

**SIMULAÇÕES NUMÉRICAS  
EM CÓDIGO ABERTO  
DE ESCOAMENTOS TURBULENTOS SOBRE  
ASAS MULTIELEMENTO EM EFEITO SOLO  
PARA GERAÇÃO DE DOWNFORCE**

**MATHEUS RIBEIRO VIDAL**

**PROJETO DE GRADUAÇÃO EM ENGENHARIA MECÂNICA  
DEPARTAMENTO DE ENGENHARIA MECÂNICA**

**FACULDADE DE TECNOLOGIA**

**UNIVERSIDADE DE BRASÍLIA**

**UNIVERSIDADE DE BRASÍLIA  
FACULDADE DE TECNOLOGIA  
DEPARTAMENTO DE ENGENHARIA MECÂNICA**

**SIMULAÇÕES NUMÉRICAS  
EM CÓDIGO ABERTO  
DE ESCOAMENTOS TURBULENTOS SOBRE  
ASAS MULTIELEMENTO EM EFEITO SOLO  
PARA GERAÇÃO DE DOWNFORCE**

**MATHEUS RIBEIRO VIDAL**

**Orientador: PROF. DR. BRAULIO GUTIERREZ PIMENTA,  
ENM/UNB**

**PROJETO DE GRADUAÇÃO EM ENGENHARIA MECÂNICA**

**PUBLICAÇÃO ENM.PG  
BRASÍLIA-DF, 26 DE JULHO DE 2023.**



**UNIVERSIDADE DE BRASÍLIA  
FACULDADE DE TECNOLOGIA  
DEPARTAMENTO DE ENGENHARIA MECÂNICA**

**SIMULAÇÕES NUMÉRICAS  
EM CÓDIGO ABERTO  
DE ESCOAMENTOS TURBULENTOS SOBRE  
ASAS MULTIELEMENTO EM EFEITO SOLO  
PARA GERAÇÃO DE DOWNFORCE**

**MATHEUS RIBEIRO VIDAL**

PROJETO DE GRADUAÇÃO SUBMETIDO AO DEPARTAMENTO DE ENGENHARIA MECÂNICA DA FACULDADE DE TECNOLOGIA DA UNIVERSIDADE DE BRASÍLIA, COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE ENGENHEIRO MECÂNICO.

APROVADA POR:

Prof. Dr. Braulio Gutierrez Pimenta, ENM/UnB  
Orientador

Prof. Dr. Adriano Possebon Rosa, ENM/UnB  
Examinador interno

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

**BRASÍLIA, 26 DE JULHO DE 2023.**

## **FICHA CATALOGRÁFICA**

MATHEUS RIBEIRO VIDAL

**Simulações Numéricas em Código Aberto de Escoamentos Turbulentos sobre Asas Multielemento em Efeito Solo para Geração de Downforce**

**2023, 228p., 201x297 mm**

(ENM/FT/UnB, Engenheiro Mecânico, Engenharia Mecânica, 2023)

Projeto de Graduação - Universidade de Brasília

Faculdade de Tecnologia - Departamento de Engenharia Mecânica

## **REFERÊNCIA BIBLIOGRÁFICA**

MATHEUS RIBEIRO VIDAL (2023) Simulações Numéricas em Código Aberto de Escoamentos Turbulentos sobre Asas Multielemento em Efeito Solo para Geração de Downforce. Projeto de Graduação em Engenharia Mecânica, Departamento de Engenharia Mecânica, Universidade de Brasília, Brasília, DF, 228p.

## **CESSÃO DE DIREITOS**

AUTOR: Matheus Ribeiro Vidal

TÍTULO: Simulações Numéricas em Código Aberto de Escoamentos Turbulentos sobre Asas Multielemento em Efeito Solo para Geração de Downforce.

GRAU: Engenheiro Mecânico ANO: 2023

É concedida à Universidade de Brasília permissão para reproduzir cópias deste projeto de graduação e para emprestar ou vender tais cópias somente para propósitos acadêmicos e científicos. O autor se reserva a outros direitos de publicação e nenhuma parte deste projeto de graduação pode ser reproduzida sem a autorização por escrito do autor.

---

Matheus Ribeiro Vidal

vidal.enm@gmail.com

# Agradecimentos

*Gostaria de agradecer, primeiramente, à minha família, pedra fundamental no meu desenvolvimento acadêmico, profissional e pessoal. Ao meu pai, Marcelo, minha mãe, Claudia, e Lucas, meu irmão, vocês sempre estiveram ao meu lado, e o suporte e encorajamento que vocês me oferecem foram de essencial importância para que eu pudesse não apenas me dedicar a este trabalho, mas também começar a construir minha vida profissional.*

*Quero também agradecer à minha melhor amiga e parceira de vida, Verônica. Seu carinho e compreensão em todos os momentos me motivam a trabalhar por nosso futuro juntos. Seu apoio nos últimos meses, principalmente, foi um dos principais fatores que possibilitaram que eu finalizasse este trabalho da melhor maneira.*

*Naturalmente, este trabalho não teria sido realizado sem a diligente orientação do meu professor orientador, Braulio Gutierrez Pimenta. Agradeço por esclarecer minhas dúvidas, pelas conversas entusiasmadas sobre motorsport e Fórmula 1 e, principalmente, por ter me aceitado como orientando e auxiliado nesta introdução à atividade acadêmica e ao CFD.*

*Agradeço também à banca avaliadora do meu projeto de graduação, composta pelos professores Adriano Rosa e Roberto Miserda, pelas sugestões, correções e ideias apresentadas ao longo deste trabalho, as quais ajudaram a esculpir este relatório para a melhor forma possível.*

*Aos meus colegas que se tornaram estimados amigos, Eduardo Folster, Valter Gonzaga, Ana Luíza Maia e Lúcio Starling, agradeço pelo companheirismo, trabalhos em equipe, conselhos e conversas. Vocês foram a melhor parte da minha vida universitária.*

*Por fim, agradeço a todos que fizeram parte direta ou indiretamente da minha jornada na Universidade de Brasília: amigos, familiares, colegas e docentes. São pessoas que já faziam ou agora também fazem parte da minha história e, de uma forma ou de outra, moldaram a pessoa que sou hoje.*

*Obrigado.*

*Matheus Ribeiro Vidal*

*"Science is more than a body of knowledge.  
It's a way of thinking.  
A way of skeptically interrogating the universe with  
a fine understanding of human fallibility."  
— Carl Sagan*

# Resumo

Este trabalho apresenta o estabelecimento e a validação de um ciclo completo de simulações em dinâmica dos fluidos computacional, utilizando ferramentas de código aberto (*open-source*). O objetivo principal é demonstrar a viabilidade e a confiabilidade de se utilizar ferramentas de código aberto para simular fenômenos fluidodinâmicos complexos. Para alcançar esse objetivo, o ciclo de simulações é estabelecido utilizando os programas: SALOME para geração de malhas e pré-processamento, SU2 para cálculos numéricos e Paraview para visualização dos resultados.

A geometria escolhida para a validação deste ciclo de simulação é a geometria dos experimentos da tese de doutorado de Zerihan (2001), que consiste em uma asa frontal de um carro de Fórmula 1 dentro de um túnel de vento. O caso foi modelado em *software* CAD, foram geradas malhas tridimensionais e feitas simulações com o intuito de comparar os valores dos coeficientes aerodinâmicos adimensionais obtidos com a literatura.

A metodologia empregada durante o ciclo de simulações é descrita em detalhes, com justificativas por decisões tomadas e observações sobre o fluxo de trabalho nos programas utilizados. Os resultados deste trabalho mostram uma boa capacidade de captura de fenômenos aerodinâmicos complexos e definem uma abordagem confiável para a solução de problemas em dinâmica dos fluidos computacional.

**Palavras-Chave:** Dinâmica dos Fluidos Computacional; Aerodinâmica; SALOME; SU2; Código Aberto, Fórmula 1.

# Abstract

This work presents the establishment and validation of a complete cycle of computational fluid dynamics simulations using open-source tools. The main objective is to demonstrate the feasibility and reliability of using open-source tools to simulate complex fluid dynamic phenomena. To achieve this goal, the simulation cycle is established using the following programs: SALOME for mesh generation and pre-processing, SU2 for numerical calculations, and Paraview for results and data visualization.

The chosen geometry for validating this simulation cycle is the geometry of the experiments in the doctoral thesis by Zerihan (2001), which consists of a Formula 1 car's front wing inside a wind tunnel. The case was modeled in CAD software, three-dimensional meshes were generated, and simulations were conducted to compare the values of the dimensionless aerodynamic coefficients obtained with the literature.

The methodology employed during the simulation cycle is described in detail, with justifications for decisions made and observations on the workflow in the programs used. The results of this work demonstrate a good capability to capture complex aerodynamic phenomena and establish a reliable approach for solving problems in computational fluid dynamics.

**Keywords:** Computational Fluid Dynamics; Aerodynamics; SALOME; SU2; Open-Source; Formula 1.

# SUMÁRIO

<b>1</b>	<b>Introdução</b> .....	<b>1</b>
1.1	MOTIVAÇÃO .....	1
1.2	ESTRUTURA DO TRABALHO .....	3
1.3	OBJETIVOS .....	4
<b>2</b>	<b>Arcabouço Teórico: Mecânica dos Fluidos e Aerodinâmica</b> .....	<b>5</b>
2.1	FORÇAS E MOMENTOS AERODINÂMICOS.....	5
2.2	EQUAÇÕES GOVERNANTES.....	7
2.2.1	EQUAÇÃO DA CONTINUIDADE .....	8
2.2.2	EQUAÇÃO DA QUANTIDADE DE MOVIMENTO .....	9
2.2.3	EQUAÇÃO DA ENERGIA.....	10
2.3	GRUPOS ADIMENSIONAIS DE INTERESSE .....	12
2.3.1	O NÚMERO DE REYNOLDS.....	12
2.3.2	OS COEFICIENTES AERODINÂMICOS.....	13
2.4	CAMADA LIMITE DE VELOCIDADE.....	14
2.4.1	DESCOLAMENTO DE CAMADA LIMITE.....	14
2.4.2	ESPESSURA DE CAMADA LIMITE.....	15
2.5	AERODINÂMICA DE CARROS DE CORRIDA.....	16
2.5.1	ELEMENTOS AERODINÂMICOS DE UM CARRO DE CORRIDA .....	16
2.6	ESCOAMENTOS TURBULENTOS .....	18
<b>3</b>	<b>Arcabouço Teórico: Métodos Computacionais para Mecânica dos Fluidos</b> .....	<b>20</b>
3.1	O MÉTODO DOS VOLUMES FINITOS.....	20
3.2	O PROCESSO DE TRABALHO EM DINÂMICA DOS FLUIDOS COMPUTACIONAL.....	23
3.2.1	PRÉ-PROCESSAMENTO.....	23
3.2.2	SIMULAÇÃO.....	23
3.2.3	PÓS-PROCESSAMENTO .....	23
3.3	FERRAMENTAS DE CÓDIGO ABERTO .....	23
3.4	MODELAGEM DE ESCOAMENTOS TURBULENTOS.....	24
3.4.1	MODELAGEM DIRETA (DNS) .....	24

3.4.2	MODELAGEM POR EQUAÇÕES DE NAVIER-STOKES FILTRADAS (LES).....	24
3.4.3	MODELAGEM POR EQUAÇÕES MÉDIAS DE REYNOLDS (RANS) ..	26
3.4.4	O MODELO DE TURBULÊNCIA SST .....	27
<b>4</b>	<b>Metodologia de Projeto .....</b>	<b>29</b>
4.1	PROPOSTA DO PROJETO .....	29
4.1.1	APRESENTAÇÃO DOS PROGRAMAS UTILIZADOS.....	29
4.1.2	OBSERVAÇÕES SOBRE OS PROGRAMAS UTILIZADOS .....	31
4.2	MODELAGEM DA GEOMETRIA DE ZERIHAN .....	31
4.2.1	COLETA E TRATAMENTO DOS DADOS DE ZERIHAN .....	32
4.2.2	MODELAGEM TRIDIMENSIONAL DO VOLUME DE AR DENTRO DO TÚNEL DE VENTO.....	32
4.2.3	MODELAGEM TRIDIMENSIONAL DA ASA FRONTAL NO INFINITO ..	36
4.3	PRÉ-PROCESSAMENTO.....	37
4.3.1	DETERMINAÇÃO DA GEOMETRIA PARA AS CONDIÇÕES DE CONTORNO DO PROBLEMA.....	37
4.3.2	CONFIGURAÇÃO DE MALHA - ALGORITMO APLICADO .....	38
4.3.3	CONFIGURAÇÃO DE MALHA - HIPÓTESE APLICADA .....	39
4.4	SIMULAÇÃO E PÓS-PROCESSAMENTO .....	39
4.4.1	O CLUSTER AMADEA.....	40
4.4.2	CONFIGURAÇÃO DOS ARQUIVOS DE CONFIGURAÇÃO .CFG.....	40
4.4.3	METODOLOGIA DA ANÁLISE DE DADOS E DOS RESULTADOS .....	41
<b>5</b>	<b>Resultados e Análise .....</b>	<b>43</b>
5.1	RESULTADOS DO PRÉ-PROCESSAMENTO .....	43
5.1.1	CASOS HX .....	43
5.1.2	CASO HINF .....	44
5.2	RESULTADOS DAS SIMULAÇÕES NUMÉRICAS E DO PÓS-PROCESSAMENTO.....	45
5.2.1	CASOS HX .....	45
5.2.2	VALIDAÇÃO DOS RESULTADOS .....	49
5.2.3	CASO HINF .....	53
<b>6</b>	<b>Conclusão.....</b>	<b>56</b>
6.1	SUGESTÕES PARA TRABALHOS FUTUROS .....	57
	<b>Referências Bibliográficas.....</b>	<b>58</b>
	<b>Anexos .....</b>	<b>60</b>
<b>I</b>	<b>Dados de coordenadas dos pontos dos perfis de asa .....</b>	<b>61</b>



<b>II</b>	<b>Resultados Experimentais de Zerihan para CD e CL.....</b>	<b>64</b>
	<b>Apêndices .....</b>	<b>65</b>
<b>A</b>	<b>Visualizações das Malhas Geradas .....</b>	<b>66</b>
<b>B</b>	<b>Gráficos de Convergência das Simulações .....</b>	<b>79</b>
<b>C</b>	<b>Comparativo dos Resultados de Simulação com os Resultados Experimentais de Zerihan .....</b>	<b>88</b>
<b>D</b>	<b>Visualizações de Velocidade - Plano xy .....</b>	<b>92</b>
<b>E</b>	<b>Visualizações de Velocidade - Plano yz - Visão Geral .....</b>	<b>105</b>
<b>F</b>	<b>Visualizações de Velocidade - Plano yz - Detalhe dos Vórtices de Ponta de Asa.....</b>	<b>118</b>
<b>G</b>	<b>Visualizações 3D das linhas de corrente.....</b>	<b>131</b>
<b>H</b>	<b>Visualizações de Pressão - Plano xy .....</b>	<b>144</b>
<b>I</b>	<b>Visualizações de Pressão - Plano yz .....</b>	<b>157</b>
<b>J</b>	<b>Tabela de Cálculo e Tratamento de Dados dos Perfis de Asa de Zerihan.....</b>	<b>170</b>
<b>K</b>	<b>Códigos em Python Desenvolvidos para Análise dos Dados de Simulação .....</b>	<b>172</b>
<b>K.1</b>	<b>MONITORADOR E GERADOR DE GRÁFICOS DOS COEFICIENTES AERODINÂMICOS AO LONGO DAS ITERAÇÕES.....</b>	<b>172</b>
<b>K.2</b>	<b>GERADOR DE GRÁFICOS COMPARATIVOS COM OS DADOS DE ZERIHAN .....</b>	<b>182</b>
<b>L</b>	<b>Código dos Arquivos de Configuração do SU2 .....</b>	<b>197</b>
<b>L.1</b>	<b>CASO MODELO HX .....</b>	<b>197</b>
<b>L.2</b>	<b>CASO HINF .....</b>	<b>212</b>

# LISTA DE FIGURAS

1.1	Mercedes W196, pilotada por Juan Manuel Fangio em 1954 .....	2
1.2	McLaren M7A, pilotada por Bruce McLaren em 1969 .....	2
1.3	Metodologias experimentais de obtenção de dados aerodinâmicos .....	3
1.4	Red Bull RB9, pilotada por Sebastian Vettel em 2013.....	3
2.1	Forças devido à pressão e ao cisalhamento na superfície de um aerofólio. Fonte: (ANDERSON, 2017) .....	6
2.2	Forças e Momentos Aerodinâmicos. Fonte:(HOUGHTON et al., 2017)....	7
2.3	Representação das camadas limites de velocidade e de temperatura em uma parede. Fonte: (ANDERSON, 2017) .....	14
2.4	Visualização de camada limite de velocidade em experimento com bolhas de hidrogênio. Fonte: (SMITS; LIM, 2012) .....	15
2.5	Visualização de descolamento camada limite de velocidade em experi- mento de escoamento em torno de um corpo rombudo. Fonte: (SMITS; LIM, 2012) .....	15
2.6	Funções da asa dianteira de um carro monoposto. Fonte: (KATZ, 1995).	16
2.7	Asa dianteira do carro modelo de regulamentação técnica para a tem- porada de 2022 .....	17
2.8	Carros McLaren e Ferrari, temporada de 1998 .....	17
2.9	Linhas de corrente por um perfil de asa multielemento. Adaptado de: (TREMAYNE, 2009) .....	18
2.10	Visualização de escoamento turbulento. Fonte: (DYKE, 2008).....	19
3.1	Função do método numérico. Fonte: (MALISKA, 2023) .....	21
3.2	MVF por metodologia de resolução <i>Cell-Center</i> . Fonte: (MALISKA, 2023) .....	22
3.3	Papel das Condições de contorno na resolução por volumes finitos. Fonte: (MALISKA, 2023) .....	22
3.4	Comparativo de metodologias de modelagem de turbulência.....	24
3.5	Representação esquemática dos tamanhos mínimos de <i>eddys</i> nos modelos DNS e LES. Fonte: (FERZIGER; PERIĆ; STREET, 2020) .....	25
3.6	Gráfico exemplo das diferenças da variação de velocidade em um ponto no escoamento nos modelos DNS e LES. Fonte: (FERZIGER; PERIĆ; STREET, 2020) .....	25

3.7	Gráfico exemplo das componentes da velocidade em um ponto em um escoamento permanente no modelo RANS. Fonte: (FERZIGER; PERIĆ; STREET, 2020) .....	26
4.1	Tyrrell 026, temporada de 1998. ....	32
4.2	Experimento de Zerihan. Fonte: (ZERIHAN, 2001) .....	33
4.3	Figura 1 de Zerihan (2001) .....	33
4.4	Posicionamento das <i>endplates</i> de acordo com a Figura 1 de Zerihan (2001)	34
4.5	Geometria da Asa Frontal .....	34
4.6	Perfil do Túnel de Vento .....	35
4.7	Exemplo de definição da altura da asa, no caso $h = 50\text{ mm}$ .....	35
4.8	Geometria do volume de ar dentro do túnel de vento .....	36
4.9	Geometria do volume de ar dentro do túnel de vento com volume de refinamento local no SALOME .....	36
4.10	Geometria do volume de ar ao redor da asa frontal, no caso <i>hinf</i> .....	37
4.11	Árvore de arquivo de malha do módulo <i>Mesh</i> no SALOME .....	38
4.12	Estado de visualização do ParaView para velocidade, pressão e linhas de corrente .....	42
5.1	Comparação das malhas dos casos h50 e h110 .....	44
5.2	Malha de inflação para captura da camada limite .....	44
5.3	Malha do caso hinf .....	45
5.4	Comparação da convergência dos casos h35 e h130 para CD, CL e CSF ..	47
5.5	Visualização de velocidade do escoamento do caso h20 no plano xy e yz .	47
5.6	Visualização de velocidade do escoamento do caso h45 no plano xy e yz .	48
5.7	Visualização de velocidade do escoamento do caso h225 no plano xy e yz	48
5.8	Visualização da formação de vórtices de ponta de asa no caso h17 .....	48
5.9	Visualização de pressão do escoamento do caso h40 no plano xy e yz .....	49
5.10	Validação do coeficiente de arrasto .....	50
5.11	Validação do coeficiente de sustentação .....	52
5.12	Comparativo da razão entre os coeficientes adimensionais. ....	53
5.13	Convergência do caso hinf para CD, CL e CSF ao longo das iterações ....	54
5.14	Visualização de velocidade do escoamento do caso hinf no plano xy e yz .	55
5.15	Visualizações do caso hinf: detalhe da velocidade no plano yz e visualização 3D do escoamento com vórtices de ponta de asa .....	55
5.16	Visualizações de pressão do caso hinf: plano xy e plano yz .....	55
A.1	Malha do caso h17 .....	66
A.2	Malha do caso h20 .....	67
A.3	Malha do caso h22 .....	67
A.4	Malha do caso h25 .....	68
A.5	Malha do caso h27 .....	68

A.6	Malha do caso h30.....	69
A.7	Malha do caso h32.....	69
A.8	Malha do caso h35.....	70
A.9	Malha do caso h37.....	70
A.10	Malha do caso h40.....	71
A.11	Malha do caso h45.....	71
A.12	Malha do caso h50.....	72
A.13	Malha do caso h60.....	72
A.14	Malha do caso h70.....	73
A.15	Malha do caso h80.....	73
A.16	Malha do caso h85.....	74
A.17	Malha do caso h90.....	74
A.18	Malha do caso h95.....	75
A.19	Malha do caso h100.....	75
A.20	Malha do caso h110.....	76
A.21	Malha do caso h120.....	76
A.22	Malha do caso h130.....	77
A.23	Malha do caso h150.....	77
A.24	Malha do caso h225.....	78
B.1	Convergência do caso h17 para CD, CL e CSF ao longo das iterações.....	79
B.2	Convergência do caso h20 para CD, CL e CSF ao longo das iterações.....	79
B.3	Convergência do caso h22 para CD, CL e CSF ao longo das iterações.....	80
B.4	Convergência do caso h25 para CD, CL e CSF ao longo das iterações.....	80
B.5	Convergência do caso h27 para CD, CL e CSF ao longo das iterações.....	80
B.6	Convergência do caso h30 para CD, CL e CSF ao longo das iterações.....	81
B.7	Convergência do caso h32 para CD, CL e CSF ao longo das iterações.....	81
B.8	Convergência do caso h35 para CD, CL e CSF ao longo das iterações.....	81
B.9	Convergência do caso h37 para CD, CL e CSF ao longo das iterações.....	82
B.10	Convergência do caso h40 para CD, CL e CSF ao longo das iterações.....	82
B.11	Convergência do caso h45 para CD, CL e CSF ao longo das iterações.....	82
B.12	Convergência do caso h50 para CD, CL e CSF ao longo das iterações.....	83
B.13	Convergência do caso h60 para CD, CL e CSF ao longo das iterações.....	83
B.14	Convergência do caso h70 para CD, CL e CSF ao longo das iterações.....	83
B.15	Convergência do caso h80 para CD, CL e CSF ao longo das iterações.....	84
B.16	Convergência do caso h85 para CD, CL e CSF ao longo das iterações.....	84
B.17	Convergência do caso h90 para CD, CL e CSF ao longo das iterações.....	84
B.18	Convergência do caso h95 para CD, CL e CSF ao longo das iterações.....	85
B.19	Convergência do caso h100 para CD, CL e CSF ao longo das iterações ...	85
B.20	Convergência do caso h110 para CD, CL e CSF ao longo das iterações ...	85
B.21	Convergência do caso h120 para CD, CL e CSF ao longo das iterações ...	86

B.22	Convergência do caso h130 para CD, CL e CSF ao longo das iterações ...	86
B.23	Convergência do caso h150 para CD, CL e CSF ao longo das iterações ...	86
B.24	Convergência do caso h225 para CD, CL e CSF ao longo das iterações ...	87
C.1	Coeficiente de arrasto sobre altura adimensionalizada $h/c$ .....	90
C.2	Coeficiente de sustentação sobre altura adimensionalizada $h/c$ .....	91
D.1	Visualização de velocidade do escoamento no plano xy do caso h17.....	92
D.2	Visualização de velocidade do escoamento no plano xy do caso h20.....	93
D.3	Visualização de velocidade do escoamento no plano xy do caso h22.....	93
D.4	Visualização de velocidade do escoamento no plano xy do caso h25.....	94
D.5	Visualização de velocidade do escoamento no plano xy do caso h27.....	94
D.6	Visualização de velocidade do escoamento no plano xy do caso h30.....	95
D.7	Visualização de velocidade do escoamento no plano xy do caso h32.....	95
D.8	Visualização de velocidade do escoamento no plano xy do caso h35.....	96
D.9	Visualização de velocidade do escoamento no plano xy do caso h37.....	96
D.10	Visualização de velocidade do escoamento no plano xy do caso h40.....	97
D.11	Visualização de velocidade do escoamento no plano xy do caso h45.....	97
D.12	Visualização de velocidade do escoamento no plano xy do caso h50.....	98
D.13	Visualização de velocidade do escoamento no plano xy do caso h60.....	98
D.14	Visualização de velocidade do escoamento no plano xy do caso h70.....	99
D.15	Visualização de velocidade do escoamento no plano xy do caso h80.....	99
D.16	Visualização de velocidade do escoamento no plano xy do caso h85.....	100
D.17	Visualização de velocidade do escoamento no plano xy do caso h90.....	100
D.18	Visualização de velocidade do escoamento no plano xy do caso h95.....	101
D.19	Visualização de velocidade do escoamento no plano xy do caso h100 .....	101
D.20	Visualização de velocidade do escoamento no plano xy do caso h110 .....	102
D.21	Visualização de velocidade do escoamento no plano xy do caso h120 .....	102
D.22	Visualização de velocidade do escoamento no plano xy do caso h130 .....	103
D.23	Visualização de velocidade do escoamento no plano xy do caso h150 .....	103
D.24	Visualização de velocidade do escoamento no plano xy do caso h225 .....	104
E.1	Visualização de velocidade do escoamento no plano yz do caso h17.....	105
E.2	Visualização de velocidade do escoamento no plano yz do caso h20.....	106
E.3	Visualização de velocidade do escoamento no plano yz do caso h22.....	106
E.4	Visualização de velocidade do escoamento no plano yz do caso h25.....	107
E.5	Visualização de velocidade do escoamento no plano yz do caso h27.....	107
E.6	Visualização de velocidade do escoamento no plano yz do caso h30.....	108
E.7	Visualização de velocidade do escoamento no plano yz do caso h32.....	108
E.8	Visualização de velocidade do escoamento no plano yz do caso h35.....	109
E.9	Visualização de velocidade do escoamento no plano yz do caso h37.....	109
E.10	Visualização de velocidade do escoamento no plano yz do caso h40.....	110

E.11	Visualização de velocidade do escoamento no plano yz do caso h45 .....	110
E.12	Visualização de velocidade do escoamento no plano yz do caso h50 .....	111
E.13	Visualização de velocidade do escoamento no plano yz do caso h60 .....	111
E.14	Visualização de velocidade do escoamento no plano yz do caso h70 .....	112
E.15	Visualização de velocidade do escoamento no plano yz do caso h80 .....	112
E.16	Visualização de velocidade do escoamento no plano yz do caso h85 .....	113
E.17	Visualização de velocidade do escoamento no plano yz do caso h90 .....	113
E.18	Visualização de velocidade do escoamento no plano yz do caso h95 .....	114
E.19	Visualização de velocidade do escoamento no plano yz do caso h100 .....	114
E.20	Visualização de velocidade do escoamento no plano yz do caso h110 .....	115
E.21	Visualização de velocidade do escoamento no plano yz do caso h120 .....	115
E.22	Visualização de velocidade do escoamento no plano yz do caso h130 .....	116
E.23	Visualização de velocidade do escoamento no plano yz do caso h150 .....	116
E.24	Visualização de velocidade do escoamento no plano yz do caso h225 .....	117
F.1	Detalhe da visualização de velocidade do escoamento no plano yz mostrando os vórtices de ponta de asa do caso h17 .....	118
F.2	Detalhe da visualização de velocidade do escoamento no plano yz mostrando os vórtices de ponta de asa do caso h20 .....	119
F.3	Detalhe da visualização de velocidade do escoamento no plano yz mostrando os vórtices de ponta de asa do caso h22 .....	119
F.4	Detalhe da visualização de velocidade do escoamento no plano yz mostrando os vórtices de ponta de asa do caso h25 .....	120
F.5	Detalhe da visualização de velocidade do escoamento no plano yz mostrando os vórtices de ponta de asa do caso h27 .....	120
F.6	Detalhe da visualização de velocidade do escoamento no plano yz mostrando os vórtices de ponta de asa do caso h30 .....	121
F.7	Detalhe da visualização de velocidade do escoamento no plano yz mostrando os vórtices de ponta de asa do caso h32 .....	121
F.8	Detalhe da visualização de velocidade do escoamento no plano yz mostrando os vórtices de ponta de asa do caso h35 .....	122
F.9	Detalhe da visualização de velocidade do escoamento no plano yz mostrando os vórtices de ponta de asa do caso h37 .....	122
F.10	Detalhe da visualização de velocidade do escoamento no plano yz mostrando os vórtices de ponta de asa do caso h40 .....	123
F.11	Detalhe da visualização de velocidade do escoamento no plano yz mostrando os vórtices de ponta de asa do caso h45 .....	123
F.12	Detalhe da visualização de velocidade do escoamento no plano yz mostrando os vórtices de ponta de asa do caso h50 .....	124
F.13	Detalhe da visualização de velocidade do escoamento no plano yz mostrando os vórtices de ponta de asa do caso h60 .....	124

F.14	Detalhe da visualização de velocidade do escoamento no plano yz mostrando os vórtices de ponta de asa do caso h70 .....	125
F.15	Detalhe da visualização de velocidade do escoamento no plano yz mostrando os vórtices de ponta de asa do caso h80 .....	125
F.16	Detalhe da visualização de velocidade do escoamento no plano yz mostrando os vórtices de ponta de asa do caso h85 .....	126
F.17	Detalhe da visualização de velocidade do escoamento no plano yz mostrando os vórtices de ponta de asa do caso h90 .....	126
F.18	Detalhe da visualização de velocidade do escoamento no plano yz mostrando os vórtices de ponta de asa do caso h95 .....	127
F.19	Detalhe da visualização de velocidade do escoamento no plano yz mostrando os vórtices de ponta de asa do caso h100 .....	127
F.20	Detalhe da visualização de velocidade do escoamento no plano yz mostrando os vórtices de ponta de asa do caso h110 .....	128
F.21	Detalhe da visualização de velocidade do escoamento no plano yz mostrando os vórtices de ponta de asa do caso h120 .....	128
F.22	Detalhe da visualização de velocidade do escoamento no plano yz mostrando os vórtices de ponta de asa do caso h130 .....	129
F.23	Detalhe da visualização de velocidade do escoamento no plano yz mostrando os vórtices de ponta de asa do caso h150 .....	129
F.24	Detalhe da visualização de velocidade do escoamento no plano yz mostrando os vórtices de ponta de asa do caso h225 .....	130
G.1	Visualização 3D do escoamento mostrando os vórtices de ponta de asa do caso h17 .....	131
G.2	Visualização 3D do escoamento mostrando os vórtices de ponta de asa do caso h20 .....	132
G.3	Visualização 3D do escoamento mostrando os vórtices de ponta de asa do caso h22 .....	132
G.4	Visualização 3D do escoamento mostrando os vórtices de ponta de asa do caso h25 .....	133
G.5	Visualização 3D do escoamento mostrando os vórtices de ponta de asa do caso h27 .....	133
G.6	Visualização 3D do escoamento mostrando os vórtices de ponta de asa do caso h30 .....	134
G.7	Visualização 3D do escoamento mostrando os vórtices de ponta de asa do caso h32 .....	134
G.8	Visualização 3D do escoamento mostrando os vórtices de ponta de asa do caso h35 .....	135
G.9	Visualização 3D do escoamento mostrando os vórtices de ponta de asa do caso h37 .....	135

G.10	Visualização 3D do escoamento mostrando os vórtices de ponta de asa do caso h40.....	136
G.11	Visualização 3D do escoamento mostrando os vórtices de ponta de asa do caso h45.....	136
G.12	Visualização 3D do escoamento mostrando os vórtices de ponta de asa do caso h50.....	137
G.13	Visualização 3D do escoamento mostrando os vórtices de ponta de asa do caso h60.....	137
G.14	Visualização 3D do escoamento mostrando os vórtices de ponta de asa do caso h70.....	138
G.15	Visualização 3D do escoamento mostrando os vórtices de ponta de asa do caso h80.....	138
G.16	Visualização 3D do escoamento mostrando os vórtices de ponta de asa do caso h85.....	139
G.17	Visualização 3D do escoamento mostrando os vórtices de ponta de asa do caso h90.....	139
G.18	Visualização 3D do escoamento mostrando os vórtices de ponta de asa do caso h95.....	140
G.19	Visualização 3D do escoamento mostrando os vórtices de ponta de asa do caso h100.....	140
G.20	Visualização 3D do escoamento mostrando os vórtices de ponta de asa do caso h110.....	141
G.21	Visualização 3D do escoamento mostrando os vórtices de ponta de asa do caso h120.....	141
G.22	Visualização 3D do escoamento mostrando os vórtices de ponta de asa do caso h130.....	142
G.23	Visualização 3D do escoamento mostrando os vórtices de ponta de asa do caso h150.....	142
G.24	Visualização 3D do escoamento mostrando os vórtices de ponta de asa do caso h225.....	143
H.1	Visualização de pressão do escoamento no plano xy do caso h17.....	144
H.2	Visualização de pressão do escoamento no plano xy do caso h20.....	145
H.3	Visualização de pressão do escoamento no plano xy do caso h22.....	145
H.4	Visualização de pressão do escoamento no plano xy do caso h25.....	146
H.5	Visualização de pressão do escoamento no plano xy do caso h27.....	146
H.6	Visualização de pressão do escoamento no plano xy do caso h30.....	147
H.7	Visualização de pressão do escoamento no plano xy do caso h32.....	147
H.8	Visualização de pressão do escoamento no plano xy do caso h35.....	148
H.9	Visualização de pressão do escoamento no plano xy do caso h37.....	148
H.10	Visualização de pressão do escoamento no plano xy do caso h40.....	149



H.11	Visualização de pressão do escoamento no plano xy do caso h45.....	149
H.12	Visualização de pressão do escoamento no plano xy do caso h50.....	150
H.13	Visualização de pressão do escoamento no plano xy do caso h60.....	150
H.14	Visualização de pressão do escoamento no plano xy do caso h70.....	151
H.15	Visualização de pressão do escoamento no plano xy do caso h80.....	151
H.16	Visualização de pressão do escoamento no plano xy do caso h85.....	152
H.17	Visualização de pressão do escoamento no plano xy do caso h90.....	152
H.18	Visualização de pressão do escoamento no plano xy do caso h95.....	153
H.19	Visualização de pressão do escoamento no plano xy do caso h100 .....	153
H.20	Visualização de pressão do escoamento no plano xy do caso h110 .....	154
H.21	Visualização de pressão do escoamento no plano xy do caso h120 .....	154
H.22	Visualização de pressão do escoamento no plano xy do caso h130 .....	155
H.23	Visualização de pressão do escoamento no plano xy do caso h150 .....	155
H.24	Visualização de pressão do escoamento no plano xy do caso h225 .....	156
I.1	Visualização de pressão do escoamento no plano yz do caso h17 .....	157
I.2	Visualização de pressão do escoamento no plano yz do caso h20 .....	158
I.3	Visualização de pressão do escoamento no plano yz do caso h22 .....	158
I.4	Visualização de pressão do escoamento no plano yz do caso h25 .....	159
I.5	Visualização de pressão do escoamento no plano yz do caso h27 .....	159
I.6	Visualização de pressão do escoamento no plano yz do caso h30 .....	160
I.7	Visualização de pressão do escoamento no plano yz do caso h32 .....	160
I.8	Visualização de pressão do escoamento no plano yz do caso h35 .....	161
I.9	Visualização de pressão do escoamento no plano yz do caso h37 .....	161
I.10	Visualização de pressão do escoamento no plano yz do caso h40 .....	162
I.11	Visualização de pressão do escoamento no plano yz do caso h45 .....	162
I.12	Visualização de pressão do escoamento no plano yz do caso h50 .....	163
I.13	Visualização de pressão do escoamento no plano yz do caso h60 .....	163
I.14	Visualização de pressão do escoamento no plano yz do caso h70 .....	164
I.15	Visualização de pressão do escoamento no plano yz do caso h80 .....	164
I.16	Visualização de pressão do escoamento no plano yz do caso h85 .....	165
I.17	Visualização de pressão do escoamento no plano yz do caso h90 .....	165
I.18	Visualização de pressão do escoamento no plano yz do caso h95 .....	166
I.19	Visualização de pressão do escoamento no plano yz do caso h100 .....	166
I.20	Visualização de pressão do escoamento no plano yz do caso h110 .....	167
I.21	Visualização de pressão do escoamento no plano yz do caso h120 .....	167
I.22	Visualização de pressão do escoamento no plano yz do caso h130 .....	168
I.23	Visualização de pressão do escoamento no plano yz do caso h150 .....	168
I.24	Visualização de pressão do escoamento no plano yz do caso h225 .....	169
J.1	Planilha de cálculo, dados relativos à Main Wing. ....	170
J.2	Planilha de cálculo, dados relativos à Flap Wing.....	171

# LISTA DE TABELAS

4.1	Programas Utilizados no Projeto. ....	29
4.2	Descrição das geometrias para condições de contorno.....	37
4.3	Tamanhos especificados de elementos da malha de acordos com zona de refinamento local.....	39
5.1	Informações gerais sobre as malhas resultantes do pré-processamento. ....	43
5.2	Propriedades da malha do caso hinf .....	45
5.3	Desvio padrão da convergência ao longo das iterações de cada caso.....	46
5.4	Coefficientes Lineares das Retas de Tendência do Arrasto .....	50
5.5	Coefficientes Lineares das Retas de Tendência da Sustentação .....	51
5.6	Desvio padrão dos coeficientes aerodinâmicos adimensionais ao longo das iterações do caso hinf.....	54
5.7	Coefficientes aerodinâmicos adimensionais médios do caso hinf .....	54
I.1	Dados coletados para a posição adimensionalizada de pontos no perfil de asa da Flap Wing. Fonte: (ZERIHAN, 2001) .....	61
I.2	Dados coletados para a posição adimensionalizada de pontos no perfil de asa da Main Wing. Fonte: (ZERIHAN, 2001) .....	63
II.1	Resultados dos coeficientes adimensionais aerodinâmicos CD e CL obtidos experimentalmente. FONTE:(ZERIHAN, 2001), retirados utilizando o programa <i>xyscan</i> .....	64
C.1	Comparativo dos resultados das simulações com os experimentos de Zerihan (2001) .....	89

# LISTA DE CÓDIGOS FONTE

K.1	history_scraper.py .....	172
K.2	results_plotter.py.....	182
L.1	hx.cfg .....	197
L.2	hinf.cfg.....	212

# LISTA DE TERMOS E SIGLAS

$\alpha$	Ângulo de ataque
$\beta$	Propriedade Intensiva Qualquer do Fluido
$\Delta m_{CD}$	Erro Percentual de Similaridade entre Dados Numéricos e Experimentais do Arrasto
$\Delta m_{CL}$	Erro Percentual de Similaridade entre Dados Numéricos e Experimentais da Sustentação
$\delta$	Espessura de Camada Limite
$\mu$	Coefficiente de Viscosidade Dinâmica
$\nu$	Coefficiente de Viscosidade Cinemática
$\bar{\phi}$	Valor médio de variável em modelagem RANS
$\Phi$	Função de Dissipação
$\phi'$	Valor particular flutuante de variável em modelagem RANS
$\rho$	Massa Específica
$\rho_\infty$	Massa Específica do escoamento Livre
$\tau$	Tensão Cisalhante
$\vec{g}$	Vetor Gravidade
$\vec{n}$	Vetor Normal à Superfície do Volume de Controle
$\vec{q}$	Vetor Difusivo de Energia Interna
$\vec{u}$	Velocidade Vetorial
$B$	Uma Propriedade Qualquer do Fluido
$C_D$	Coefficiente Adimensional de Arrasto
$C_L$	Coefficiente Adimensional de Sustentação

$C_M$	Coeficiente Adimensional de Momento
$C_P$	Coeficiente Adimensional de Pressão
$D$	Drag
$e$	Energia Interna Intensiva
$E_t$	Energia Total por Unidade de Volume
$F_{aero}$	Força Aerodinâmica
$gs_{CD}$	Grau de Similaridade Percentual entre Dados Numéricos e Experimentais do Arrasto
$gs_{CL}$	Grau de Similaridade Percentual entre Dados Numéricos e Experimentais da Sustentação
$h$	Altura da Asa com Relação ao Chão (Ground Height)
$k_t$	Coeficiente de Difusão Térmica
$L$	Lift
$L_R$	Roll
$L_{car}$	Escala Característica de Comprimento
$M$	Pitch
$m$	Massa
$m_{CD;exp}$	Coeficiente Linear de Reta de Tendência do Arrasto Experimental
$m_{CD;sim}$	Coeficiente Linear de Reta de Tendência do Arrasto Numérico
$m_{CL;exp}$	Coeficiente Linear de Reta de Tendência da Sustentação Experimental
$m_{CL;sim}$	Coeficiente Linear de Reta de Tendência da Sustentação Numérica
$N$	Yaw
$p$	Pressão
$p_\infty$	Pressão Característica do escoamento
$P_{ij}$	Tensor de Tensões
$Q$	Calor Gerado Por Agentes Externos por Unidade de Volume
$q$	Calor intensivo Gerado Por Agentes Externos por Unidade de Volume

$q_\infty$	Pressão Dinâmica
$Re$	Número do Reynolds
$S$	Fronteira do Volume de Controle
$S_{car}$	Superfície Característica do Corpo
$T$	Temperatura
$t$	Tempo
$u_\infty$	Escala Característica de Velocidade
$V$	Volume
$Y$	Cross-Wind
$c$	Corda
CFD	Computational Fluid Dynamics
CSF	Coeficiente de Força Lateral
DNS	<i>Direct Numerical Simulation</i> ou Modelagem Direta de Turbulência
ED	Equação Diferencial
FIA	<i>Fédération Internationale de l'Automobile</i>
LES	<i>Large Eddy Simulations</i> ou Equações de Navier-Stokes Filtradas
MVF	Método dos Volumes Finitos
RANS	<i>Reynolds-Averaged Navier-Stokes Simulations</i> ou Equações Médias de Reynolds
SST	Shear Stress Transport
SU2	<i>Stanford University Unstructured</i>
TTR	Teorema do Transporte de Reynolds

# Capítulo 1

## Introdução

### 1.1 Motivação

A dinâmica dos fluidos computacional é uma área crescente e cada vez mais importante dentro da engenharia e da física, pois permite a simulação de sistemas fluidos complexos em uma variedade de aplicações, desde o design de aeronaves e carros de competição até a previsão de vazamentos em tubulações industriais. No entanto, a realização de simulações de dinâmica dos fluidos computacional é um processo complexo que envolve muitas etapas, desde a escolha da malha até a validação dos resultados.

Um dos desafios da dinâmica dos fluidos computacional é estabelecer e validar um ciclo de simulações confiável e eficiente. Isso envolve a escolha de ferramentas adequadas, a definição de parâmetros de simulação apropriados e a validação dos resultados da simulação com dados experimentais ou teóricos. A utilização de ferramentas de código aberto pode ser uma vantagem neste processo, pois elas são geralmente gratuitas, flexíveis e podem ser ajustadas para atender às necessidades específicas de uma aplicação.

Este trabalho visa consolidar as práticas recomendadas para a realização de simulações de dinâmica dos fluidos computacional confiáveis e eficientes. Além disso, a pesquisa também pretende contribuir para a comunidade de ferramentas de código aberto, fornecendo *insights* valiosos sobre como melhorar e aperfeiçoar essas ferramentas para aplicações de fluidodinâmica computacional.

Isso será feito por meio da simulação CFD de uma asa frontal de um carro de fórmula 1. Ambientes competitivos são, muitas vezes, berços de inovações e na fórmula 1, isso não é diferente. Na começo, o esporte era dominado pelas construtoras que conseguissem desenvolver o melhor carro do ponto de vista dinâmico, ou seja, com boa estabilidade nas curvas e que possuísse o melhor sistema de potência.

Conforme o esporte foi evoluindo, os carros foram ficando mais rápidos até que se chegou em um ponto em que considerações sobre aspectos aerodinâmicos se mostraram

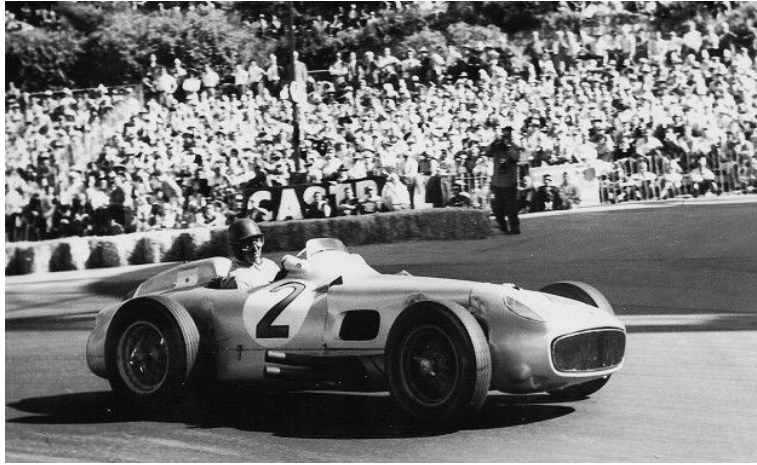


Figura 1.1: Mercedes W196, pilotada por Juan Manuel Fangio em 1954

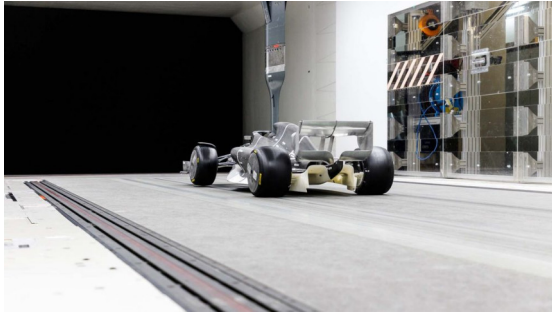
obrigatórias para equipes. Eventualmente, percebeu-se que, para um carro fazer uma volta de maneira mais rápida no circuito, o ponto crítico é seu desempenho nas curvas. Como o desempenho de um carro em movimento de curva é diretamente proporcional à velocidade e à aderência dos pneus ao asfalto, procurou-se formas de aumentar a aderência do carro. Dentre os métodos testados, o que mais se destacou foi a adição de elementos aerodinâmicos ao veículo. Estes elementos teriam o propósito de gerar sustentação negativa e portanto "prender" o carro à pista.



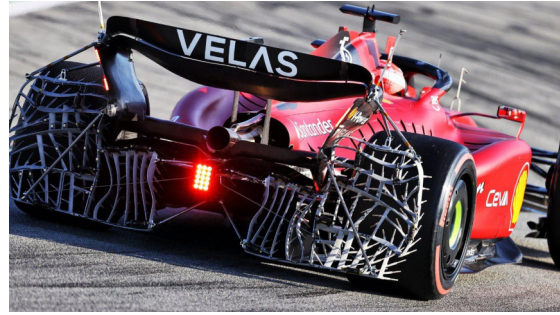
Figura 1.2: McLaren M7A, pilotada por Bruce McLaren em 1969

Atualmente, a aerodinâmica é considerada uma das áreas de estudo mais importantes para as equipes de competição, sendo investida grande parte do orçamento para investigações experimentais e numéricas, utilizando ciclo de simulações CFD, túneis de vento próprios e até experimentos em sessões de treino para correlação dos dados de pesquisa.





(a) Modelo de teste de F1 em túnel de vento



(b) Dispositivo experimental de correlação de dados de túnel de vento em carro Ferrari durante sessão de treino

Figura 1.3: Metodologias experimentais de obtenção de dados aerodinâmicos

O resultado disso é a carenagem e elementos aerodinâmicos de geometria complexa utilizados pelas equipes do *grid*. Por isso, este trabalho utilizará um modelo de asa frontal de fórmula 1 como geometria a ser estudada.



Figura 1.4: Red Bull RB9, pilotada por Sebastian Vettel em 2013

## 1.2 Estrutura do Trabalho

Este trabalho está estruturado da seguinte forma:

- Os capítulos 2 e 3 trazem uma revisão bibliográfica e de conceitos teóricos relacionados à mecânica dos fluidos e à dinâmica dos fluidos computacional, citando trabalhos importantes e revisando conceitos fundamentais para a compreensão deste projeto.
- O capítulo 4 diz respeito à execução do projeto, denotando metodologia e procedimentos realizados.
- Em Resultados e Análise, capítulo 5, são apresentados os resultados obtidos durante a realização deste projeto. Assim como também serão feitas análises das observações realizadas.

- Por fim, na Conclusão, resumem-se os resultados do capítulo 5 assim como também são propostas linhas de pesquisa para trabalhos futuros.

## 1.3 Objetivos

Este trabalho é um projeto de graduação submetido à universidade de Brasília como parte dos requisitos para obtenção do grau de engenheiro mecânico.

O principal objetivo deste projeto é a investigação da viabilidade de utilização de programas de código aberto para o estabelecimento de um ciclo de simulações em dinâmica dos fluidos computacional que apresente resultados de boa qualidade e coerentes com a realidade. Para isto, os objetivos são:

1. Criação de uma geometria que represente os experimentos de Zerihan (2001)
2. Realização de testes de capacidade e limitações de geração de malha dos programas Gmsh e SALOME.
3. Geração de malhas para as geometrias dos experimentos da tese de doutorado de Zerihan (2001) com os programas de código aberto propostos.
4. Preparo dos arquivos necessários para simulações em dinâmica dos fluidos computacional no programa SU2.
5. Validação dos resultados numéricos do SU2 por comparação dos coeficientes aerodinâmicos adimensionais com os resultados experimentais de Zerihan (2001).

Além disso, também há o objetivo secundário do projeto de, caso os resultados numéricos do ciclo sejam validados, realizar uma simulação com a asa frontal de Zerihan (2001) em uma situação não testada experimentalmente para investigar o comportamento do escoamento e entender melhor os efeitos de instalação presentes nos experimentos.

# Capítulo 2

## Arcabouço Teórico: Mecânica dos Fluidos e Aerodinâmica

Neste capítulo estão dispostos os conceitos fundamentais da mecânica dos fluidos que servem de base para este trabalho e permitem a compreensão das decisões tomadas na realização do projeto e dos resultados e análises subsequentes.

### 2.1 Forças e Momentos Aerodinâmicos

Anderson (2017) separa as forças e momentos aerodinâmicos atuantes em um corpo submerso em apenas duas:

- Forças devido à distribuição de pressão na superfície do corpo;
- Forças devido aos esforços cisalhantes na superfície do corpo.

Por um lado, este método de classificação torna o estudo de aerodinâmica em uma disciplina com um ponto de foco singular, de entender essas duas forças possíveis. Pelo outro lado, como ambas forças atuam no corpo por unidade de área de superfície e com direções e sentidos altamente dependentes da geometria específica do corpo, essa tarefa que, a princípio parecia simples, se torna muito mais complexa.

A força devido à pressão  $p$  age sempre na direção normal à superfície, e a força devido ao cisalhamento  $\tau$  age sempre tangencialmente à superfície. Desta forma, as duas podem ser tratadas como componentes de uma força resultante  $F_{aero}$  que age no centro de pressão do corpo.

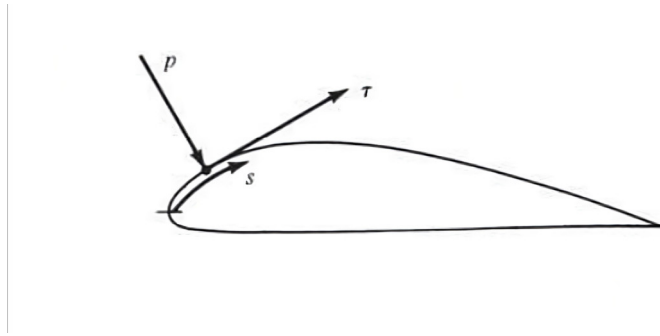


Figura 2.1: Forças devido à pressão e ao cisalhamento na superfície de um aerofólio. Fonte: (ANDERSON, 2017)

Para o estudo de corpos e geometrias complexas em aerodinâmica, é importante denotar que essa força resultante pode ser decomposta em 3 forças e 3 momentos fundamentais (HOUGHTON et al., 2017):

- Sustentação (Lift,  $L$ ): A força de sustentação é resultado da diferença de pressão entre o intradorso e o extradorso do perfil de asa. Essa força é perpendicular ao escoamento e em aeronaves, é responsável em vencer a força peso e permitir aeronaves alçarem voo.
- Arrasto (Drag,  $D$ ): É a componente de força agindo em sentido paralelo ao escoamento, derivada majoritariamente da tensão cisalhante. Essa força resiste o movimento do corpo no fluido.
- Força Lateral (Side ou Cross-wind Force,  $Y$ ): É a componente de força simultaneamente perpendicular às forças de arrasto e sustentação.
- Arfagem (Pitch,  $M$ ): Momento que inclui as forças de sustentação e arrasto, é responsável por modificar o ângulo de ataque  $\alpha$  do corpo no escoamento.
- Guinada (Yaw,  $N$ ): É o momento que tende a rotacionar o corpo em torno da força de sustentação.
- Rolagem (Roll,  $L_R$ ): É o momento que tende a rolar o corpo em torno da força de arrasto.

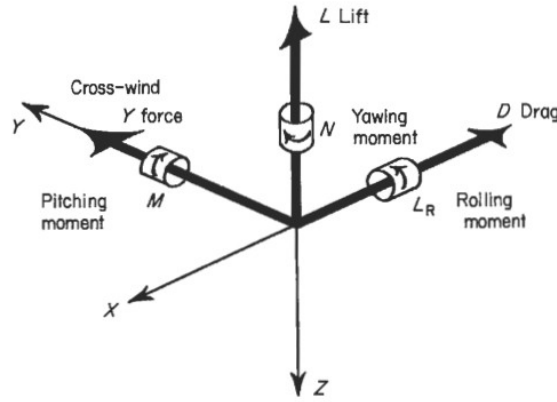


Figura 2.2: Forças e Momentos Aerodinâmicos. Fonte:(HOUGHTON et al., 2017)

Para a determinação de cada uma dessas forças e momentos, porém, deve-se ter acesso à geometria em questão, o ângulo de ataque ( $\alpha$ ) do corpo em relação ao escoamento, os número de Reynolds e Mach do escoamento (que influenciam principalmente na condição de natureza do escoamento, laminar/turbulento ou compressível/incompressível) e nas características do escoamento como velocidade ( $u_\infty$ ) e pressão ( $p_\infty$ ).

## 2.2 Equações Governantes

Antes de explicar as equações governantes em mecânica dos fluidos que se aplicam a este trabalho, é importante denotar que, como a grande maioria dos problemas em mecânica dos fluidos se dá em escalas de dimensão maiores que escalas moleculares, pode se considerar que esses problemas se encaixam em problemas de meios contínuos, ou seja, pode-se analisar esses problemas por meio do cálculo diferencial. Também deve-se observar que as velocidades do escoamento que serão estudadas neste trabalho são de no máximo  $v = 30 \text{ m/s}$  e, portanto, não será necessário considerar a compressibilidade do ar. Por fim, uma das características mais importantes do escoamento escolhido para este projeto é sua natureza turbulenta, sendo, portanto, necessário levar isso em conta.

Além disso, também é necessário lembrar que, em mecânica dos fluidos, tem-se, mais frequentemente, uma descrição euleriana dos escoamentos, isto é, por volumes de controle (ÇENGEL; CIMBALA, 2014). Deste modo, para a correta apresentação das equações governantes do problema, é necessária sua conversão para uma representação integral (WHITE; CHUL, 2018). Isto se dá pela utilização do teorema do transporte de Reynolds.

