swen (2011) simularam a configuração de alta sustentação utilizada no primeiro *AIAA High-Lift Prediction Workshop*. A abordagem numérica e a técnica para geração de malha foram as mesmas utilizadas por Rajani et al. (2008), assim como o software, PowerFLOW®. A geometria ensaiada no túnel de vento Ames de seção 14'x22' da NASA (Langley) é uma asa trapezoidal composta por três elementos, corda média aerodinâmica de 39.6", alongamento 4.56, é mostrada na Figura (10) abaixo. O Mach e o Reynolds do experimento são 0.2 e  $4.3 \cdot 10^6$ , respectivamente.



Figura 10 - Geometria ensaiada no túnel de vento AMES (NASA Langley) (Ehan e Swen, 2011)

Nesse estudo, uma análise de resolução de malha foi realizada considerando dois tamanhos mínimos de elementos em partes da superfície, 1.0mm (setup 1) e 1.25mm (setup2). O setup 2 foi capaz de reproduzir todos os resultados do setup 1 com menos de 0.25% de diferença com relação aos coeficientes de arrasto e sustentação e com uma redução de aproximadamente 50% no tempo computacional. A Figura (11) mostra um corte ao longo da envergadura para as duas resoluções de malha.

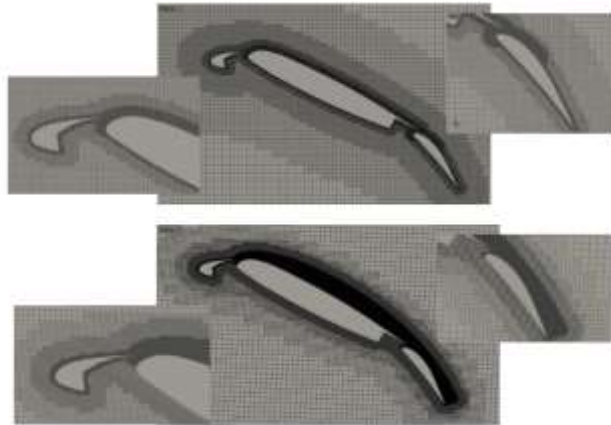


Figura 11 - Corte na malha ao longo da envergadura para o setup 2 (superior) e setup 1 (inferior)

Os resultados obtidos para os coeficientes de arrasto e sustentação podem ser vistos nas Figuras (12) e (13) mostradas abaixo, onde nota-se grande concordância com os dados experimentais.

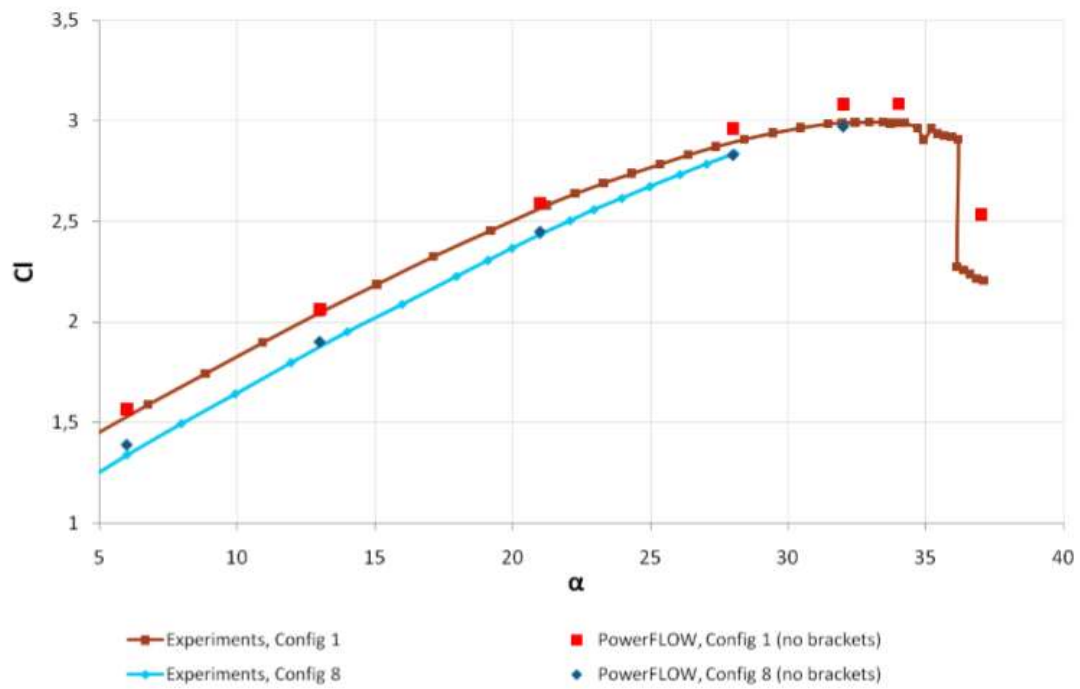


Figura 12 – Comparação para CL entre a simulação (PowerFLOW) e os dados experimentais

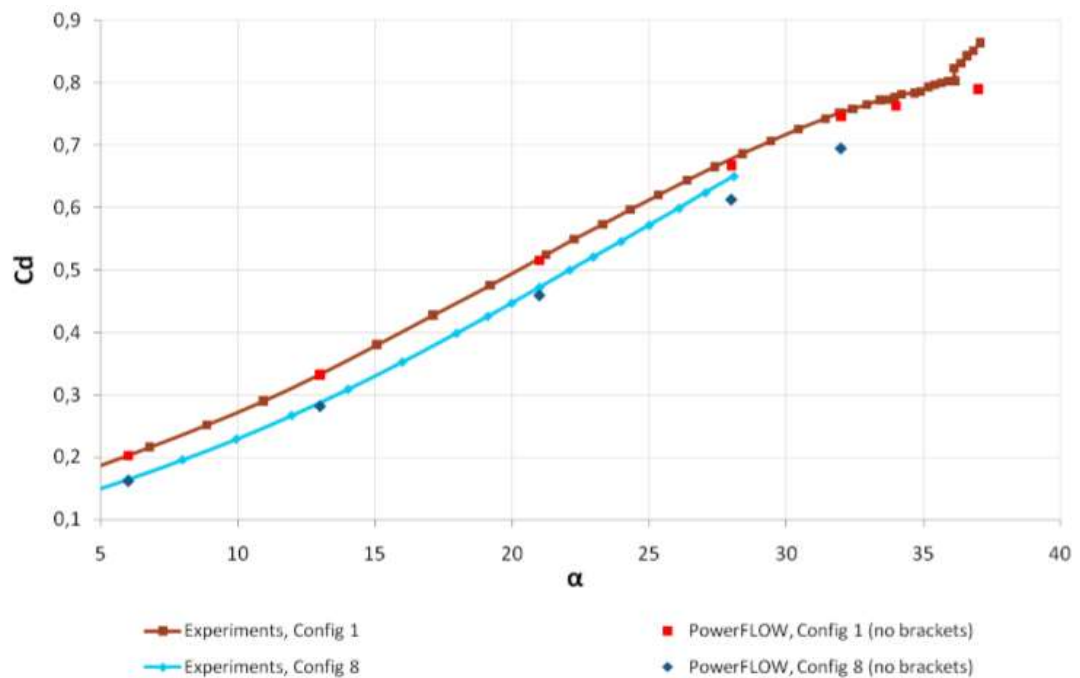


Figura 13 - Comparação para CD entre a simulação (PowerFLOW) e os dados experimentais

Outros estudos foram realizados por Anthony et al. (2008) utilizando a mesma abordagem VLES utilizada na plataforma PowerFLOW®. Esses estudos foram realizados sobre as variantes de conjunto asa-fuselagem DLR-F4 e DLR-F6 utilizadas em três *AIAA Drag Prediction Workshops*. No entanto, apesar do número de Reynolds ter sido mantido igual ao simulado em túnel de vento ( $3 \cdot 10^6$ ) o número de Mach adotado foi 0.2, ao invés de 0.6 e 0.75, já que o método LB ainda é limitado a baixos números de Mach. Os resultados obtidos por Anthony et al. (2008) apresentaram boa concordância com os dados experimentais, no entanto o arrasto foi levemente superestimado (30-40 counts) e a sustentação subestimada. Como apontado pelo autor, uma das possíveis razões foi a ausência de regiões laminares nas simulações, já que foi assumido camada limite totalmente turbulenta.

