

PROJETO DE GRADUAÇÃO

**Simulação de escoamentos em aerofólios
bidimensionais de dupla cunha e em asas
sob regime supersônico**

Pedro Henrique Rosa dos Santos

Brasília, novembro de 2018

UNIVERSIDADE DE BRASÍLIA

FACULDADE DE TECNOLOGIA

UNIVERSIDADE DE BRASÍLIA
Faculdade de Tecnologia

PROJETO DE GRADUAÇÃO

**Simulação de escoamentos em aerofólios
bidimensionais de dupla cunha e em asas
sob regime supersônico**

Pedro Henrique Rosa dos Santos

*Relatório submetido ao Departamento de Engenharia
Mecânica como requisito parcial para obtenção
do grau de Engenheiro Mecânico*

Banca Examinadora

Prof. Bráulio Gutierrez Pimenta, ENM/UnB _____
Orientador

Prof. Roberto Francisco Bobenrieth Miserda, _____
ENM/UnB
Examinador interno

Prof. Adriano Possebon Rosa, ENM/UnB _____
Examinador externo

RESUMO

A dinâmica dos fluidos computacional está em desenvolvimento acelerado pela era digital, e ela abre possibilidades de desenvolvimento de aeronaves supersônicas com novos patamares de eficiência e de baixa emissão de ruído. Embora as alternativas *OpenSources* não entreguem ainda a mesma performance como as plataformas comerciais, os *softwares* livres possuem a capacidade de superar os comerciais na razão de performance e custo.

Diante disso, o presente trabalho se propôs a descrever o desempenho do *OpenFOAM* no *cluster* do CAALab no que se refere à precisão e custo computacional em escoamentos supersônicos externos, comparando-se com os resultados do *software* VAT, desenvolvido pelo próprio laboratório. Foi feita a compilação do OpenFOAM de forma otimizada para a arquitetura do *cluster*. Para o caso bidimensional, foi desenvolvido uma rotina de cálculo do escoamento sobre o aerofólio dupla cunha e comparou-se com o resultado da simulação, com e sem malha adaptativa. Já no caso tridimensional, utilizou-se o resultado apresentado na literatura, obtido em túnel de vento a partir de asas cujos perfis são de dupla cunha. Quanto à análise do custo computacional, foi realizada uma análise de escalabilidade do OpenFOAM, variando-se a quantidade de subdivisões do domínio e o número de elementos.

Obteve-se precisão de 0,94% no caso sem malha adaptativa e de 0,10% no caso de malha adaptativa na simulação bidimensional do OpenFOAM, enquanto o resultado do VAT obteve precisão de 1,08%. Em ambas as simulações, o ângulo do choque está de acordo com a teoria. Na análise de escalabilidade, o OpenFOAM não apresentou bom desempenho comparado ao VAT. O resultado sugere também que uma razão adequada para o presente caso no OpenFOAM seria de 80 mil elementos por processador. Por fim, no caso tridimensional, a simulação no OpenFOAM se tornou pouco viável, uma vez que o tempo por iteração estava na ordem de 30 segundos no melhor caso. Notou-se que a estratégia de controle automático de resolução da malha acarreta um prejuízo significativo no tempo de processamento, e uma estratégia adequada seria de aplicar o refinamento automático após atingir-se a convergência com a malha menos densa. Contudo, o resultado seguiu a tendência apresentada na literatura.

Palavras-chave: OpenFOAM, Computação de Alto Desempenho, Dinâmica dos Fluidos Computacional, Escalabilidade Computacional.

ABSTRACT

The Computational Fluid Dynamics has been in fast progress recently due to the outset of digital transformation, and it brings possibilities for the development of supersonic aircraft with high efficiency and low noise emission. Although the OpenSource alternatives do not deliver yet the same performance as commercial platforms, the free softwares have the potential to overcome the commercials in the performance and cost ratio.

In this scenario, the present work has been proposed to describe the performance of the OpenFOAM in the Caalab cluster regarding to the accuracy and computational cost in external supersonic flows, as compared with the results of the VAT, an in-house cfd code. OpenFOAM was compiled optimally for the architecture of the cluster. For the two-dimensional case, a routine was developed to calculate the flow on the double wedge Aeropholium and compared with the simulation result with and without adaptive mesh. The results presented in the literature, obtained in wind tunnel from wings whose profiles are double wedge, were used to validate the three-dimensional case. Regarding to the analysis of the computational cost, an OPENFOAM scalability analysis was performed by varying the amount of subdivisions of the domain and the number of elements.

OpenFOAM's simulations obtained accuracy of 0.94 % in the case without adaptive mesh and 0.10 % in the case of adaptive mesh in the two-dimensional case, while the result of the VAT obtained accuracy of 1.08 %. In both simulations, the angle of shock is in line with the theory. In the scalability analysis, OpenFOAM did not perform well, if compared to VAT. The result also suggests that an appropriate ratio for the present case in OpenFOAM would be 80000 elements per processor. Finally, in the three-dimensional case, the simulation in OpenFOAM became almost unviable, since the time per iteration was in the order of 30 seconds in the best case. It was noted that the strategy of automatic control of the mesh resolution entails significant damage in the processing time, and an appropriate strategy would be to apply the automatic refinement after achieving convergence with the less dense mesh. However, the result followed the trend presented in the literature.

Keywords: OpenFOAM, High Performance Computing, Computational Scalability, Computational Fluid Dynamics.

SUMÁRIO

1	INTRODUÇÃO	1
1.1	CONTEXTUALIZAÇÃO	1
1.2	DEFINIÇÃO DO PROBLEMA	2
1.3	REVISÃO BIBLIOGRÁFICA	3
1.4	OBJETIVOS	7
1.5	APRESENTAÇÃO DO TRABALHO	8
2	FUNDAMENTAÇÃO TEÓRICA	9
2.1	DINÂMICA DOS FLUIDOS COMPUTACIONAL	9
2.1.1	MÉTODO DOS VOLUMES FINITOS	9
2.1.2	ESTABILIDADE NUMÉRICA	9
2.2	REGIMES DE ESCOAMENTO	10
2.3	EQUAÇÕES GOVERNANTES E CONDIÇÕES DE CONTORNO	11
2.4	FORÇAS SOBRE UM AEROFÓLIO	15
3	METODOLOGIA	17
3.1	CÁLCULO ANALÍTICO DO ESCOAMENTO	17
3.1.1	EQUAÇÕES DE ONDAS DE CHOQUE OBLÍQUAS	17
3.1.2	EQUAÇÕES DE LEQUE DE EXPANSÃO	19
3.1.3	DADOS DE ENTRADA	20
3.1.4	CÁLCULO DA ONDA DE CHOQUE	21
3.1.5	CÁLCULO DO LEQUE DE EXPANSÃO	21
3.1.6	DADOS DE SAÍDA	21
3.2	METODOLOGIA NUMÉRICA DO VAT	22
3.2.1	EQUAÇÕES GOVERNANTES NO VAT	22
3.2.2	ESQUEMA NUMÉRICO	24
3.2.3	DISSIPACÃO ARTIFICIAL	26
3.2.4	FRONTEIRA ÍMERSA	27
3.2.5	MARCHA TEMPORAL	28
3.3	METODOLOGIA NUMÉRICA DO OPENFOAM	28
3.3.1	EQUAÇÕES GOVERNANTES	29
3.3.2	ESQUEMA NUMÉRICO	29
3.3.3	MÉTODO DE SOLUÇÃO DO SISTEMA DE EQUAÇÕES LINEAR	32

3.3.4	ALGORITMO DE SOLUÇÃO DO SISTEMA DE EQUAÇÕES.....	32
3.4	CONFIGURAÇÃO DO <i>cluster</i> DO CAALAB.....	34
3.5	CAMPANHA DE SIMULAÇÕES	35
3.5.1	AEROFÓLIO E ASA ESTUDADOS	35
3.5.2	MALHA COMPUTACIONAL DO OPENFOAM.....	35
3.5.3	MALHA COMPUTACIONAL DO VAT	40
3.5.4	PARÂMETROS DE SIMULAÇÃO NO OPENFOAM.....	40
3.5.5	PARÂMETROS DE SIMULAÇÃO NO VAT.....	42
4	RESULTADOS E DISCUSSÕES	43
4.1	RESULTADOS DO CÁLCULO NO MATLAB ®.....	43
4.2	RESULTADO DA SIMULAÇÃO DO AEROFÓLIO BIDIMENSIONAL	44
4.2.1	SIMULAÇÃO NO OPENFOAM SEM MALHA ADAPTATIVA	44
4.2.2	SIMULAÇÃO NO OPENFOAM COM MALHA ADAPTATIVA	56
4.2.3	SIMULAÇÃO NO VAT	62
4.2.4	COMPARAÇÃO ENTRE VAT E OPENFOAM	66
4.3	SIMULAÇÃO DA ASA 3D NO OPENFOAM	67
5	CONCLUSÃO	70
	REFERÊNCIAS BIBLIOGRÁFICAS	72
	ANEXOS.....	75
I	ROTINA DE CÁLCULO DO ESCOAMENTO SOBRE O AEROFÓLIO DE DUPLA CUNHA	76

LISTA DE FIGURAS

1.1	Aeronaves capazes de voar acima da velocidade de som.	1
1.2	Crescimento da complexidade das simulações de dinâmica dos fluidos computacional	2
1.3	Projeto de aeronave supersônica <i>low-boom</i>	3
1.4	Variação de pressão em a) aerofólio dupla cunha b) aerofólio biconvexo c) placa plana quando submetidos a escoamento supersônico	4
1.5	Resultados da validação do OpenFOAM	4
1.6	Evolução do erro numérico com o número de volumes para uma malha homogênea e uma malha adaptativa	5
1.7	a) tempo de simulação e b) escalabilidade do algoritmo <i>snappyHexMesh</i> de geração de malha do OpenFOAM	6
1.8	escalabilidade do algoritmo <i>simpleFoam</i> de solução da simulação no OpenFOAM	7
2.1	Processo de simulação computacional	10
2.2	Ilustração de diferentes regimes de voo	12
2.3	Desenho esquemático de um volume de controle	13
2.4	Desenho esquemático de um elemento infinitesimal sob a ação de forças de contato advindas da pressão e de efeitos viscosos, todas na direção x	13
2.5	Esquema de um aerofólio	16
3.1	Esquema de um choque normal adiabático unidimensional.....	18
3.2	Escoamento supersônico passando por uma quina côncava e gerando um choque oblíquo, colado à superfície	18
3.3	Escoamento supersônico passando por uma quina côncava e gerando um choque destacado da superfície	19
3.4	Decomposição das velocidades e determinação do volume de controle pelas linhas pontilhadas	20
3.5	Gráfico da relação $\theta - \beta - Ma$. Para cada número de Mach existe um ângulo máximo de inclinação da superfície. Acima desse ângulo, o choque se torna destacado	21
3.6	Representação do leque de expansão	22
3.7	Escoamento supersônico sobre o aerofólio de dupla cunha em diferentes ângulos de ataque. a) choque no extra e intradorso, b) choque apenas no intradorso, c) choque no intradorso e leque de expansão no extradorso	22

3.8	Volume de controle do OpenFOAM. Na figura, p representa o volume de controle considerado, N representa o centro de um volume de controle vizinho, d é a distância entre os dois, \mathbf{S}_f é o vetor da área da interface entre os dois volumes e \mathbf{d}_{fn} é a distância entre o centro da área da interface e o centro do volume vizinho	29
3.9	Aerofólio de dupla cunha	35
3.10	Vista projetada da asa submetida à simulação numérica	36
3.11	Malha estática bidimensional A do aerofólio de dupla cunha	37
3.12	Malha estática bidimensional B do aerofólio de dupla cunha	38
3.13	Malha estática bidimensional C do aerofólio de dupla cunha	39
3.14	Malha com a)1024 b)2048 e c)3072 elementos por comprimento característico.....	41
3.15	Regiões da malha do OpenFOAM para aplicação das condições de contorno.....	42
4.1	Resultados obtidos por rotina de cálculo do MATLAB®e comparados com os resultados da referência na literatura	44
4.2	Resultados para a malha C (2,5 milhões de elementos).....	45
4.3	Resultados para a malha B (1 milhão de elementos)	46
4.4	Resultados para a malha A (500 mil elementos).....	47
4.5	Coefficiente de arrasto em função do número de volumes, OpenFOAM	48
4.6	Escoamento simulado no OpenFoam com a malha computacional C. A imagem mostra a pressão em escala de cinza no fundo e as linhas de corrente em escala colorida. O ângulo do choque oblíquo simulado coincide com valor calculado.....	49
4.7	Pressão obtida pela simulação e pela teoria de choque e expansão.	49
4.8	Tempo de execução da simulação com 256 subdivisões em função do número de volumes. É apresentado uma regressão polinomial de terceiro grau para verificar a ordem de aumento do tempo de execução.....	50
4.9	Resultados da escalabilidade do OpenFOAM em função do número de volumes. Quanto maior a quantidade de volumes, maior a eficiência e maior o <i>Speed Up</i>	52
4.10	Gráfico de barras do tempo total de simulação por iteração (dt clock/iter) e do tempo de execução por iteração (dt exec/iter) em função do número de subdivisões para diferentes números de volumes.....	54
4.11	Escoamento nas proximidades do truncamento na malha do OpenFOAM.....	55
4.12	Malha resultante da simulação com refinamento automático nas regiões de alto gradiente de velocidade.....	56
4.13	Malha resultante da simulação com possibilidade de aumentar e diminuir o refinamento em função do gradiente de velocidade	57
4.14	Malha resultante da simulação com refinamento automático nas regiões de alto gradiente e com ângulo de ataque de 5 graus.....	57
4.15	Resultados da malha adaptativa com apenas refinamento	59
4.16	Resultados da malha adaptativa com possibilidade de aumento e diminuição do refinamento	60
4.17	Resultados da malha adaptativa com apenas refinamento e com ângulo de ataque 5 graus	61

4.18	Coeficiente de arrasto em função do número de volumes, VAT	62
4.19	Escoamento simulado no VAT. A imagem mostra a variável $\beta_t = \sqrt[10]{\nabla T}$ em escala de cinza no fundo. O ângulo do choque oblíquo simulado coincide com valor calculado	64
4.20	Distribuição de pressão sobre o aerofólio na simulação no VAT	65
4.21	Coeficiente de arrasto, sustentação e momento para a simulação no VAT	65
4.22	Tempo total de simulação por número de volumes e por iteração para o VAT e para o OpenFOAM	66
4.23	Vista do escoamento com linhas de corrente sobre a asa	67
4.24	Campo de pressão em diferentes seções da asa.....	68
4.25	Coeficiente de pressão obtido pela simulação e obtido experimentalmente em Love (1949) para a seção $y = 0.255$ da envergadura	69

LISTA DE TABELAS

3.1	Dimensões utilizadas no aerofólio de dupla cunha	35
3.2	Parâmetros geométricos das asas tridimensionais simuladas.....	35
3.3	Informações das malhas estáticas bidimensionais sobre o aerofólio de dupla cunha	36
3.4	Parâmetros do escoamento sobre o aerofólio dupla cunha simulado no OpenFOAM...	40
3.5	Condições de contorno aplicadas nas regiões destacadas na Fig. 3.15.....	40
3.6	Condições de contorno aplicadas na simulação do aerofólio dupla cunha no VAT	42
4.1	Valores de coeficiente de arrasto para a simulação no OpenFOAM sem malha adaptativa.....	55
4.2	Valores de coeficiente de arrasto para a simulação no OpenFOAM com malha adaptativa.....	58
4.3	Valores de coeficiente de arrasto para a simulação no OpenFOAM com malha adaptativa e com ângulo de ataque.....	58
4.4	Erro da simulação no VAT comparado com o resultado teórico	63
4.5	Comparação entre os erros das simulações com o VAT e com o OpenFOAM.....	67

LISTA DE SÍMBOLOS

Símbolos Latinos

C_p	Calor específico à pressão constante	[kJ/kg.K]
C_v	Calor específico à volume constante	[kJ/kg]
h	Entalpia específica	[kJ/kg]
E	Energia total específica	[kJ/kg]
e	Energia interna específica	[kJ/kg]
\dot{m}	vazão mássica	[kg/s]
T	Temperatura	[°C]
t	Tempo	[s]
V	Volume	[m³]
\mathbf{u}	Vetor velocidade	[m/s]
\mathbf{n}	Vetor unitário normal à superfície	
\mathbf{i}	Vetor unitário na direção x	
\mathbf{j}	Vetor unitário na direção y	
\mathbf{k}	Vetor unitário na direção z	
a	Velocidade do som	[m/s]
S	Área	[m²]
P	Pressão	[N/m²]
L	Força de sustentação	[N]
D	Força de arrasto	[N]
A	Força axial	[N]
N	Força normal	[N]
\mathcal{R}	Resultante de forças sobre um aerofólio	[N]
k	Condutividade térmica	[W/(m K)]
M	Momento longitudinal	[N.m]
Q	Calor	[kJ]
Q	Termo de fonte	
\mathbf{q}	fluxo de calor	[W/m²]
\mathcal{W}	Trabalho	[kJ]
c	corda	[m]
c_d	Coefficiente de arrasto de perfil aerodinâmico	
c_l	Coefficiente de sustentação de perfil aerodinâmico	

c_m	Coefficiente de momento de perfil aerodinâmico	
R	Constante do ar	[J/Kg.K]
F	Fluxo volumétrico	[m ³ /s]
\mathcal{M}	Fluxo de massa	[kg/s]
\mathbf{f}	Força de campo	[N/kg]
\mathbf{d}	Distância entre dois volumes	[m]
$\mathbf{d}_{f,n}$	Distância entre o centro da face e o centroide do volume vizinho	[m]
I	Matriz identidade	
\mathcal{A}	Área da planta (vista superior) de uma asa	[m ²]

Símbolos Gregos

ρ	Massa específica	[kg/m ³]
τ	Tensão de cisalhamento	[N/m ²]
α	Ângulo de ataque	[°]
μ	Viscosidade dinâmica ou ângulo de onda de mach	[Pa/s] ou [rad]
λ	Viscosidade expansional	[Pa/s]
δ	Quantidade infinitesimal	
γ	Razão entre calores específicos	
Γ	Constante de difusão	[m ² /K]
θ	Ângulo de deflexão das linhas de corrente no choque oblíquo	[°]
β	Ângulo do choque oblíquo	[°]

Grupos Adimensionais

Re	Número de Reynolds
Ma	Número de Mach

Subscritos

SC	Superfície de controle
VC	Volume de controle
T	Total
x	Direção x
y	Direção y
z	Direção z
xx	Plano x e direção x
xy	Plano x e direção y
xz	Plano x e direção z
yx	Plano y e direção x
yy	Plano y e direção y
yz	Plano y e direção z
zx	Plano z e direção x
zy	Plano z e direção y
zz	Plano z e direção z
∞	Escoamento não perturbado
ref	referência
n	Direção normal
t	Direção tangencial
f	Face
p	Centro do VC analisado
N	Centro do VC vizinho

Sobrescritos

\cdot	Variação temporal
—	Valor médio
*	Grandeza adimensional
<i>n</i>	n-ésima iteração
REF	Referência
b	Fronteira imersa

Siglas

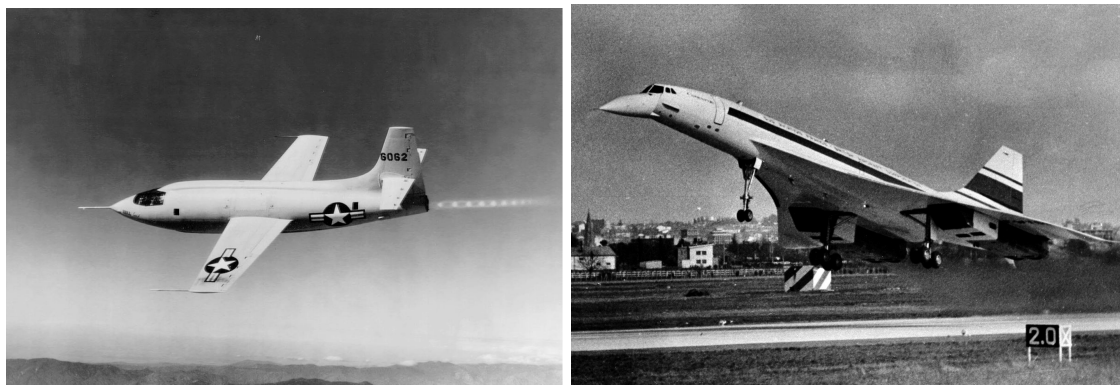
CAALab	Laboratório de Aero Acústica Computacional da UnB
VAT	Virtual Aeroacoustic Tunnel
PISO	Pressure Implicit Split Operator
M.A.C	Mean aerodynamic chord
SST	Supersonic Transportation
CFD	Computational Fluid Dynamics
HPC	High Performance Computing

Capítulo 1

Introdução

1.1 Contextualização

A primeira aeronave a voar acima da velocidade do som foi o *Bell XS-1* (Fig. 1.1a), que atingiu a velocidade de Mach $Ma = 1,06$ em 1947 (1). Até a década de 1960, as aeronaves que possuíam capacidade de ultrapassar a velocidade do som eram exclusivamente militares (2), até que surgiram projetos de aeronaves de transporte supersônico civil (*Supersonic Transport, SST*), e um deles resultou no anglo-francês *Concorde* (Fig 1.1b), que era capaz de voar a $Ma = 2$. Apesar de ter sido considerado o transporte do futuro, o *Concorde* foi uma falha comercial se comparado com as aeronaves contemporâneas, com apenas 16 unidades comerciais produzidas, o que marcou o fim temporário dos *SSTs*. Isso se deve em parte à questão do elevado gasto de combustível e ao elevado ruído gerado, ambos causados principalmente pela onda de choque (3).



(a) *Bell XS-1*, o primeiro avião a voar acima da velocidade do som

(b) *Concorde* em seu primeiro voo

Figura 1.1: Aeronaves capazes de voar acima da velocidade de som.
(4,5)

Nas últimas décadas, a computação se desenvolveu muito, e deu início à chamada *era digital*. Estima-se que a capacidade de computação de CPUs dobre a cada 2 anos (6). Consequentemente, a modelagem de escoamentos pela dinâmica dos fluidos computacional (*Computational Fluid Dy-*

namics, CFD) é cada vez mais complexa, como pode ser visto pela Fig. 1.2.

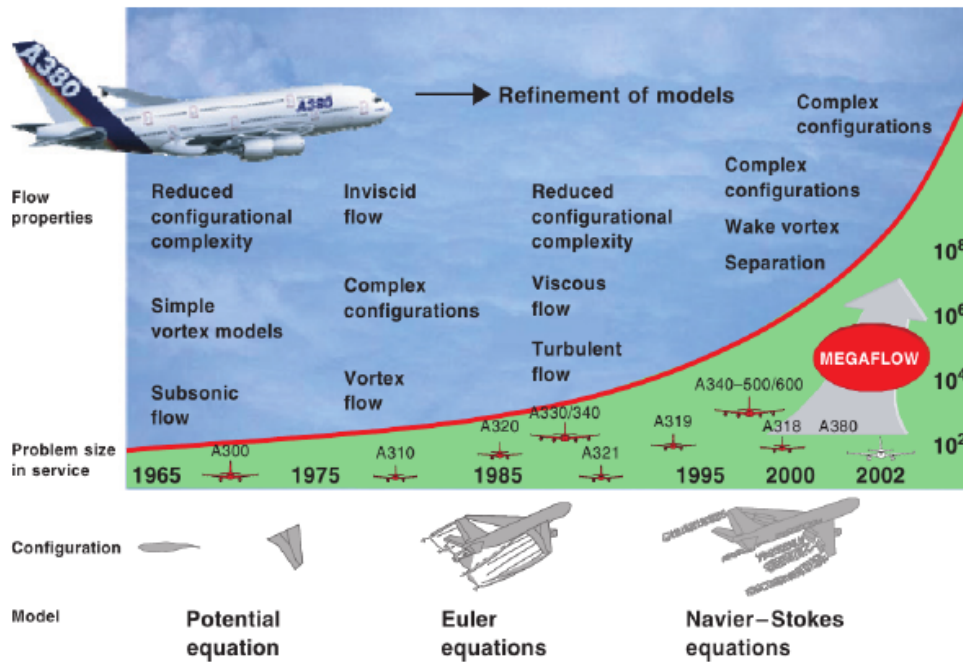


Figura 1.2: Crescimento da complexidade das simulações de dinâmica dos fluidos computacional (7)

Com o avanço da era digital, houve também um avanço nas linhas de pesquisas dedicadas ao estudo da geração de ondas de choque, e concomitantemente novos projetos de SSTs estão sendo lançados. A *National Aeronautics and Space Administration* (NASA) e a *Lockheed Martin*, por exemplo, começaram em 2018 a construção de um protótipo de um jato *low-boom* (Fig. 1.3).

1.2 Definição do problema

Com o supracitado, é notório o interesse da indústria aeronáutica pela aviação supersônica civil, já que ela pode diminuir substancialmente o tempo para transportar bens e pessoas. Contudo, para que uma aeronave supersônica seja utilizada comercialmente em larga escala é necessário, entre muitos outros fatores, atenuar a geração de ondas de choque, para que ela voe sobre os países sem causar poluição sonora acima do aceitável e para que não consuma combustível demasiadamente. Para isso, os *softwares* de CFD precisam estar adaptados para lidar com problemas complexos e para seguir o crescimento do poder computacional, sobretudo em arquiteturas de computação de alto desempenho (*High Performance Computing, HPC*)

Segundo (1), a solução do escoamento tridimensional é ainda um problema do estado da arte da dinâmica dos fluidos computacional, no ponto de vista da precisão numérica, da eficiência de cálculo da solução e dos métodos apropriados para apresentar, estudar e interpretar a grande

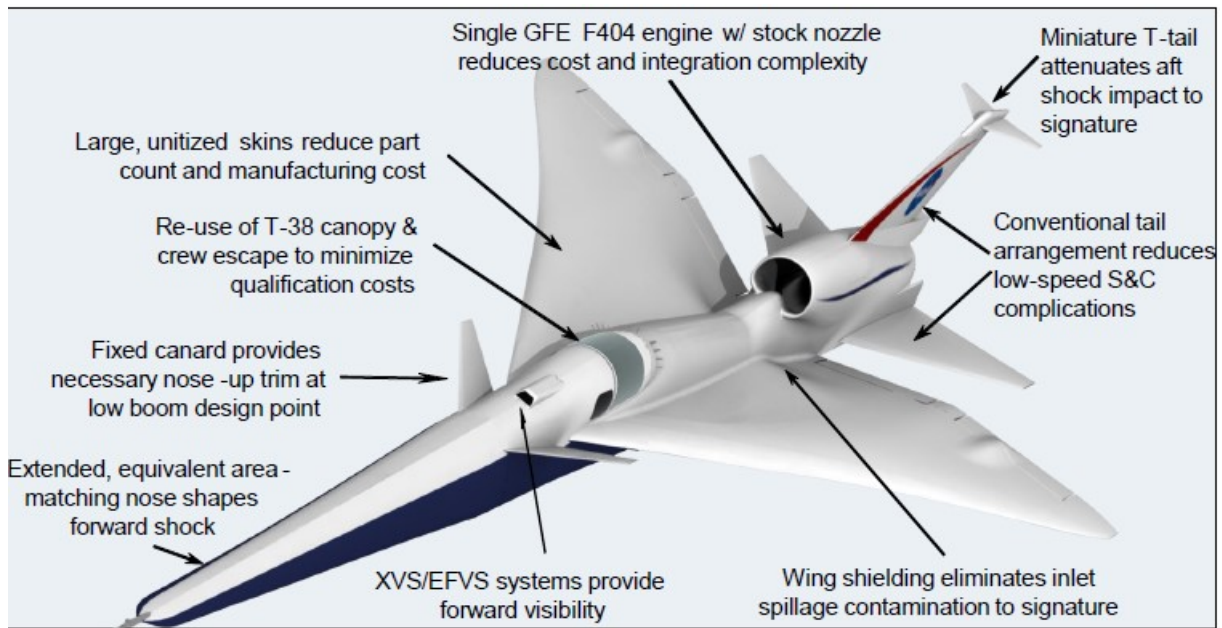


Figura 1.3: Projeto de aeronave supersônica *low-boom*
(8)

quantidade de dados. Além disso, em simulações de volumes finitos, a abordagem de captura de choque possui a vantagem de não requerer o conhecimento *a priori* do número ou do formato das ondas de choque, porém uma desvantagem é que requer mais elementos, sendo que muitos volumes no escoamento não perturbado são desnecessários.

Atualmente há *softwares* de CFD privados e livres, como o OpenFOAM. Os *softwares* livres estão sob a *General Public License (GNU)*, que estabelece 4 liberdades:

- A liberdade de executar o programa, para qualquer propósito
- A liberdade de estudar como o programa funciona e adaptá-lo às suas necessidades.
- A liberdade de redistribuir cópias.
- A liberdade de aperfeiçoar o programa e liberar os seus aperfeiçoamentos.

É consenso que o desempenho de um *software* de CFD livre tende a não ser melhor que um privado. Porém, a relação entre preço e desempenho pode ser determinante na hora de se escolher qual utilizar. Diante disso, surge a pergunta: como é o desempenho de CFD código livre em um *cluster* em questão de precisão e custo computacional? O presente trabalho se propõe a responder essa pergunta no âmbito de simulações de escoamentos supersônicos.

1.3 Revisão bibliográfica

A referência (9) relata que um dos perfis com maior eficiência em regimes supersônicos é a placa plana (Fig. 1.4c). Mas, por motivos estruturais, perfis reais precisam ter uma espessura finita, e

entre os perfis monoplanos com espessura finita, os que possuem uma das melhores razões entre sustentação e arrasto são o perfil dupla cunha (Fig. 1.4a) e o perfil biconvexo (Fig. 1.4b). Pelo fato de o perfil de dupla cunha possuir uma simetria e simplicidade que ajudam na definição analítica do escoamento, ele foi utilizado em ensaios experimentais em túnel de vento de alta velocidade para a validação de teorias de escoamento supersônico pela medição da força resultante e da distribuição de pressão na superfície de asas (10, 11, 12).

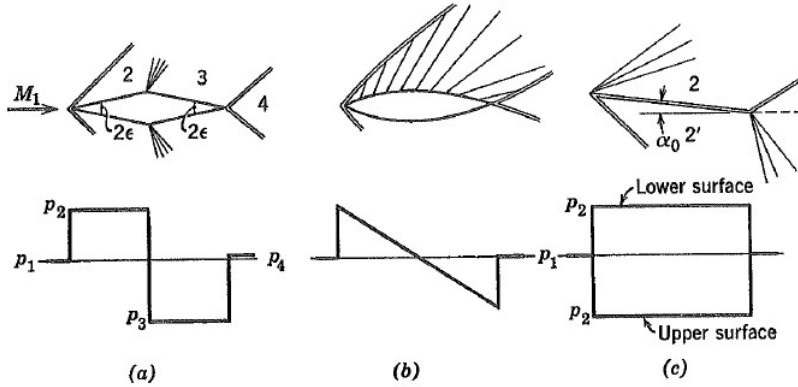


Figura 1.4: Variação de pressão em a) aerofólio dupla cunha b) aerofólio biconvexo c) placa plana quando submetidos a escoamento supersônico (13)

Em (14) foi feita a validação do VAT a partir de diferentes casos de escoamento supersônico. Um desses casos foi o do aerofólio dupla cunha, em que se analisou apenas a deflexão das linhas de corrente ao passar pelo choque oblíquo. Também foram feitos trabalhos para a validação do OpenFOAM em escoamentos supersônicos (15, 16, 17, 18). Os resultados apresentaram grande correlação com os valores teóricos (Fig. 1.5).

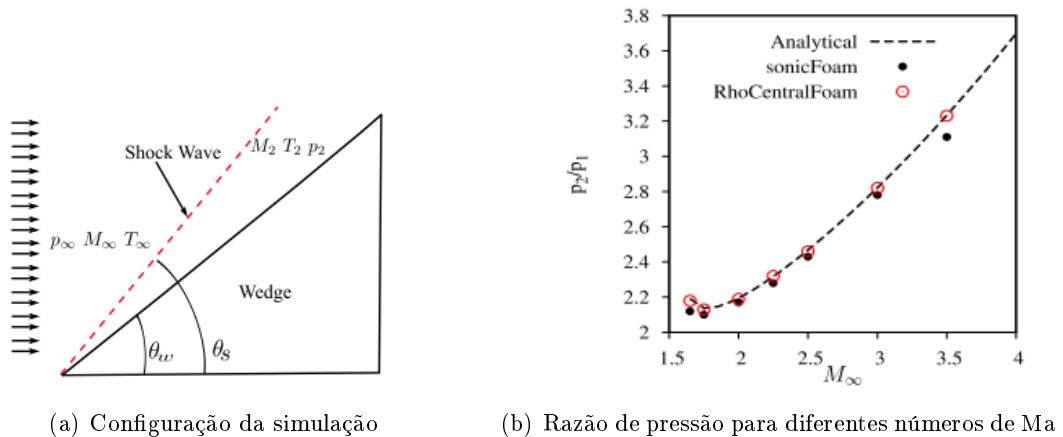


Figura 1.5: Resultados da validação do OpenFOAM (16)

Além disso, desenvolveram-se trabalhos acerca de metodologias para acelerar a convergência e para melhorar a precisão sem aumentar em demasia o custo computacional (19,20). Um dos

mecanismos que poderia ser citado é o refinamento automático de malha, que pode ser atrelado a regiões de alto gradiente, com o objetivo de homogeneizar o erro no domínio e otimizar o custo computacional. O método consiste em iniciar o cálculo com uma malha inicial, e por meio de um parâmetro, como o gradiente de velocidade, decide-se acerca das mudanças necessárias na malha. Então a presente malha é modificada de acordo com os parâmetros pré definidos e o resultado numérico é mapeado para a nova malha, sendo utilizado como estimativa inicial para a próxima iteração. A Fig. 1.6 mostra uma comparação feita em (19) da evolução do erro numérico em uma simulação com malha adaptativa e outra com malha homogênea.