O teorema do transporte de Reynolds é dado a partir da ideia que, sendo  $B$  uma propriedade qualquer do fluido, e sua complementar propriedade intensiva dada por:

$$\beta = \frac{dB}{dm} \quad (2.1)$$

Pode-se enunciar:

$$B_V = \int_V \beta \, dm = \int_V \beta \, \rho \, dV \quad (2.2)$$

Na descrição euleriana de sistemas, o volume de controle é fixo no espaço e, portanto, recebe um fluxo de massa por sua fronteira  $S$ . Logo, cada elemento infinitesimal de área da fronteira percebe uma velocidade do escoamento  $\vec{u}$ . Pode-se dizer que a variação da propriedade  $B$  em um período de tempo infinitesimal  $dt$  pode ser expressa pela variação advinda de fluxo de massa pela fronteira somada com a variação interna do volume de controle. Para expressar isso, ou seja, a derivada de (2.1) em termos matemáticos, deve-se definir um vetor normal à superfície da fronteira do volume de controle  $\vec{n}$ . Assim, o Teorema do Transporte de Reynolds (TTR) é dado por:

$$\frac{dB}{dt} = \frac{d}{dt} \left( \int_V \beta \, \rho \, dV \right) + \oint_S \beta \, \rho \, (\vec{u} \cdot \vec{n}) \, dS \quad (2.3)$$

Para completar o arsenal teórico necessário para a representação integral das equações governantes da mecânica dos fluidos, também será necessária a utilização de dois teoremas do cálculo diferencial e integral exposto por Stewart (2016). Estes são:

1. O Teorema da Divergência

$$\oint_S \vec{F} \cdot \vec{n} \cdot dS = \int_V \vec{\nabla} \cdot \vec{F} \quad (2.4)$$

2. O Teorema da Localização

$$\int_D \vec{F}(x) \, dx = 0 \, \forall D \subset \Omega \Rightarrow \vec{F}(x) = 0 \, \forall x \in \Omega \quad (2.5)$$

### 2.2.1 Equação da Continuidade

A equação da continuidade é a relação que rege a propriedade de conservação de massa e, portanto, é responsável pelo balanço de massa no sistema descrito. Esta pode ser obtida por meio da aplicação do TTR (2.3) na propriedade massa  $m$ :

$$\frac{dB}{dt} = \frac{dm}{dt} = 0 \quad (2.6)$$

$$\beta = \frac{dm}{dm} = 1 \quad (2.7)$$

$$\int_V \frac{\partial \rho}{\partial t} \, dV + \oint_S \rho \vec{u} \cdot \vec{n} \, dS = 0 \quad (2.8)$$

Assim, pode-se aplicar o teorema da divergência (2.2) para transformar a integral de superfície em uma integral de volume:

$$\int_V \left[ \frac{\partial \rho}{\partial t} + \vec{\nabla} \cdot (\rho \vec{u}) \right] dV = 0 \quad (2.9)$$

Por meio do teorema da localização, pode-se simplificar para:

$$\frac{\partial \rho}{\partial t} + \vec{\nabla} \cdot (\rho \vec{u}) = 0 \quad (2.10)$$

Por fim, chega-se à equação da continuidade ao aplicar a condição incompressível do caso, ou seja, de variação de massa específica nula.

$$\vec{\nabla} \cdot (\rho \vec{u}) = 0 \quad (2.11)$$

Ou

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = 0 \quad (2.12)$$

### 2.2.2 Equação da Quantidade de Movimento

A segunda lei de Newton, que versa sobre a conservação da quantidade de movimento, pode ser descrita para uma partícula observada a partir de um referencial não inercial, passando por um volume de controle fixo infinitesimal como:

$$\frac{\partial}{\partial t} (\rho \vec{u}) + \vec{\nabla} \cdot \rho \vec{u} \vec{u} = \rho \vec{g} + \vec{\nabla} \cdot P_{ij} \quad (2.13)$$

Em que, segundo Anderson, Pletcher e Tannehill (2013), Neto (2020) e Çengel e Cimbala (2014):

- $\frac{\partial}{\partial t} (\rho \vec{u})$  : Taxa de variação de quantidade de momento linear contida em uma partícula de fluido.
- $\vec{\nabla} \cdot \rho \vec{u} \vec{u}$  : Fluxo líquido advectivo de quantidade de movimento pela superfície de uma partícula.
- $\rho \vec{g}$  : Força peso da partícula.
- $\vec{\nabla} \cdot P_{ij}$  : Fluxo líquido difusivo de quantidade de movimento linear devido às interações intermoleculares.

A partir da expansão do termo de fluxo advectivo de quantidade de momento linear pela superfície da partícula e aplicando a equação da continuidade (2.10), pode-se

reduzir (2.13) a:

$$\rho \frac{D\vec{u}}{Dt} = \rho \vec{g} + \vec{\nabla} \cdot P_{ij} \quad (2.14)$$

O termo  $P_{ij}$  representa o tensor de tensões, ou seja, as tensões atuantes sobre a partícula de fluido. Este tensor pode ser aberto e reescrito da seguinte forma (MUNSON; YOUNG; OKIISHI, 2006):

$$P_{ij} = -p\delta_{ij} + \mu \left[ \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) - \frac{2\delta_{ij}}{3} \frac{\partial u_k}{\partial x_k} \right]; \quad i, j, k = 1, 2, 3 \quad (2.15)$$

Onde:

- $-p\delta_{ij}$ : Efeitos de pressão do tensor de tensões.
- $\mu \left[ \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) - \frac{2}{3}\delta_{ij} \frac{\partial u_k}{\partial x_k} \right]$ : Efeitos de viscosidade do tensor de tensões.

Assim, pode-se substituir (2.15) em (2.14) para obter:

$$\rho \frac{D\vec{u}}{Dt} = \rho \vec{g} - \vec{\nabla} p + \frac{\partial}{\partial x_j} \left[ \mu \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) - \frac{2}{3}\delta_{ij}\mu \frac{\partial u_k}{\partial x_k} \right] \quad (2.16)$$

A equação (2.16) é conhecida como **equação de Navier-Stokes** por ter sido desenvolvida pelos físicos *Claude-Louis Navier* e *George Gabriel Stokes*. Esta equação é de fundamental importância para a fluidodinâmica e, principalmente, para o estudo de escoamentos viscosos. Esta equação, por ter uma natureza não-linear, não pode ser resolvida analiticamente. Por isso, em problemas com escoamentos de interesse para a engenharia, quando se percebe que os efeitos viscosos são significativos, a resolução é feita de maneira numérica ou experimental. Porém, esses métodos devem sempre tentar se aproximar o máximo possível da equação de Navier-Stokes.

Neste projeto, por se tratar de um escoamento incompressível e newtoniano, foi assumido o coeficiente de viscosidade  $\mu$  constante. Assim, a equação de Navier-Stokes (2.16) toma forma específica dada por:

$$\rho \frac{D\vec{u}}{Dt} = \rho \vec{g} - \vec{\nabla} p + \mu \nabla^2 \vec{u} \quad (2.17)$$

### 2.2.3 Equação da Energia

A equação da energia é derivada da primeira lei da termodinâmica, que dita a conservação da energia (ÇENGEL; CIMBALA, 2014). Quando aplicada a uma partícula



de fluido, passando por um volume de controle infinitesimal, a primeira lei pode ser escrita como:

$$\frac{\partial E_t}{\partial t} + \vec{\nabla} \cdot E_t \vec{u} = \frac{\partial Q}{\partial t} - \vec{\nabla} \cdot \vec{q} + \rho \vec{g} \cdot \vec{u} + \vec{\nabla} \cdot (P_{ij} \cdot \vec{u}) \quad (2.18)$$

De modo que  $E_t$  é a energia total por unidade de volume, que também pode ser escrita como um somatório da energia interna intensiva  $e$  multiplicada pela massa específica  $\rho$  somadas aos outros tipos de energia como potencial, cinética, e  $\vec{q}$  é o vetor difusivo de energia interna.

Em (2.18) cada termo é responsável pela modelagem de algum tipo de movimento de energia possível. Estes modos de movimentos são explicados por Anderson, Pletcher e Tannehill (2013), Neto (2020) e Çengel e Cimbala (2014) como:

- $\frac{\partial E_t}{\partial t}$  : Taxa de variação total da energia contida na partícula ao longo do tempo.
- $\vec{\nabla} \cdot E_t \vec{u}$  : Fluxo líquido advectivo de energia total através da superfície da partícula;
- $\frac{\partial Q}{\partial t}$  : Taxa de calor por unidade de volume gerada por agentes externos;
- $\vec{\nabla} \cdot \vec{q}$  : Fluxo líquido difusivo de energia térmica;
- $\rho \vec{g} \cdot \vec{u}$  : Termo de modelagem de conversão de energia potencial gravitacional em energia cinética;
- $\vec{\nabla} \cdot (P_{ij} \cdot \vec{u})$  : Fluxo líquido de energia térmica advinda do trabalho de deformação do volume da partícula por meio do Tensor de Tensões.

Pode-se utilizar a equação da continuidade (2.11) para reescrever os termos do lado esquerdo da equação da primeira lei da termodinâmica (2.18):

$$\rho \frac{D}{Dt} \left( \frac{E_t}{\rho} \right) = \frac{\partial E_t}{\partial t} + \vec{\nabla} \cdot E_t \vec{u} = \rho \frac{De}{Dt} + \rho \frac{D}{Dt} \left( \frac{u^2}{2} \right) \quad (2.19)$$

Alternativamente, pode-se observar que, fazendo o produto escalar da equação da quantidade de movimento reduzida encontrada em (2.14) com o vetor velocidade  $\vec{u}$ , tem-se:

$$\rho \frac{D\vec{u}}{Dt} \cdot \vec{u} = \rho \vec{g} \cdot \vec{u} - \vec{\nabla} p \cdot \vec{u} + (\vec{\nabla} \cdot \tau_{ij}) \cdot \vec{u} \quad (2.20)$$

Combinando (2.19) com (2.20), pode-se substituir o resultado na equação da primeira lei da termodinâmica (2.18).

$$\rho \frac{De}{Dt} + p (\vec{\nabla} \cdot \vec{u}) = \frac{\partial Q}{\partial t} - \vec{\nabla} \cdot \vec{q} + \vec{\nabla} \cdot (\tau_{ij} \cdot \vec{u}) - (\vec{\nabla} \cdot \tau_{ij}) \cdot \vec{u} \quad (2.21)$$

Pode-se combinar em um termo denominado função de dissipação ( $\Phi$ ) os termos  $(\vec{\nabla} \cdot (\tau_{ij} \cdot \vec{u}))$  e  $(-\vec{\nabla} \cdot \tau_{ij}) \cdot \vec{u}$ . Pode-se também definir entalpia ( $h$ ) como:

$$h = e + \frac{p}{\rho} \quad (2.22)$$

E utilizar a equação da continuidade (2.10) para reescrever (2.21) como:

$$\rho \frac{Dh}{Dt} = \frac{Dp}{Dt} + \frac{\partial Q}{\partial t} - \vec{\nabla} \cdot \vec{q} + \Phi \quad (2.23)$$

Finalmente, como o escoamento é compressível, pode-se considerar um coeficiente de difusão térmica  $k_t$  constante, reduzindo a equação para uma função da temperatura  $T$ :

$$\rho \frac{De}{Dt} = \frac{\partial Q}{\partial t} + k_t \nabla^2 T + \Phi \quad (2.24)$$

## 2.3 Grupos Adimensionais de Interesse

Em mecânica dos fluidos, é desejada a caracterização de diversos tipos de escoamentos, porém, a resolução analítica das equações governantes (2.10), (2.16) e (2.24) nem sempre é possível ou viável. Por isso, muitas vezes procuram-se respostas sobre o comportamento de um escoamento por meio de testes experimentais em modelo (ÇEN-GEL; CIMBALA, 2014). Também pode-se modelar o escoamento em uma simulação computacional de fluidodinâmica, como feito neste trabalho. O problema destes métodos é a dificuldade de garantir que os modelos são fidedignos à realidade. Para isso, deve-se procurar uma similaridade do modelo com a situação real, logo, foram desenvolvidos parâmetros adimensionais que podem ser utilizados como comparativos entre os escoamentos de uma dada categoria.

### 2.3.1 O Número de Reynolds

O número de Reynolds ( $R_e$ ) é um parâmetro adimensional nascido da relação da natureza (laminar, transitória ou turbulenta) do escoamento com a razão das forças inerciais pelas forças viscosas do escoamento. Ou seja, ele é um fator determinante na natureza do escoamento e denota que esta é dependente da proporção das forças no escoamento. Se as forças viscosas forem grandes o suficiente, podem se sobrepor às forças inerciais, induzindo o escoamento a se comportar de maneira mais parecida com um escoamento laminar. Do mesmo modo, em escoamentos com forças inerciais grandes o suficiente, pode-se gerar elementos turbilhonários e caracterizar um escoamento

turbulento.

O Número de Reynolds é dado por:

$$Re = \frac{\rho u_{\infty} L_{car}}{\mu} = \frac{u_{\infty} L_{car}}{\nu} \quad (2.25)$$

Sendo que  $L_{car}$  é o comprimento característico que deve ser adotado conforme o problema em questão. No caso, por se tratar de uma asa, será adotado o valor da corda ( $c$ ). Além disso,  $u_{\infty}$  é o valor da velocidade característica do escoamento, no caso, pode-se adotar a velocidade do escoamento longe o suficiente da asa.

### 2.3.2 Os Coeficientes Aerodinâmicos

Outros grupos adimensionais importantes para este projeto são advindos da aerodinâmica. Estes são derivados da definição de pressão dinâmica ( $q_{\infty}$ ), dada como:

$$q_{\infty} = \frac{\rho_{\infty} u_{\infty}^2}{2} \quad (2.26)$$

Onde  $q_{\infty}$  é a massa específica no escoamento livre e  $u_{\infty}$  a velocidade do escoamento livre (ANDERSON, 2017). Assim, pode-se definir a superfície característica do corpo no escoamento como  $S_{car}$  e enumerar os coeficientes aerodinâmicos de interesse para este projeto:

- Coeficiente de Pressão:

$$C_P = \frac{p - p_{\infty}}{q_{\infty}} \quad (2.27)$$

- Coeficiente de Sustentação:

$$C_L = \frac{L}{q_{\infty} S_{car}} \quad (2.28)$$

- Coeficiente de Momento:

$$C_M = \frac{M}{q_{\infty} S_{car} L_{car}} \quad (2.29)$$

- Coeficiente de Arrasto:

$$C_D = \frac{D}{q_{\infty} S_{car}} \quad (2.30)$$

Nesta pesquisa, tomou-se  $S_{car}$  como a projeção da área da asa vista de uma visão superior da asa em questão.

## 2.4 Camada Limite de Velocidade

A camada limite de velocidade é fruto da interação entre dois fenômenos da mecânica dos fluidos: a hipótese de não-escorregamento (*no-slip condition*) e a natureza viscosa dos fluidos (ANDERSON, 2017):

- A condição de não escorregamento foi proposta por Prandtl em 1904 e é definida como a hipótese de velocidade de escoamento do fluido nula quando relativa à parede e para a partícula na sua superfície.
- A viscosidade dos fluidos gera um gradiente de velocidade de dimensão finita nas imediações da superfície imersa pois a velocidade do escoamento deve ir de forma contínua de  $u_\infty \rightarrow 0$  na parede.

A interação entre esses dois fenômenos gera a camada limite de velocidade como na figura 2.3.

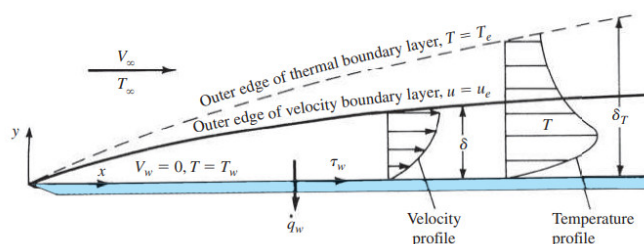


Figura 2.3: Representação das camadas limites de velocidade e de temperatura em uma parede. Fonte: (ANDERSON, 2017)

### 2.4.1 Descolamento de Camada Limite

A camada limite está presente em todos os objetos imersos em um escoamento. Dependendo da geometria em questão e sua disposição espacial em relação ao escoamento, pode ocorrer o fenômeno de descolamento da camada limite. Este fenômeno acontece quando as forças inerciais do escoamento são grandes o suficiente para forçar a camada limite a se desprender da superfície do corpo, gerando, portanto, elementos turbilhonares em sua esteira. Isso faz com que todo o escoamento se descole do corpo e, portanto, tem efeitos bastante significativos nas forças e momentos resultantes.

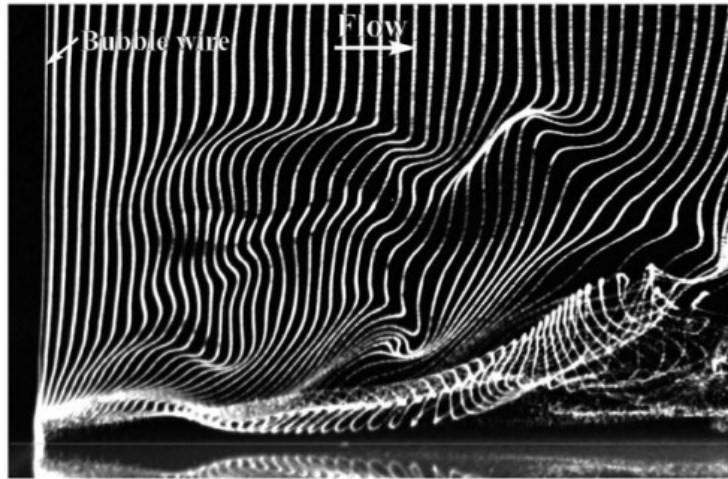


Figura 2.4: Visualização de camada limite de velocidade em experimento com bolhas de hidrogênio. Fonte: (SMITS; LIM, 2012)

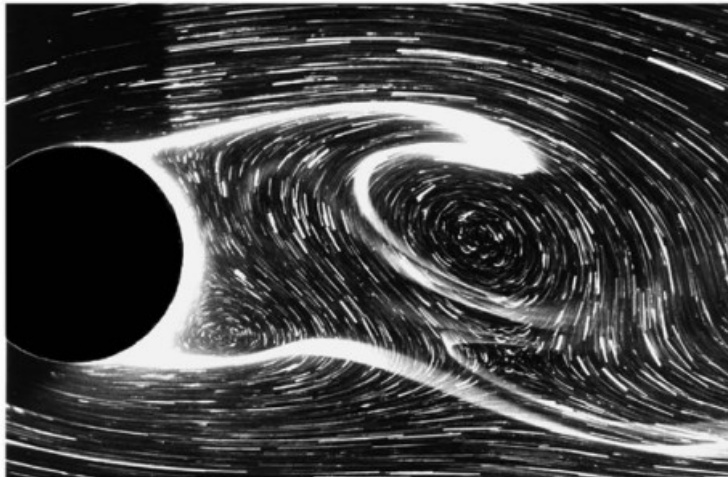


Figura 2.5: Visualização de descolamento camada limite de velocidade em experimento de escoamento em torno de um corpo rombudo. Fonte: (SMITS; LIM, 2012)

### 2.4.2 Espessura de Camada Limite

A camada limite é uma região de transição de velocidade e, portanto, não tem uma espessura bem definida. O que se faz frequentemente é atribuir uma espessura tal que a velocidade do fluido naquele ponto seja igual a 99% da velocidade do escoamento livre (WHITE; CHUL, 2018), ou seja:

$$\frac{u}{u_{\infty}} = 0,99 \quad (2.31)$$

Para a tradução disso em espessura dimensional da camada limite, utiliza-se a tabela do perfil de velocidade de Blasius para completar os dados necessários para a

equação de Blasius, resultando em:

$$\frac{\delta_{99\%}}{x} \approx \frac{5,0}{Re_x^{\frac{1}{2}}} \quad (2.32)$$

## 2.5 Aerodinâmica de Carros de Corrida

O estudo de aerodinâmica, apesar de ser geralmente focado para a indústria aeronáutica, se tornou igualmente importante na indústria automotiva, inicialmente com a intenção de criar carros de competição mais competitivos, que minimizassem o arrasto e pudessem alcançar velocidades maiores sem perder a estabilidade (KATZ, 1995). Posteriormente, este conhecimento encontrou aplicação também na indústria de carros de produção em massa pela possibilidade de aumentar a eficiência energética dos carros populares melhorando a autonomia desses carros e reduzindo as emissões de efeito estufa por quilometro rodado, contribuindo para o alcance de metas de desenvolvimento sustentável.

### 2.5.1 Elementos Aerodinâmicos de um Carro de Corrida

Em carros de corrida, especialmente, há o uso de elementos aerodinâmicos como asas, flaps, difusores e carenagem para um controle mais preciso do escoamento. Este controle é extremamente importante pois as forças aerodinâmicas de arrasto e sustentação impactam o comportamento dinâmico do carro de maneira significativa (HOUGHTON et al., 2017).

Ao se utilizar asas, os projetistas visam a geração de *downforce* com a mínima geração possível de arrasto, além de direcionar o escoamento para locais de interesse no carro, como freios e arrefecimento do motor.

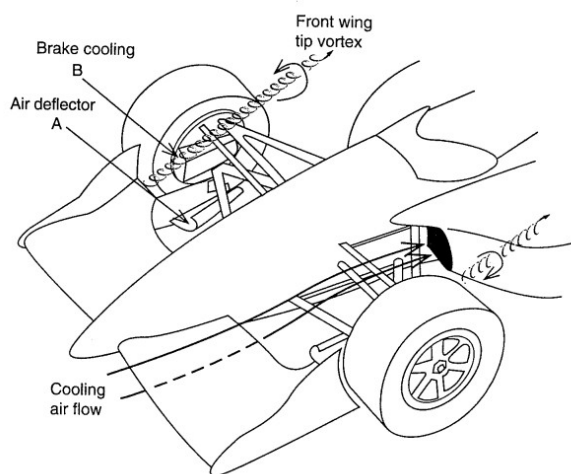


Figura 2.6: Funções da asa dianteira de um carro monoposto. Fonte: (KATZ, 1995)

### 2.5.1.1 A Asa Dianteira

Em um carro de fórmula 1, um dos elementos aerodinâmicos mais importantes é a asa dianteira. pela regulamentação técnica do esporte (FIA, 2022), para a temporada de 2023, a asa dianteira poderá ter até quatro elementos principais e elementos secundários como *Gourney Flaps* e *endplates*.

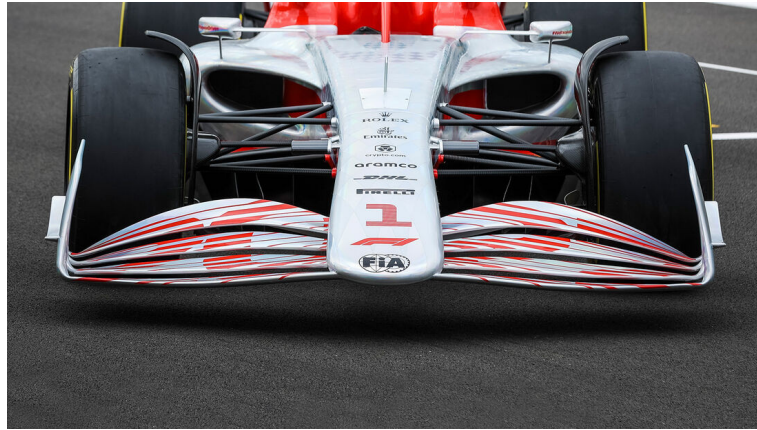


Figura 2.7: Asa dianteira do carro modelo de regulamentação técnica para a temporada de 2022

No caso estudado neste projeto, porém, a asa possuirá apenas dois elementos e as *endplates* e é baseada nas asas utilizadas segundo a regulamentação técnica da temporada de Fórmula 1 de 1998.



(a) McLaren MP4/13, pilotado por Mika Häkkinen durante a temporada de 1998



(b) Ferrari F300, pilotado por Michael Schumacher durante a temporada de 1998

Figura 2.8: Carros McLaren e Ferrari, temporada de 1998

Esta configuração permite um ângulo de ataque maior com uma menor possibilidade de descolamento da camada limite do que uma asa de único elemento como visto na figura 2.9

### 2.5.1.2 Sustentação e Downforce

Asas utilizadas em aplicações automobilísticas possuem o mesmo fundamento de funcionamento que asas utilizadas em aplicações aeroespaciais. Porém, como o objetivo

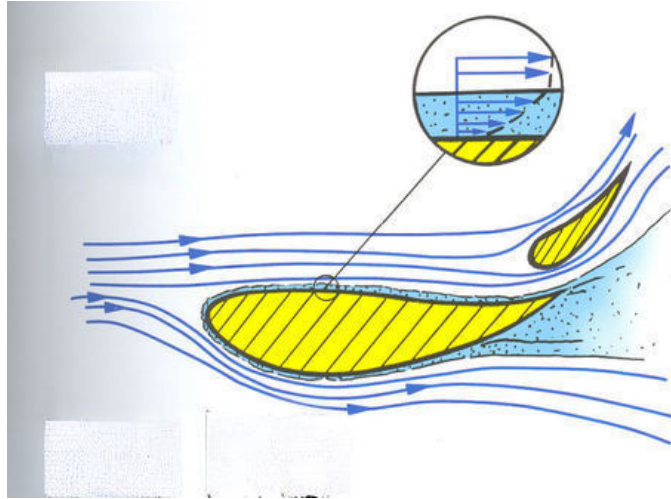


Figura 2.9: Linhas de corrente por um perfil de asa multielemento. Adaptado de: (TREMAYNE, 2009)

final é a geração de *Lift* ou sustentação negativa, esta força é renomeada para *Downforce* pois possuem sua superfície de sucção voltada na direção do chão e sua superfície de pressão voltada para cima.

## 2.6 Escoamentos Turbulentos

A caracterização de escoamentos entre laminares ou turbulentos nem sempre é trivial, sendo possível ainda que o escoamento em estudo se encontre em um ponto de transição dentro os dois regimes. Porém, a condição turbulenta é bem definida e pode ser utilizada para determinar se um dado escoamento possui uma natureza turbulenta. Segundo Tennekes e Lumley (1972), escoamentos turbulentos possuem as seguintes características:

- Irregularidade: A aleatoriedade é intrínseca a escoamentos turbulentos, tornando uma solução analítica bastante difícil. Em vez disso, são utilizados métodos estatísticos para lidar com os fenômenos turbulentos.
- Difusividade: A grande variação de velocidade ao longo de uma linha de corrente gera um efeito no escoamento de rápida mistura do fluido e de grandes taxas de transferência de calor, momento e massa.
- Grandes Números de Reynolds: Escoamentos turbulentos ocorrem quando as forças inerciais do escoamento se sobrepõem em grande medida com relação às forças viscosas. De modo geral, os escoamentos podem ser considerados totalmente turbulentos a partir de  $Re > 10^6$  (WHITE; CHUL, 2018).
- Flutuações tridimensionais de vorticidade: A turbulência é um fenômeno tridimensional e rotacional, portanto é caracterizada por altos níveis de flutuação de



vorticidade.

- Dissipação: Os efeitos viscosos advindos do cisalhamento de camadas do fluido são responsáveis pela dissipação da energia cinética do escoamento em calor.
  - Natureza contínua: Apesar do decaimento dos vórtices turbulentos em vórtices cada vez menores durante a dissipação da energia do escoamento, existe uma escala de vórtices que é pequena o suficiente para se dissipar e não gerar novos *eddys*. Essa é a escala de Kolmogorov e, apesar de suas dimensões serem diminutas, não são pequenas o suficiente para alcançar a escala molecular. Portanto, o fenômeno da turbulência pode ser entendido como inteiramente dentro da mecânica dos meios contínuos.

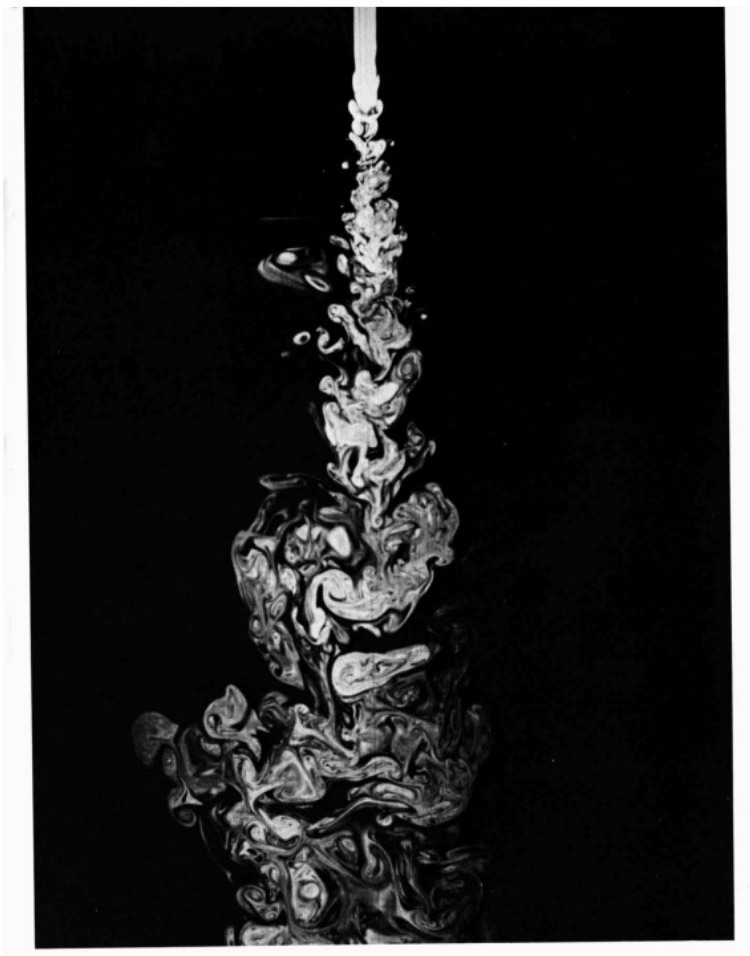


Figura 2.10: Visualização de escoamento turbulento. Fonte: (DYKE, 2008)

# Capítulo 3

## Arcabouço Teórico: Métodos Computacionais para Mecânica dos Fluidos

Tradicionalmente, nas aplicações de engenharia, projetistas se viram forçados a utilizar métodos empíricos para trabalhar com fluidodinâmica devido à impossibilidade de solução analítica para a equação de Navier Stokes (2.16). Estes métodos são adequados para problemas que possam ser descritos como função de poucos parâmetros, mas em problemas de geometrias complexas, que são dependentes de diversos parâmetros, os resultados empíricos só podem ser alcançados por meio de hipóteses simplificadoras que limitem a quantidade de parâmetros independentes (FERZIGER; PERIĆ; STREET, 2020). Com o advento dos computadores modernos, foi possível o desenvolvimento de métodos numéricos que podem complementar resultados experimentais ou até servir como alternativa em situações quando métodos empíricos não seriam suficientes ou inviáveis tecnologicamente.

A disciplina que estuda os métodos computacionais numéricos que podem ser aplicados em mecânica e dinâmica dos fluidos é chamada de **dinâmica dos fluidos computacional** ou, em inglês, **CFD** (*Computational Fluid Dynamics*).

Neste projeto, o método empregado para a solução numérica das equações governantes foi o método dos volumes finitos.

### 3.1 O Método dos Volumes Finitos

O método dos volumes finitos é, antes de tudo, um método numérico de solução de equações diferenciais. Isso significa que ele utiliza uma aproximação algébrica para "simular" uma integração em um domínio finito e definido (MALISKA, 2023). Para isso, deve-se discretizar os valores diferenciais da ED. Logo, quanto maior for a resolução da

discretização do domínio diferencial, melhores serão os resultados do método numérico.

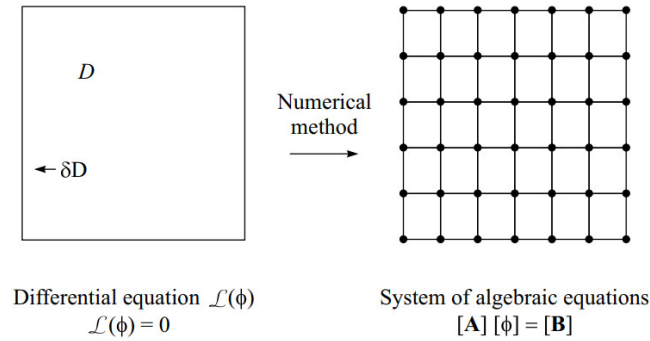


Figura 3.1: Função do método numérico. Fonte: (MALISKA, 2023)

Idealmente, uma malha de resolução global de dimensão na escala de Kolmogorov seria suficiente para uma quase perfeita representação do escoamento da situação estudada. Naturalmente, apesar dos métodos numéricos serem matematicamente capazes de resolver problemas de resolução tão alta, este tipo de empreitada se torna tecnologicamente inviável devido às limitações de capacidade de processamento e memória dos computadores que estarão resolvendo as equações algébricas correspondentes. Por isso, deve-se encontrar um meio-termo entre resolução boa o suficiente para representar de forma acurada os fenômenos do escoamento e baixa o suficiente para que a simulação seja feita em tempo hábil e sem desperdício de recursos computacionais, visto que a influência dos erros de discretização presentes em malhas de resolução mais baixa podem ser minimizada com uma geração inteligente da malha. Na geometria complexa estudada neste trabalho, esta minimização da influência dos erros será obtida por meio da utilização de zonas de refinamento local de malha e/ou malha de camada limite bem definida nas superfícies que apresentam a condição de não-escorregamento.

O método dos volumes finitos possui um procedimento de resolução das equações algébricas advindas das equações governantes do problema segundo uma metodologia *Cell-Center* (FERZIGER; PERIĆ; STREET, 2020). Isto significa que, durante a resolução das equações, a variável desconhecida se encontra dentro do elemento de volume gerado na discretização do problema, e portanto, as equações são aplicadas de um elemento para o próximo tomando cada um como um volume de controle finito.

Também é importante denotar que, para o começo da resolução do sistema de equações, são necessários dados e valores das variáveis de interesse e informação sobre a natureza das fronteiras do volume total do problema. A isto se dá o nome de condições de contorno do problema. As condições de contorno são o ponto de partida e suas propriedades devem ser definidas *a priori*. Um exemplo da utilização da condição de contorno para início da resolução de um problema se encontra na figura 3.3.

Por fim, existem dois métodos para a solução final do sistema de equações no MVF,

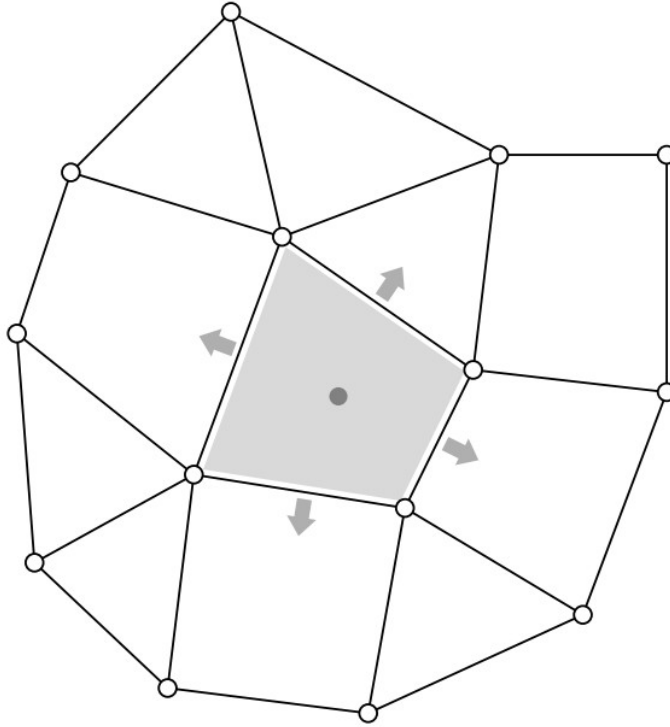


Figura 3.2: MVF por metodologia de resolução *Cell-Center*. Fonte: (MALISKA, 2023)

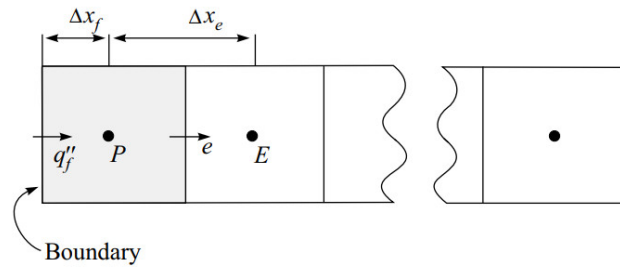


Figura 3.3: Papel das Condições de contorno na resolução por volumes finitos. Fonte: (MALISKA, 2023)

pode-se resolver o sistema de equações de maneira direta, por algoritmos complexos de álgebra linear, ou pode-se utilizar um método iterativo. O método direto de resolução possui um custo computacional muito grande, por isso, a grande maioria dos trabalhos em CFD se dá por meio do método iterativo. Este consiste em estimar inicialmente um valor para as variáveis dependentes em todo o domínio e testar para ver se as equações governantes resultam nas condições de contorno na fronteira. No caso da falha da verificação, ajustam-se os valores das variáveis dependentes iterativamente até que se encontre uma convergência pré-determinada dos valores, resultando em uma diferença mínima nas equações governantes.

## 3.2 O Processo de Trabalho em Dinâmica dos Fluidos Computacional

Dada uma geometria pré-definida do problema, o processo de trabalho para a realização de uma simulação de dinâmica dos fluidos computacional pode ser dividido em três procedimentos.

### 3.2.1 Pré-Processamento

O pré-processamento é o procedimento pelo qual se discretiza o domínio do problema em uma malha de elementos finitos de resolução suficientemente grande para a captura dos fenômenos a serem estudados como exposto na figura 3.1.

Neste passo, também é importante a definição das entidades geométricas da fronteira da malha que serão utilizadas para as condições de contorno.

Seu resultado, na prática, é um arquivo de malha que será utilizado no programa de simulação para a resolução do sistema de equações.

### 3.2.2 Simulação

Na simulação, deve-se preparar as configurações do *software solver*, ou solucionador, que fará os cálculos algébricos de aproximação das equações governantes do problema. Além disso, neste passo, também se roda a simulação. Para isso é importante que tanto a malha quanto a configuração do *solver* sejam adequados e condizentes com os recursos computacionais disponíveis.

O resultado deste procedimento é a coletânea dos valores das variáveis dependentes em cada ponto da malha.

### 3.2.3 Pós-Processamento

O pós-processamento é responsável pela tradução dos resultados numéricos dos cálculos algébricos da simulação em informação útil para o projetista ou pesquisador. Isso pode ser feito por meio de gráficos, tabelas ou visualizações de valor de variáveis no domínio.

## 3.3 Ferramentas de Código Aberto

Os softwares open source são programas cujo código-fonte está disponível para uso, modificação e distribuição livremente. Isso significa que qualquer pessoa pode colaborar

e melhorar o software. Algumas das vantagens deste tipo de programa são: o baixo custo ou gratuidade, a colaboração de muitos desenvolvedores, segurança aumentada, flexibilidade e ampla utilização em muitas áreas.

No entanto, há também desvantagens relacionadas ao software open source, como a falta de suporte técnico formal, conflitos entre desenvolvedores, falta de investimento financeiro e, muitas vezes, capacidade e funcionalidades limitadas em relação aos programas comerciais.

### 3.4 Modelagem de Escoamentos Turbulentos

Como a grande maioria dos escoamentos possuem natureza turbulenta (POPE, 2000), é de grande interesse a investigação de métodos para levar a turbulência em conta em simulações computacionais. Para isso, existem três grandes métodos.

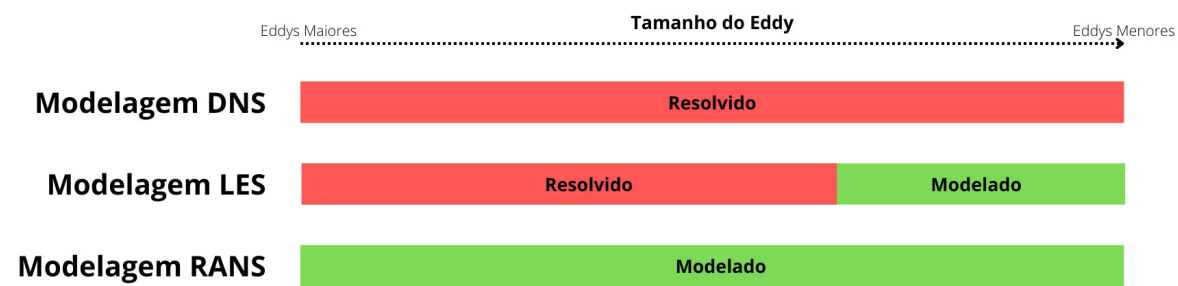


Figura 3.4: Comparativo de metodologias de modelagem de turbulência

#### 3.4.1 Modelagem Direta (DNS)

A modelagem direta de turbulência é um método de simulação que busca solucionar completamente o espectro dimensional de estruturas turbilhonares, logo, existe uma necessidade implícita de resolução altíssima de malha. Isso faz com que este método seja ineficiente em aplicações reais de engenharia devido à grande capacidade computacional necessária, apesar de ser o método que melhor representa a turbulência (FERZIGER; PERIĆ; STREET, 2020).

#### 3.4.2 Modelagem por Equações de Navier-Stokes Filtradas (LES)

Os modelos de turbulência da categoria LES são caracterizados por não resolver todo o espectro dimensional de *eddys* formados no escoamento. Este modelo se prende à solucionar os *eddys* de tamanho relativamente grande. A partir de uma dimensão especificada, o modelo começa a tratar turbulência como uma viscosidade *eddy*. Isso

permite que se tenha uma boa noção do comportamento da turbulência sem elevar demasiadamente o custo computacional da simulação.

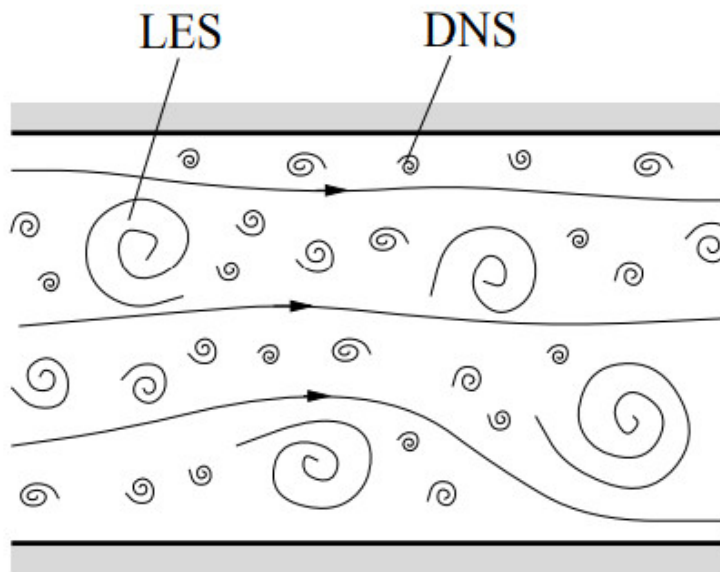


Figura 3.5: Representação esquemática dos tamanhos mínimos de *eddys* nos modelos DNS e LES. Fonte: (FERZIGER; PERIĆ; STREET, 2020)

Ao longo do tempo, os modelos DNS e LES poderiam desenhar um gráfico da velocidade de um ponto no escoamento como na figura 3.6.

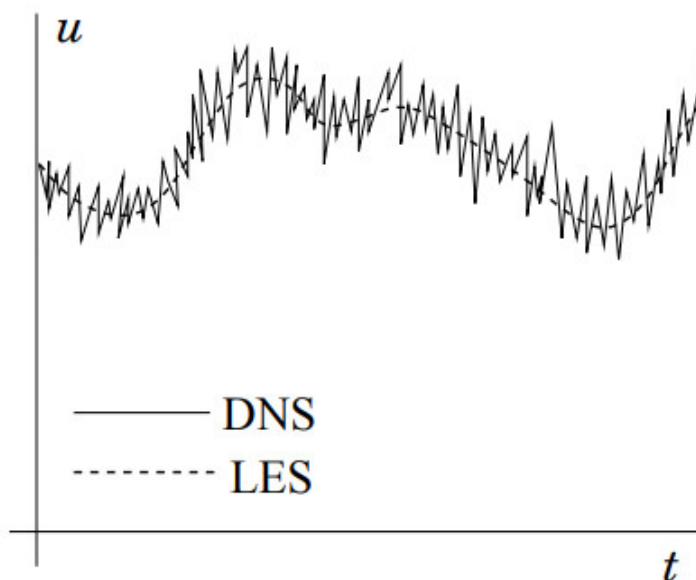


Figura 3.6: Gráfico exemplo das diferenças da variação de velocidade em um ponto no escoamento nos modelos DNS e LES. Fonte: (FERZIGER; PERIĆ; STREET, 2020)

### 3.4.3 Modelagem por Equações Médias de Reynolds (RANS)

Os métodos DNS e LES são, para algumas aplicações da engenharia, exageradamente precisos, e portanto, pode-se desejar um modelo que necessite de menos recursos computacionais. Neste caso, o modelo mais adequado é o modelo RANS (FERZIGER; PERIĆ; STREET, 2020). Na prática, pode-se utilizar a modelagem RANS de turbulência em situações de escoamento permanente pois toda a instabilidade é absorvida na média das equações governantes e tomada como parte da turbulência.

Em um escoamento permanente, cada variável pode ser escrita como o somatório de um valor médio ( $\bar{\phi}$ ) com um valor particular flutuante ( $\phi'$ ). Assim, tem-se:

$$\phi(x_i) = \bar{\phi}(x_i) + \phi'(x_i, t) \quad (3.1)$$

Sendo que:

$$\bar{\phi}(x_i) = \lim_{T \rightarrow \infty} \int_0^T \phi(x_i, t) dt \quad (3.2)$$

Isto está visualizado na figura 3.7

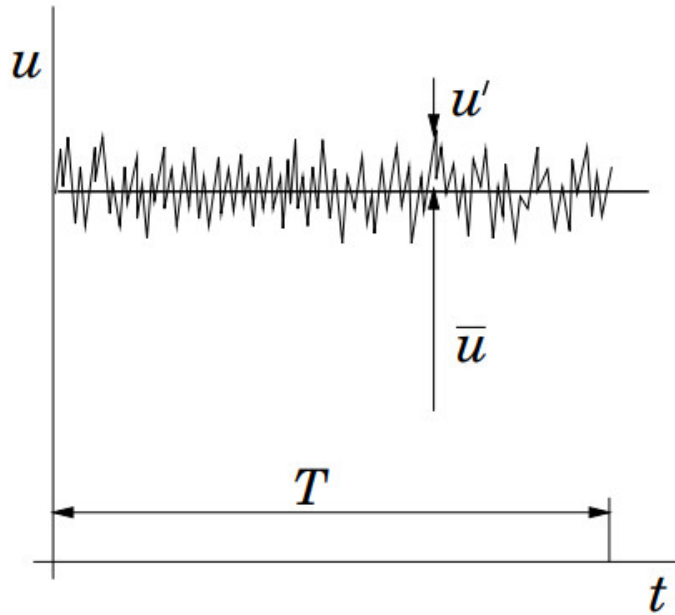


Figura 3.7: Gráfico exemplo das componentes da velocidade em um ponto em um escoamento permanente no modelo RANS. Fonte: (FERZIGER; PERIĆ; STREET, 2020)

O modelo RANS se baseia na ideia de tomar a média de todas as variáveis do escoamento, como  $\bar{\phi}' = 0$ , ao tomar a média das equações governantes. Fazendo isso, obtém-se um sistema de equações aberto (devido à presença da tensão de Reynolds e da taxa de turbulência escalar) que, quando fechado por meio de alguma aproximação, como o modelo de turbulência SST, desenvolvido por Menter (1994), pode ser resolvido



numericamente.

### 3.4.4 O Modelo de Turbulência SST

O modelo SST, utiliza uma abordagem de duas equações para descrever a turbulência. Ele utiliza a equação  $k$ - $\omega$  próximo à parede, onde o fluxo é fortemente afetado pela camada limite. Essa equação fornece uma melhor previsão da turbulência nessas regiões. Fora da camada limite, o modelo muda para a equação  $k$ - $\epsilon$ , que é mais adequada para descrever o comportamento turbulento em áreas distantes da parede. (MENTER, 1994)

Ao combinar essas duas equações, o modelo SST é capaz de lidar com uma ampla gama de fluxos, desde fluxos suaves até fluxos separados e recirculações. Ele se mostrou eficaz em várias aplicações de engenharia, como aeronáutica, engenharia de veículos e projetos de turbomáquinas. (MENTER; KUNTZ; LANGTRY, 2003)

O equacionamento de Menter, Kuntz e Langtry (2003) para o modelo SST de turbulência é dado como:

$$\frac{\partial(\rho k)}{\partial t} + \frac{\partial(\rho U_i k)}{\partial x_i} = \tilde{P}_k - \beta^* \rho k \omega + \frac{\partial}{\partial x_i} \left[ (\mu + \sigma_k \mu_t) \frac{\partial k}{\partial x_i} \right] \quad (3.3)$$

$$\begin{aligned} \frac{\partial(\rho \omega)}{\partial t} + \frac{\partial(\rho U_i \omega)}{\partial x_i} = & \alpha \rho S^2 - \beta \rho \omega^2 + \frac{\partial}{\partial x_i} \left[ (\mu + \sigma_\omega \mu_t) \frac{\partial \omega}{\partial x_i} \right] \\ & + 2(1 - F_1) \rho \sigma_{\omega 2} \frac{1}{\omega} \frac{\partial k}{\partial x_i} \frac{\partial \omega}{\partial x_i} \end{aligned} \quad (3.4)$$

Para uma função de mistura  $F_1$  definida como:

$$F_1 = \tanh \left\{ \left\{ \min \left[ \max \left( \frac{\sqrt{k}}{\beta^* \omega y}, \frac{500v}{y^2 \omega} \right), \frac{4\rho \sigma_{\omega 2} k}{CD_{k\omega} y^2} \right] \right\}^4 \right\} \quad (3.5)$$

Sendo:

$$CD_{k\omega} = \max \left( 2\rho \sigma_{\omega 2} \frac{1}{\omega} \frac{\partial k}{\partial x_i} \frac{\partial \omega}{\partial x_i}, 10^{-10} \right) \quad (3.6)$$

e  $y$  a distância até a parede mais próxima.

$F_1$  é igual a zero longe da superfície, recaindo no modelo  $k$ - $\epsilon$ , e se torna igual a 1 dentro da camada limite formando o modelo  $k$ - $\omega$ .

A viscosidade eddy turbulenta é definida por:

$$v_t = \frac{a_1 k}{\max(a_1 \omega, S F_2)} \quad (3.7)$$

Onde  $S$  é a medida invariante da taxa de deformação e  $F_2$  é uma segunda função de mistura definida como:

$$F_2 = \tanh \left[ \left[ \max \left( \frac{2\sqrt{k}}{\beta^* \omega y}, \frac{500v}{y^2 \omega} \right) \right]^2 \right] \quad (3.8)$$

Além disso, um limitador de produção é utilizado para prevenir o acúmulo de turbulência em regiões de estagnação:

$$P_k = \mu_t \frac{\partial U_i}{\partial x_j} \left( \frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i} \right) \rightarrow \tilde{P}_k = \min(P_k, 10 \cdot \beta^* \rho k \omega) \quad (3.9)$$

Onde, todas as constantes são calculadas por uma combinação das constantes correspondentes dos modelos  $k - \varepsilon$  e  $k - \omega$  via  $\alpha = \alpha_1 F + \alpha_2 (1 - F)$  etc. (MENTER; KUNTZ; LANGTRY, 2003)

As constantes para esse modelo são:

- $\beta^* = 0.09$
- $\alpha_1 = 5/9$
- $\beta_1 = 3/40$
- $\sigma_{kl} = 0.85$
- $\sigma_{\omega 1} = 0.5$
- $\alpha_2 = 0.44$
- $\beta_2 = 0.0828$
- $\sigma_{k2} = 1$
- $\sigma_{\omega 2} = 0.856$

# Capítulo 4

## Metodologia de Projeto

### 4.1 Proposta do Projeto

A metodologia deste trabalho será dividida em quatro etapas, referentes aos processos necessários para a execução de um ciclo de simulações bem sucedido, sendo que também haverá a análise dos resultados qualitativos e quantitativos obtidos após o pós-processamento para comparação dos resultados com os experimentos feitos por Zerihan (2001) para a validação do ciclo como um todo.

Os programas utilizados em cada etapa estão dispostos na tabela a seguir:

Fase	Programas		
Modelagem da Geometria	Fusion 360	SALOME (Módulo Geometry)	
Pré-Processamento	SALOME (Módulo Mesh)	Gmsh	Paraview
Simulações Preliminares	SU2		
Pós-Processamento	Paraview		

Tabela 4.1: Programas Utilizados no Projeto.

#### 4.1.1 Apresentação dos Programas Utilizados

##### 4.1.1.1 SALOME

SALOME é um *software* de simulação numérica de código aberto desenvolvido por uma parceria entre a Open-Cascade e a Rede Nacional Francesa para Pesquisa e Inovação em Tecnologia de *software* (*French National Network for Research and Innovation in software Technologies (RNNTL)*). Ele é projetado para resolver problemas de física e facilitar a integração de soluções de cálculo específicas. O módulo Mesh do SALOME é responsável pela geração de malhas e inclui uma variedade de algoritmos de

malha, especialmente adequados para métodos de elementos finitos e volumes finitos. As malhas podem ser enriquecidas com grupos / rótulos para distinguir diferentes regiões da geometria e permitir a diferenciação das propriedades das malhas ou até mesmo dos tipos de malhas. (SALOME, 2022b)

Além disso, o SALOME inclui diversas funções de controle de qualidade da malha e permite a edição de malhas para excluir, adicionar ou transformar elementos de malha. As operações de malha também podem ser gerenciadas completamente por scripts Python, garantindo a reprodutibilidade e parametrização do processo de simulação. O projeto SALOME começou no início dos anos 2000 com base em duas necessidades industriais: implementar uma solução para problemas multi-físicos e facilitar a integração de soluções de cálculo específicas. (SALOME, 2022a)

#### 4.1.1.2 Gmsh

Gmsh é um *software* de malha aberto que permite a criação de malhas para elementos finitos, utilizadas na análise numérica. Ele é capaz de lidar com geometrias complexas e oferece uma grande variedade de opções de malha. (GEUZAINÉ, 2022)

Uma das principais características do Gmsh é a sua capacidade de importar e exportar vários formatos de arquivos de geometria, incluindo IGES, STEP, STL, VRML, entre outros. Isso permite ao usuário trabalhar com geometrias já existentes e exportar as malhas para outros *softwares* de análise.

Além das opções de malha, o Gmsh também oferece ferramentas para a visualização e análise de malhas, incluindo ferramentas para medir elementos, calcular áreas e volumes, e visualizar a distribuição de nós. Ele também oferece suporte para a criação de scripts, permitindo que os usuários automatizem tarefas repetitivas e apliquem configurações de malha padrão.

#### 4.1.1.3 SU2

A suite SU2 (*Stanford University Unstructured*) de simulação de problemas físicos foi desenvolvida em uma parceria entre a universidade de *Stanford* e a *Boeing* com o objetivo de resolver equações diferenciais parciais sobre malhas não-estruturadas (ECONOMON et al., 2015). O seu grande diferencial é a sua natureza *open-source* e flexibilidade.

Para a utilização do SU2 são necessários dois arquivos:

- Arquivo de malha em formato .cgns ou .su2 nativo obtido da exportação da malha do pré-processamento
- Arquivo de configuração de simulação .cfg

#### 4.1.1.4 ParaView

Paraview é um *software* de visualização de dados de simulação científica e de engenharia de alta performance. É desenvolvido como *software* livre pela *Kitware, Inc.* e apoiado pela *U.S. Department of Energy (DOE)* e pelo *National Nuclear Security Administration (NNSA)*.

O Paraview permite aos usuários visualizar, explorar e analisar grandes conjuntos de dados em três dimensões. Ele oferece uma ampla gama de ferramentas de visualização, incluindo cortes, sondas, recortes e renderizações, bem como suporte para animações. Além disso, o Paraview permite aos usuários aplicar uma ampla gama de operações matemáticas e estatísticas em seus dados, incluindo filtragem, agregação e derivação (HANSEN; JOHNSON, 2005).