#### **4.2. Modelos LBM-LES**

Embora simulações LES ainda não sejam práticas para aplicações industriais, muitos no meio acadêmico vem pesquisando as vantagens dessa metodologia quando aplicada ao método LB. Huidan (2004) em sua tese estudou as capacidades da equação do presente método aplicado a escoamentos em regime turbulento envolvendo simulação direta (LBM-DNS) e simulação de grandes escalas (LBM-LES).

Huidan (2004) foi capaz de concluir, a partir das comparações realizadas entre LBM-LES e NS-LES, que o primeiro caso preserva as estruturas do fluido com maior acuracidade. Em uma segunda análise, para jatos de mistura utilizando o modelo D3Q19 e LBM-LES (MRT) incorporado ao modelo de submalha de Smagorinsky, o autor obteve boa concordância com dados experimentais e concluiu que, diferente dos valores tradicionais para a constante de Smagorinsky ( $C_{sm} = 0.17$ ) utilizados na abordagem NS-LES, valores de  $C_{sm} = 0.10$  permitem obter melhores resultados para LBM-LES.

Krause e Nathen (2013), usando a metodologia LBM-LES (MRT) por meio do código OpenLB estudaram o escoamento sobre cilindro quadrado aplicando o esquema de interpolação proposto por Bouzidi (2002). Os resultados para regime laminar tiveram boa concordância com os obtidos por Breuer et al. (2000) utilizando método de volumes finitos e são consideravelmente melhores quando comparados com o método LB padrão (BGK, bounce-back simples), como pode ser visto na Figura (14) mostra abaixo.



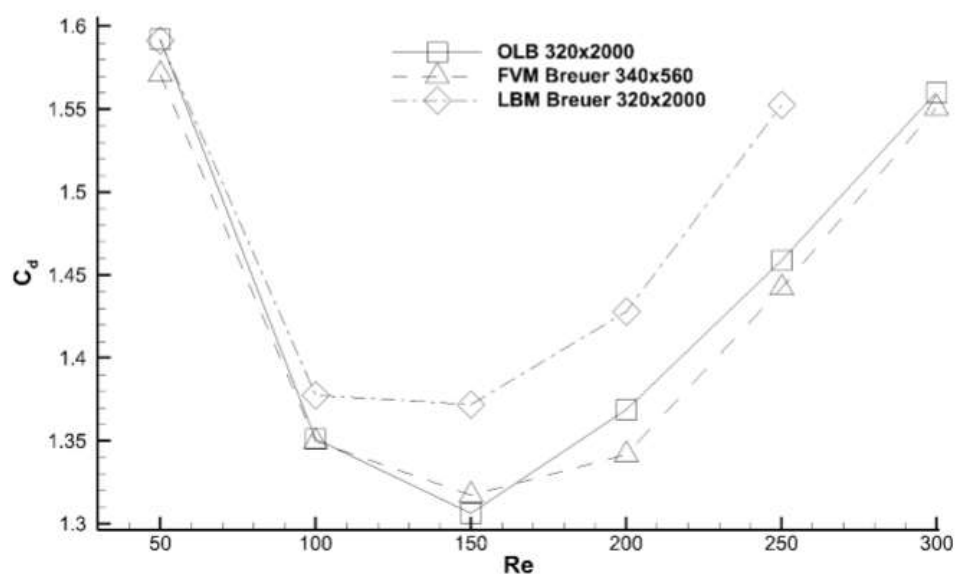


Figura 14 - Coeficiente de arrasto para um cilindro quadrado em função do número de Reynolds

Krause e Nathen (2013) estenderam a simulação para Reynolds 21400 e compararam os resultados para coeficiente de arrasto e número de Strouhal com Wienken (2003) e Bouris (1999) além de dados experimentais obtidos por Lyn e Rodi (1994). A Tabela (2) mostra os resultados para os dois parâmetros observados, onde observa-se que para altos números de Reynolds o código OpenLB é incapaz de prever com exatidão o coeficiente e arrasto.

Tabela 2 - Coeficiente de arrasto e número de Strouhal para diferentes autores

Caso	CD	Strouhal
OpenLB (N=100)	1.472	0.124
OpenLB (N=200)	1.590	0.129
Experimento	2.1	0.132
Wienken	2.28	0.128
Bouris	2.28	0.134

Apesar do coeficiente de arrasto não ter sido previsto de maneira suficiente para afirmar que há concordância com os resultados experimentais de Lyn e Rodi (1994), os autores compararam o perfil de velocidade ao longo da linha central com os perfis obtidos experimentalmente e utilizando RANS e RSE, onde reconhece-se que o perfil obtido utilizando o código OpenLB é o que apresenta melhor concordância com o resultado experimental, como pode ser visto na Figura (15).



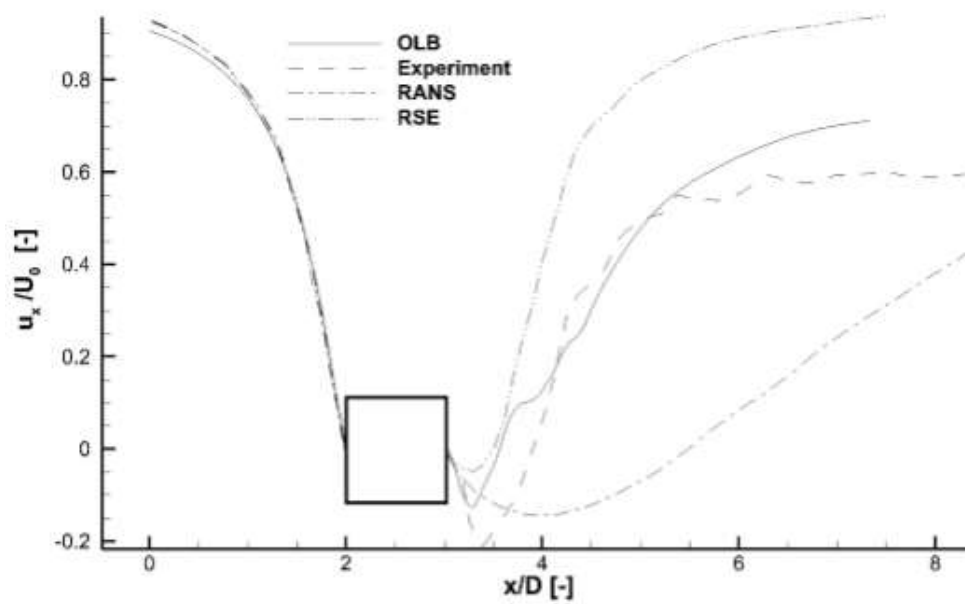


Figura 15 - Comparação do perfil de velocidade de três métodos numéricos com resultado experimental

Segundo os autores, os resultados mostrados acima podem indicar uma possível deficiência na metodologia empregada para o cálculo do arrasto, *momentum-exchange*.