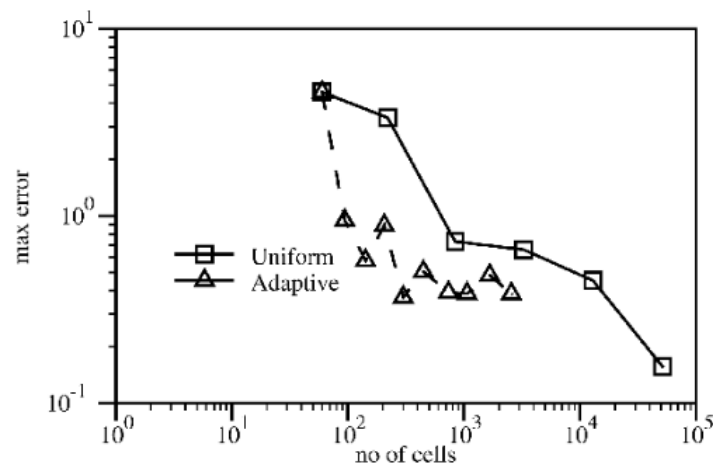


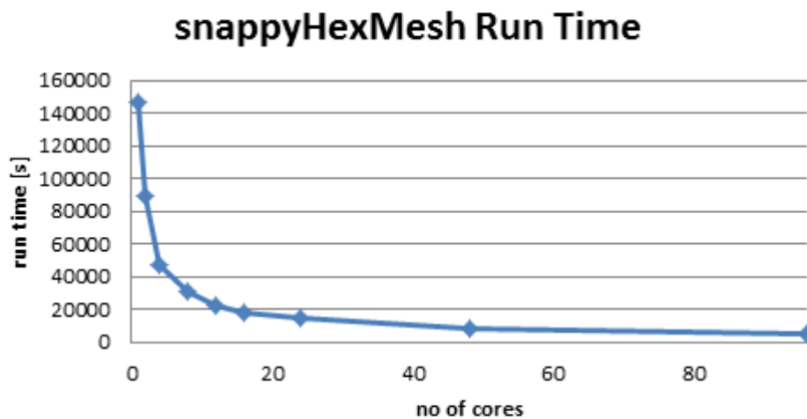
Figura 1.6: Evolução do erro numérico com o número de volumes para uma malha homogênea e uma malha adaptativa

(19)

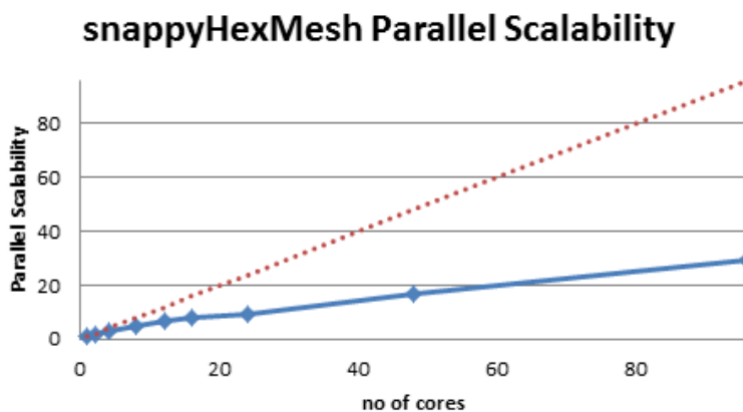
Em (21) foi feito um estudo da escalabilidade do OpenFOAM em execução paralela em ambiente *HPC*, com processo de geração de malha e simulação de escoamento incompressível com equações médias de Navier-Stokes. As simulações foram executadas no *cluster Cray XC40 Beskow* no *KTH Royal Institute of Technology*, e no *cluster MareNostrum*, na Espanha. O *Beskow* é baseado em Intel *Haswell* e tecnologia de comunicação *Cray Aries*, consistindo de 1676 nós de processamento, cada um com 32 núcleos Intel Xeon E5-2698v3. Já o *MareNostrum* possui 36 racks dedicados a cálculos. Esses racks possuem um total de 48448 núcleos Intel *SandyBride* com uma frequência de 2.6 GHz e 94,625 TB de memória total. Comparou-se o desempenho na distribuição da geometria *STL* em vez da abordagem padrão em que o dado da geometria é copiado por todo nó de computação, e portanto reduzindo redundância e consumo de memória. A distribuição do domínio na abordagem padrão seguiu o método *scoth* (22).

A escalabilidade do *snappyHexMesh* (código para gerar a malha do OpenFOAM) foi avaliada com a geração de uma malha de 160 milhões de elementos no *Beskow* (Fig. 1.7) e a escalabilidade do *simpleFoam* (algoritmo de solução do escoamento) foi avaliada em uma malha de 110 milhões de elementos (Fig. 1.8). Verificou-se que a distribuição do arquivo STL pode reduzir na ordem de 30% o consumo de memória, mas acarreta em um aumento de tempo de processamento significativo. O tempo de execução por *time step* reduziu de 13,874 para 1,158 segundos quando executado em

2048 núcleos no *Marenstrum* e de 17,057 para 0,643 segundos quando executado em 4096 núcleos no *Beskow*. Concluiu-se em (21) que, apesar de a escalabilidade do processo de geração de malha ser pobre, foi demonstrado que um caso complexo poderia ser desenvolvido em *clusters* em um tempo limitado, já que uma malha de 350 milhões de elementos foi gerada em menos de 25 minutos utilizando 4096 núcleos.



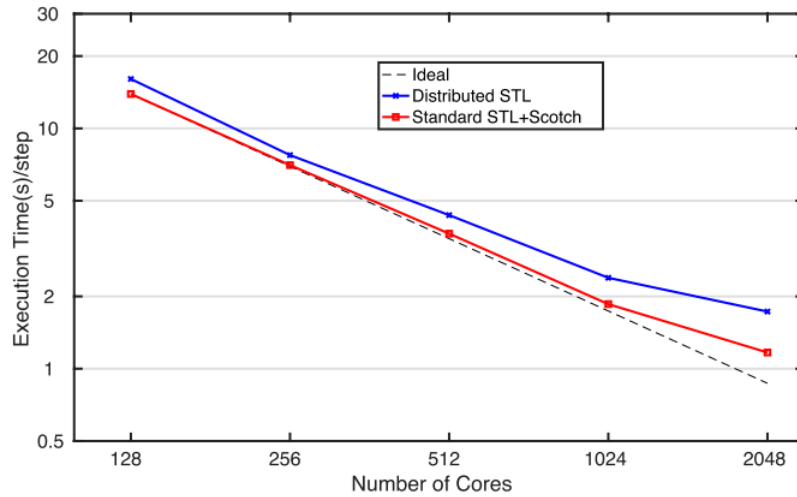
(a)



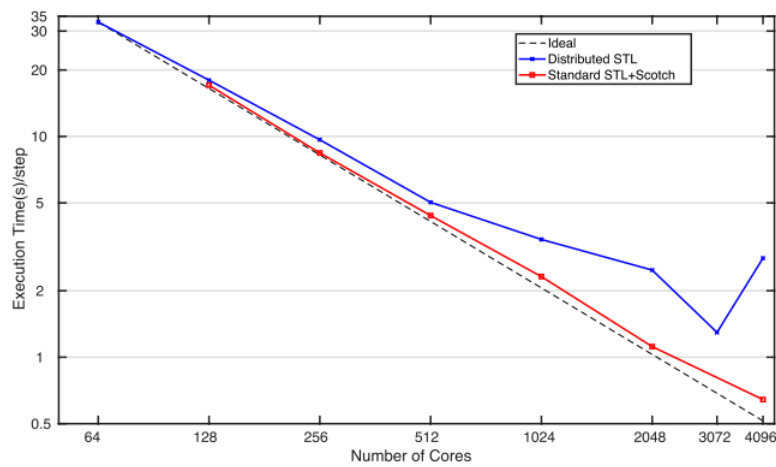
(b)

Figura 1.7: a) tempo de simulação e b) escalabilidade do algoritmo *snappyHexMesh* de geração de malha do OpenFOAM

(21)



(a) *cluster MareNostrum*



(b) *cluster Beskow*

Figura 1.8: escalabilidade do algoritmo *simpleFoam* de solução da simulação no OpenFOAM (21)

1.4 Objetivos

O presente trabalho tem como objetivo geral descrever o desempenho do OpenFOAM no cluster do Laboratório de Aeroacústica Computacional (CAALab) da UnB em simulações de escoamentos supersônicos sobre aerofólios bidimensionais e sobre asas.

Estabeleceu-se os seguintes objetivos específicos:

1. Desenvolver uma metodologia para cálculo analítico dos parâmetros do escoamento supersônico sobre o aerofólio de dupla cunha para servir de base de comparação.
2. Validar as simulações de escoamentos supersônicos bidimensionais sobre aerofólios de dupla cunha e sobre asas no OpenFOAM, comparando com os resultados analíticos e experimentais.

3. Comparar o desempenho do OpenFOAM com relação ao VAT, que é um código desenvolvido no CAALab.

1.5 Apresentação do trabalho

No capítulo 2 é feita uma revisão dos fundamentos teóricos sobre o tema de estudo. Em seguida, o capítulo 3 descreve a metodologia empregada no desenvolvimento do projeto incluindo a campanha de simulações. Os resultados das simulações são apresentados e discutidos no capítulo 4, seguido das conclusões no capítulo 6. Os anexos contém material complementar.

Capítulo 2

Fundamentação teórica

2.1 Dinâmica dos fluidos computacional

2.1.1 Método dos Volumes finitos

Entre técnicas utilizadas para implementar o CFD, o Método de Volumes Finitos (*Finite Volume Method*, FVM) é muito utilizado. Segundo a referência (9), o FVM é uma técnica numérica que transforma as equações que representam princípios físicos (conservação da massa, balanço de força, entre outros) sobre um domínio contínuo em equações algébricas linearizadas sobre volumes finitos (ou elementos de discretização).

O primeiro passo no processo de solução pelo FVM é a discretização do domínio geométrico em volumes finitos. As equações diferenciais parciais são então transformadas em equações algébricas, avaliadas sobre cada elemento discreto. O sistema de equações algébricas é então resolvido para calcular os valores da variável dependente para cada um dos elementos. Alguns dos termos das equações são transformados em fluxos de faces e avaliados nas faces de volume finito. Pelo fato do fluxo que entra em um determinado volume ser idêntico ao que deixa o volume adjacente, o FVM é dito ser conservador e faz com que seja método preferido em CFD.

2.1.2 Estabilidade numérica

Segundo (23), não há garantia para a precisão e estabilidade de um sistema baseado em diferenças finitas sob qualquer condição de contorno. Mas, para sistemas lineares, há uma maneira formal para avaliar a precisão e estabilidade, e essas maneiras podem servir de guias para controlar o comportamento de sistemas de equações não lineares e de maior complexidade. Em (23) é derivada a relação de estabilidade denominada número de Courant (Eq. 2.1).

$$Co = \frac{c + u_{\infty}}{\left(\frac{\Delta x}{\Delta t}\right)} \quad (2.1)$$

Na Eq. 2.1, Δx é o menor comprimento de um volume da malha computacional, c é a velocidade

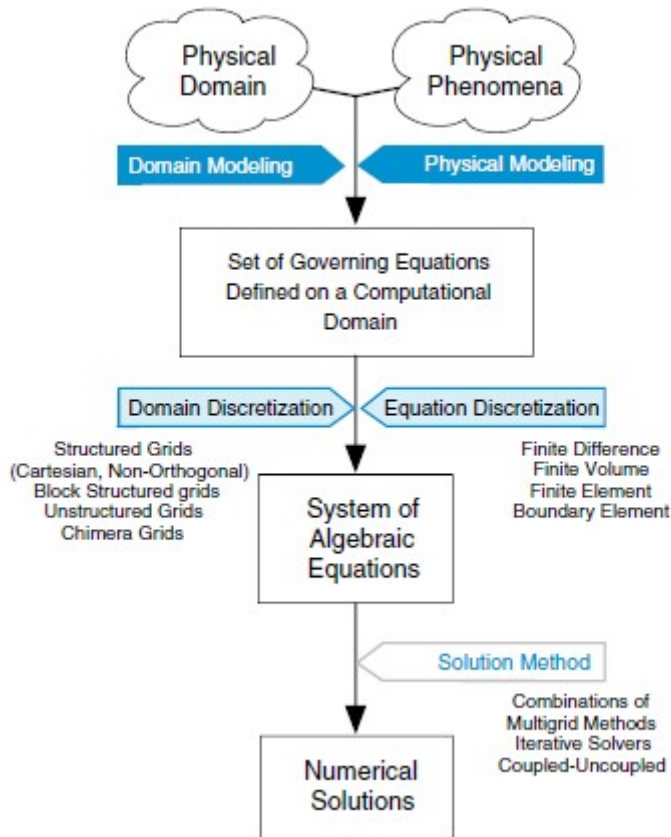


Figura 2.1: Processo de simulação computacional
(9)

do som, u_∞ é a velocidade do escoamento, $c + u_\infty$ é a maior velocidade do fluido e Δt é o passo de tempo adotado. Para que a simulação em que se utiliza métodos explícitos, é necessário que $Co < 1$. Mas caso a simulação utilize métodos implícitos de modo a atingir mais rapidamente a convergência, Co pode ser maior que um. Como regra geral, utiliza-se um número de Courant suficientemente baixo, e para isso é feito um ajuste no Δt .

2.2 Regimes de escoamento

Segundo (1), o número de Mach é definido pela Eq. 2.2 , na qual Ma é o número de Mach e a é a velocidade do som no meio.

$$Ma = \frac{u}{a} \quad (2.2)$$

Ainda segundo a referência (1), de acordo com o número de Mach do escoamento livre à montante, pode-se estabelecer três regimes de escoamento: subsônico, transônico, supersônico e hipersônico.

No regime subsônico, todas os pontos possuem $Ma < 1$ (Fig.2.2a), e, portanto, todos os pontos estão com velocidade abaixo da velocidade do som. O escoamento é caracterizado por linhas de corrente suaves, com variação contínua das propriedades e que começam a defletir a uma grande distância do aerofólio. É comum classificar o escoamento como subsônico quando $Ma_\infty < 0,8$.

No regime transônico, em algumas regiões o número de Mach é menor que 1, e em outras é maior que 1. Na Fig. 2.2b, o escoamento livre à montante é subsônico, mas torna-se supersônico no extradorso do aerofólio, passa por uma onda de choque e volta a ser subsônico. Já na Fig. 2.2c, o escoamento livre à montante possui número de Mach pouco maior que a unidade, mas gera um choque destacado que é seguido por uma região subsônica, que por sua vez é seguida por outra região supersônica. É comum classificar o escoamento como transônico quando $0,8 < Ma_\infty < 1,2$.

Já no caso supersônico, o escoamento possui $Ma > 1$ em todos os pontos à frente da onda de choque, uma vez que a presença do corpo não é "informada" ao escoamento à montante, devido a velocidade do som ser menor que a velocidade do escoamento. Na Fig. 2.2d, é possível ver o escoamento supersônico sobre uma superfície afiada, e é ilustrada a presença do choque oblíquo, que se conecta com a ponta. Dentro do choque oblíquo, a linha de corrente deflete de forma descontínua. Antes do choque, as linhas de corrente são paralelas e horizontais, e depois do choque as linhas de corrente seguem a geometria do corpo, e continuam retas e paralelas.

Por fim, no regime hipersônico, a temperatura, pressão e massa específica aumentam exponencialmente após a onda de choque (Fig. 2.2e). A onda de choque se aproxima muito da superfície, e causa efeitos de ionização e dissociação devido às altas temperaturas.

2.3 Equações governantes e condições de contorno

De acordo com (1), para se obter as equações governantes do movimento de fluidos, deve-se seguir três passos. O primeiro consiste em considerar os princípios físicos que dominam o problema, que no caso são o princípio da conservação da massa, a segunda lei de Newton e a primeira lei da termodinâmica. O segundo passo consiste em aplicar esses princípios em um modelo apropriado do escoamento. Já o terceiro consiste em derivar as equações matemáticas a partir dos princípios físicos.

A equação da continuidade utiliza o princípio de que a massa de um determinado sistema não pode variar. Considere então um volume de controle como o representado na Fig. 2.3. Pela aplicação do Teorema do Transporte de Reynolds para a massa, tem-se que o fluxo de massa que sai pela superfície será igual à taxa de decréscimo da massa desse volume de controle (Eq. 2.3a). Aplicando-se o Teorema de Gauss e o Teorema da Localização, obtém-se a Eq. 2.3b.

$$\frac{\partial}{\partial t} \int_V \rho dV + \int_S \rho \mathbf{u} \cdot d\mathbf{S} = 0 \quad (2.3a)$$

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0 \quad (2.3b)$$

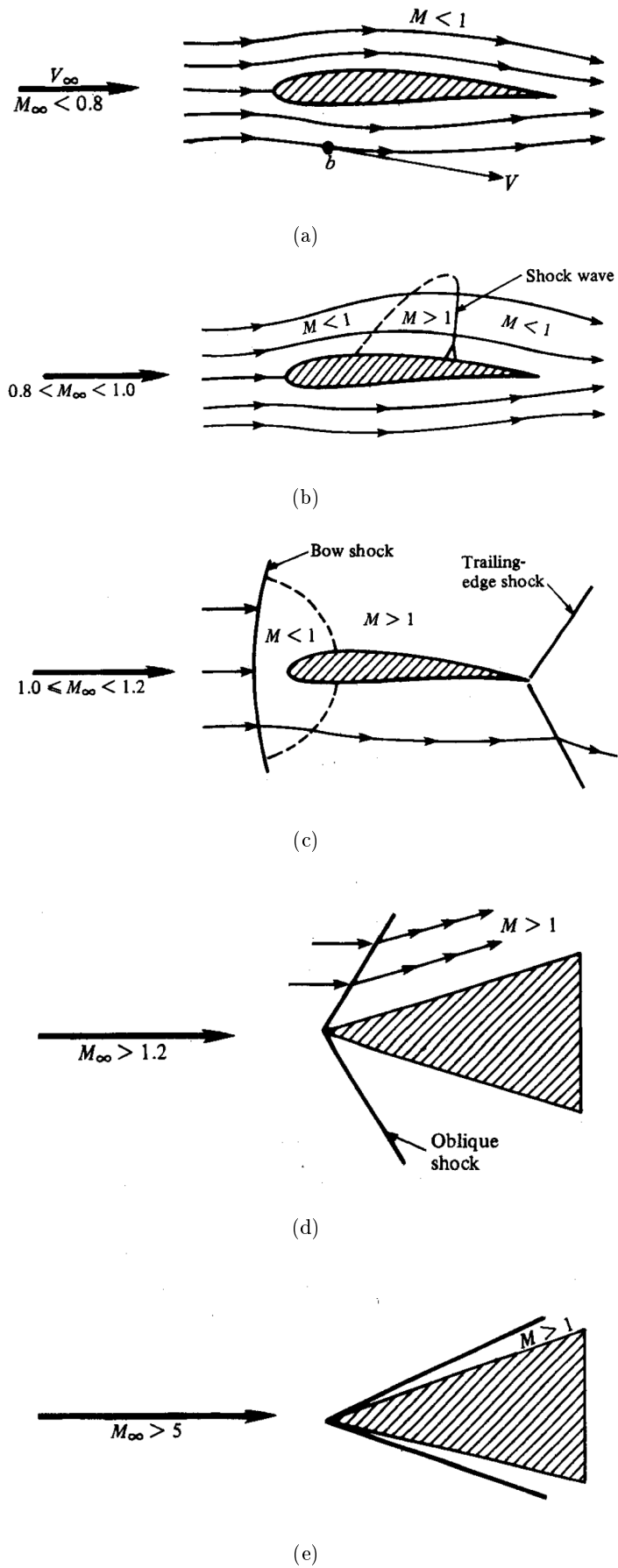


Figura 2.2: Ilustração de diferentes regimes de voo
(1)

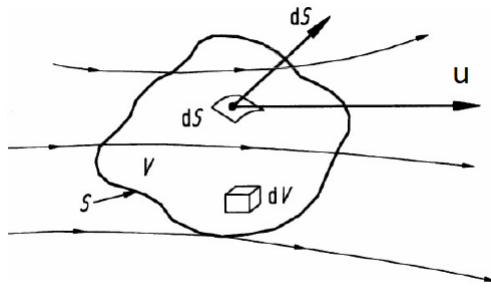


Figura 2.3: Desenho esquemático de um volume de controle
(1)

Já a equação da quantidade de movimento utiliza a segunda lei de Newton, que diz que a força resultante sobre um sistema, que é o somatório das forças de campo e de contato, é igual a sua taxa de variação da quantidade de movimento. Na Fig. 2.4 estão representadas as forças de contato que atuam no elemento de fluido na direção x devido à pressão e às tensões de cisalhamento, que estão presentes por causa da viscosidade do fluido (1). Apesar de estarem aqui representadas apenas as forças na direção x, a equação da quantidade de movimento é derivada para as três direções do espaço.

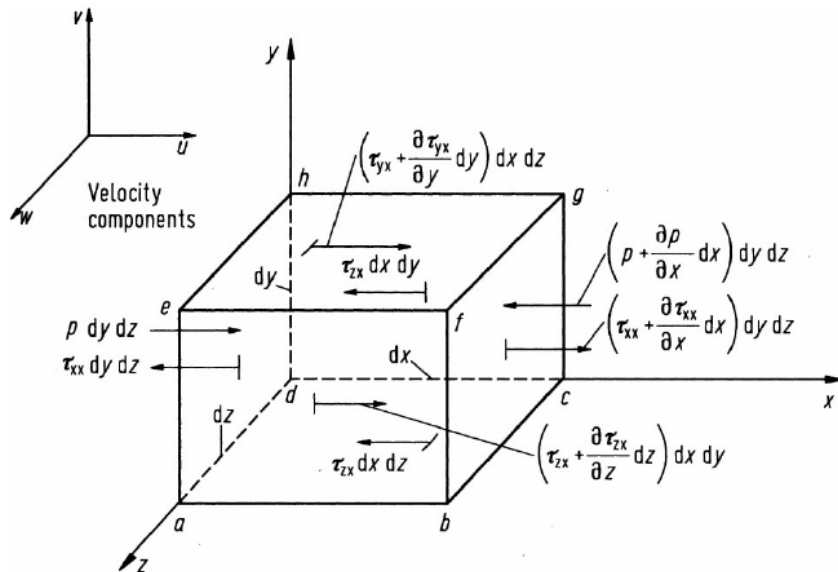


Figura 2.4: Desenho esquemático de um elemento infinitesimal sob a ação de forças de contato advindas da pressão e de efeitos viscosos, todas na direção x
(1)

Após a realização do somatório de forças, iguala-se o resultado à taxa de variação da quantidade de movimento, e com a utilização do Teorema do Transporte de Reynolds, obtém-se a Eq. 2.4a. Com a aplicação do Teorema de Gauss e do Teorema da Localização, obtém-se a Eq. 2.4b. O termo \mathbf{f} , referente à força de campo, pode ser desconsiderada, pois no caso que será analisado a

única força de campo é a gravitacional, e ela é consideravelmente menor que as forças de contato. Para um fluido newtoniano, pode-se considerar que o tensor taxa de deformação é dado pela Eq. 2.4c.

$$\int_V \frac{\partial(\rho \mathbf{u})}{\partial t} dV + \int_S (\rho \mathbf{u} \mathbf{u}) d\mathbf{S} = - \int_S P \bullet d\mathbf{S} + \int_S \underline{\tau} \bullet d\mathbf{S} + \int_V \rho \mathbf{f} dV \quad (2.4a)$$

$$\frac{\partial(\rho \mathbf{u})}{\partial t} + \nabla \bullet (\rho \mathbf{u} \mathbf{u}) = -\nabla P + \nabla \bullet \underline{\tau} + \rho \mathbf{f} \quad (2.4b)$$

$$\underline{\tau} = \mu \left(\nabla \mathbf{u} + (\nabla \mathbf{u})^T - \frac{2}{3} (\nabla \bullet \mathbf{u}) \underline{I} \right) \quad (2.4c)$$

Por fim, a equação de energia utiliza o princípio da conservação da energia de um sistema, que se traduz na primeira lei da termodinâmica, segundo a qual a variação da energia interna de um sistema é igual ao calor recebido menos o trabalho realizado. Essa relação é apresentada na forma de taxa de variação temporal pela Eq. 2.5a.

$$\frac{dE}{dt} = \delta \dot{Q} - \delta \dot{W} \quad (2.5a)$$

$$E = e + \frac{|\mathbf{u}|^2}{2} \quad (2.5b)$$

Na Eq. 2.5a, E é a energia específica total, Q é o calor transferido para o sistema e W é o trabalho realizado pelo sistema. Pode-se considerar a energia total específica de um sistema como a soma da energia interna e da energia cinética específicas, como é apresentado na Eq. 2.5b.

É possível demonstrar, a partir do Teorema do Transporte de Reynolds, que a equação da energia toma a forma da Eq. 2.6a, desconsiderando-se novamente os efeitos das forças de campo. Já com o Teorema de Gauss e o Teorema da Localização, obtém-se a Eq.

$$\int_V \frac{\partial \rho E}{\partial t} dV + \int_S (\rho E) \mathbf{u} \bullet d\mathbf{S} = - \int_S \mathbf{q} \bullet d\mathbf{S} - \int_S P \mathbf{u} \bullet d\mathbf{S} + \int_S \underline{\tau} \bullet \mathbf{u} \bullet d\mathbf{S} \quad (2.6a)$$

$$\frac{\partial \rho E}{\partial t} + \nabla \bullet ([\rho E] \mathbf{u}) = -\nabla \bullet \mathbf{q} - \nabla \bullet (P \mathbf{u}) + \nabla \bullet (\underline{\tau} \bullet \mathbf{u}) \quad (2.6b)$$

Dessa forma, tem-se as equações governantes na forma conservativa, e o conjunto de equações 2.3b, 2.4b e 2.6b é comumente chamado de equações de Navier Stokes. Se nessas equações os efeitos viscosos forem desprezados, obtém-se o conjunto de equações denominado de equações de Euler.

O conjunto de equações acima possui mais incógnitas que equações. Dessa forma, é necessário estabelecer relações de fechamento, e utiliza-se as relações de estado termodinâmico. A primeira relação utilizada é a partir da consideração de gás ideal, pela Eq. 2.7. Essa equação é chamada de

equação de estado. Nessa equação, R é a constante do ar.

$$P = \rho RT \quad (2.7)$$

A segunda relação utilizada é baseada na consideração de gás caloricamente e termicamente perfeito, pela Eq. 2.8, em que C_v e C_p são os calores específicos a volume e a pressão constante.

$$e = C_v T \quad (2.8)$$

$$h = C_p T \quad (2.9)$$

Além das relações de fechamento, é necessário considerar valores para as fronteiras. Pode-se considerar que um fluido viscoso obedece a condição de não escorregamento, sendo então a velocidade na região da parede nula em todas as direções. Essa condição pode ser descrita pela Eq. 2.10. Após a parede, existe uma região com altos gradientes de velocidade até chegarmos na velocidade do escoamento livre.

$$\mathbf{u} = 0 \quad (2.10)$$

Quando consideramos um fluido pouco viscoso ou com baixa taxa de deformação, podemos usar a condição de impenetrabilidade, ou seja, as linhas de corrente não podem penetrar a parede (Eq. 2.11).

$$\mathbf{u} \cdot \mathbf{n} = 0 \quad (2.11)$$

Além das condições de contorno listadas, existem outras inúmeras que podem ser utilizadas, dependendo do problema a ser tratado.

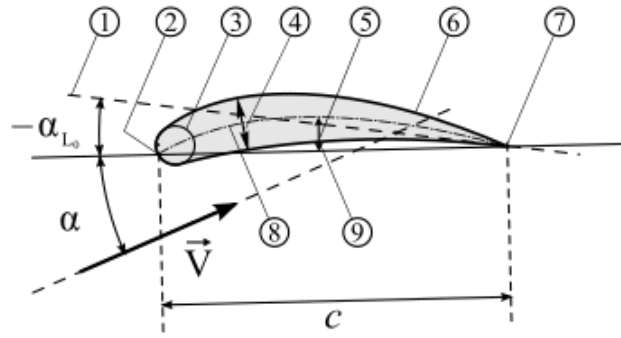
2.4 Forças sobre um aerofólio

Um aerofólio convencional pode ser esquematizado de acordo com a Fig. 2.4a. O ar passa pela superfície superior e inferior com diferentes valores de pressões, e formam as forças de acordo com a Fig. 2.4b. As forças de interesse são as forças de sustentação L e a força de arrasto D , calculadas a partir da Eq. 2.12.

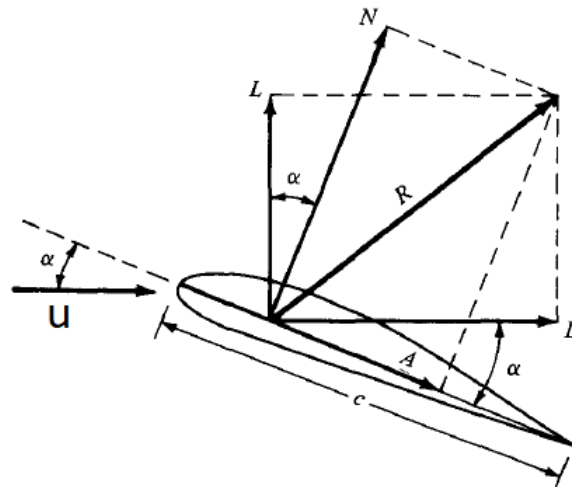
$$L = N \cos \alpha - A \sin \alpha \quad (2.12a)$$

$$D = N \sin \alpha + A \cos \alpha \quad (2.12b)$$

A utilização de coeficientes aerodinâmicos é útil em diversas análises, e eles podem ser definidos



(a) 1-linha de sustentação zero, 2 -bordo de ataque, 3- círculo de nariz, 4-espessura máxima, 5-arqueamento, 6- extradorso, 7-bordo de fuga, 8-linha média, 9- intradorso



(b) Forças resultantes sobre um aerofólio

Figura 2.5: Esquema de um aerofólio
(24,1)

pela Eq. 2.13a, em que \mathcal{A} é a área da vista superior da asa. Em casos bidimensionais, considera-se que a espessura é unitária, e no lugar de \mathcal{A} , considera-se c , que é a corda.

$$c_l = \frac{L}{\frac{1}{2}\rho_\infty u_\infty^2 \mathcal{A}} \quad (2.13a)$$

$$c_d = \frac{D}{\frac{1}{2}\rho_\infty u_\infty^2 \mathcal{A}} \quad (2.13b)$$

Capítulo 3

Metodologia

3.1 Cálculo analítico do escoamento

3.1.1 Equações de ondas de choque oblíquas

Segundo a referência (1), em um choque unidimensional adiabático (Fig. 3.1), as equações governantes na forma integral podem ser reescritas como nas Eq. 3.1 a 3.4, que consideram o fluido como caloricamente e termicamente perfeito. Denomina-se a região 1 como sendo à montante do choque e a região 2 à jusante.

$$\rho_1 u_1 = \rho_2 u_2 \quad (3.1)$$

$$P_1 + \rho_1 u_1^2 = P_2 + \rho_2 u_2^2 \quad (3.2)$$

$$h_1 + \frac{u_1^2}{2} = h_2 + \frac{u_2^2}{2} \quad (3.3)$$

$$h = e + \frac{P}{\rho} \quad (3.4)$$

Se o escoamento passa por uma região onde há uma quina côncava, ocorrerá uma onda de choque oblíqua, colada à superfície (Fig. 3.2), ou destacada (Fig. 3.3), a depender da diferença angular e do número de Mach à montante.

Para analisar a onda de choque oblíqua, é necessário decompor a velocidade à montante e à jusante do choque nas direções tangencial e normal, conforme a Fig. 3.4. Ao integrar as equações governantes sobre o volume de controle destacado na Fig. 3.4, verifica-se que a velocidade tangencial não se altera. Já a velocidade normal ao choque segue as equações relativas ao choque normal. Dessa forma, é possível calcular as propriedades da região de saída com base no ângulo do choque (β) e nas propriedades da região de entrada.

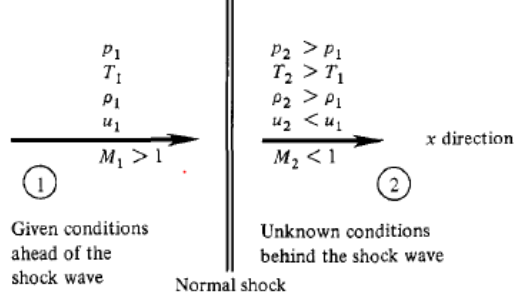


Figura 3.1: Esquema de um choque normal adiabático unidimensional
(1)

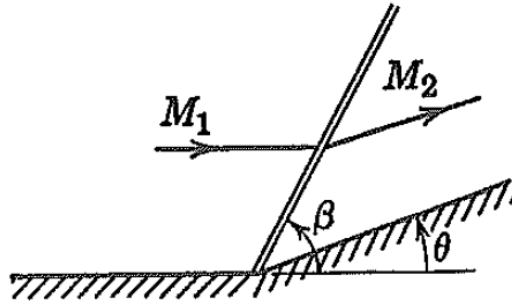


Figura 3.2: Escoamento supersônico passando por uma quina côncava e gerando um choque oblíquo, colado à superfície
(13)

$$\frac{P_2}{P_1} = 1 + \frac{2\gamma}{(\gamma + 1)} (\text{Ma}_1^2 \sin^2(\beta) - 1) \quad (3.5)$$

$$\text{Ma}_{n,1} = \text{Ma}_1 \sin(\beta) \quad (3.6)$$

$$\text{Ma}_{n,2} = \frac{\text{Ma}_{n,1}^2 + \left(\frac{2}{\gamma-1}\right)}{\left(\frac{2\gamma}{(\gamma-1)} \text{Ma}_{n,1}^2 - 1\right)} \quad (3.7)$$

$$\text{Ma}_2 = \frac{\text{Ma}_{n,2}}{\sin(\beta - \theta)} \quad (3.8)$$

Pode-se demonstrar que, para um choque oblíquo, a Eq. 3.9, conhecida como relação $\theta-\beta-\text{Ma}$, é válida. Essa equação é apresentada graficamente na Fig. 3.5. Nessa figura, nota-se que há duas soluções para cada número de Mach e ângulo de inclinação das linhas de corrente. A solução que se encontra na região inferior se refere ao choque fraco, cujo escoamento após o choque ainda é supersônico, e a que se encontra na região superior se refere ao choque forte, cujo escoamento após o choque se torna subsônico. Em escoamentos externos, considera-se o choque fraco. Nessa gráfico

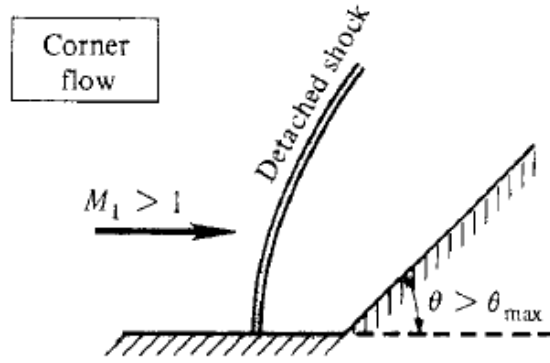


Figura 3.3: Escoamento supersônico passando por uma quina côncava e gerando um choque descolado da superfície

(13)

nota-se também que já um aumento de deflexão máximo para cada número de Mach de entrada. Caso o ângulo ultrapasse esse máximo, o choque se torna descolado.

$$\tan(\theta) = 2 \cot(\beta) \left[\frac{\text{Ma}_1 - 1}{\text{Ma}_1(\gamma + \cos(2\beta)) + 2} \right] \quad (3.9)$$

3.1.2 Equações de leque de expansão

Ao contrário do choque oblíquo, o leque de expansão ocorre quando se tem uma quina convexa. Além disso, ele ocorre de forma contínua e isentrópica. As linhas de corrente se defletem de forma suave, e o leque de expansão é formado por infinitas ondas de Mach, cujo formato se assemelha a um leque (Fig. 3.6).