### 4.1.2 Observações Sobre os Programas Utilizados

É importante denotar que o programa Paraview, apesar de não ser um gerador de malha como os programas Gmsh e SALOME, foi utilizado no pré-processamento para a visualização das malhas geradas, dado que os geradores de malha não eram capazes de visualizar a malha gerada na máquina utilizada para este projeto.

Observa-se também que, apesar da capacidade do Gmsh de gerar malhas tridimensionais, não há como gerar malhas tridimensionais de camada limite (GEUZAINÉ, 2022), um dos aspectos importantes da malha deste trabalho. Portanto, as malhas foram geradas pelo SALOME e a utilização do Gmsh se limita a conversão do arquivo de malha gerado pelo SALOME para um arquivo de malha que o *solver* SU2 é capaz de ler.

## 4.2 Modelagem da Geometria de Zerihan

A principal geometria de interesse deste projeto é a asa multielemento dentro do túnel de vento da universidade de Southampton, na Inglaterra, dos experimentos e tese de doutorado de Zerihan (2001).

O processo de criação de geometria a ser utilizada nos programas de geração de malha pode ser realizado diretamente dentro dos programas de geração de malha Gmsh e SALOME (sendo que o SALOME trabalha na geometria dentro de um módulo diferente do módulo de geração de malha e o Gmsh no mesmo ambiente) ou em programas CAD diferentes, exportando a geometria em formatos compatíveis para importação nos referidos *softwares* de geração de malha.

Para a simulação dos experimentos de Zerihan (2001) é necessária a modelagem de uma asa frontal de um carro de fórmula 1, mais especificamente do Tyrrell 026 da

temporada de 1998.



Figura 4.1: Tyrrell 026, temporada de 1998.

#### 4.2.1 Coleta e Tratamento dos Dados de Zerihan

Em sua tese de doutorado, Zerihan (2001) faz medições e apresenta uma tabela com as coordenadas dos pontos medidos nos perfis da asa principal, chamada de *Main Wing*, e do flap auxiliar, chamado de *Flap Wing*. esses dados são apresentados nas tabelas I.2 e I.1 nos anexos, em "Dados de coordenadas dos pontos dos perfis de asa".

É importante denotar que nas tabelas I.2 e I.1, os dados são adimensionalizados em relação à corda do conjunto Main-Flap Wing, a ser chamado de asa frontal. A corda da asa frontal para efeitos de adimensionalização é dada na tese como  $c = 380 \text{ mm}$ . Assim, para a modelagem tridimensional da asa frontal pode-se multiplicar cada um destes pontos pela corda e importar no programa de CAD como uma *spline* para suavização do perfil. É importante mostrar também que no bordo de fuga de cada asa os pontos não se encontram, necessitando uma linha para fechar a área de cada perfil. Isso se dá pois, como os pontos são advindos de um experimento real, o processo de fabricação das asas utilizadas por Zerihan deve ser levado em conta, não sendo portanto possível a criação de um perfil de asa com bordo de fuga de pontos exatamente coincidentes.

Esses cálculos de redimensionalização das coordenadas dos pontos da asa foram feitos por meio de uma planilha de cálculos, que se encontra no apêndice J.

#### 4.2.2 Modelagem Tridimensional do Volume de Ar Dentro do Túnel de Vento

Como o método de volumes finitos pede a discretização do espaço tridimensional a ser simulado, ou seja, o volume de fluido, além da geometria da asa também deverá ser modelada a geometria interna do túnel de vento.

#### 4.2.2.1 Geometria da Asa Frontal

Com os perfis desenhados no Fusion 360, pode-se extrudá-los 550 mm para cada lado para se produzir a asa de 1100 mm de envergadura que Zerihan (2001) utilizou em sua tese. Além disso, pode-se observar nas imagens dos testes experimentais que a asa frontal possui, além dos perfis das main e flap wings, uma endplate em cada lado, como visto na figura 4.2.



Figura 4.2: Experimento de Zerihan. Fonte: (ZERIHAN, 2001)

Como em seu trabalho Zerihan observa que as endplates são baseadas nas utilizadas no Tyrrell 026 e possuem dimensões 400mm x 170mm x 4mm pode-se facilmente criá-las no modelo CAD. O problema é seu posicionamento. Infelizmente, Zerihan não dá as coordenadas de posicionamento das endplates, então, para a modelagem CAD da asa, utilizou-se uma imagem da retirada da tese como referência para o posicionamento do componente.

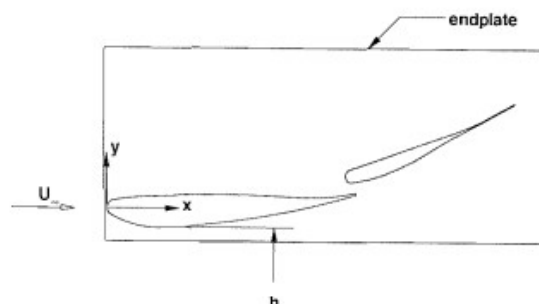


Figura 4.3: Figura 1 de Zerihan (2001)

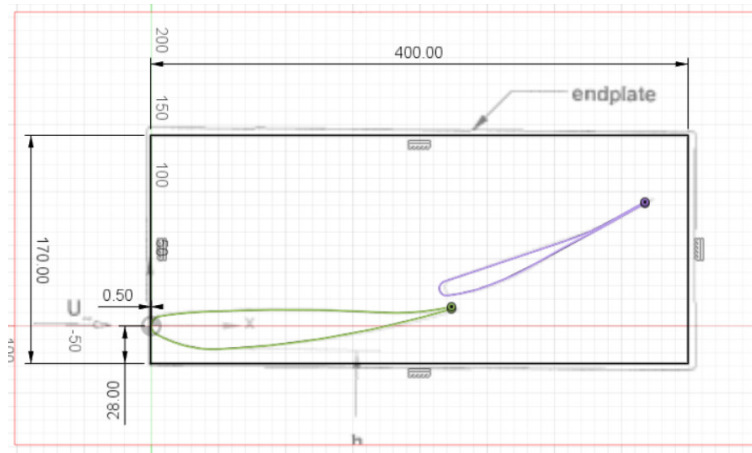


Figura 4.4: Posicionamento das *endplates* de acordo com a Figura 1 de Zerihan (2001)

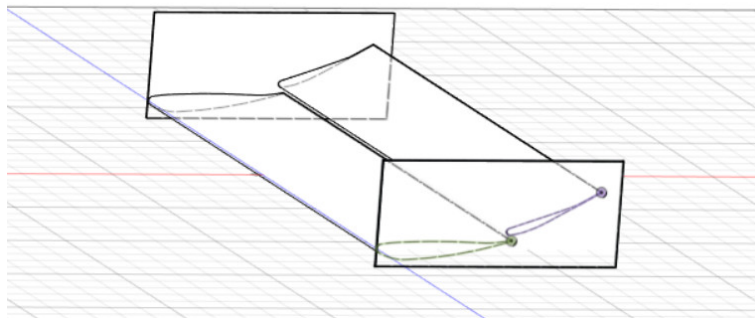


Figura 4.5: Geometria da Asa Frontal

#### 4.2.2.2 Geometria do Túnel de Vento

O túnel de vento também é descrito na tese de doutorado e possui perfil de dimensões 2.1m x 1.7m. Na imagem 4.2 pode-se ver que esse túnel também possui chanfros nas arestas do perfil. Como a dimensão destes chanfros também não é dada na tese de doutorado, foi adotado uma dimensão que fizesse a geometria CAD se assemelhar o máximo possível às fotos do experimento. Após isso, extrudou-se o perfil de modo que a dimensão longitudinal do túnel de vento fosse 8 metros, com a asa frontal no centro. Essa dimensão também não está descrita na tese de Zerihan, portanto foi arbitrada de maneira exagerada com o intuito de garantir a visualização do desenvolvimento completo do escoamento nas simulações.

#### 4.2.2.3 Geometria do Volume de Ar

Para a modelagem do volume de ar dentro do túnel de vento, é necessário definir a altura da asa até o chão. Como o intuito deste trabalho é a validação de seus resultados com os obtidos por Zerihan (2001), foram criadas geometrias referentes a todas as alturas  $h$  para as quais Zerihan fez medições dos coeficientes adimensionais de arrasto e sustentação. Essas alturas estão dispostas na tabela II.1 do anexo II: "Resultados



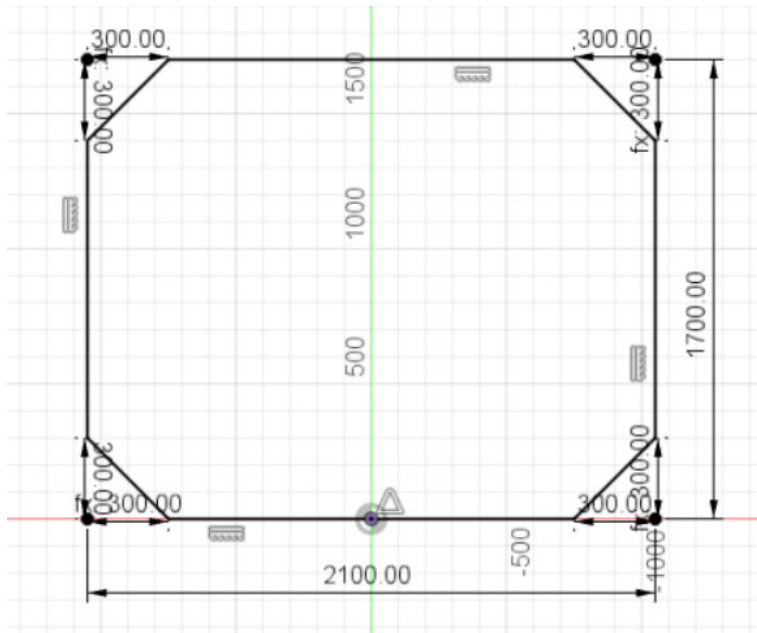


Figura 4.6: Perfil do Túnel de Vento

Experimentais de Zerihan para CD e CL". No total, são 24 geometrias com a posição da asa em alturas variando entre  $h = 17 \text{ mm}$  e  $h = 225 \text{ mm}$ .

Assim, para se obter a geometria tridimensional do volume de ar dentro do túnel de vento, cortou-se o volume da asa frontal do volume do túnel de vento para a modelagem da massa de ar dentro do túnel de vento a ser discretizada na fase de pré-processamento.

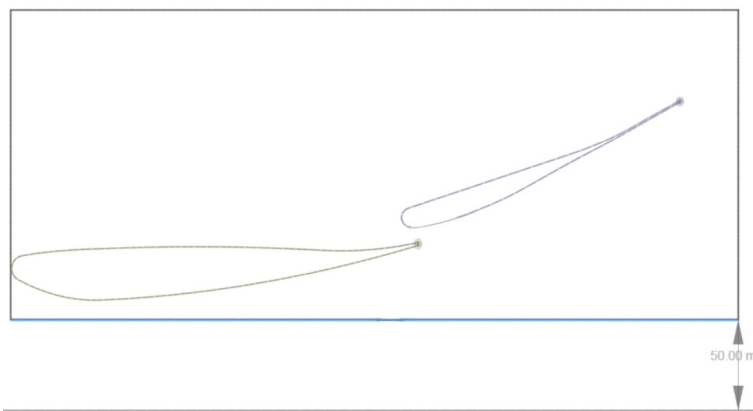


Figura 4.7: Exemplo de definição da altura da asa, no caso  $h = 50 \text{ mm}$

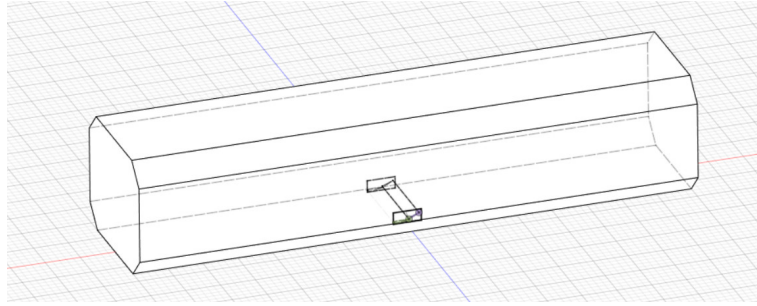


Figura 4.8: Geometria do volume de ar dentro do túnel de vento

#### 4.2.2.4 Estabelecimento do Volume de Refinamento Local de Malha

Com um volume de ar tão grande, é impraticável e desnecessário um alto refinamento global de malha, porém, nas imediações da asa frontal, ainda se deseja uma boa resolução dos elementos de malha. No SALOME, é necessária a criação de uma geometria secundária que esteja associada à zona de refinamento local de malha. Para isso, utilizou-se o módulo *Geometry* do SALOME para a construção de um paralelepípedo dentro do qual se encontra a asa frontal e a malha será mais refinada.

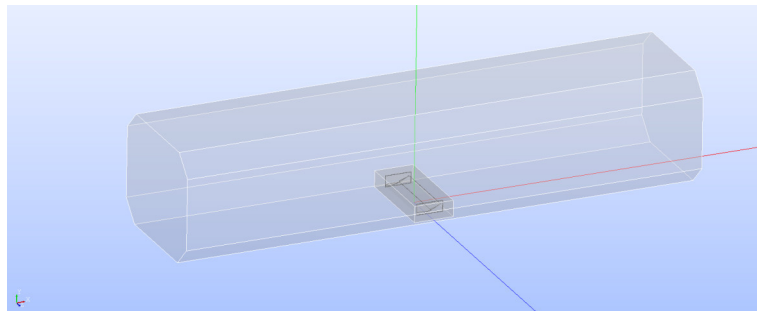


Figura 4.9: Geometria do volume de ar dentro do túnel de vento com volume de refinamento local no SALOME

### 4.2.3 Modelagem Tridimensional da Asa Frontal no Infinito

Além das simulações dos casos da asa frontal dentro do túnel de vento, também será feita uma simulação do caso em que a asa frontal se encontra em um ambiente sem imposições geométricas limitantes ao volume de ar que influenciem em seus coeficientes aerodinâmicos, ou "no infinito", para comparação dos efeitos do túnel de vento nos coeficientes.

Este caso, denominado *hinf*, utiliza a mesma geometria da asa frontal, porém incluída em um cubo de  $20 \times 20 \text{ m}$ , em vez do túnel de vento.

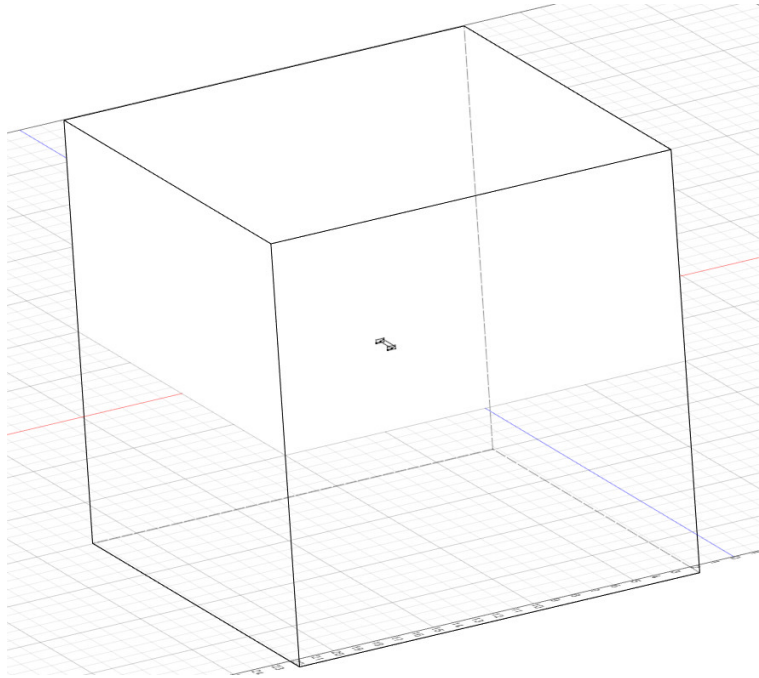


Figura 4.10: Geometria do volume de ar ao redor da asa frontal, no caso *hinf*

## 4.3 Pré-Processamento

### 4.3.1 Determinação da Geometria para as Condições de Contorno do Problema

Antes de começar a geração de malha de fato, é importante definir as geometrias que serão utilizadas nas condições de contorno do problema. Após a importação e estabelecimento do volume de refinamento de malha da geometria escolhida, pode-se determinar as superfícies que irão ser definidas no solver SU2 como condições de contorno. Nomeadamente, no caso da asa frontal no túnel de vento estas superfícies são dadas na tabela 4.2

Superfície	Descrição
Inlet	Superfície de entrada do ar no túnel de vento
Outlet	Superfície de saída do ar do túnel de vento
Walls	Teto e paredes do túnel de vento
Ground	Chão do túnel de vento
Wing	Superfícies da geometria da asa frontal, contando com as endplates

Tabela 4.2: Descrição das geometrias para condições de contorno

Para a definição dessas geometrias no SALOME, deve-se, ainda no módulo *Geometry* selecionar as faces das superfícies e agrupá-las em grupos de superfícies.

Após esse passo, pode-se abrir a geometria construída no módulo *Mesh* e seguir o procedimento para a geração de malha. Após o fim da geração de malha, pode-se

importar os grupos de superfícies do módulo *Geometry* para a malha gerada no módulo *Mesh* em *Mesh > Create Groups from Geometry*. Para a geometria dos experimentos de Zerihan (2001), a árvore do arquivo de malha no módulo *Mesh* ficou como na figura 4.11

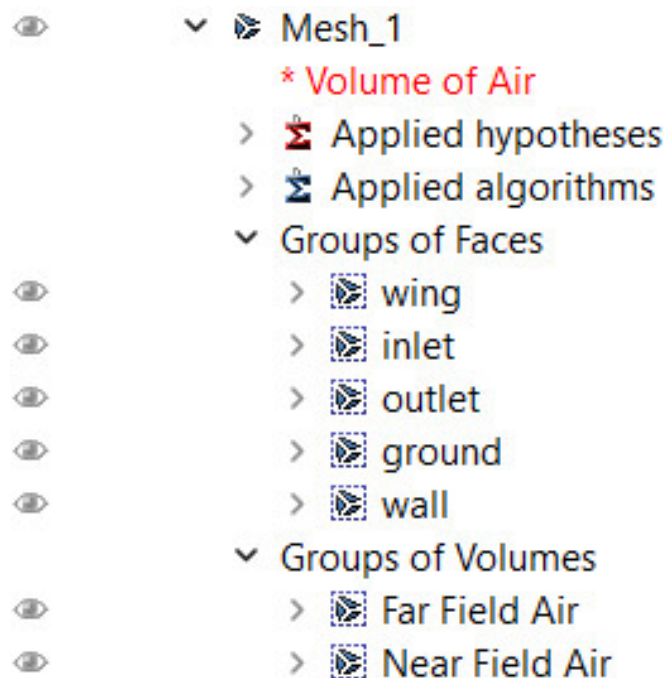


Figura 4.11: Árvore de arquivo de malha do módulo *Mesh* no SALOME

É importante observar que na árvore do arquivo de malha, no caso da geometria dos experimentos de Zerihan (2001), também existem grupos de volumes. Estes grupos são gerados automaticamente a partir da determinação de sub-geometrias de volume para o refinamento local da malha, realizada em 4.2.2 "Modelagem Tridimensional do Volume de Ar Dentro do Túnel de Vento". Isto é importante pois é com esses grupos que serão definidas as condições de geração especial de malha como zona de refinamento local e malha de inflação para camada limite de escoamento. Porém, para a exportação da malha gerada, deve-se excluir estes grupos, pois não são reconhecidos pelo SU2 durante o processo de simulação.

### 4.3.2 Configuração de Malha - Algoritmo Aplicado

O SALOME possui, em seu módulo *Mesh*, a possibilidade de geração de malha por diversos algoritmos estruturados, não estruturados e algoritmos híbridos avançados. Para este caso, foi escolhido o algoritmo NETGEN 1D-2D-3D (SCHOEBERL, 1997) que utiliza os mesmo parâmetros definidos para a geração das malhas 1D, 2D e 3D. este algoritmo foi escolhido pela possibilidade de gerar a zona de refinamento local tridimensional e, especialmente, pela viabilidade de definição da hipótese de geração de malha de inflação.

Assim, utilizaram-se os volumes definidos anteriormente para determinar os tamanhos médios dos elementos em cada um.

Geometria Alvo	Tamanho de Elemento Especificado
Far Field Air	500 mm
Near Field Air	10 mm
wing	4 mm

Tabela 4.3: Tamanhos especificados de elementos da malha de acordos com zona de refinamento local

Como o SALOME permite a seleção de diversas geometrias para zonas de refinamento, foram escolhidos: o volume chamado "Near Field Air" que foi definido em Determinação da Geometria para as Condições de Contorno do Problema e a o conjunto de superfícies da asa frontal "wing". Isto foi feito de forma a aumentar a resolução da malha nos locais de maior interesse e onde se espera que os efeitos aerodinâmicos sejam mais complexos.

### 4.3.3 Configuração de Malha - Hipótese Aplicada

Além da zona de refinamento local de malha também foi estabelecida uma hipótese adicional no algoritmo, esta hipótese comanda a geração de uma malha de inflação das superfícies da asa.

#### 4.3.3.1 Dimensionamento da Malha de Camada Limite

Para o dimensionamento da malha de camada limite foi levado em conta a dificuldade do algoritmo de gerar uma malha de boa qualidade quando duas malhas de camada limite se interceptam no espaço. Por isso, fez-se uma malha de camada limite de espessura  $\delta = 3.5$  mm, 23 camadas e fator de crescimento de 1.2 ao longo de todas superfícies da asa frontal. Isso gera uma primeira camada da malha de camada limite de tamanho  $\Delta S = 1.1 \cdot 10^{-5} m = 0.011 mm$ , calculada pelo método  $y+$  (ANDERSON, 2017).

## 4.4 Simulação e Pós-Processamento

As etapas de simulação e pós-processamento deste projeto foram feitas nos programas SU2 e Paraview, respectivamente.

#### 4.4.1 O Cluster Amadea

Este projeto utilizou o cluster Amadea do Laboratório de Aeroacústica Computacional (Caa Lab) para a realização dos cálculos numéricos do *solver* SU2. O Amadea é baseado no sistema operacional CentOS 7, que é uma distribuição Linux gratuita e de classe corporativa baseada em códigos fonte da Red Hat Enterprise Linux. O CentOS 7 oferece acesso a *softwares* padrão da indústria, incluindo compatibilidade com pacotes de *software* da Red Hat Enterprise Linux. O SU2 também já está instalado no cluster e funciona de maneira comprovada por trabalhos anteriores realizados nele.

O cluster Amadea possui 8 processadores Intel Xeon Phi KNL de 64 *cores* e 256 *threads*, com desempenho computacional de 21,28 Teraflops, 128 GB de memória MC-DRAM e 768 GB de memória SDRAM DDR4. O nó de visualização possui 256 GB de memória RAM, um processador Intel Xeon com 64 "cores" e uma placa de vídeo Nvidia Quadro P5000.

A infraestrutura de *software* básica do cluster Amadea inclui o gerenciador de fila de simulação *Portable Batch System* (PBS), que é um *software* de código aberto que permite a gestão da execução de códigos numéricos usando os recursos de hardware do cluster. O PBS aloca tarefas computacionais de maneira eficiente entre os recursos computacionais disponíveis.

Os nós de cálculo e de visualização são parte da rede interna do cluster Amadea e podem ser acessados diretamente pela rede da Universidade de Brasília ou de forma remota pela internet. A conexão com esses nós é feita por tunelamento SSH através do nó de gerenciamento, que tem conexão com a rede da Universidade. A conexão externa ao nó de gerenciamento é protegida por senha para segurança do cluster. O acesso ao cluster pode ser feito de qualquer ponto da internet, sem a necessidade de presença física do usuário na Universidade. Qualquer computador pessoal com sistemas operacionais diferentes, incluindo ambiente virtual, pode operar o cluster, já que toda a carga computacional dos processos de simulação é suportada pelo cluster.

#### 4.4.2 Configuração dos Arquivos de Configuração .cfg

O arquivo .cfg é um arquivo de texto que contém as opções de simulação do caso em estudo. Para as simulações deste projeto, como foram realizadas 24 simulações relativas aos experimentos de Zerihan (2001), um modelo de arquivo .cfg utilizado se encontra no apêndice deste relatório em "Código dos Arquivos de Configuração do SU2", apêndice L. Os principais parâmetros adicionados aos arquivos .cfg neste trabalho foram:

- $SOLVER = INC\_RANS$  : Indica o tipo de simulação, no caso, incompressível por modelagem RANS de turbulência.
- $INC\_VELOCITY\_INIT = (30.0, 0.0, 0.0)$  : Determina qual a velocidade in-

cial do escoamento e em qual direção, no caso, 30  $m/s$  como nos experimentos de Zerihan (2001).

- $SURFACE\_MOVEMENT = MOVING\_WALL$  ;  $MARKER\_MOVING = (ground)$  e  $SURFACE\_TRANSLATION\_RATE = 30.00.00.0$ : Indicam que o chão possui movimento de 30  $m/s$  como nos experimentos de Zerihan (2001).
- $MARKER\_HEATFLUX = (wing, 0, ground, 0)$  ;  $MARKER\_EULER = (wall)$  e  $MARKER\_FAR = (inlet, outlet)$  : Indicam as condições de contorno do problema, levando em consideração a condição de não-escorregamento apenas para a asa frontal e para o chão.

E a única modificação realizada para cada simulação foi o nome dos arquivos de malha utilizados em cada caso.

Também nos apêndices, há um arquivo `.cfg` relativo ao caso *hinf*, em que não há superfície em movimento e nem paredes ou chão discriminados na malha.

### 4.4.3 Metodologia da Análise de Dados e dos Resultados

#### 4.4.3.1 Análise de Convergência de Iterações de Simulação

Para a análise da convergência dos coeficientes adimensionais aerodinâmicos obtidos por cada simulação ao logo das iterações, foi desenvolvido um código em *Python* que abre as pastas de cada simulação e lê a saída dos valores de CL, CD e CSF (Coeficiente de Força Lateral) e cria um gráfico mostrando a evolução dos valores desde o início da simulação.

Para a comparação com a literatura e validação dos resultados, esse código também calcula a média do último quinto dos valores obtidos e o desvio padrão em formato numérico e adimensional percentual.

Após isso, o programa salva os resultados de cálculo em um arquivo de dados para ser lido posteriormente com mais facilidade.

O código desenvolvido pode ser encontrado nos apêndices deste relatório, em K.1: Monitorador e Gerador de Gráficos dos Coeficientes Aerodinâmicos ao Longo das Iterações.

#### 4.4.3.2 Visualizações de Velocidade e Pressão

Para as visualizações dos campos de velocidade e de pressão do escoamento, foi criado um estado de visualização do ParaView que permitisse gerar as figuras necessárias para a comparação dos resultados devido à diferença de altura de asa  $h$  e que gerasse as

linhas de corrente necessárias para a compreensão do comportamento do escoamento em cada caso.

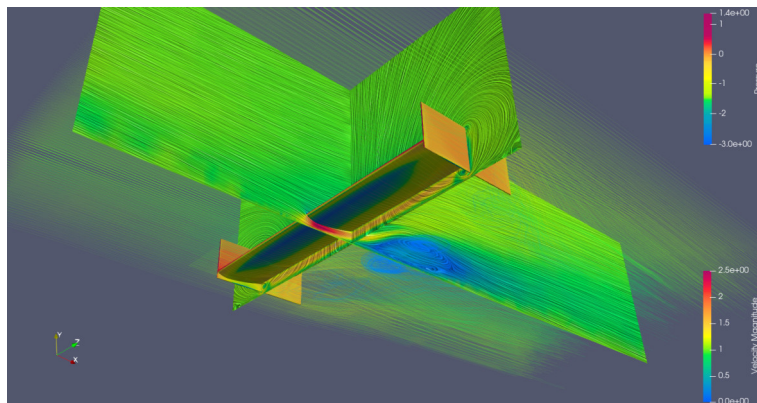


Figura 4.12: Estado de visualização do ParaView para velocidade, pressão e linhas de corrente.

#### 4.4.3.3 Validação de Resultados

Para a validação dos resultados, por meio comparativo com os resultados de Zerihan (2001), foi desenvolvido um código em *Python* que lê os valores da tabela II.1 e os resultados gerados pelo Monitorador e Gerador de Gráficos dos Coeficientes Aerodinâmicos ao Longo das Iterações e gera dois tipos de gráficos, um que é a comparação direta dos resultados de coeficiente de arrasto e de sustentação entre os resultados numéricos e os resultados experimentais, e outro que é a diferença em formato dimensional e em formato percentual entre os resultados numéricos obtidos e os resultados de Zerihan (2001).



# Capítulo 5

## Resultados e Análise

### 5.1 Resultados do Pré-Processamento

Os resultados do pré-processamento foram as malhas de simulação em formato .su2 geradas pelo SALOME e convertidas pelo gmsh.

#### 5.1.1 Casos hx

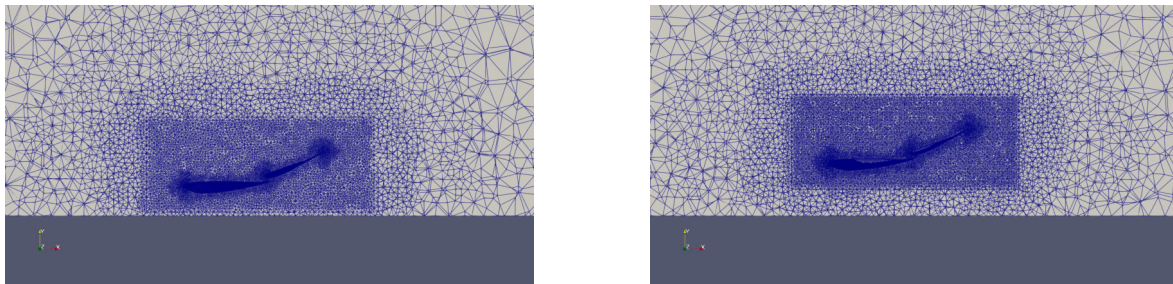
Para os casos hx, todas as malhas foram geradas utilizando exatamente os mesmos parâmetros, porém, devido ao algoritmo de geração de malhas não-estruturadas empregado pelo SALOME e às diferenças geométricas de cada caso, existe variação no número de elementos de malha em cada uma. Por isso, o processo de geração foi realizado de modo que essa variação não fosse grande demais, a fim de diminuir a influência da quantidade de elementos de malha na acurácia dos resultados numéricos de caso a caso, devido aos erros relativos à discretização. Os valores da quantidade mínima, máxima e média de elementos de malha dentre todos os casos estão dispostos na tabela 5.1.

Parâmetro	Número de Elementos de Malha	Caso
Mínimo	$15.016 \cdot 10^6$	h50
Máximo	$18.655 \cdot 10^6$	h110
Médio	$15.837 \cdot 10^6$	—
Desvio Padrão	$0.895 \cdot 10^6$	—

Tabela 5.1: Informações gerais sobre as malhas resultantes do pré-processamento.

Pela tabela 5.1 pode-se ver que, com uma média de 15.837 milhões de elementos e um desvio padrão de 895.392 mil elementos, o desvio padrão da quantidade de elementos de malha para o conjunto de casos é de apenas 5.65%. Logo, será utilizada a hipótese de que os erros de discretização dos casos são da mesma ordem de grandeza.

As malhas com a maior e a menor quantidade de elementos estão representadas nas figuras 5.1a e 5.1b.



(a) Malha do caso h50

(b) Malha do caso h110

Figura 5.1: Comparação das malhas dos casos h50 e h110

Nestas figuras, pode-se ver a zona de refinamento local de malha retangular ao redor do perfil da asa, assim como também o refinamento maior perto da superfície, como determinadas durante a etapa de Pré-Processamento. Já a malha de inflação para captura de camada limite pode ser melhor visualizada na figura 5.2.

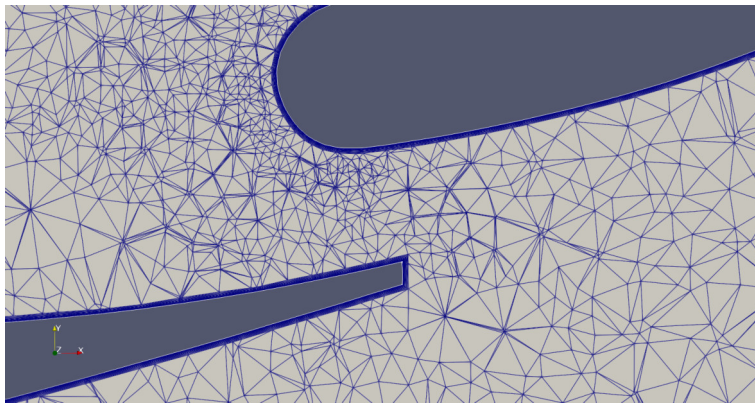


Figura 5.2: Malha de inflação para captura da camada limite.

Figuras relativas às malhas de todos os casos  $hx$  se encontram nos apêndices, em A. Visualizações das Malhas Geradas

### 5.1.2 Caso hinf

Para o caso hinf, como a asa está no espaço não-restrito pelo túnel de vento, espera-se que não haja influência de efeitos de instalação. Logo, pode se usar uma malha mais leve, com uma quantidade menor de elementos de malha, para a simulação do caso.

O resultado foi uma malha com as mesmas zonas de refinamento local, porém com elementos ligeiramente maiores em cada uma. A malha de inflação de camada limite foi replicada dos casos  $hx$ .

Assim, a malha gerada para o caso hinf tem suas propriedades como mostrado na tabela 5.2 e pode ser vista em corte no plano  $xy$  na figura 5.3

Malha hinf	
Número de Elementos de Malha	$10.620 \cdot 10^6$
Tamanho Máximo de Elemento	500 mm
Tamanho Mínimo de Elemento	0.01 mm

Tabela 5.2: Propriedades da malha do caso hinf

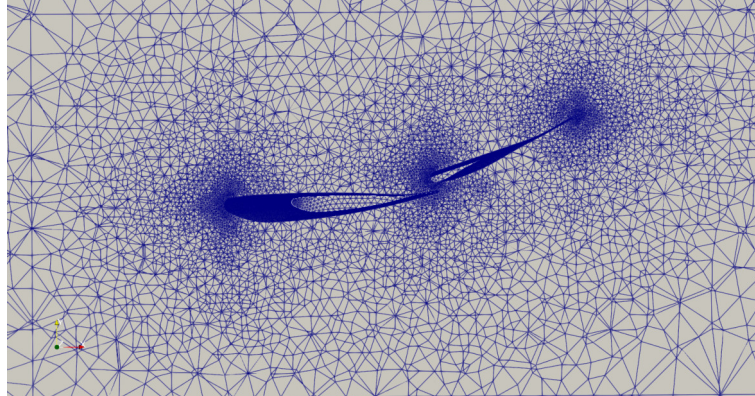


Figura 5.3: Malha do caso hinf

## 5.2 Resultados das Simulações Numéricas e do Pós-Processamento

Dos resultados numéricos após as simulações podem ser avaliados os campos de velocidade e pressão, além de verificar a ocorrência de vórtices de ponta de asa como descrito na literatura. (KATZ, 1995)

Além disso, para a comparação com Zerihan (2001) e validação dos resultados, deve-se analisar o processo de convergência dos valores dos coeficientes aerodinâmicos adimensionais ao longo das iterações de simulação e verificar se não há instabilidade numérica nos casos realizados.

### 5.2.1 Casos hx

#### 5.2.1.1 Convergência das Iterações

A convergência das iterações é importante para verificar se não foram obtidos valores errados para os coeficientes aerodinâmicos devido à instabilidade numérica do método iterativo.

Neste trabalho, foram usados como valores nominais dos coeficientes aerodinâmicos, os valores médios tomados com base nos valores para  $C_D$ ,  $C_L$  e  $C_{SF}$  do último quinto das iterações realizadas em cada simulação. Como foram realizadas 15000 iteração em cada caso, essa média é relativa aos últimos 3000 valores obtidos pelo SU2. Também foi determinado o desvio padrão dos valores utilizados para o cálculo da média para a

verificação da convergência e esses valores se encontram na tabela 5.3.

h	CD	CL
h17	4.61%	1.79%
h20	3.59%	1.04%
h22	4.29%	1.44%
h25	4.49%	1.56%
h27	3.29%	1.48%
h30	2.81%	1.49%
h32	2.16%	1.08%
h35	2.52%	1.07%
h37	2.42%	1.04%
h40	2.67%	1.26%
h45	5.35%	1.53%
h50	3.36%	1.14%
h60	4.24%	1.14%
h70	2.93%	1.38%
h80	2.66%	1.52%
h85	3.40%	1.75%
h90	3.87%	1.93%
h95	5.98%	2.87%
h100	5.58%	2.29%
h110	6.81%	3.21%
h120	7.01%	3.11%
h130	5.75%	2.85%
h150	7.30%	3.17%
h225	8.66%	4.56%

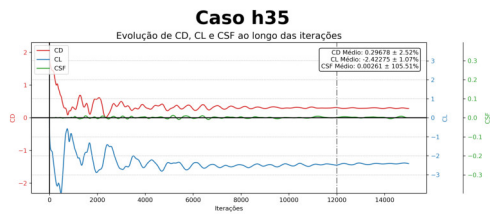
Tabela 5.3: Desvio padrão da convergência ao longo das iterações de cada caso

É importante observar mesmo os maiores desvios padrão de CD, CL são pequenos o suficiente para que possa se dizer que houve convergência dos valores obtidos.

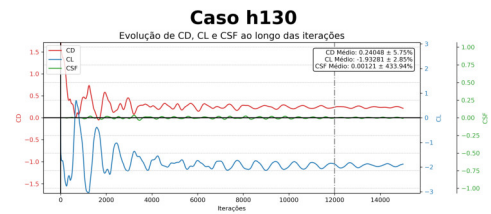
Além disso, apesar dos desvios padrão do coeficiente de força lateral serem muito grandes em todos os casos, isso pode ser explicado pelo fato que, como os casos são simétricos no plano xy, o coeficiente de força lateral esperado é nulo e a metodologia implícita de resolução das equações de Navier-Stokes empregada pelo SU2 flutua os resultados ao redor de zero. De fato, isso pode ser confirmado ao se observar o comportamento de convergência do CSF nas figuras 5.4a, 5.4b e nas presentes no apêndice B.

Dois exemplos de convergência, para um dos casos mais estáveis (h35) e um dos casos menos estáveis (h110) se encontram nas figuras 5.4a e 5.4b

Figuras relativas às convergências de todos os casos hx se encontram nos apêndices, em B. Gráficos de Convergência das Simulações



(a) Convergência do caso h35 para CD, CL e CSF ao longo das iterações



(b) Convergência do caso h130 para CD, CL e CSF ao longo das iterações

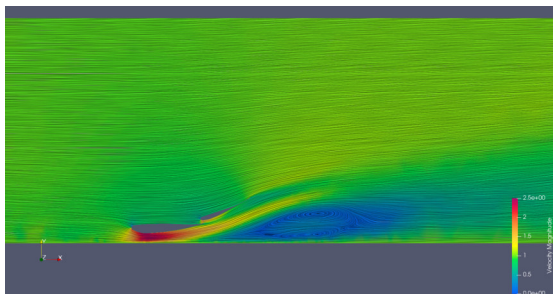
Figura 5.4: Comparação da convergência dos casos h35 e h130 para CD, CL e CSF

### 5.2.1.2 Campo de Velocidade

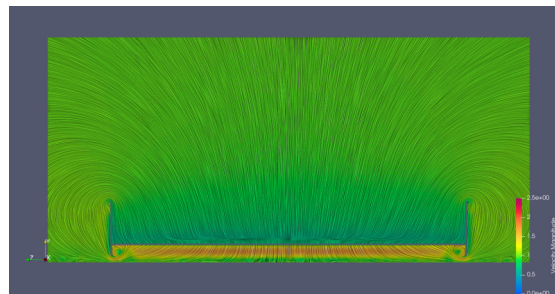
No campo de velocidade, é importante a visualização dos fenômenos de:

- *Stall* e recirculação do escoamento a jusante da asa;
- Aceleração do escoamento devido ao efeito solo;
- Escoamento quase não perturbado por efeito solo.

O fenômeno do *stall* do aerofólio e conseqüente recirculação do escoamento pode ser visto na figura 5.5, onde a baixa altura do aerofólio acaba ocasionando a constrição do escoamento. A bolha de recirculação é bem visível em azul a jusante da asa.



(a) Visualização de velocidade do escoamento no plano xy do caso h20



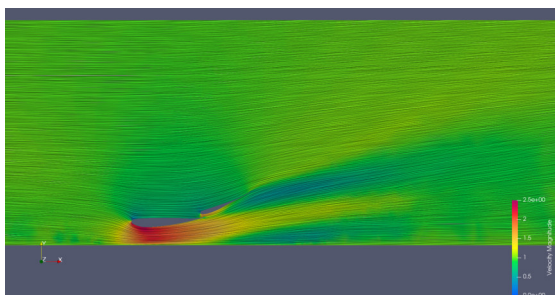
(b) Visualização de velocidade do escoamento no plano yz do caso h20

Figura 5.5: Visualização de velocidade do escoamento do caso h20 no plano xy e yz

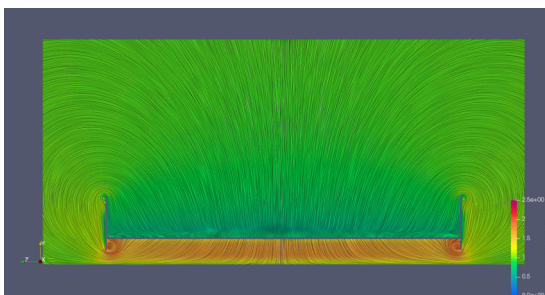
Nas figuras 5.6 e 5.7 tem-se, respectivamente, uma situação em que o efeito solo acelera o escoamento de maneira eficiente e ocasiona um aumento do *downforce* gerado e uma situação onde a altura do aerofólio é grande demais e acaba por não aproveitar do efeito solo para geração de *downforce* da melhor maneira.

Figuras relativas às visualizações de velocidade de todos os casos hx se encontram nos apêndices, em D. Visualizações de Velocidade - Plano xy e E. Visualizações de Velocidade - Plano yz - Visão Geral.



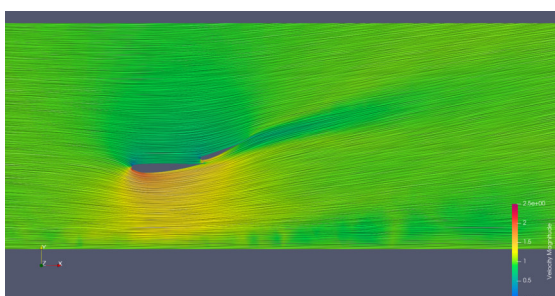


(a) Visualização de velocidade do escoamento no plano xy do caso h45

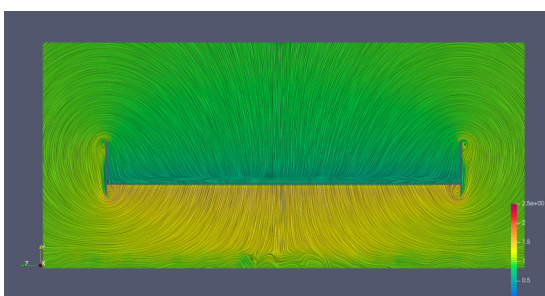


(b) Visualização de velocidade do escoamento no plano yz do caso h45

Figura 5.6: Visualização de velocidade do escoamento do caso h45 no plano xy e yz



(a) Visualização de velocidade do escoamento no plano xy do caso h225

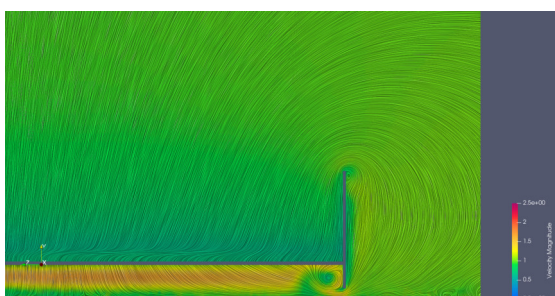


(b) Visualização de velocidade do escoamento no plano yz do caso h225

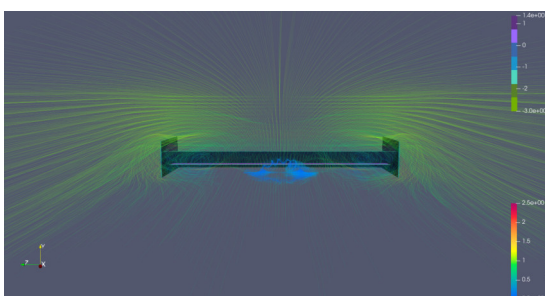
Figura 5.7: Visualização de velocidade do escoamento do caso h225 no plano xy e yz

### 5.2.1.3 Vórtices de Ponta de Asa

A ocorrência de formação dos vórtices de ponta de asa foi observada em todas as simulações, como o esperado. Um exemplo pode ser visto com o caso h17, na figura 5.8.



(a) Detalhe da visualização de velocidade do escoamento no plano yz mostrando os vórtices de ponta de asa do caso h17



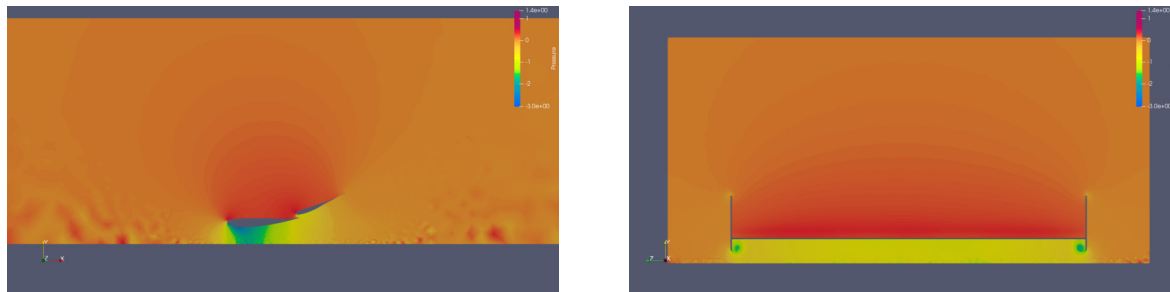
(b) Visualização 3D do escoamento mostrando os vórtices de ponta de asa do caso h17

Figura 5.8: Visualização da formação de vórtices de ponta de asa no caso h17

Figuras relativas às visualizações dos vórtices de ponta de asa de todos os casos hx se encontram nos apêndices, em F. Visualizações de Velocidade - Plano yz - Detalhe dos Vórtices de Ponta de Asa e G. Visualizações 3D das linhas de corrente.

### 5.2.1.4 Campo de Pressão

A principal visualização do escoamento relativa ao campo de pressão é para a visualização das zonas de alta e baixa pressão formadas devido ao escoamento ao redor da asa e sua interação com o túnel de vento.



(a) Visualização de pressão do escoamento no plano xy do caso h40

(b) Visualização de pressão do escoamento no plano yz do caso h40

Figura 5.9: Visualização de pressão do escoamento do caso h40 no plano xy e yz

Pode-se ver na figura 5.9 que as zonas se formaram conforme o esperado, ocorrendo inclusive a captura da zona de baixa pressão no centro dos vórtices de ponta de asa.

Figuras relativas às visualizações pressão de todos os casos hx se encontram nos apêndices, em H. Visualizações de Pressão - Plano xy e I. Visualizações de Pressão - Plano yz.

## 5.2.2 Validação dos Resultados

Para a validação dos resultados, deve-se compará-los com os resultados obtidos por Zerihan (2001) em sua tese de doutorado. Versões maiores das figuras 5.10 e 5.11 estão disponíveis no apêndice C. Comparativo dos Resultados de Simulação com os Resultados Experimentais de Zerihan. Assim como a tabela dos valores de cada gráfico.

É importante notar que as simulações que possuem bolha de recirculação a jusante do escoamento, como é o caso das simulações h17 até h45, introduzem uma dificuldade significativa em capturar com acurácia os coeficientes aerodinâmicos adimensionais devido aos efeitos da modelagem de turbulência utilizada. Portanto, separou-se, nos gráficos apresentados, os pontos antes e depois do desaparecimento da bolha de recirculação e serão levados em conta para a análise quantitativa proposta apenas os pontos após o desaparecimento da bolha.

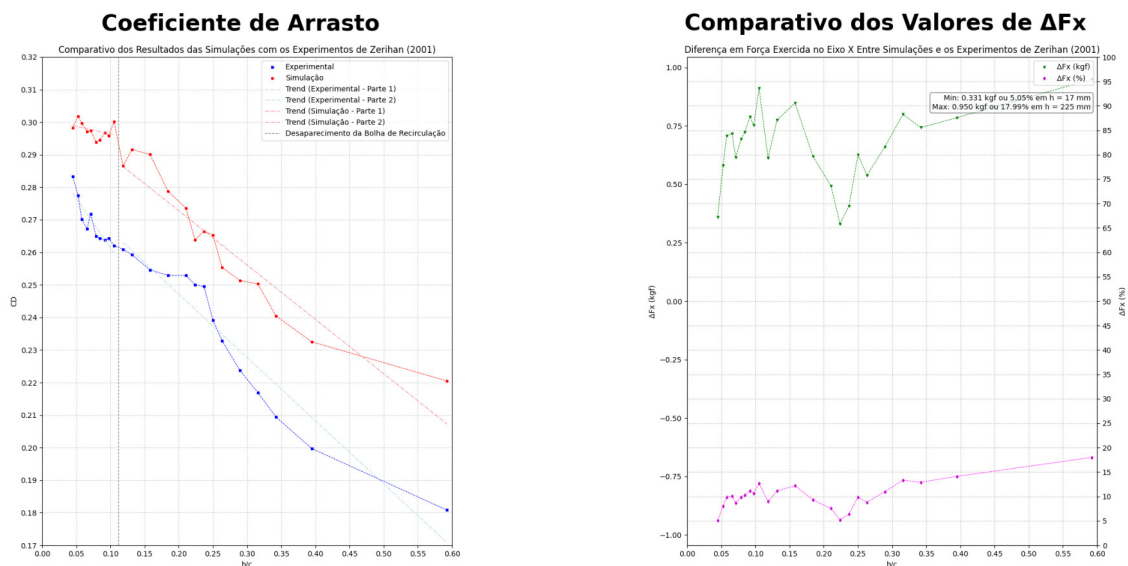
### 5.2.2.1 Coeficiente de Arrasto

Na figura 5.10a tem-se os resultados de simulação junto com os dados obtidos por Zerihan (2001) para o coeficiente de arrasto. Pode-se ver que, apesar do arrasto das simulações ter sido sempre maior do que o arrasto medido experimentalmente, a tendência do arrasto em função da altura de asa foi bem capturada.

Isso pode ser medido ao se comparar os valores dos coeficientes lineares das retas de tendência da segunda parte dos dados experimentais e numéricos ( $m_{CD;exp}$  e  $m_{CD;sim}$ ), dispostos na tabela 5.4.

Coeficiente	Valor
$m_{CD;exp}$	-0.194550
$m_{CD;sim}$	-0.163843

Tabela 5.4: Coeficientes Lineares das Retas de Tendência do Arrasto



(a) Comparativo do Coeficiente de Arrasto

(b) Diferença entre os resultados obtidos nas simulações e os resultados de Zerihan (2001) dimensionalizados para kgf e em formato percentual

Figura 5.10: Validação do coeficiente de arrasto

Na figura 5.10b, a diferença entre os pontos numéricos e os experimentais foi medida para cada simulação a fim de verificar a maior e menor discrepância entre os experimentos. Em termos percentuais, a maior disparidade foi para o caso h225 com 17.99%. Dimensionalizando isso para  $kgf$ , tem-se uma noção melhor da ordem de grandeza do erro, com  $0.950\ kgf$  de diferença máxima. Já a menor discrepância foi para o caso h17 com 5.05% ou  $0.331\ kgf$ .



Se a diferença absoluta entre os coeficientes lineares das retas de tendência for transformada em um formato percentual relativo ao maior coeficiente linear dentre as tendências, como exposto em (5.1) e (5.2), pode-se criar um coeficiente de grau de similaridade percentual ( $gs_{CD}$ ) adimensionalizado que quantifica o paralelismo das retas de tendência dos dados. E também cria-se um valor complementar de erro de similaridade percentual adimensional ( $\Delta m_{CD}$ ).

$$\Delta m_{CD} = \frac{|m_{CD;exp} - m_{CD;sim}|}{|MAX(m_{CD;exp}; m_{CD;sim})|} = 15.78\% \quad (5.1)$$

$$gs_{CD} = 1 - \Delta m_{CD} = 84.22\% \quad (5.2)$$

Ou seja, para o arrasto, a tendência dos dados obtidos após o desaparecimento da bolha de recirculação foi capturada com 84.22% de precisão.

### 5.2.2.2 Coeficiente de Sustentação

Na figura 5.11a tem-se os resultados de simulação obtidos junto com os valores obtidos por Zerihan (2001) para o coeficiente de sustentação. De modo inverso ao arrasto, pode-se observar que o coeficiente de sustentação obtido numericamente foi sempre menor que sua contrapartida experimental, mas assim como o coeficiente de arrasto, a tendência da curva *altura x downforce* foi bem capturada.

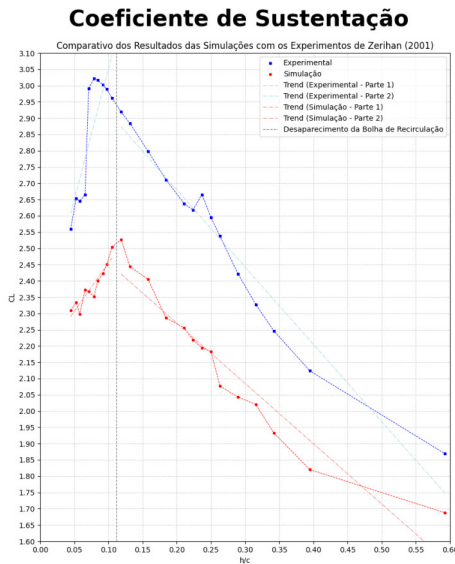
Assim como feito para o arrasto, isso pode ser medido ao se comparar os valores dos coeficientes lineares das retas de tendência da segunda parte dos dados experimentais e numéricos ( $m_{CL;exp}$  e  $m_{CL;sim}$ ), dispostos na tabela 5.5.

Coeficiente	Valor
$m_{CL;exp}$	-2.375747
$m_{CL;sim}$	-1.849632

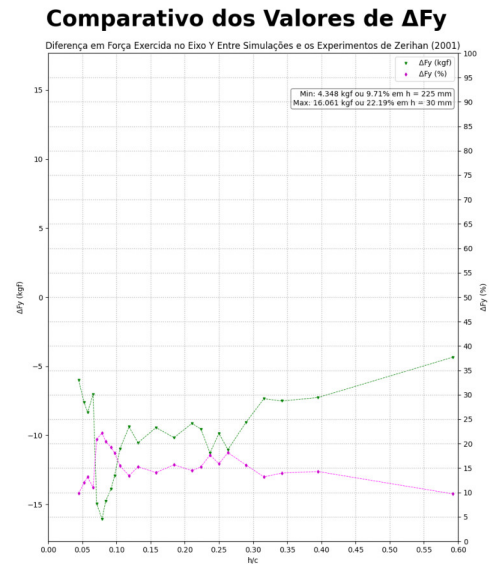
Tabela 5.5: Coeficientes Lineares das Retas de Tendência da Sustentação

Do mesmo jeito que para o coeficiente de arrasto, na figura 5.11b, a diferença entre os pontos numéricos e os experimentais foi medida para cada simulação. A maior disparidade foi para o caso h30 com 22.19% ou 16.061 *kgf* de diferença máxima. Já a menor discrepância foi para o caso h30 com 9.71% ou 4.348 *kgf*.

Fazendo o mesmo procedimento que foi feito para o arrasto, pode se calcular os coeficientes de erro de similaridade ( $\Delta m_{CL}$ ) e de grau de similaridade percentual das tendências da sustentação ( $gs_{CL}$ ) nas equações (5.3) e (5.4).