## 5. SIMULAÇÃO DE ESCOAMENTO SOBRE CILINDROS

As simulações de escoamentos sobre cilindros foram realizadas com objetivo de comparar os resultados obtidos pelo código desenvolvido na Universidade Federal de Uberlândia com os resultados fornecidos pelo código *open source* OpenLB e com o método de volumes finitos.

### 5.1. OpenLB

OpenLB é uma plataforma de simulação numérica para fluido dinâmica computacional criada por estudantes e pesquisadores baseada no método Lattice-Boltzmann. O código foi desenvolvido em linguagem C++ por meio de estruturas orientadas com a intenção de que o mesmo sirva para diferentes propósitos e para que programadores implementem diferentes geometrias da maneira mais autônoma possível.

O código possui vários modelos já implementados, desde a metodologia BGK até a metodologia MRT com o modelo de Smagorinsky para simulações LES. As estruturas D2Q9 e D3Q19 estão disponíveis no código em bibliotecas pré-compiladas. Outros arranjos como o D3Q13 também estão presentes, no entanto este requer um tipo específico de dinâmica e os demais arranjos podem ser implementados pelo programador.

O código também possui a maioria das condições de contorno já implementadas, tanto para paredes retas como curvas. Um diferencial do OpenLB com relação a outros códigos é a da condição de contorno de Bouzidi (2002) presente para paredes curvas, onde um esquema de interpolação é realizado próximo a parede do corpo, de forma a representar melhor a geometria a ser simulada.

A maneira geral de funcionamento do código pode ser dividida em sete fases, mostradas abaixo e que se mantêm em todas as aplicações com o OpenLB.

1. Inicialização: Conversão entre as unidades físicas e as utilizadas no método LB. Definição do tipo de estrutura a ser utilizada (DnQm) e local de armazenamento.
2. Preparação da Geometria: Definição da geometria por meio de arquivos .STL ou por meio de funções. Geração e configuração da malha computacional por meio da função `prepareGeometry`.

3. Preparação dos lattices: Configuração da dinâmica dos lattices, dependente do tipo de tempo de relaxação utilizado (BGK ou MRT). Definição do tipo de fluido (compressível ou incompressível).
4. Loop principal: Inicialização do contador e do ciclo principal (colisão e propagação).
5. Definição das condições de contorno iniciais: Inicialização da função `setBoundaryValues` dentro do loop principal.
6. Função `collideAndStream`: A cada iteração essa função é chamada para realizar os passos de colisão e propagação.
7. Cálculo das saídas do programa e resultados: A cada iteração a função `getResults` é chamada pra saída dos resultados (arquivos `.vti` ou `.ppm`) em determinados passos de tempo especificados.

## **5.2. Código Lattice-Boltzmann Desenvolvido**

O código utilizando o método Lattice-Boltzmann foi inicialmente desenvolvido no Laboratório de Mecânica dos Fluidos da Universidade Federal de Uberlândia durante uma iniciação científica (BIBIC-CNPq) em 2015. O código foi construído em linguagem Fortran 90 pelo autor do trabalho foi utilizado para simulação em cavidades, canais e cilindros a números de Reynolds abaixo de 40. O mesmo foi baseado no material de Mohamad (2011).

A estrutura do código é semelhante à descrita na seção de fundamentação teórica. Foi utilizado a metodologia BGK e nenhuma função de interpolação foi utilizada para paredes. A Figura (16) mostra a comparação das linhas de corrente obtidas com o código em Fortran 90 ao simular uma cavidade para número de Reynolds igual a 1000 com os resultados obtidos por Ghia (1982).

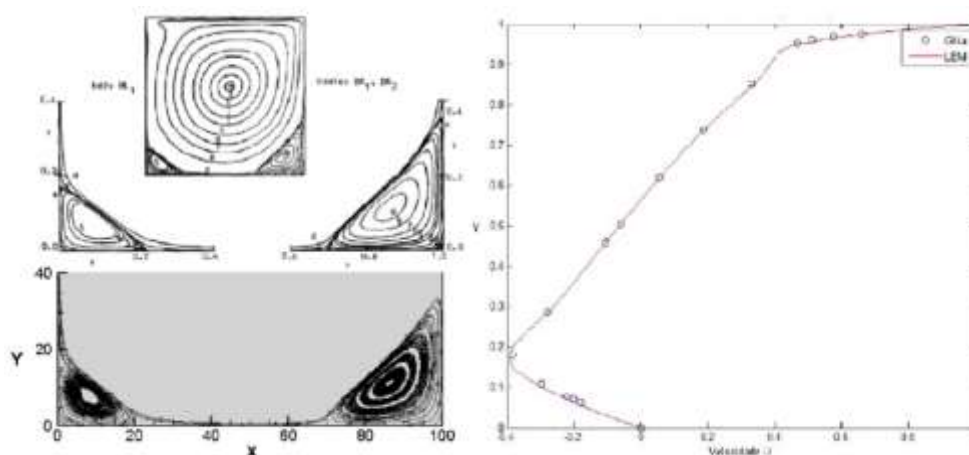


Figura 16 - Comparação das linhas e corrente do perfil de velocidade com Ghia (1982).

### 5.3. Simulação Sobre Cilindros

Com a intenção de comparar os resultados obtidos pelo código desenvolvido em Fortran 90 com outras fontes que utilizam o mesmo método de solução e com o método de volumes finitos, uma simulação para número de Reynolds igual a 200 foi realizada sobre cilindro utilizando o software Fluent®, o código OpenLB e o código em Fortran 90.

#### 5.3.1. Parâmetros da Simulação

A geometria simulada consiste em um cilindro com diâmetro ( $D$ ) igual a 0.1m posicionado a 5D da entrada em um domínio de dimensões  $10D \times 20D$ .

A simulação foi realizada para número de Reynolds igual a 200. O tamanho físico de cada lattice na simulação é de  $2.5 \cdot 10^{-3}m$ . Assim o domínio possui tamanho  $400 \times 800$  e o cilindro possui diâmetro de 40 unidades (lattices). O valor da viscosidade adotada no método foi 0.02, visto que esse é acima do valor de segurança necessário para evitar problemas de instabilidade quando se utiliza a metodologia BGK (0.01) e possibilitará o estudo da influência de malhas mais grossas ainda utilizando valores de viscosidade maiores ou iguais a 0.01. A velocidade  $u_x$  calculada foi 0.1, fazendo com que o número de Mach da simulação, calculado em 0.1732, esteja dentro do regime incompressível com uma velocidade  $u_x$  não tão baixa, a fim de reduzir o tamanho do passo de tempo. A Figura (17) mostra o esquema da geometria e do domínio empregados na simulação.

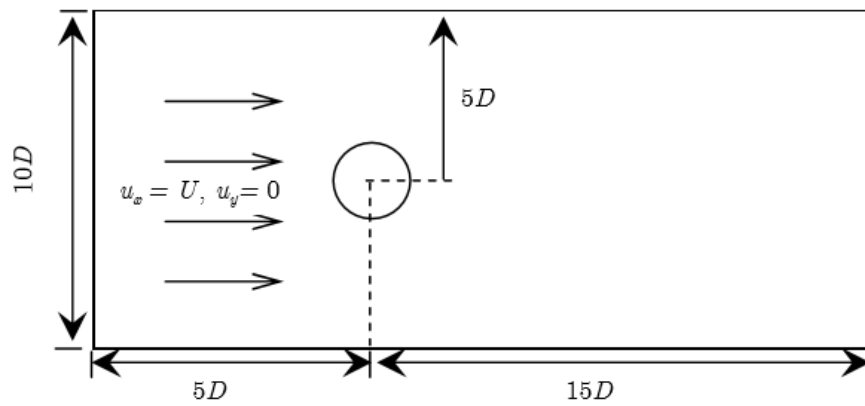


Figura 17 - Representação da geometria e do domínio empregados na simulação

Na simulação realizada, alguns pontos devem ser observados. Como não existe nenhum modelo de turbulência implementado o Reynolds para o canal não deve exceder o valor indicado para transição ( $Re \sim 2500$ ).