Os ângulos iniciais e finais (μ_1 e μ_2) do leque de expansão são tais que o número de Mach da componente da velocidade normal ao choque na entrada e na saída, respectivamente, seja unitário. Assim, têm-se a Eq. 3.10.

$$\mu = \sin^{-1} \left(\frac{1}{\text{Ma}} \right) \quad (3.10)$$

Para o cálculo do escoamento antes e depois do leque de expansão, pode-se utilizar Eq. 3.11, baseada na teoria de expansão proposta por Prandtl e Meyer (1).

$$\theta = \nu(\text{Ma}_2) - \nu(\text{Ma}_1) \quad (3.11a)$$

$$\nu(\text{Ma}) = \left(\frac{\gamma + 1}{\gamma - 1} \right)^{\frac{1}{2}} \tan^{-1} \left(\left[\frac{\gamma - 1}{\gamma + 1} (\text{Ma}^2 - 1) \right]^{\frac{1}{2}} \right) - \tan^{-1} \left([\text{Ma}^2 - 1]^{\frac{1}{2}} \right) \quad (3.11b)$$

Como foi considerado que o leque de expansão é formado por ondas de Mach, tem-se que é um

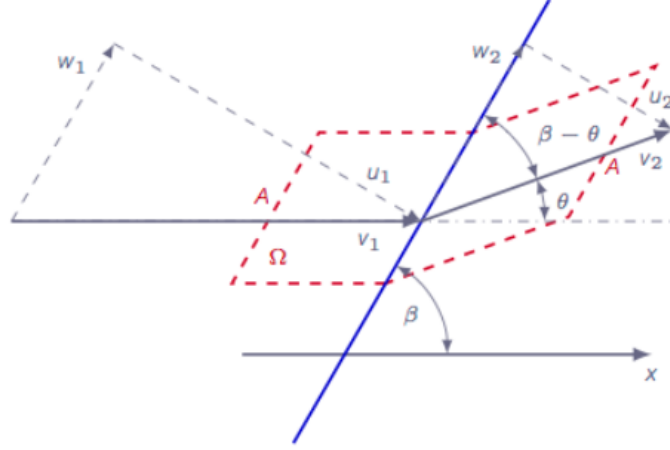


Figura 3.4: Decomposição das velocidades e determinação do volume de controle pelas linhas pontilhadas

(25)

fenômeno isentrópico. Com isso, pode-se mostrar que as propriedades após o leque de expansão podem ser calculadas a partir das Eq. 3.12 e 3.13.

$$\frac{P_2}{P_1} = \left(\frac{1 + \frac{(\gamma-1)}{2} \text{Ma}_1^2}{1 + \frac{(\gamma-1)}{2} \text{Ma}_2^2} \right)^{\frac{\gamma}{\gamma-1}} \quad (3.12)$$

$$\frac{T_2}{T_1} = \frac{1 + \frac{(\gamma-1)}{2} \text{Ma}_1^2}{1 + \frac{(\gamma-1)}{2} \text{Ma}_2^2} \quad (3.13)$$

3.1.3 Dados de entrada

Para realizar o cálculo analítico do escoamento supersônico invíscido sobre o aerofólio dupla cunha, primeiro é necessário fornecer os dados de geometria e do escoamento. A partir desses dados, analisa-se a ocorrência de choque oblíquo ou destacado e de leque de expansão no bordo de ataque do aerofólio. Caso ocorra destacamento, o cálculo é interrompido.

O cálculo do ângulo máximo de deflexão das linhas de corrente sem que ocorra destacamento é feito a partir da derivação da Eq. 3.9 com relação a β e igualando a zero, para encontrar máximo da função.

$$\frac{\partial \theta}{\partial \beta} = \frac{\partial}{\partial \beta} \left(\tan^{-1} (2 \cot(\beta)) \left[\frac{\text{Ma}_1^2 \sin(\beta)^2 - 1}{\text{Ma}_1^2 (\gamma + \cos(2\beta) + 2)} \right] \right) = 0 \quad (3.14)$$

Para encontrar o valor de β que zera a Eq. 3.14, utilizou-se o método de Newton-Raphson. Então a Eq. 3.9 é utilizada novamente para encontrar θ_{max}

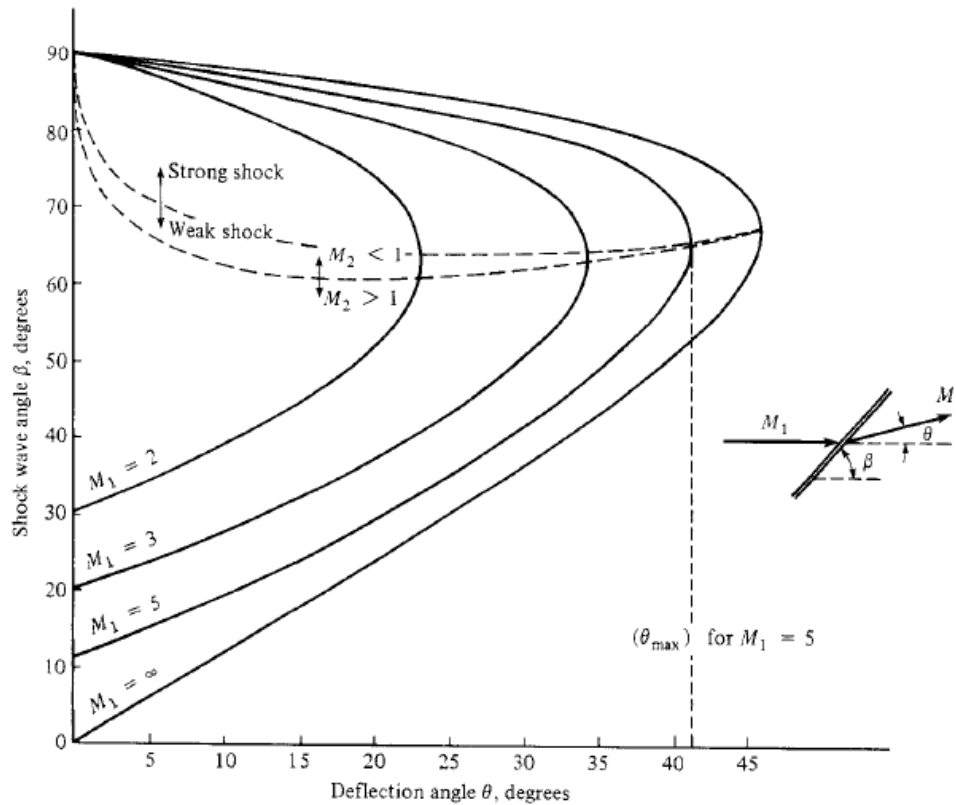


Figura 3.5: Gráfico da relação $\theta - \beta - Ma$. Para cada número de Mach existe um ângulo máximo de inclinação da superfície. Acima desse ângulo, o choque se torna destacado
(1)

3.1.4 Cálculo da onda de choque

Para o cálculo das ondas de choque, utilizou-se o método de Newton-Raphson na Eq. 3.9 para encontrar o ângulo do choque (β). A partir de então, calcula-se a pressão e o número de Mach após o choque pelas Eq. 3.5 a 3.8.

3.1.5 Cálculo do leque de expansão

Para o leque de expansão, utilizou-se o método de Newton-Raphson na Eq. 3.11 para encontrar o número de Mach após o choque, e então utilizou-se a Eq. 3.12 para encontrar a pressão na após o choque.

3.1.6 Dados de saída

Após a realização dos cálculos das ondas de choque oblíquas e dos leques de expansão, o programa entrega os valores dos coeficientes de sustentação e de arrasto no aerofólio. É utilizada como referência a velocidade do escoamento não perturbado

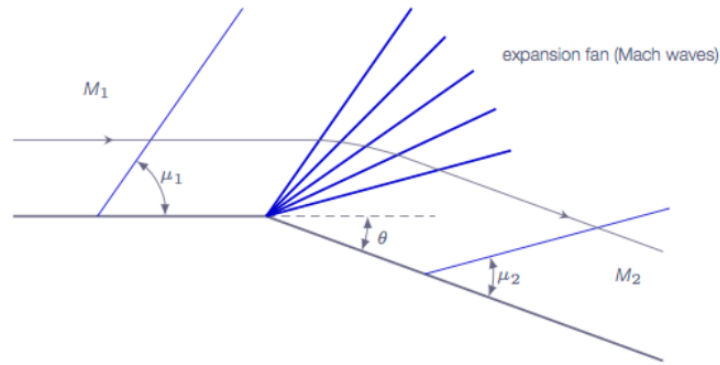


Figura 3.6: Representação do leque de expansão
(25)

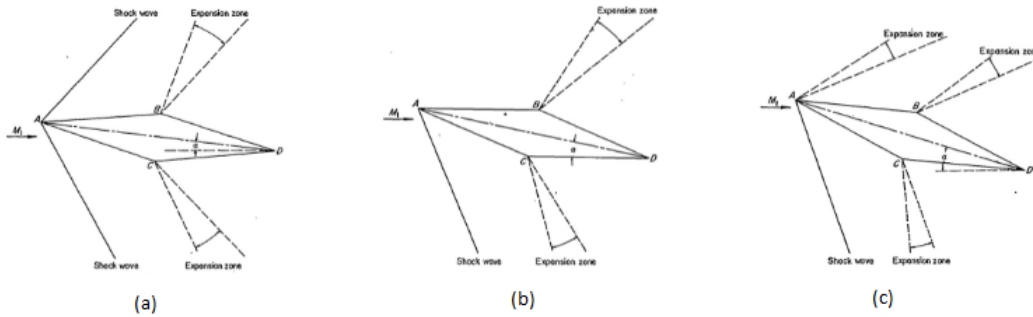


Figura 3.7: Escoamento supersônico sobre o aerofólio de dupla cunha em diferentes ângulos de ataque. a) choque no extra e intradorso, b) choque apenas no intradorso, c) choque no intradorso e leque de expansão no extradorso
(26)

3.2 Metodologia numérica do VAT

O *Virtual Aeroacoustic Tunnel* (VAT) é um código desenvolvido pelo CAALab e é baseado em volumes finitos e fronteira imersa, a qual utiliza malha cartesiana em que se impõe condições de contorno nos volumes que representam as fronteiras. A metodologia numérica é apresentada com base em (27).

3.2.1 Equações governantes no VAT

Utiliza-se as equações de Euler na forma não permanente.

$$\frac{\partial \rho}{\partial t} + \frac{\partial (\rho u_a)}{\partial x_a} = 0 \quad (3.15)$$

$$\frac{\partial (\rho u_a)}{\partial t} + \frac{\partial (\rho u_a u_b)}{\partial x_b} = -\frac{\partial P}{\partial x_a} + f_b \quad (3.16)$$

$$\frac{\partial (\rho E)}{\partial t} + \frac{\partial (\rho E u_a)}{\partial x_a} = -\frac{\partial (P u_a)}{\partial x_a} + f_a u_a \quad (3.17)$$

$$E = e + \frac{u_a u_a}{2} \quad (3.18)$$

Na equação 3.16 e 3.17, o termo f_a e $f_a u_a$ referentes à força de campo e ao trabalho dessa força sobre o escoamento, respectivamente, são utilizados para acelerar o escoamento do estado estacionário até a velocidade determinada na região de entrada (\mathbf{u}_∞). Isso é feito para melhorar a convergência da simulação nos estágios iniciais ao atenuar a variação abrupta da velocidade. A atuação da força de campo é limitada a um intervalo de tempo máximo.

Para o fechamento do sistema de Euler, utiliza-se as relações constitutivas a partir da hipótese de gás ideal (Eq. 3.19), caloricamente e termicamente perfeito (Eq. 3.20). O valor definido na condição de contorno do escoamento é considerado de referência, e a velocidade do som e o número de Mach de referência são calculados pelas Eq. 3.21 e 3.22.

$$P = \rho R T \quad (3.19)$$

$$e = c_v T \quad (3.20)$$

$$a_{ref} = \sqrt{\gamma R T_{ref}} \quad (3.21)$$

$$\text{Ma}_{ref} = \frac{u_{ref}}{a_{ref}} \quad (3.22)$$

A Eq. 3.19 pode ser reescrita, para um gás caloricamente e termicamente perfeito, na forma da equação 3.23 e 3.24.

$$P = (\gamma - 1) \rho e \quad (3.23)$$

$$T = \frac{\gamma \text{Ma}_{ref}^2 P}{\rho} \quad (3.24)$$

As Eq. 3.15, 3.16, 3.17, 3.23 e 3.24 são adimensionalizadas conforme as equações 3.25 .

$$x_a^* = \frac{x_a}{c}, t^* = \frac{t}{\frac{c}{u_{ref}}}, u_a^* = \frac{u_a}{u_{ref}}, P^* = \frac{P}{\rho_{ref} u_{ref}^2}, \rho^* = \frac{\rho}{\rho_{ref}}, T^* = \frac{T}{T_{ref}}, e^* = \frac{e}{u_{ref}^2}, f^* = \frac{f}{\frac{\rho_{ref} u_{ref}}{c}} \quad (3.25)$$

As equações 3.15 a 3.17 são organizadas pela Eq. 3.26.

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{E}}{\partial x} + \frac{\partial \mathbf{F}}{\partial y} + \frac{\partial \mathbf{G}}{\partial z} = \mathbf{R},$$

$$\mathbf{U} = \begin{bmatrix} \rho \\ \rho u_x \\ \rho u_y \\ \rho u_z \\ \rho E \end{bmatrix}, \mathbf{E} = \begin{bmatrix} \rho u_x \\ \rho u_x u_x + P \\ \rho u_x u_y \\ \rho u_x u_z \\ (\rho E + P) u_x \end{bmatrix}, \mathbf{F} = \begin{bmatrix} \rho u_y \\ \rho u_y u_x \\ \rho u_y u_y + P \\ \rho u_y u_z \\ (\rho E + P) u_y \end{bmatrix}, \mathbf{G} = \begin{bmatrix} \rho u_z \\ \rho u_z u_x \\ \rho u_z u_y \\ \rho u_z u_z + P \\ (\rho E + P) u_z \end{bmatrix}, \mathbf{R} = \begin{bmatrix} 0 \\ f_x \\ f_y \\ f_z \\ f_x u_x + f_y u_y + f_z u_z \end{bmatrix} \quad (3.26)$$

Pode-se definir um tensor $\underline{\Pi} = \mathbf{E} \otimes \mathbf{i} + \mathbf{F} \otimes \mathbf{j} + \mathbf{G} \otimes \mathbf{k}$, e dessa forma a Eq. 3.26 pode ser reescrita na forma da Eq. 3.27.

$$\frac{\partial \mathbf{U}}{\partial t} + \nabla \bullet \underline{\Pi} = \mathbf{R} \quad (3.27)$$

3.2.2 Esquema numérico

A Eq. 3.27 é integrada no volume de controle e, com a utilização do teorema da Gauss, tem-se a Eq. 3.29.

$$\int_V \frac{\partial \mathbf{U}}{\partial t} dV = - \int_S (\underline{\Pi} \bullet \mathbf{n}) dS + \int_V \mathbf{R} dV \quad (3.28)$$

Manipulando a integral do lado esquerdo e dividindo pelo volume, obtém-se a Eq. 3.29.

$$\frac{\partial}{\partial t} \left[\frac{1}{V} \int_V \mathbf{U} dV \right] = \frac{1}{V} \left[- \int_S (\underline{\Pi} \bullet \mathbf{n}) dS + \int_V \mathbf{R} dV \right] \quad (3.29)$$

Com auxílio da notação da média espacial (Eq. 3.30), a Eq. 3.29 é reescrita na forma da Eq. 3.31.

$$\bar{\mathbf{U}} = \frac{1}{V} \int_V \mathbf{U} dV, \bar{\mathbf{R}} = \frac{1}{V} \int_V \mathbf{R} dV \quad (3.30)$$

$$\frac{\partial \bar{\mathbf{U}}}{\partial t} = \frac{1}{V} \left[- \int_S (\underline{\Pi} \bullet \mathbf{n}) dS \right] + \bar{\mathbf{R}} \quad (3.31)$$

Utilizou-se a versão 2D do VAT, cuja malha é cartesiana regular, e com isso utiliza-se subscritos i e j para indicar um vértice de cada volume. A Eq. 3.31 precisa ser avaliada em cada volume de controle, e então é reescrita pela Eq. 3.32. Nessa equação, o termo $S_{i+\frac{1}{2}}$ indica a área da interface entre o volume (i,j) com o volume $(i+1,j)$, e os demais termos de área possuem notação análoga.

$$\left(\frac{\partial \bar{\mathbf{U}}}{\partial t}\right)_{i,j} = -\frac{1}{V_{i,j}} \left[\int_{S_{i+\frac{1}{2}}} (\underline{\Pi} \bullet \mathbf{n}) dS + \int_{S_{i-\frac{1}{2}}} (\underline{\Pi} \bullet \mathbf{n}) dS + \int_{S_{j+\frac{1}{2}}} (\underline{\Pi} \bullet \mathbf{n}) dS + \int_{S_{j-\frac{1}{2}}} (\underline{\Pi} \bullet \mathbf{n}) dS \right] + \bar{\mathbf{R}}_{i,j} \quad (3.32)$$

É feita um aproximação em que o tensor $\underline{\Pi}$ é considerado constante nas superfícies de cada volume, e a derivada temporal é avaliada em passo de tempo finito Δt . Utilizando a notação $\mathbf{S} = \int_S \mathbf{n} dS$, obtém-se a Eq. 3.33.

$$\left(\frac{\partial \bar{\mathbf{U}}}{\partial t}\right)_{i,j} \approx \frac{\Delta \bar{\mathbf{U}}}{\Delta t} \approx -\frac{1}{V_{i,j}} \left[(\underline{\Pi} \bullet \mathbf{S})_{i+\frac{1}{2}} + (\underline{\Pi} \bullet \mathbf{S})_{i-\frac{1}{2}} + (\underline{\Pi} \bullet \mathbf{S})_{j+\frac{1}{2}} + (\underline{\Pi} \bullet \mathbf{S})_{j-\frac{1}{2}} \right] + \bar{\mathbf{R}}_{i,j} \quad (3.33)$$

Da Eq. 3.33 obtém-se a função de fluxo $\mathcal{F}_{i,j}$, que será utilizada para o cálculo das variáveis de fluxo (Eq. 3.34). Também se define $\mathcal{R}_{i,j}$ na Eq. 3.35.

$$\mathcal{F}_{i,j} = -\frac{\Delta t}{V_{i,j}} \left[(\underline{\Pi} \bullet \mathbf{S})_{i+\frac{1}{2}} + (\underline{\Pi} \bullet \mathbf{S})_{i-\frac{1}{2}} + (\underline{\Pi} \bullet \mathbf{S})_{j+\frac{1}{2}} + (\underline{\Pi} \bullet \mathbf{S})_{j-\frac{1}{2}} \right] \quad (3.34)$$

$$\mathcal{R}_{i,j} = \Delta t \bar{\mathbf{R}}_{i,j} \quad (3.35)$$

A partir da Eq. 3.34 e 3.35, é possível reescrever a Eq. 3.33 na forma da Eq. 3.36. Nessa equação, o termo $\mathcal{D}_{i,j}$ é referente à dissipação artificial, e sua função é compensar os erros numéricos das aproximações feitas.

$$\Delta \bar{\mathbf{U}}_{,j} = -\mathcal{F}_{i,j} + \mathcal{D}_{i,j} + \mathcal{R}_{i,j} \quad (3.36)$$

Para calcular as variáveis primitivas, utiliza-se o método da Média de Favre (Eq. 3.37). Para avaliar essa variáveis nas faces dos volumes, é feita a interpolação indicada na Eq. 3.38, cujo desenvolvimento fora feito em (27). Nessa equação, ϕ pode assumir qualquer variável conservativa ou primitiva.

$$\begin{aligned} \tilde{u}_a &= \frac{\overline{\rho u_a}}{\bar{\rho}}, \tilde{E}_a = \frac{\overline{\rho E}}{\bar{\rho}} \\ \tilde{E} &= \tilde{e} + \frac{\tilde{u}_a \tilde{u}_a}{2}, \tilde{P} = (\gamma - 1) \bar{\rho} \tilde{e} \\ \tilde{T} &= \frac{\gamma \text{Ma}_{ref}^2 \tilde{P}}{\bar{\rho}} \end{aligned} \quad (3.37)$$

$$(\phi)_{i+\frac{1}{2}} = \frac{9}{16} (\phi_{i,j} + \phi_{i+1,j}) - \frac{1}{16} (\phi_{i-1,j} + \phi_{i+2,j}) \quad (3.38)$$

3.2.3 Dissipação artificial

O VAT utiliza como base a dissipação artificial proposta em (28) com adaptações feitas em (27). O primeiro passo é o cálculo das derivadas das propriedades nos vértices dos volumes de controle, e para isso utiliza-se o teorema da divergência na forma proposta em (7). Com isso, seja ϕ uma variável do escoamento, calcula-se a integral de $\nabla\phi$ sobre quatro volumes adjacentes ao vértice (i,j) e, com a utilização do teorema de Guass, obtém-se a Eq. 3.39, que define a função $\mathbf{f}(\phi)$. Segundo (27), essa integração proporciona quarta ordem de precisão no espaço.

$$\int_V (\nabla\phi) dV = \int_S \phi \bullet \mathbf{n} dS = \mathbf{f}(\phi) \quad (3.39)$$

O gradiente de ϕ no vértice (i,j) é dado como a média espacial do gradiente na região de integração (Eq. 3.40).

$$(\overline{\nabla\phi})_{i,j} = \frac{(\mathbf{f}(\phi))_{i,j}}{V_{i,j}} = \frac{(f_x(\phi)\mathbf{i} + f_y(\phi)\mathbf{j} + f_z(\phi)\mathbf{k})_{i,j}}{V_{i,j}} \quad (3.40)$$

As componentes de $\mathbf{f}(\phi)$ são calculados a partir dos valores das propriedades interpoladas para as faces com um esquema de interpolação diferente da Eq. 3.38, e que é apresentado na Eq. 3.41.

$$\phi_{i+\frac{1}{2},j} = \frac{1}{2} (\phi_{i,j} + \phi_{i+1,j}) \quad (3.41)$$

Uma vez obtidas as derivadas nos vértices dos volumes de controle, calcula-se as derivadas nas faces a partir da Eq. 3.42 . Os índices do termo à esquerda se referem à face e os índices dos termos à direita se referem aos vértices.

$$(\nabla\phi)_{i+\frac{1}{2},j} = \frac{1}{2} \left((\nabla\phi)_{i+1,j} + (\nabla\phi)_{i+1,j+1} \right) \quad (3.42a)$$

$$\left(\frac{\partial\phi}{\partial x_a} \right)_{i+\frac{1}{2},j} = \frac{1}{2} \left(\left(\frac{\partial\phi}{\partial x_a} \right)_{i+1,j} + \left(\frac{\partial\phi}{\partial x_a} \right)_{i+1,j+1} \right) \quad (3.42b)$$

Com as derivadas calculadas, é possível se obter a dissipação artificial por meio de três fatores: ψ_{div} , ψ_{prs} e ψ_{den} , conforme a Eq.3.43. Nessa equação é apresentado cálculo para a face $i + \frac{1}{2}$, mas a dissipação deve ser calculada para as demais faces também.

$$(\psi_{div})_{i+1/2} = K_{div}^{(2)} \bullet | \nabla \bullet \tilde{u} |_{i+\frac{1}{2}} \quad (3.43a)$$

$$(\psi_{prs})_{i+1/2} = K_{prs}^{(2)} \bullet | \nabla \tilde{P} |_{i+\frac{1}{2}} \quad (3.43b)$$

$$(\psi_{den})_{i+1/2} = K_{den}^{(2)} \bullet | \nabla \tilde{\rho} |_{i+\frac{1}{2}} \quad (3.43c)$$

Em seguida calcula-se os termos $\epsilon_{i+\frac{1}{2}}^{(2)}$ e $\epsilon_{i+\frac{1}{2}}^{(4)}$ pelas Eq. 3.44 e o termo $d_{i+\frac{1}{2}}$ pela Eq. 3.45. Por fim, calcula-se o termo $\mathcal{D}_{i,j}$ da Eq. 3.36 pela Eq.

$$\epsilon_{i+\frac{1}{2}}^{(2)} = K^{(2)} \max[\psi_{div}, \psi_{prs}, \psi_{den}] \quad (3.44a)$$

$$\epsilon_{i+\frac{1}{2}}^{(4)} = K^{(2)} \max\left[0, \left(K^{(4)} - \epsilon_{i+\frac{1}{2}}^{(2)}\right)\right] \quad (3.44b)$$

$$d_{i+\frac{1}{2}} = \epsilon_{i+\frac{1}{2}}^{(2)} [\bar{\mathbf{U}}_{i+1,j} - \bar{\mathbf{U}}_{i,j}] - \epsilon_{i+\frac{1}{2}}^{(4)} [\bar{\mathbf{U}}_{i+2,j} - \bar{\mathbf{U}}_{i+1,j} + \bar{\mathbf{U}}_{i,j} - \bar{\mathbf{U}}_{i-1,j}] \quad (3.45)$$

$$\mathcal{D}_{i,j} = \left(d_{i+\frac{1}{2}} - d_{i-\frac{1}{2}}\right) + \left(d_{j+\frac{1}{2}} - d_{j-\frac{1}{2}}\right) \quad (3.46)$$

Os parâmetros $K^{(2)}$ e $K^{(4)}$ são coeficientes de ajuste da função de dissipação artificial, e dependem do caso a ser simulado. É necessário arbitrar seus valores através de tentativas até que a simulação convirja.

3.2.4 Fronteira Imersa

Para aplicação das condições de fronteira, os valores das propriedades dos volumes de fronteira são impostos diretamente em cada iteração. No aerofólio utilizou-se a condição de impenetrabilidade, parede adiabática e fator de escorregamento de 100% (Eq. ??).

A cada volume de controle de fronteira é associado um vetor unitário normal $\mathbf{n} = n_x \mathbf{i} + n_y \mathbf{j}$ a depender da geometria e, com base nos valores dessas componentes, são calculadas as variáveis do escoamento nos volumes de fronteira. Para o caso de $n_x > 0$, a derivada da variável genérica ϕ é calculada pela Eq. 3.47. Pela referência (27), esse método proporciona quarta ordem de precisão espacial e se baseia no ajuste polinomial proposto em (29).

$$\left(\frac{\partial \phi}{\partial x}\right)_{i,j}^b = \frac{1}{12\Delta x} \left[-25\phi_{i,j}^b + 48\phi_{i+1,j} - 36\phi_{i+2,j} + 16\phi_{i+3,j} - 3\phi_{i+4,j} + \mathcal{O}(\Delta X)^4\right] \quad (3.47)$$

A Eq. 3.47 pode ser reescrita na forma da Eq. 3.48. Para outras direções e sentido dos componentes do vetor normal, basta trocar o sinal. O valor de $\phi_{i,j}^b$ é então calculado por média ponderada (Eq. 3.49).

$$\left(\frac{\partial \phi}{\partial x}\right)_{i,j}^b = \frac{25}{12\Delta x} \left[-\phi_{i,j}^b + D_i^+ \phi + \mathcal{O}(\Delta X)^4\right] \quad (3.48a)$$

$$D_i^+ \phi = \frac{1}{25} (48\phi_{i+1,j} - 36\phi_{i+2,j} + 16\phi_{i+3,j} - 3\phi_{i+4,j}) \quad (3.48b)$$

$$\phi_{i,j}^b = \frac{|n_x| D_i \phi + |n_y| D_j \phi}{|n_x| + |n_y|} \quad (3.49)$$

A velocidade no aerofólio pode ser decomposta nas componentes tangencial e normal à superfície. Porém, como foi utilizada a condição de impenetrabilidade, a componente normal é nula, e portanto tem-se a Eq.3.50.

$$\tilde{\mathbf{u}}_{i,j}^b = (\tilde{\mathbf{u}}_t)_{i,j}^b \quad (3.50)$$

3.2.5 Marcha Temporal

A marcha temporal é obtida pelo método de Runge Kutta de terceira ordem (30). A sua execução se dá pela Eq. 3.51.

$$\bar{\mathbf{U}}_{i,j}^{(1)} = \bar{\mathbf{U}}_{i,j}^{n-1} - \left(\mathcal{F}_{i,j}^{n-1} - \mathcal{D}_{i,j}^{n-1} - \mathcal{R}_{i,j}^{n-1} \right) \quad (3.51a)$$

$$\bar{\mathbf{U}}_{i,j}^{(2)} = \frac{3}{4} \bar{\mathbf{U}}_{i,j}^{n-1} + \frac{1}{4} \bar{\mathbf{U}}_{i,j}^{(1)} - \left(\mathcal{F}_{i,j}^{(1)} - \mathcal{D}_{i,j}^{(1)} - \mathcal{R}_{i,j}^{(1)} \right) \quad (3.51b)$$

$$\bar{\mathbf{U}}_{i,j}^n = \frac{1}{3} \bar{\mathbf{U}}_{i,j}^{n-1} + \frac{2}{3} \bar{\mathbf{U}}_{i,j}^{(2)} - \frac{2}{3} \left(\mathcal{F}_{i,j}^{(2)} - \mathcal{D}_{i,j}^{(2)} - \mathcal{R}_{i,j}^{(2)} \right) \quad (3.51c)$$

3.3 Metodologia numérica do OpenFOAM

O OpenFOAM é um *software* baseado no método de volumes finitos, é distribuído sob a licença *GNU* e abrange um amplo leque de aplicações na dinâmica dos fluidos computacional. As equações governantes são solucionadas uma de cada vez para uma variável considerada dominante por meio de iterações que são realizadas até que se atinja o limite de iterações ou o mínimo estabelecido para a variação entre os resultados (resíduo). Além disso, utiliza-se métodos implícitos, de forma que o resultado de um determinado tempo de simulação passo de tempo seja obtido não somente a partir do resultado do passo de tempo anterior. Isso permite aumentar o *Courant* acima da unidade, porém acarreta em imprecisão temporal, e por isso é utilizado preferencialmente em problemas de solução permanente.

A configuração da simulação é feita a partir de arquivos de texto denominados *dictonaries*, com os quais é possível definir, entre outros, a condição de contorno, o algoritmo de solução iterativa das matrizes e o método de solução do sistema de equações. Apesar da documentação do OpenFOAM ser reduzida, uma descrição básica do método numérico é apresentado a seguir, em que foi utilizado como base a referência (17). A notação utilizada a seguir é baseada na Fig. 3.8.