(a) Comparativo do Coeficiente de Sustentação



(b) Diferença entre os resultados obtidos nas simulações e os resultados de Zerihan (2001) dimensionalizados para kgf e em formato percentual

Figura 5.11: Validação do coeficiente de sustentação

$$\Delta m_{CL} = \frac{|m_{CL;exp} - m_{CL;sim}|}{|MAX(m_{CL;exp}; m_{CL;sim})|} = 22.15\% \quad (5.3)$$

$$gs_{CL} = 1 - \Delta m_{CL} = 77.85\% \quad (5.4)$$

Ou seja, para a sustentação, a tendência dos dados obtidos após o desaparecimento da bolha de recirculação foi capturada com 77.96% de precisão.

### 5.2.2.3 Gráfico CL/CD

Também pode-se utilizar um gráfico CL/CD para comparar os dados obtidos com os dados fornecidos por Zerihan (2001).

Ao analisar a figura 5.12, pode-se ver que o comportamento aerodinâmico da asa se assemelha com em ambos os casos, com uma dificuldade de captura do ponto de maior razão entre os coeficientes e uma anomalia no comportamento após o desaparecimento da bolha de recirculação, que pode estar atrelada à problemas relativos ao modelo de turbulência utilizado.

Apesar disso, observa-se uma tendência geral de diminuição da eficácia aerodinâmica, medida por  $\frac{CL}{CD}$ , proporcional ao aumento da altura da asa com relação ao chão adimensionalizada pela corda  $\frac{h}{c}$ , após o desaparecimento da recirculação por constrição

do escoamento, como antecipado pela teoria analítica e demonstrado nos experimentos de Zerihan (2001).

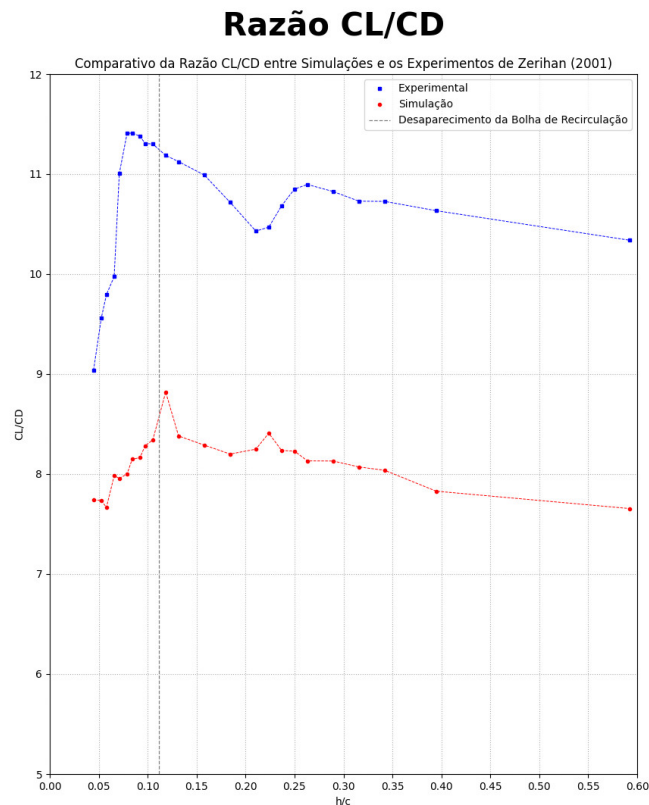


Figura 5.12: Comparativo da razão entre os coeficientes adimensionais.

### 5.2.3 Caso hinf

Com o ciclo de simulações validado, pode-se utilizar o processo desenvolvido para estudar casos em que não houveram testes experimentais. O caso escolhido foi o hinf, onde a asa está no infinito.

#### 5.2.3.1 Convergência das Iterações e Coeficientes Aerodinâmicos Adimensionais

Pela convergência das iterações de simulação, pode-se observar que não há instabilidade numérica notável que impacte na qualidade dos resultados obtidos, como visto na tabela 5.6.

Assim, pode-se dizer com certo nível de confiabilidade que os coeficientes aerodinâmicos obtidos estão condizentes com a realidade. Estes coeficientes podem ser vistos na figura 5.13 e também estão dispostos na tabela 5.7.

Caso	CD	CL	CSF
hinf	3.06%	1.30%	333.75%

Tabela 5.6: Desvio padrão dos coeficientes aerodinâmicos adimensionais ao longo das iterações do caso hinf

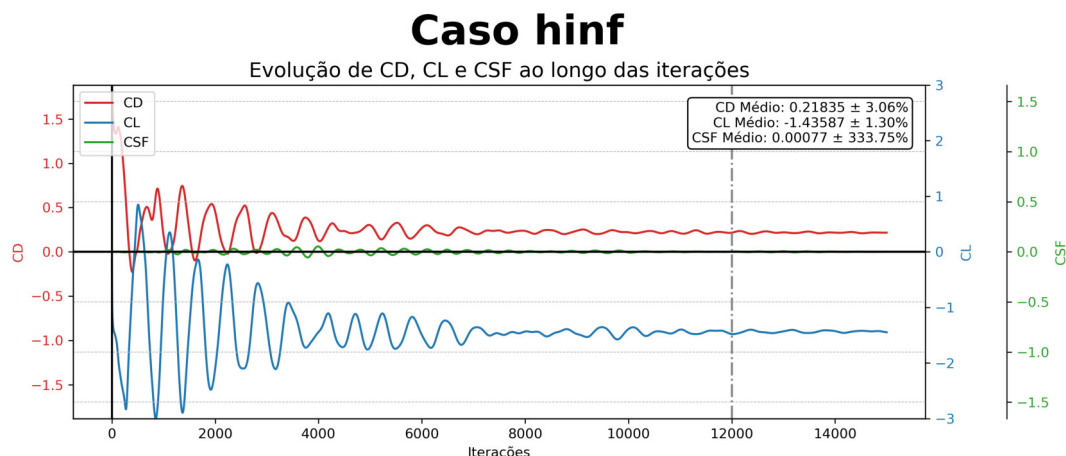


Figura 5.13: Convergência do caso hinf para CD, CL e CSF ao longo das iterações

Caso	CD	CL	CSF
hinf	0.21835	-1.43587	0.00077

Tabela 5.7: Coeficientes aerodinâmicos adimensionais médios do caso hinf

É interessante notar que estes valores seguem a tendência dos gráficos das figuras 5.10a e 5.11a, com uma queda nos coeficientes de arrasto e sustentação. Isso indica que, mesmo a 225 milímetros do solo, ainda há a influência da constrição do escoamento na aerodinâmica do corpo, apesar de ser pouca.

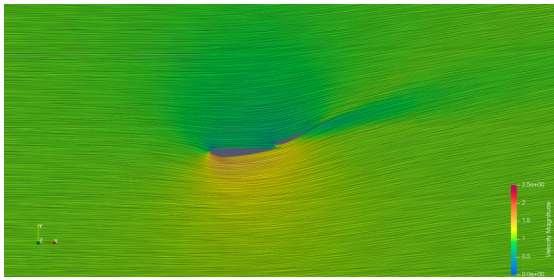
### 5.2.3.2 Campo de Velocidade

No campo de velocidade, como pode ser observado na figura 5.14, o escoamento se assemelha com o caso h225, onde o escoamento é bem-comportado ao redor da asa, sem a ocorrência de *stall*, e não há perturbação devido a efeitos de instalação.

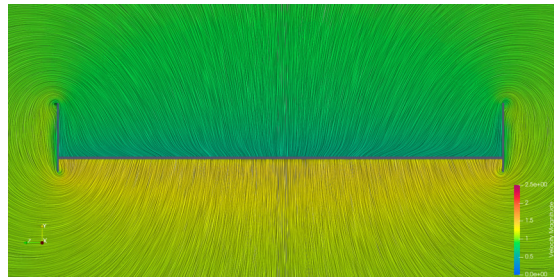
### 5.2.3.3 Vórtices de Ponta de Asa e Campo de Pressão

Assim como nos casos dentro do túnel de vento, há a formação de vórtices de ponta de asa nas bordas das *endplates* do conjunto. Isso pode ser visualizado nas imagens 5.15a e 5.15b.

Já no campo de pressão, visível na figura 5.16, é relativamente fácil de observar a influência do túnel de vento e do solo no comportamento do escoamento, sem a geometria limitante das paredes, teto e chão do túnel de vento, o aerofólio gera zonas

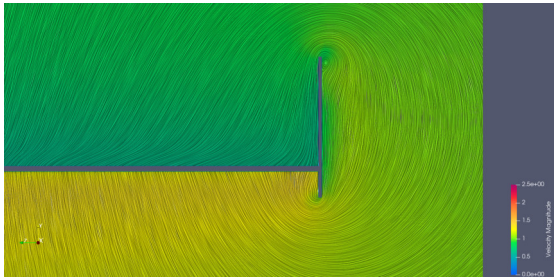


(a) Visualização de velocidade do escoamento no plano xy do caso hinf

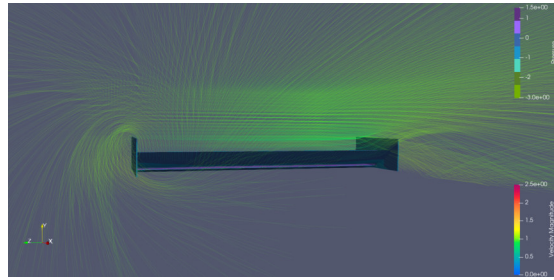


(b) Visualização de velocidade do escoamento no plano yz do caso hinf

Figura 5.14: Visualização de velocidade do escoamento do caso hinf no plano xy e yz



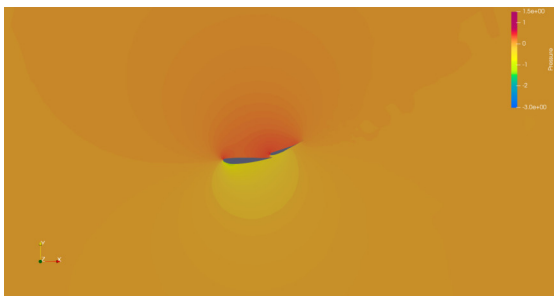
(a) Detalhe da visualização de velocidade do escoamento no plano yz mostrando os vórtices de ponta de asa do caso hinf



(b) Visualização 3D do escoamento mostrando os vórtices de ponta de asa do caso hinf

Figura 5.15: Visualizações do caso hinf: detalhe da velocidade no plano yz e visualização 3D do escoamento com vórtices de ponta de asa

bem definidas de alta e baixa pressão e é nítido o tamanho do gradiente de pressão em ambas.



(a) Visualização de pressão do escoamento no plano xy do caso hinf



(b) Visualização de pressão do escoamento no plano yz do caso hinf

Figura 5.16: Visualizações de pressão do caso hinf: plano xy e plano yz

# Capítulo 6

## Conclusão

Com base nos resultados apresentados nesta seção, pode-se concluir que o pré-processamento gerou malhas de simulação adequadas para os casos estudados. Houve pouca variação no número de elementos de malha entre os casos, o que sugere que os erros de discretização são da mesma ordem de grandeza. Isso permite comparar os resultados numéricos de forma consistente entre os casos.

As simulações mostraram resultados satisfatórios em relação à convergência dos coeficientes aerodinâmicos adimensionais ao longo das iterações. E, dada a estabilidade numérica das simulações, foram utilizados os valores médios dos últimos 3000 valores para cálculo dos coeficientes aerodinâmicos.

A visualização dos campos de velocidade e pressão revelou fenômenos importantes, como o *stall* do aerofólio em efeito solo, formação de vórtices de ponta de asa e aceleração do escoamento devido ao efeito solo. As zonas de alta e baixa pressão também foram identificadas conforme esperado.

Para validar os resultados, foram comparados os coeficientes de arrasto e sustentação obtidos nas simulações com os dados experimentais de Zerihan (2001). Embora os valores numéricos para o arrasto tenham sido ligeiramente maiores e os valores numéricos para a sustentação, ou no caso, *downforce*, ligeiramente menores do que os experimentais, as tendências gerais foram consistentes, com variação de até 17.99% para o coeficiente de arrasto e até 22.19% para o coeficiente de sustentação.

Em suma, os resultados numéricos obtidos forneceram uma boa representação do comportamento aerodinâmico dos casos estudados, com resultados próximos aos experimentais. Esses resultados contribuem para o entendimento e a validação do método de simulação empregado e podem ser usados para análises mais aprofundadas do desempenho aerodinâmico da asa estudada e dos efeitos de instalação em túneis de vento.

## 6.1 Sugestões para Trabalhos Futuros

Com base nos resultados e nas observações feitas durante este estudo, surgem várias sugestões para trabalhos futuros que podem expandir e aprimorar ainda mais o conhecimento sobre o comportamento aerodinâmico da asa estudada. Algumas sugestões incluem:

1. Investigar o efeito de diferentes geometrias de asa nos coeficientes aerodinâmicos, explorando variações nas dimensões, perfis e configurações estruturais da asa. Isso pode ajudar a identificar formas mais eficientes e otimizar o desempenho aerodinâmico.
2. Realizar estudos em diferentes condições de operação, como ângulos de ataque variados, velocidades do fluxo de ar e altitudes. Isso permitiria analisar o comportamento da asa em uma variedade de cenários e melhor compreender sua resposta aerodinâmica em diferentes situações.
3. Simular um modelo de asa frontal de fórmula 1 de uma temporada mais recente. Isso pode ajudar a observar a evolução do conhecimento sobre aerodinâmica e do *design* desses elementos ao longo do tempo.

Essas sugestões para trabalhos futuros têm o objetivo de expandir os conhecimentos sobre o desempenho aerodinâmico da asa estudada, bem como explorar outras áreas de pesquisa relacionadas à aerodinâmica e CFD.

# Referências Bibliográficas

ANDERSON, D. A.; PLETCHER, R. H.; TANNEHILL, J. C. *Computational fluid mechanics and heat transfer*. Third edition. Boca Raton: CRC Press, Taylor & Francis Group, 2013. (Series in computational and physical processes in mechanics and thermal sciences). ISBN 978-1-59169-037-5.

ANDERSON, J. D. *Fundamentals of aerodynamics*. Sixth edition. New York, NY: McGraw Hill Education, 2017. (McGraw-Hill series in aeronautical and aerospace engineering). ISBN 978-1-259-12991-9.

DYKE, M. V. (Ed.). *An album of fluid motion*. 12. printing. ed. Stanford, Calif: Parabolic Press, 2008. ISBN 978-0-915760-02-2 978-0-915760-03-9.

ECONOMON, T. D. et al. SU2: An Open-Source Suite for Multiphysics Simulation and Design. *AIAA Journal*, dez. 2015. Publisher: American Institute of Aeronautics and Astronautics. Disponível em: <<https://arc.aiaa.org/doi/10.2514/1.J053813>>.

FERZIGER, J. H.; PERIĆ, M.; STREET, R. L. *Computational methods for fluid dynamics*. Fourth edition. Cham: Springer, 2020. ISBN 978-3-319-99691-2.

FIA. *Regulamentação Técnica FIA F1 2023*. 2022.

GEUZAINÉ, C. *Gmsh 4.11.1*. 2022. Disponível em: <<https://gmsh.info/doc/texinfo/gmsh.html>>.

HANSEN, C. D.; JOHNSON, C. R. (Ed.). *The visualization handbook*. Amsterdam ; Boston: Elsevier-Butterworth Heinemann, 2005. ISBN 978-0-12-387582-2.

HOUGHTON, E. L.; CARPENTER, P. W.; COLLICOTT, S. H.; VALENTINE, D. T. *Aerodynamics for engineering students*. Seventh edition. Oxford: Butterworth-Heinemann, is an imprint of Elsevier, 2017. OCLC: ocn948548390. ISBN 978-0-08-100194-3.

KATZ, J. *Race car aerodynamics: designing for speed*. Cambridge, MA, USA: R. Bentley, 1995. ISBN 978-0-8376-0142-7.

MALISKA, C. R. *FUNDAMENTALS OF COMPUTATIONAL FLUID DYNAMICS the finite volume method*. S.l.: SPRINGER INTERNATIONAL PU, 2023. OCLC: 1364316604. ISBN 978-3-031-18235-8.

MENTER, F. R. Two-equation eddy-viscosity turbulence models for engineering applications. *AIAA Journal*, v. 32, n. 8, p. 1598–1605, ago. 1994. ISSN 0001-1452, 1533-385X. Disponível em: <<https://arc.aiaa.org/doi/10.2514/3.12149>>.



- MENTER, F. R.; KUNTZ, M.; LANGTRY, R. Ten Years of Industrial Experience with the SST Turbulence Model. *Heat and Mass Transfer*, 2003.
- MUNSON, B. R.; YOUNG, D. F.; OKIISHI, T. H. *Fundamentals of fluid mechanics*. 5th ed. ed. Hoboken, NJ: J. Wiley & Sons, 2006. ISBN 978-0-471-67582-2.
- NETO, A. *Escoamentos Turbulentos: Análise Física e Modelagem Teórica*. [S.l.]: Composer, 2020. ISBN 6599036511.
- POPE, S. B. *Turbulent flows*. Cambridge ; New York: Cambridge University Press, 2000. ISBN 978-0-521-59125-6 978-0-521-59886-6.
- SALOME. *History of SALOME*. 2022. Disponível em: <[https://www.salome-platform.org/?page\\_id=676](https://www.salome-platform.org/?page_id=676)>.
- SALOME. *Meshing with SMESH*. 2022. Disponível em: <[https://www.salome-platform.org/?page\\_id=374](https://www.salome-platform.org/?page_id=374)>.
- SCHOEBERL, J. NETGEN An advancing front 2D/3D-mesh generator based on abstract rules. *Computing and Visualization in Science*, v. 1, p. 41–52, jul. 1997.
- SMITS, A. J.; LIM, T. T. (Ed.). *Flow visualization: techniques and examples*. 2nd ed. ed. London : Singapore ; Hackensack, NJ: Imperial College Press ; Distributed by World Scientific Press, 2012. OCLC: ocn792748131. ISBN 978-1-84816-791-9.
- STEWART, J. *Calculus*. Eighth edition. Boston, MA, USA: Cengage Learning, 2016. OCLC: ocn892432745. ISBN 978-1-285-74062-1 978-1-305-27176-0.
- TENNEKES, H.; LUMLEY, J. L. *A first course in turbulence*. Cambridge, Mass: MIT Press, 1972. ISBN 978-0-262-20019-6.
- TREMAYNE, D. *The science of Formula 1 design: expert analysis of the anatomy of the modern Grand Prix car*. 3rd ed. ed. Sparkford, NR Yeovil, Somerset, U.K. : Newbury Park, Calif: Haynes Pub. ; Haynes North America, 2009. ISBN 978-1-84425-718-8.
- WHITE, F. M.; CHUL, R. Y. *Fluid mechanics: Frank M. White ; adapted by Professor Rhim Yoon Chul*. Vuitena edició a si units. Chennai: McGraw Hill Education, 2018. OCLC: 1293962210. ISBN 978-93-85965-49-4.
- ZERIHAN, J. D. C. *An Investigation into the Aerodynamics of Wings in Ground Effect*. Tese (Doutorado) — University of Southampton, 2001.
- ÇENGEL, Y. A.; CIMBALA, J. M. *Fluid mechanics: fundamentals and applications*. Third edition. New York: McGraw Hill, 2014. ISBN 978-0-07-338032-2.

# Anexos

# ANEXO I

## Dados de coordenadas dos pontos dos perfis de asa

Flap Wing / Corda			
Corda ( $c$ )		380 mm	
Ângulo de Ataque ( $\alpha$ )		14.1 deg	
Superfície de Sucção		Superfície de Pressão	
$x/c$	$y/c$	$x/c$	$y/c$
0.5643	0.0757	0.5643	0.0757
0.5687	0.0633	0.5720	0.0863
0.5744	0.0601	0.5785	0.0886
0.5805	0.0597	0.5849	0.0908
0.5868	0.0605	0.5914	0.0929
0.5931	0.0614	0.5979	0.0951
0.6248	0.0678	0.6302	0.1059
0.6570	0.0779	0.6625	0.1167
0.6896	0.0908	0.6948	0.1275
0.7240	0.1065	0.7271	0.1383
0.7573	0.1249	0.7594	0.1491
0.7895	0.1434	0.7918	0.1600
0.8229	0.1617	0.8241	0.1709
0.8566	0.1789	0.8567	0.1836
0.8895	0.1967	0.8897	0.1996
0.9230	0.2158	0.9232	0.2186
0.9564	0.2345	0.9567	0.2372
0.9669	0.2402	0.9670	0.2429

Tabela I.1: Dados coletados para a posição adimensionalizada de pontos no perfil de asa da Flap Wing. Fonte: (ZERIHAN, 2001)

Main Wing / Corda			
Corda ( $c$ )		380 mm	
Ângulo de Ataque ( $\alpha$ )		14.1 deg	
Superfície de Sucção		Superfície de Pressão	
$x/c$	$y/c$	$x/c$	$y/c$
0.0000	0.0000	0.0000	0.0000
0.0006	-0.0044	0.0006	0.0047
0.0011	-0.0063	0.0012	0.0064
0.0029	-0.0098	0.0030	0.0102
0.0058	-0.0134	0.0059	0.0136
0.0087	-0.0156	0.0089	0.0159
0.0117	-0.0173	0.0118	0.0176
0.0146	-0.0188	0.0148	0.0184
0.0175	-0.0203	0.0177	0.0189
0.0205	-0.0217	0.0207	0.0194
0.0234	-0.0231	0.0236	0.0199
0.0263	-0.0245	0.0265	0.0204
0.0293	-0.0258	0.0295	0.0208
0.0322	-0.0270	0.0324	0.0212
0.0351	-0.0283	0.0354	0.0217
0.0410	-0.0306	0.0412	0.0225
0.0469	-0.0328	0.0471	0.0232
0.0528	-0.0348	0.0530	0.0239
0.0586	-0.0366	0.0589	0.0245
0.0704	-0.0398	0.0707	0.0256
0.0821	-0.0422	0.0824	0.0265
0.0939	-0.0441	0.0942	0.0272
0.1056	-0.0452	0.1059	0.0278
0.1174	-0.0457	0.1177	0.0282
0.1468	-0.0448	0.1471	0.0293
0.1762	-0.0430	0.1765	0.0303
0.2056	-0.0407	0.2059	0.0310
0.2350	-0.0379	0.2353	0.0314
0.2644	-0.0347	0.2647	0.0316
0.2938	-0.0309	0.2941	0.0315
0.3232	-0.0267	0.3234	0.0311
0.3526	-0.0219	0.3528	0.0305
0.3820	-0.0167	0.3822	0.0295
0.4114	-0.0110	0.4116	0.0283

Continua na próxima página

**Tabela I.2 – continuação da última página**

$x/c$	$y/c$	$x/c$	$y/c$
0.4409	-0.0049	0.4410	0.0268
0.4703	0.0018	0.4704	0.0257
0.4997	0.0090	0.4998	0.0261
0.5291	0.0166	0.5292	0.0282
0.5409	0.0198	0.5409	0.0295
0.5527	0.0230	0.5527	0.0311
0.5644	0.0264	0.5645	0.0330
0.5762	0.0298	0.5762	0.0352
0.5821	0.0316	0.5821	0.0364
0.5880	0.0333	0.5880	0.0377

Tabela I.2: Dados coletados para a posição adimensionalizada de pontos no perfil de asa da Main Wing. Fonte: (ZERIHAN, 2001)

## ANEXO II

### Resultados Experimentais de Zerihan para CD e CL

h	h/c	CL - Zerihan	CD - Zerihan
17.00000	0.04527	2.56022	0.28326
20.00000	0.05358	2.65305	0.27754
22.00000	0.05912	2.64552	0.27010
25.00000	0.06651	2.66559	0.26724
27.00000	0.07206	2.99176	0.27182
30.00000	0.07945	3.02186	0.26495
32.00000	0.08499	3.01685	0.26438
35.00000	0.09330	3.00179	0.26381
37.00000	0.09792	2.98925	0.26438
40.00000	0.10624	2.96165	0.26209
45.00000	0.11917	2.91900	0.26094
50.00000	0.13210	2.88387	0.25923
60.00000	0.15889	2.79857	0.25465
70.00000	0.18568	2.71075	0.25293
80.00000	0.21155	2.63799	0.25293
85.00000	0.22448	2.61792	0.25007
90.00000	0.23741	2.66559	0.24950
95.00000	0.25035	2.59534	0.23920
100.00000	0.26420	2.53763	0.23290
110.00000	0.29007	2.42222	0.22375
120.00000	0.31594	2.32688	0.21688
130.00000	0.34273	2.24659	0.20944
150.00000	0.39538	2.12366	0.19971
225.00000	0.59222	1.86954	0.18083

Tabela II.1: Resultados dos coeficientes adimensionais aerodinâmicos CD e CL obtidos experimentalmente. FONTE:(ZERIHAN, 2001), retirados utilizando o programa *xyscan*