Como visto na seção de fundamentação teórica, o método LB não necessita obedecer à similaridade para número de Mach. Supondo que o fluido em questão fosse ar (condição ISA) a velocidade na entrada e o número de Mach seriam respectivamente  $3.5788 \cdot 10^{-2}$  m/s e  $1.056 \cdot 10^{-4}$ , sendo o último valor muito abaixo do empregado na simulação numérica.

Caso queira-se aumentar o tamanho do passo de tempo, para reduzir o tempo de simulação, pode-se aumentar o Mach simulado para 0.25, alterando a velocidade  $u_x$  para 0.1443. No entanto deve-se lembrar que o erro associado ao método é da ordem de  $Ma^2$  e, valores muito próximos ao limite de Mach igual a 0.3-0.4 devem ser evitados.

A malha gerada no software Ansys ICEM®, a fim de manter as características do da malha do método em estudo, possui o mesmo tamanho de elemento em todo o domínio. A região que contém o cilindro pode ser vista em detalhe na Figura (18).

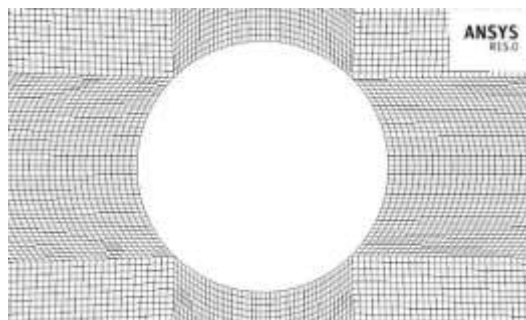


Figura 18 - Detalhe da malha para a simulação utilizando volumes finitos

### 5.3.2. Condições de Contorno

Para evitar os problemas de transição devido ao diâmetro do canal onde o cilindro será simulado aplicamos condição de periodicidade nas paredes superior e inferior do domínio. Dessa maneira, perde-se o efeito de transição associado ao diâmetro do canal além de eliminar-se o efeito de interferência da parede sobre o escoamento, capaz de alterar significativamente o número de Strouhal, dado pela Equação (5.1) mostrada abaixo.

$$St = \frac{fL}{U} \quad (5.1)$$

Onde  $f$  é frequência de emissão de vórtices,  $L$  é o comprimento característico e  $U$  é a velocidade. As condições de Zhu e He foram utilizadas como condições de contorno para a entrada, já que temos velocidade imposta. Na saída do domínio utilizamos extrapolação para calcular as funções de distribuição desconhecidas por meio de interpolações (segunda ordem).

Para conservação de massa e momento aplicamos o esquema de bounce-back exatamente na parede do cilindro, devido à simplicidade de implementação.

Nas simulações utilizando o método de volumes finitos por meio do software Fluent® condição de simetria, velocidade de entrada, condição de outflow e parede foram aplicadas nas paredes superiores e inferiores, entrada, saída e parede do cilindro, respectivamente. Regime laminar transiente foi definido e utilizou-se inicialização híbrida. O esquema SIMPLE foi selecionado para solução do acoplamento pressão velocidade visto que problemas de convergência para a simulação não são esperados devido à simplicidade do caso. Como deve-se simular um determinado tempo para

comparação dos resultados, a escolha de outro esquema (SIMPLEC, PISO, Coupled) só acarretaria em esforço computacional desnecessário e maior tempo de simulação.

### 5.3.3. Resultados Obtidos

Todas as simulações realizadas foram computadas para um tempo de simulação igual a 100 segundos, exceto para o caso utilizando volumes finitos, onde simulou-se 200 segundos. Os parâmetros de comparação aqui tratados são o coeficiente de arrasto e o número de Strouhal.

Inicialmente, analisou-se o efeito da inserção da condição de contorno de Bouzidi (2002) sobre o coeficiente de arrasto em comparação com o bounce-back simples no código OpenLB. Diferente do esperado, a condição de contorno de Bouzidi não ajudou a representar significativamente a condição de parede curva no método LB e isso pode ser observado pela baixa diferença no coeficiente de arrasto e leve distanciamento de resultados experimentais, onde os valores de coeficiente de arrasto para a condição simulada ficam em torno de 1.0 e 1.1. A metodologia utilizada para cálculo do coeficiente de arrasto foi o *momentum-exchange*. A Figura (19) mostra os resultados obtidos para comparação feita, onde a média do coeficiente de arrasto por meio do bounce-back simples foi 3.1365 e com a condição de Bouzidi foi 3.1681.

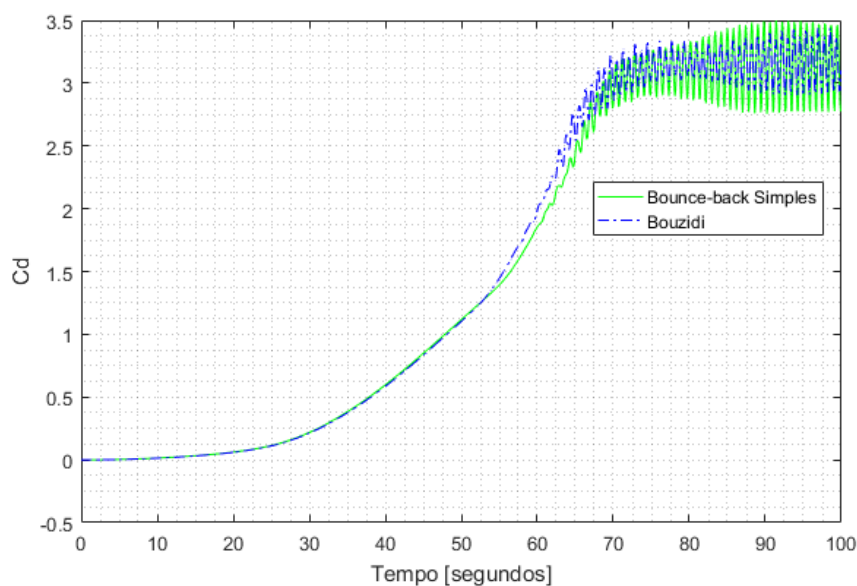


Figura 19 - Coeficiente de arrasto utilizando LB com a condição de Bouzidi e bounce-back simples

A comparação entre o código OpenLB e o código desenvolvido em Fortran 90 permite concluir que ambos fornecem valores de coeficiente de arrasto bem acima dos obtidos em ensaios experimentais, no entanto os valores obtidos pelo código desenvolvido ainda são levemente acima dos fornecidos pelo OpenLB. Enquanto o coeficiente de arrasto fornecido pelo OpenLB foi de 3.1365 o obtido com o código em Fortran foi 3.5157, valor 12% acima do informado anteriormente. A figura (20) abaixo mostra a comparação entre os coeficientes de arrasto para os dois códigos.