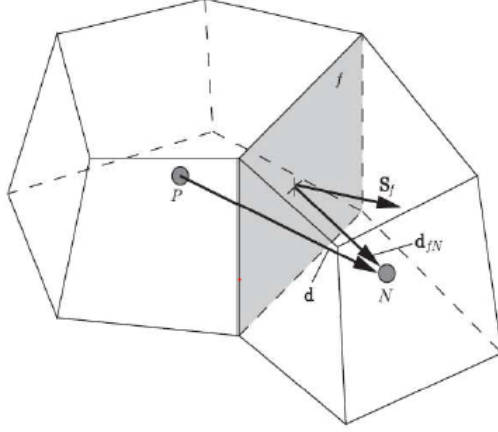


Figura 3.8: Volume de controle do OpenFOAM. Na figura, p representa o volume de controle considerado, N representa o centro de um volume de controle vizinho, d é a distância entre os dois, \mathbf{S}_f é o vetor da área da interface entre os dois volumes e \mathbf{d}_{fn} é a distância entre o centro da área da interface e o centro do volume vizinho

$$(17)$$

3.3.1 Equações governantes

As equações utilizadas no OpenFOAM são as equações de Navier-Stokes, que podem ser generalizadas na forma da Eq. 3.52, obtida pela aplicação do Teorema do Transporte de Reynolds a um escalar ϕ (9).

$$\underbrace{\frac{\partial \rho \phi}{\partial t}}_{\text{acumulo}} + \underbrace{\nabla \cdot (\rho \mathbf{u} \phi)}_{\text{advectivo}} = \underbrace{\nabla \cdot (\Gamma^\phi \nabla \phi)}_{\text{difusivo}} + \underbrace{Q_\phi}_{\text{fonte}} \quad (3.52)$$

3.3.2 Esquema numérico

A Eq. 3.52 é integrada sobre o volume de controle com centro em P (Fig. 3.8) e também é integrada em relação ao tempo, resultando na Eq. 3.53.

$$\int_t^{t+\Delta t} \left(\frac{\partial}{\partial t} \int_{V_p} \rho \phi dV + \int_{V_p} \nabla \cdot [\rho \mathbf{u} \phi] dV \right) dt = \int_t^{t+\Delta t} \left(\int_{V_p} \nabla \cdot [\Gamma^\phi \nabla \phi] dV + \int_{V_p} Q_\phi dV \right) dt \quad (3.53)$$

Utilizando o teorema de Gauss, obtém-se a Eq. 3.54, em que \mathbf{S}_p é o vetor da superfície do volume de controle.

$$\int_t^{t+\Delta t} \left(\frac{\partial}{\partial t} \int_{V_p} \rho \phi dV + \int_{S_p} [\rho \mathbf{u} \phi] \cdot d\mathbf{S}_p \right) dt = \int_t^{t+\Delta t} \left(\int_{S_p} [\Gamma^\phi \nabla \phi] \cdot d\mathbf{S}_p + \int_{V_p} Q_\phi dV \right) dt \quad (3.54)$$

3.3.2.1 Termo de acúmulo

O termo de acúmulo pode ser representado pela Eq.3.55, em que considera-se $\int_{V_p} \rho\phi(\mathbf{x}, t)dV dt = \rho\phi_p$, assumindo que ϕ varia linearmente, e acarretando em segunda ordem de precisão no tempo. A aproximação da integração temporal dos demais termos é feita pelo método de Euler implícito.

$$\int_t^{t+\Delta t} \left(\frac{\partial}{\partial t} \int_{V_p} \rho\phi(\mathbf{x}, t)dV \right) dt = (\rho\phi_p^n - \rho\phi_p^{n-1}) V_p \quad (3.55)$$

3.3.2.2 Termo advectivo

O termo advectivo é aproximado pelo somatório dos fluxos nas faces do volume de controle, com a aproximação de que as variáveis do escoamentos sejam contantes nas faces (Eq. 3.56), o que leva à segunda ordem de precisão, por se considerar uma variação linear. Nessa equação, $F_f = \mathbf{S}_f \bullet \mathbf{u}_f$ é o fluxo volumétrico através das faces.

$$\int_{S_p} (\rho\mathbf{u}\phi) \bullet d\mathbf{S}_p \approx \sum_f \mathbf{S}_f \bullet \mathbf{u}_f \phi_f = \sum_f F_f \phi_f \quad (3.56)$$

Os valores de ϕ_f e F_f são obtidos por interpolação a partir dos valores nos centroides dos volumes de controle vizinhos. Um método possível é o esquema de diferenças centradas, que assume variação linear entre os centroides p e N (Fig. 3.8 e Eq. 3.57).

$$\phi_f = \omega_f \phi_p + (1 - \omega_f) \phi_N \quad (3.57a)$$

$$\omega_f = \frac{|\mathbf{S}_f \bullet \mathbf{d}_f|}{\mathbf{S}_f \bullet \mathbf{d}} \quad (3.57b)$$

Outro método possível é o esquema *upwind*, que assume que ϕ é constante no volume de controle e a interpolação é feita com base na direção do fluxo (Eq. 3.58). Esse método é dito ser *bounded* e de primeira ordem (31).

$$\phi_f = \phi_p, \text{ se } F_f \geq 0 \quad (3.58a)$$

$$\phi_f = \phi_N, \text{ se } F_f < 0 \quad (3.58b)$$

O método utilizado é o esquema *flux-limited*, que é obtido pela mistura do esquema de diferenças centrais e *upwind* (Eq. 3.59). Nessa equação, ω_f é obtido pela Eq. 3.57b, $(\nabla\phi)_p = \frac{1}{V_p} \sum_f \mathbf{S}_f \phi_f$ é o gradiente no centroide do volume de controle e $(\nabla_{\mathbf{d}}\phi)_f = \phi_N - \phi_p$ é o gradiente avaliado na face, na direção de \mathbf{S}_f e multiplicado por $|\mathbf{d}|$. O termo ϕ_f nos gradientes são calculados pela Eq. 3.57.

$$\phi_f = (1 - \psi[1 - \omega_f])\phi_p + \psi(1 - \omega_f)\phi_N \quad (3.59a)$$

$$\psi = \psi(r) = \frac{r + |r|}{1 + r} \quad (3.59b)$$

$$r = 2 \frac{\mathbf{d} \bullet (\nabla \phi)_p}{(\nabla \mathbf{d} \phi)_f} \quad (3.59c)$$

3.3.2.3 Termo difusivo

De maneira análoga, o termo difusivo pode ser escrito pela Eq. 3.60a. O termo Γ_f^ϕ é a difusividade avaliada nas faces do volume a partir do método de diferenças centrais (Eq. 3.57). Já o termo $\mathbf{S}_f \bullet (\nabla \phi)_f$ é o gradiente normal à superfície, avaliado na superfície, e é calculado pela Eq. 3.60b, em que $\mathbf{n} = \frac{\mathbf{S}_f}{|\mathbf{S}_f|}$, e Θ é o ângulo entre \mathbf{n} e \mathbf{d} . O esquema utilizado leva à segunda ordem de precisão pela hipótese de variação linear das variáveis do escoamento.

$$\int_{S_p} (\Gamma^\phi \nabla \phi) \bullet d\mathbf{S}_p \approx \sum_f \Gamma_f^\phi \mathbf{S}_f \bullet (\nabla \phi)_f \quad (3.60a)$$

$$\mathbf{S}_f \bullet (\nabla \phi)_f = |\mathbf{S}_f| \left(\chi \overbrace{\frac{\phi_p - \phi_N}{|\mathbf{d}|}}^{\text{implicito}} + \overbrace{(\mathbf{n} - \chi \mathbf{d}) \bullet (\nabla \phi)_f}^{\text{explicito}} \right) \quad (3.60b)$$

$$\chi = \frac{1}{\cos \Theta} \quad (3.60c)$$

3.3.2.4 Termo de fonte

Um termo de fonte generalista Q_ϕ é aproximado pela Eq. 3.61. Q_ϕ é geralmente uma função não linear de ϕ .

$$\int_{V_p} Q_\phi dV \approx Q_{\phi p} V_p \quad (3.61)$$

Para melhorar a estabilidade do código, esse termo pode ser linearizado, de acordo com (31). Dessa forma, o termo de fonte pode ser dividido em um termo contante $Q_{\phi, const}$ e um coeficiente multiplicado por ϕ (Eq. 3.62). A integração volumétrica ficará como a Eq. 3.63.

$$Q_\phi = Q_{\phi, const} + Q_{\phi, slope} \phi \quad (3.62)$$

$$\int_{V_p} Q_\phi dV \approx Q_{\phi, const} V_p + Q_{\phi, slope} \phi_p V_p \quad (3.63)$$

3.3.3 Método de solução do sistema de equações linear

Combinando todos os tratamentos mostrados, a equação do transporte de ϕ para um volume de controle resulta na forma algébrica da Eq. 3.64. Os termos a_p e a_N são os coeficientes das equações algébricas.

$$a_p \phi_p^n + \sum_f a_N \phi_N^n = C_p \quad (3.64)$$

Na Eq. 3.64, os termos conhecidos são incluídos no lado direito, e os termos desconhecidos são incluídos no lado esquerdo. Dessa forma, as equações aplicadas a todos os volumes de controle formam o sistema da Eq. 3.65.

$$[\mathcal{A}][\phi] = [\mathcal{B}] \quad (3.65)$$

A matriz \mathcal{A} é grande e dispersa. Além disso, no presente caso, as equações são acopladas, sendo necessário solucioná-las simultaneamente. Isso requer muito poder computacional, e para contornar esse problema, o sistema é solucionado de maneira sequencial, em que cada equação governante é tratada implicitamente para a variável transportada e o restante é tratado explicitamente. O método utilizado para a solução do sistema algébrico é baseado no método Gauss-Seidel, com exceção da solução da Eq. 3.66, que é tratada explicitamente em um método direto.

3.3.4 Algoritmo de solução do sistema de equações

A solução do escoamento é obtida pela solução sequencial das equações governantes. Isso introduz um *lag* na solução, o que requer a inclusão de novas iterações dentro de um mesmo passo de tempo para a correção. O algoritmo utilizado foi o *sonicFoam*, cuja descrição é apresentada abaixo.

1. Inicialização das variáveis dependentes ($\rho^{(0)}$, $\mathbf{u}^{(0)}$, $P^{(0)}$ e $e^{(0)}$) e do fluxo de massa $\mathcal{M}_f^{(0)} = \mathbf{S}_f \bullet (\rho^{(0)} \mathbf{u}^{(0)})_f$

2. Início do laço de tempo

- (a) Utilização da equação da continuidade para se obter a massa específica inicial $\rho_p^{(init)}$

$$V_p \frac{\rho_p^{(init)} - \rho_p^{(0)}}{\Delta t} + \sum_f \mathcal{M}_f^{(0)} = 0 \quad (3.66)$$

- (b) Utilização da equação da quantidade de movimento para se obter a velocidade $\mathbf{u}^{(init)}$ aproximada nas duas ou três direções consideradas

$$\begin{aligned}
V_p \frac{\rho_p^{(init)} \mathbf{u}_p^{(init)} - \rho_p^{(0)} \mathbf{u}_p^{(0)}}{\Delta t} + \sum_f \mathcal{M}_f^{(0)} \mathbf{u}_f^{(init)} - \sum_f \mu_f^{(0)} \mathbf{S}_f \bullet (\nabla \mathbf{u})_f^{(init)} \\
- \sum_f \mu_f^{(0)} \mathbf{S}_f \bullet \left[(\nabla \mathbf{u})^T - \frac{2}{3} \nabla \bullet \underline{\mathbf{u}} \right]_f^{(0)} = - \sum_f \mathbf{S}_f P_f^{(0)} \quad (3.67)
\end{aligned}$$

- (c) Utilização da equação da energia sem o termo referente ao trabalho sobre o escoamento para se obter a energia interna $e^{(n)}$

$$V_p \frac{\rho_p^{(init)} e^{(n)} - \rho_p^{(0)} e_p^{(0)}}{\Delta t} + \sum_f \mathcal{M}_f^{(0)} e_f^{(n)} - \sum_f \left(k_f^{(0)} / C_v \right) \mathbf{S}_f \bullet (\nabla e)_f^{(n)} = - \sum_f \left(\mathcal{M}_f^{(0)} / \rho_f^{(n)} \right) P_f^{(0)} \quad (3.68)$$

- (d) A temperatura $T^{(n)}$ é atualizada a partir de $e^{(n)}$, a viscosidade $\mu^{(n)}$ é atualizada a partir de $T^{(n)}$ pela lei de Sutherland e o coeficiente de condutividade térmica $k^{(n)}$ é atualizado a partir de $T^{(n)}$.

Nesse momento a velocidade e a pressão não solucionam simultaneamente a equação da continuidade e da quantidade de movimento, e por isso são necessárias mais iterações para correção desses valores. Isto é feito a partir do algoritmo conhecido como *Pressure-implicit with splitting of operations* (PISO) (32,33).

- (e) início do PISO

- i. Atualizar a massa específica $\rho^{(n)}$ pela equação do estado, $P_p^{(n-1)} = \rho_p^{(n)} R T_p^{(n)}$
- ii. Calcular a velocidade intermediária \mathbf{u}^* pela equação do momento sem o gradiente de pressão e a partir de $\mathbf{u}^{(n-1)}$ (quando for a primeira iteração PISO, utiliza-se $\mathbf{u}^{(init)}$)

$$\begin{aligned}
V_p \frac{\rho_p^{(n)} \mathbf{u}_p^{(*)} - \rho_p^{(n-1)} \mathbf{u}_p^{(n-1)}}{\Delta t} + \sum_f \mathcal{M}_f^{(n-1)} \mathbf{u}_f^{(*)} - \sum_f \mu_f^{(n-1)} \mathbf{S}_f \bullet (\nabla \mathbf{u})_f^{(*)} \\
- \sum_f \mu_f^{(n-1)} \mathbf{S}_f \bullet \left[(\nabla \mathbf{u})^T - \frac{2}{3} \nabla \bullet \underline{\mathbf{u}} \right]_f^{(n-1)} = 0 \quad (3.69)
\end{aligned}$$

- iii. Calcular o fluxo volumétrico multiplicado por $\frac{1}{RT}$,

$$F_f^{(*)} = \frac{1}{RT_f^{(n)}} \mathbf{u}_f^{(*)} \bullet \mathbf{S}_f \quad (3.70)$$

- iv. Uma equação para a pressão do tipo Poisson é obtida pela substituição de $\mathbf{u}_p = \mathbf{u}_p^{(*)} - \frac{1}{a_p} (\nabla P)_p$ na equação da continuidade. O termo a_p vem da Eq. 3.64. A Eq.

3.71 é solucionada para P^n .

$$V_p \frac{\frac{1}{RT_p^{(n)}} P_p^{(n)} - \frac{1}{RT_p^{(n-1)}} P_p^{(n-1)}}{\Delta t} + \sum_f F_f^{(*)} P_f^{(n)} - \sum_f \frac{\rho_f^{(n)}}{a_p} \mathbf{S}_f \cdot (\nabla P)_f^{(n)} = 0 \quad (3.71)$$

v. Corrigir o campo de velocidade com a pressão corrigida no passo anterior, $\mathbf{u} = \mathbf{u} - \frac{1}{a_p} (\nabla P)^{(n)}$

(f) Repetir o PISO até atingir o critério de convergência

3. Próximo passo de tempo até o limite de tempo estabelecido

3.4 Configuração do *cluster* do CAALab



Esse cluster possui 1 nó de gerência conectado à internet com 13 TB de HD, 16 cores Intel Xeon e 64GB de memória RAM. A rede conecta-se ao cluster por cabos *infiniband*, que permitem a passagem de 56 Gb/s. Também possui 8 nós de cálculo, cada um composto por 1 processador Intel Xeon-Phi Knights Landing de 64 núcleos, cada um com 2 VPU's de 8 operações de ponto flutuante por ciclo, e com 16GB de MCDRAM, 96 GB de RAM DDR4/5 e 1TB de HD. Por fim, há também 1 nó de visualização, com 32 núcleos Intel Xeon, 256GB de RAM e 1 TB de memória SSD. Cada core do processador Knights Landing pode alocar 4 threads. Dessa forma, as simulações podem ser divididas em até 256 processos paralelos em cada nó.

3.5 Campanha de simulações

3.5.1 Aerofólio e asa estudados

Para realizar as simulações bidimensionais, foi considerado um aerofólio dupla cunha, conforme a Fig. 3.9 e a Tab. 3.1. Optou-se por ele pois o cálculo analítico do escoamento supersônico é simplificado pela sua simetria.

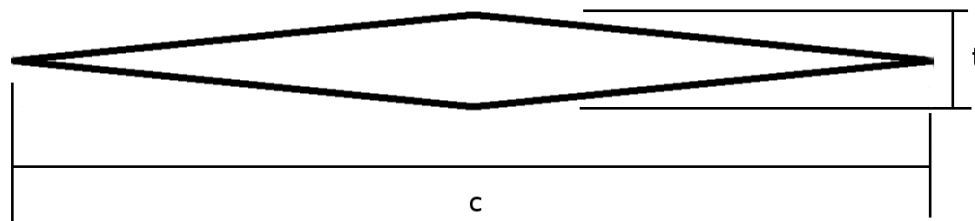


Figura 3.9: Aerofólio de dupla cunha

Tabela 3.1: Dimensões utilizadas no aerofólio de dupla cunha

	t (mm)	c (mm)	posição da espessura máxima
VAT	52.4078	1000	50% da corda
OpenFOAM	100.0000	1000	50% da corda

Para realizar as simulações tridimensionais, foram usadas duas asas da referência(12). Essas asas possuem perfil dupla cunha e foram testadas em túnel de vento supersônico.

Tabela 3.2: Parâmetros geométricos das asas tridimensionais simuladas

Asa	M.A.C. (m)	b (m)	cr (m)	XC	YC	ϵ (graus)
a	0.078334	0.124663	0.117653	0.021178	0.009412	27.914318
b	0.046634	0.141122	0.070104	0.012619	0.005608	45.186226

3.5.2 Malha computacional do OpenFOAM

A malha estática bidimensional utilizada no OpenFOAM foi gerada pelo *software Salomé*, que também é de código aberto, sob a licença GNU. A partir a geometria do aerofólio estudado, foram feitas duas regiões: uma região circular refinada próxima ao aerofólio e outra região quadrada com elementos não refinados distantes do aerofólio. Os elementos utilizados foram triangulares, com

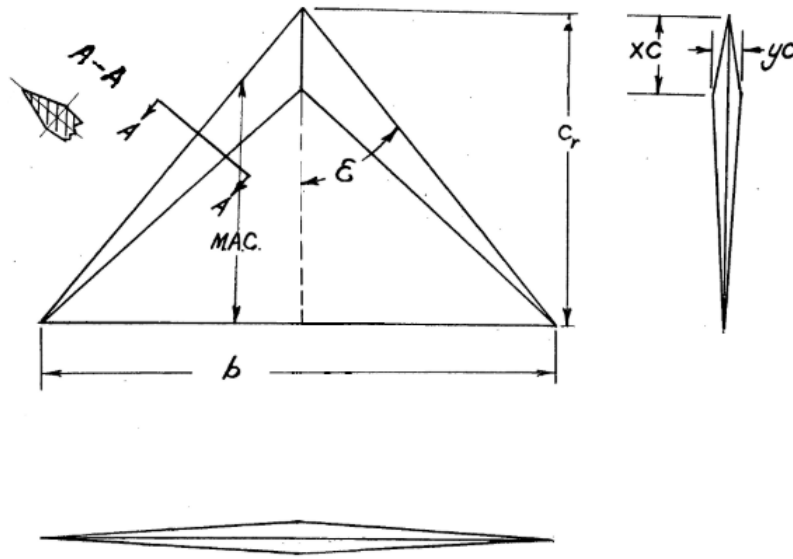


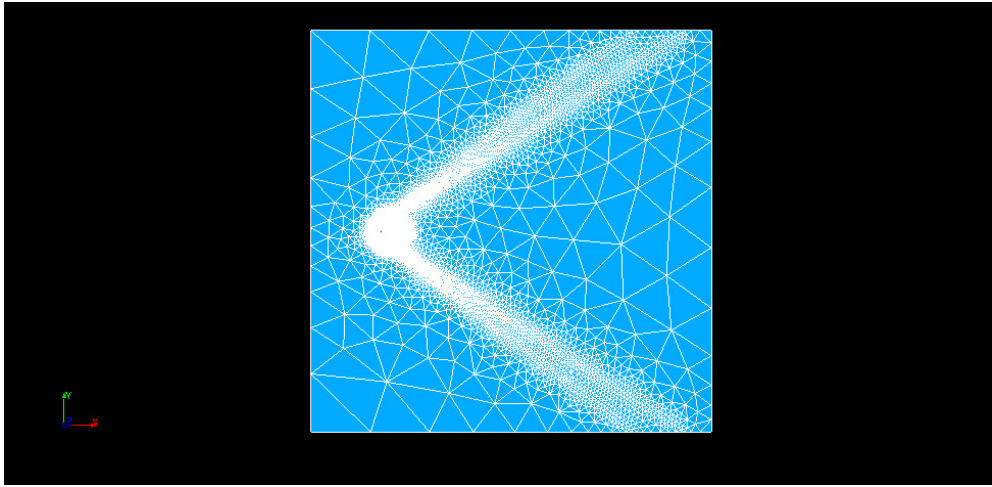
Figura 3.10: Vista projetada da asa submetida à simulação numérica (12, adaptado)

exceção dos elementos próximo à superfície, que são retangulares. Por ser melhor prática, fez-se a malha no formato externo quadrilátero, para melhor aplicação das condições de contorno, e feito refinamento na região próxima ao aerofólio e um refinamento ainda maior na região da onda de choque. Foram feitas três malhas para analisar a convergência de malha, e elas são mostradas nas figuras 3.11 a 3.13. Para melhorar a conformação da malha, trancou-se o aerofólio nas pontas. Esse truncamento possui tamanho de 1 a 3 volumes de controle, a depender da malha.

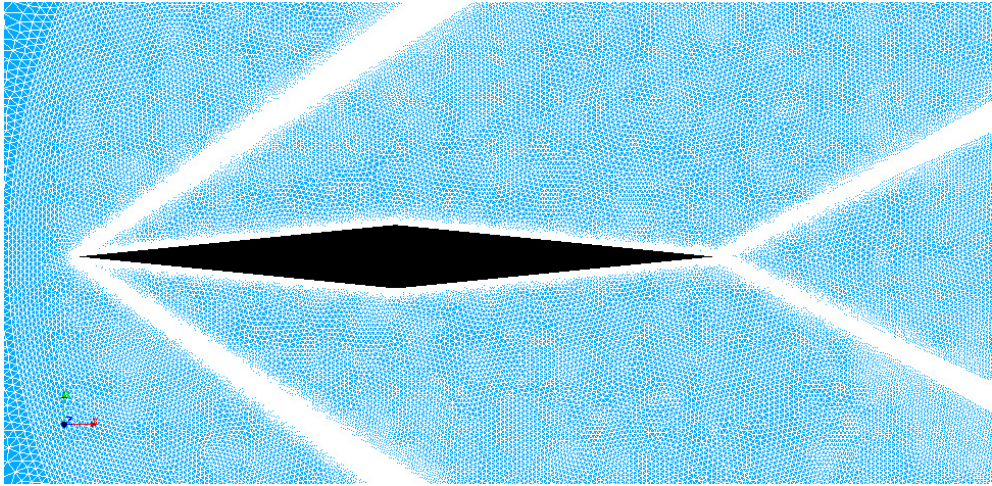
Tabela 3.3: Informações das malhas estáticas bidimensionais sobre o aerofólio de dupla cunha

Malha	Número de volumes	Número de volumes sobre o aerofólio	Comprimento do volume sobre o aerofólio dividido pela corda
A	524660	1679	0.119%
B	1009021	3345	0.059%
C	2530875	6675	0.030%

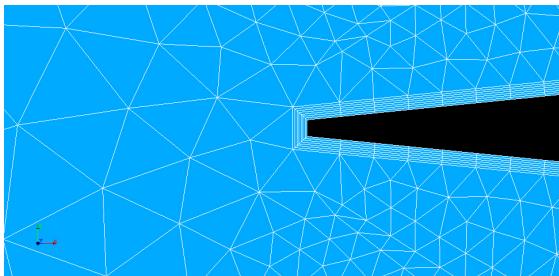
Também foram feitas simulações com malha adaptativa. Nessa simulações, iniciou-se com um domínio de 3 aerofólios no comprimento por 2 na largura. A malha inicial foi formada por hexágonos, e à medida que o escoamento desenvolveu-se, o mecanismo de refinamento automático selecionou os elementos da geometria que possuíam altos gradientes e posteriormente os refinava em 8 novos elementos hexagonais. O processo de refinamento é tridimensional, porém a simulação foi feita com as variáveis apenas bidimensionais.



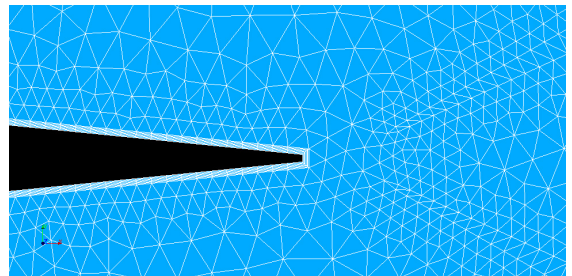
(a)



(b)

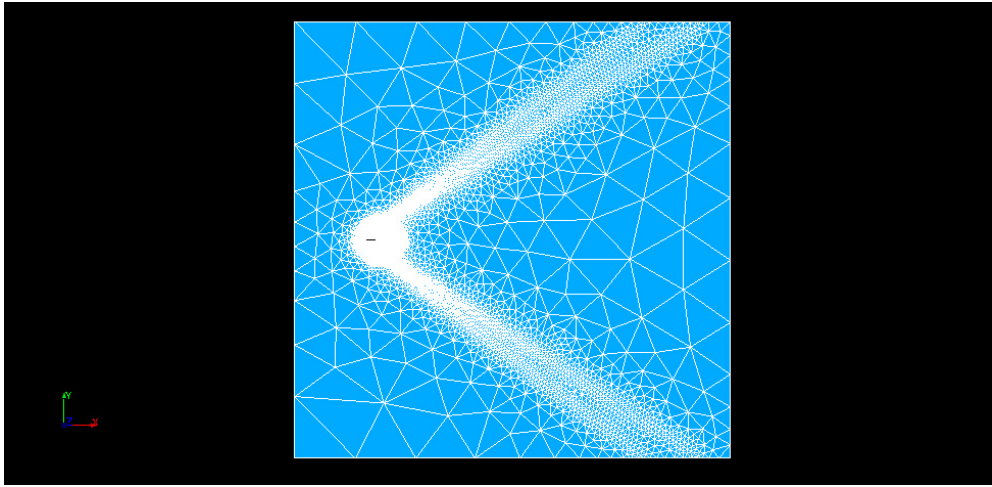


(c)

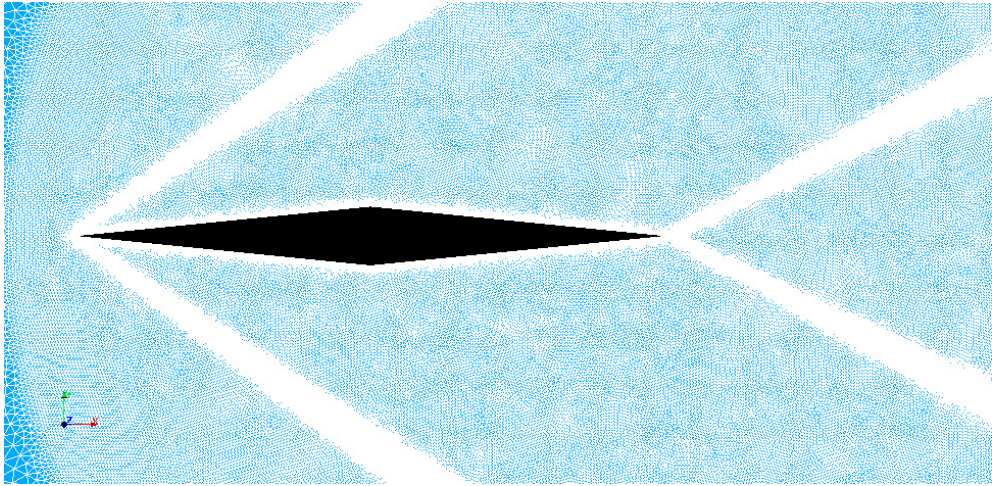


(d)

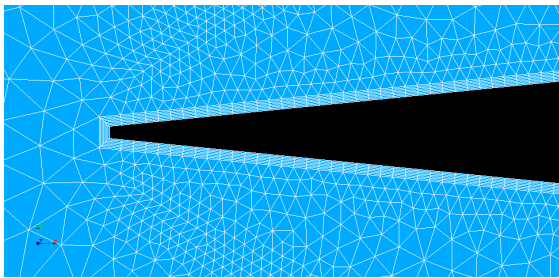
Figura 3.11: Malha estática bidimensional A do aerofólio de dupla cunha



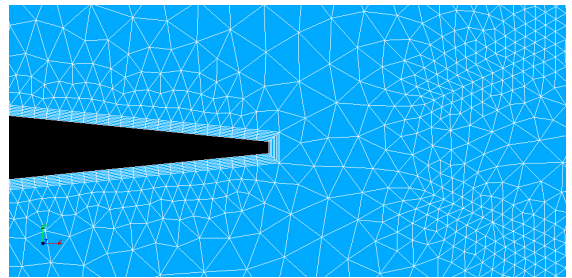
(a)



(b)

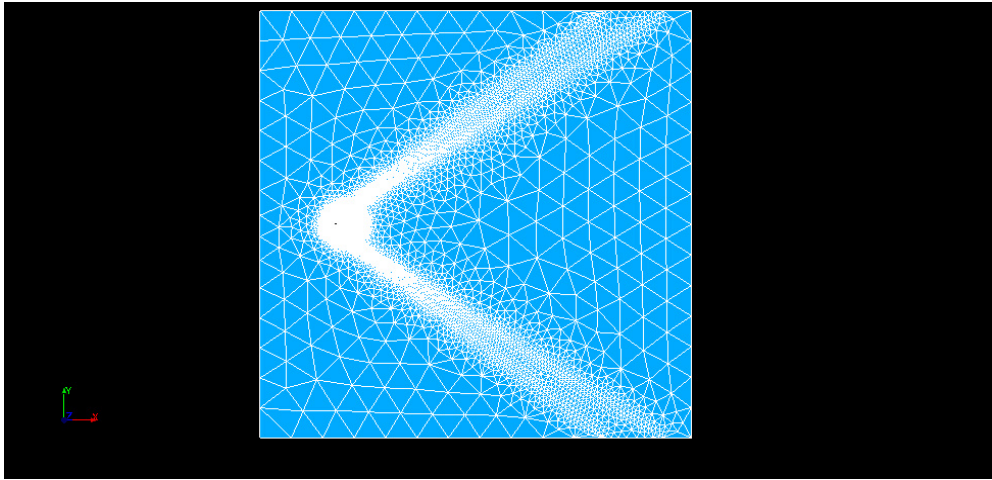


(c)

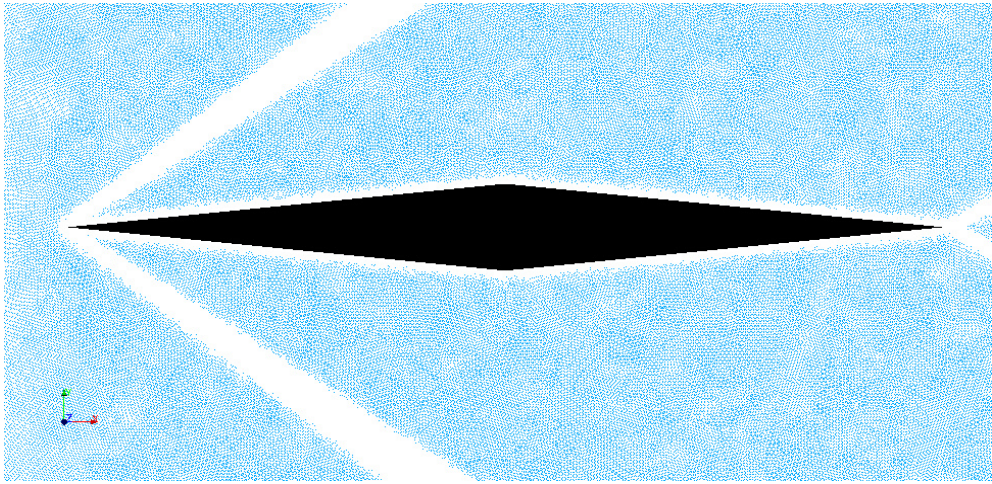


(d)

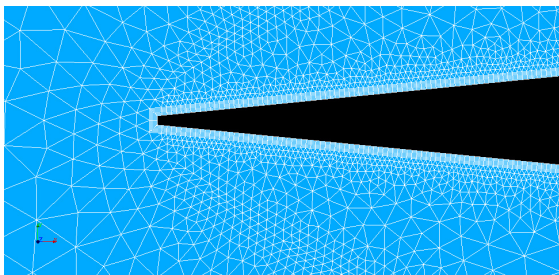
Figura 3.12: Malha estática bidimensional B do aerofólio de dupla cunha



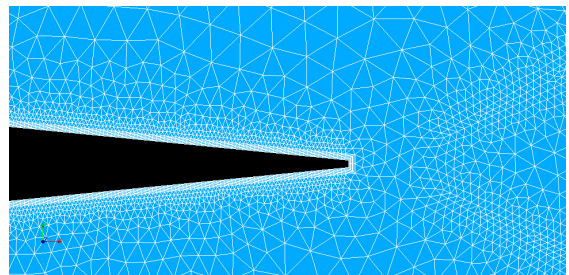
(a)



(b)



(c)



(d)

Figura 3.13: Malha estática bidimensional C do aerofólio de dupla cunha

Já para a simulação da asa tridimensional, utilizou-se somente a malha adaptativa com base no gradiente da velocidade. Configurou-se um intervalo de gradiente para que o refinamento fosse aplicado apenas no choque. A malha inicial possui 20 aerofólios na direção da sustentação, 5 aerofólios na direção normal ao plano do arrasto e sustentação e 4 aerofólios na direção do arrasto, e era formada por hexágonos. Foi utilizado o código *snappyHexMesh* e *blockMesh* do OpenFOAM para a realização da malha inicial.

3.5.3 Malha computacional do VAT

A malha computacional utilizada no VAT possui uma região cartesiana regular com elementos que representam a superfície do aerofólio (fronteira imersa). Nessa região utilizou-se 3 comprimentos característicos na direção x e 2 na direção y. Há ainda uma região de *stretching* de 512 elementos para cada direção, com um fator de alongamento de 1,05. Utilizou-se 1024, 2048 e 3072 elementos por comprimento característico. A Fig. 3.14 mostra a resolução da malha nas proximidades do aerofólio.

3.5.4 Parâmetros de simulação no OpenFOAM

A campanha de simulação no OpenFOAM consistiu na simulação do aerofólio dupla cunha e na simulação da asa tridimensional com as condições de contorno da Tab. 3.4.

Tabela 3.4: Parâmetros do escoamento sobre o aerofólio dupla cunha simulado no OpenFOAM

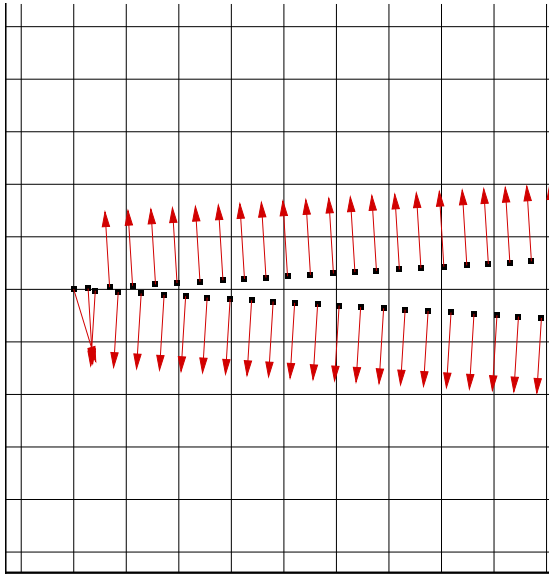
Caso	U (m/s)	T(K)	P(Pa)	Ma	Re	γ
3D, asa A	545	303	40900	1.562	1.0749E+6	1.4
3D, asa B	545	303	24250	1.5620	3.7941E+5	1.4
2D, aerofólio de dupla cunha	650	300	100000	1,8722	4.12E+07	1.4

A aplicação das condições de contorno seguiram o esquema da Fig. 3.15 e da Tab. 3.5.