# Apêndices

# APÊNDICE A

## Visualizações das Malhas Geradas

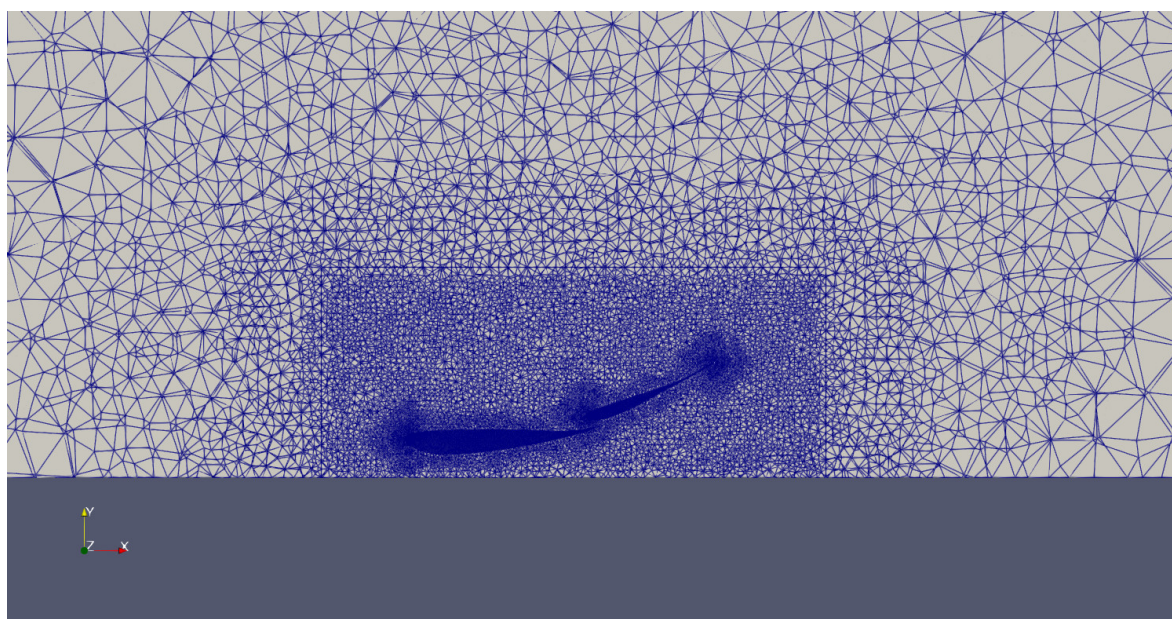


Figura A.1: Malha do caso h17



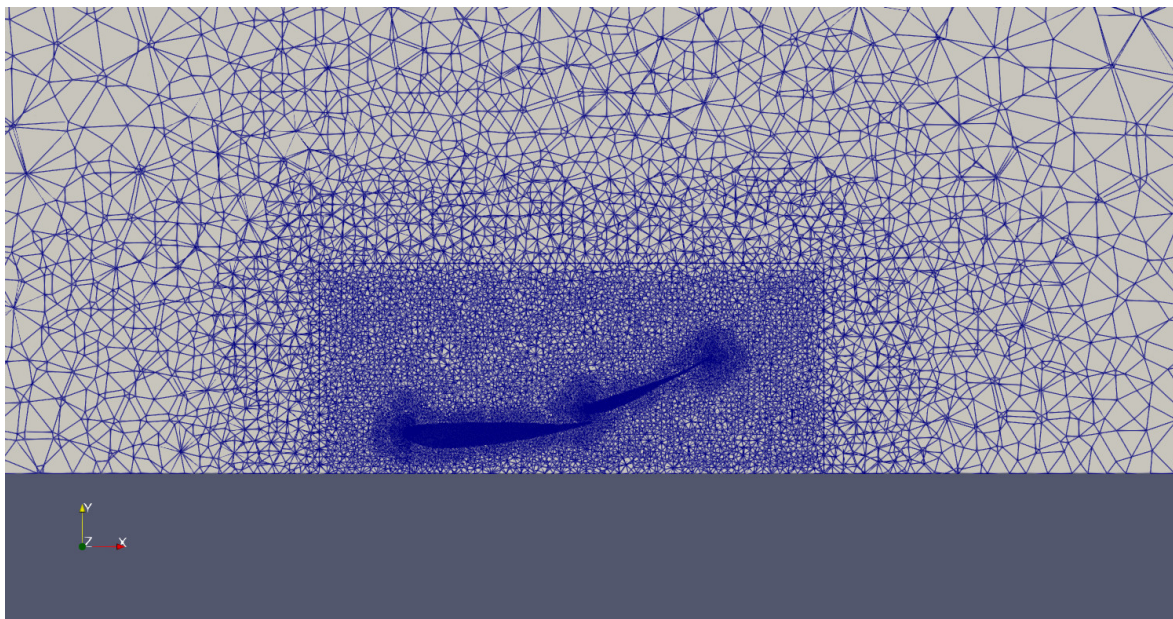


Figura A.2: Malha do caso h20

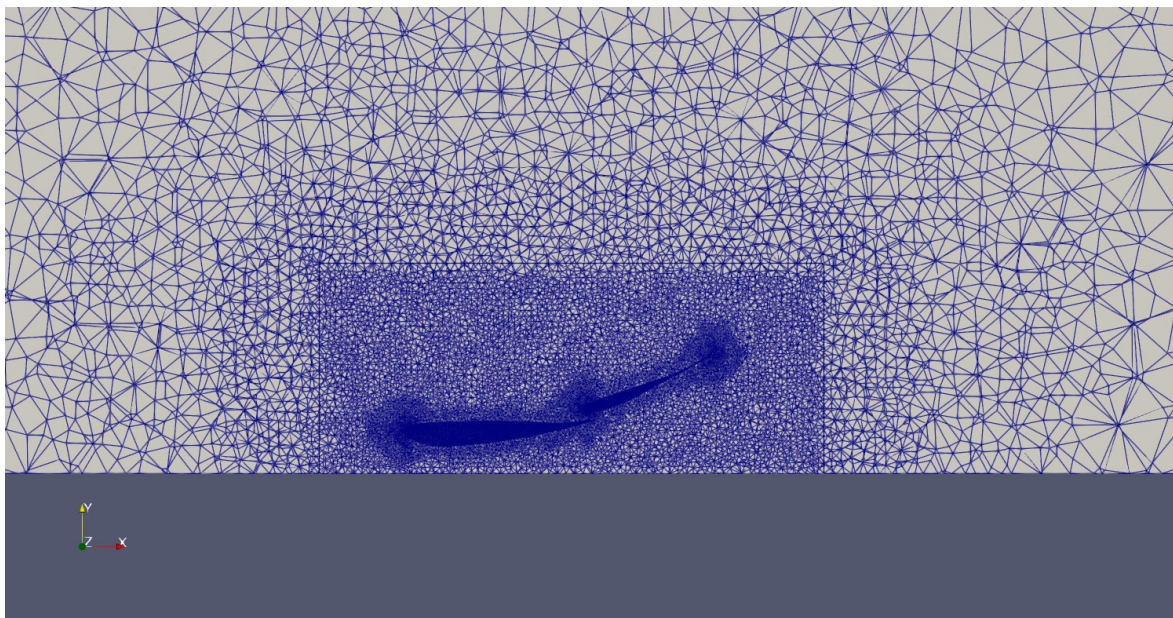


Figura A.3: Malha do caso h22



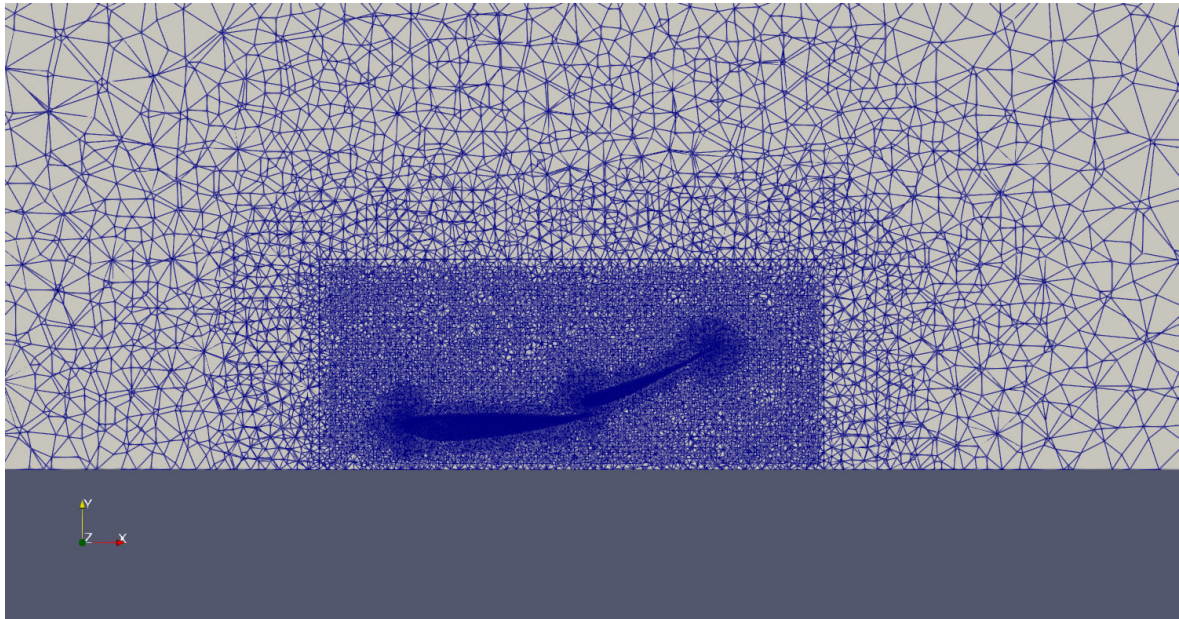


Figura A.4: Malha do caso h25

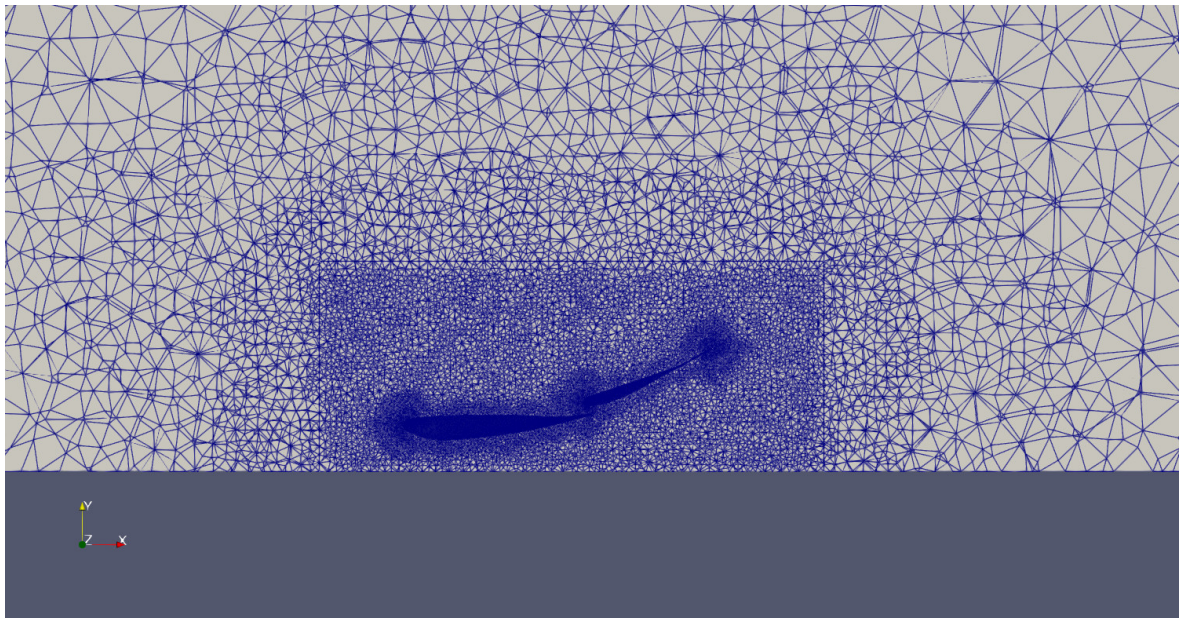


Figura A.5: Malha do caso h27



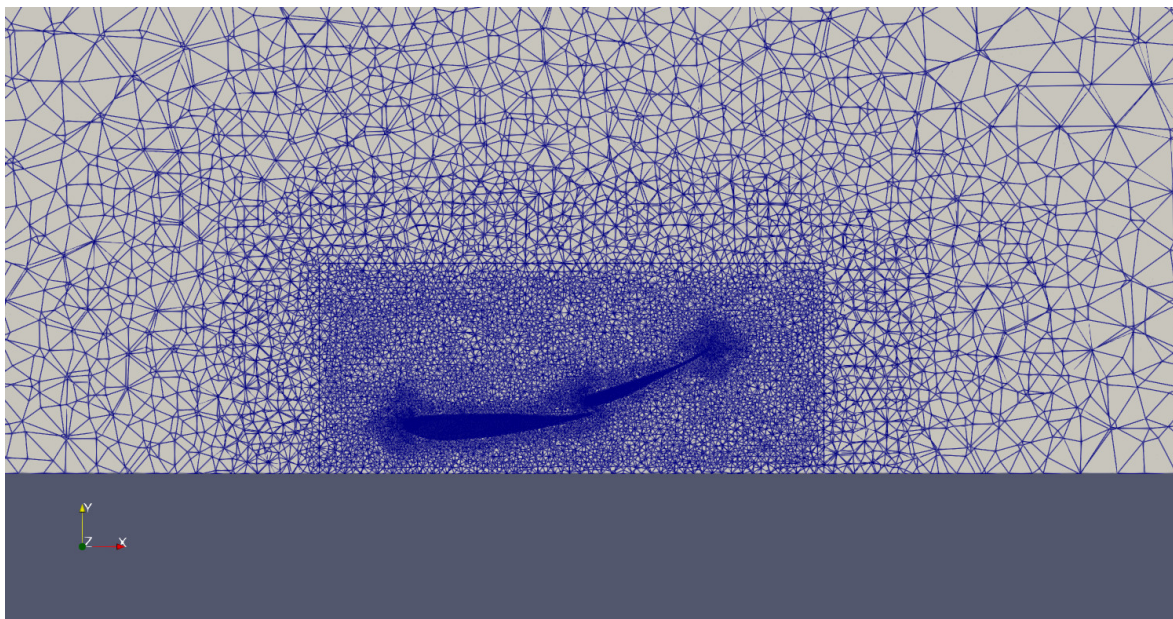


Figura A.6: Malha do caso h30

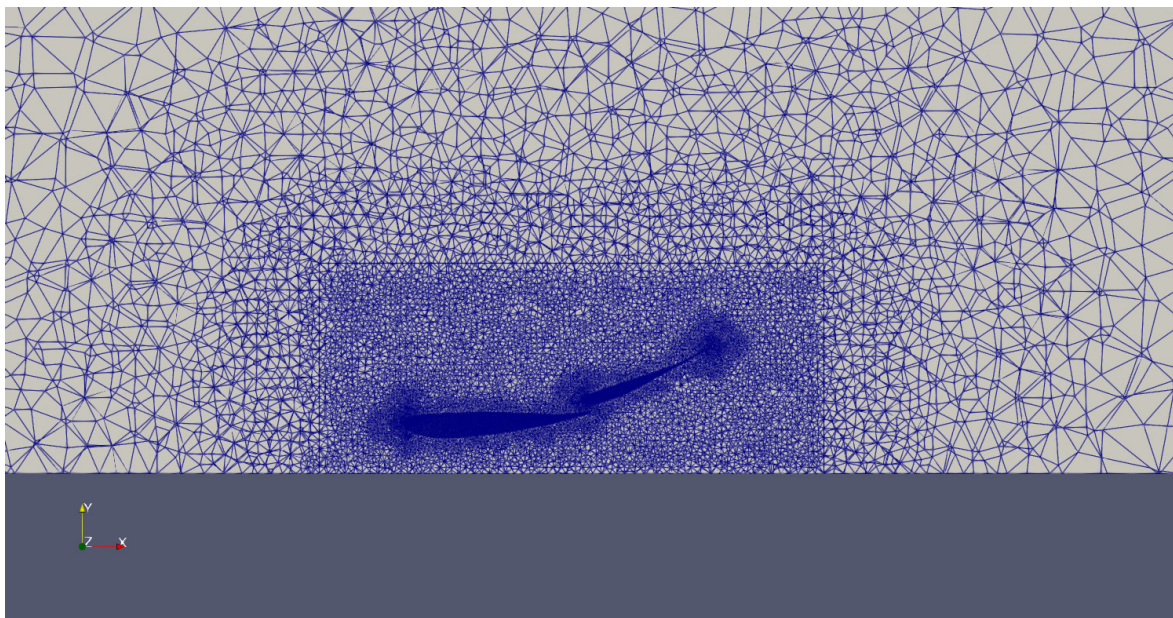


Figura A.7: Malha do caso h32



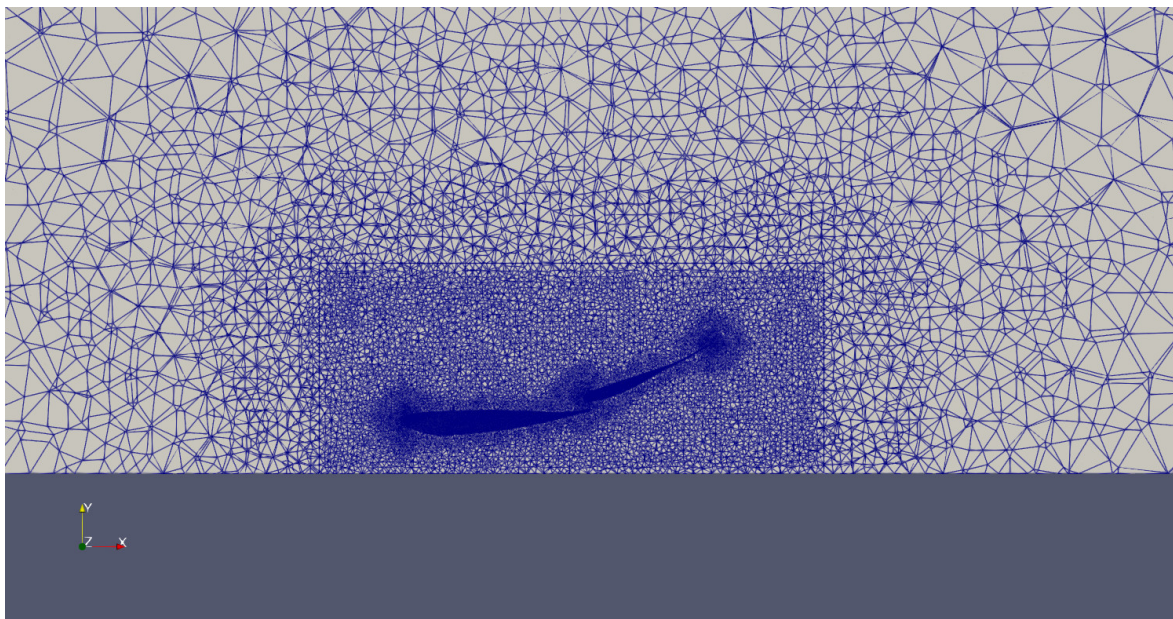


Figura A.8: Malha do caso h35

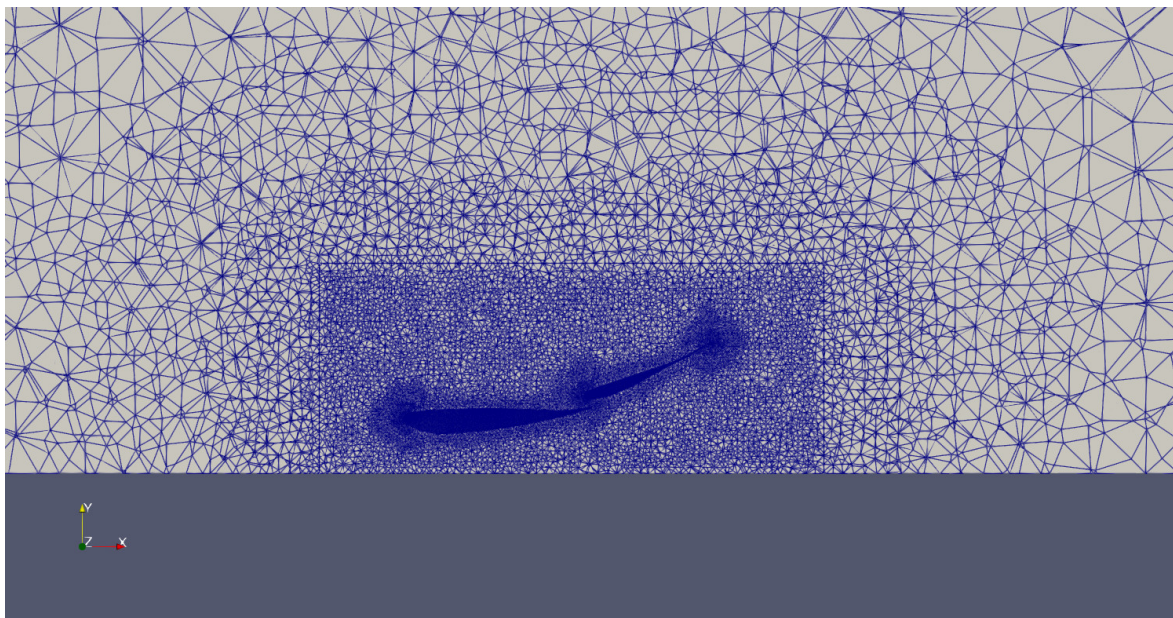


Figura A.9: Malha do caso h37



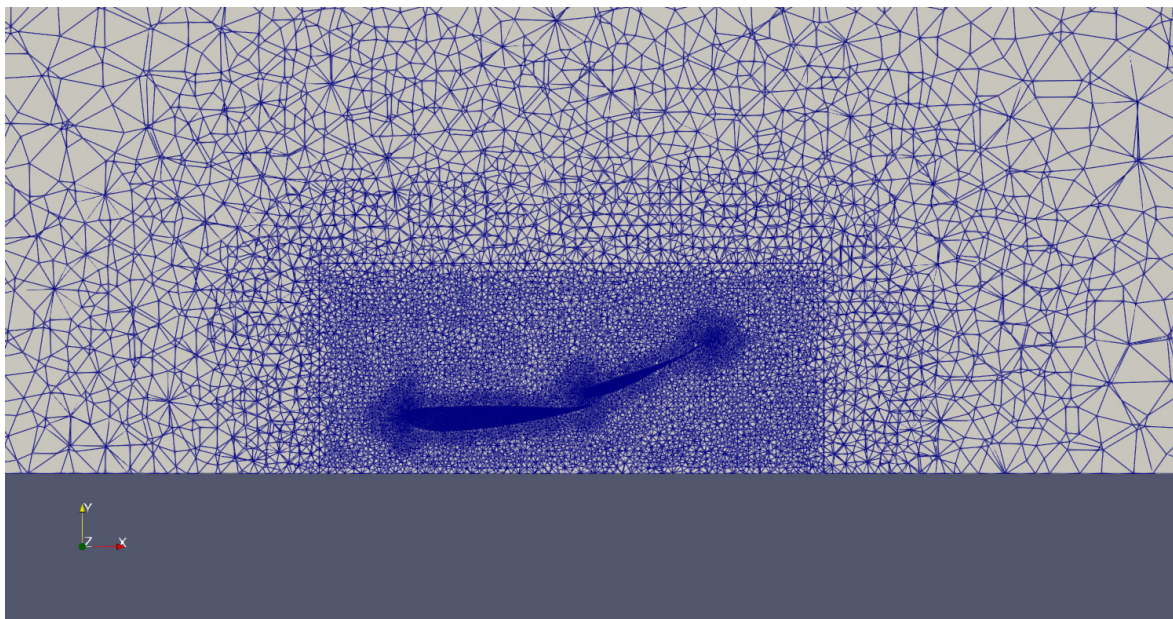


Figura A.10: Malha do caso h40

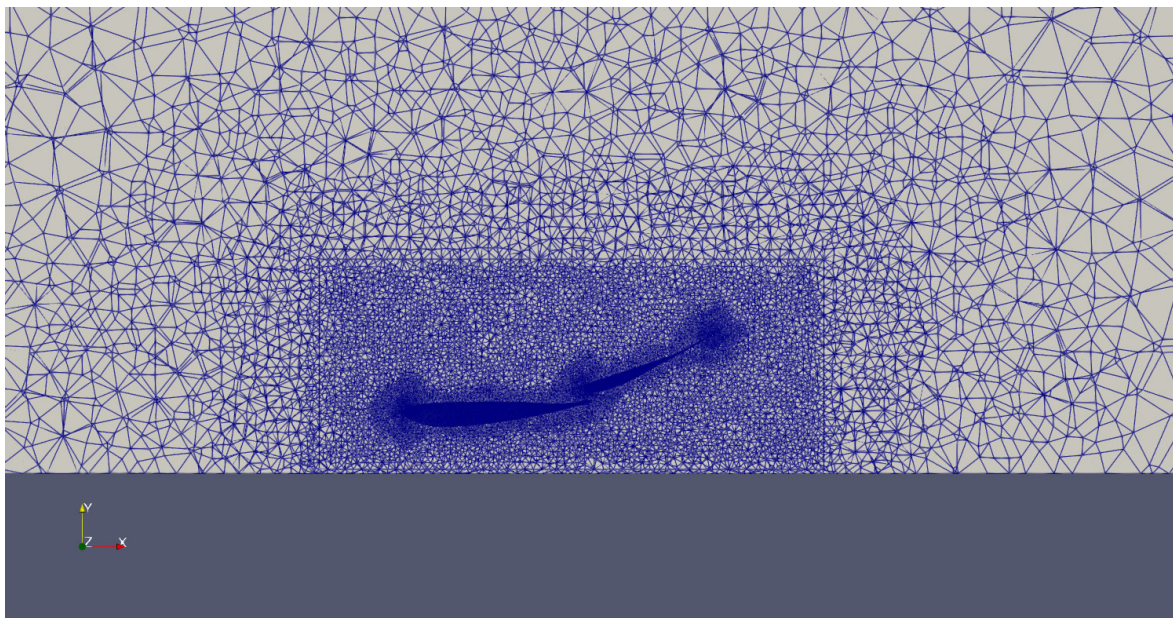


Figura A.11: Malha do caso h45



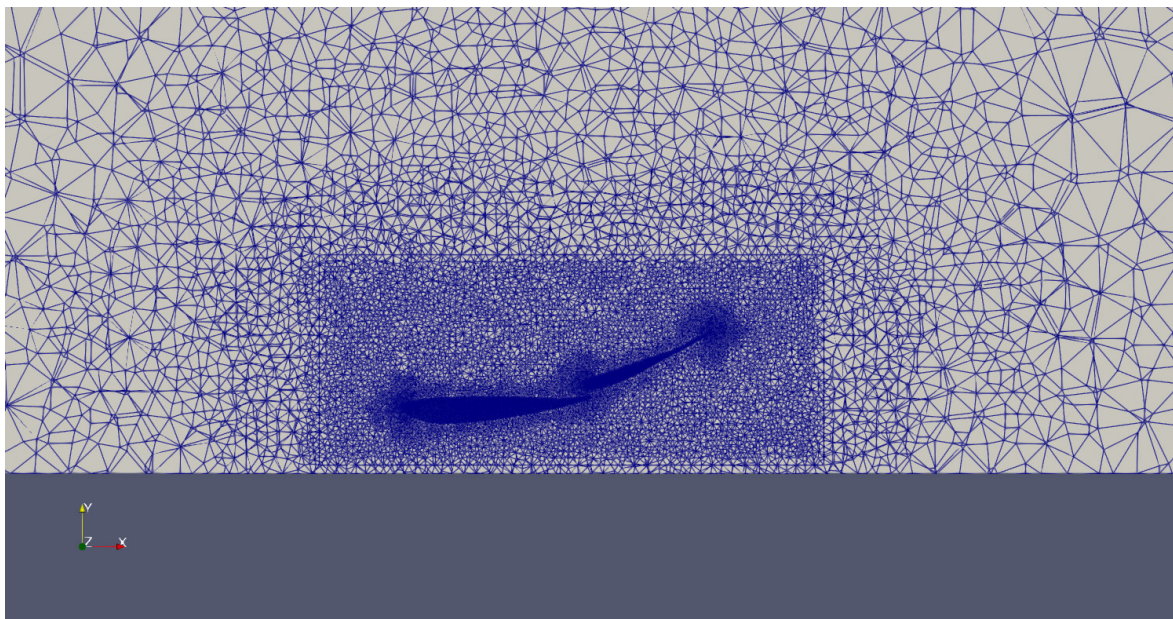


Figura A.12: Malha do caso h50

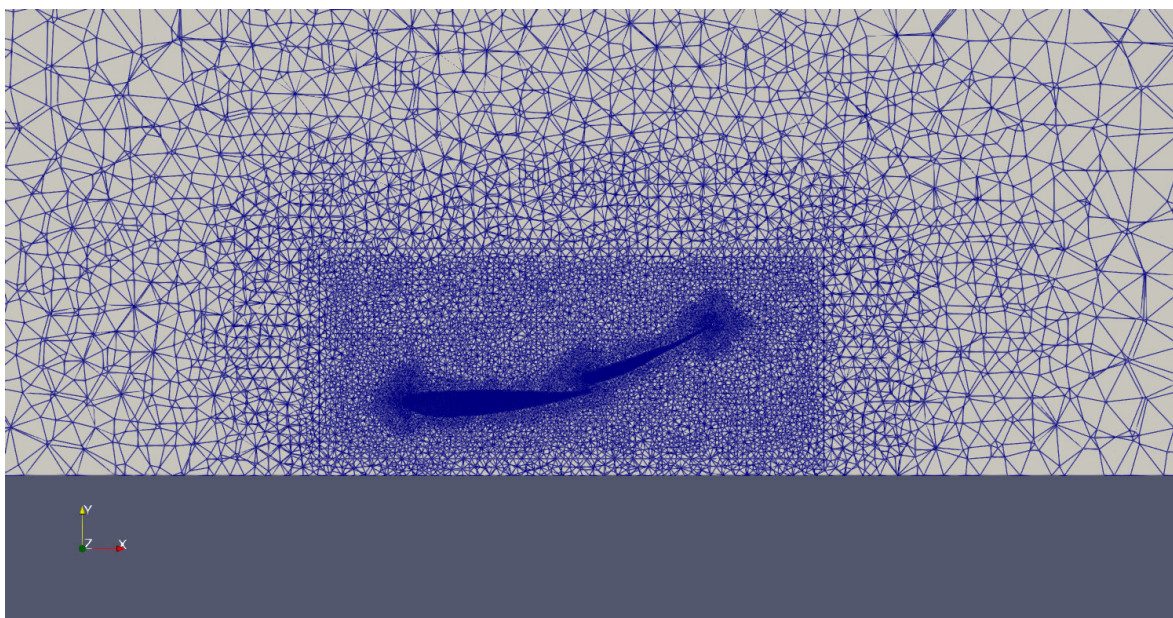


Figura A.13: Malha do caso h60



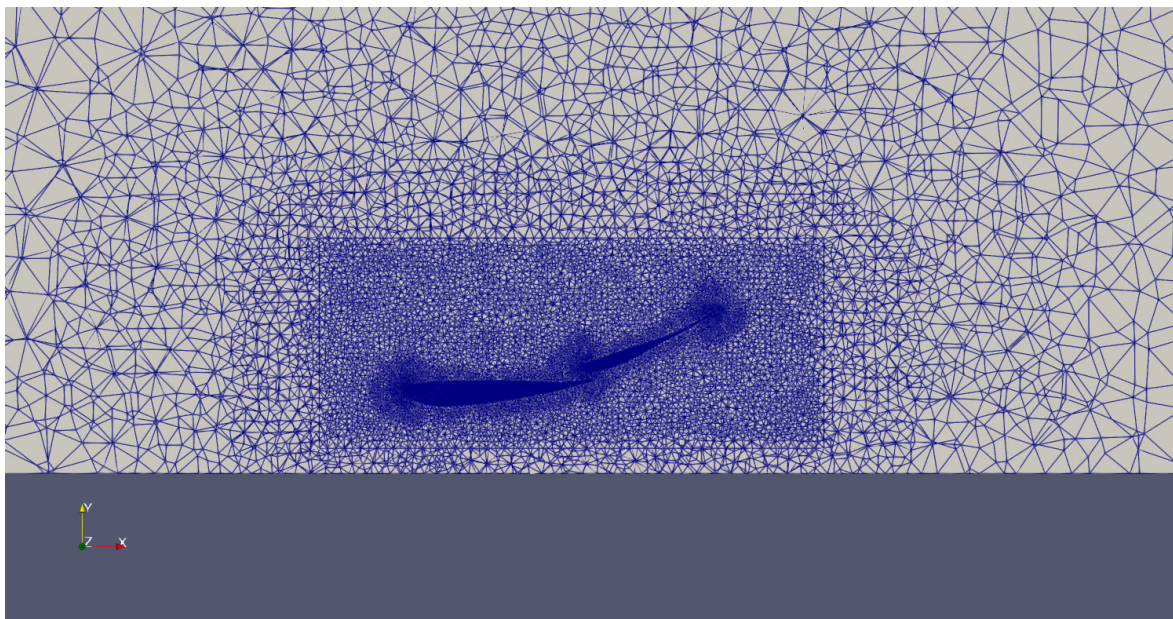


Figura A.14: Malha do caso h70

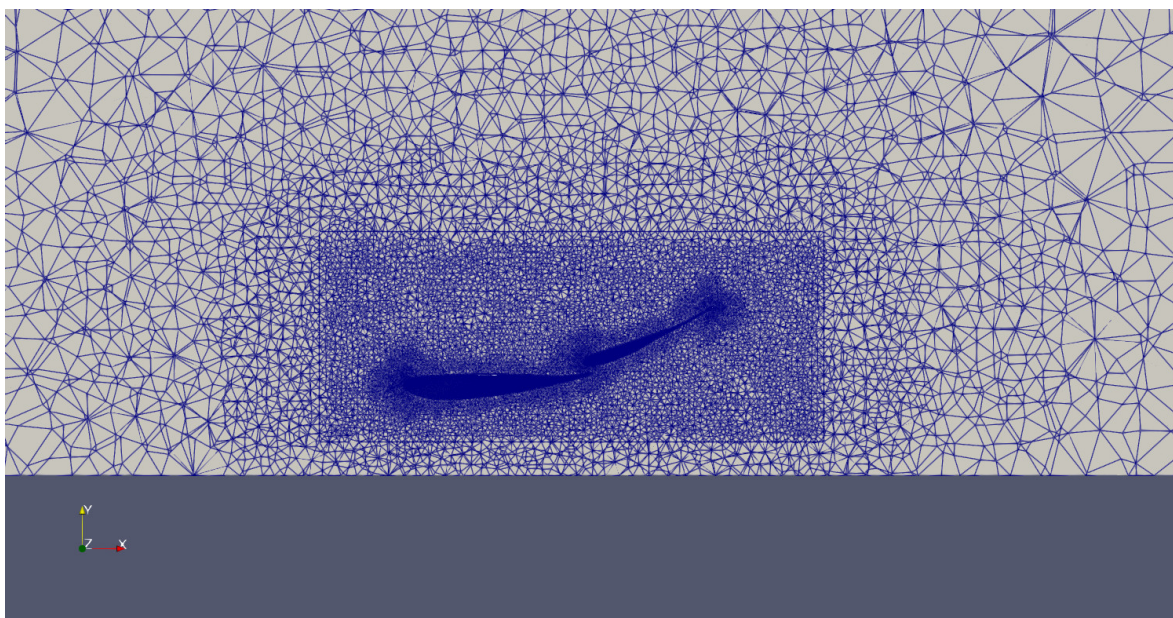


Figura A.15: Malha do caso h80



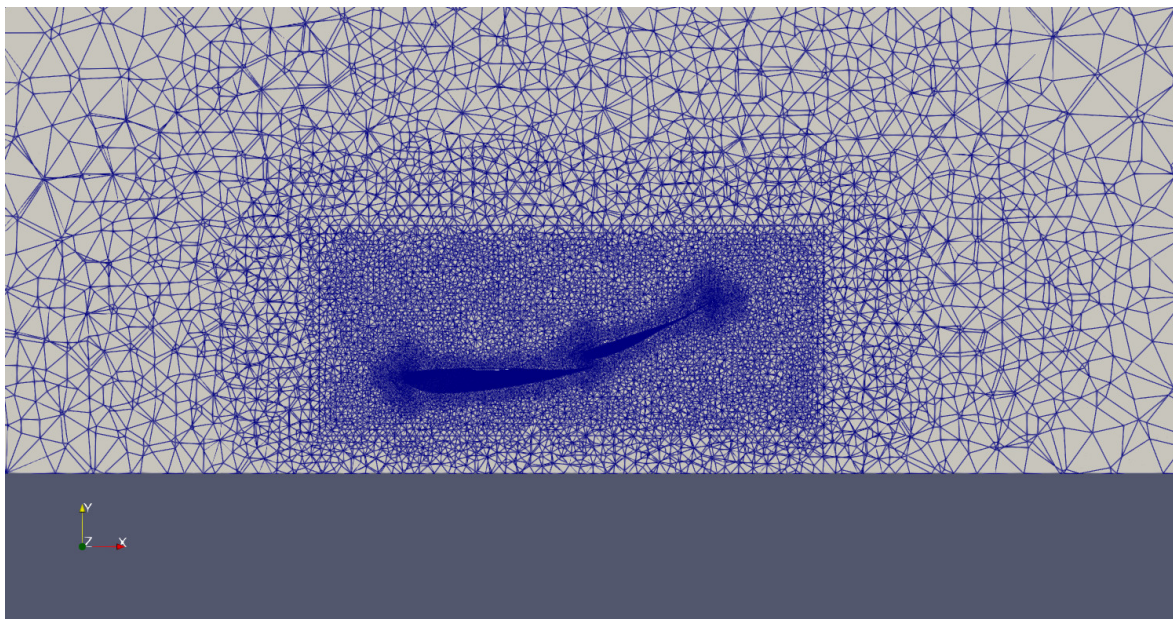


Figura A.16: Malha do caso h85

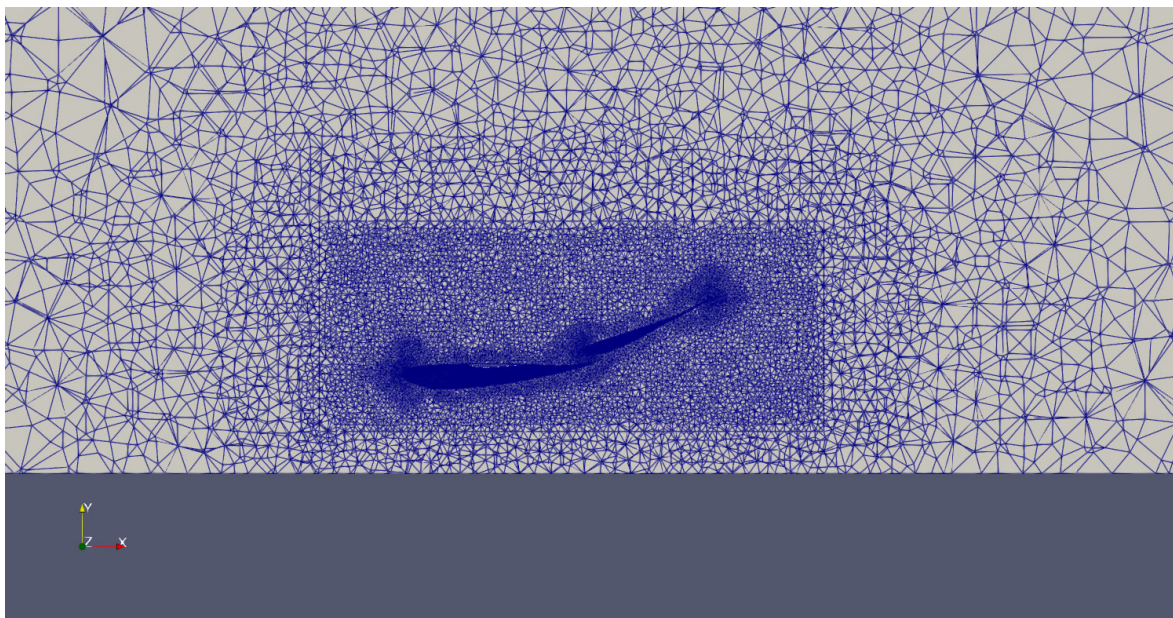


Figura A.17: Malha do caso h90



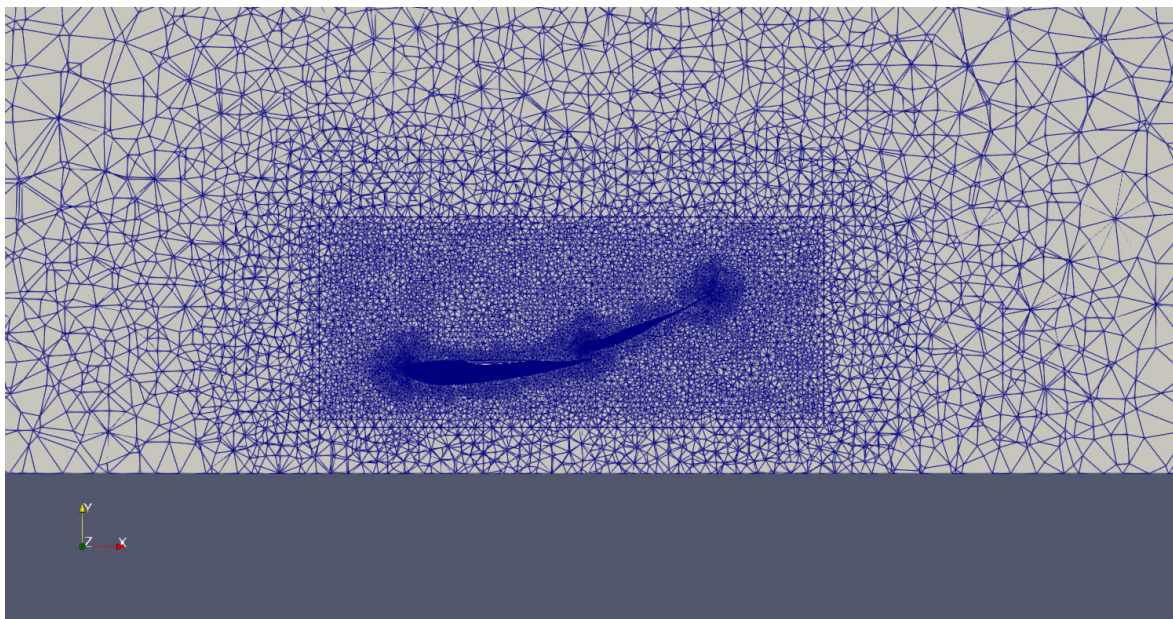


Figura A.18: Malha do caso h95

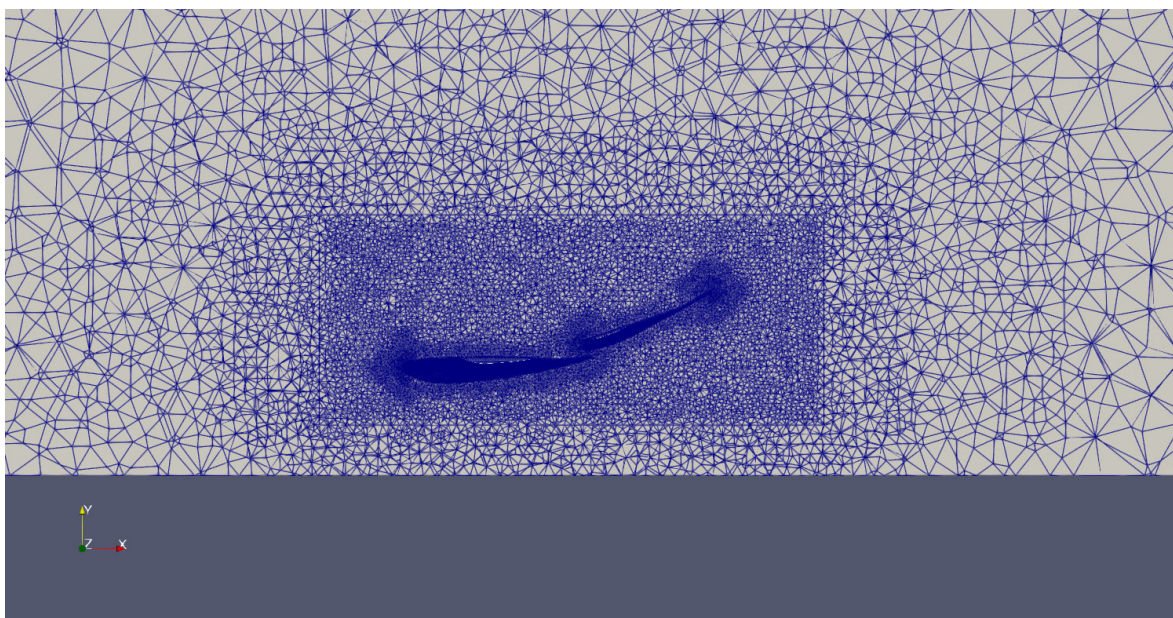


Figura A.19: Malha do caso h100



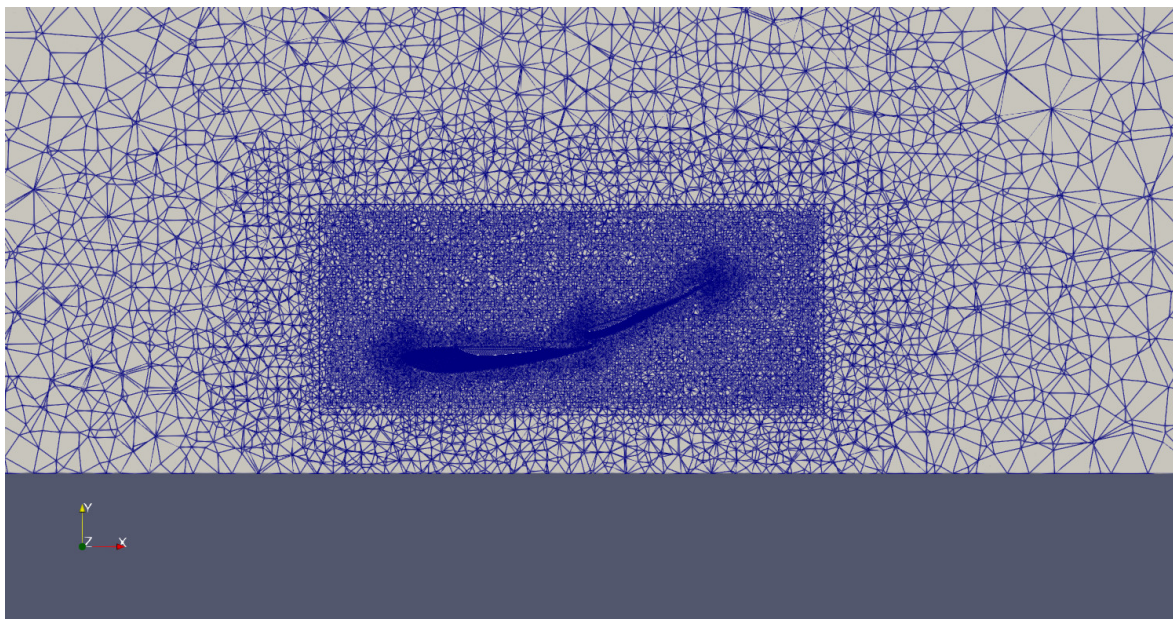


Figura A.20: Malha do caso h110

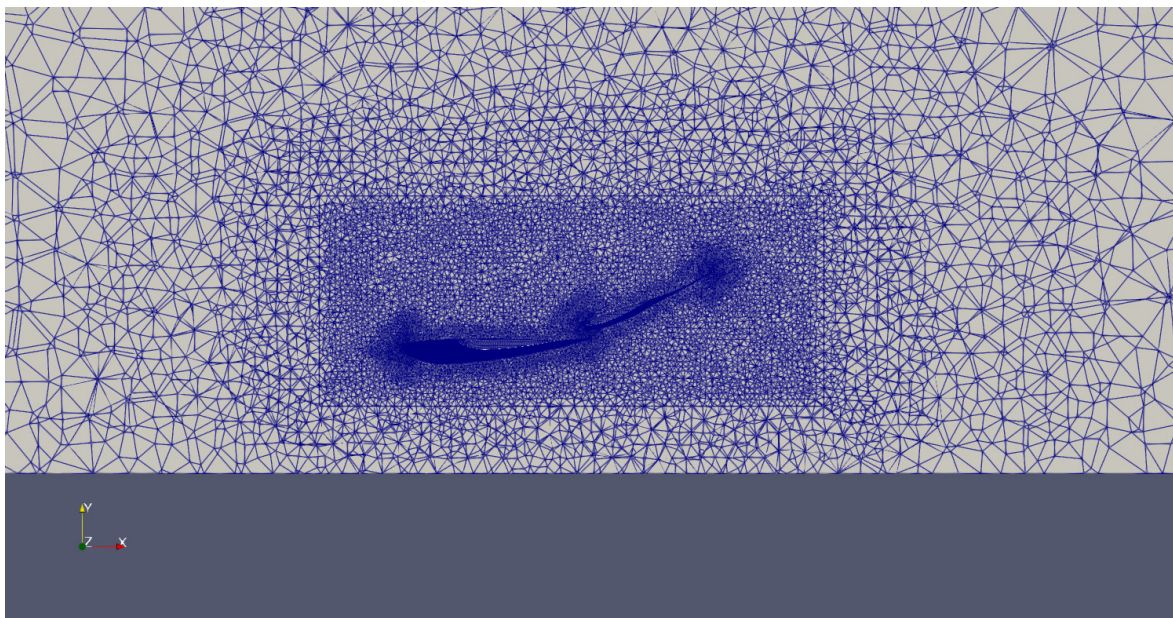


Figura A.21: Malha do caso h120



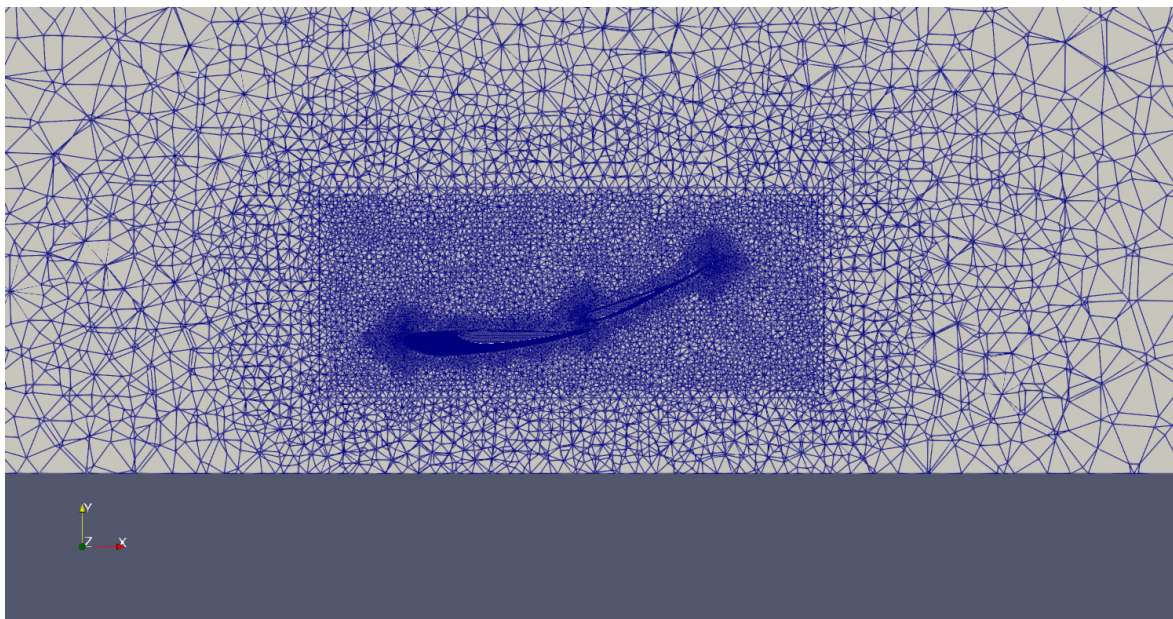


Figura A.22: Malha do caso h130

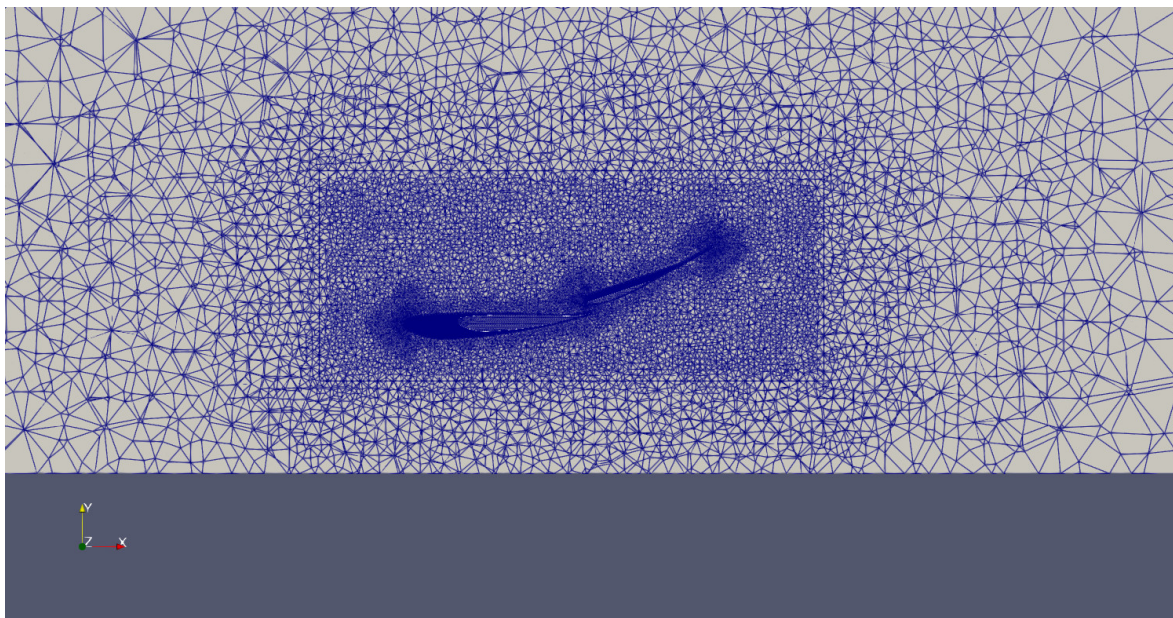


Figura A.23: Malha do caso h150

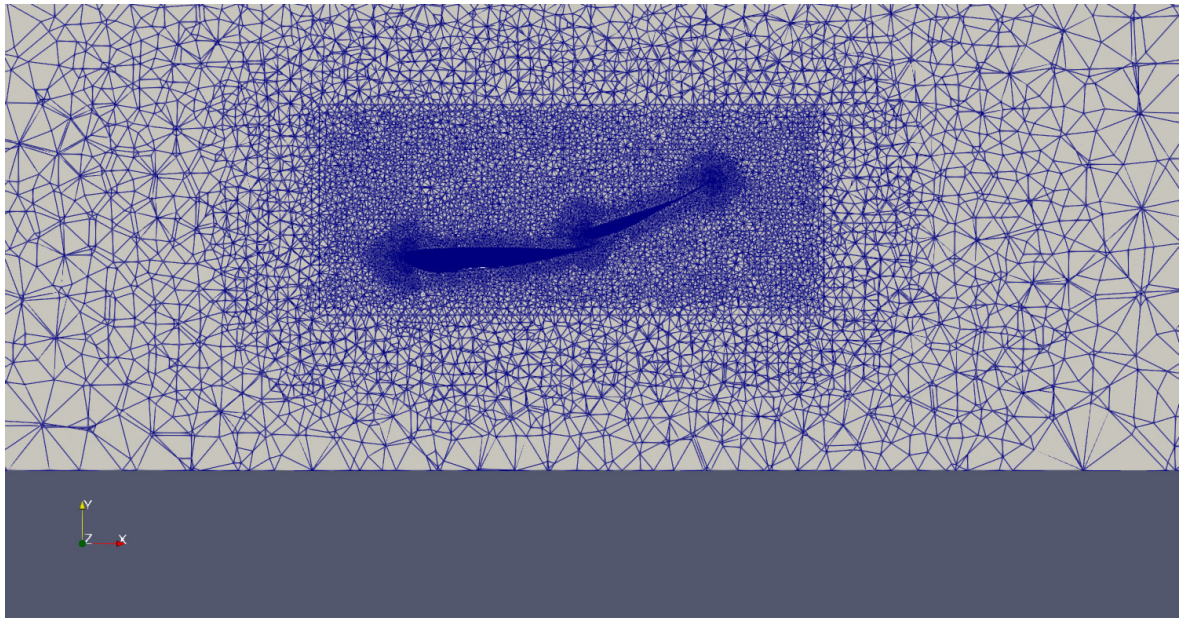


Figura A.24: Malha do caso h225



# APÊNDICE B

## Gráficos de Convergência das Simulações

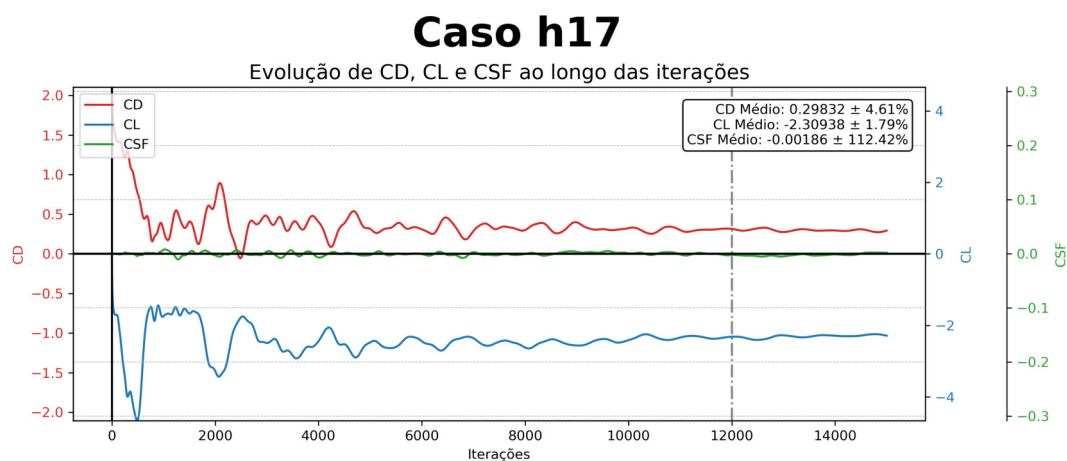


Figura B.1: Convergência do caso h17 para CD, CL e CSF ao longo das iterações

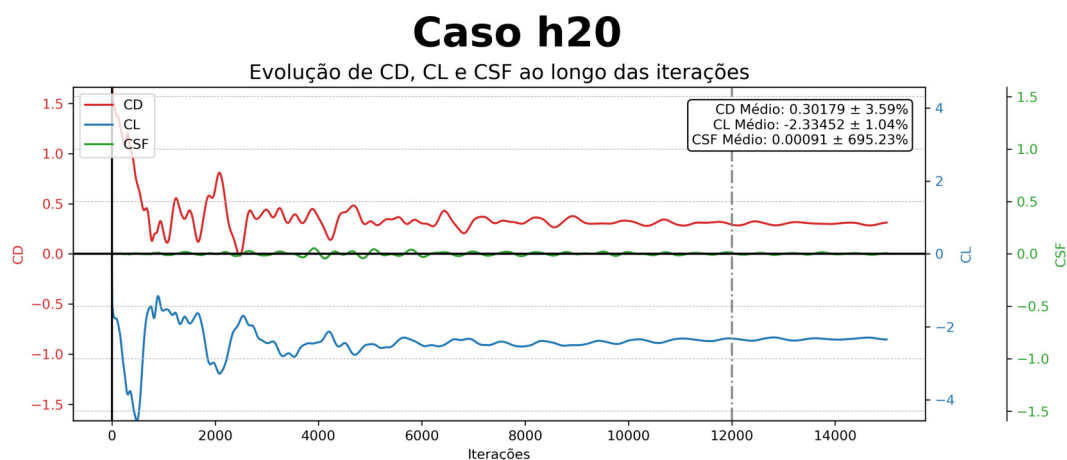


Figura B.2: Convergência do caso h20 para CD, CL e CSF ao longo das iterações

## Caso h22

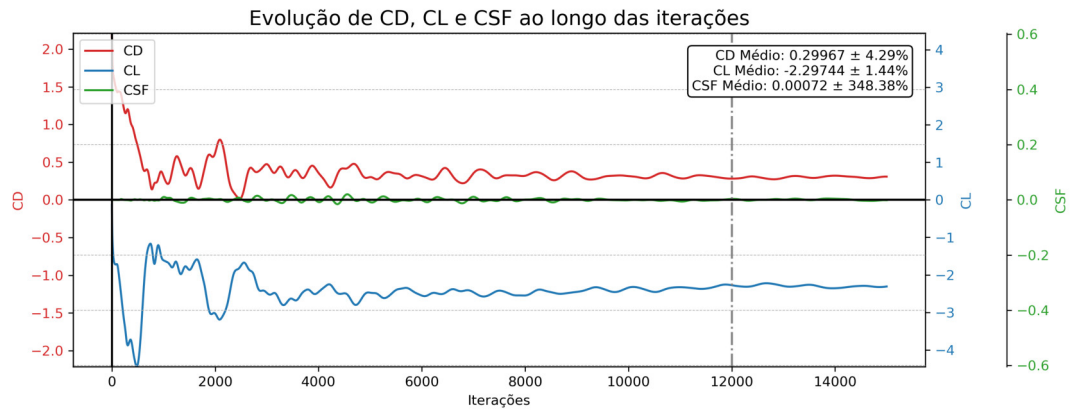


Figura B.3: Convergência do caso h22 para CD, CL e CSF ao longo das iterações

## Caso h25

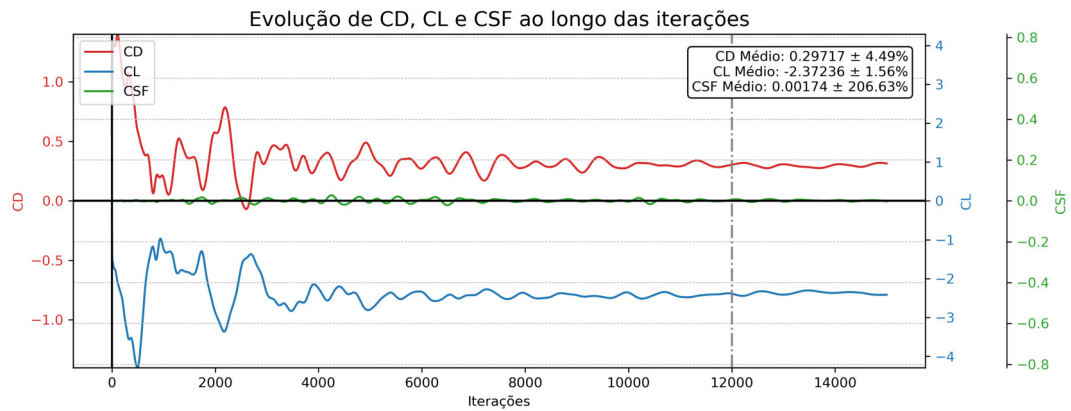


Figura B.4: Convergência do caso h25 para CD, CL e CSF ao longo das iterações

## Caso h27

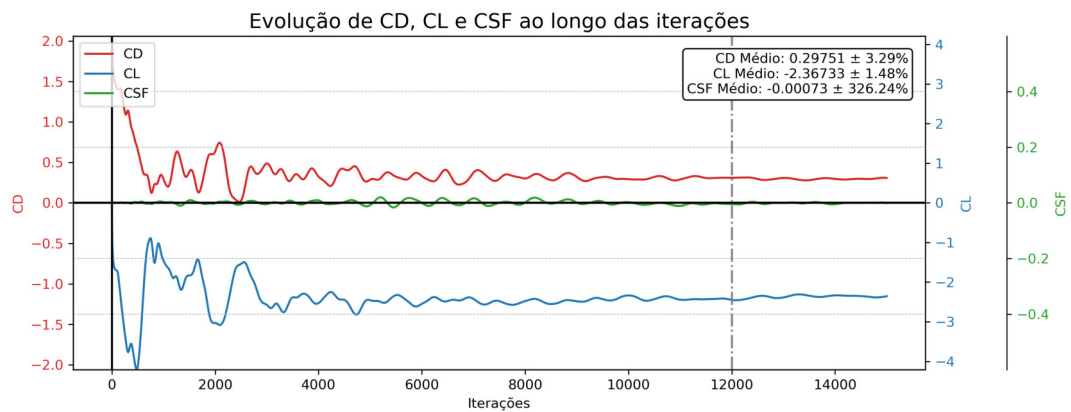


Figura B.5: Convergência do caso h27 para CD, CL e CSF ao longo das iterações

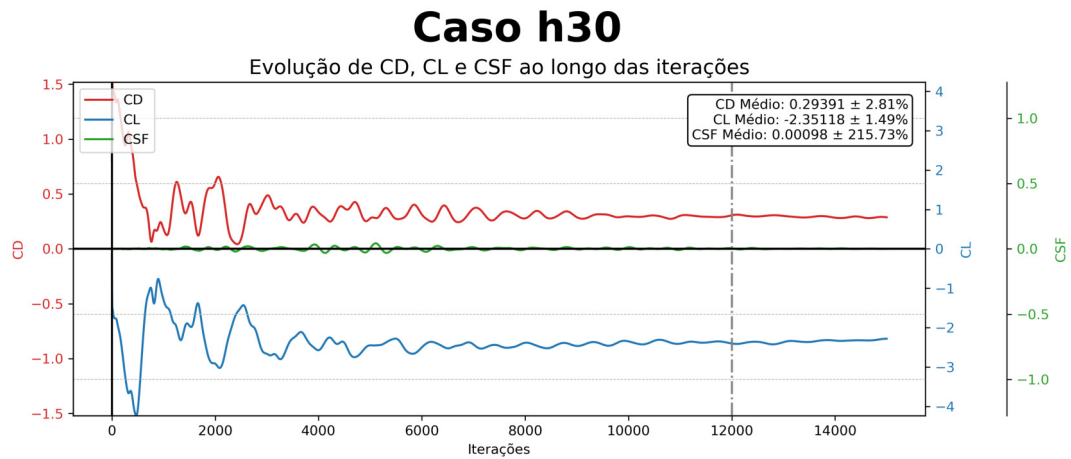


Figura B.6: Convergência do caso h30 para CD, CL e CSF ao longo das iterações

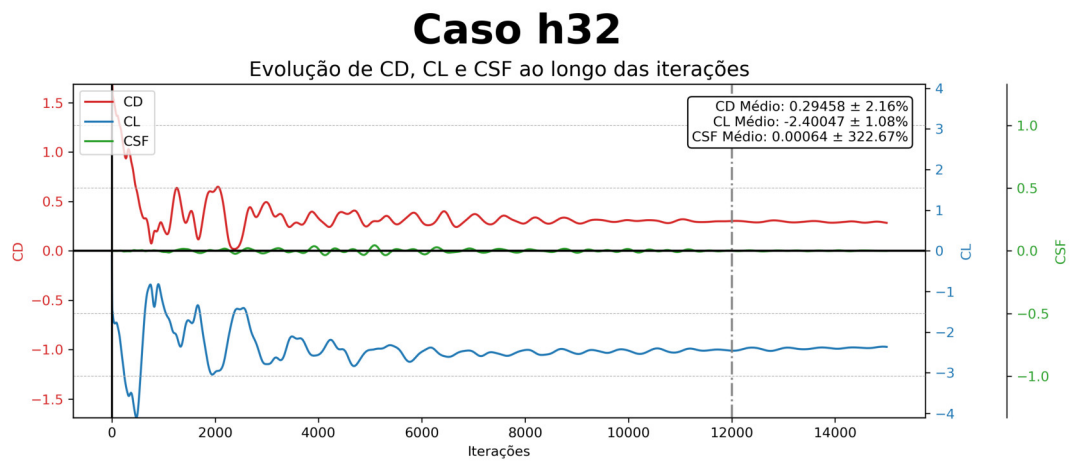


Figura B.7: Convergência do caso h32 para CD, CL e CSF ao longo das iterações

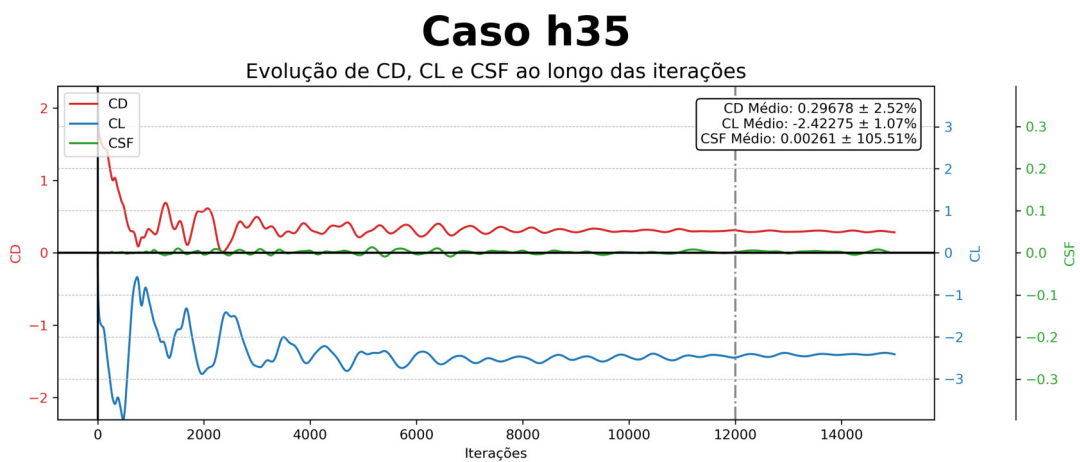


Figura B.8: Convergência do caso h35 para CD, CL e CSF ao longo das iterações

## Caso h37

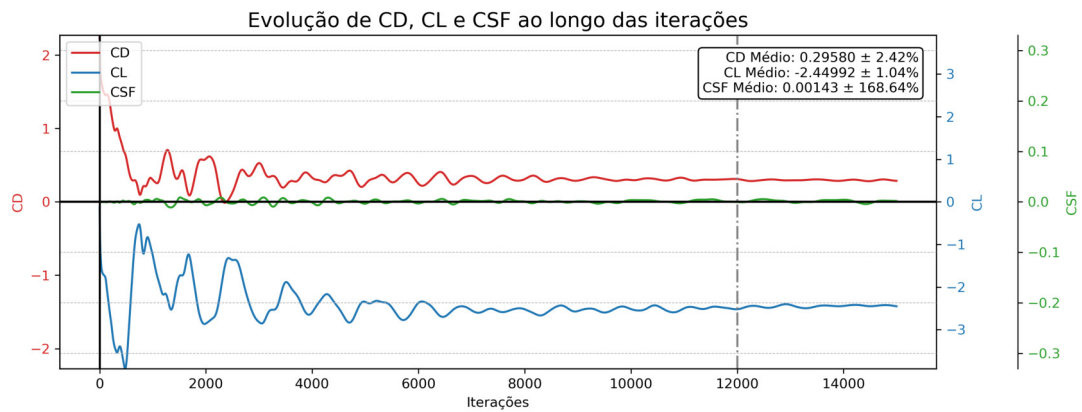


Figura B.9: Convergência do caso h37 para CD, CL e CSF ao longo das iterações

## Caso h40

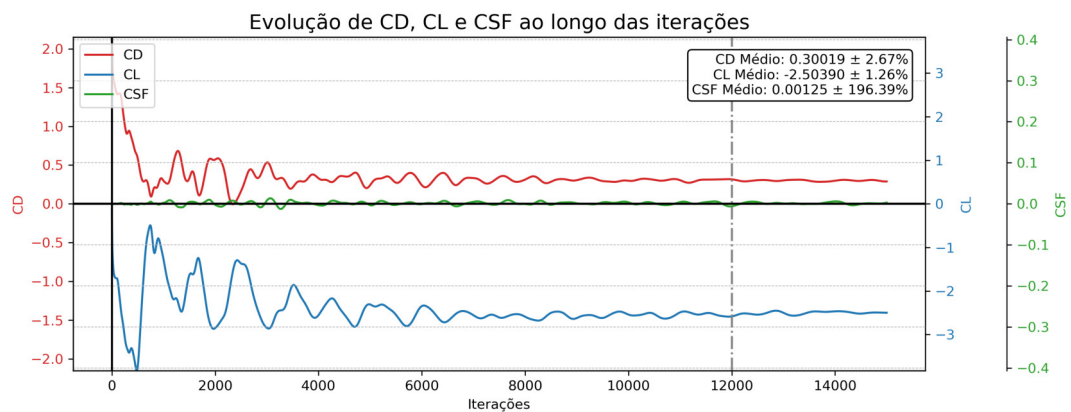


Figura B.10: Convergência do caso h40 para CD, CL e CSF ao longo das iterações

## Caso h45

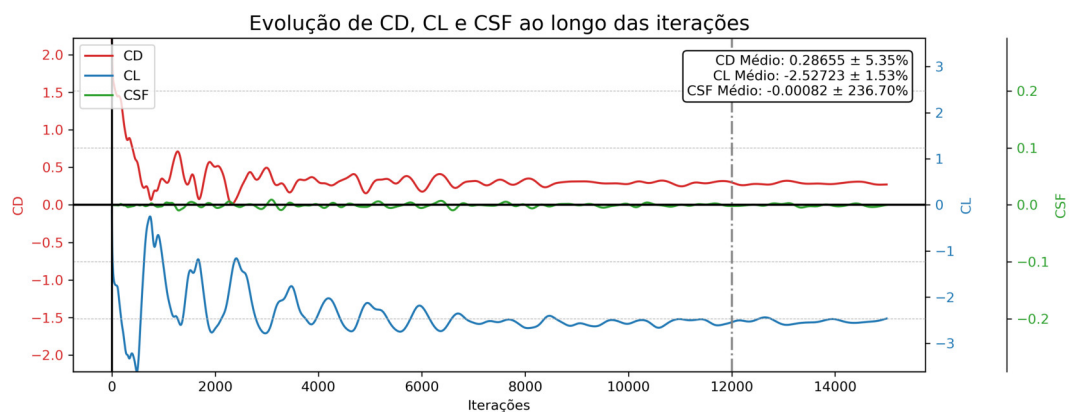


Figura B.11: Convergência do caso h45 para CD, CL e CSF ao longo das iterações



## Caso h50

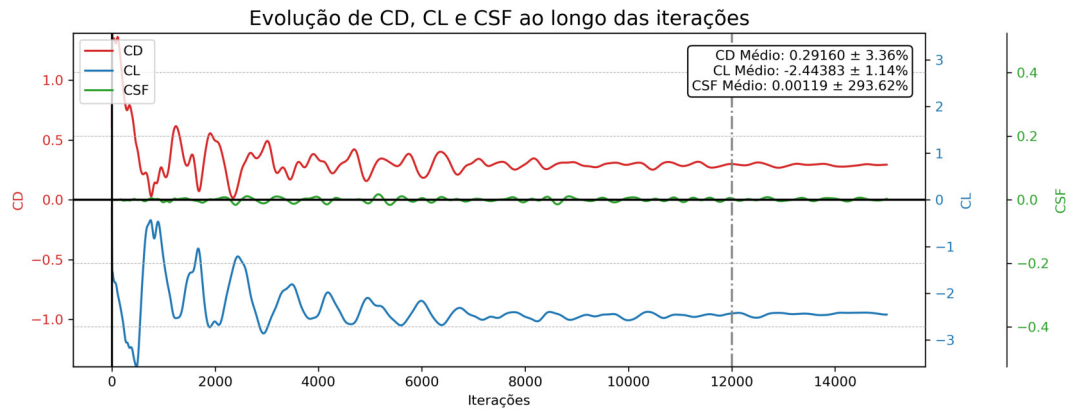


Figura B.12: Convergência do caso h50 para CD, CL e CSF ao longo das iterações

## Caso h60

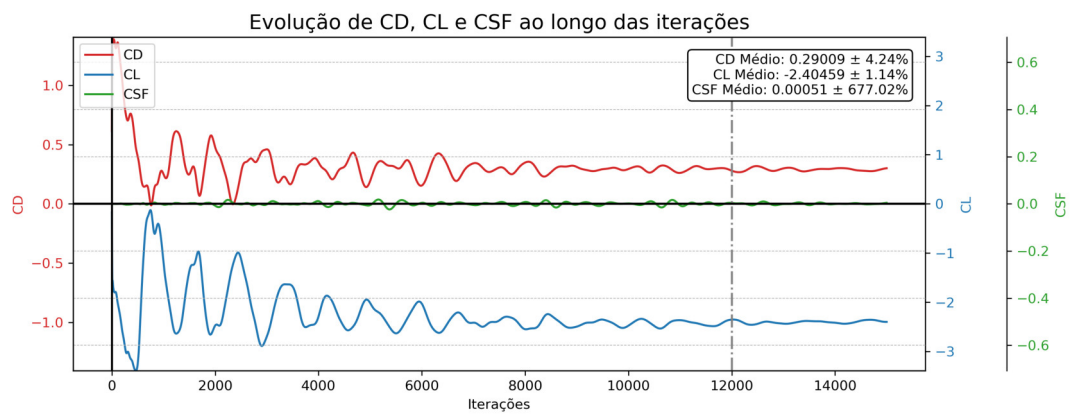


Figura B.13: Convergência do caso h60 para CD, CL e CSF ao longo das iterações

## Caso h70

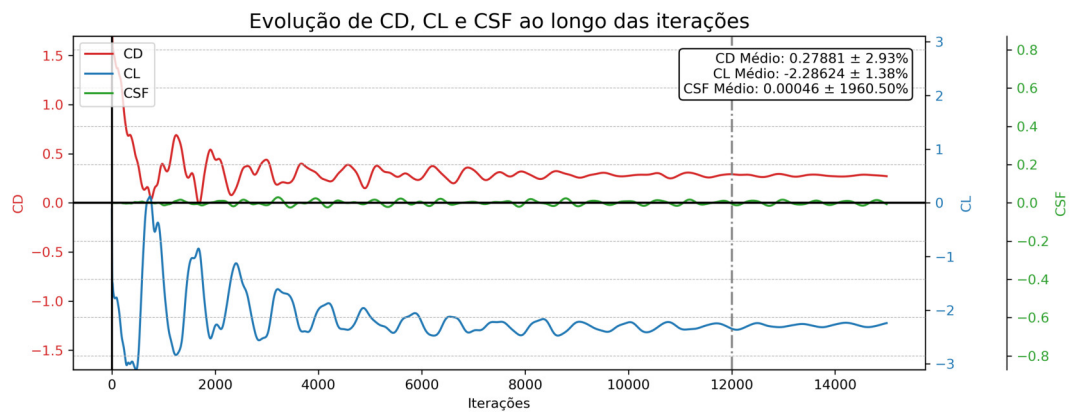


Figura B.14: Convergência do caso h70 para CD, CL e CSF ao longo das iterações

## Caso h80

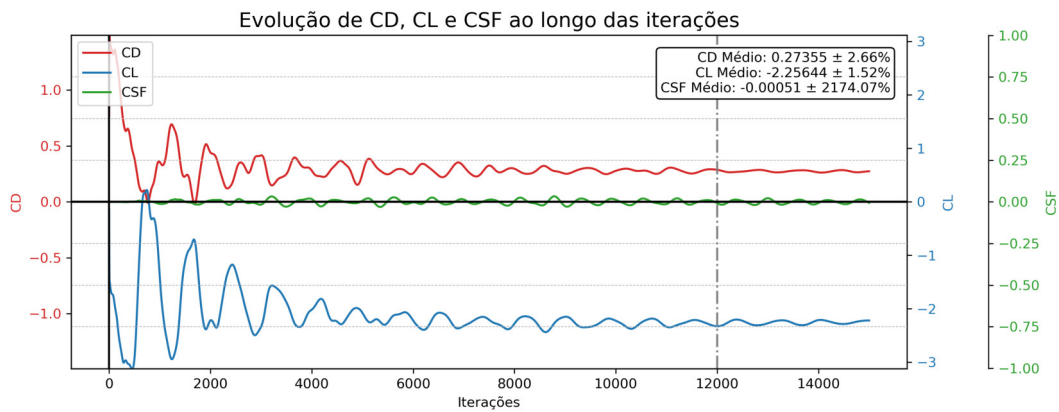