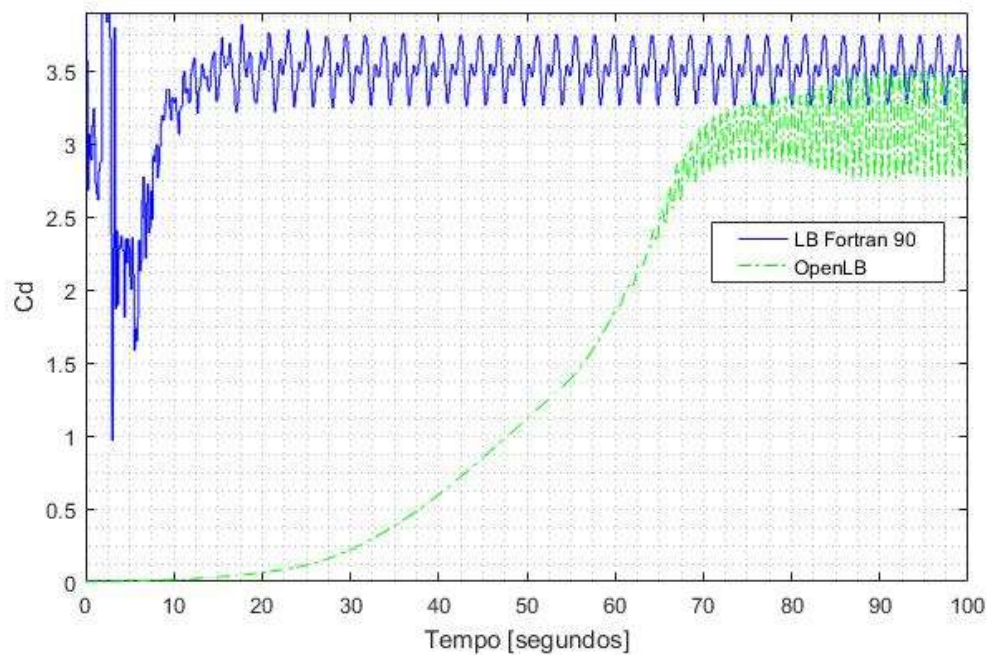


Figura 20 - Coeficiente de arrasto com o código em Fortran 90 e OpenLB

Com relação a frequência de emissão de vórtices, os dois códigos apresentaram resultados distantes. O número de Strouhal obtido com o OpenLB foi 3.511 e com o código em Fortran, 1.426. Os dois valores obtidos são bem acima do conseguido por Zheng e Zhang (2008), 0.202, o que representa uma diferença de 706% para o código desenvolvido e de 1738% para o OpenLB. A Figura (21) mostra o campo de velocidade e as isolinhas de velocidade  $U$  instantes após a geração dos vórtices, utilizando o código desenvolvido.



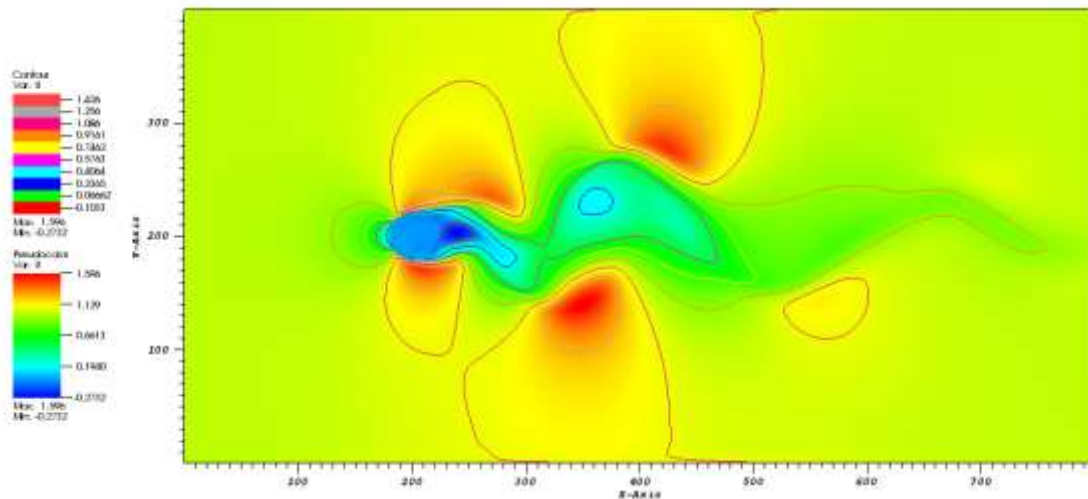


Figura 21 - Campo de velocidade e isolinhas de velocidade U

Embora o código desenvolvido não tenha apresentado valores para coeficientes de arrasto melhores que o OpenLB, alguns parâmetros da simulação se sobressaíram com relação ao OpenLB. A conservação de massa no código desenvolvido foi mantida a valores de  $10^{-8}$  enquanto o código OpenLB fornece apenas  $10^{-6}$ .

Outro ponto importante diz respeito ao significado físico da curva do coeficiente de arrasto mostrada pelos os dois códigos. Sabe-se que antes do escoamento se estabilizar, o cilindro presencia todos os números de Reynolds abaixo de 200, no caso da simulação realizada. Isto acontece em função de estarmos trabalhando no regime subsônico, onde as partículas a frente do perfil de velocidade imposto na entrada sentem o escoamento chegando e se pré-organizam, especialmente na simulação em questão, onde esse tempo de pré-organização é elevado, da ordem de 10 segundos.

Assim, espera-se que o comportamento da curva mostrada na Figura (20) respeite a tendência dos valores para coeficiente de arrasto para os diferentes Reynolds presenciados pelo cilindro até que ocorra a geração de vórtices. He e Doolen (1997), Nieuwstadt e Keller (1993) e Guo e Shu (2013) obtiveram os seguintes valores para números de Reynolds iguais a 10, 20 e 40, mostrados na Tabela (3) abaixo.

Tabela 3 - Coeficientes de arrasto para Reynolds entre 10 e 40

Reynolds	Coeficiente de Arrasto		
	10	20	40
<b>He &amp; Doolen</b>	3.170	2.152	1.499
<b>Nieuwstadt &amp; Keller</b>	2.828	2.053	1.550
<b>Guo &amp; Shu</b>	2.942	2.155	1.604

Observa-se que o comportamento da curva do coeficiente de arrasto obtido pelo código em Fortran 90 segue a tendência dos valores mostrados acima, enquanto que o código OpenLB fornece valores que iniciam em zero e tem tendência crescente, oposto ao mostrado nos resultados da tabela.

Por fim, nota-se que no OpenLB as instabilidades só aparecem a aproximadamente 65 segundos de simulação, o que corresponde, aproximadamente, a um tempo de residência, enquanto que no código desenvolvido esse fenômeno ocorre segundos após o perfil de velocidade imposto na entrada ter atingido o cilindro, um comportamento que pode indicar alguma fonte de instabilidade no código em Fortran90.

Os resultados obtidos pela simulação utilizando volumes finitos, por meio do software Ansys Fluent®, apresentam a mesma tendência esperada para os números de Reynolds sentidos pelo cilindro abaixo do simulado, no entanto o decaimento dos valores para o coeficiente de arrasto ocorre muito mais rápido que o observado utilizando o código desenvolvido. O coeficiente de arrasto para essa faixa obtido pelo método de volumes finitos pode ser visto na Figura (22) mostrada abaixo.