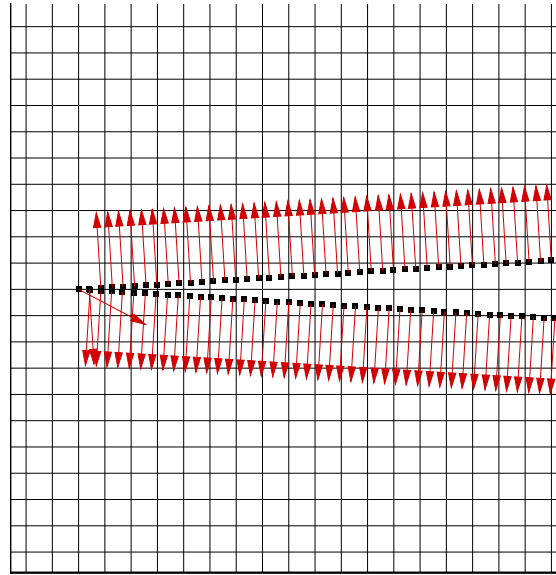
Tabela 3.5: Condições de contorno aplicadas nas regiões destacadas na Fig. 3.15

Região	Pressão	Velocidade	Temperatura
Entrada	Valor fixo	Valor fixo	Valor fixo
Topo e fundo	Sem reflexão de onda	Sem reflexão de onda	Sem reflexão de onda
Saída	Sem reflexão de onda	Sem reflexão de onda	Sem reflexão de onda
Aerofólio	Gradiente normal nulo	Escorregamento com fator de 100%	Gradiente normal nulo

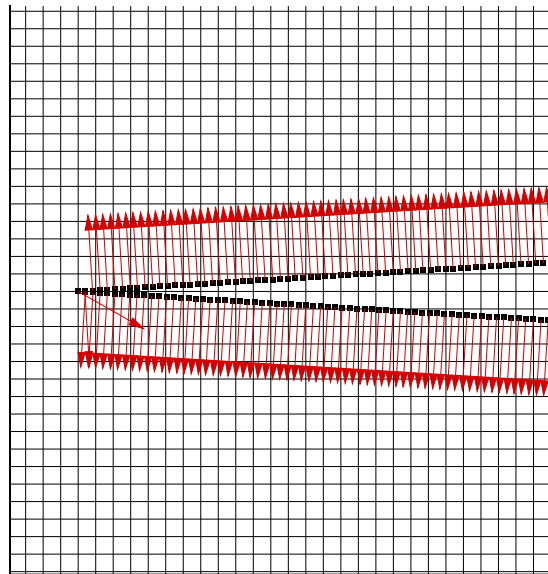
No caso tridimensional, nos planos do domínio que são paralelos à sustentação e ao arrasto foram aplicadas as mesmas condições de contorno do topo e fundo, conforme a Tab. 3.5.



(a)



(b)



(c)

Figura 3.14: Malha com a)1024 b)2048 e c)3072 elementos por comprimento característico

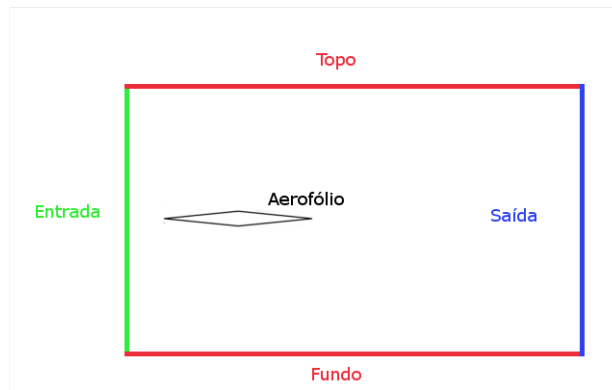


Figura 3.15: Regiões da malha do OpenFOAM para aplicação das condições de contorno

3.5.5 Parâmetros de simulação no VAT

Para a simulação foram utilizadas as equações governantes na forma não viscosa, e a condição de contorno foi de impenetrabilidade, com fator de escorregamento de 100%. O comprimento característico considerado foi 1 m, que é a corda do aerofólio.

Tabela 3.6: Condições de contorno aplicadas na simulação do aerofólio dupla cunha no VAT

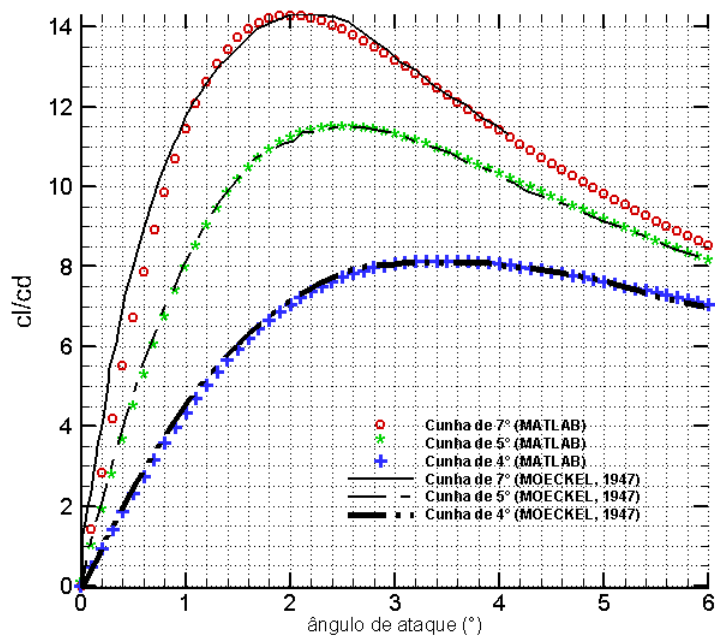
Velocidade (m/s)	Temperatura (K)	Pressão (Pa)	Mach	alpha
694.377	300	101325	2	0

Capítulo 4

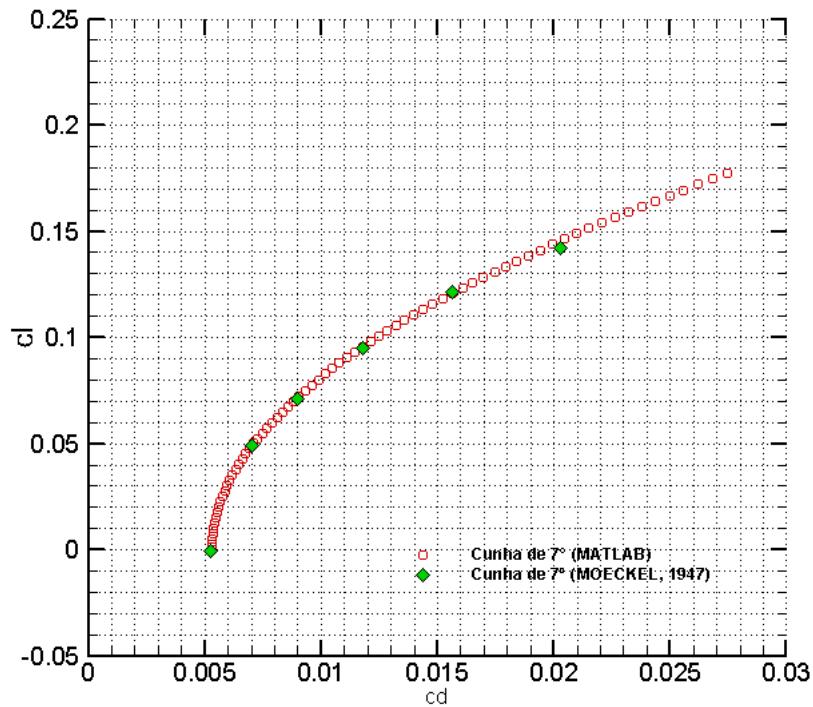
Resultados e discussões

4.1 Resultados do cálculo no MATLAB®

A rotina de cálculo foi validada pela comparação com os resultados apresentados em (34). Dessa forma, pela Figura 4.1 é possível concluir que há satisfatória precisão dos resultados obtidos pelo cálculo no MATLAB®.



(a) Razão c_l/c_d em função do ângulo de ataque para a configuração de aerofólios dupla cunha com ângulo da cunha de 4, 5 e 7 graus e para $Ma = 3,0$



(b) Polar de arrasto para a configuração de aerofólio dupla cunha com ângulo da cunha igual a 7 graus e com $Ma = 3,0$

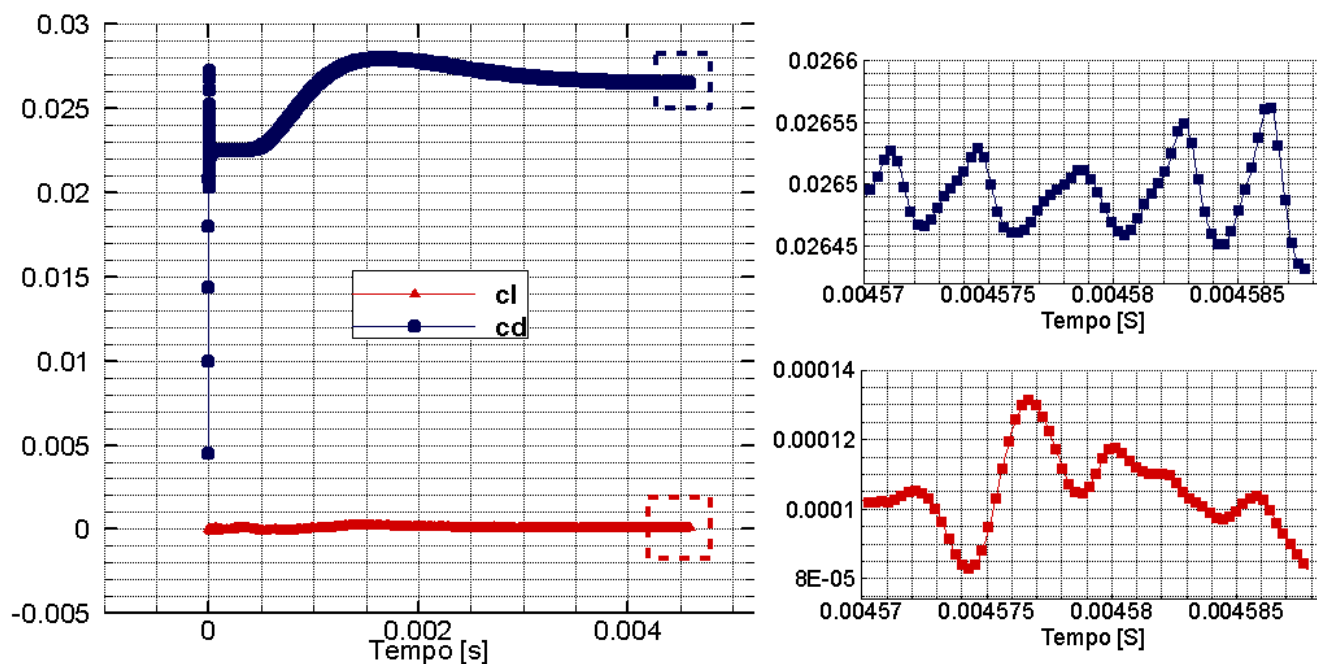
Figura 4.1: Resultados obtidos por rotina de cálculo do MATLAB® e comparados com os resultados da referência na literatura

(34)

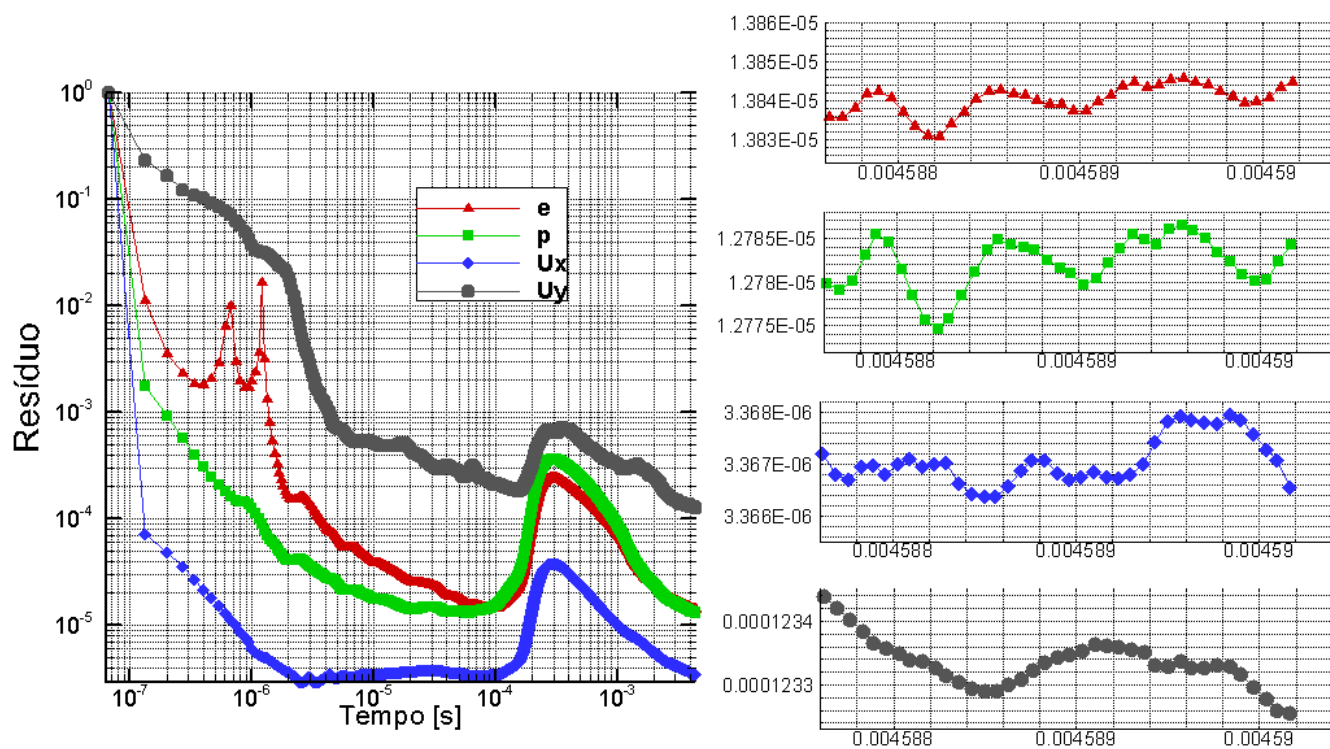
4.2 Resultado da simulação do aerofólio bidimensional

4.2.1 Simulação no OpenFOAM sem malha adaptativa

Para cada simulação realizada verificou-se a convergência dos resultados a partir do coeficiente de arrasto e do erro inicial de cada iteração. As figuras 4.4, 4.3 e 4.2 mostram a evolução dos coeficientes de sustentação e arrasto e dos erros iniciais de cada iteração para as malhas A, B e C, respectivamente. Os resultados apresentaram convergência para o estado permanente, a se julgar pelo gráfico de arrasto em função do tempo, para cada uma das malhas.

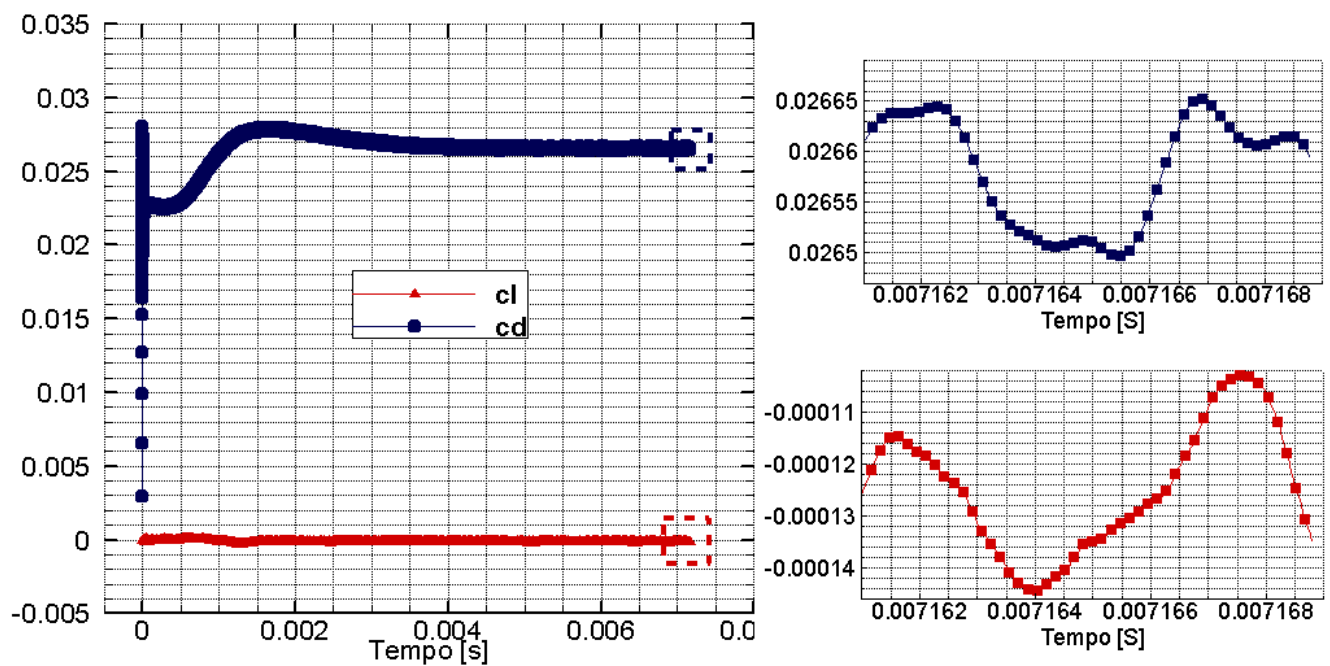


(a) coeficiente de arrasto e de sustentação

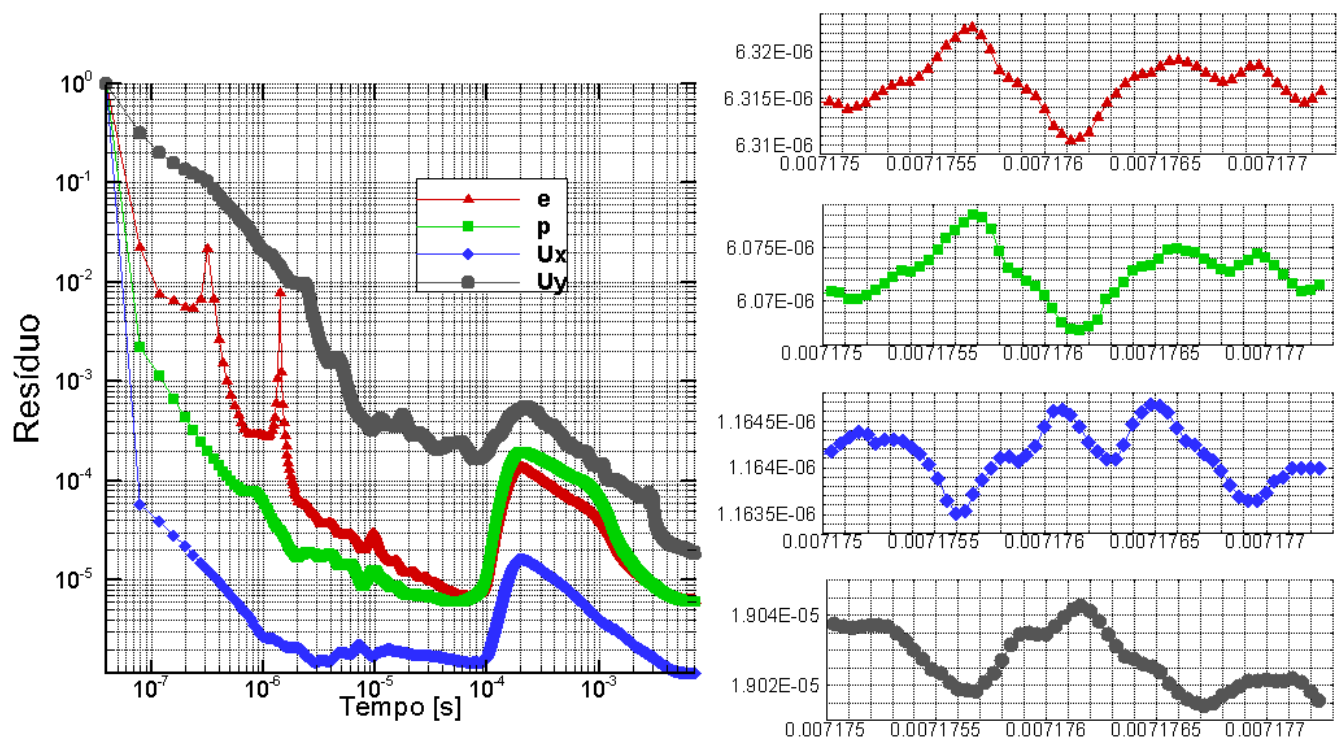


(b) erro inicial de cada iteração

Figura 4.2: Resultados para a malha C (2,5 milhões de elementos)

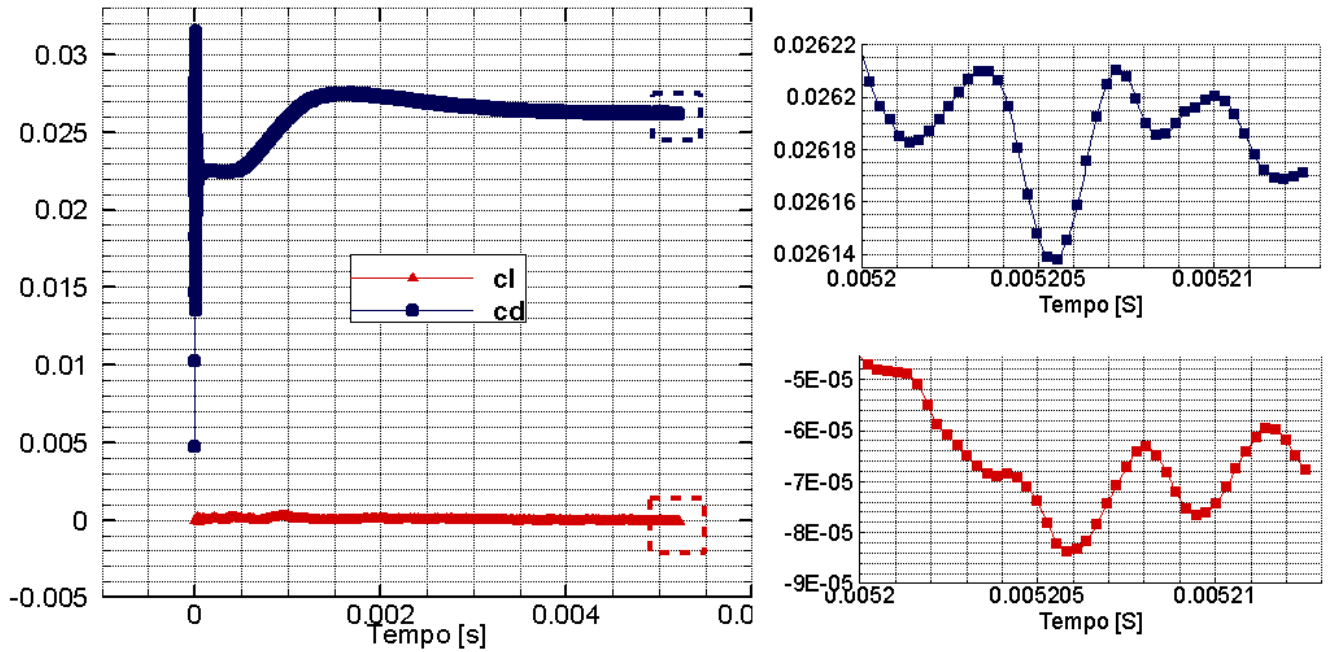


(a) coeficiente de arrasto e de sustentação

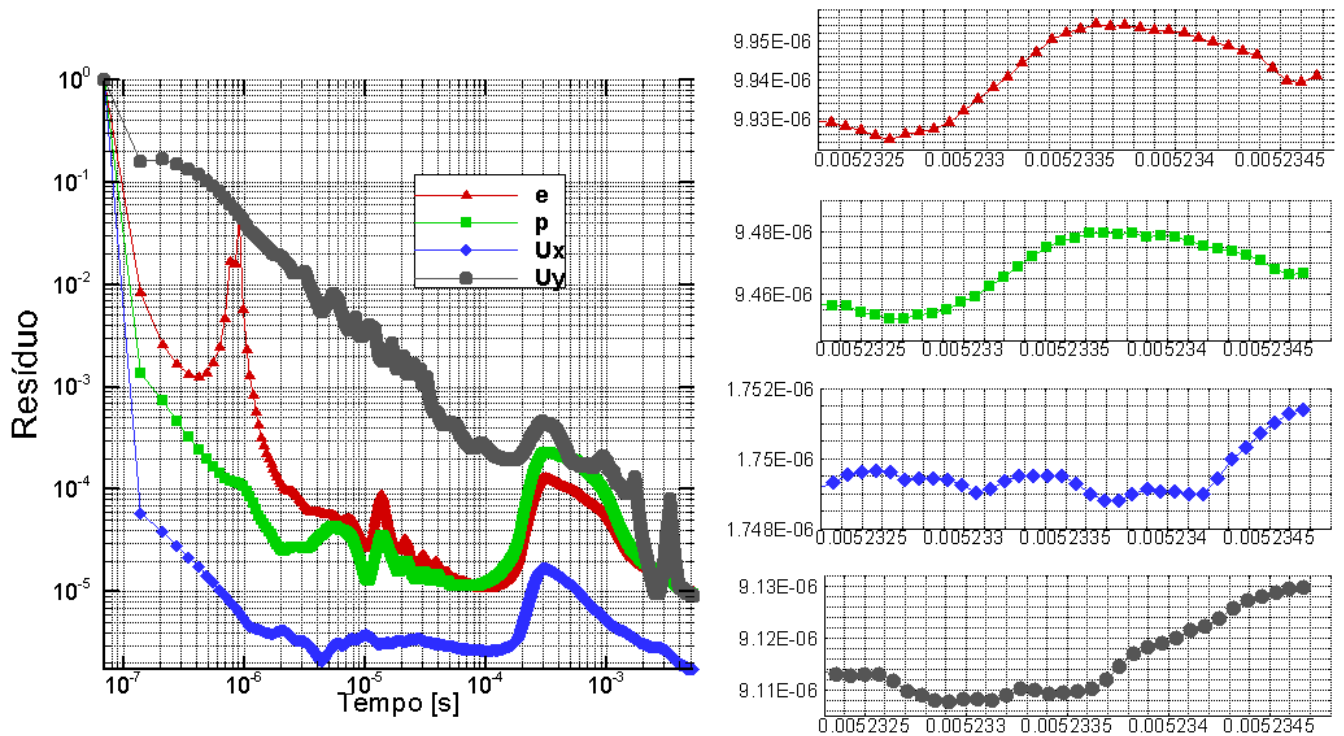


(b) erro inicial de cada iteração

Figura 4.3: Resultados para a malha B (1 milhão de elementos)



(a) coeficiente de arrasto e de sustentação



(b) erro inicial de cada iteração

Figura 4.4: Resultados para a malha A (500 mil elementos)

Para verificar a independência dos resultados com relação ao número de volumes foi feita a Figura 4.5. Verificou-se que o resultado varia apenas 0,6% entre as duas últimas malhas.

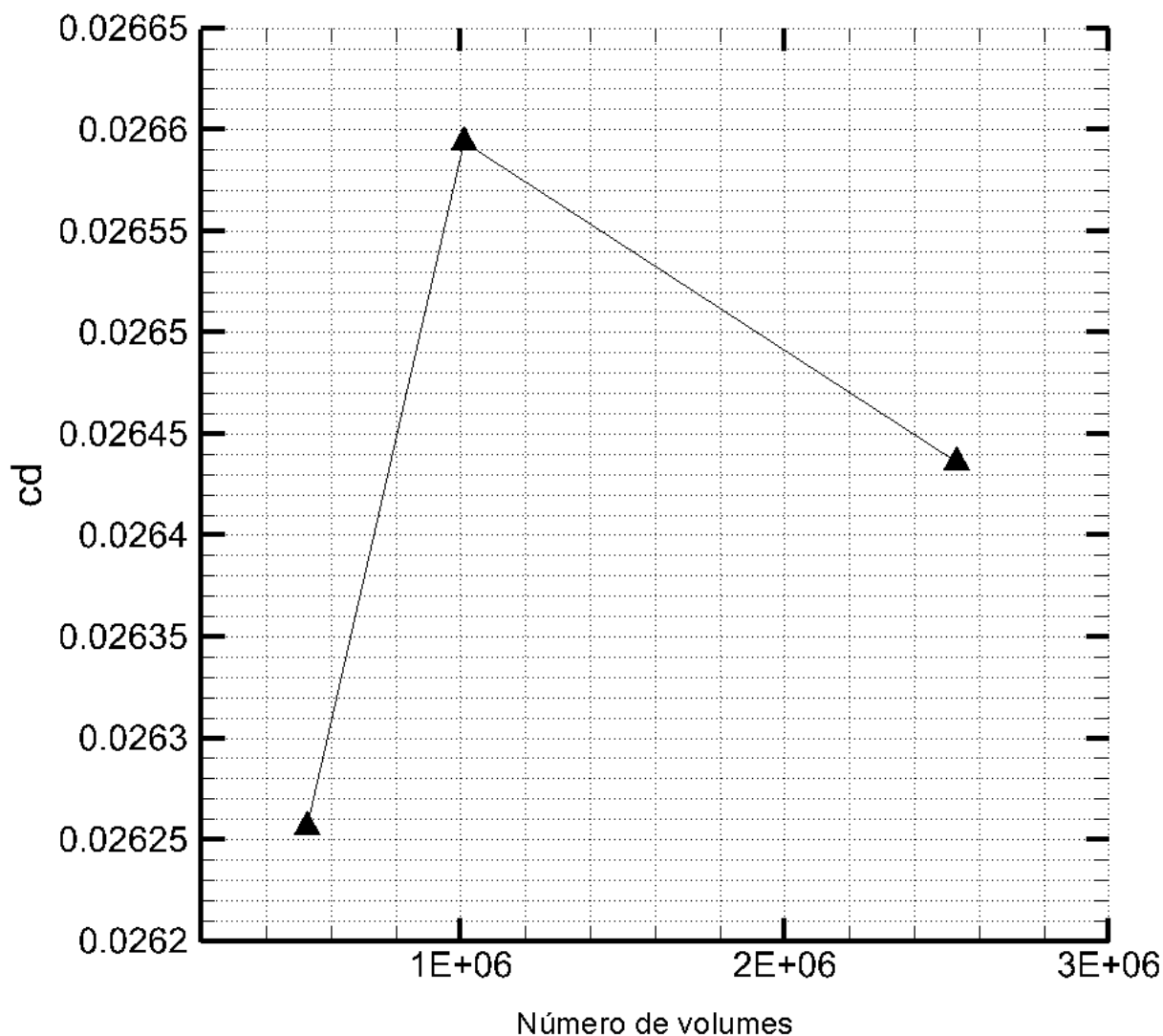


Figura 4.5: Coeficiente de arrasto em função do número de volumes, OpenFOAM

Com a última iteração da malha C foi feita a Figura 4.6, na qual é destacado o ângulo de choque e as linhas de corrente. Com auxílio da rotina no MATLAB ®obteve-se um ângulo de choque $\beta = 37,52^\circ$, que coincide com o resultado da simulação.

A figura 4.7 mostra a pressão no extradorso do aerofólio obtido na simulação e obtido analiticamente. Pela figura pode-se perceber que há satisfatória semelhança entre esses valores, com exceção das regiões de mudança brusca de pressão, que são o bordo de ataque, de fuga e o vértice na região central.

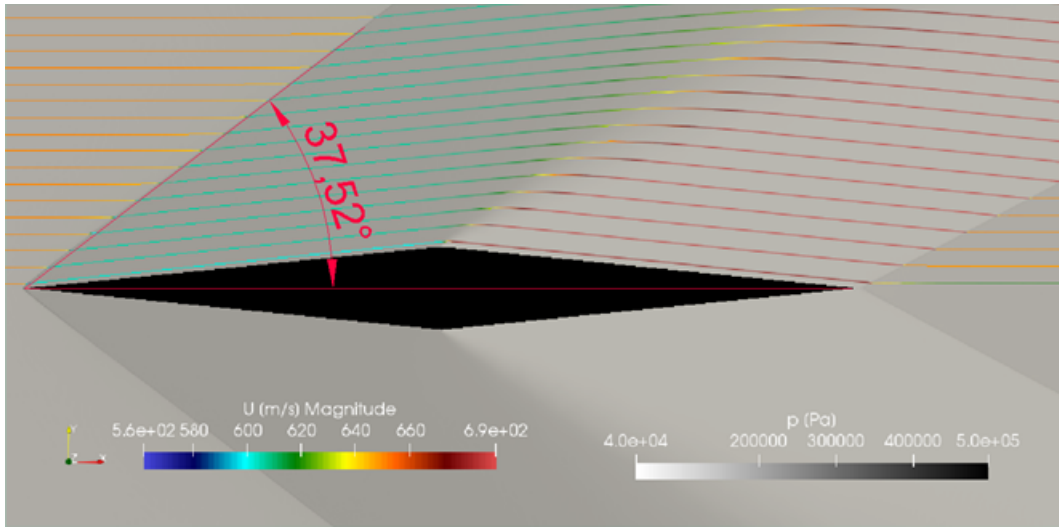


Figura 4.6: Escoamento simulado no OpenFoam com a malha computacional C. A imagem mostra a pressão em escala de cinza no fundo e as linhas de corrente em escala colorida. O ângulo do choque oblíquo simulado coincide com valor calculado

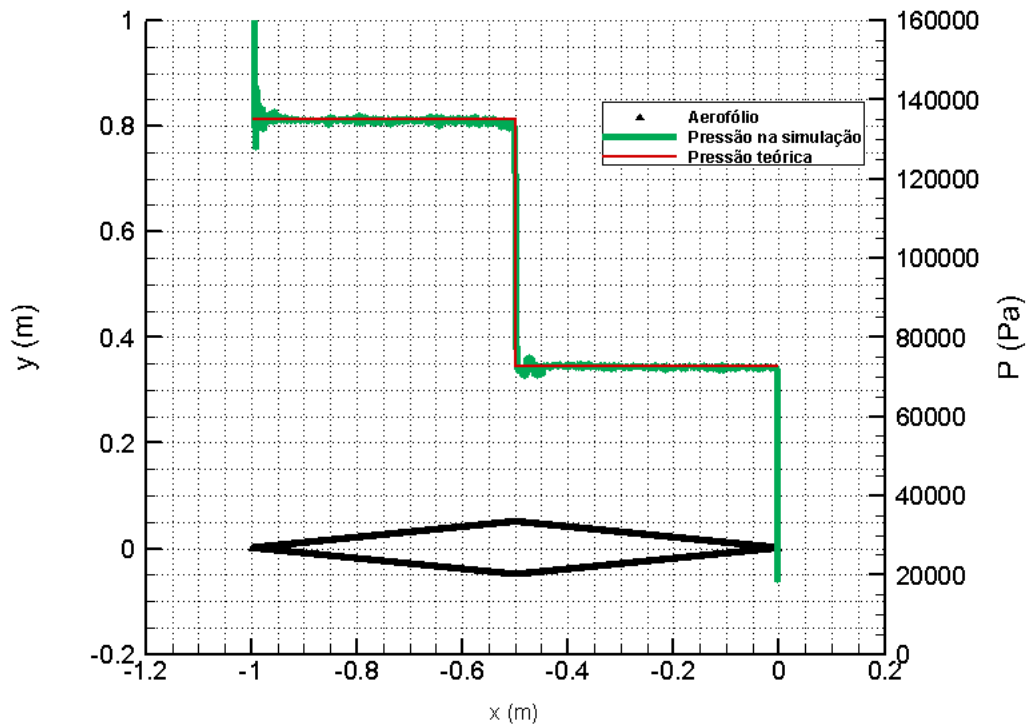


Figura 4.7: Pressão obtida pela simulação e pela teoria de choque e expansão.