Figura B.15: Convergência do caso h80 para CD, CL e CSF ao longo das iterações

## Caso h85

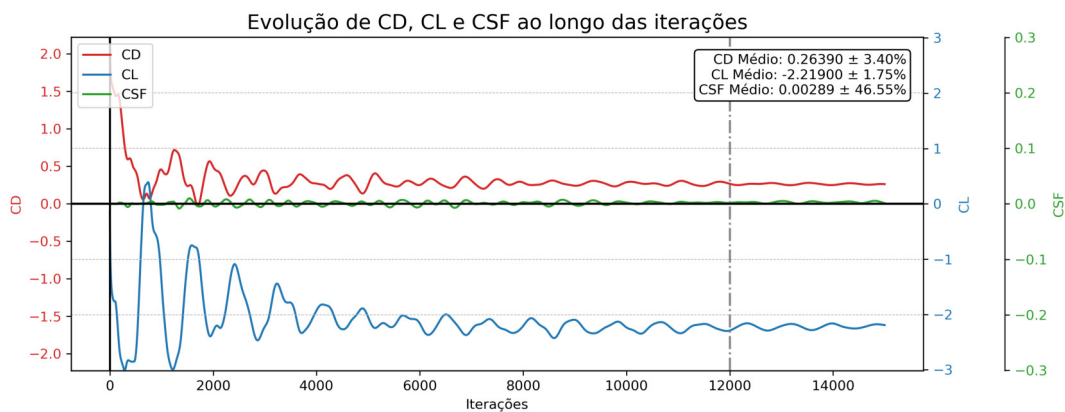


Figura B.16: Convergência do caso h85 para CD, CL e CSF ao longo das iterações

## Caso h90

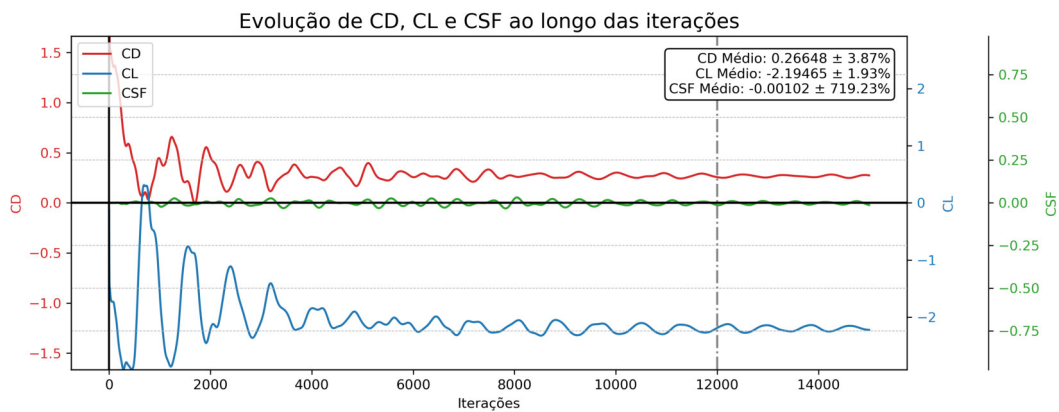


Figura B.17: Convergência do caso h90 para CD, CL e CSF ao longo das iterações

## Caso h95

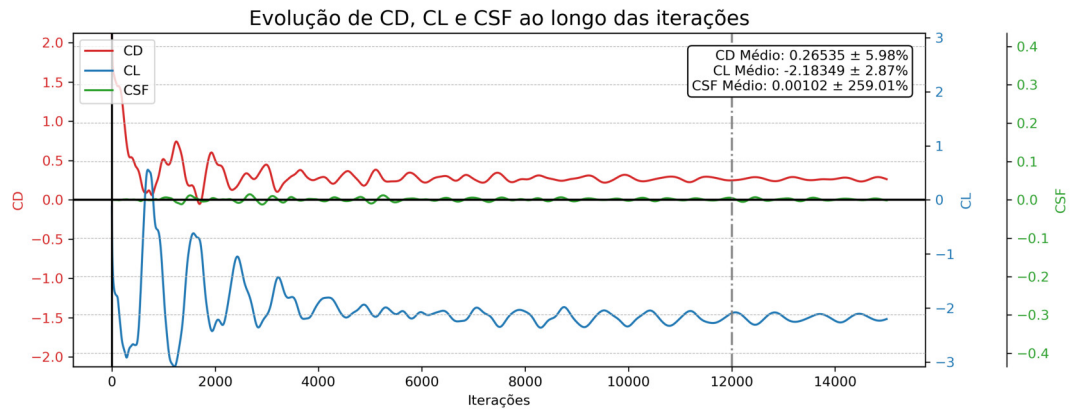


Figura B.18: Convergência do caso h95 para CD, CL e CSF ao longo das iterações

## Caso h100

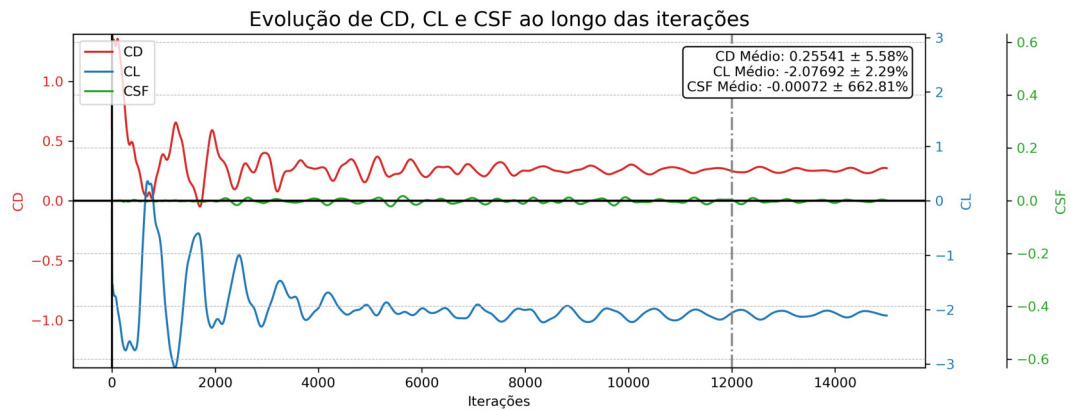


Figura B.19: Convergência do caso h100 para CD, CL e CSF ao longo das iterações

## Caso h110

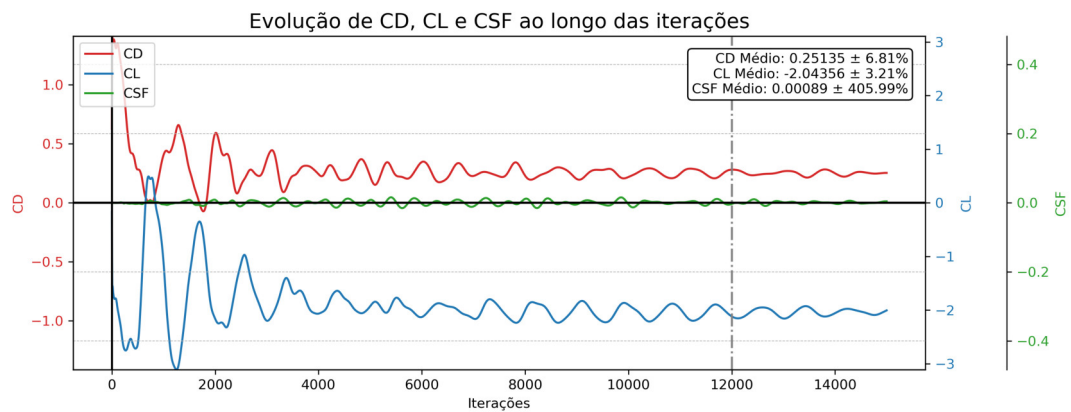


Figura B.20: Convergência do caso h110 para CD, CL e CSF ao longo das iterações

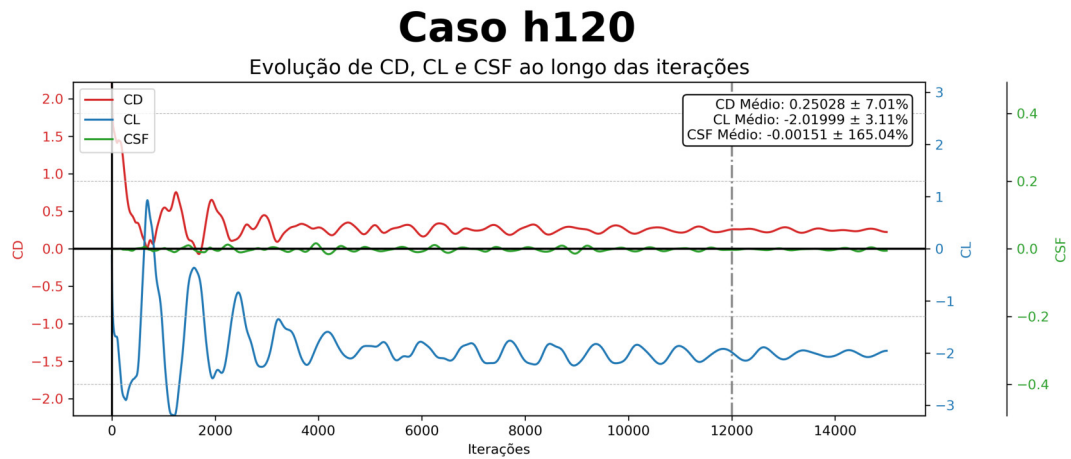


Figura B.21: Convergência do caso h120 para CD, CL e CSF ao longo das iterações

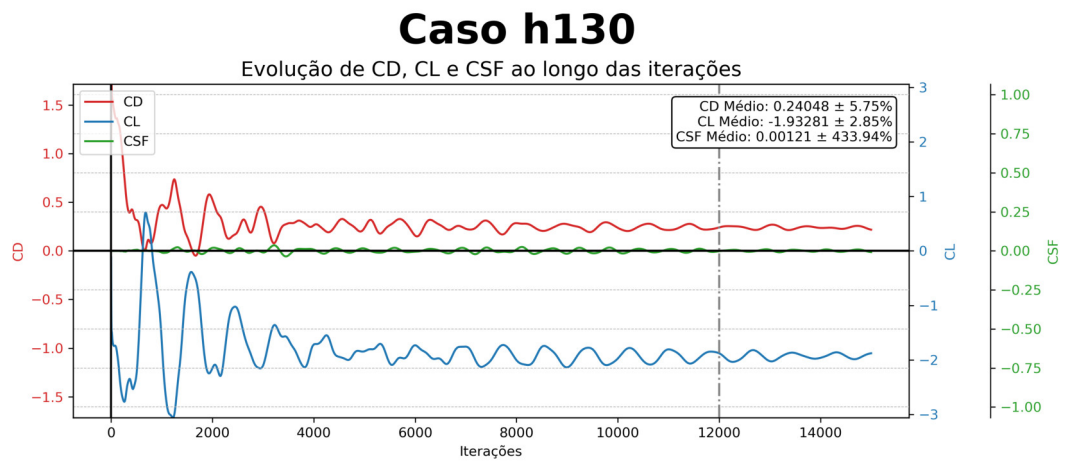


Figura B.22: Convergência do caso h130 para CD, CL e CSF ao longo das iterações

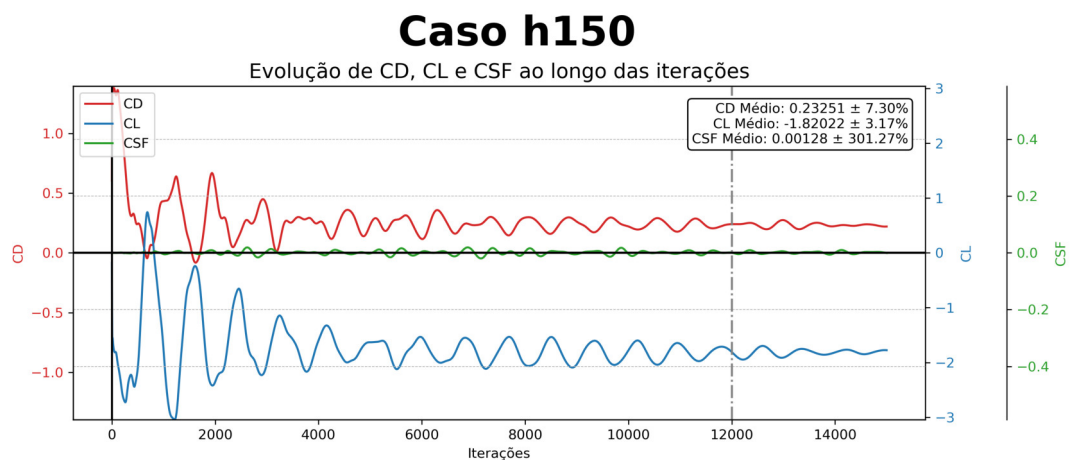


Figura B.23: Convergência do caso h150 para CD, CL e CSF ao longo das iterações

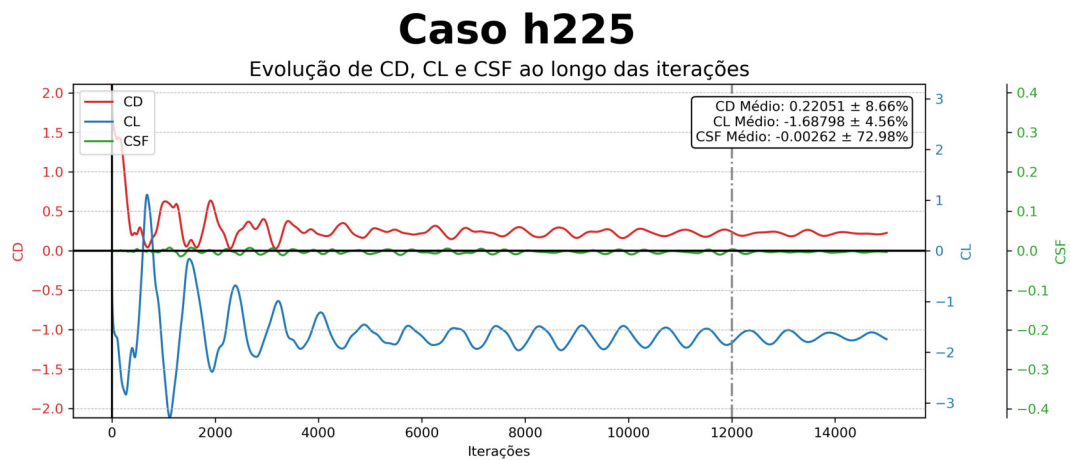


Figura B.24: Convergência do caso h225 para CD, CL e CSF ao longo das iterações

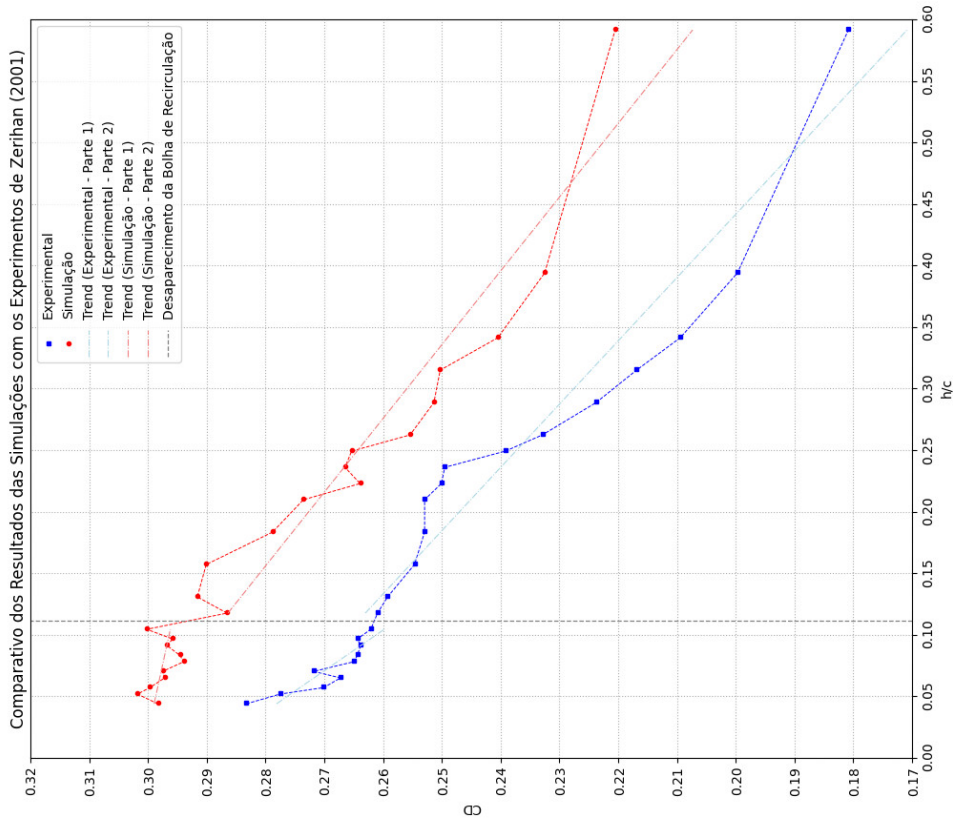
# APÊNDICE C

## Comparativo dos Resultados de Simulação com os Resultados Experimentais de Zerihan

h	h/c	CL - Zerihan	CD - Zerihan	CL - Simulation	Delta CL	CD - Simulation	Delta CD
17	0.04527	2.56022	0.28326	-2.30938	-0.25084	0.29832	0.01506
20	0.05358	2.65305	0.27754	-2.33452	-0.31853	0.30179	0.02425
22	0.05912	2.64552	0.27010	-2.29744	-0.34808	0.29967	0.02957
25	0.06651	2.66559	0.26724	-2.37236	-0.29323	0.29717	0.02993
27	0.07206	2.99176	0.27182	-2.36733	-0.62443	0.29751	0.02569
30	0.07945	3.02186	0.26495	-2.35118	-0.67068	0.29391	0.02896
32	0.08499	3.01685	0.26438	-2.40047	-0.61638	0.29458	0.03020
35	0.09330	3.00179	0.26381	-2.42275	-0.57904	0.29678	0.03297
37	0.09792	2.98925	0.26438	-2.44992	-0.53933	0.29580	0.03142
40	0.10624	2.96165	0.26209	-2.50390	-0.45775	0.30019	0.03810
45	0.11917	2.91900	0.26094	-2.52723	-0.39177	0.28655	0.02561
50	0.13210	2.88387	0.25923	-2.44383	-0.44004	0.29160	0.03237
60	0.15889	2.79857	0.25465	-2.40459	-0.39398	0.29009	0.03544
70	0.18568	2.71075	0.25293	-2.28624	-0.42451	0.27881	0.02588
80	0.21155	2.63799	0.25293	-2.25644	-0.38155	0.27355	0.02062
85	0.22448	2.61792	0.25007	-2.22550	-0.39242	0.26633	0.01626
90	0.23741	2.66559	0.24950	-2.12249	-0.54310	0.26968	0.02018
95	0.25035	2.59534	0.23920	-1.63974	-0.95560	0.25288	0.01368
100	0.26420	2.53763	0.23290	-2.14627	-0.39136	1.28244	1.04954
110	0.29007	2.42222	0.22375	-1.93141	-0.49081	0.36419	0.14044
120	0.31594	2.32688	0.21688	0.00000	-2.32688	0.00000	-0.21688
130	0.34273	2.24659	0.20944	0.00000	-2.24659	0.00000	-0.20944
150	0.39538	2.12366	0.19971	0.00000	-2.12366	0.00000	-0.19971
225	0.59222	1.86954	0.18083	-1.68798	-0.18156	0.22051	0.03968

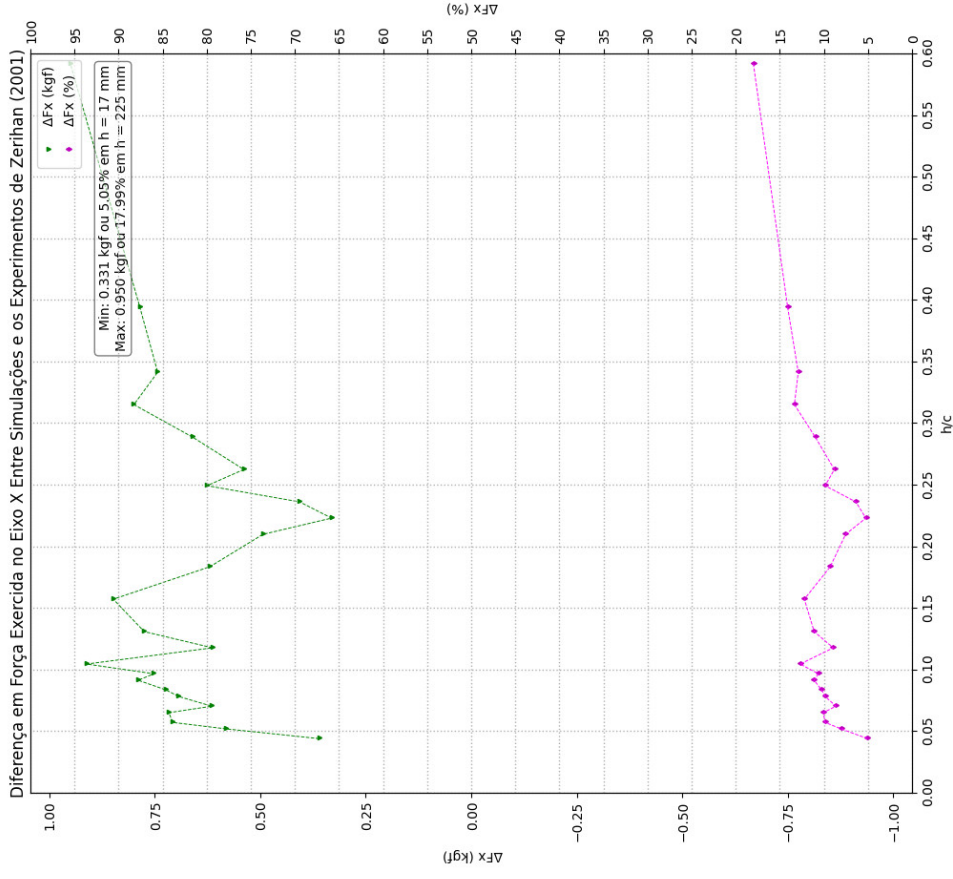
Tabela C.1: Comparativo dos resultados das simulações com os experimentos de Zerihan (2001)

## Coeficiente de Arrasto



(a) Comparativo do Coeficiente de Arrasto

## Comparativo dos Valores de $\Delta Fx$

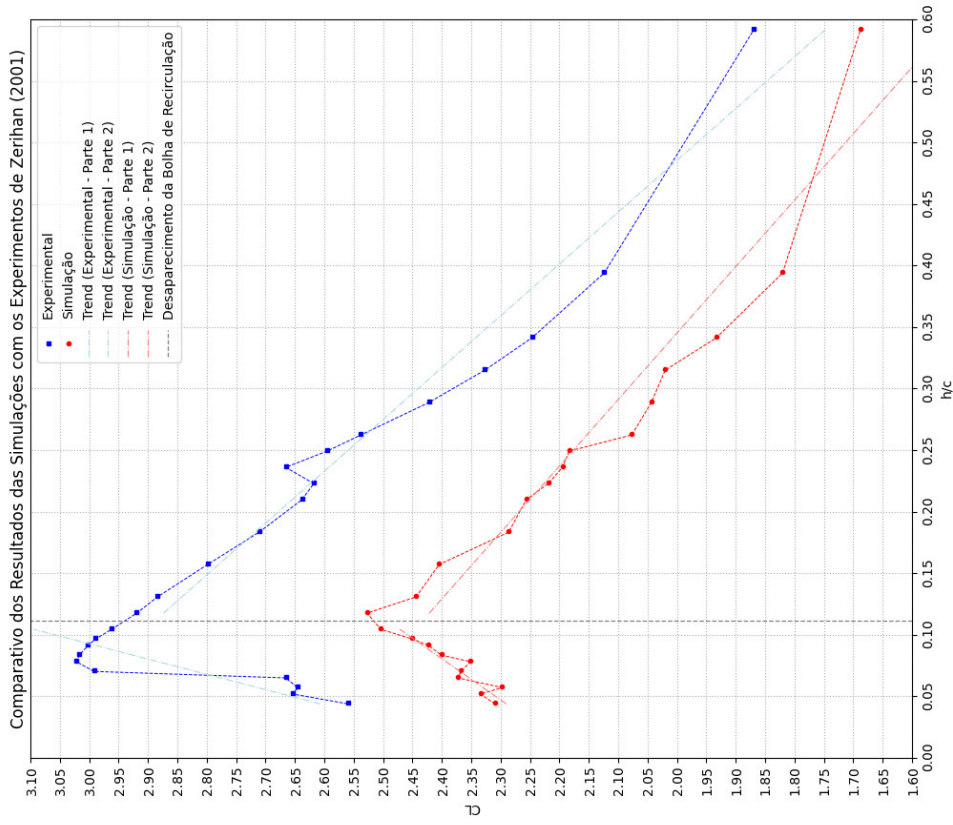


(b) Diferença entre os resultados obtidos nas simulações e os resultados de Zerihan (2001) dimensionalizados para kgf e em formato percentual

Figura C.1: Coeficiente de arrasto sobre altura adimensionalizada  $h/c$

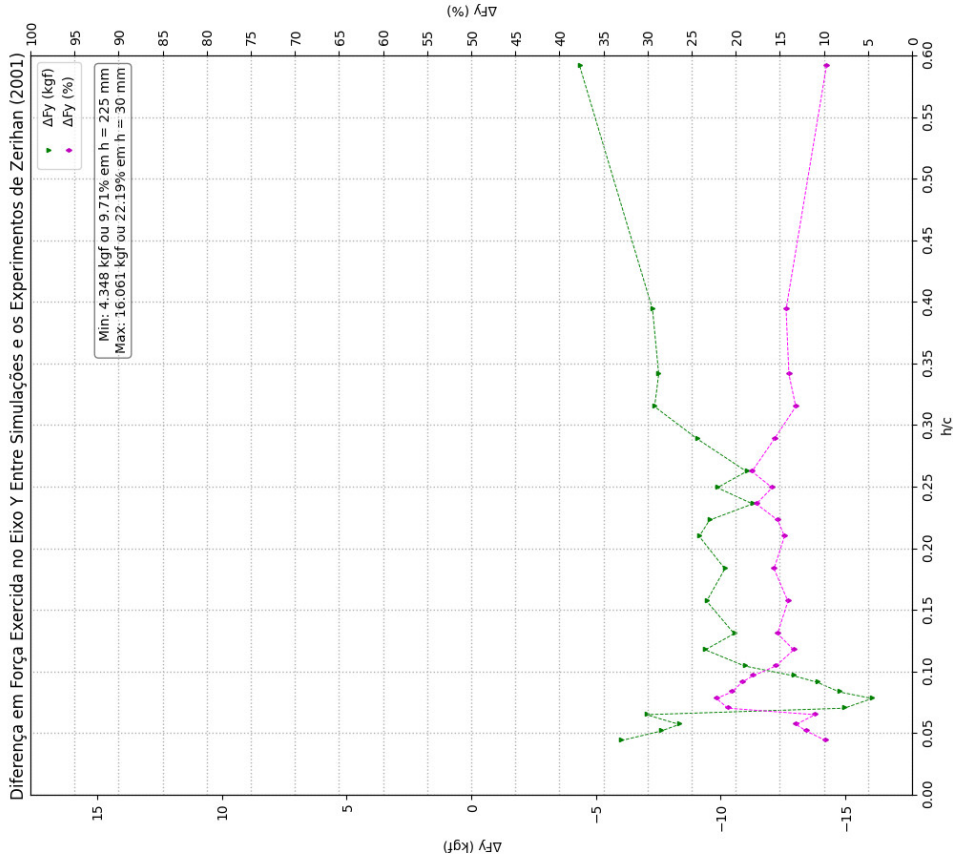


## Coeficiente de Sustentação



(a) Comparativo do Coeficiente de Sustentação

## Comparativo dos Valores de $\Delta F_y$



(b) Diferença entre os resultados obtidos nas simulações e os resultados de Zerihan (2001) dimensionalizados para kgf e em formato percentual