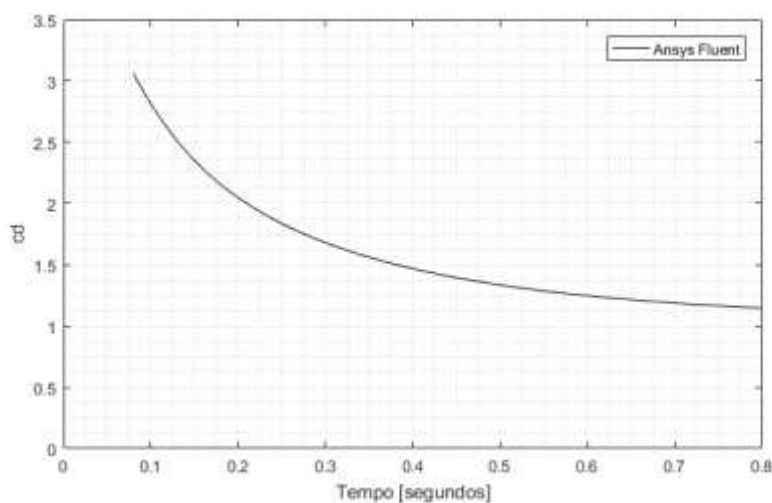


Figura 22 - Tendência do coeficiente de arrasto no início da simulação para o método de volumes finitos

A comparação entre os resultados obtidos pelo Fluent® com os obtidos pelo método LB podem ser vistos na Figura (23) mostrada abaixo. Nota-se que o método de volumes finitos foi superior ao método LB, pois o mesmo é capaz de prever com certa precisão os valores de coeficiente de arrasto em comparação com resultados experimentais.

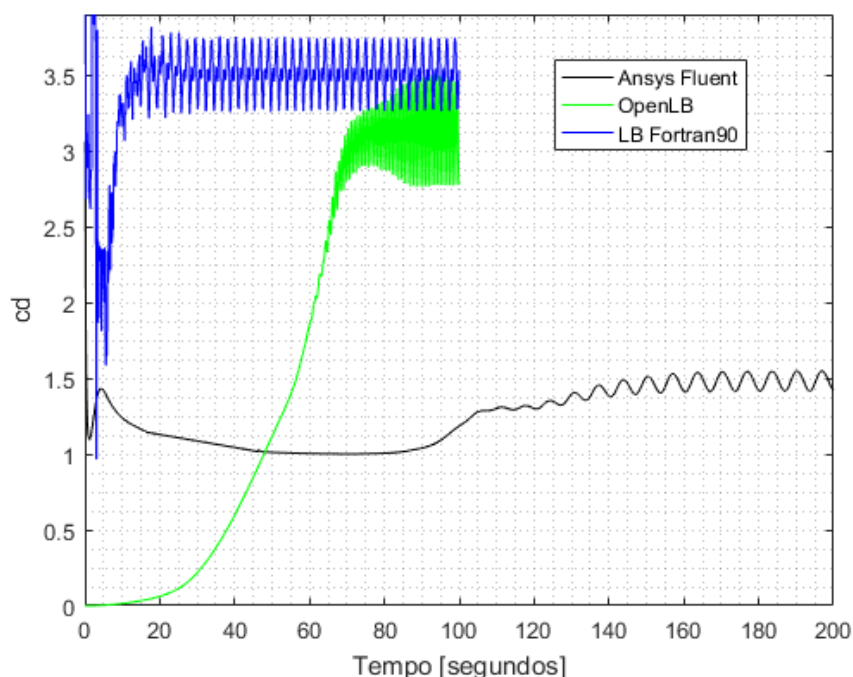


Figura 23 - Comparação dos três métodos de solução para coeficiente de arrasto

A Tabela (4) sumariza os resultados obtidos pelos dois métodos e mostra os resultados obtidos por Guo e Shu (2013) utilizando o método LB além dos resultados utilizando volumes finitos e os dados experimentais mostrados por Çengel (2015), todos para Reynolds 200.

Tabela 4 - Comparação dos resultados

	<b>Cd</b>	<b>Strouhal</b>
<b>LB Fortan 90</b>	3.5157	1.426
<b>OpenLB</b>	3.1365	3.511
<b>OpenLB Bouzidi</b>	3.1681	3.511
<b>Guo &amp; Shu</b>	1.4350	0.202
<b>Ansys Fluent</b>	1.4847	0.419
<b>Experimental</b>	1.0 – 1.1	-----

Embora o valor obtido para o coeficiente de arrasto utilizando o método de volumes finitos tenha sido bom em comparação com resultados experimentais, os resultados aqui obtidos, mostrados nas três primeiras linhas da Tabela 4, não podem ser interpretados como uma deficiência do método LB. Guo & Shu (2013) também foram capazes de obter resultado semelhante ao método de volumes finitos utilizando o método LB e mostrando melhor concordância para o número de Strouhal com relação a estudos

realizados por outros autores. Outros estudos conduzidos utilizando o método LB já mostraram que é possível obter bons resultados para os coeficientes aerodinâmicos. Além disso, com relação ao tempo computacional, o método se mostra muito superior a abordagem utilizando volumes finitos. A Tabela 5 mostra a comparação entre os três programas utilizados nesse trabalho para diferentes números de núcleos de processamento ao simular o escoamento sobre cilindro circular por 10 segundos. Em todas as simulações empregou-se o mesmo passo de tempo para todos os métodos e tentou-se manter o mesmo tamanho de elemento em todas as malhas geradas.

Tabela 5 - Tempo de simulação

<b>Núcleos</b>	<b>Tempo (segundos)</b>		
	OpenLB	Fortran90	Fluent®
<b>1</b>	127.3	986	102600
<b>2</b>	89.3	- - -	55800
<b>4</b>	74.7	- - -	46800
<b>8</b>	78.8	- - -	42600

## 6. INFLUÊNCIA DO TAMANHO DA MALHA E DO NÚMERO DE MACH

Com objetivo de analisar a interferência do tamanho da malha (resolução do cilindro) e do número de Mach empregado sobre os resultados para coeficiente de arrasto e número de Strouhal, mais duas simulações foram conduzidas para Reynolds igual a 200 utilizando OpenLB. A Tabela 6 mostra os detalhes de cada uma das simulações.

Tabela 6 - Parâmetros das simulações realizadas

<b><i>Simulação</i></b>	<b><i>Malha</i></b>	<b><i><math>u_x</math></i></b>	<b><i>Mach</i></b>	<b><i><math>Ma^2</math></i></b>	<b><i>Viscosidade<sub>LBM</sub></i></b>
<b>Caso 1</b>	200x400	0.1	0.1732	0.0300	0.01
<b>Caso 2</b>	400x800	0.1	0.1732	0.0300	0.02
<b>Caso 3</b>	400x800	0.05	0.0866	0.0075	0.01

Os resultados obtidos com as três simulações mostraram considerável queda do coeficiente de arrasto com o aumento da resolução da malha e redução da velocidade característica. A amplitude na oscilação desse parâmetro foi altamente alterada com o refinamento das simulações, como pode ser visto na Figura 24. O número de Strouhal se manteve inalterado para os três casos.