Cada processador Intel Xeon Phi [®] possui 64 núcleos, com a possibilidade de alocação de 4 processos por núcleo, totalizando 256 possíveis subdivisões da simulação. Com mais de 64 subdivisões, é feita uma fila de execução em cada processador, que executa o primeiro processo até que os dados necessários não estejam mais disponíveis, e então é executado o segundo, caso os dados necessários estejam disponíveis. Essa metodologia é chamada de *Hyper-Threading Technology*, e tem o potencial de diminuir a latência no processamento pela execução concorrente.

Estima-se que o tempo de execução, que é obtido pela contagem de ciclos de processamento real, aumenta com $O(n^3)$, enquanto que o tempo de comunicação aumenta com $O(n^2)$, sendo n o número de volumes. Isso ocorre por que a comunicação é estabelecida por áreas de interação entre os volumes de cada processo. Além disso, com um número maior de subdivisões há um maior tempo de comunicação, ocasionado pelo aumento no número de interfaces. A ordem do aumento do tempo de execução com o número de volumes pode ser verificada de maneira aproximada pela Figura 4.8. Não foi possível obter dados do tempo de comunicação para se fazer uma análise semelhante.

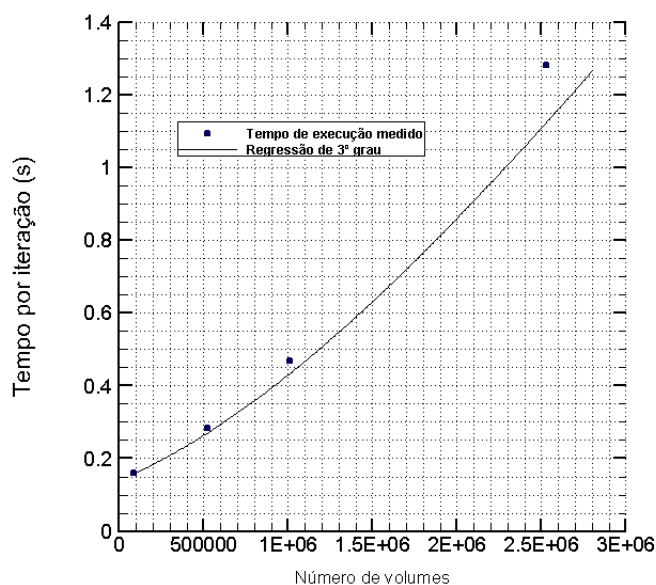


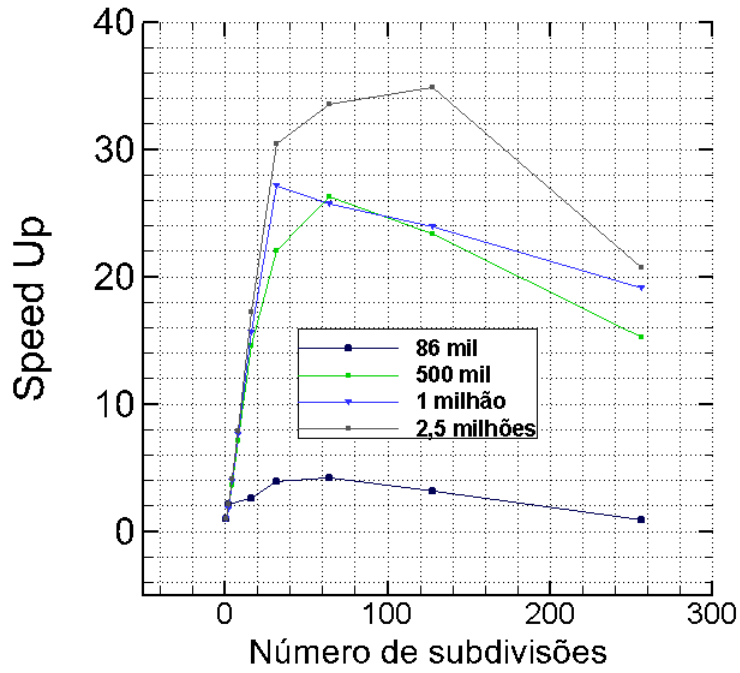
Figura 4.8: Tempo de execução da simulação com 256 subdivisões em função do número de volumes. É apresentado uma regressão polinomial de terceiro grau para verificar a ordem de aumento do tempo de execução

O valor do tempo total de simulação é a soma do tempo de execução, que pode ser paralelizado, com o tempo de comunicação e de outros comandos que são seriais. Assim, a redução no tempo total de simulação pelo processamento paralelo tende a ser mais expressiva em simulações de grande número de volumes e menos eficiente ou até mesmo inexistente em simulações com menor número de volumes, uma vez que se diminui o tempo de execução com a penalidade de se aumentar o número de interfaces no domínio, e conseqüentemente o tempo de comunicação entre os processos é maior.

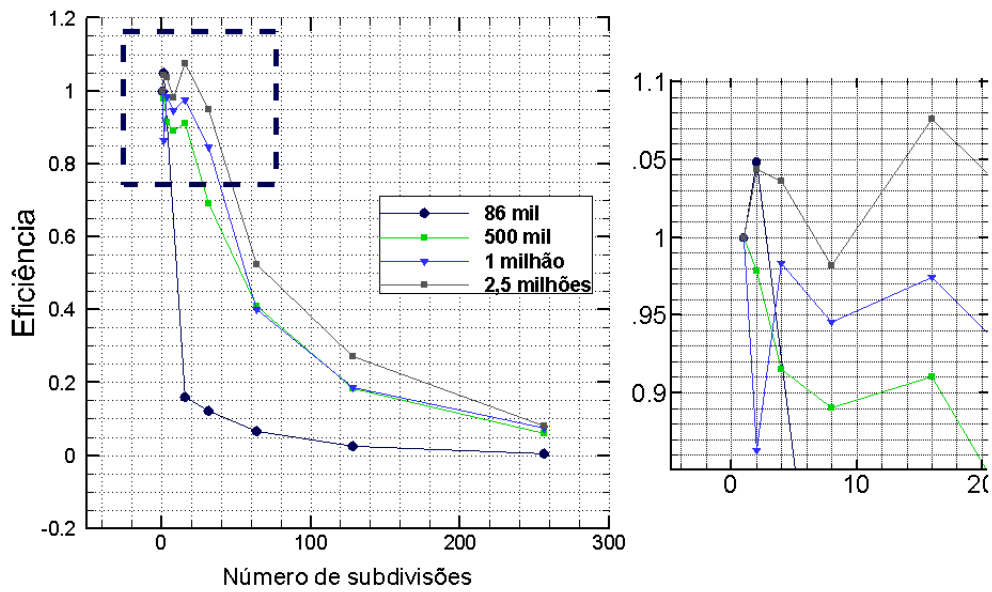
Para a análise da escalabilidade do OpenFOAM, realizou-se medição do tempo total de simulação (*clock time*) e o tempo de execução (*execution time*) em função diferentes números de subdivisões da malha de cálculo. Foram utilizados de 1 a 256 subdivisões, variando em uma sequência com progressão geométrica de razão 2, começando com 1. Utilizou-se também malhas de cálculo com 86053, 524660, 1009021 e 2530875 elementos simulando o mesmo caso. O tempo total de simulação e de execução foram divididos pela quantidade de iterações para se obter o tempo médio por iteração, e o resultado está apresentado na Figura 4.10. Para o caso de 86 mil elementos, o tempo total de simulação é 77% maior que o tempo de execução, ao passo que no caso de 2,5 milhões de elementos a diferença é de 15%, considerando o caso de 256 subdivisões. Essa redução da diferença corrobora com o que foi dito no parágrafo anterior.

Define-se *speed up* como sendo a razão entre o tempo de simulação de um código em paralelo e o do mesmo código na forma serial. Também pode-se definir a eficiência como a razão entre o *speed up* e o número de instâncias do processamento executado em paralelo. Com isso, foi feita a Figura 4.9 na qual é possível notar que quanto maior for a quantidade de volumes, maior será a eficiência e o *speed up*. Além disso, há alguns pontos com eficiência maior que a unidade, o que é chamado de *speed up superlinear*. Alguns fatores para que isso ocorra são a tolerância à latência da comunicação, o aumento da capacidade de memória, subdivisão do problema (tarefas menores geram menos *cache misses*) e aleatoriedade da computação.

Pela figura 4.9, nota-se que há um acréscimo do *speed Up* com mais de 64 subdivisões somente no caso de 2,5 milhões de elementos e de 500 mil elementos. Isso sugere que a latência é um fator que começa a se tornar relevante por volta desse tamanho de simulação. Além disso, o *speed up* do caso de 2,5 milhões de elementos se mantém quase linear até 32 instâncias de processamento. Isso sugere que uma relação adequada de volumes e instâncias é da ordem de 80 mil elementos por instância.



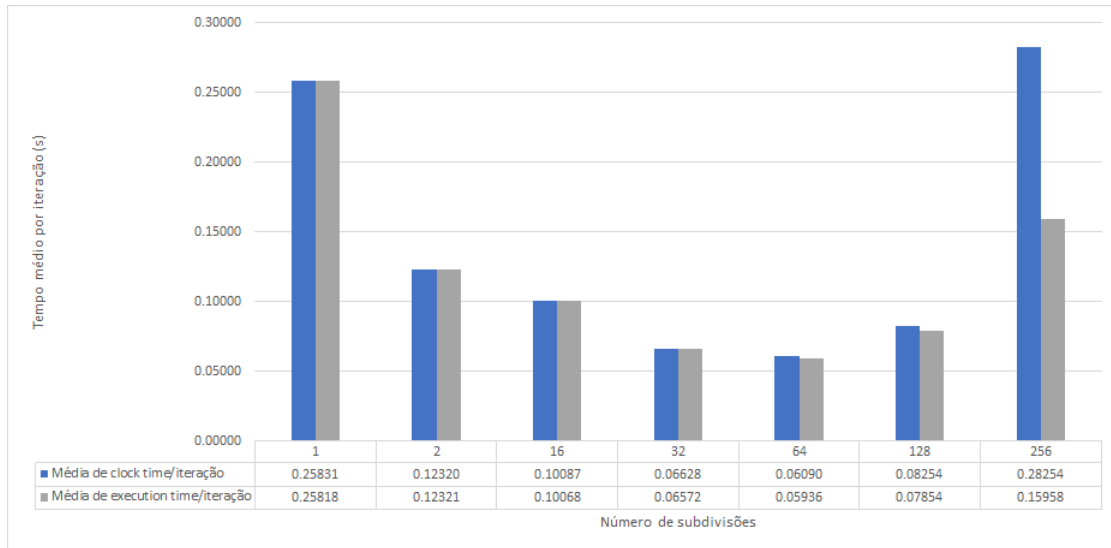
(a) *Speed Up*



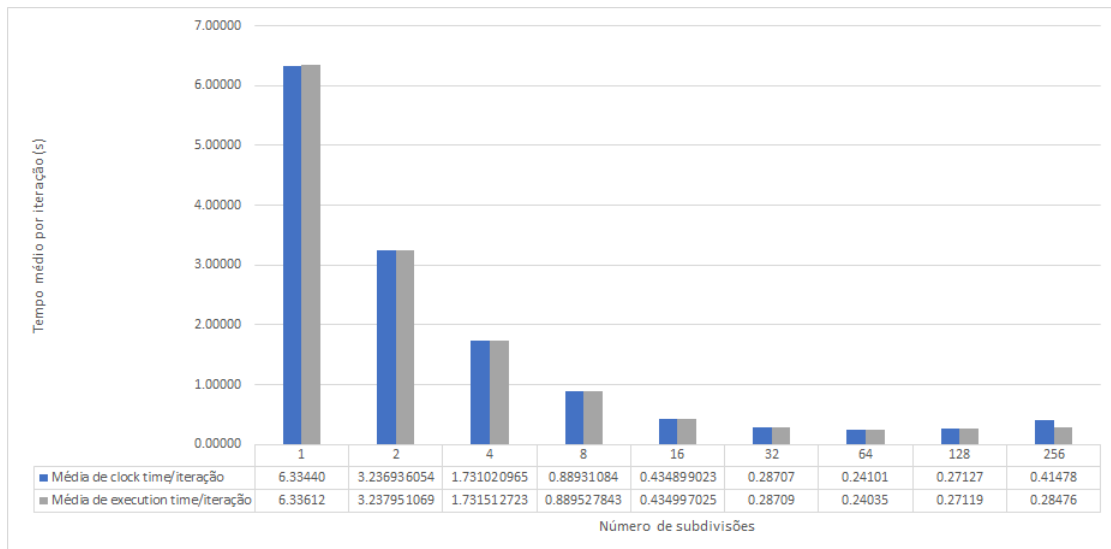
(b) Eficiência da simulação

Figura 4.9: Resultados da escalabilidade do OpenFOAM em função do número de volumes. Quanto maior a quantidade de volumes, maior a eficiência e maior o *Speed Up*.

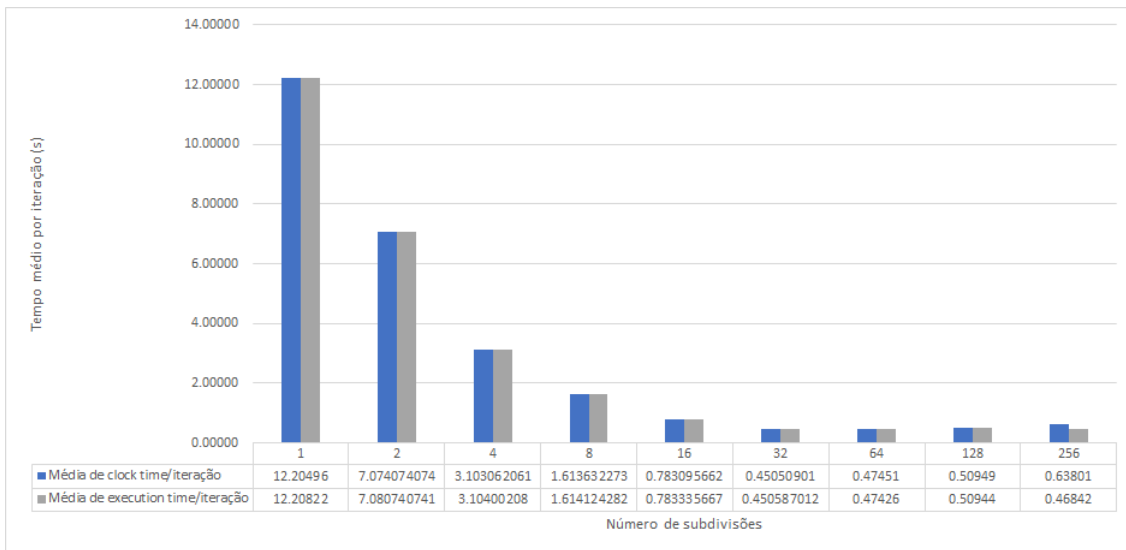
Obteve-se o valor do coeficiente de arrasto pelas bibliotecas do OpenFOAM (*libforces.so*) e pela rotina de integração da pressão feita no MATLAB. O resultado é apresentado na Tabela 4.1.



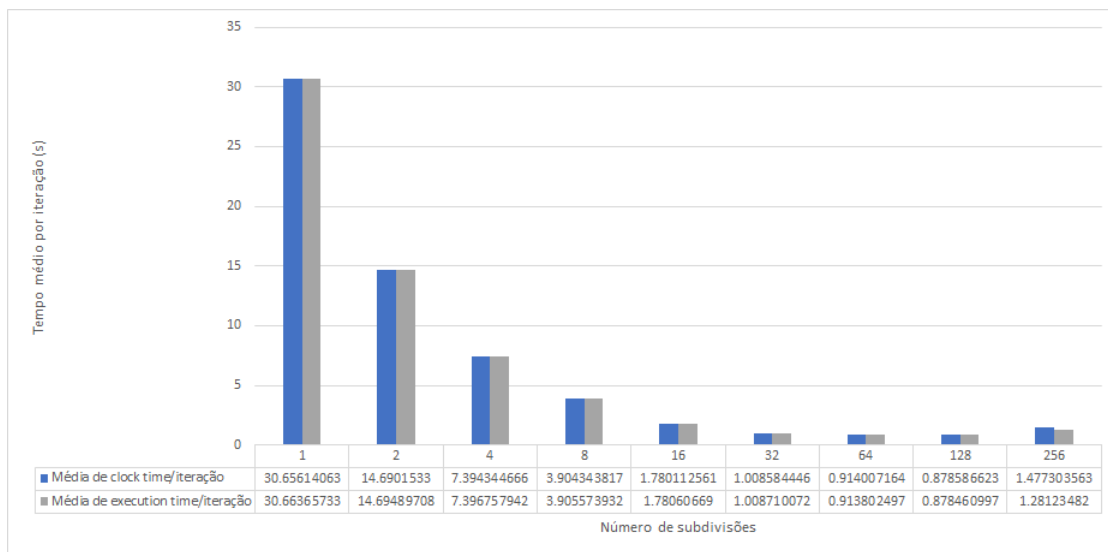
(a) 86 mil elementos



(b) 500 mil elementos



(c) 1 milhão de elementos



(d) 2,5 milhões de elementos

Figura 4.10: Gráfico de barras do tempo total de simulação por iteração (dt clock/iter) e do tempo de execução por iteração (dt exec/iter) em função do número de subdivisões para diferentes números de volumes.

Tabela 4.1: Valores de coeficiente de arrasto para a simulação no OpenFOAM sem malha adaptativa

	cd	Erro
Cálculo do OpenFOAM	0.026436	4.14%
Cálculo da rotina em MATLAB	0.025147	0.94%
Valor teórico	0.025386	-

A diferença entre o erro do cálculo do OpenFOAM (4,14%) e do cálculo por rotina desenvolvida em MATLAB (0,94%) pode ser explicada pelo fato de que a malha possui um truncamento na ponta, que só é levado em conta no cálculo do OpenFOAM. A Figura 4.11 mostra o escoamento nas proximidades desse truncamento.

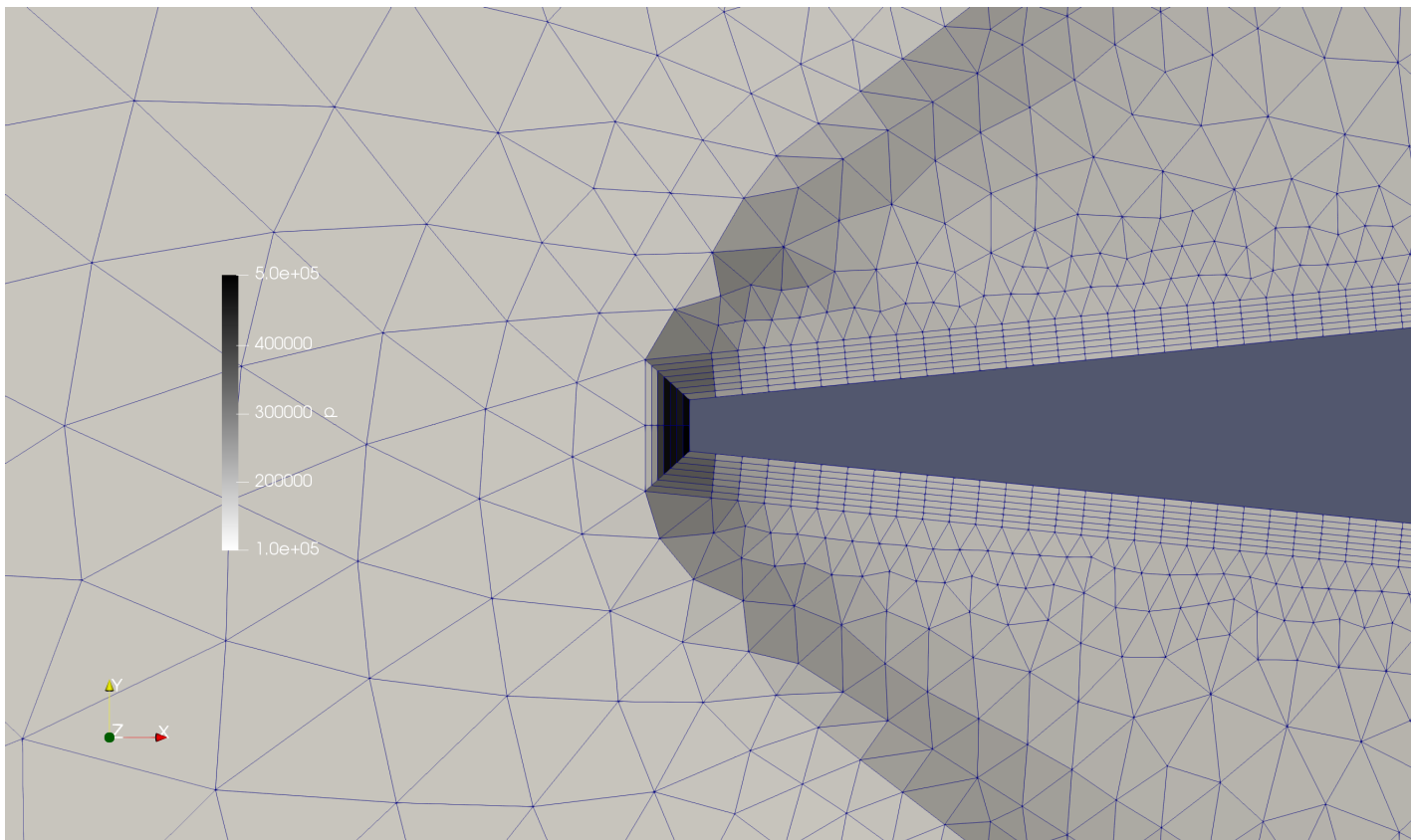


Figura 4.11: Escoamento nas proximidades do truncamento na malha do OpenFOAM

4.2.2 Simulação no OpenFOAM com malha adaptativa

A simulação com malha adaptativa foi realizada com as mesmas condições de contorno da simulação sem malha adaptativa. A malha foi gerada a partir de hexágonos regulares e foi utilizado o algoritmo *snappyHexMesh* do OpenFOAM. É importante ressaltar que o mecanismo de geração de malha adaptativa está configurado para funcionar apenas a partir de elementos hexaédricos na versão 1712 do OpenFOAM.

Definiu-se que a malha iria ser refinada em regiões de alto gradiente de velocidade, e para isso foi necessário inserir o seu cálculo durante o processamento da simulação, para que essa variável de campo entrasse no registro de objetos do OpenFOAM. Dessa forma, alterou-se o algoritmo para acrescentar o cálculo do gradiente de velocidade e foi feita a recompilação. Definiu-se também um valor máximo para o Co , para que a cada iteração o OpenFOAM possa calcular o passo temporal de acordo com o tamanho do menor volume da malha, que muda no decorrer da simulação.

Primeiro simulou-se o caso de ângulo de ataque nulo sem refinamento. Verificou-se que o gradiente de velocidade na região do choque oblíquo estava entre 2000 a 8000 s^{-1} . Com esse dado alterou-se o mecanismo de simulação para que acontecesse o refinamento dos elementos que possuem gradiente de velocidade nessa faixa, e que isso fosse feito a cada 1000 iterações. Configurou-se o máximo número de volumes para 5 milhões, para que quando o número de elementos passe desse máximo, o mecanismo de refinamento seja interrompido, e aconteça apenas a diminuição desse refinamento. A malha resultante é apresentada na Figura 4.12. O resultado da simulação é apresentado na Figura 4.15.

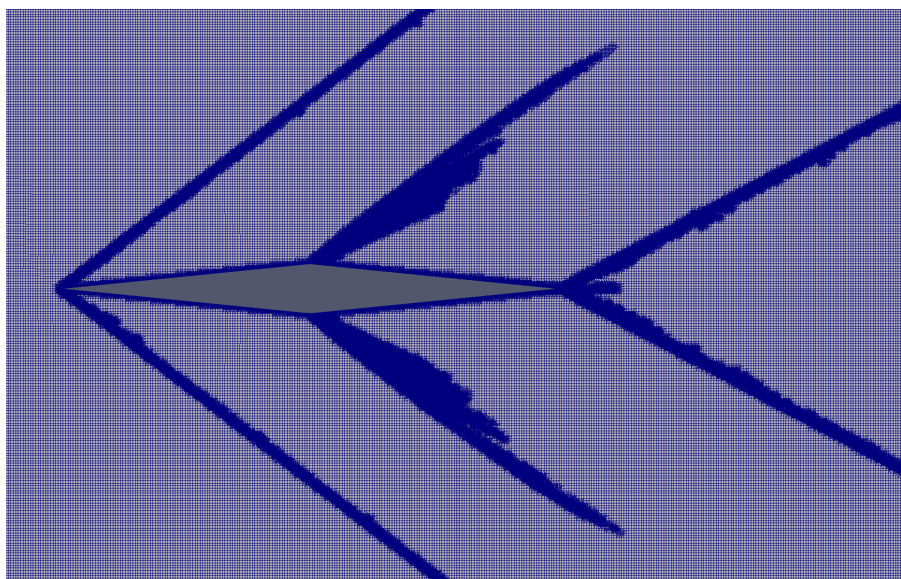


Figura 4.12: Malha resultante da simulação com refinamento automático nas regiões de alto gradiente de velocidade

Foi relatado em (19) que é possível atingir um menor erro com a possibilidade de aumento e diminuição do refinamento. Diante disso foi realizada também uma simulação em que a malha inicial apresentou um nível maior de refinamento, e o mecanismo de adaptação da malha poderia

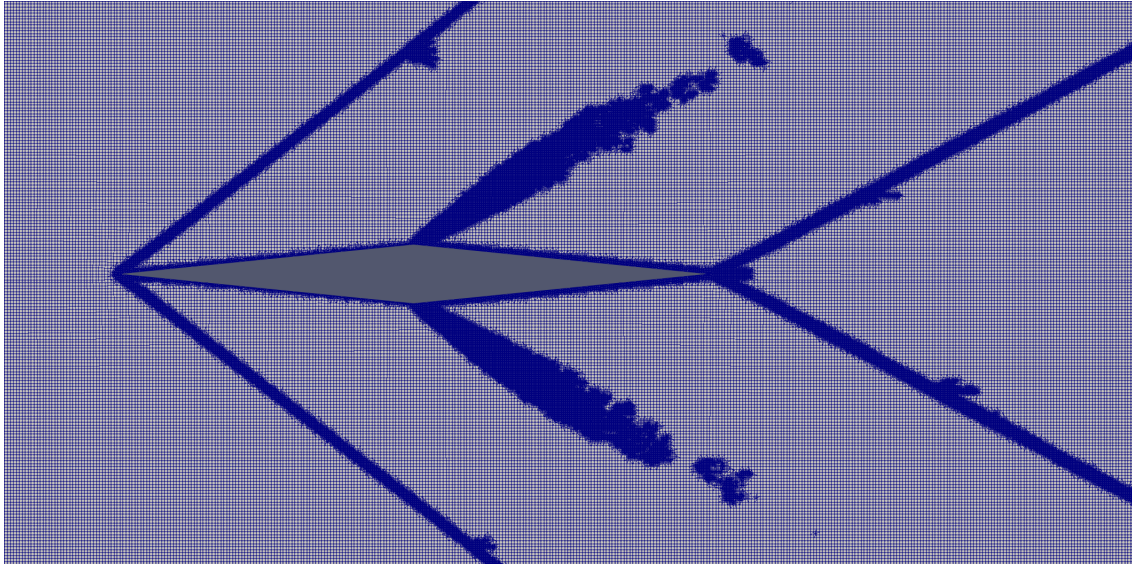


Figura 4.13: Malha resultante da simulação com possibilidade de aumentar e diminuir o refinamento em função do gradiente de velocidade

refinar e engrossar a malha de acordo com o gradiente de pressão. O resultado dessa simulação é apresentado na Figura 4.16 e a malha resultante é apresentada na Figura 4.13.

Por fim realizou-se uma simulação com ângulo de ataque igual a 5 graus. A malha resultante está presente na Figura 4.14 e os resultados são apresentados na Figura 4.17

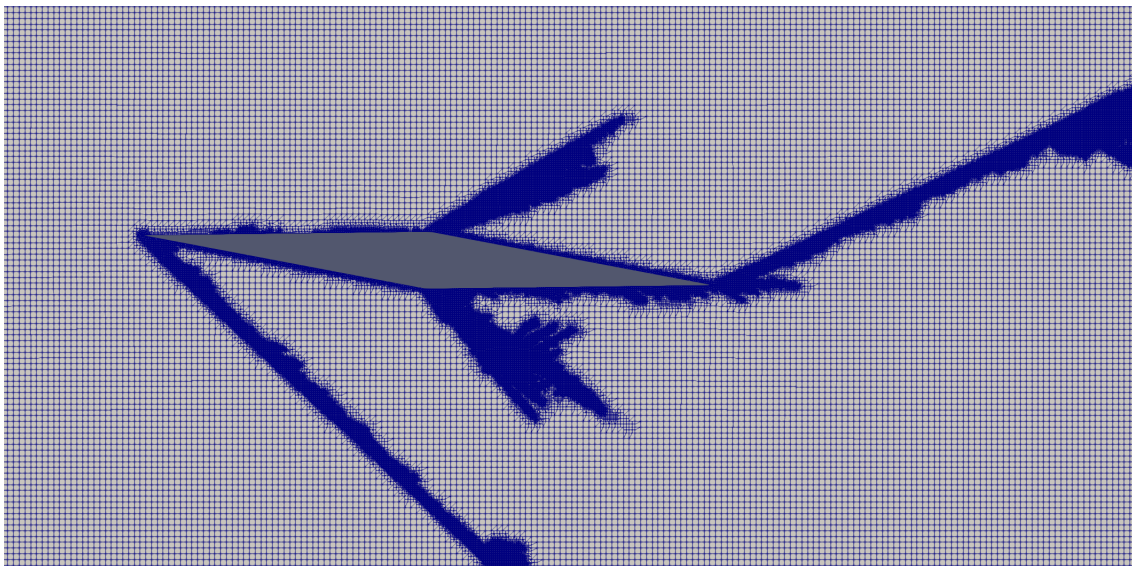


Figura 4.14: Malha resultante da simulação com refinamento automático nas regiões de alto gradiente e com ângulo de ataque de 5 graus

Durante o curso das simulações, notou-se que cada ciclo de refinamento acarreta em um prejuízo de tempo significativo, porém infelizmente não foi possível medir o tempo dispendido em cada ciclo. Essa penalidade faz com que a viabilidade de simulações com refinamento automático seja reduzida,

se justificando apenas quando o número de ciclos de refinamento e o número de volumes são baixos. Diante disso, a metodologia para a aplicação eficiente do refinamento automático seria primeiro atingir-se o estado permanente com uma malha menor, para então aplicar-se o refinamento, e esse processo poderia ser repetido poucas vezes.

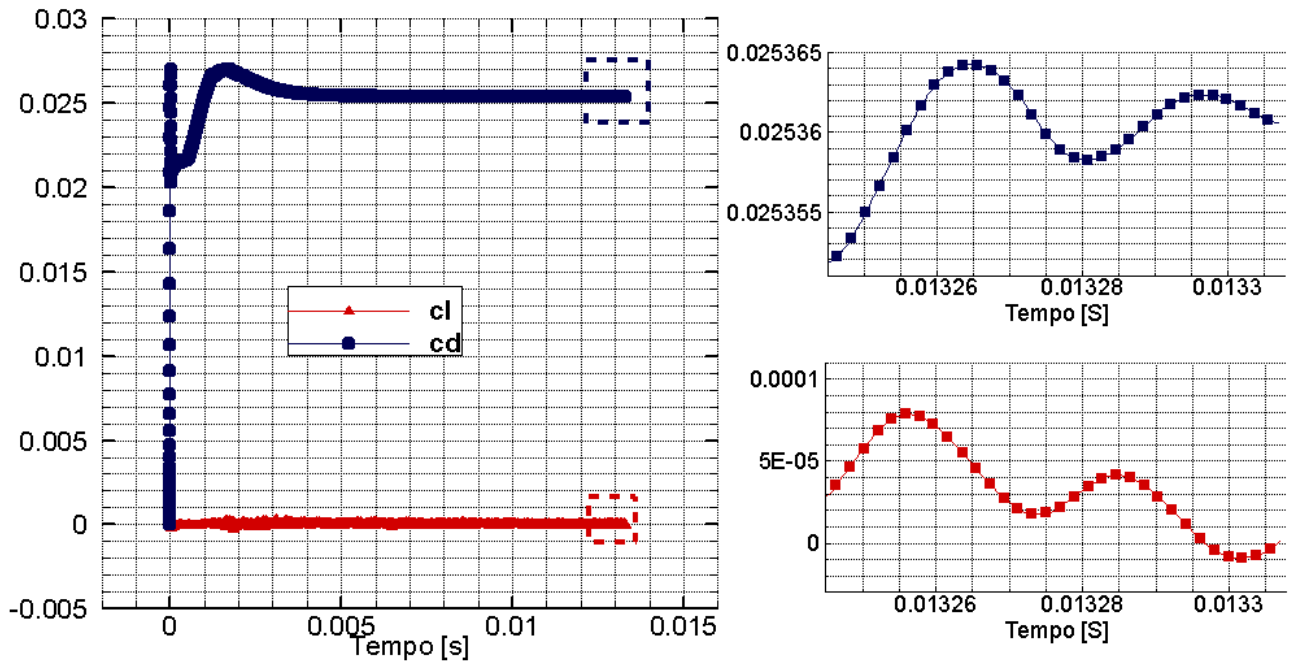
De posse dos dados de coeficiente de arrasto e sustentação das simulações com malha adaptativa, foi possível montar a Tabela 4.2 e 4.3. É possível perceber que há uma pequena piora dos resultados quando se aplica a possibilidade de aumentar e diminuir o refinamento, o que não corrobora com os resultados de (19). Contudo, a diferença é pequena, e é da ordem da variação numérica do próprio resultado no escoamento convergido.