Figura C.2: Coeficiente de sustentação sobre altura adimensionalizada  $h/c$

# APÊNDICE D

## Visualizações de Velocidade - Plano xy

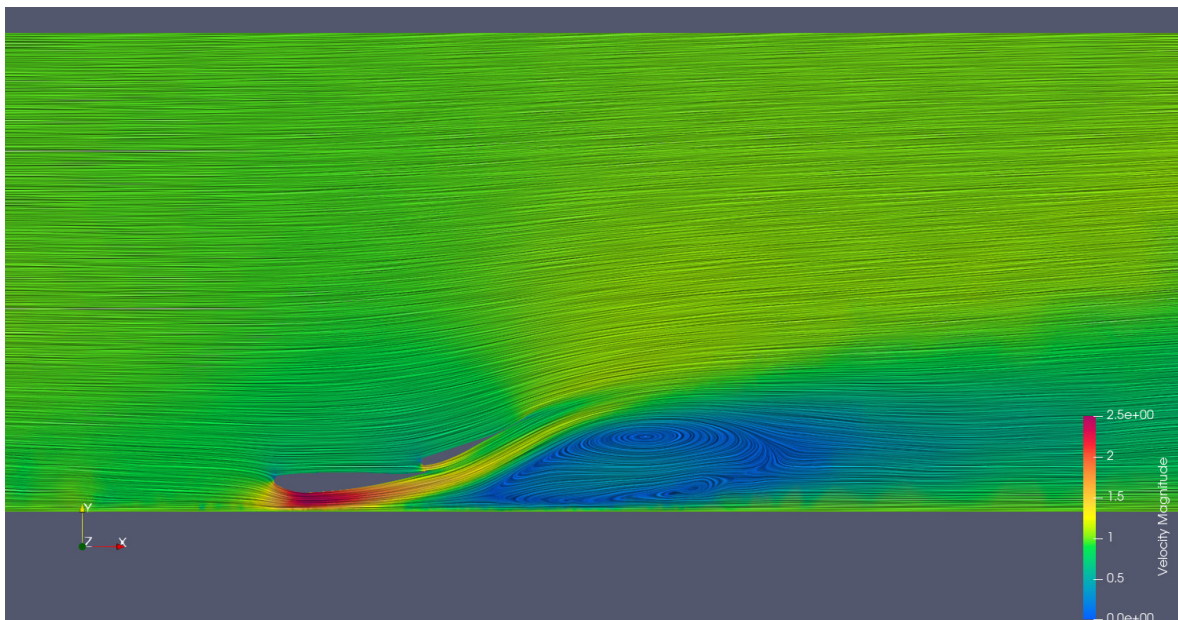


Figura D.1: Visualização de velocidade do escoamento no plano xy do caso h17

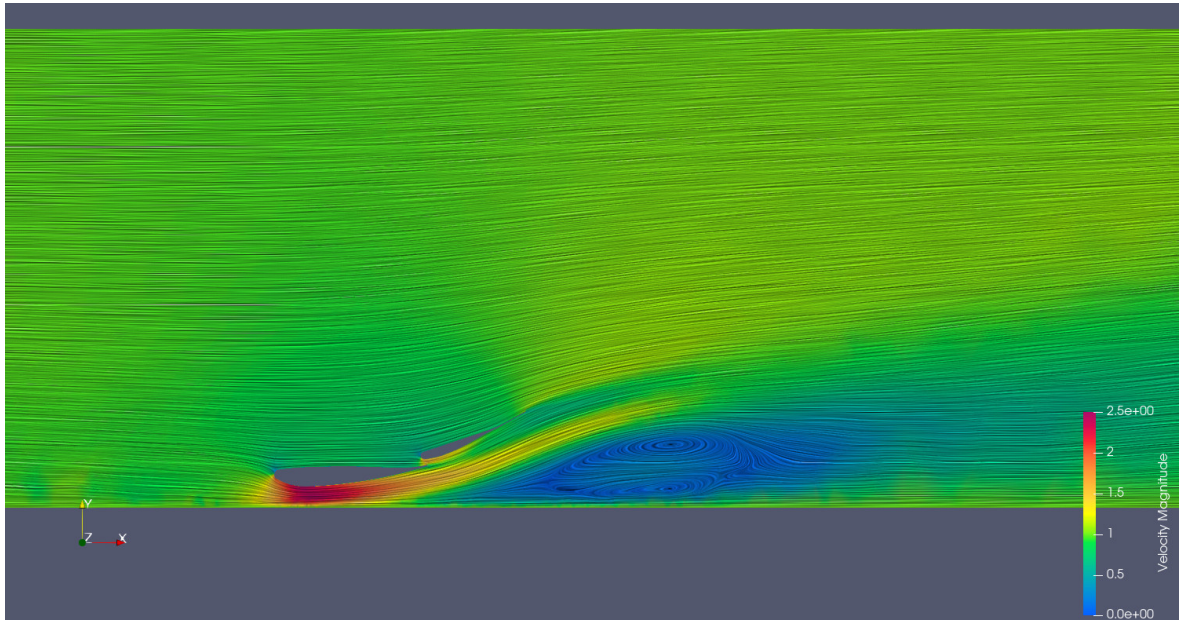


Figura D.2: Visualização de velocidade do escoamento no plano xy do caso h20

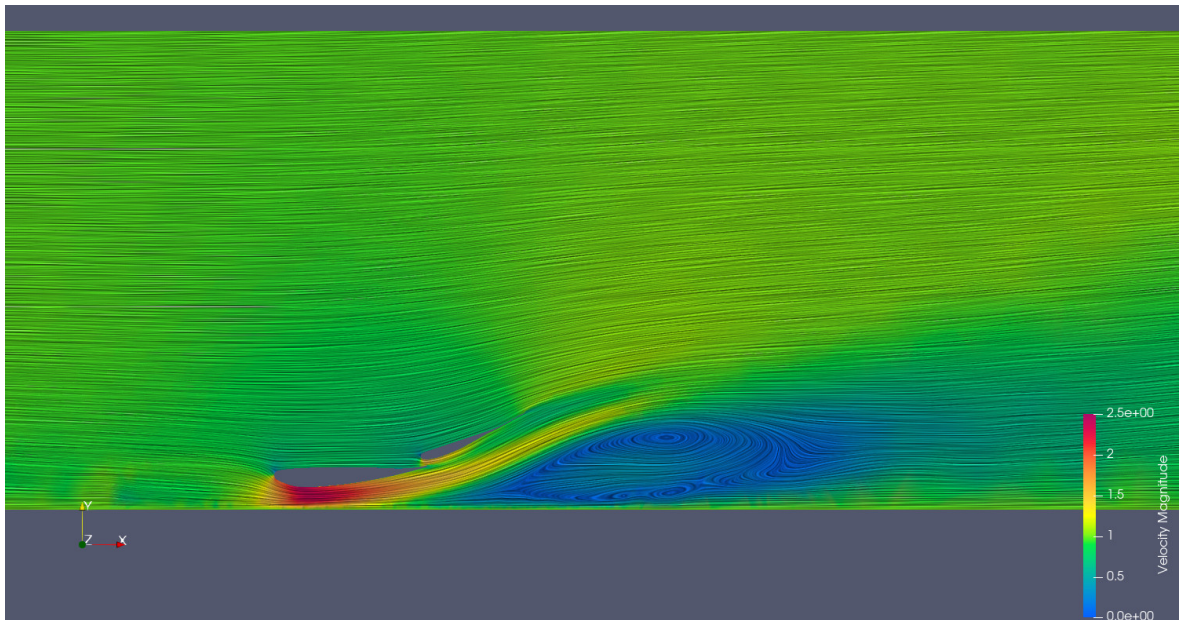


Figura D.3: Visualização de velocidade do escoamento no plano xy do caso h22



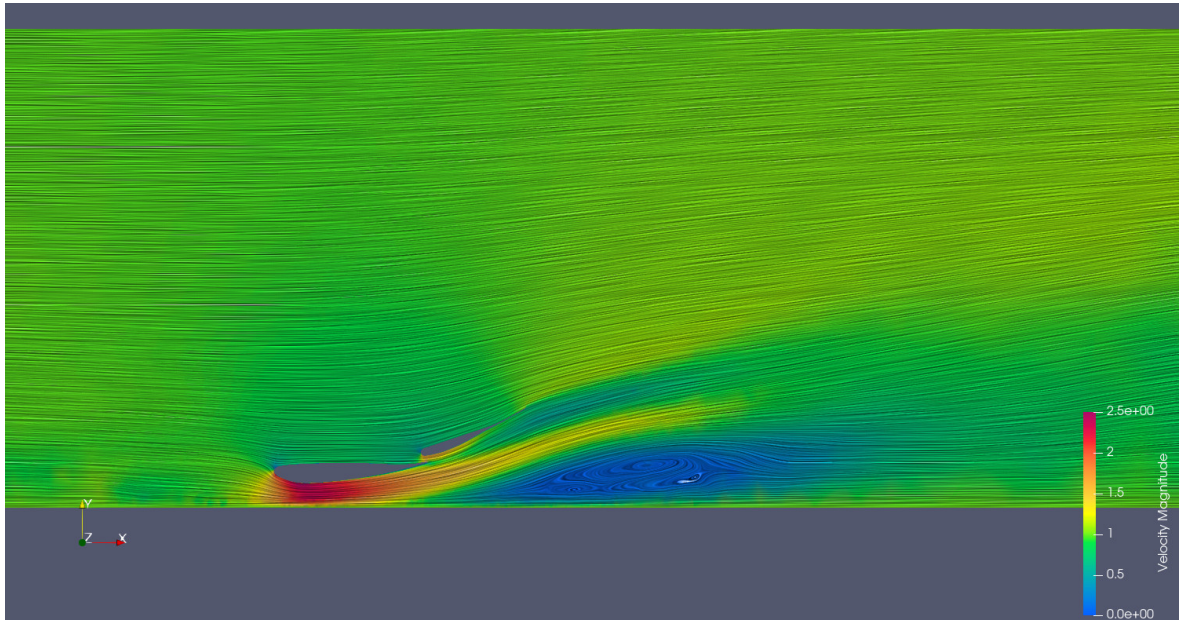


Figura D.4: Visualização de velocidade do escoamento no plano xy do caso h25

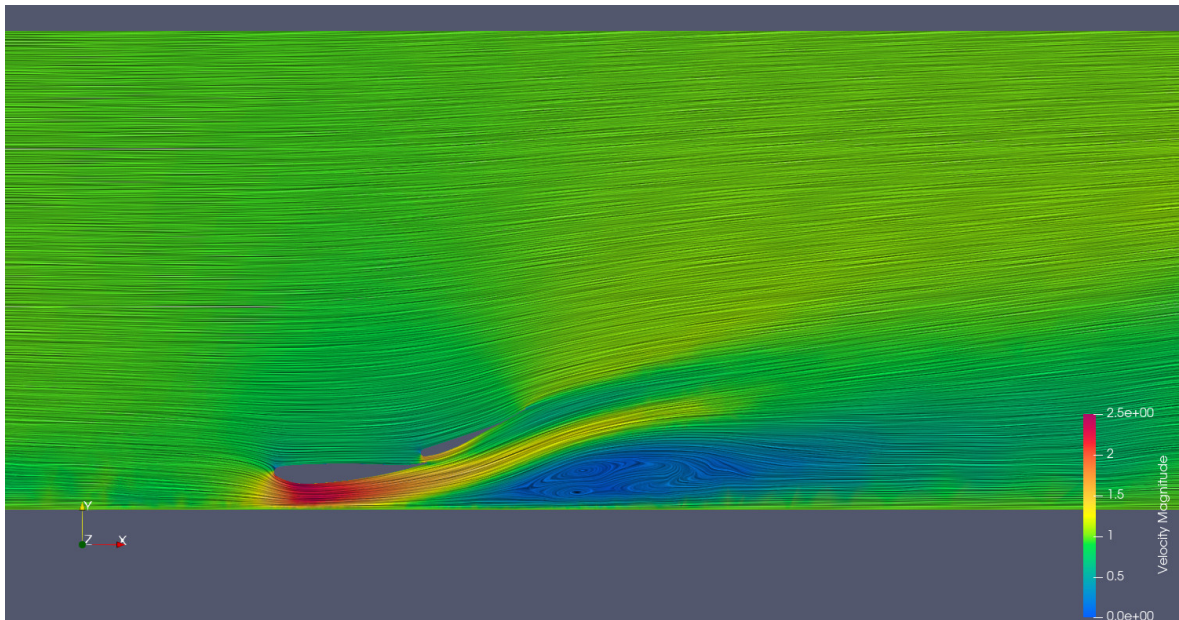


Figura D.5: Visualização de velocidade do escoamento no plano xy do caso h27

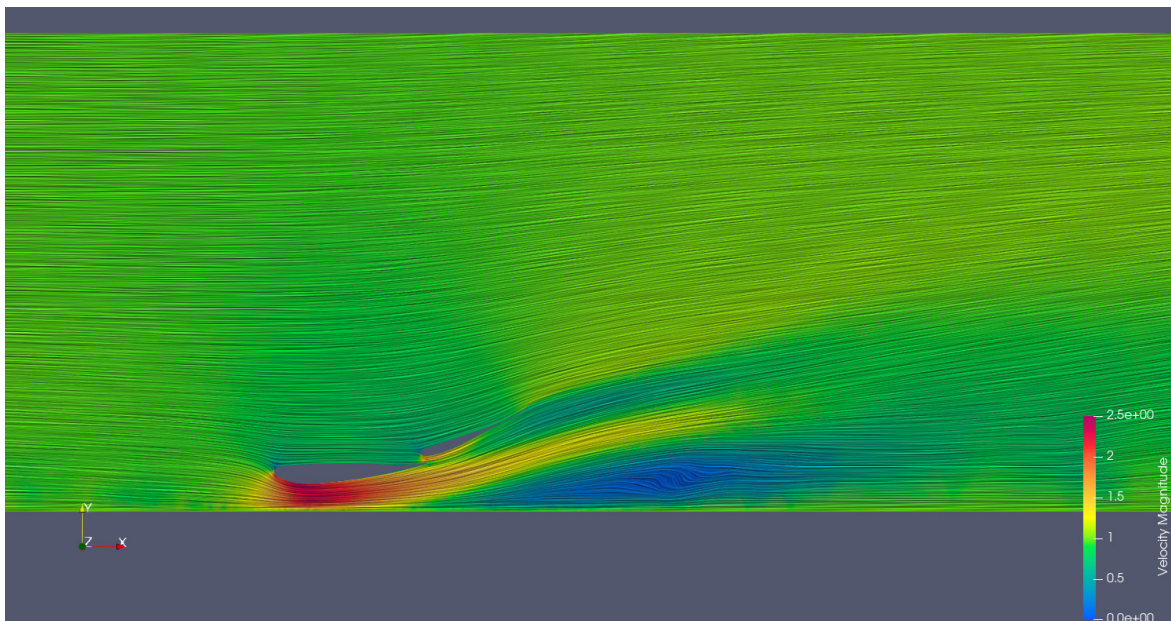


Figura D.6: Visualização de velocidade do escoamento no plano xy do caso h30

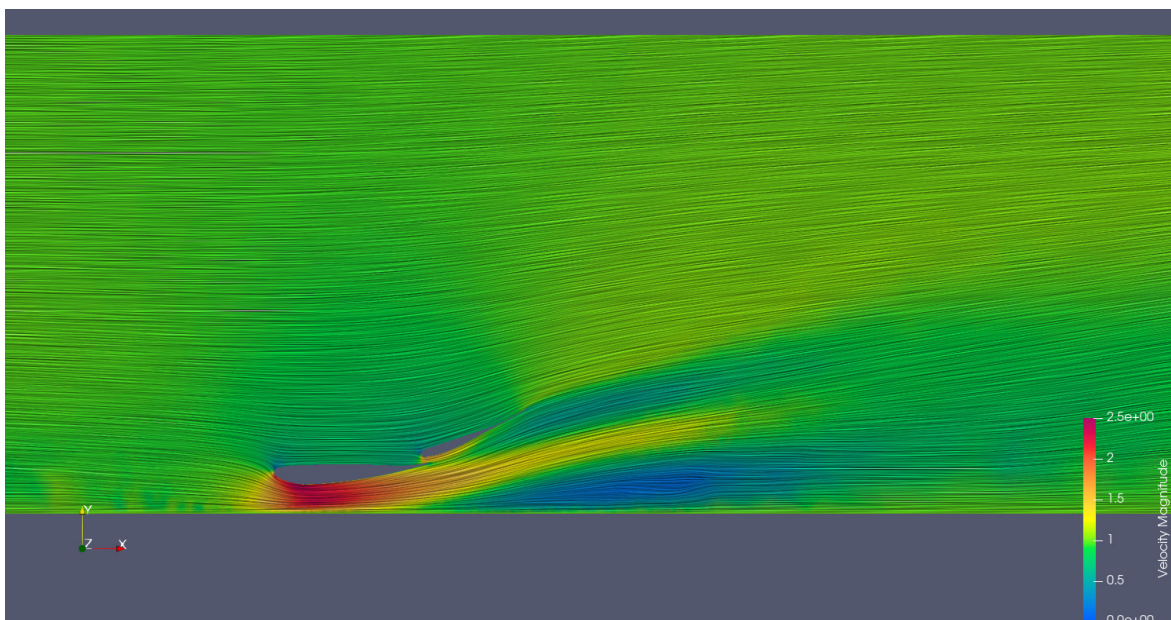


Figura D.7: Visualização de velocidade do escoamento no plano xy do caso h32



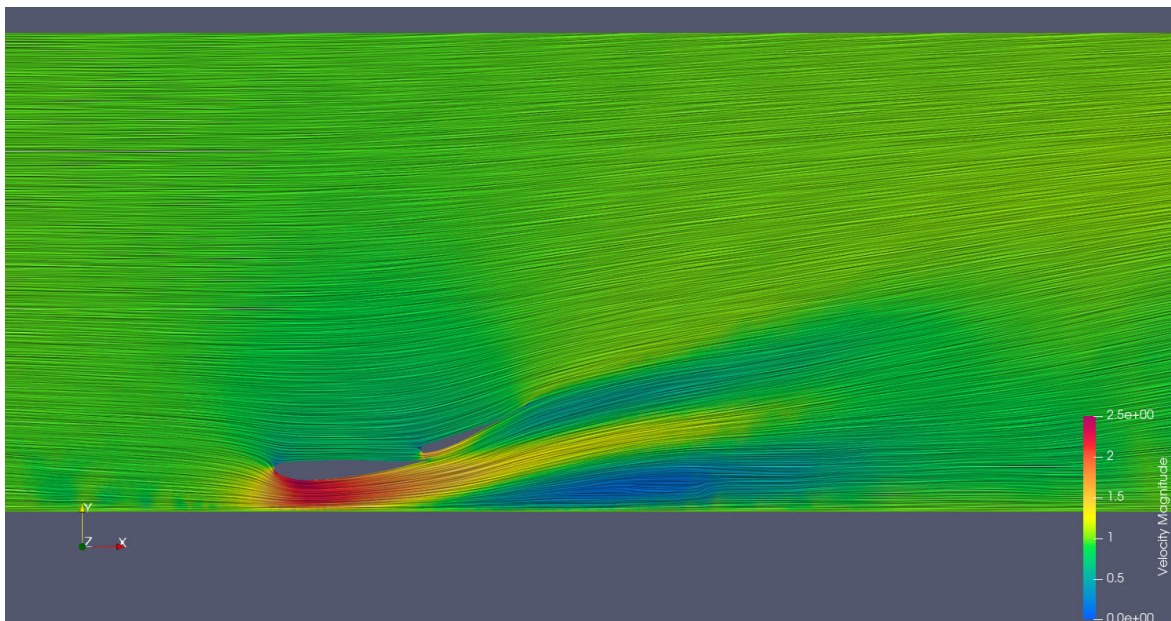


Figura D.8: Visualização de velocidade do escoamento no plano xy do caso h35

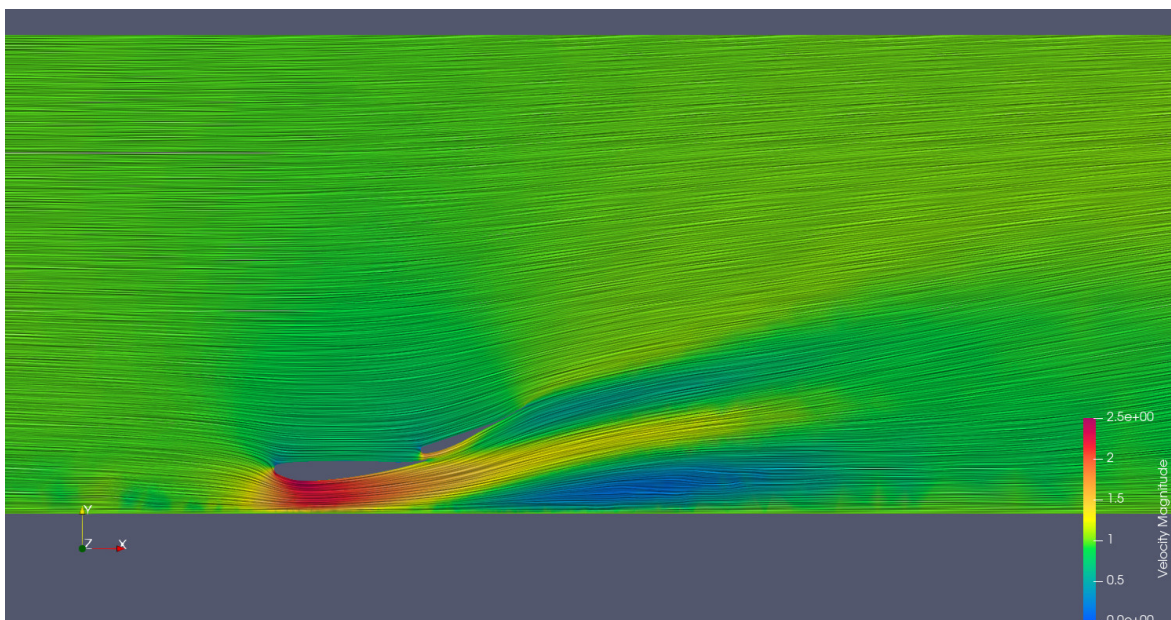


Figura D.9: Visualização de velocidade do escoamento no plano xy do caso h37

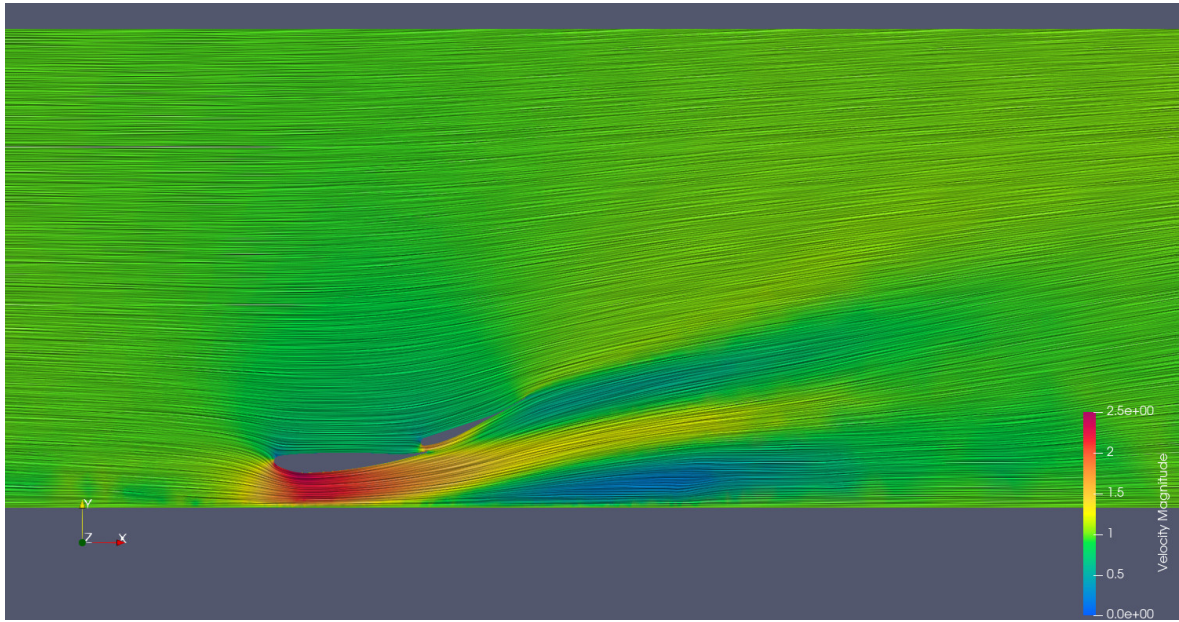


Figura D.10: Visualização de velocidade do escoamento no plano xy do caso h40

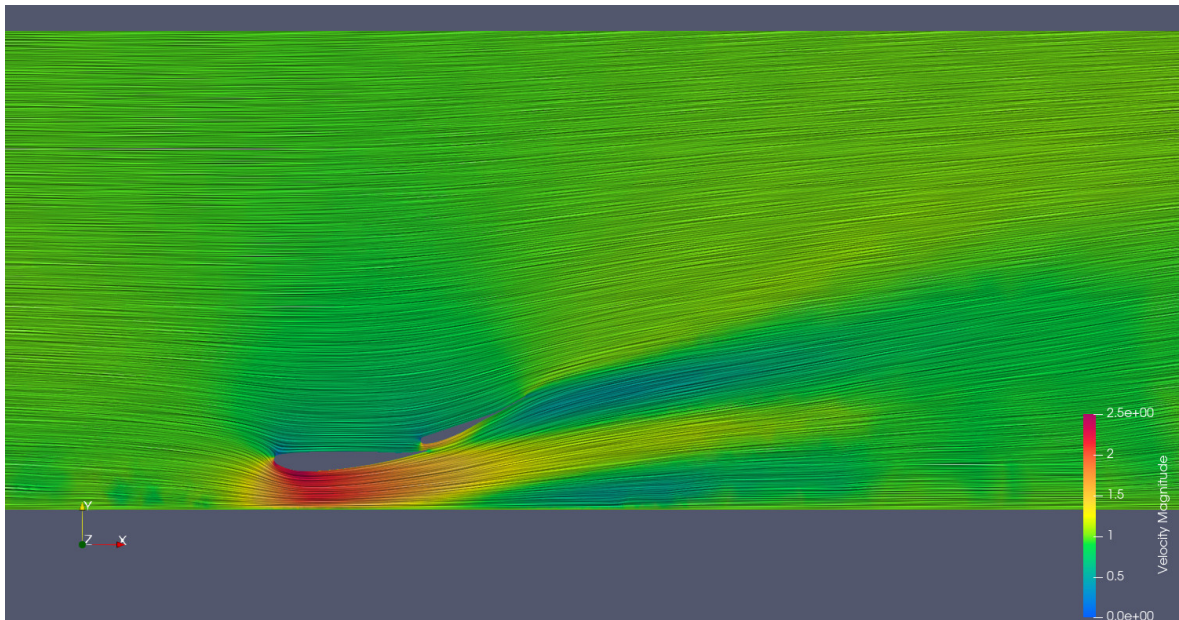


Figura D.11: Visualização de velocidade do escoamento no plano xy do caso h45



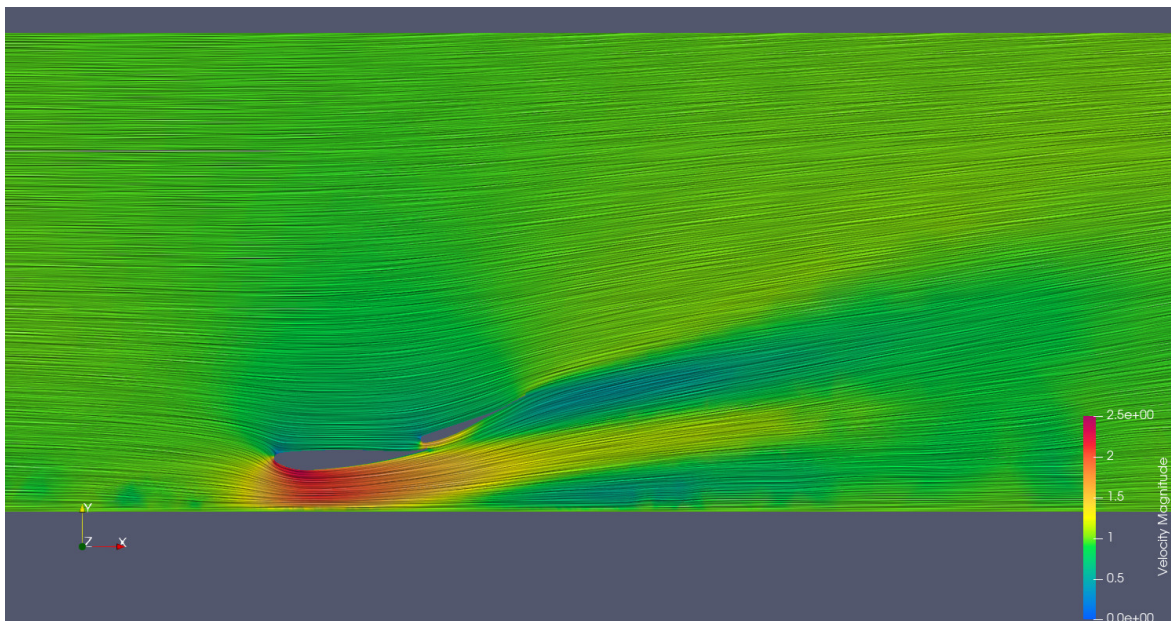


Figura D.12: Visualização de velocidade do escoamento no plano xy do caso h50

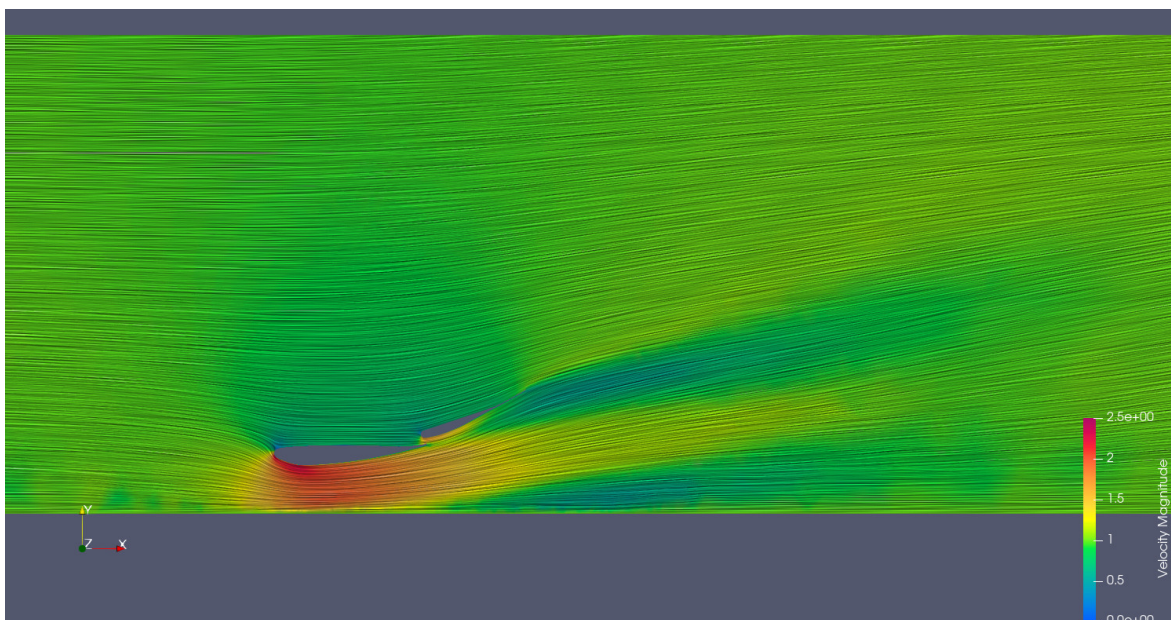


Figura D.13: Visualização de velocidade do escoamento no plano xy do caso h60



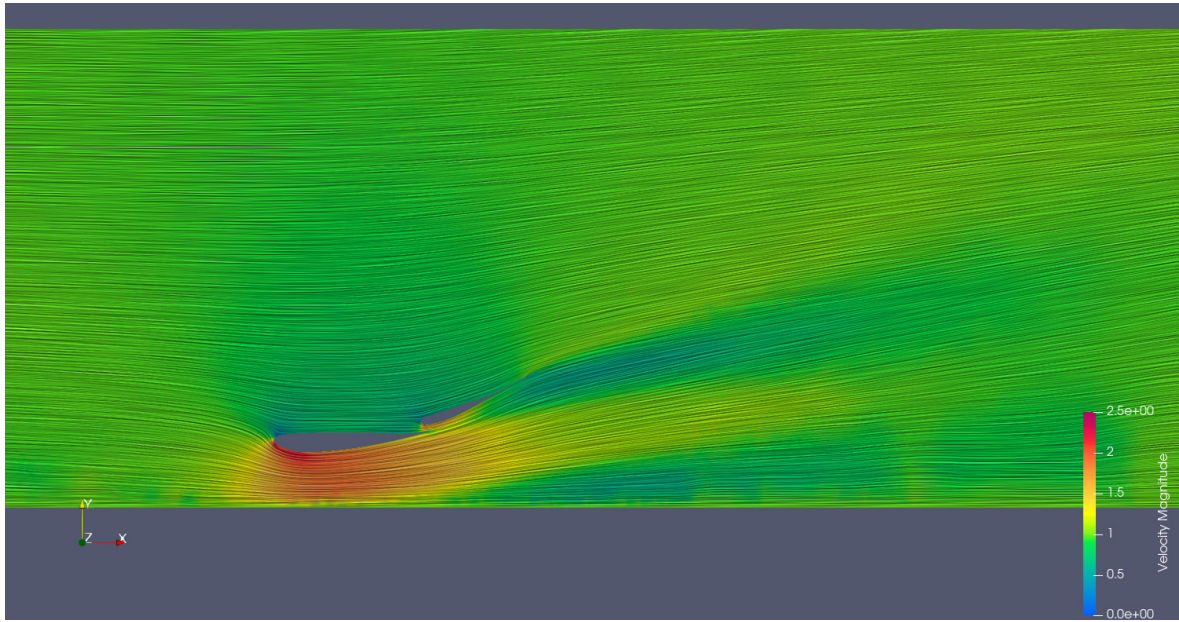


Figura D.14: Visualização de velocidade do escoamento no plano xy do caso h70

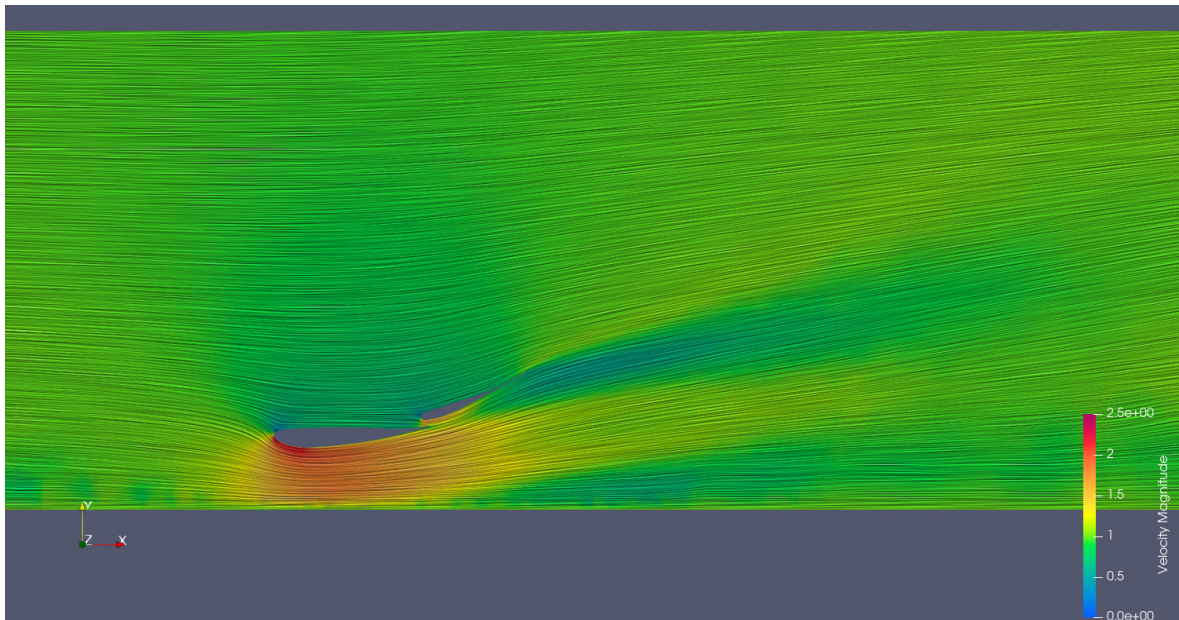


Figura D.15: Visualização de velocidade do escoamento no plano xy do caso h80

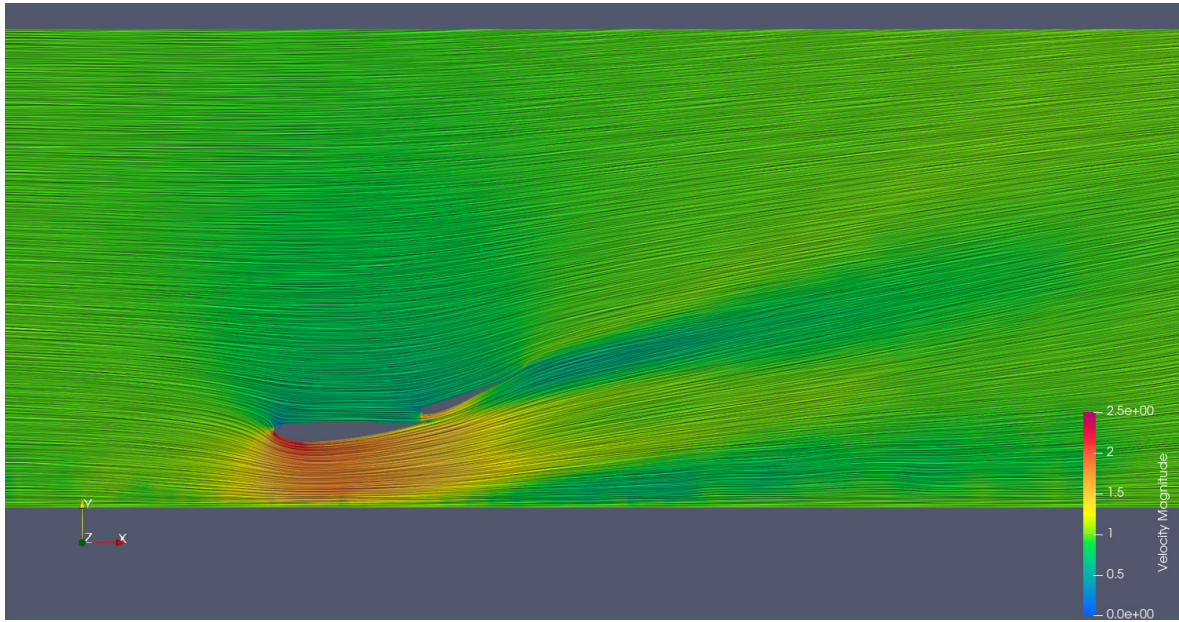


Figura D.16: Visualização de velocidade do escoamento no plano xy do caso h85

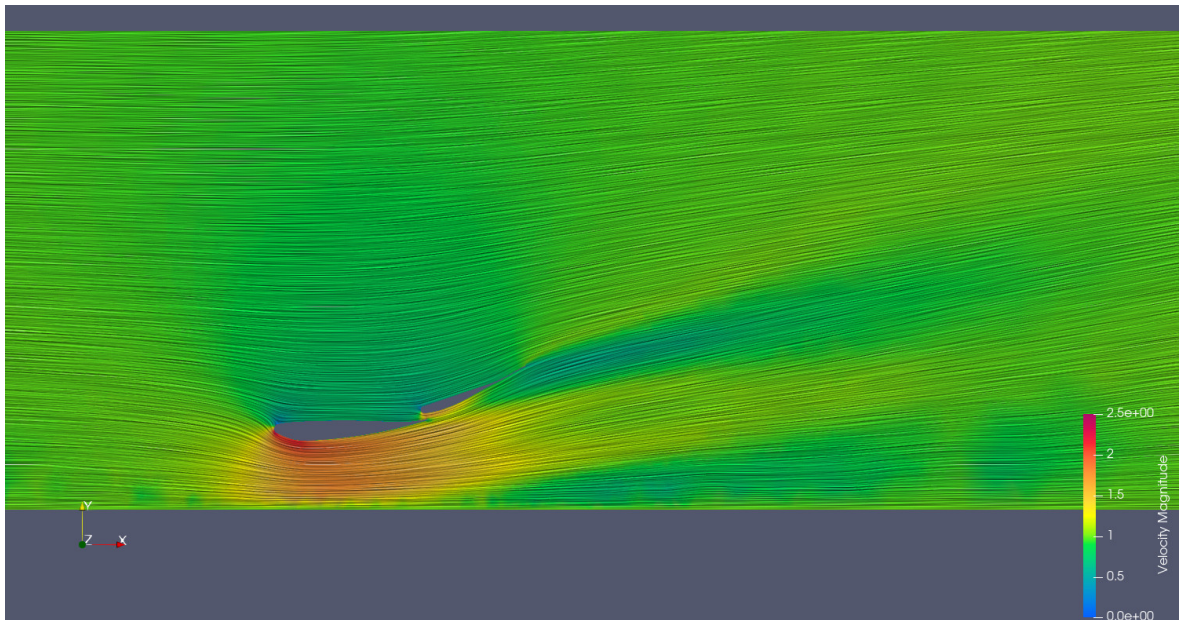


Figura D.17: Visualização de velocidade do escoamento no plano xy do caso h90



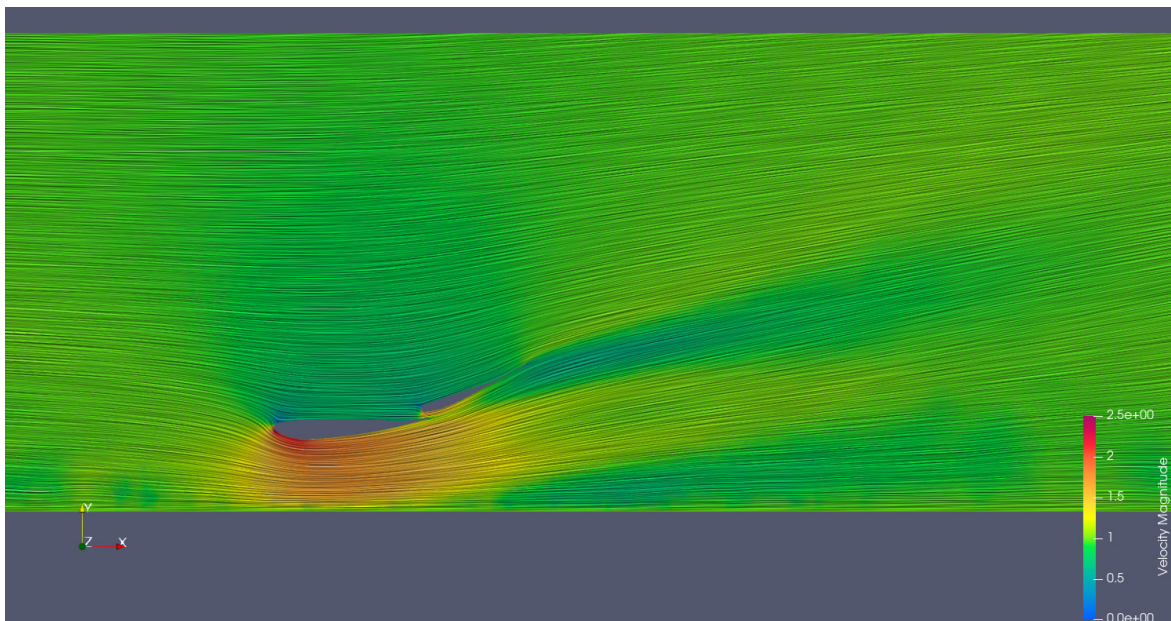


Figura D.18: Visualização de velocidade do escoamento no plano xy do caso h95

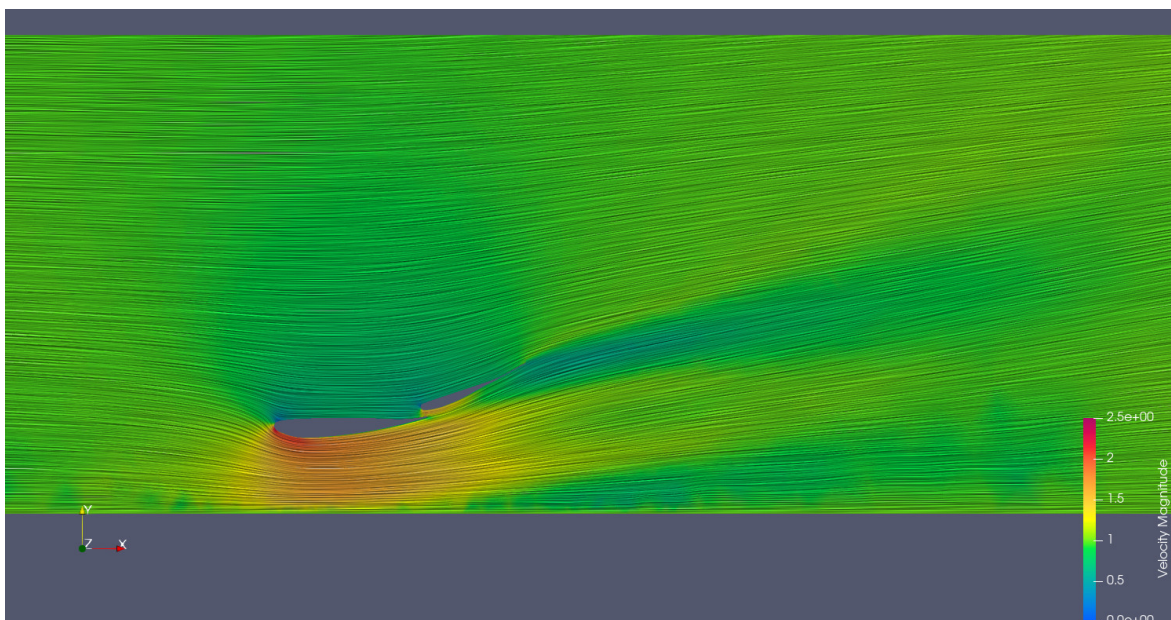


Figura D.19: Visualização de velocidade do escoamento no plano xy do caso h100

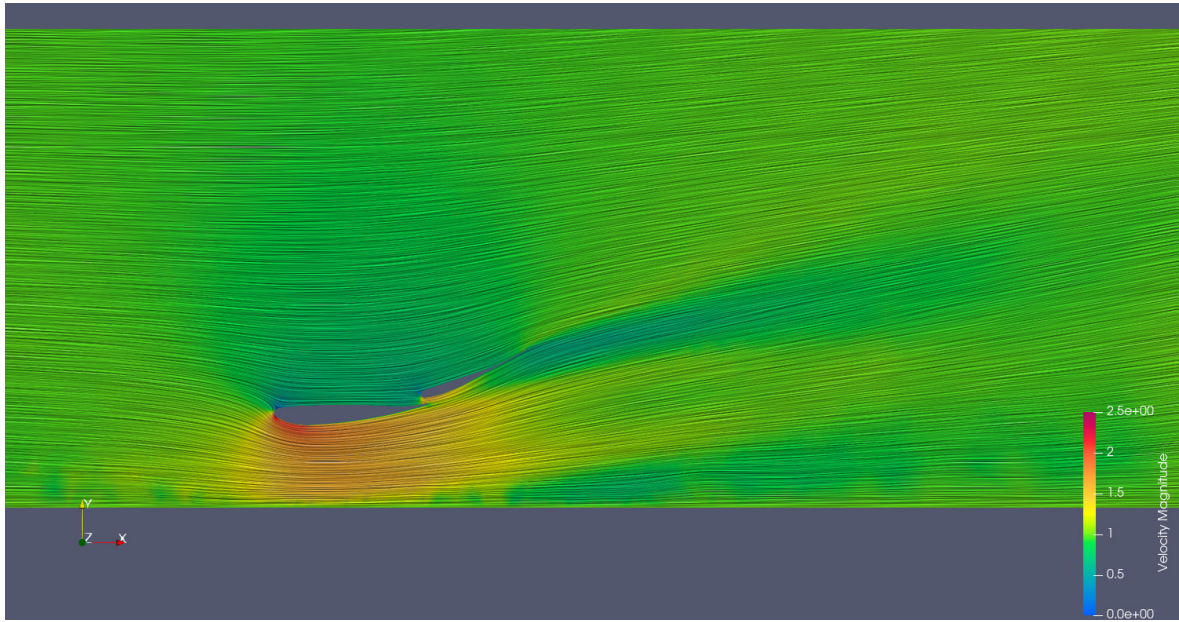


Figura D.20: Visualização de velocidade do escoamento no plano xy do caso h110

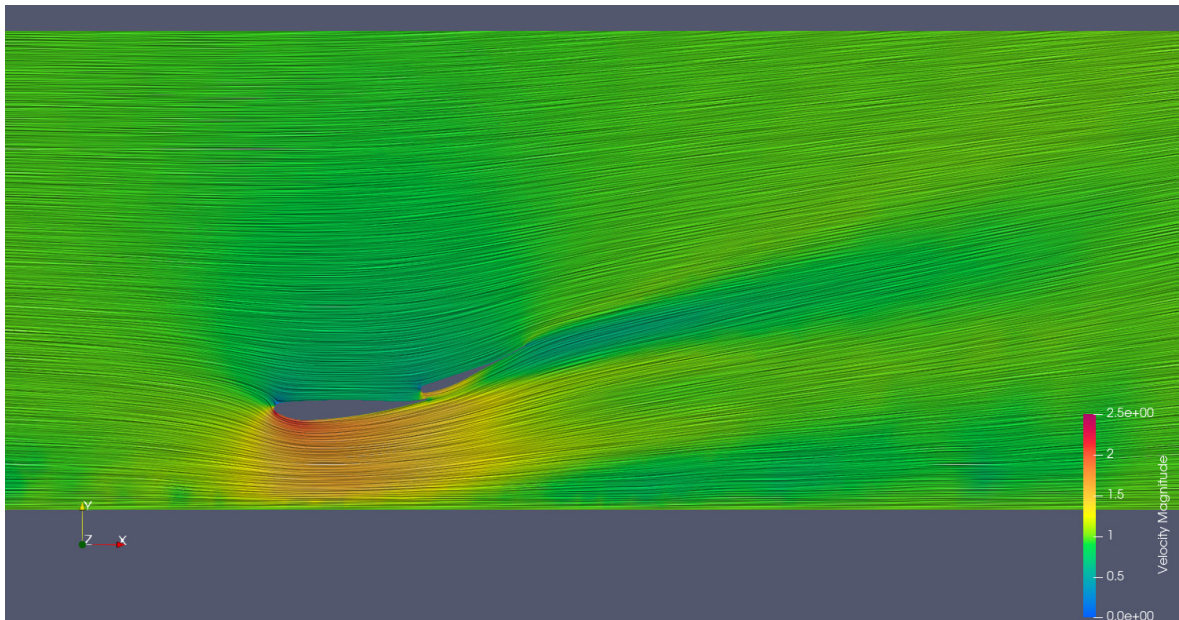


Figura D.21: Visualização de velocidade do escoamento no plano xy do caso h120



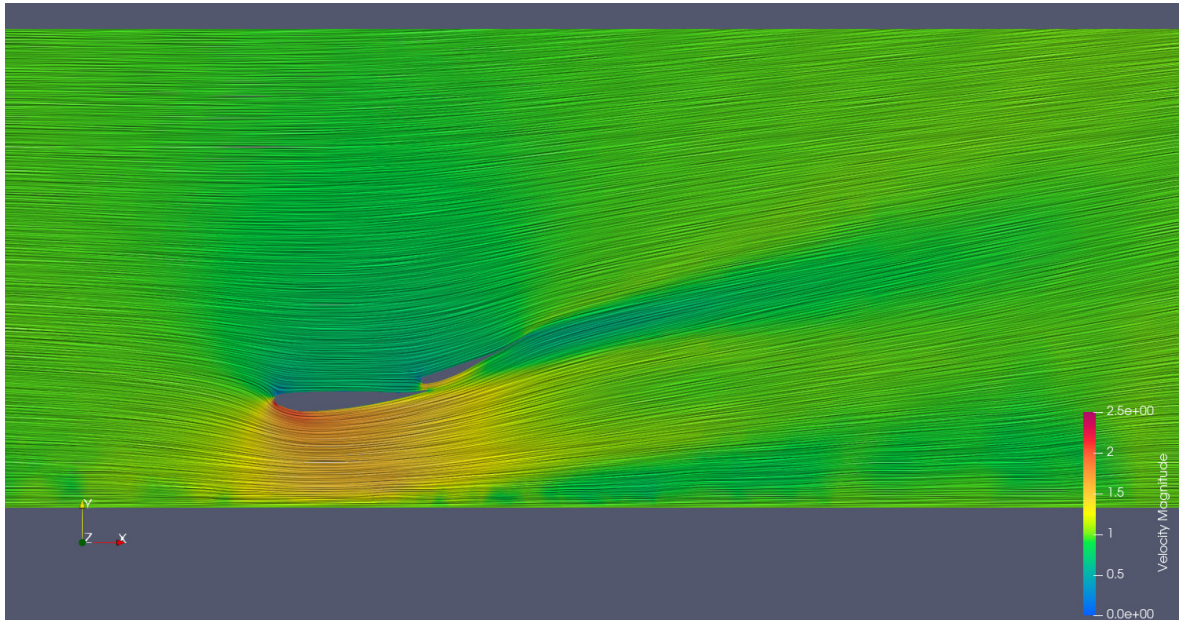


Figura D.22: Visualização de velocidade do escoamento no plano xy do caso h130

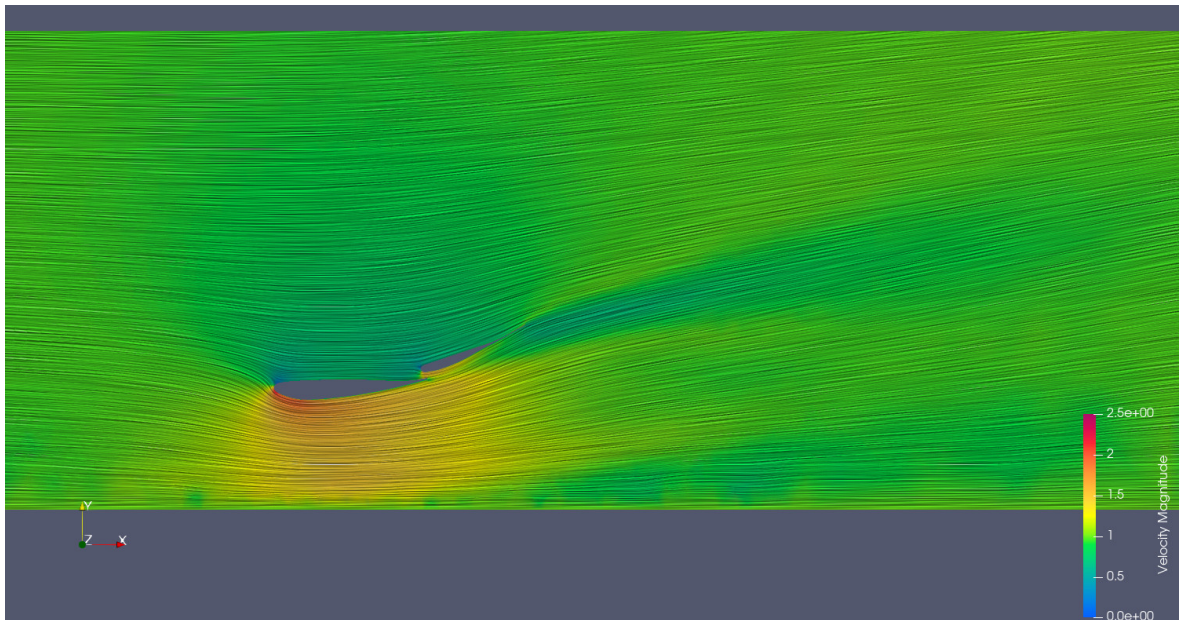


Figura D.23: Visualização de velocidade do escoamento no plano xy do caso h150

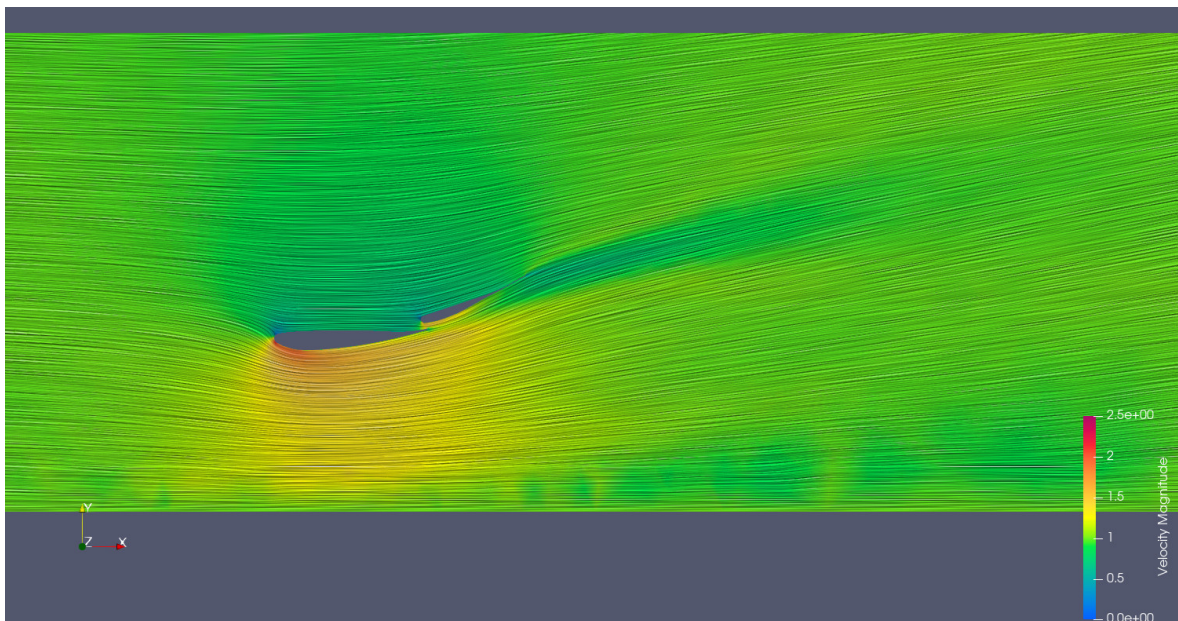


Figura D.24: Visualização de velocidade do escoamento no plano xy do caso h225



# APÊNDICE E

## Visualizações de Velocidade - Plano yz - Visão Geral

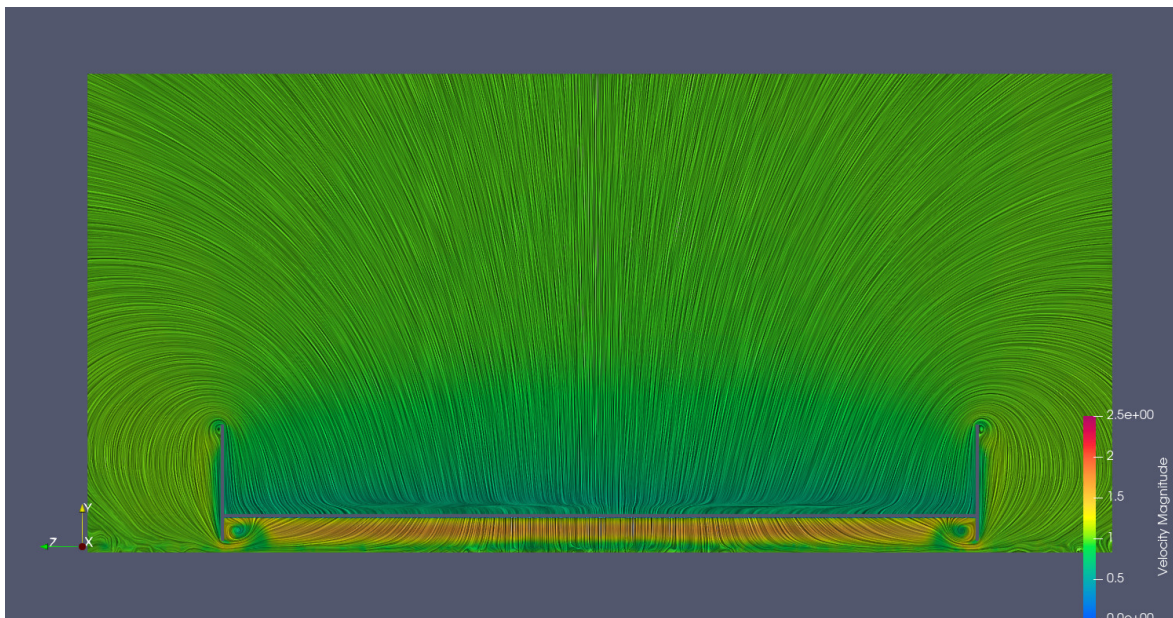


Figura E.1: Visualização de velocidade do escoamento no plano yz do caso h17

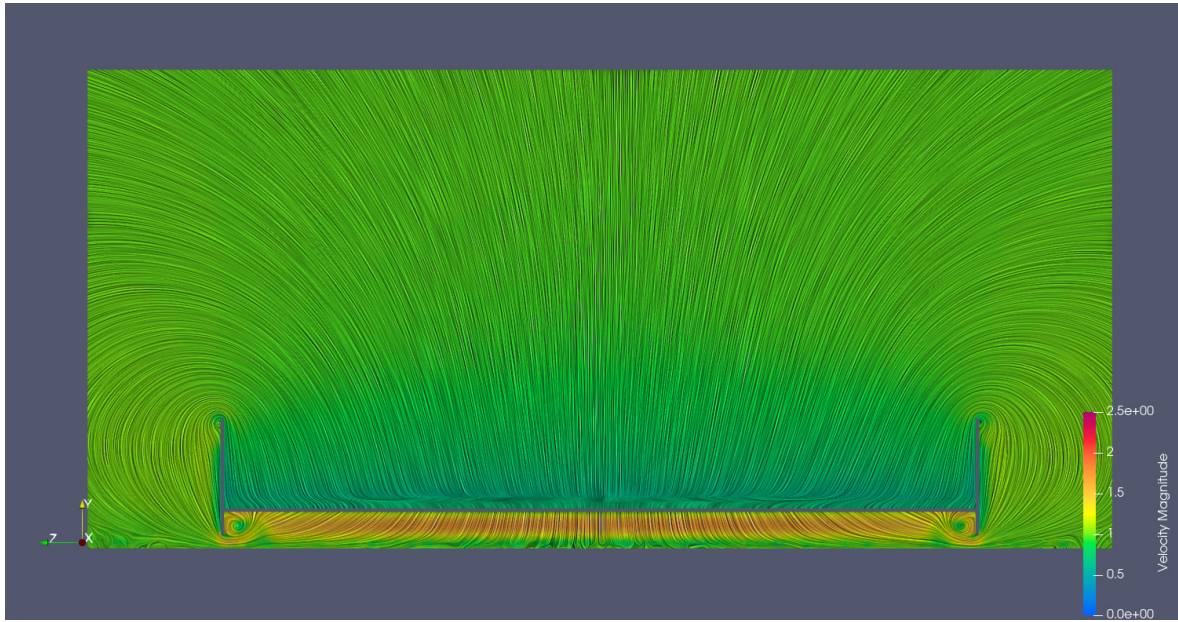


Figura E.2: Visualização de velocidade do escoamento no plano yz do caso h20

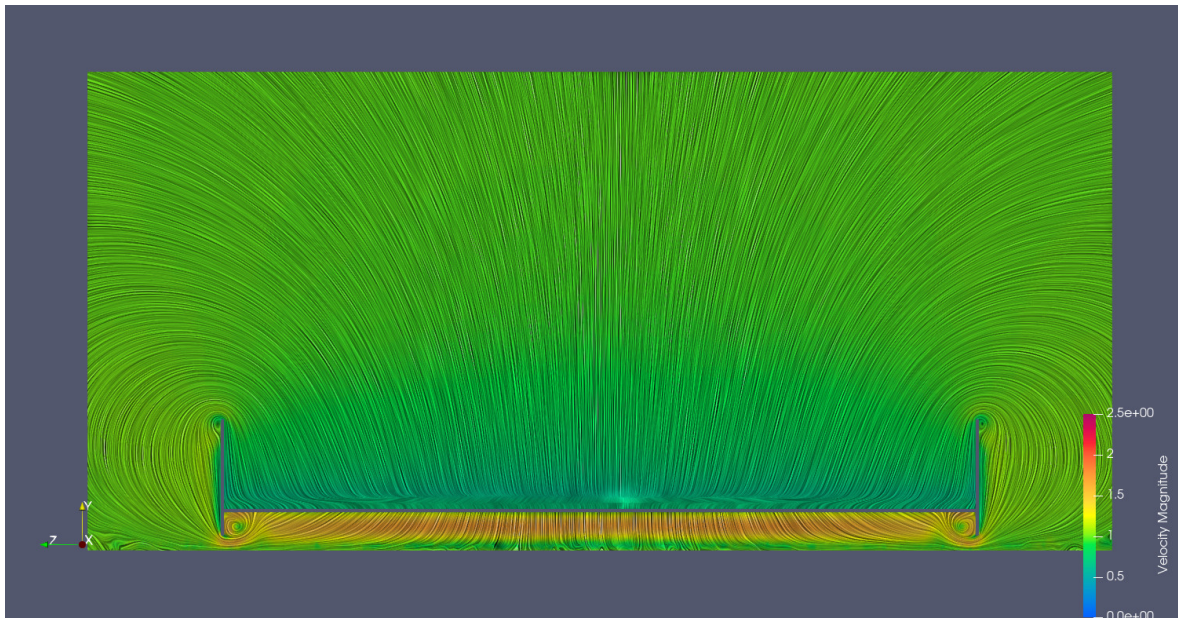


Figura E.3: Visualização de velocidade do escoamento no plano yz do caso h22



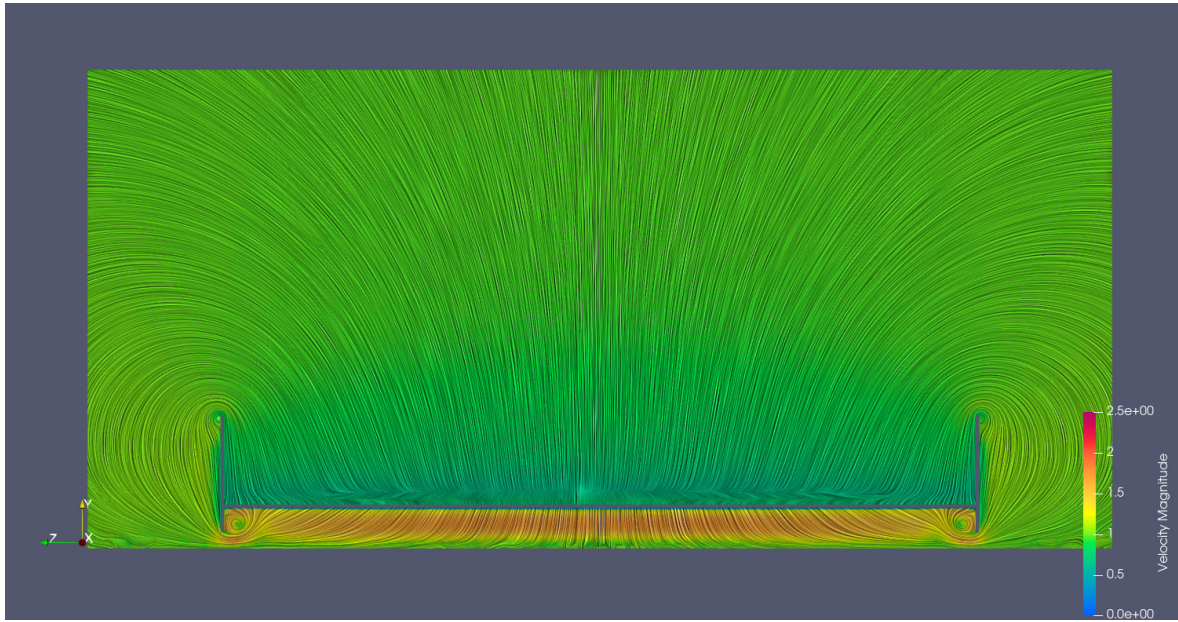


Figura E.4: Visualização de velocidade do escoamento no plano yz do caso h25

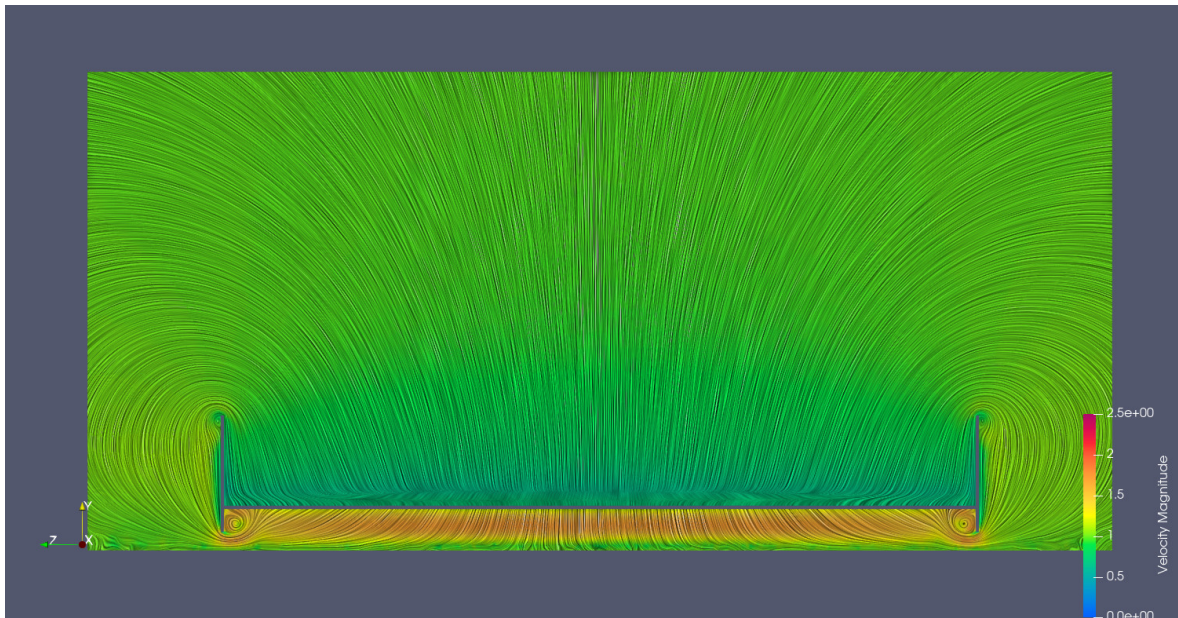


Figura E.5: Visualização de velocidade do escoamento no plano yz do caso h27



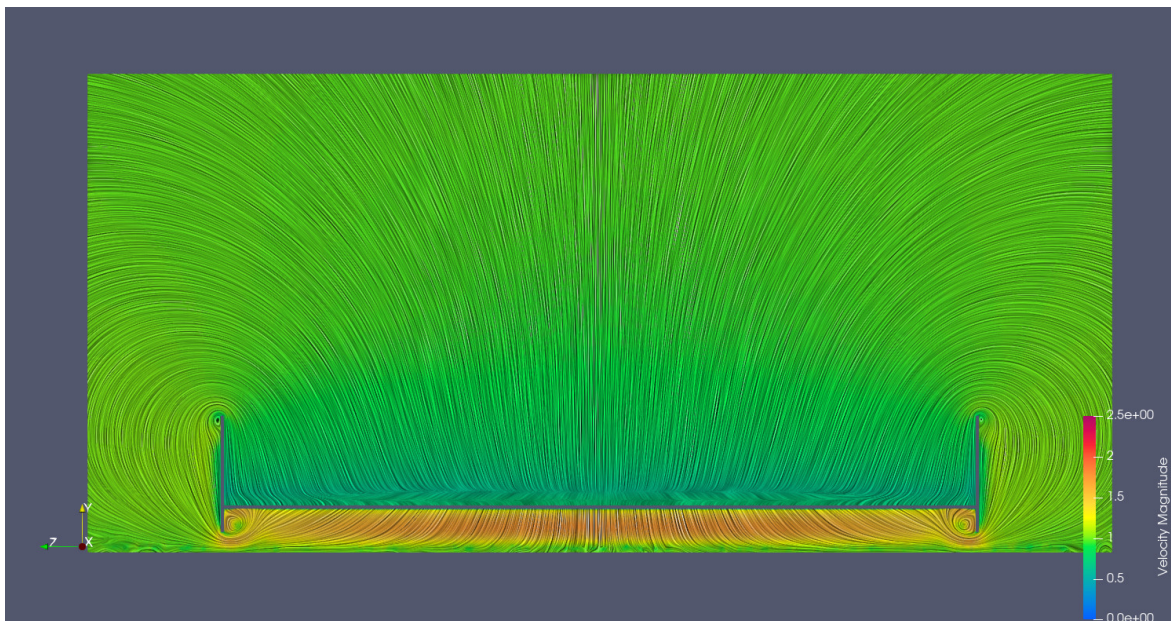


Figura E.6: Visualização de velocidade do escoamento no plano yz do caso h30

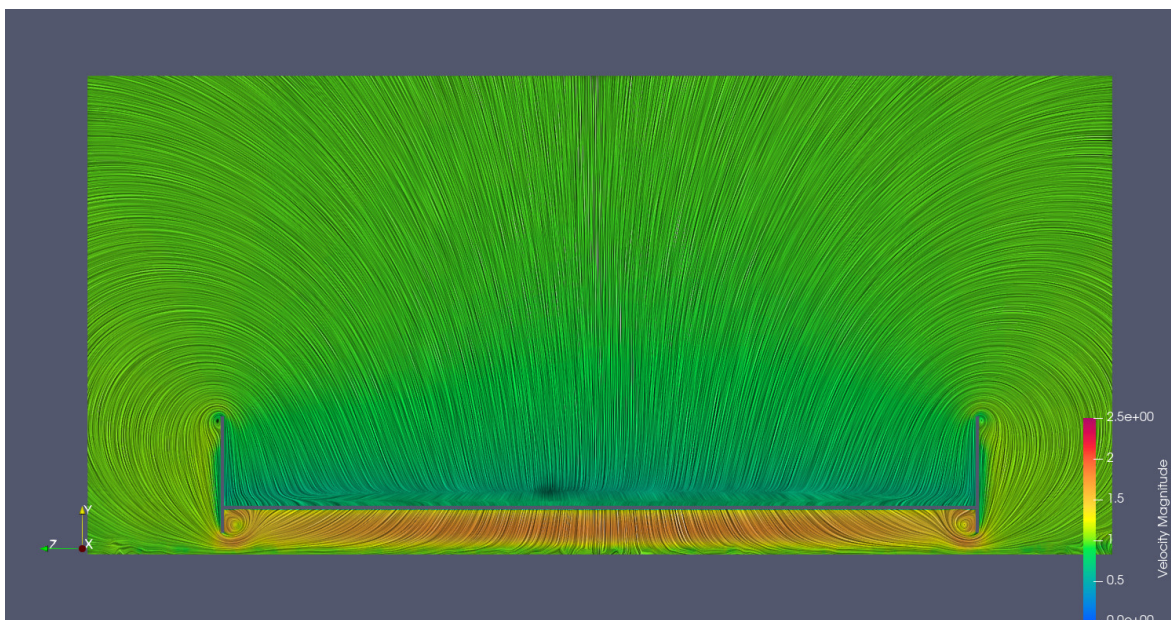


Figura E.7: Visualização de velocidade do escoamento no plano yz do caso h32



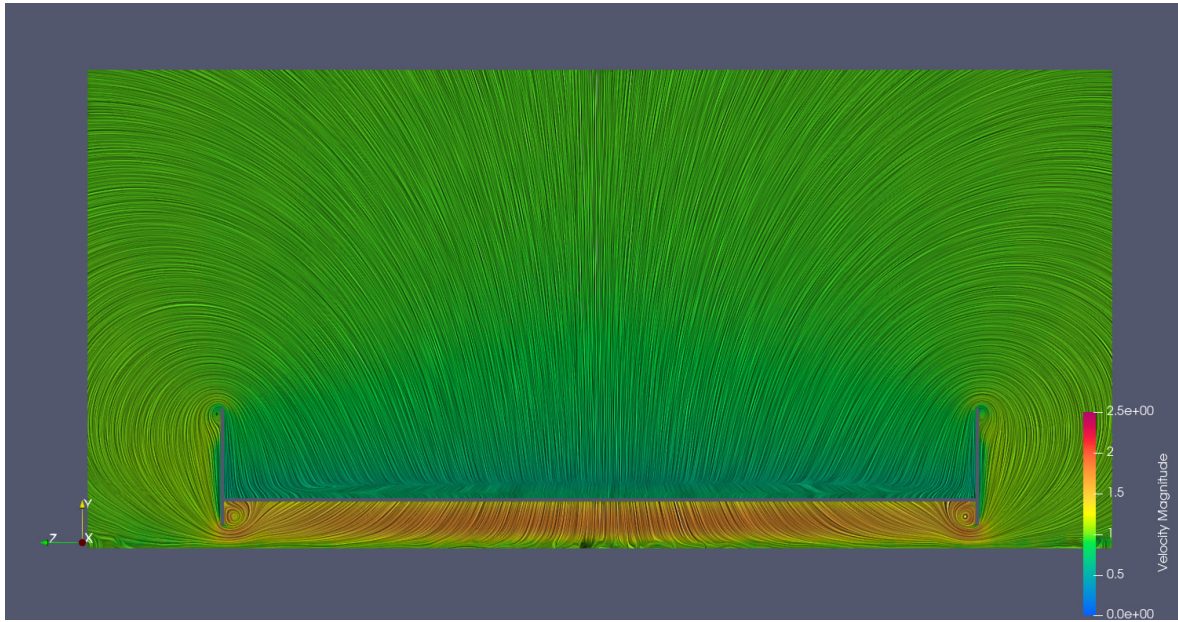


Figura E.8: Visualização de velocidade do escoamento no plano yz do caso h35