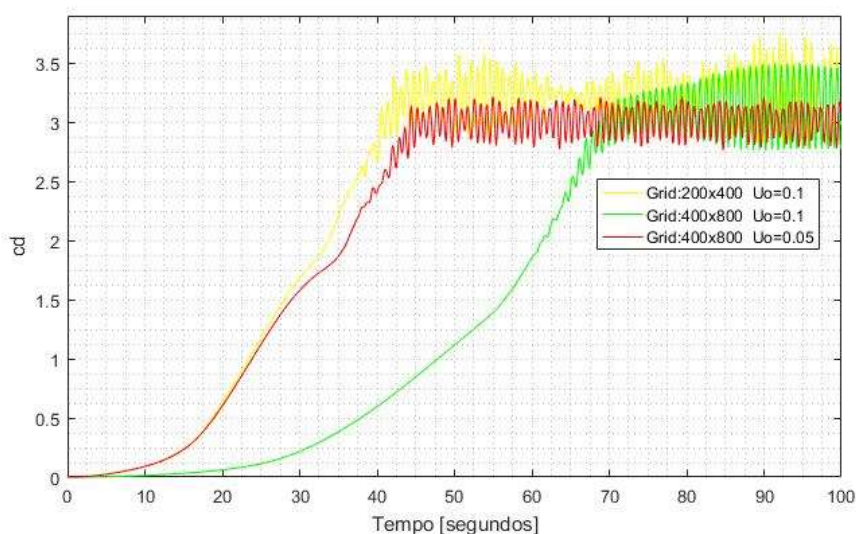


Figura 24 - Coeficiente de arrasto para as três simulações

A Tabela 7 mostra os resultados para coeficiente de arrasto e número de Strouhal para as três simulações realizadas.

Tabela 7 - Resultados para diferentes parâmetros

<b>Simulação</b>	<b>Cd</b>	<b>Strouhal</b>
<b>Caso 1</b>	3.2280	3.511
<b>Caso 2</b>	3.1395	3.511
<b>Caso 3</b>	3.0119	3.511

## 7. CONCLUSÃO

O método Lattice-Boltzmann vem se destacando no cenário de desenvolvimento automotivo e aeronáutico, principalmente no que diz respeito simulações onde soluções em regime permanente não são efetivas. O método ainda é utilizado para solução de problemas em outras áreas, como aeroacústica.

No presente trabalho propôs-se a investigação do método LB quanto a capacidade de prever, principalmente, o coeficiente de arrasto sobre cilindros para números de Reynolds intermediários. Assim, foram simulados escoamentos sobre cilindros circulares.

As análises foram realizadas utilizando-se três abordagens. Inicialmente usando o código *open source* OpenLB que utiliza o método Lattice-Boltzmann como solução para escoamentos, em seguida utilizando o código desenvolvido em Fortran 90 por meio da mesma abordagem e por fim, utilizando o método de volumes finitos por meio do software comercial Fluent®.

Embora alguns resultados, obtidos por outros autores, mostrem que o método é capaz de prever o coeficiente de arrasto com um erro em torno ou menor que 40%, os resultados obtidos nas simulações realizadas mostram que tanto o código OpenLB quanto o código desenvolvido pelo autor não são capazes de prever os coeficientes de arrasto com a mesma faixa de erro mencionada. Os resultados obtidos fornecem coeficientes de arrasto três vezes maiores que resultados experimentais, enquanto que a frequência de emissão de vórtices, utilizando o método LB, manteve certa concordância somente para estudos conduzidos por outros autores. Tanto o código desenvolvido, quanto o OpenLB forneceram valores bem acima do esperado para esse parâmetro.

O alto erro no cálculo do coeficiente de arrasto pode ser atribuído ao método *momentum-exchange*, principalmente após a geração dos vórtices, já que tanto em trabalhos anteriores desenvolvidos pelo autor, utilizando o mesmo método, sobre escoamentos em cavidades, canais e cilindros ( $Re < 40$ ) quanto por Krause e Nathen (2013) observa-se que o método LB prever com exatidão as linhas de corrente e perfis de velocidade ao longo do domínio para o escoamento.

Embora os resultados mencionados acima não tenham sido promissores, alguns softwares que utilizam o método LB, como PowerFLOW®, mostram alguns resultados para coeficiente de arrasto, sustentação e momento com grande concordância quando comparados à dados experimentais.

Como esperado, a simplicidade do método ficou clara ao compararmos o tempo de simulação necessário para simularmos o escoamento sobre cilindros utilizando o método de volumes finitos, por meio do software Ansys Fluent®, e o tempo necessário para simularmos utilizando o método LB, mais de 600 vezes menor para quatro núcleos de processamento e malhas com mesmo tamanho de elementos. O conjunto de equações diferenciais parciais que a metodologia de volumes finitos necessita resolver faz com que soluções transientes sejam extremamente dispendiosas utilizando esse método de solução, no entanto é importante lembrar que o método LB, mesmo quando se utiliza as metodologias MRT e TRT, necessita de malhas muito mais finas quando comparado ao método de volumes finitos, isso pode ser constatado pela considerável variação obtida no coeficiente de arrasto ao se duplicar o tamanho da malha. Uma malha consideravelmente mais grossa poderia ter sido utilizada para o caso simulado no Fluent®, fazendo com que a diferença de custo computacional entre os dois métodos fosse reduzida. No entanto, mesmo que uma malha mais grossa fosse utilizada, pode-se afirmar que o tempo computacional do método LB ainda seria significativamente menor.



## 8. REFERÊNCIAS BIBLIOGRÁFICAS

FERZIGER J. H., PERIC M. **Computational Methods for Fluid Dynamics**. Ed. Springer, 2003.

SUCCI, S. **The Lattice-Boltzmann Equation for Fluid Dynamics**. Oxford University Press, 2001.

WOLF-GLADROW D. A. **Lattice-Gas Cellular Automata and Lattice-Boltzmann Models**. Ed. Springer, 2005

CHAPMAN S., COWLING T. G. **The Mathematical Theory of Nonuniform Gases**. Cambridge University Press, 1970.

MOHAMAD A. A. **Lattice-Boltzmann Method – Fundamentals and Engineering Applications with Computer Codes**. Springer, 2011.

KRUGER T., KUSUMAATMAJA H., KUZMIN A., SHARDT O., SILVA C., VIGGEN E. M., **The Lattice-Boltzmann Method – Principles and Practice**. Ed. Spriger, 2017.

GINZBURG I., d'HUMIERES D., KUZMIN A., STAT J., **Phys**, 1090, 2005.

NIE X., SHAN X., CHEN H., **Lattice-Boltzmann / Finite-Difference Hybrid Simulation of Transonic Flow**. 47th AIAA Aerospace Science Meeting, Florida, 2009.

CHEN X. **Applications of Lattice-Boltzmann Method to Turbulent Flow Around Two-Dimensional Airfoil**. Engineering Applications of Computation Fluid Mechanics, Vol. 6, No. 4, pp. 572-580, 2012.

COUNSIL J. N. N., BOULAMA K. G., **Validating the URANS Shear Stress Transport  $\gamma$ -Re Model for Low-Reynolds-Number External Aerodynamics**. Int. J. Num. Methods Fluids, 69, pp. 1411-1432, 2012.

RUMSEY C. L., SPALART P. R. **Turbulence Model Behavior in Low Reynolds Number Regions of Aerodynamics Flow Fields**. AIAA J. 47(4), pp. 982-993, 2009.

RAJANI S., YANBING L., RICHARD S. NOELTING S. **Simulation of Flow Over a 3-Element Airfoil Using a Lattice-Boltzmann Method**. 46th AIAA Aerospace Science Meeting, Nevada, 2008.

EHAB F., SWEN N. **Unsteady Flow Simulation of a High-Lift Configuration Using a Lattice-Boltzamnn Approach**. 49th AIAA Aerospace Science Meeting, Orlando, 2011.

ANTHONY K., JEREMY B., RICHARD S., **Lattice-Boltzmann Simulations of the DLR-F4, DLR-F6 and Variants**. 46th AIAA Aerospace Science Meeting, Nevada, 2008.

HUIDAN Y. **Lattice-Boltzmann Equation Simulations of Turbulence, Mixing, and Combustion**. PhD Thesis, Texas A&M University, 2004.