Tabela 4.2: Valores de coeficiente de arrasto para a simulação no OpenFOAM com malha adaptativa

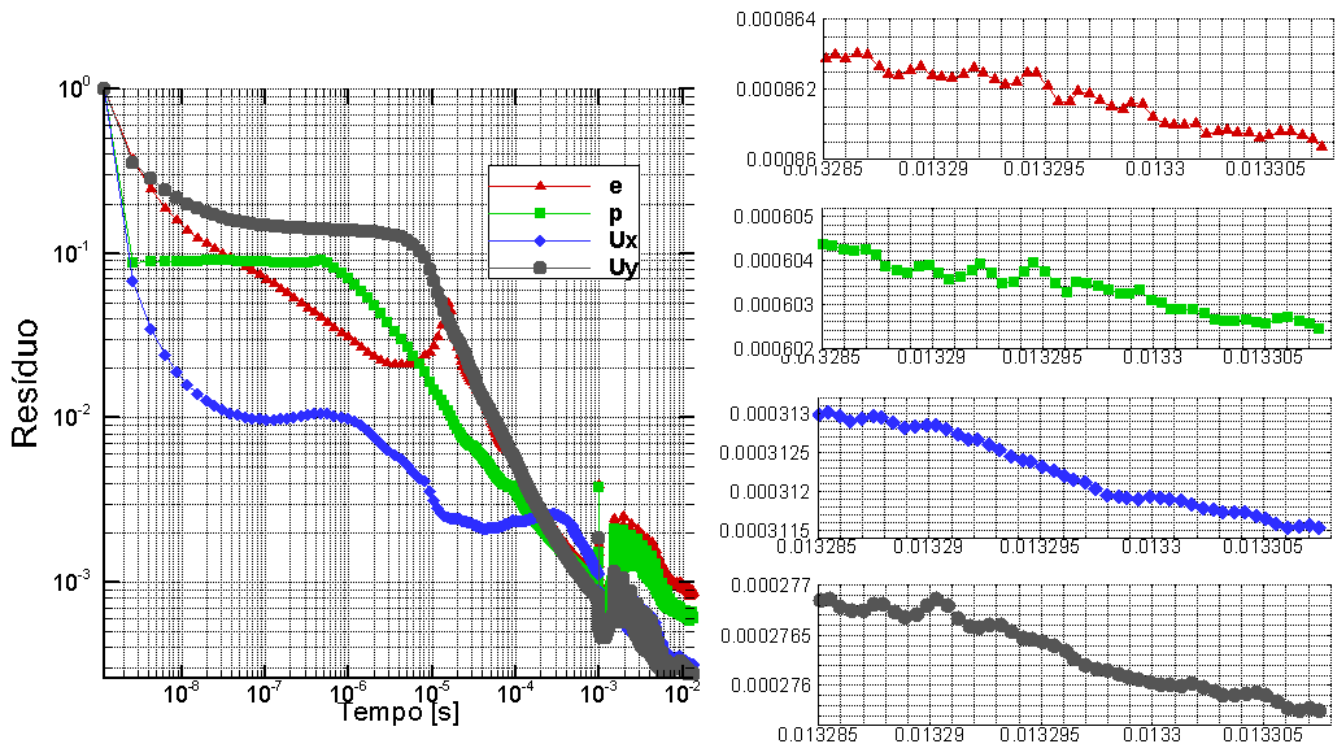
	cd	Erro
Só refinamento	0.025360575	0.10%
Aumentando e diminuindo o refinamento	0.025614422	0.90%
Valor teórico	0.025386	-

Tabela 4.3: Valores de coeficiente de arrasto para a simulação no OpenFOAM com malha adaptativa e com ângulo de ataque

	cd	Erro cd	cl	Erro cl
Malha com apenas refinamento e com ângulo de ataque	4.56E-02	0.21%	2.24E-01	0.64%
Valores teóricos	0.045747	-	0.22584	-

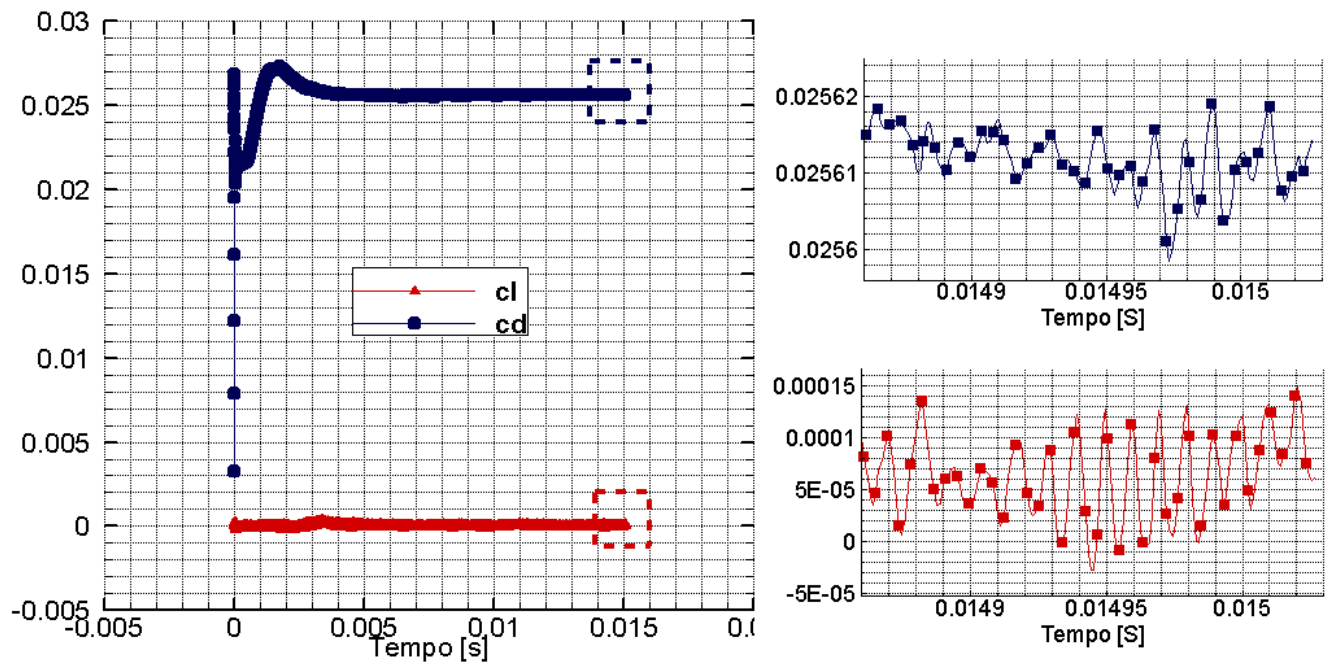


(a) coeficiente de arrasto e de sustentação

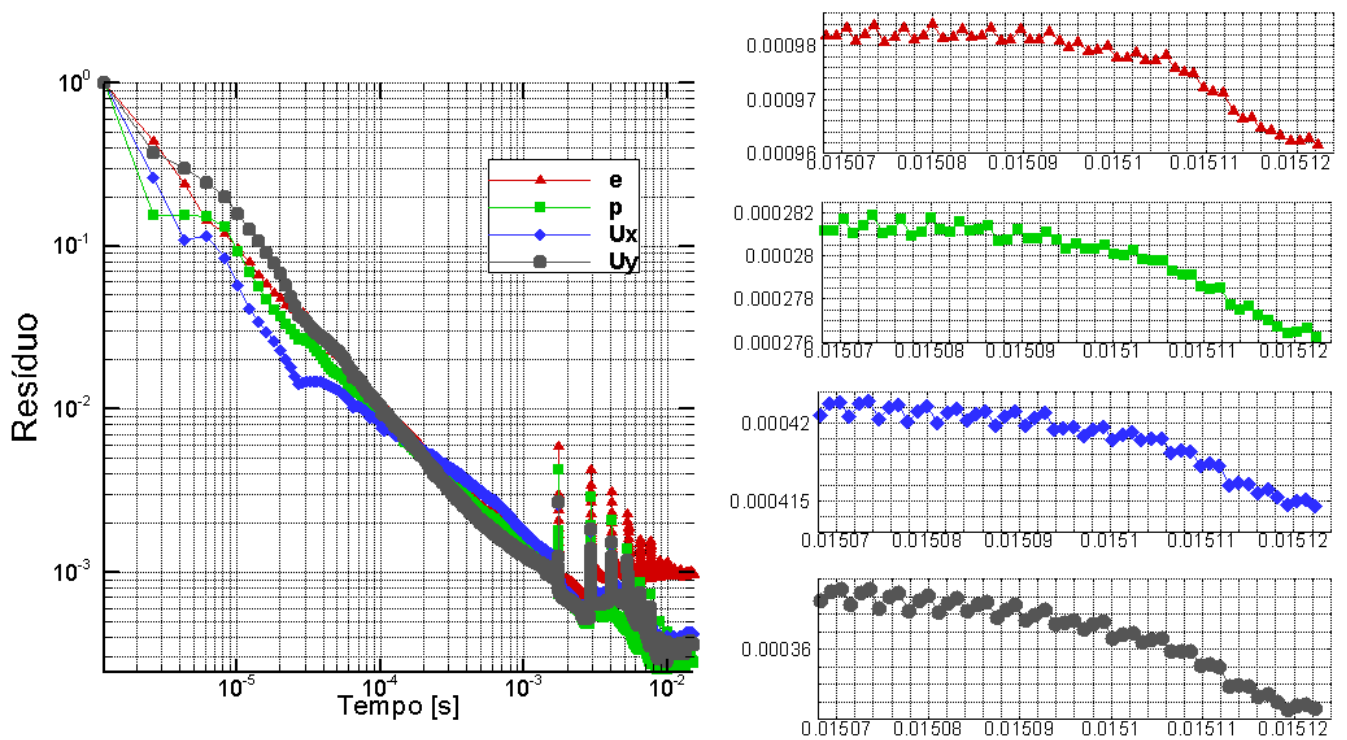


(b) erro inicial de cada iteração

Figura 4.15: Resultados da malha adaptativa com apenas refinamento

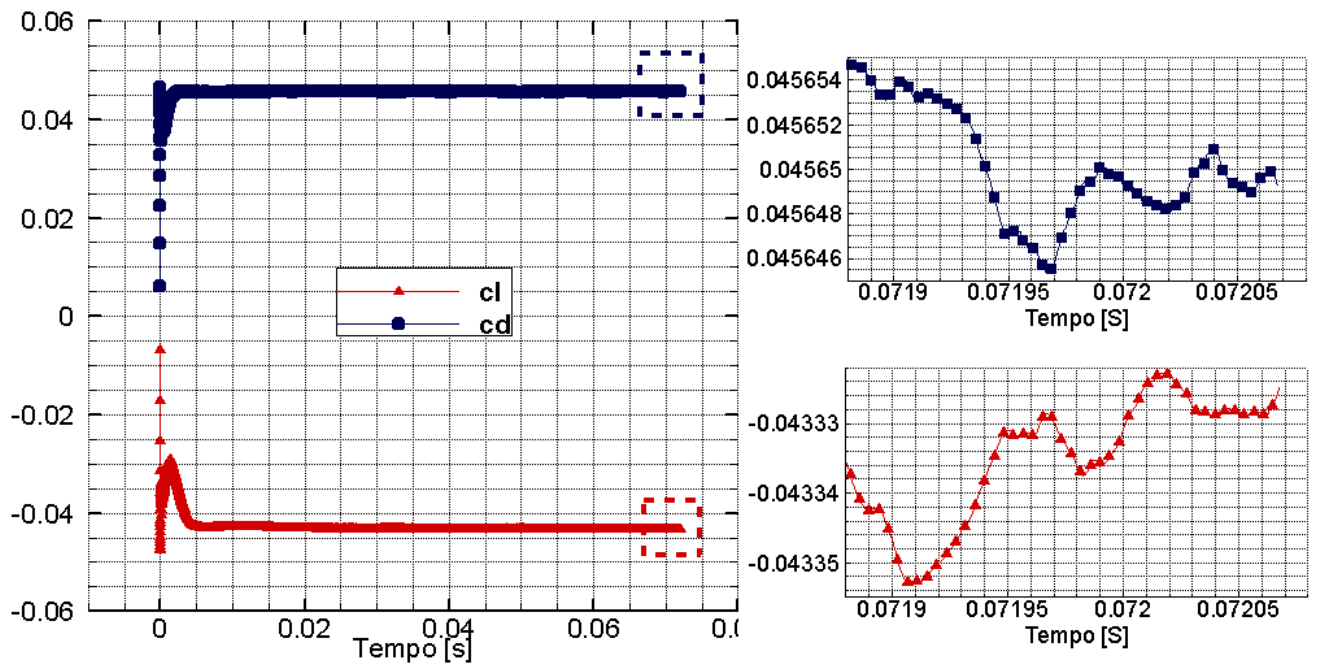


(a) coeficiente de arrasto e de sustentação

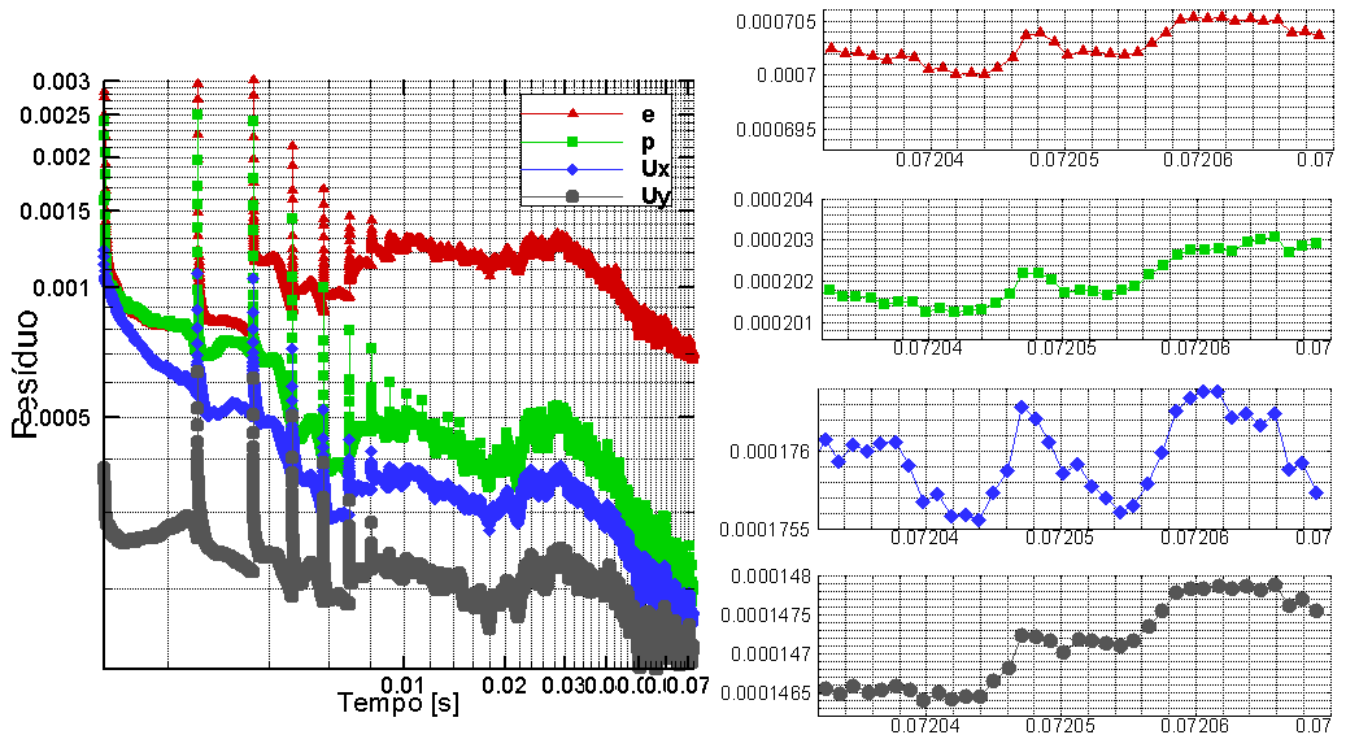


(b) erro inicial de cada iteração

Figura 4.16: Resultados da malha adaptativa com possibilidade de aumento e diminuição do refinamento



(a) coeficiente de arrasto e de sustentação



(b) erro inicial de cada iteração

Figura 4.17: Resultados da malha adaptativa com apenas refinamento e com ângulo de ataque 5 graus

4.2.3 Simulação no VAT

No VAT foram feitas 3 simulações, com região regular de 3m por 2m e com 12,6, 36,7 e 73,4 milhões de volumes volumes. Utilizou-se como parâmetros $Ma = 2,0$ e ângulo da cunha de 7 graus. O resultado de cada simulação está representado na Figura 4.18. A diferença entre os dois últimos valores é de apenas 0.29%. Os valores dos coeficientes foram obtidos pela rotina de integração desenvolvida em (35).

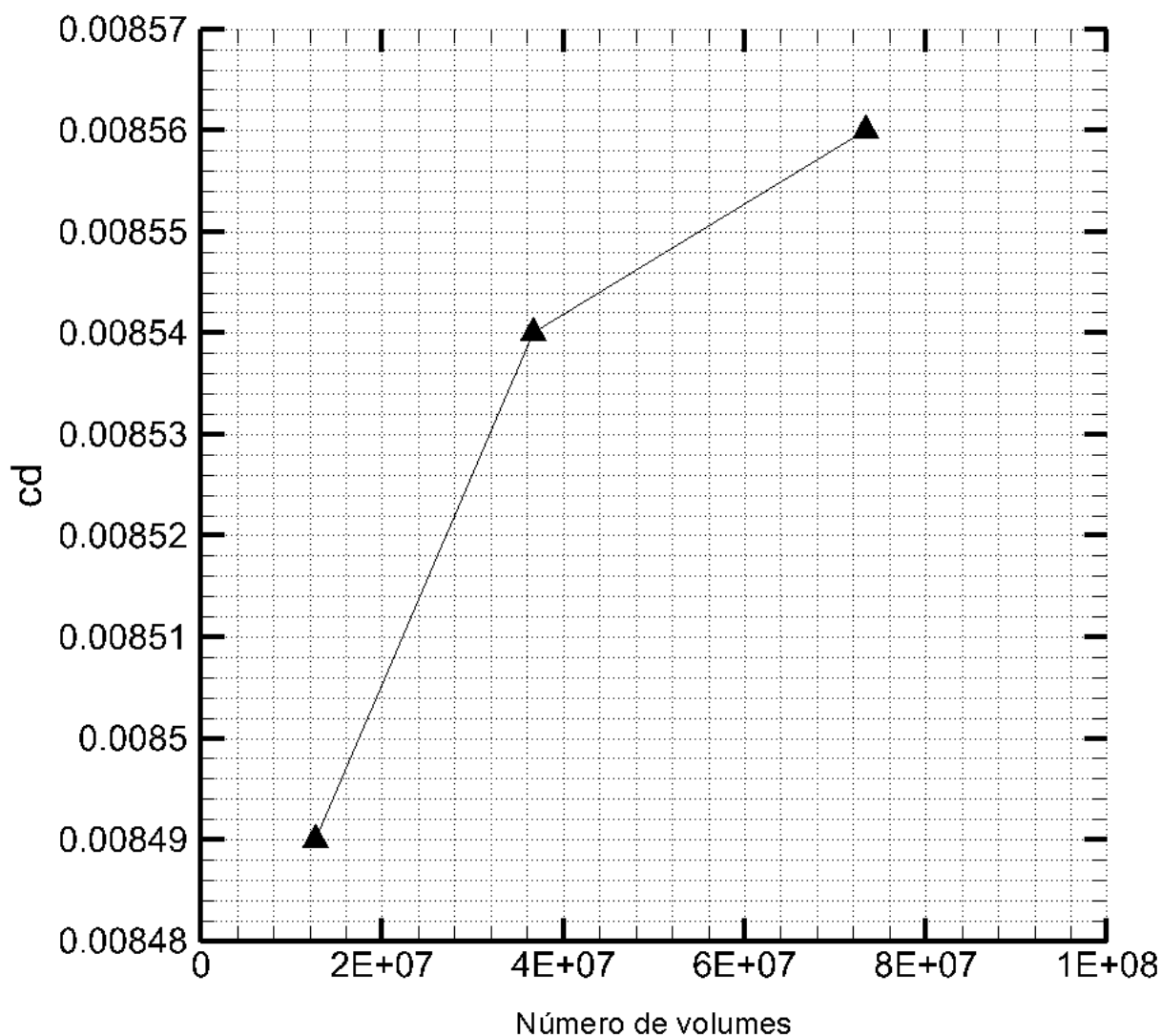


Figura 4.18: Coeficiente de arrasto em função do número de volumes, VAT

O escoamento desenvolvido até o estado permanente é apresentado na Figura 4.19 e a partir dos dados de pressão, foi possível fazer também a Figura 4.20. A Figura 4.21 mostra a evolução dos coeficientes aerodinâmicos. Já a Tabela 4.4 apresenta o erro do valor do coeficiente de arrasto

obtido no VAT se comparado com o valor teórico obtido pelo MATLAB ®.

Tabela 4.4: Erro da simulação no VAT comparado com o resultado teórico

	cd	Erro
VAT	0.008560	1.08%
Terórico	0.008654	-

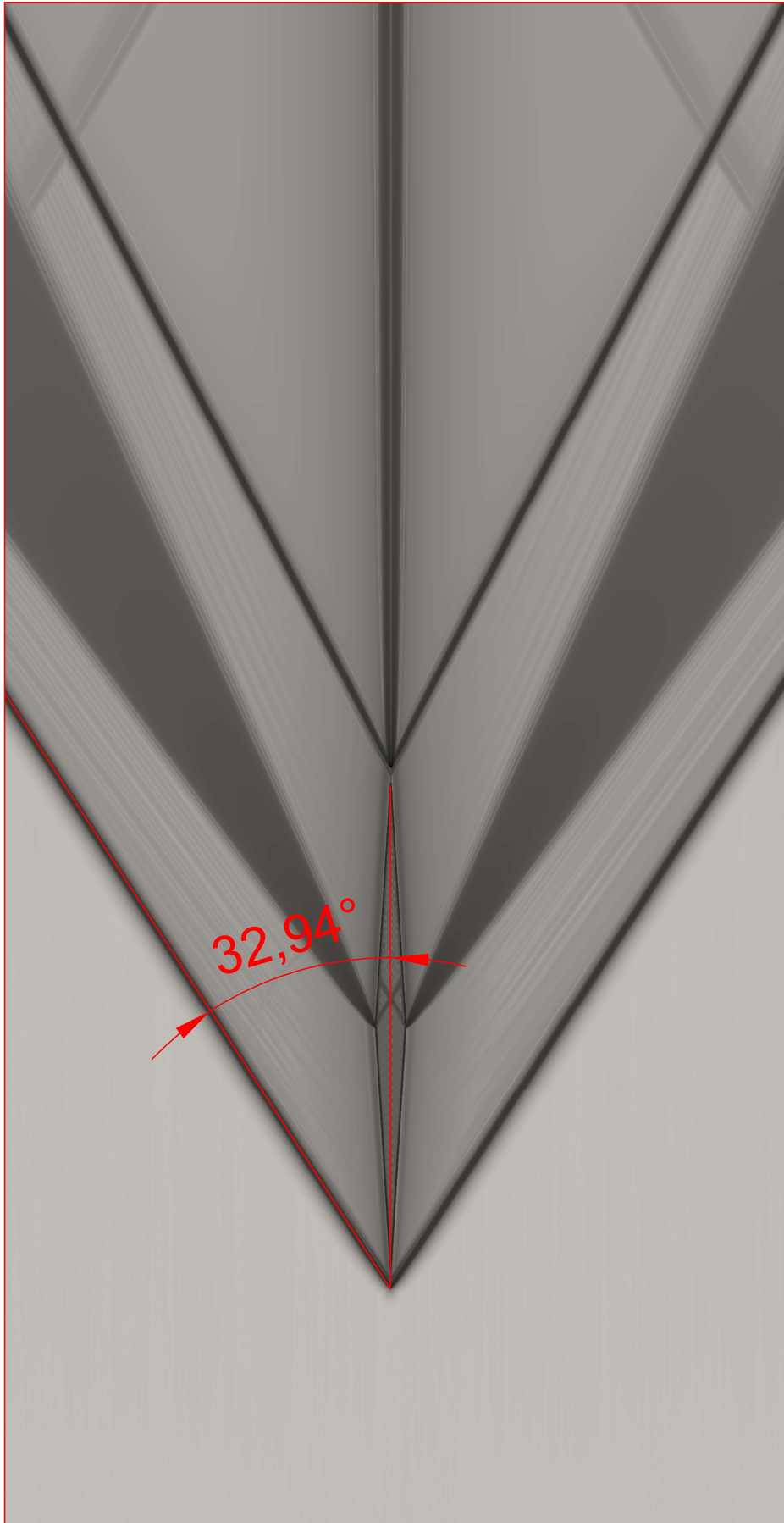


Figura 4.19: Escoamento simulado no VAT. A imagem mostra a variável $\beta_t = \sqrt[10]{\nabla T}$ em escala de cinza no fundo. O ângulo do choque oblíquo simulado coincide com valor calculado

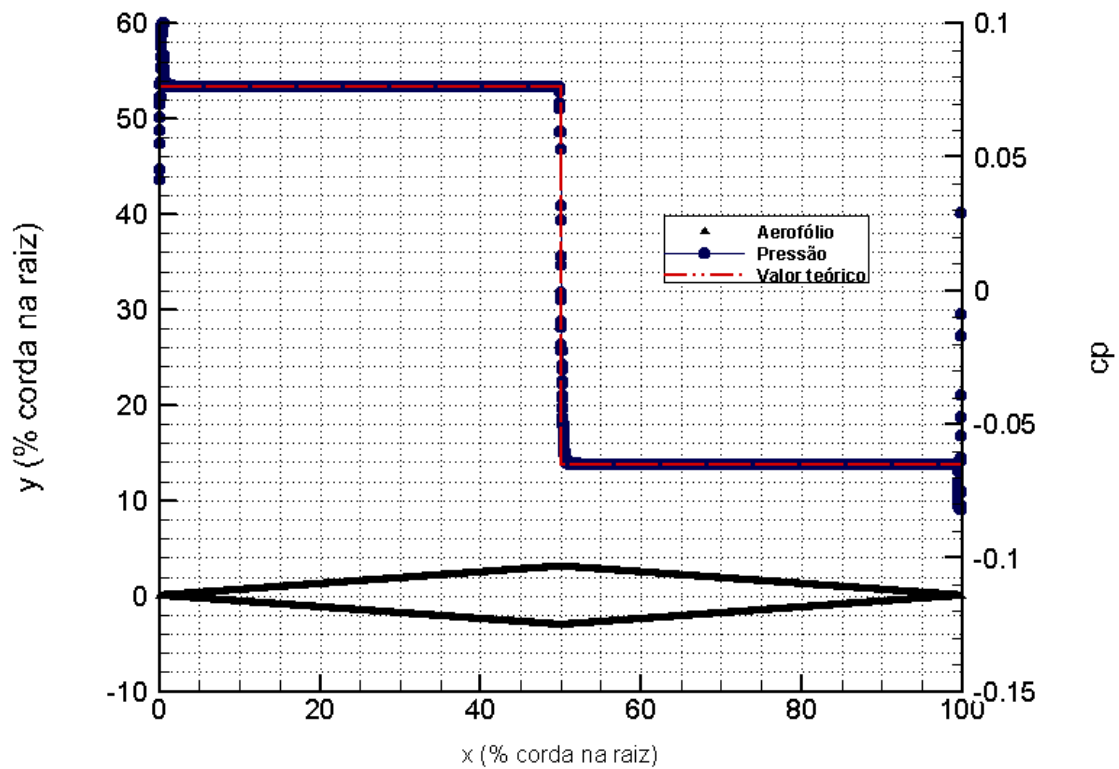


Figura 4.20: Distribuição de pressão sobre o aerofólio na simulação no VAT

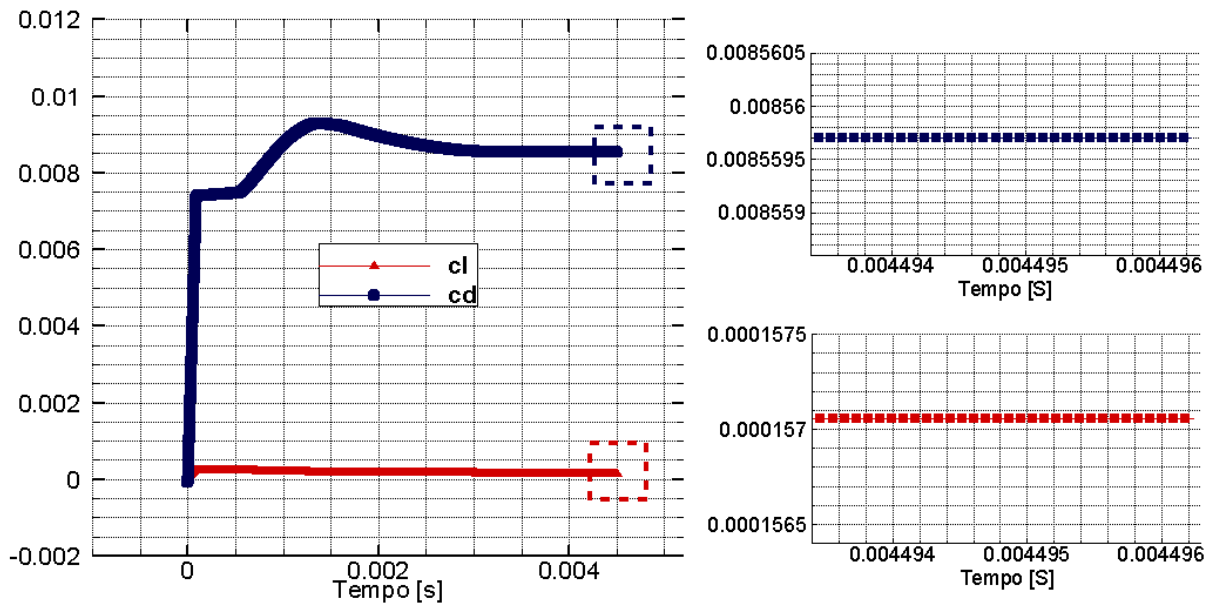


Figura 4.21: Coeficiente de arrasto, sustentação e momento para a simulação no VAT

4.2.4 Comparação entre VAT e OpenFOAM

Calculou-se tempo total de simulação do maior caso sem malha adaptativa na configuração que foi a mais otimizada, que é a de 128 subdivisões, e dividiu-se pelo número de volumes e pelo número de iterações. Esse cálculo contou o tempo em que o OpenFOAM levou para gravar o resultado de diferentes tempos de simulação. O mesmo foi feito para o VAT na configuração de maior número de elementos, e obteve-se então a Figura 4.22. Nessa figura nota-se que o VAT leva aproximadamente 8,7 vezes menos tempo por elemento de discretização e por iteração.

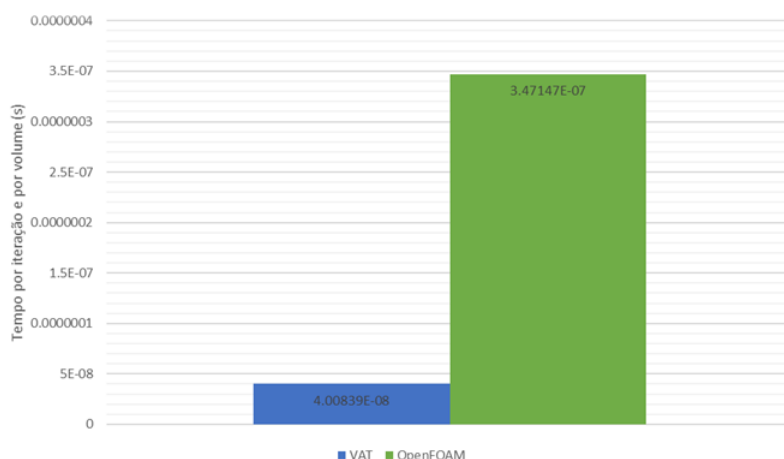


Figura 4.22: Tempo total de simulação por número de volumes e por iteração para o VAT e para o OpenFOAM

Esse resultado pode ser explicado em parte pelo fato de que o VAT utiliza diretivas para a alocação do caso na memória rápida, a nível do *cache* do processador. Além disso, o OpenFOAM utiliza somente a biblioteca MPI, de memória distribuída. Isso leva a um maior tempo de comunicação entre os processadores. Também pode-se explicar pelo fato de que o OpenFOAM utiliza métodos implícitos, que levam um maior tempo por iteração.

Em compensação, o tempo de execução gasto para atingir-se o escoamento permanente no VAT é maior que no OpenFOAM, pois o VAT necessita de menores passos de tempo, por ser um método explícito. Com base nos gráficos de coeficiente de arrasto por tempo de simulação, é possível considerar que em ambos os softwares atingiu-se o estado permanente em 0,003 s. No caso com maior número de volumes, o OpenFOAM levou cerca de 11 horas, ao passo que o VAT levou cerca de 112h para atingir o escoamento permanente.

Por fim, a Tabela 4.5 fornece a comparação direta entre os erros em cada *software*. Os erros estão próximos entre si. Isso sugere que não há necessidade de se saber a priori as características do escoamento, como o ângulo do choque, para realizar refinamento manual da malha, uma vez que no VAT a malha é suficientemente refinada para se obter as variações de pequena ordem que o choque possui, e o OpenFOAM possui mecanismos para refinamento automático em regiões de alto gradiente.