KRAUSE M. J., NATHEN P. **An Extension of the Lattice-Boltzmann Method for Simulating Turbulent Flows Around Rotating Geometries of Arbitrary Shape**. Conference Paper, 2013.

BOUZIDI M., FIRDAOUS M., LALLEMAND P. **Momentum Transfer of a Boltzmann-Lattice Fluid with Boundaries**, ICASE Report. No. 2001-22, 2002.

BREUER M., BERNSDORF J., ZEISER T., DURST F. **Accurate Computations of the laminar Flow Past a Square Cylinder Based on Two Different Methods: Lattice-Boltzmann and Finite-Volume**. International Journal of Heat and Fluid Flow 21(2): 186-196, 2000.

WIENKEN, W., **Die Large Eddy Simulations Mittels der Finite-Element-Methode zur Bestimmung des Kavitationsbeginns**. PhD Thesis, Technische Universität Dresden, Dresden, Germany, 2003.

BOURIS D., BERGELES G. **2D LES of Vortex Shedding from a Square Cylinder**. J. of Wind Eng. and Indust. Aerodynamics 80: pp. 31-46, 1999.

LYN D. A., RODI W. **The Flapping Shear Layer Formed by Flow Separation from the Forward Corner of a Square Cylinder**, J. Fluid Mech. 267, pp. 353-376, 1994.

GHIA U., GHIA N., SHIN C. T. **High-Re Solutions for Incompressible Flow Using Navier Stokes Equations and Multigrid Method**. University of Cincinnati, 1982.

ZHENG Z. C., ZHANG N. **Frequency Effects on Lift and Drag for Flow Past an oscillating Cylinder**. J. Fluids Struct. 24, pp. 527-541, pp. 382-399, 2008.

HE X., DOOLEN G. **Lattice-Boltzmann Method on Curvilinear Coordinates System: Flow Around a Circular Cylinder**. J. Comput. Phys, 134, pp. 306-315, 1997.

NIEUWSTADT F., KELLER H. B. **Viscous Flow Past Circular Cylinders**. J. Fluid Mech, 1, pp. 59-71, 1973.

GUO Z., SHU C. **Lattice-Boltzmann Methods and its Applications in Engineering**. Ed. World Scientific, 2013.

ÇENGEL. Y, CIMBALA. J. **Mecânica dos Fluidos – Fundamentos e Aplicações**. Ed. McGraw-Hill, 3<sup>a</sup> ed, 2015.

## **9. APÊNDICE**

## APÊNDICE A - Código Para Reconhecimento de Fronteira de Cilindros

```
%CODIGO PARA RECONHECIMENTO DE FRONTEIRA DE CILINDROS
%AUTOR: PEDRO PAULO DE C. BRITO

clear all
close all
clc

%Centro e Raio da circunferencia
xo = 0;
yo = 0;
raio = 20;

%X e Y de deslocamento em relação a origem do domínio para xo e yo
xdes = 200;
ydes = 200;

%Gerando coordenadas para a equacao
x = xo-raio:0.01:xo+raio;
y = ((raio*raio-(x-xo).^2).^0.5)+yo;

%Plotando equação da circunferencia
figure(1)
plot(x,y)
hold on
plot(x,-y)

%Plotando pontos internos ao raio
for i = -raio:raio
    for j = -raio:raio
        if ((i-xo)^2+(j-yo)^2)<=((raio)^2)
            plot(i,j,'ok')
        end
    end
end

sum = 0;
cont= 0;
%Plotando pontos da fronteira
for i = -raio:raio
    for j = -raio:raio

        if (((-1+i-xo)^2+(j-yo)^2)<=((raio)^2) && ((i-xo)^2+(j-
yo)^2)<=...
            ((raio)^2))%<--
            sum = sum+1;
        end

        if (((+1+i-xo)^2+(j-yo)^2)<=((raio)^2) && ((i-xo)^2+(j-
yo)^2)<=...
            ((raio)^2))%-->
            sum = sum+1;
        end
    end
end
```

```

        if(((i-xo)^2+(+1+j-yo)^2)<=((raio)^2)&&((i-xo)^2+(j-
yo)^2)<=...
            ((raio)^2))%Up
            sum = sum+1;
        end

        if(((i-xo)^2+(-1+j-yo)^2)<=((raio)^2)&&((i-xo)^2+(j-
yo)^2)<=...
            ((raio)^2))%Down
            sum = sum+1;
        end

%-----
if((sum==1||sum==2)&&(i<xo && j>yo)) %Primeiro quadrante
    plot(i+1,j,'og')
    cont = cont+1;
    v(cont,1)=i+1;
    v(cont,2)=j;
    plot(i,j-1,'*g')
    cont = cont+1;
    v(cont,1)=i;
    v(cont,2)=j-1;
end
if((sum==1||sum==2)&&(i<xo && j<yo)) %Quarto quadrante
    plot(i+1,j,'og')
    cont = cont+1;
    v(cont,1)=i+1;
    v(cont,2)=j;
    plot(i,j+1,'*g')
    cont = cont+1;
    v(cont,1)=i;
    v(cont,2)=j+1;
end

if((sum==1||sum==2)&&(i>xo && j>yo)) %Segundo quadrante
    plot(i-1,j,'og')
    cont = cont+1;
    v(cont,1)=i-1;
    v(cont,2)=j;
    plot(i,j-1,'*g')
    cont = cont+1;
    v(cont,1)=i;
    v(cont,2)=j-1;
end

if((sum==1||sum==2)&&(i>xo && j<yo)) %Terceiro quadrante
    plot(i-1,j,'og')
    cont = cont+1;
    v(cont,1)=i-1;
    v(cont,2)=j;
    plot(i,j+1,'*g')
    cont = cont+1;
    v(cont,1)=i;
    v(cont,2)=j+1;
end

if(sum==3||sum==1||sum==2)
    plot(i,j,'or',i,j,'.r')
    cont = cont+1;
    v(cont,1)=i;

```

```

        v(cont,2)=j;
    end

    if(sum==1&& i<xo)
        plot(i+1,j,'or')
        cont = cont+1;
        v(cont,1)=i+1;
        v(cont,2)=j;
    end

    if(sum==1&& i>xo)
        plot(i-1,j,'or')
        cont = cont+1;
        v(cont,1)=i-1;
        v(cont,2)=j;
    end

    if(sum==1&& j<yo)
        plot(i,j+1,'or')
        cont = cont+1;
        v(cont,1)=i;
        v(cont,2)=j+1;
    end

    if(sum==1&& j>yo)
        plot(i,j-1,'or')
        cont = cont+1;
        v(cont,1)=i;
        v(cont,2)=j-1;
    end
    sum = 0;
end
end

%Plotando pontos selecionados, mas ainda sobrepostos
xx = v(1:end,1);
yy = v(1:end,2);
%figure(2)

[linha c] = size(v);

%Eliminando coordenadas sobrepostas
n=1;
for k = 1:linha-1
    n = n+1;
    c1 = v(k,1);
    c2 = v(k,2);
    for r = n:linha
        if(v(r,1)==c1 && v(r,2)==c2)
            v(r,1)=3*raio;
            v(r,2)=3*raio;
        end
    end
end
end

pos = 0;
for k = 1:linha
    if(v(k,1)~=3*raio && v(k,2)~=3*raio)
        pos = pos+1;
        pontos(pos,1)=v(k,1);
    end
end

```

```
        pontos(pos,2)=v(k,2);  
    end  
end  
  
figure(3)  
x = pontos(1:end,1)+xdes;  
y = pontos(1:end,2)+ydes;  
  
plot(x,y,'or')
```