Tabela 4.5: Comparação entre os erros das simulações com o VAT e com o OpenFOAM

	Erro
VAT	1,08%
OpenFOAM sem malha adaptativa	0,94%
OpenFOAM com malha adaptativa	0,10%

4.3 Simulação da asa 3D no OpenFOAM

Para a validação do escoamento tridimensional, utilizou-se os resultados presentes em (12). Dessa forma, foi feita uma malha em torno da geometria de uma asa no formato de delta, cujos perfis são aerofólios de dupla cunha, com a espessura máxima deslocada à montante e com ângulo de enflexamento igual a 28 e 45 graus. Permitiu-se o refinamento automático nas regiões de gradiente de velocidade entre 36 mil e 40 mil s^{-1}

A Figura 4.23 mostra, para fins de análise qualitativa, algumas linhas de corrente do escoamento supersônico ($Ma = 1.6$) simulado. Percebe-se que as linhas de corrente inicialmente defletem-se no choque para o sentido da ponta da asa, e logo após a expansão elas defletem-se no sentido da raiz da asa, e por fim defletem-se e seguem paralelas.

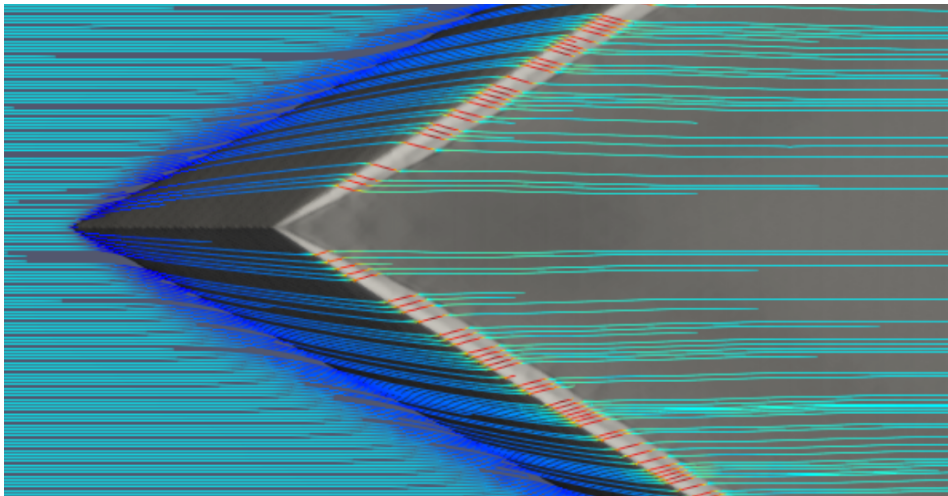


Figura 4.23: Vista do escoamento com linhas de corrente sobre a asa

Na simulação considerou-se a parede com condição de escorregamento total, uma vez que se queria capturar os fenômenos inerciais. A Figura 4.24 mostra a distribuição de pressão em diferentes seções da asa. Pode-se notar, portanto, que o choque ocorre de forma destacada em algumas seções.

A referência (12) fornece a distribuição de pressão na seção a 25,5% da semi envergadura. A Figura 4.25 mostra a distribuição de pressão na seção para a asa com enflexamento de 28 e 45 graus e faz uma comparação com os resultados experimentais. Nota-se que há uma satisfatória

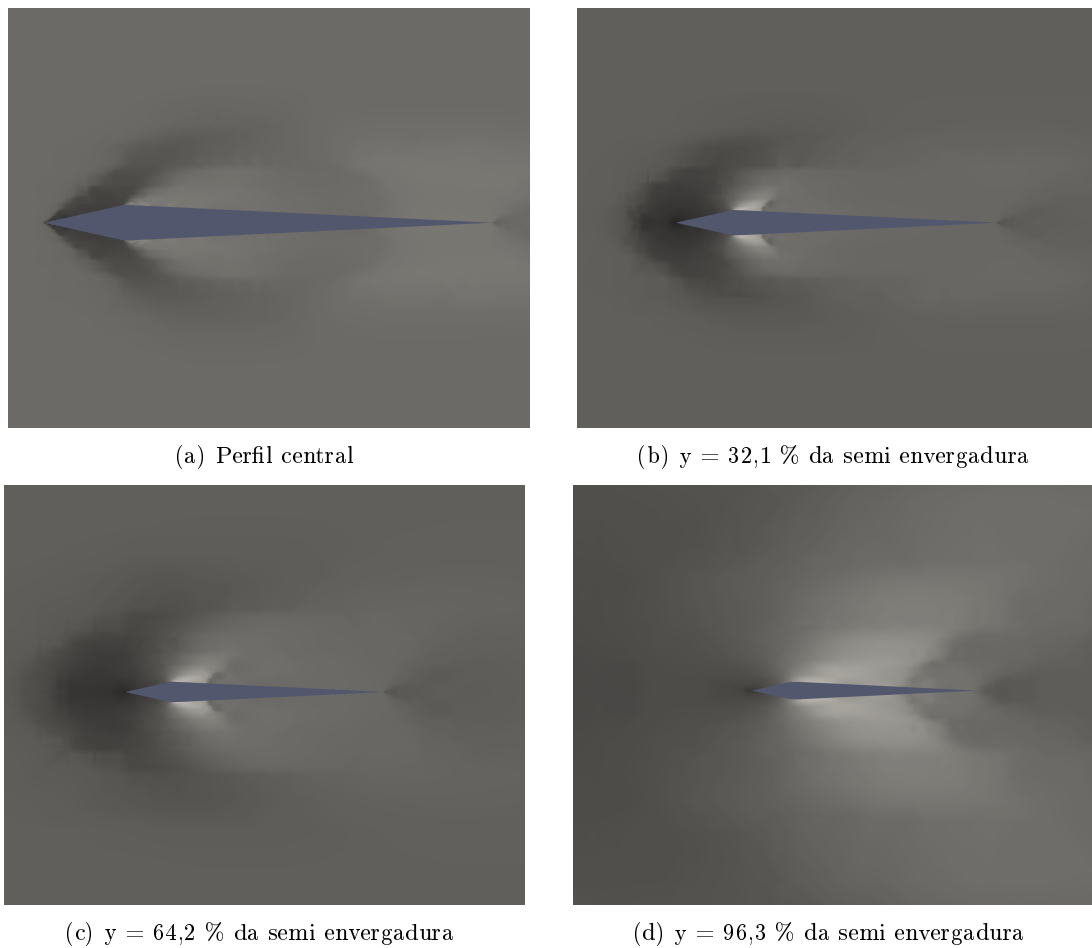
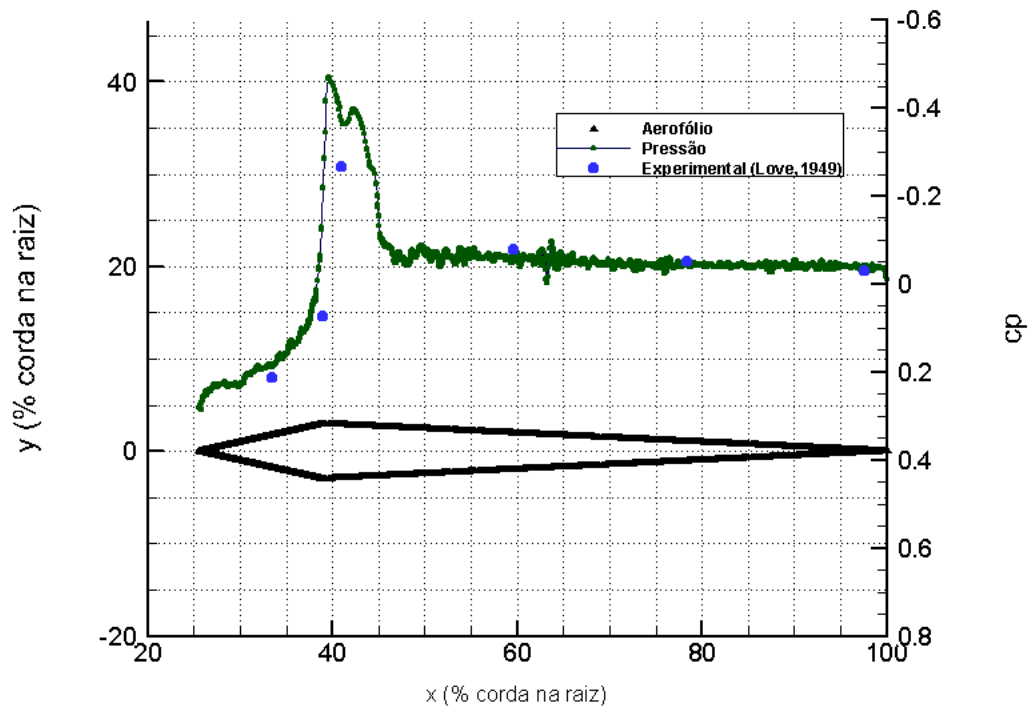
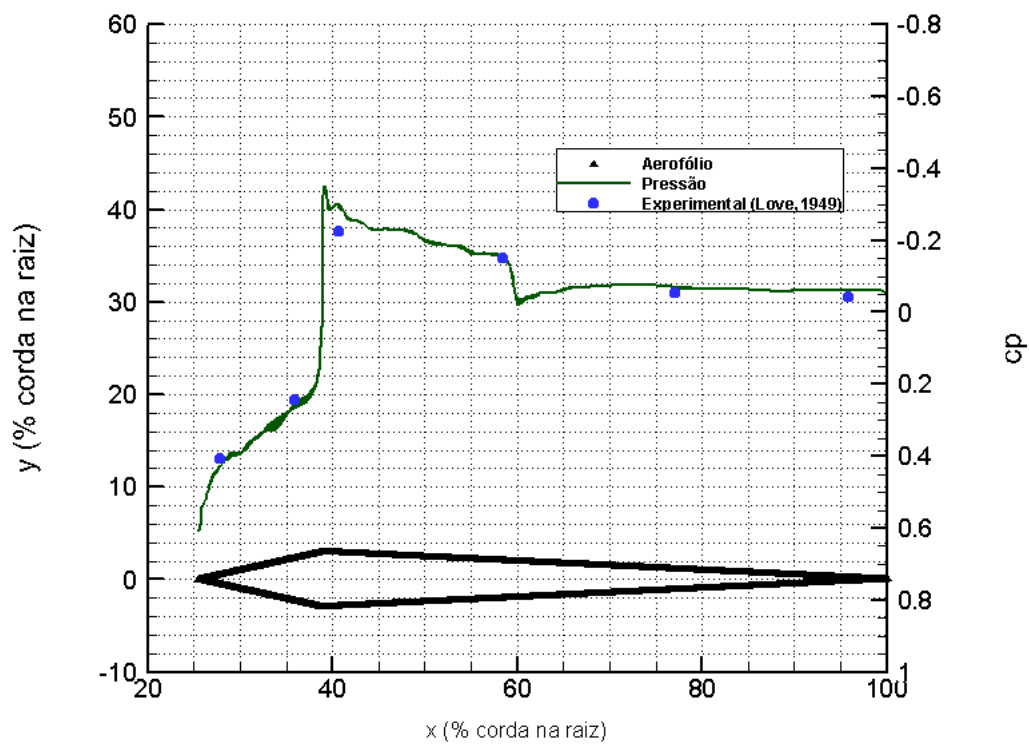


Figura 4.24: Campo de pressão em diferentes seções da asa

relação entre o experimento e a simulação, uma vez que a tendência de aumento ou diminuição da pressão foi capturada.



(a) Asa com enflexamento de 28 graus



(b) Asa com enflexamento de 45 graus

Figura 4.25: Coeficiente de pressão obtido pela simulação e obtido experimentalmente em Love (1949) para a seção $y = 0.255$ da envergadura

Capítulo 5

Conclusão

A indústria aeronáutica caminha para a utilização de *SSTs* em um novo patamar de eficiência energética e de nível de ruído comparado com a geração passada do *Concorde*. Entre outras tecnologias, a dinâmica dos fluidos computacional está em desenvolvimento acelerado pela era digital, e auxilia esse desenvolvimento diferenciado de novos *SSTs*. Muitos desafios nessa área estão aptos a serem resolvidos por essa ferramenta, mas para tirar maior proveito do desenvolvimento dos *hardwares*, os *softwares* de *CFD* necessitam estar adaptados aos novos ambientes *HPC*, principalmente no que se refere ao desempenho em computação paralela. No momento atual existem diferentes *softwares* *CFD* disponíveis, sendo alguns comerciais e outros livres (*OpenSource*). Embora as alternativas *OpenSources* não entreguem ainda a mesma performance como as plataformas comerciais, os *softwares* livres possuem a capacidade de superar os comerciais na razão de performance e custo.

Diante disso, o presente trabalho se propôs a descrever o desempenho do CFD de código livre *OpenFOAM* no *cluster* do CAALab no que se refere à precisão e custo computacional em escoamentos supersônicos externos, comparando-se com os resultados do *software* VAT, desenvolvido pelo próprio laboratório. Foi feita a compilação do OpenFOAM de forma otimizada para a arquitetura do *cluster*. Para o caso bidimensional, foi desenvolvido uma rotina de cálculo do escoamento sobre o aerofólio dupla cunha e comparou-se com o resultado da simulação com e sem malha adaptativa. Já no caso tridimensional, utilizou-se o resultado apresentado em (12), obtido em túnel de vento a partir de asas cujos perfis são de dupla cunha. Quanto à análise do custo computacional, foi realizada uma análise de escalabilidade do OpenFOAM, variando-se a quantidade de subdivisões do domínio e o número de elementos.

Obteve-se precisão de 0,94% no caso sem malha adaptativa e de 0,10% no caso de malha adaptativa na simulação bidimensional do OpenFOAM, enquanto o resultado do VAT obteve precisão de 1,08%. Em ambas as simulações, o ângulo do choque está de acordo com a teoria. Na análise de escalabilidade, o OpenFOAM não apresentou bom desempenho. Pela Fig. 4.6, nota-se que para problemas menores a estratégia de paralelização é menos eficiente que para problemas maiores. Além disso, o resultado sugere também que uma razão adequada para o presente caso seria de 80 mil elementos por processador. Por fim, na simulação tridimensional, o OpenFOAM demonstrou

um desempenho inferior, e a simulação se tornou pouco viável, uma vez que o tempo por iteração estava na ordem de 30 segundos. Notou-se que a estratégia de controle automático de resolução da malha acarreta um prejuízo significativo no tempo de processamento, e uma estratégia adequada seria de aplicar o refinamento automático após atingir-se a convergência com a malha menos densa. Contudo, o resultado seguiu a tendência apresentada em (12), como pode ser visto pela Fig. 4.25.

Como trabalho futuro para otimização do desempenho do OpenFOAM, podem ser elencados 3 pontos. O primeiro seria avaliar o desempenho do software com diferentes opções de métodos de solução de sistemas lineares, de forma a minimizar a comunicação entre processadores e com a utilização possível de *Local Time Stepping*, para que cada processador dependa menos da sincronia com os outros. Outro ponto seria a adoção de paradigmas de memória compartilhada, como o *openMP*, de forma a diminuir a também comunicação entre os processadores em um mesmo nó. Por fim, apesar de o OpenFOAM ter sido compilado de maneira otimizada para o *cluster* do CAALab, ele não possui diretivas para utilização da vetorização e da memória rápida do sistema, e por isso seria adequado a implementação dessas diretivas.

REFERÊNCIAS BIBLIOGRÁFICAS

- 1 ANDERSON, J. D. *Modern compressible flow: with historical perspective*. 3. ed. New York: McGraw-Hill, 2003. (McGraw-Hill Series in Aeronautical and Aerospace Engineering).
- 2 WIKIPEDIA. *List of supersonic aircraft — Wikipedia, The Free Encyclopedia*. 2017. Acesso em 15 abr. 2018. Disponível em: <https://en.wikipedia.org/w/index.php?title=List_of_supersonic_aircraft&oldid=799791623>.
- 3 SMITH, M. J. *Aircraft noise*. Cambridge: Cambridge University Press, 2004. v. 3.
- 4 WIKIPEDIA. *Bell X-1 — Wikipedia, a enciclopedia livre*. 2017. Acessado em 15 abr. 2018. Disponível em: <https://pt.wikipedia.org/w/index.php?title=Bell_X-1&oldid=49595045>.
- 5 WIKIPEDIA. *Concorde — Wikipedia, a enciclopedia livre*. 2018. Acesso em 15 abr. 2018. Disponível em: <<https://pt.wikipedia.org/w/index.php?title=Concorde&oldid=51880349>>.
- 6 MOORE, G. E. *Moore's Law: how overall processing power for computers will double every two years*. 2018. Disponível em: <<http://www.mooreslaw.org/>>.
- 7 HIRSCH, C. *Numerical computation of internal and external flows: The fundamentals of computational fluid dynamics*. 2. ed. Oxford: Elsevier, 2007.
- 8 NASA. *New NASA X-Plane Construction Begins Now*. 2018. Acesso em 29 abr. 2018. Disponível em: <<https://www.nasa.gov/lowboom/new-nasa-x-plane-construction-begins-now>>.
- 9 MOUKALLED, F. et al. *The finite volume method in computational fluid dynamics: An advanced introduction with openfoam and matlab*. Switzerland: Springer, 2016. v. 113. (Fluid Mechanics and Its Applications, v. 113).
- 10 FERRI, A. Experimental results with airfoils tested in the high-speed tunnel at guidonia. *National Advisory Committee for Aeronautics; Washington, DC, Estados Unidos*, 1940. NACA Technical Memorandum 946.
- 11 VINCENTI, W. G. Comparison between theory and experiment for wings at supersonic speeds. *National Advisory Committee for Aeronautics; Washington, DC, Estados Unidos*, 1951. NACA Technical Report 1033.
- 12 LOVE, E. S. Investigations at supersonic speeds of 22 triangular wings representing two airfoil sections for each of 11 apex angles. *National Advisory Committee for Aeronautics; Washington, DC, Estados Unidos*, 1949. NACA Research Memorandum L9D07.
- 13 LIEPMAN, H.; ROSHKO, A. *Elements of gas dynamics*. New York: Dover, 1957.
- 14 MISERDA, R. B. et al. Direct computation of noise generated by complex geometries using a high-order immersed boundary method. In: AIAA AEROACOUSTICS CONFERENCE, 30., 2009, Miami. *Proceedings...* Miami: American Institute of Aeronautics and Astronautics, 2009. p. 3181.

- 15 BONDAREV, A. E.; KUVSHINNIKOV, A. E. Analysis of the accuracy of openfoam solvers for the problem of supersonic flow around a cone. In: INTERNATIONAL CONFERENCE ON COMPUTATIONAL SCIENCE, 18., 2018, Wuxi. *Proceedings...* [S.l.]: Springer, 2018. p. 221–230.
- 16 GUTIERREZ, L.; TAMAGNO, J.; ELASKAR, S. High speed flow simulation using openfoam. *Mecanica Computacional*, v. 31, p. 2939–2959, 2012.
- 17 ASPROULIAS, I. *RANS modelling for compressible turbulent flows involving shock wave boundary layer interactions*. Tese (Doutorado) — University of Manchester, 2014.
- 18 KHODADADI, A. R.; MALEKBALA, M. R.; KHODADADI, A. F. Evaluate shock capturing capability with the numerical methods in openfoam. *Thermal Science*, v. 17, n. 4, p. 1255–1260, 2013.
- 19 JASAK, H.; GOSMAN, A. Automatic resolution control for the finite-volume method, part 2: Adaptive mesh refinement and coarsening. *Numerical Heat Transfer: Part B: Fundamentals*, Taylor & Francis, v. 38, n. 3, p. 257–271, 2000.
- 20 ESPINOZA, D. E.; SCANLON, T. J.; BROWN, R. E. Validation of tools to accelerate high-speed cfd simulations using openfoam. In: AIAA INTERNATIONAL SPACE PLANES AND HYPERSONIC SYSTEMS AND TECHNOLOGIES CONFERENCE, 20., 2015, Glasgow. *Proceedings...* [S.l.]: American Institute of Aeronautics and Astronautics, 2015. p. 3566.
- 21 LARSSON, T. et al. Enhancing computational aero-acoustic processes for ground vehicles resolving open source cfd. In: THE 13TH OPENFOAM WORKSHOP, 13., 2018, Shanghai. *Proceedings...* [S.l.], 2018. p. 1–4.
- 22 PELLEGRINI, F. *Software package and libraries for sequential and parallel graph partitioning, static mapping and clustering, sequential mesh and hypergraph partitioning, and sequential and parallel sparse matrix block ordering*. 2018. Disponível em: <<http://www.labri.fr/perso/pelegrin/scotch/>>.
- 23 ANDERSON, J. D.; WENDT, J. *Computational fluid dynamics: An introduction*. 3. ed. Berlin: Springer, 1995.
- 24 Wikipedia contributors. *NACA airfoil* — *Wikipedia, The Free Encyclopedia*. 2018. <https://en.wikipedia.org/w/index.php?title=NACA_airfoil&oldid=866926852>. [Online; accessed 18-November-2018].
- 25 ANDERSON, N. *Oblique shocks and expansions waves*. 2018. Disponível em: <<http://nikander.github.io/compflow/Anderson/Chapter4/>>.
- 26 SERAFINI, J. S. Impingement of water droplets on wedges and double-wedge airfoils at supersonic speeds. *National Advisory Committee for Aeronautics; Washington, DC, Estados Unidos*, 1954. NACA Technical Report 1159.
- 27 GUTIERREZ, B. P. *Numerical simulation of nonlinear waves in gas dynamics and rotor-stator interaction noise of aeronautic turbofan engines*. Tese (Doutorado) — UnB, 2016.
- 28 JAMESON, A.; SCHMIDT, W.; TURKEL, E. Numerical solution of the euler equations by finite volume methods using runge kutta time stepping schemes. In: 14TH FLUID AND PLASMA DYNAMICS CONFERENCE, 14., 1981. *Proceedings...* [S.l.], 1981. p. 1259.
- 29 ANDERSON, D.; TANNEHILL, J. C.; PLETCHER, R. H. *Computational fluid mechanics and heat transfer*. [S.l.]: CRC Press, 2016.

- 30 YEE, H. C. Explicit and implicit multidimensional compact high-resolution shock-capturing methods: formulation. *JOURNAL OF COMPUTATIONAL PHYSICS*, v. 113, p. 216–232, 1997.
- 31 PATANKAR, S. *Numerical heat transfer and fluid flow*. [S.l.]: CRC press, 1980.
- 32 ISSA, R. I. Solution of the implicitly discretised fluid flow equations by operator-splitting. *Journal of computational physics*, Elsevier, v. 62, n. 1, p. 40–65, 1986.
- 33 FERZIGER, J. H.; PERIC, M. *Computational methods for fluid dynamics*. [S.l.]: Springer Science & Business Media, 2012.
- 34 MOECKEL, W. *Theoretical Aerodynamic Coefficients of Two-dimensional Supersonic Biplanes*. [S.l.], 1947.
- 35 CARVALHO, A. E. B. de. *Simulacao numerica de Aerofolios com Oscilacao Vertical em Regime Supersonico*. 2018. Departamento de Engenharia Mecanica, UnB. Trabalho de Conclusao de Curso em Engenharia Mecanica.

ANEXOS

I. ROTINA DE CÁLCULO DO ESCOAMENTO SOBRE O AEROFÓLIO DE DUPLA CUNHA

A função principal é apresentada a seguir.

```
%Program for calculation of busemann airfoil.
% it uses:
% - expansion_wave function
% - shock_wave function
% - theta_max function

function [alphadeg, x , c_l, c_d, c_l_c_d, v, a, m_inf, P_u_1, P_u_2, m_u_1, m_u_2, P_d_1, P_d_2] = busemann(alpha, h, c, t, r_mt)

%%
%inputs

T_inf = 300; %27+273.0; %[k]
P_inf = 101325; %100000.000000000000000000; %[Pa]
gamma = 1.4000000000000000; %[] 1.3
R = 287; %287.0530; %[]
T = T_inf; %[]
P = P_inf; %[]

a = sqrt(gamma*R*T); %[m/s]
v = 694.377418987686; %m_inf*a; %656.8; %[m/s] 650;
m_inf = v/a %1.9779923000000000000000; %[]

rho_inf = P/(R*T) %1.2250000000000000; %[kg/m3]
h = 0.5; %[] height ratio (0<h<1). for symetric airfoil, h=0.5
c = 1.0000000000000000; %[m]
t = tand(theta/2)*c; %0.1000000000000000; % %[] %0.0611626200150000; %[m]
x = 2*(atand(t/c));
t_u = t*h; %[m]
t_d = t*(1-h); %[m]
r_mt = 0.5000000000000000; %[] maximum thickness in percent of the chord
alphadeg = alpha;
alpha = alpha*pi/180.00000000000000; %[rad]
```

```

epsilon_u1 = atan((t_u)/(r_mt*c));
epsilon_d1 = atan((t_d)/(r_mt*c));
epsilon_u2 = atan((t_u)/(c*(1-r_mt)));
epsilon_d2 = atan((t_d)/(c*(1-r_mt)));
%epsilon = atan(t/c)%[rad]
%epsilon= 5*pi/180.0;
sw_1 = 0;%shock wave flag
sw_2 = 0;%shock wave flag
ew_1 = 0;%expansion wave flag
ew_2 = 0;%expansion wave flag
precision = 1.0e-12;
r = 8.3145/0.0289645; %gas real constant [SI]
nu_air = 16e-6; %[m/s^2] viscosity

% see if there is destacation
d= 0;

[theta_max] = theta_max_function (m_inf,gamma,precision);
theta_max_D = theta_max*180.0/pi;

%extradorso
theta_u_1 = epsilon_u1-alpha;
if theta_u_1>theta_max
    s_error = sprintf('destacation ocurred on upper surface region 1');
    disp(s_error);
    d=1;
end

%intradorso

theta_d_1 = epsilon_d1+alpha;
if theta_d_1>theta_max
    s_error = sprintf('destacation ocurred on down surface region 1');
    disp(s_error);
    d=1;
end

if d ==1
alpha_min = epsilon_u1-theta_max;
alpha_max = theta_max-epsilon_d1;
s_alphaboundary = sprintf('alpha must be between %0#.4e [Degree] and %0#.4e [Degree]',(alp

```

```

disp(s_alphaboundary)
else
%%
%calculation
disp('Calculation: ')
%extradorso
R=1;

sw_1=0;
ew_1=0;

%region 1

theta_u_1= epsilon_u1-alpha;
s_theta_u_1 = sprintf(' theta_u_1 = %0#.15e [degree]',(theta_u_1*180.0/pi));
disp(s_theta_u_1)

%see if there is a shock or expansion
if theta_u_1<0
display('-expansion wave on upper surface region 1')
ew_1=1;
else
if theta_u_1>theta_max
display('alpha exceded the maximum absolute angle, destacation ocurred. stop p
else
disp('-shock wave on upper surface region 1')
sw_1=1;
end
end

%shock wave calculation
if sw_1==1
[mn_inf,mn_u_1,m_u_1,beta_1,P_u_1] = shock_wave(precision,m_inf,gamma,P_inf,theta_u_1,0,R);
end

%expansion wave calculation
if ew_1==1
[m_u_1,P_u_1] = expansion_wave (precision,gamma,m_inf,P_inf,theta_u_1,0,R);
%flow variables after expansionwave
%mn_1 = m_inf*sin(beta_1);
%P_u_1 = P_inf*(1+(2*gamma/(gamma+1))*(mn_1^2-1));%[Pa]

```

```

end

%region 2

%always expansion wave
%theta_u_2 = 2*epsilon; %[rad]
theta_u_2 = epsilon_u1+epsilon_u2; %[rad]

%expansion wave calculation
disp('-expansion wave on upper surface region 2')

[m_u_2,P_u_2] = expansion_wave (precision,gamma,m_u_1,P_u_1,theta_u_2,theta_u_1,R);

%intradorso
R=2;
sw_1=0;
ew_1=0;

%Region 1

theta_d_1 = -epsilon_d1-alpha;
s_theta_d_1 = sprintf(' theta_d_1 = %0#.15e [degree]',(theta_d_1*180.0/pi));
disp(s_theta_d_1)
%see if there is a shock or expansion
if theta_d_1>0
    display('-expansion wave on down surface region 1')
    ew_1=1;
else
    if theta_d_1>theta_max
        display('alpha exceded the maximum absolute angle, destacation ocurred. stop p
    else
        disp('-shock wave on down surface region 1')
        sw_1=1;
    end
end

%shock wave calculation
if sw_1==1
    [mn_inf,mn_d_1,m_d_1,beta_1,P_d_1] = shock_wave(precision,m_inf,gamma,P_inf,abs(th
end

```

```

%expansion wave calculation
if ew_1==1
    [m_d_1,P_d_1] = expansion_wave (precision,gamma,m_inf,P_inf,theta_d_1,0,R);
    %flow variables after expansionwave
    %mn_1 = m_inf*sin(beta_1);
    %P_u_1 = P_inf*(1+(2*gamma/(gamma+1))*(mn_1^2-1));%[Pa]
end

%Region 2

%always expansion wave
%theta_d_2 = 2*epsilon; %[rad]
theta_d_2 = -epsilon_d1-epsilon_d2; %[rad]

%expansion wave calculation
disp('-expansion wave on down surface region 2')

[m_d_2,P_d_2] = expansion_wave (precision,gamma,m_d_1,P_d_1,theta_d_2,theta_d_1,R);

%region 3
disp('-shock on region 3')
%always shock
[mn_u_2,mn_u_3,m_u_3,beta_3,P_u_3] = shock_wave(precision,m_u_2,gamma,P_u_2,epsilon_u2

%%
%outputs

%net forces
A = (P_u_1-P_u_2)*t_u+(P_d_1-P_d_2)*t_d; % [N]
N = (P_d_1-P_u_1)*(c*r_mt)+(P_d_2-P_u_2)*(c*(1-r_mt)); % [N]
D = A*cos(alpha)+N*sin(alpha);
L = N*cos(alpha)-A*sin(alpha);
a_inf = sqrt(gamma*r*T_inf);
u_inf = m_inf*a_inf;
c_d = D/(rho_inf*u_inf^2/2);
c_l = L/(rho_inf*u_inf^2/2);
c_l_c_d = c_l/c_d;
end

```

A função para o cálculo da onda de choque é apresentado a seguir

```

%Program for calculation of shock wave characteristics

function [mn_1,mn_2,m_2,beta,P_2] = shock_wave (precision,m_1,gamma,P_1,theta)

%shock wave calculation

%calculation of beta, NEWTON-RHPASON
%beta = theta; %[rad] inicial guest
beta = precision; %[rad] inicial guest
i = 0; %indicator of convergence
while i==0
    f=tan(theta)-2*(tan(beta))^( -1)*(m_1^2*sin(beta)^2-1)/(m_1^2*(gamma+cos(2*beta))+2)
    f_derived = -4*m_1^2*cos(beta)^2/(m_1*(cos(2*beta)+gamma)+2)-(4*m_1^2*sin(2*beta)*
    if abs(f)<precision
        i=1;
        %beta_u_1 = beta;
        s_beta = sprintf(' converged, beta = %0#.15e [degree]',(beta*180.0/pi));
        disp(s_beta);
    else
        beta = beta - f/f_derived;
    end
end
%flow variables after shockwave
mn_1 = m_1*sin(beta);
P_2 = P_1*(1+(2*gamma/(gamma+1))*(mn_1^2-1)); %[Pa]
mn_2 =sqrt((mn_1^2+(2/(gamma-1)))/((2*gamma/(gamma-1))*mn_1^2-1));
m_2 = mn_2/sin(beta-theta);

```

A função do cálculo do leque de expansão é apresentada a seguir.

```

%Program for calculation of the expansion wave characteristics

function [m_2,P_2] = expansion_wave (precision,gamma,m_1,P_1,theta,theta0,R)

    thetad = abs(theta)*180.0/pi;
    theta0d = theta0*180.0/pi;
%calculation of m_2
i=0;%indicator of convergence
m_2 = m_1*1.001; %inicial guest
while i==0

```

```

f = sqrt((gamma+1)/(gamma-1))*(180.0/pi)*atan(sqrt((m_2^2-1)*(gamma-1)/(gamma+1)))
f_derived = (1007958012753983*m_2*((gamma + 1)/(gamma - 1))^(1/2)*(gamma - 1))/(17
if abs(f)<precision
    i=1;
    s_m_2 = sprintf(' converged, m_2 = %0#.15e',(m_2));
    disp(s_m_2)
else
    m_2 = m_2 - f/f_derived;
end
end

P_2 = P_1*((1+(gamma-1)*m_1^2/2)/(1+(gamma-1)*m_2^2/2))^(gamma/(gamma-1));
if R==1
    mu_1 = theta0d+asin(1/m_1)*180.0/pi();
    mu_2 = theta0d-thetad+asin(1/m_2);
else
    mu_1 = theta0d-asin(1/m_1)*180.0/pi();
    mu_2 = theta0d+thetad-asin(1/m_2);
end
s_mu = sprintf(' expansion angles are mu_1 = %0#.15e and mu_2 = %0#.15e [degree]',(mu_
disp(s_mu)

```

Por fim, a função que verifica a ocorrência de destacamento do choque é apresentada a seguir

```
% Program for calculation of the theta max
```

```
function [theta] = theta_max_function (m,gamma,precision)
```

```

i = 0;
beta = 70*pi/180.0;
while i==0
    %f = ((2*m^2*cos(beta))/(tan(beta)*((cos(2*beta) + 7/5)*m^2 + 2)) - (2*(sin(beta)*m^2
f = ((4*m^2*sin(2*beta)*(m^2*sin(beta)^2 - 1))/(tan(beta)*((gamma + cos(2*beta))*m^2 +
%f_derived = - ((2*m^2*sin(beta))/(tan(beta)*((cos(2*beta) + 7/5)*m^2 + 2)) - (2*(2*si
f_derived = - ((4*m^2*sin(beta)^2)/(tan(beta)*((gamma + cos(2*beta))*m^2 + 2)) - (4*m^
if abs(f)<precision
    i=1;
    %s_beta = sprintf('converged, beta = %0#.15e [degree]',(beta*180.0/pi));
    %disp(s_beta);
else
    beta = beta - f/f_derived;
end
end

```

```
end

theta = atan(2*tan(beta)^(-1)*((m^2*sin(beta)^2-1)/(m^2*(gamma+cos(2*beta))+2)));
end
```