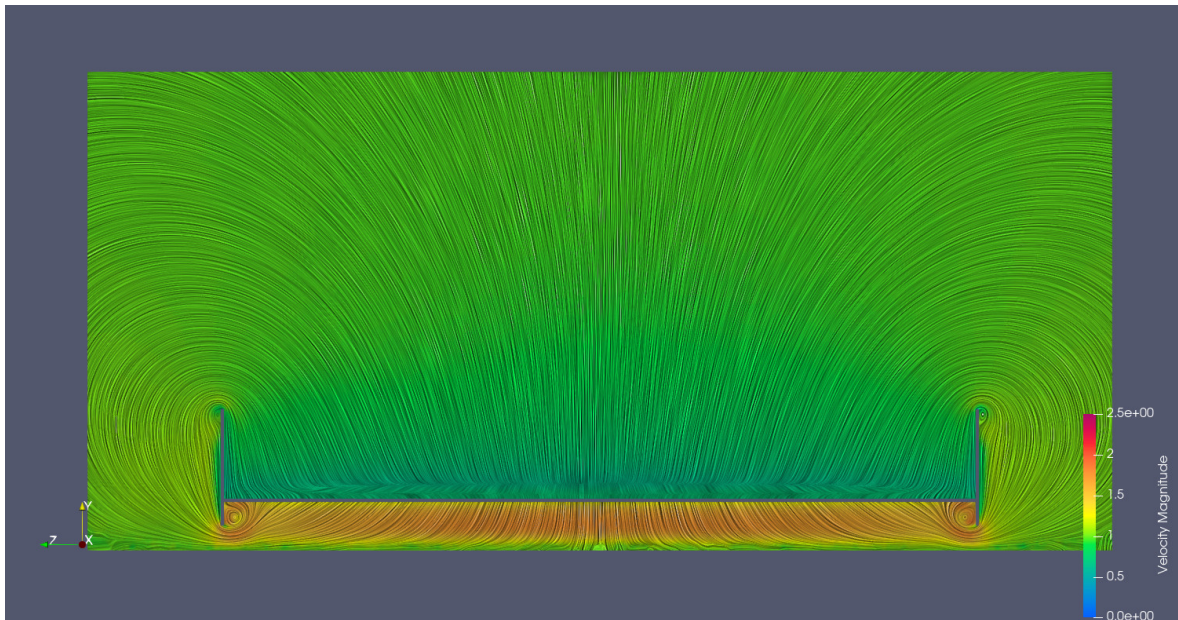


Figura E.9: Visualização de velocidade do escoamento no plano yz do caso h37



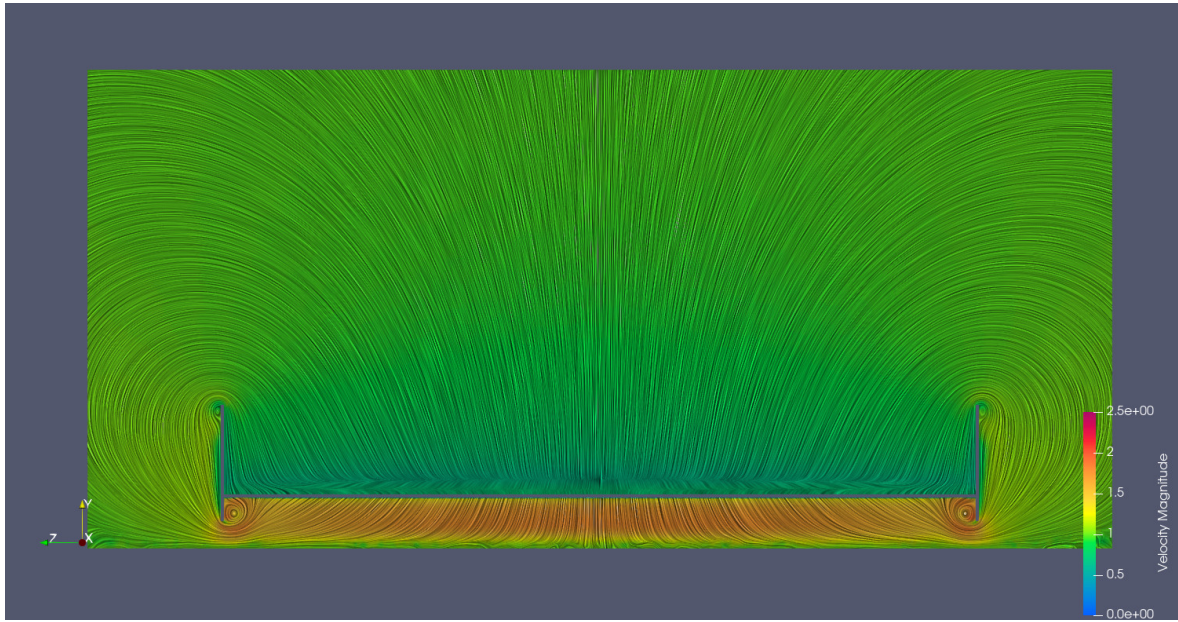


Figura E.10: Visualização de velocidade do escoamento no plano yz do caso h40

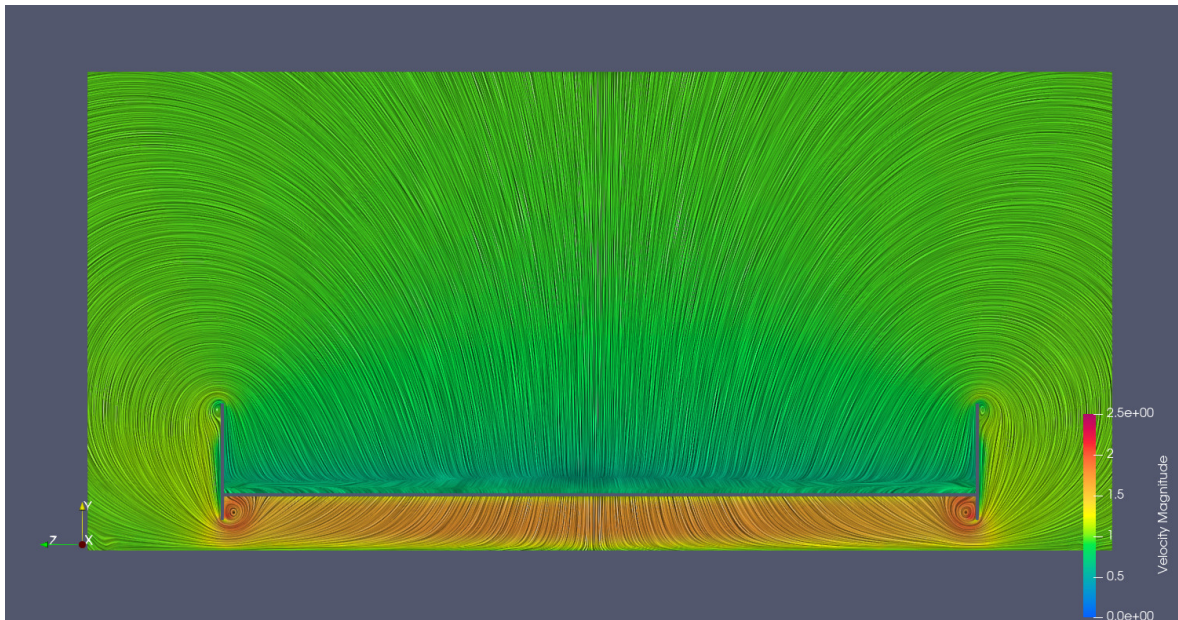


Figura E.11: Visualização de velocidade do escoamento no plano yz do caso h45



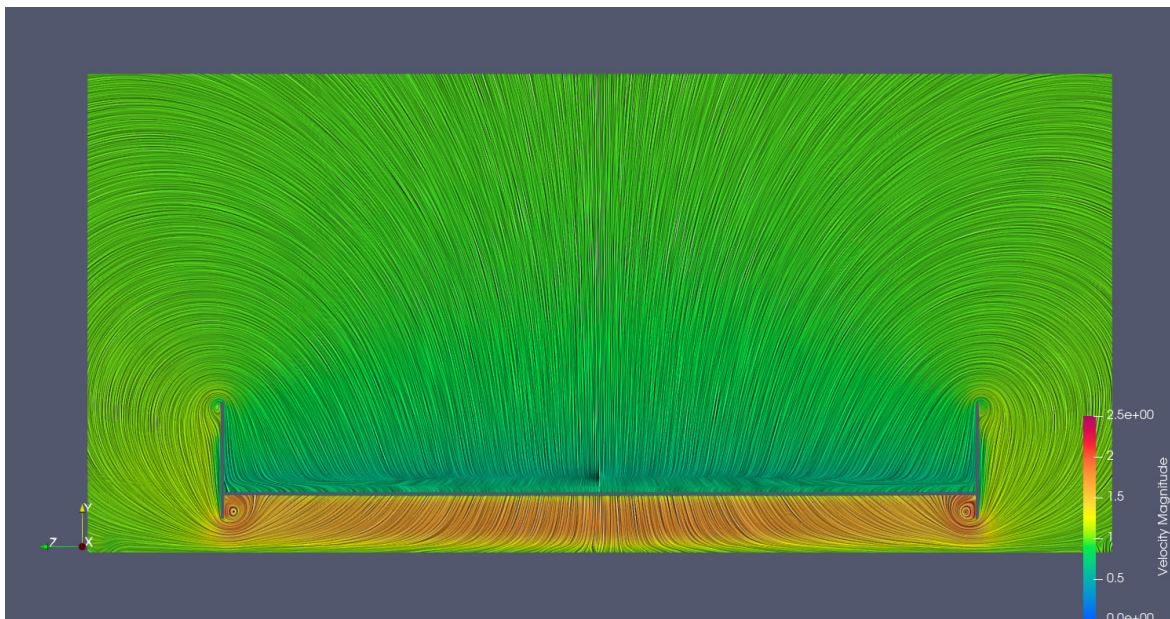


Figura E.12: Visualização de velocidade do escoamento no plano yz do caso h50

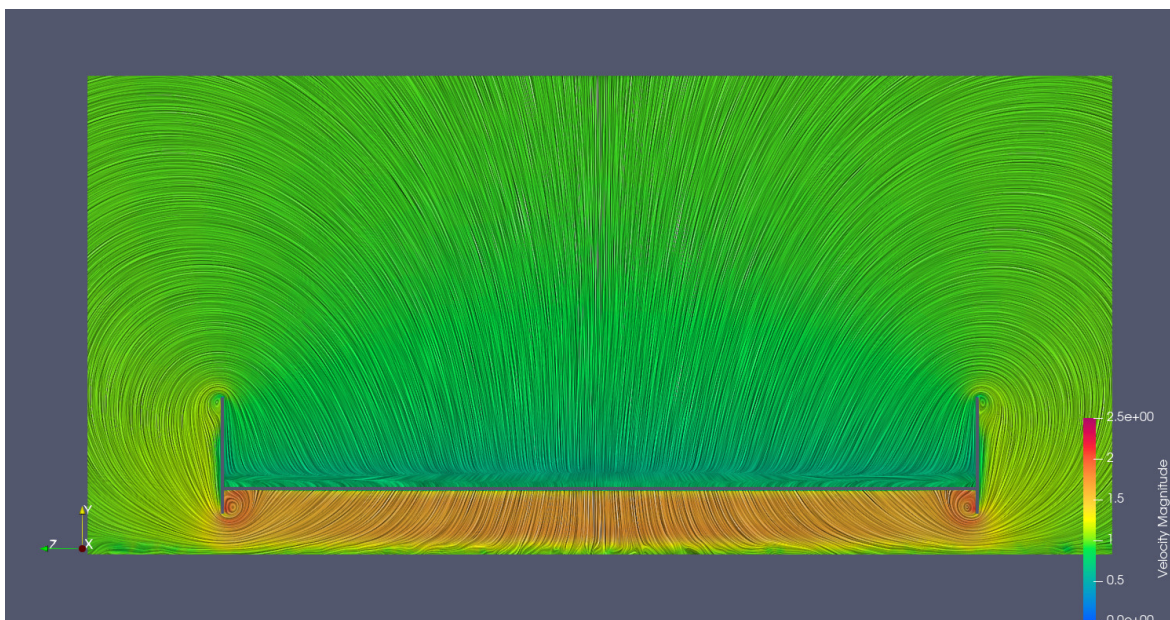


Figura E.13: Visualização de velocidade do escoamento no plano yz do caso h60



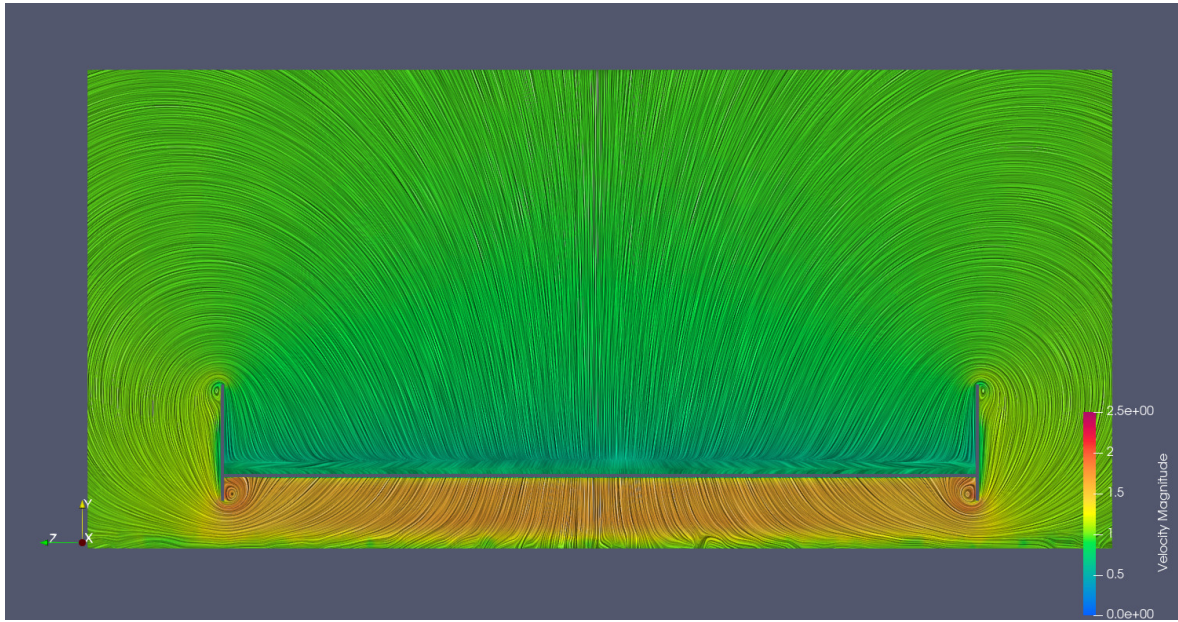


Figura E.14: Visualização de velocidade do escoamento no plano yz do caso h70

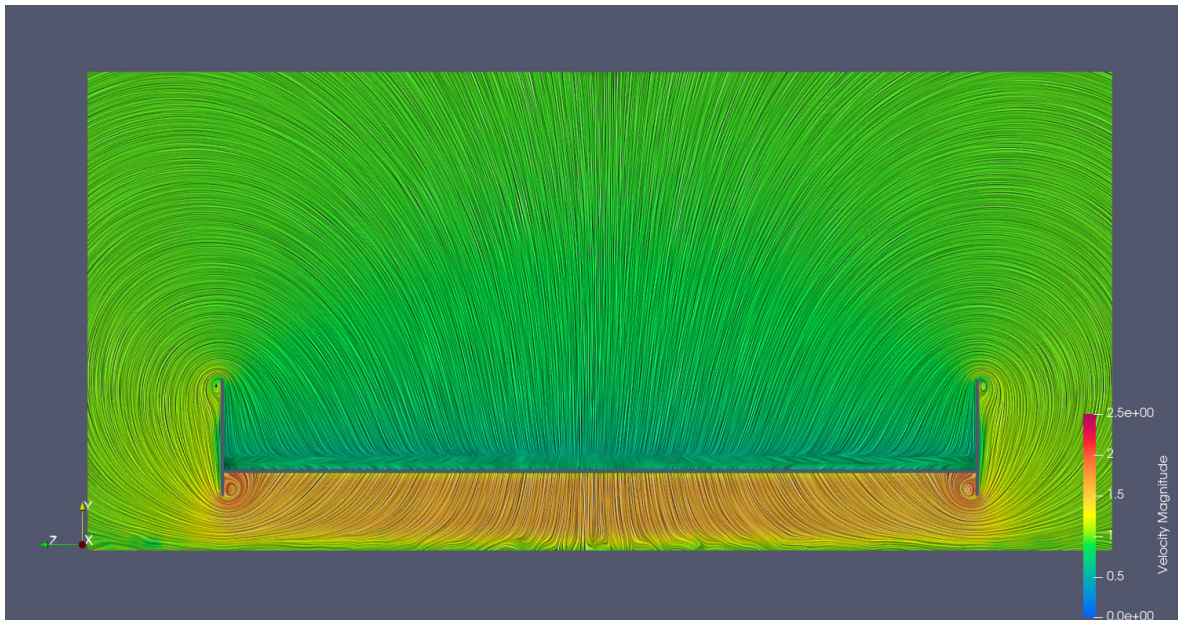


Figura E.15: Visualização de velocidade do escoamento no plano yz do caso h80



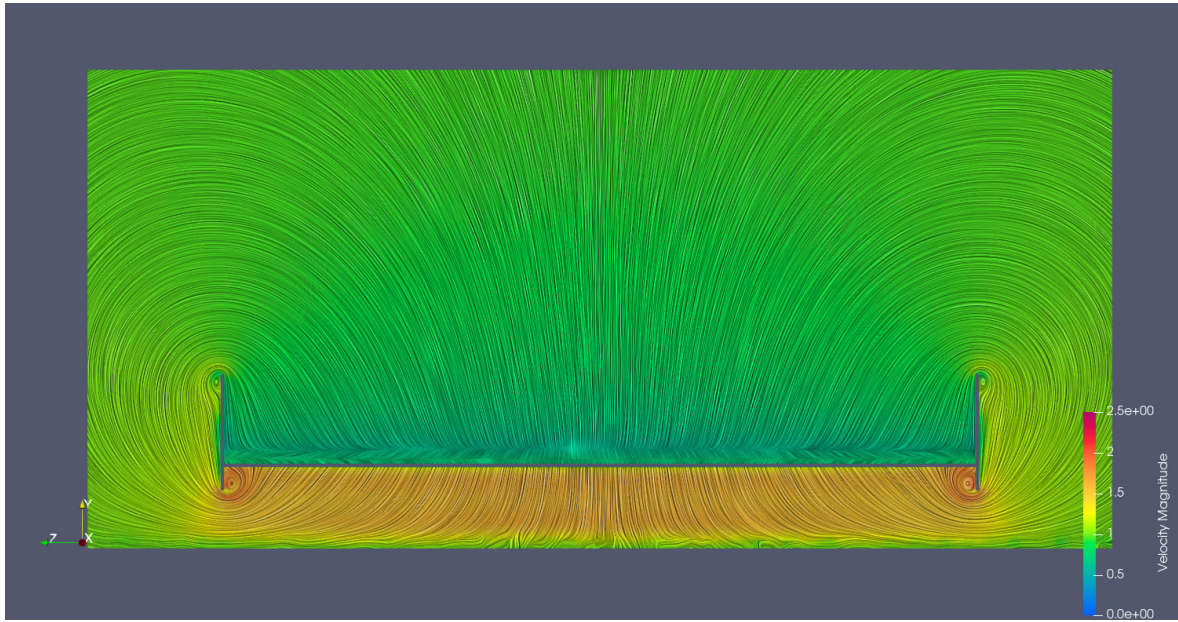


Figura E.16: Visualização de velocidade do escoamento no plano yz do caso h85

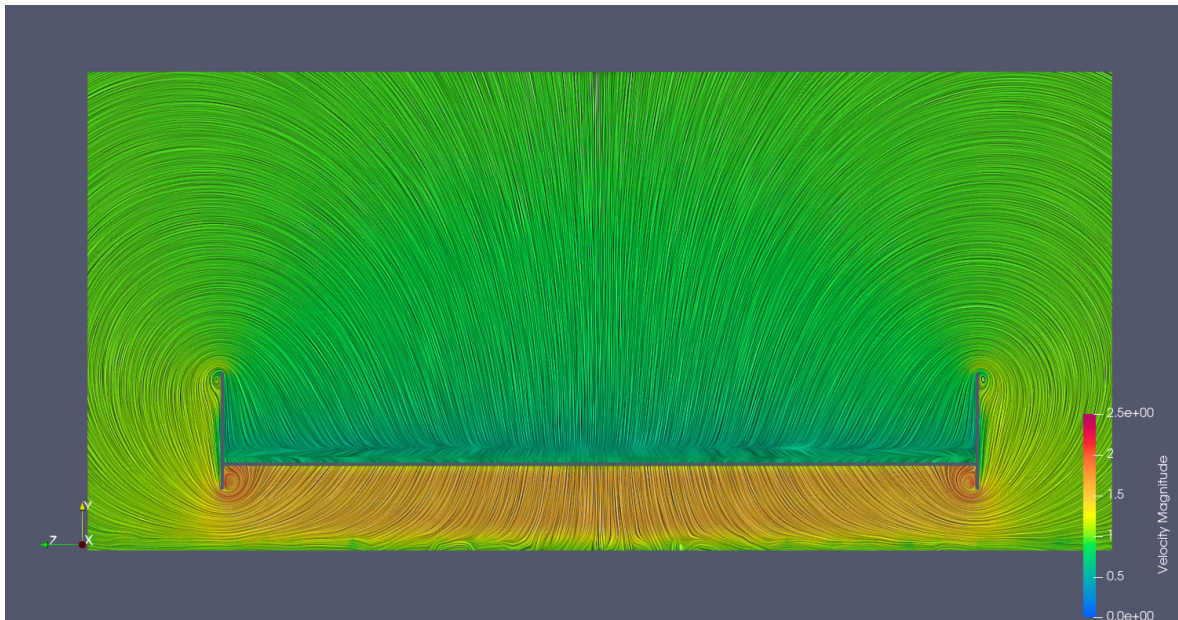


Figura E.17: Visualização de velocidade do escoamento no plano yz do caso h90



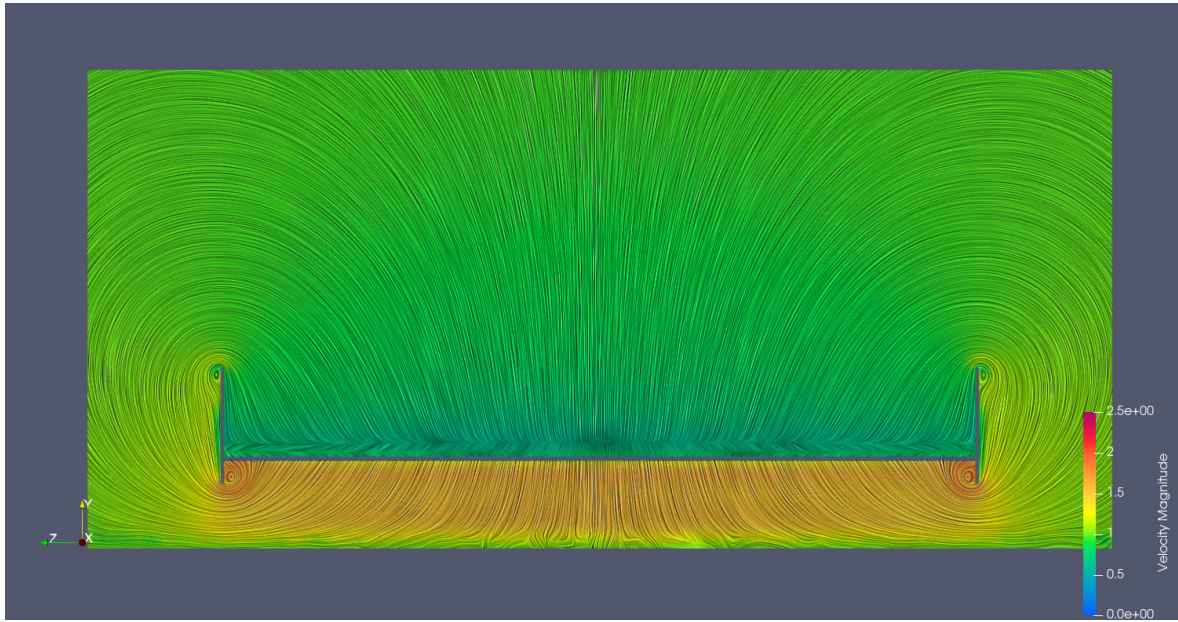


Figura E.18: Visualização de velocidade do escoamento no plano yz do caso h95

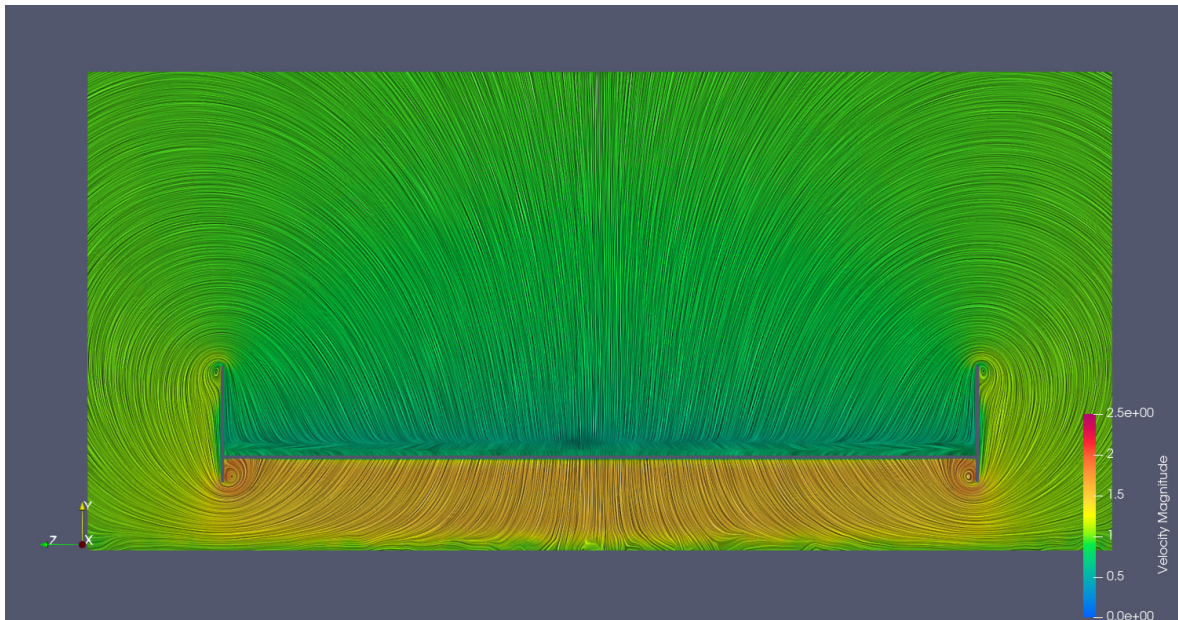


Figura E.19: Visualização de velocidade do escoamento no plano yz do caso h100



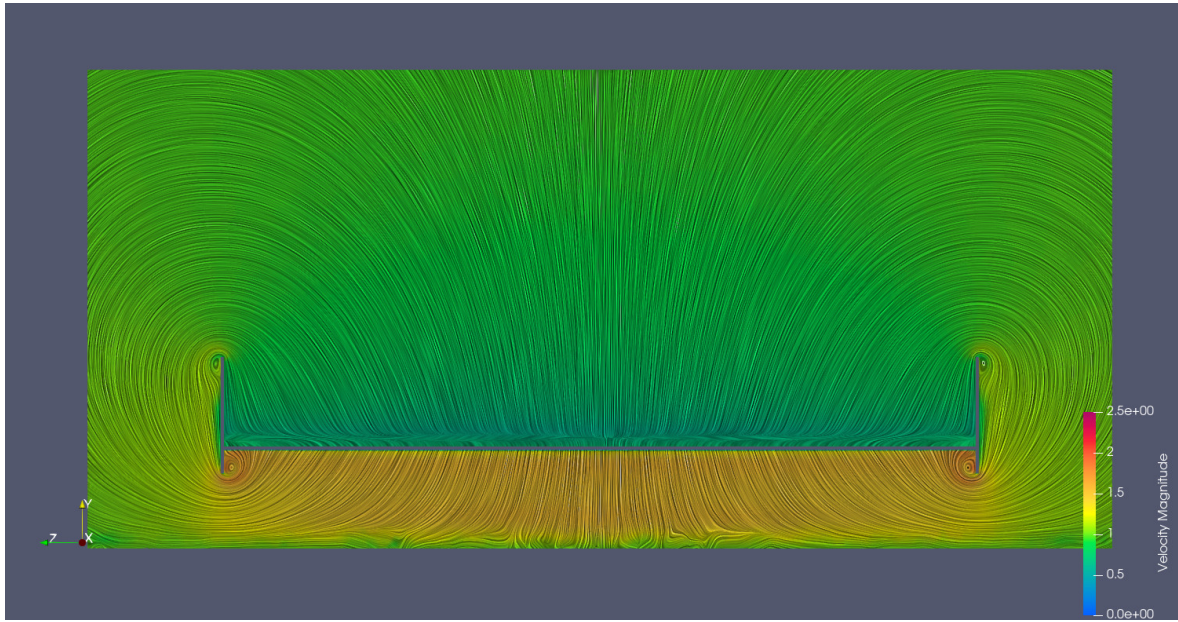


Figura E.20: Visualização de velocidade do escoamento no plano yz do caso h110

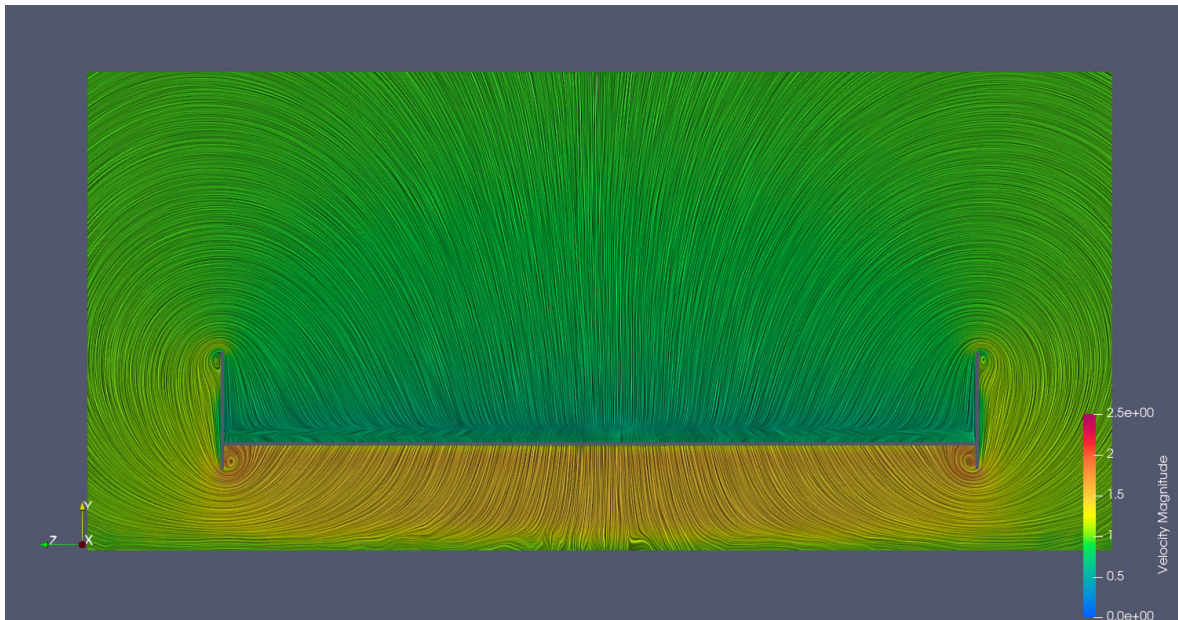


Figura E.21: Visualização de velocidade do escoamento no plano yz do caso h120

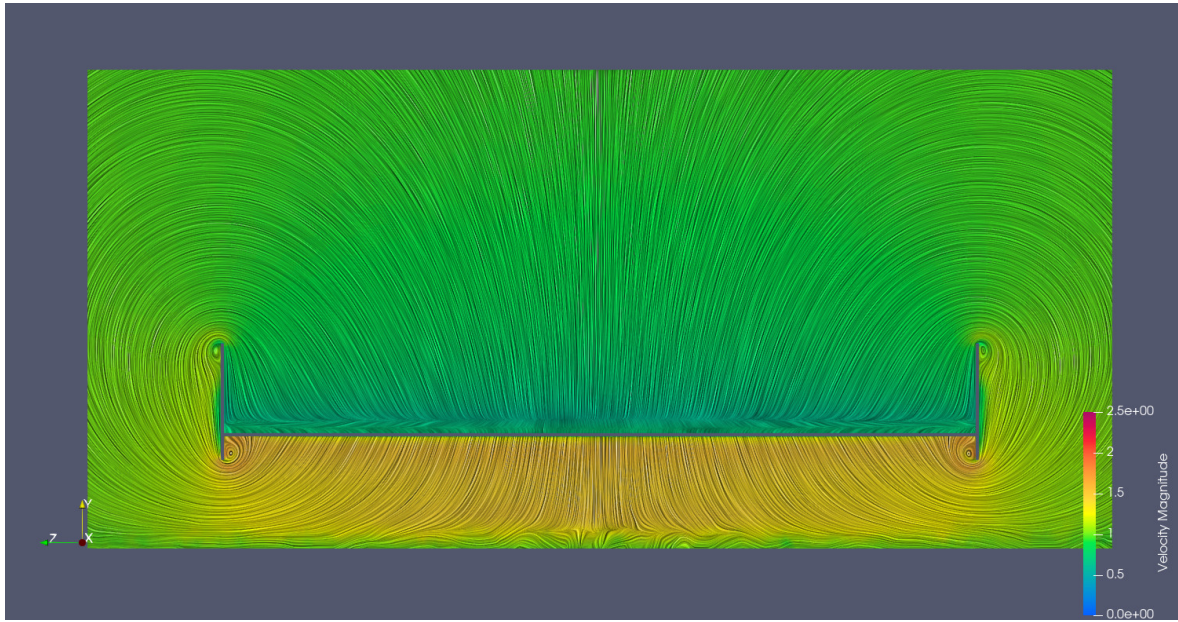


Figura E.22: Visualização de velocidade do escoamento no plano yz do caso h130

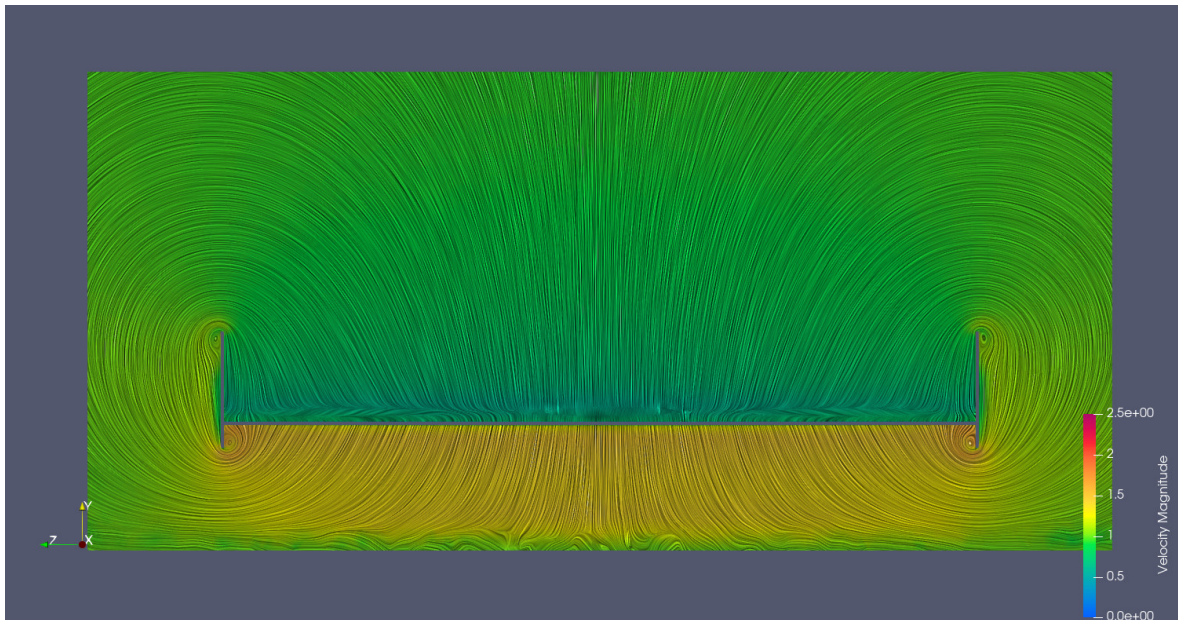


Figura E.23: Visualização de velocidade do escoamento no plano yz do caso h150



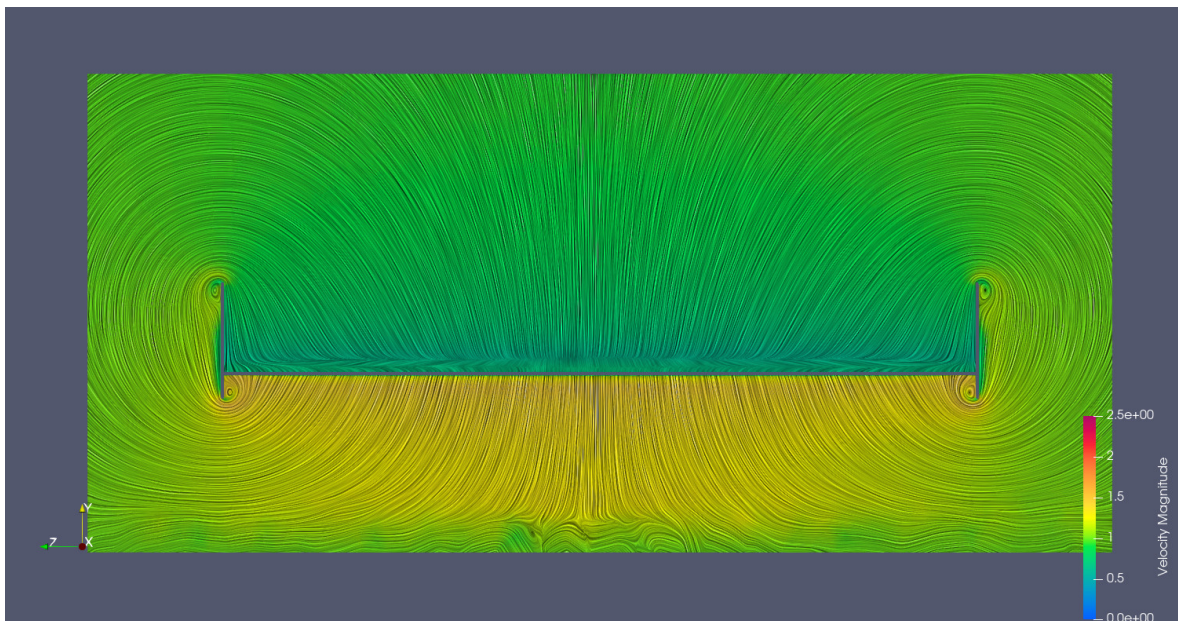


Figura E.24: Visualização de velocidade do escoamento no plano yz do caso h225

# APÊNDICE F

## Visualizações de Velocidade - Plano yz - Detalhe dos Vórtices de Ponta de Asa de Asa

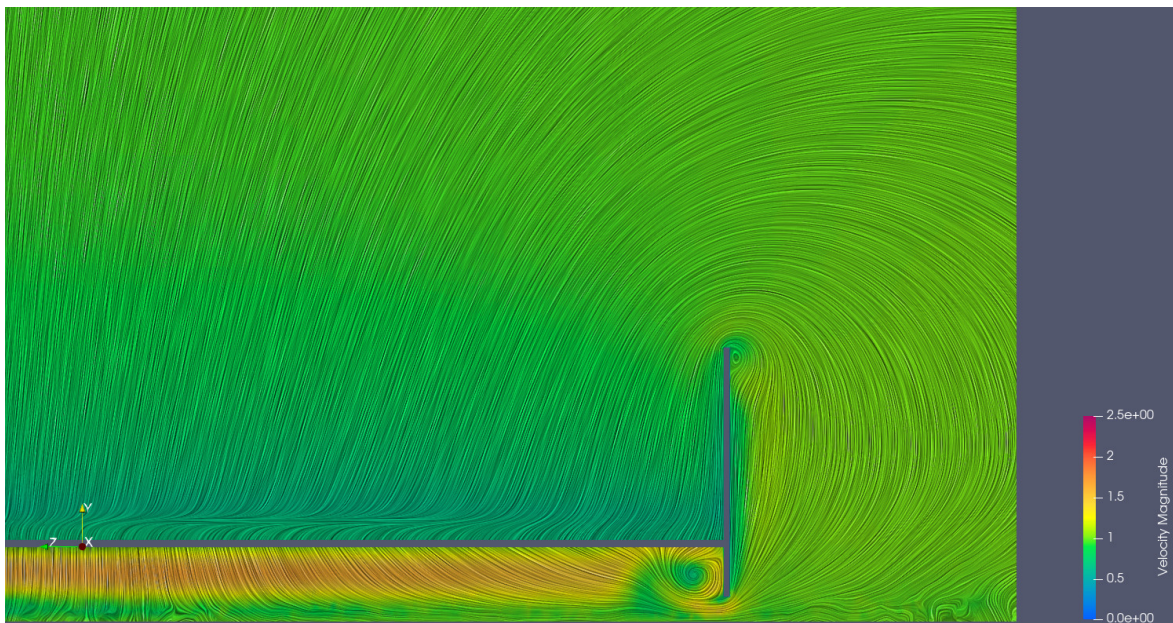


Figura F.1: Detalhe da visualização de velocidade do escoamento no plano yz mostrando os vórtices de ponta de asa do caso h17



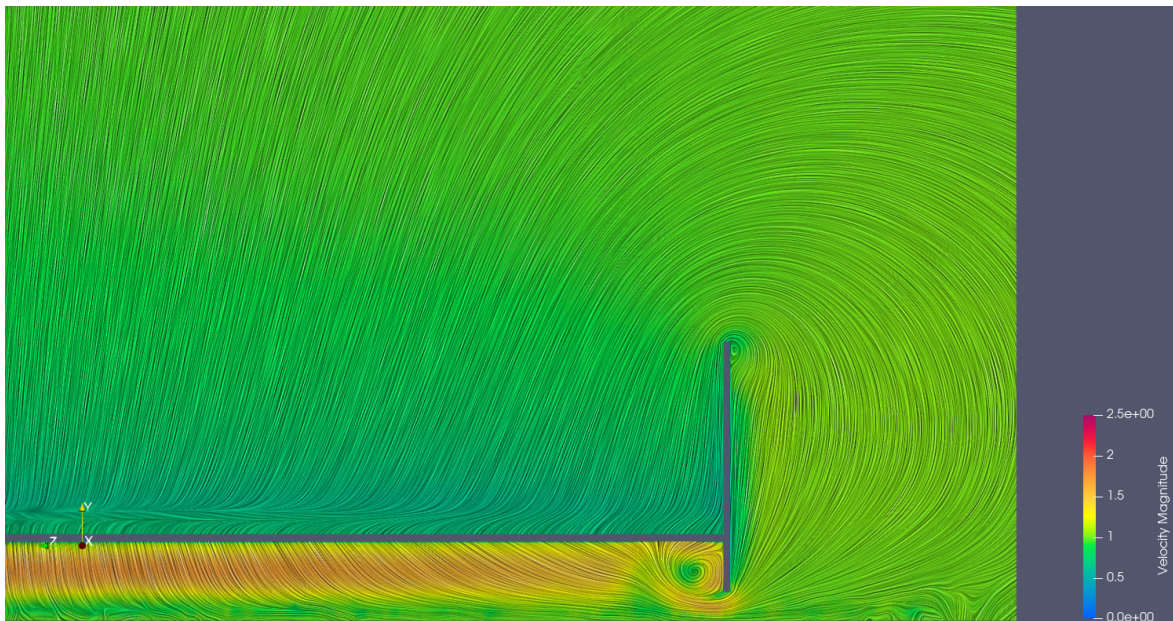


Figura F.2: Detalhe da visualização de velocidade do escoamento no plano yz mostrando os vórtices de ponta de asa do caso h20

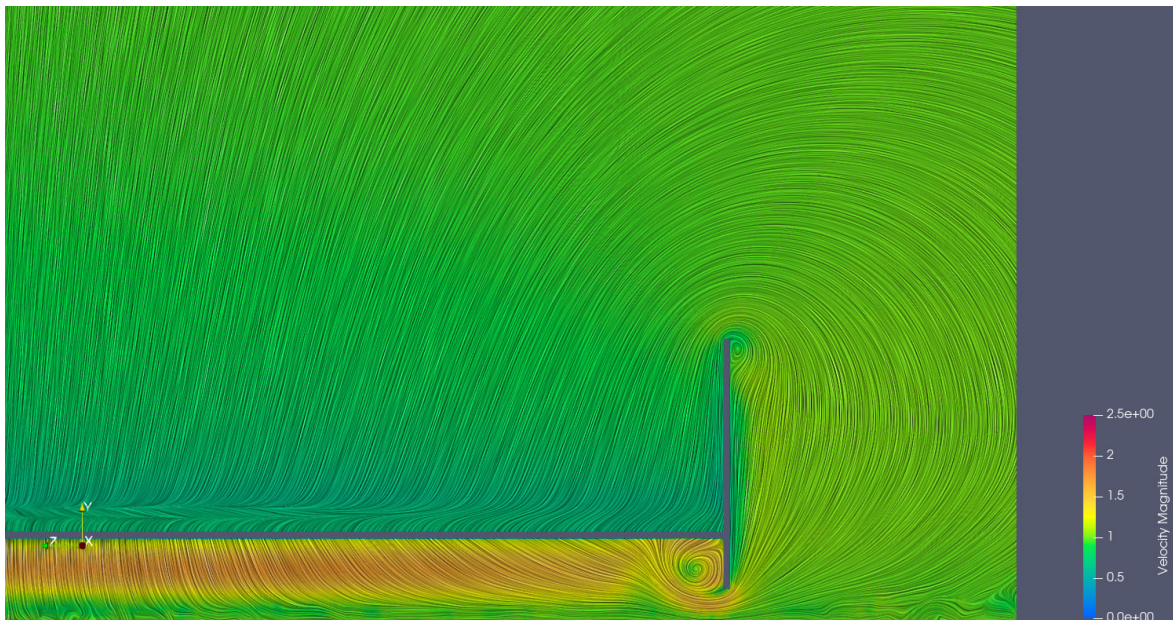


Figura F.3: Detalhe da visualização de velocidade do escoamento no plano yz mostrando os vórtices de ponta de asa do caso h22



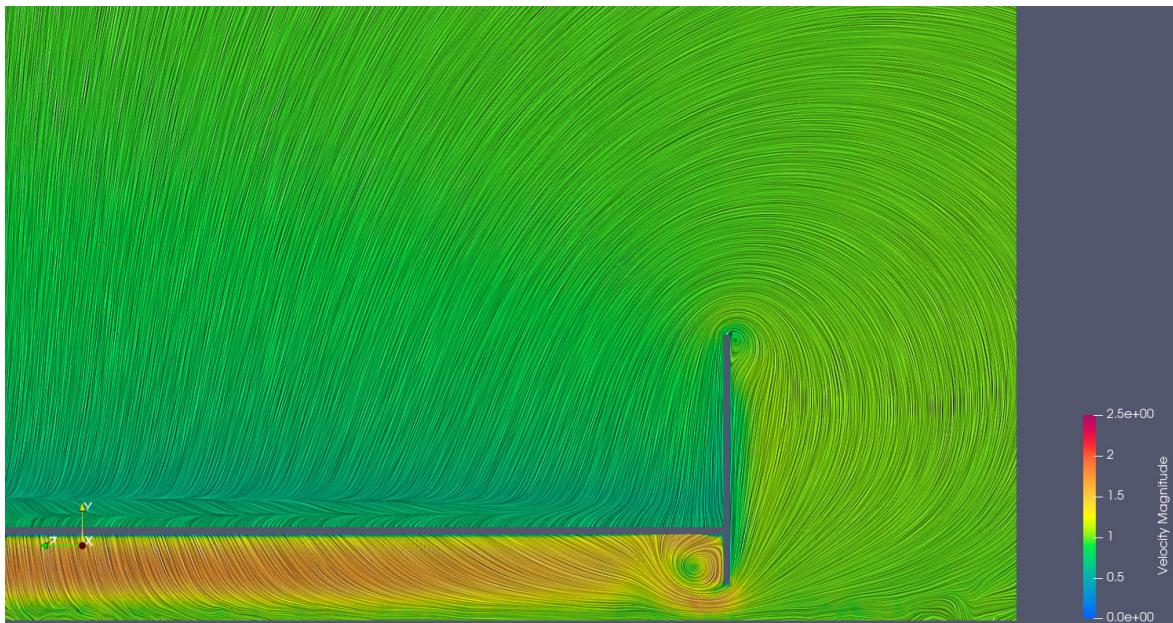


Figura F.4: Detalhe da visualização de velocidade do escoamento no plano yz mostrando os vórtices de ponta de asa do caso h25

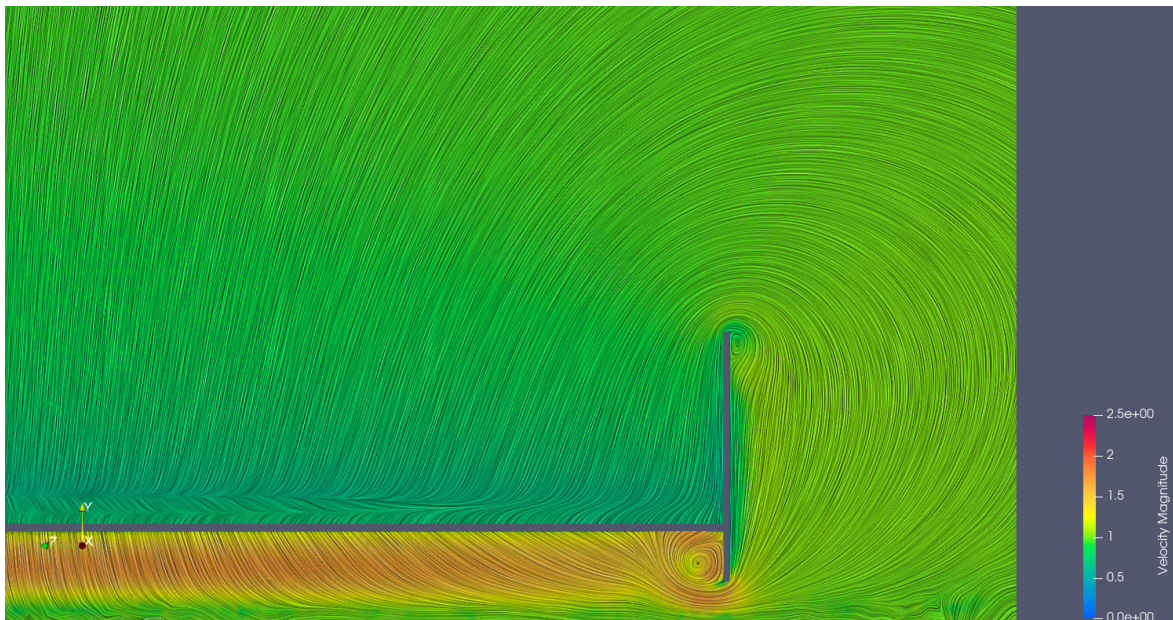


Figura F.5: Detalhe da visualização de velocidade do escoamento no plano yz mostrando os vórtices de ponta de asa do caso h27



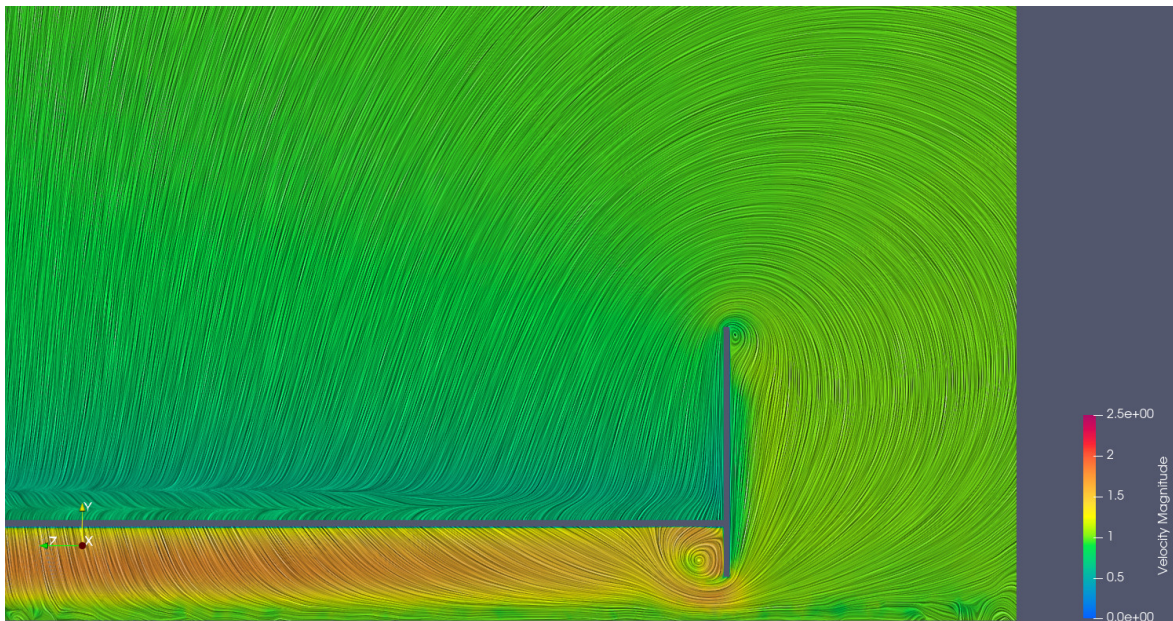


Figura F.6: Detalhe da visualização de velocidade do escoamento no plano yz mostrando os vórtices de ponta de asa do caso h30

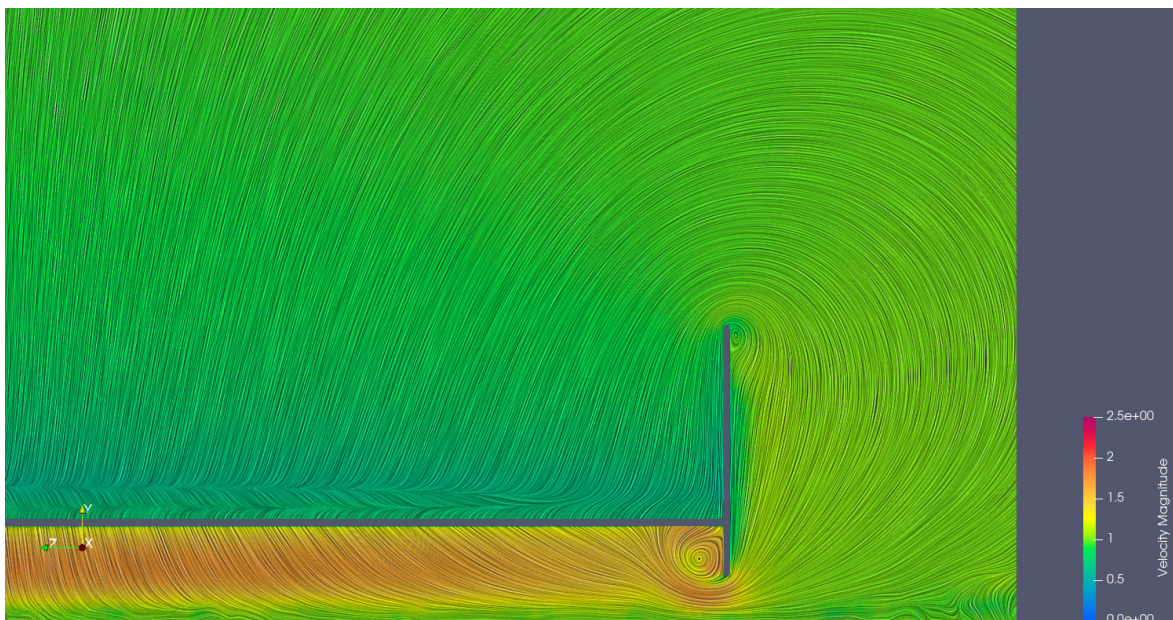


Figura F.7: Detalhe da visualização de velocidade do escoamento no plano yz mostrando os vórtices de ponta de asa do caso h32



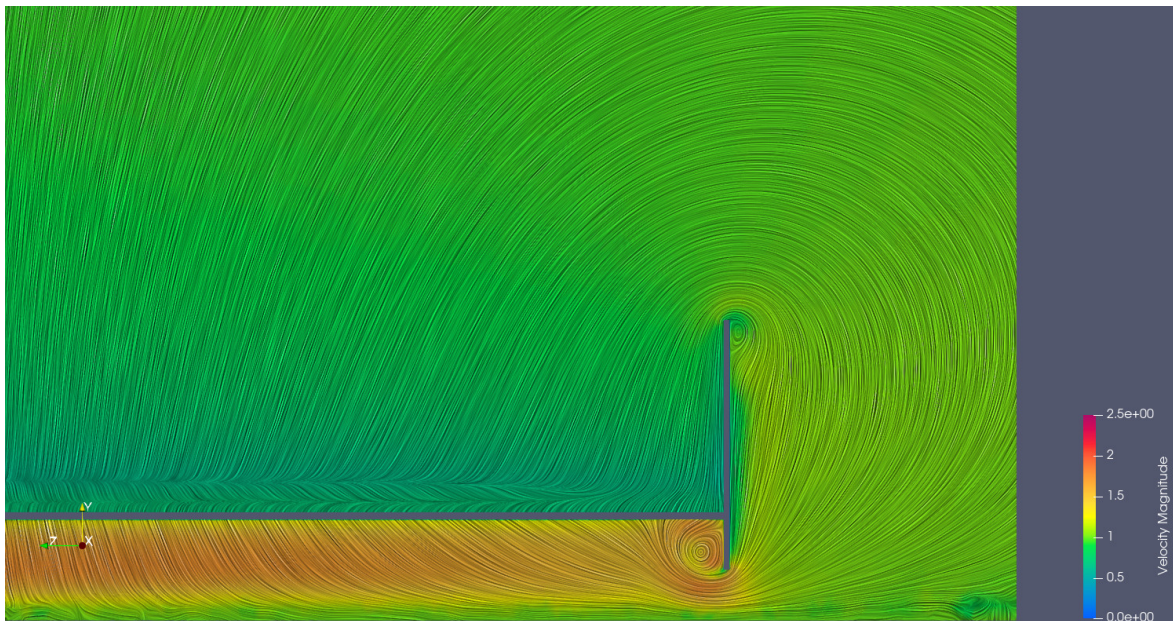


Figura F.8: Detalhe da visualização de velocidade do escoamento no plano yz mostrando os vórtices de ponta de asa do caso h35

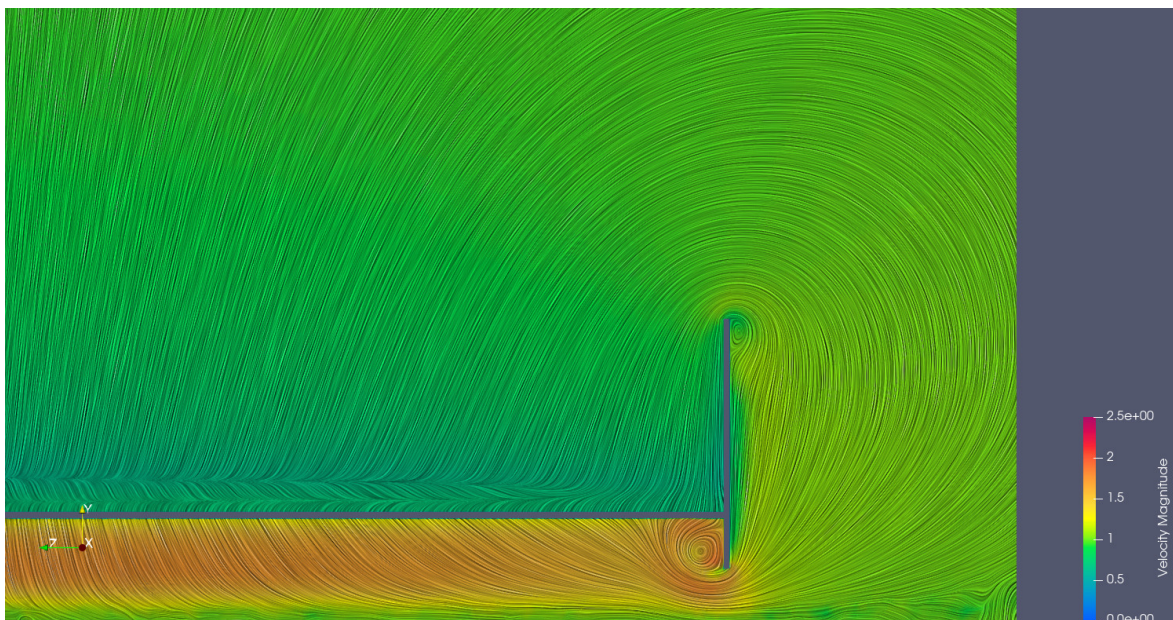


Figura F.9: Detalhe da visualização de velocidade do escoamento no plano yz mostrando os vórtices de ponta de asa do caso h37



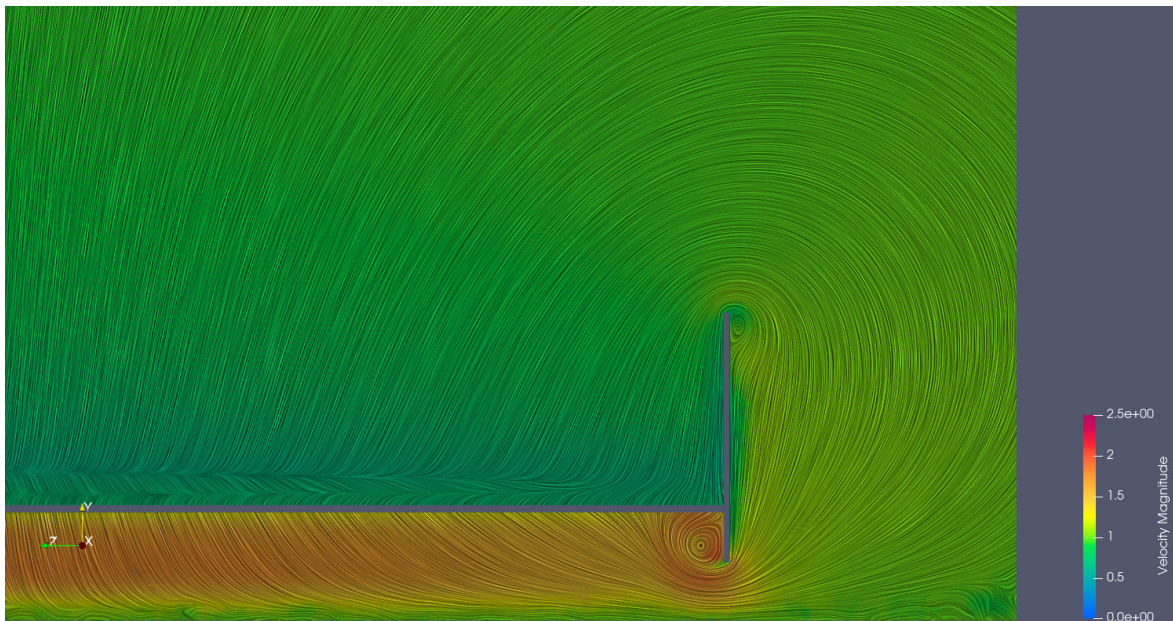


Figura F.10: Detalhe da visualização de velocidade do escoamento no plano yz mostrando os vórtices de ponta de asa do caso h40

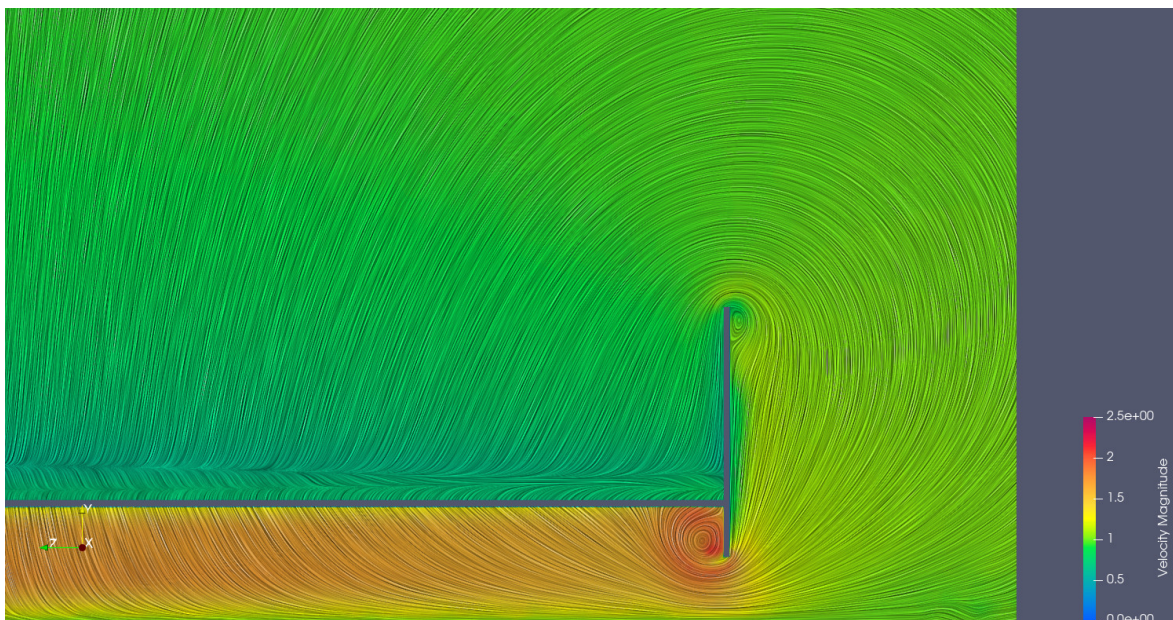


Figura F.11: Detalhe da visualização de velocidade do escoamento no plano yz mostrando os vórtices de ponta de asa do caso h45



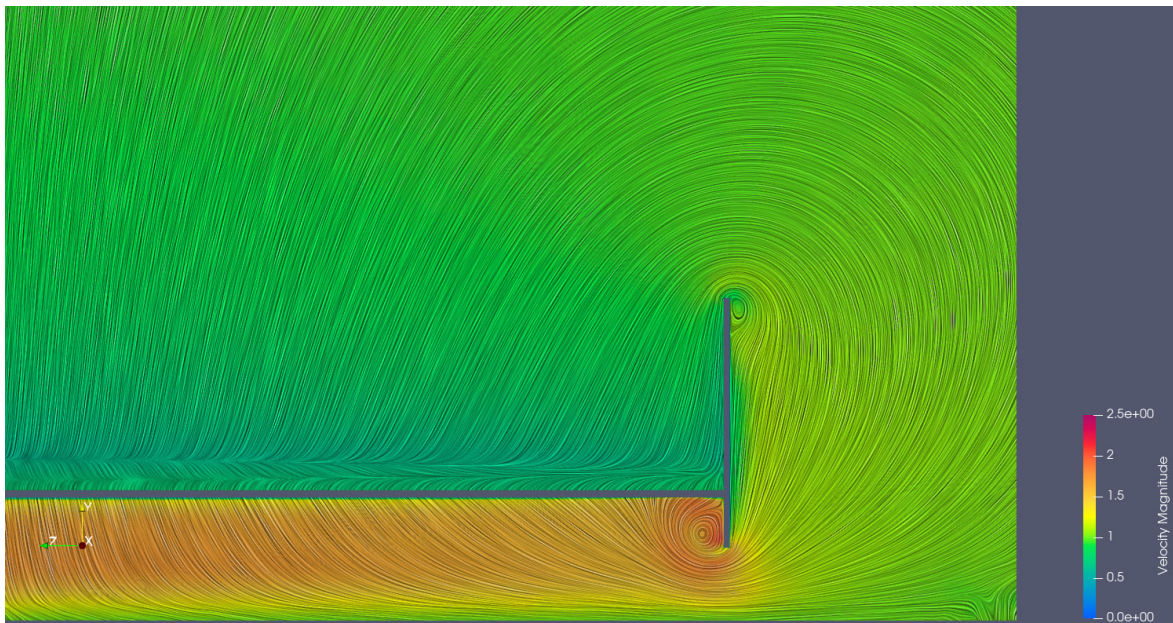


Figura F.12: Detalhe da visualização de velocidade do escoamento no plano yz mostrando os vórtices de ponta de asa do caso h50

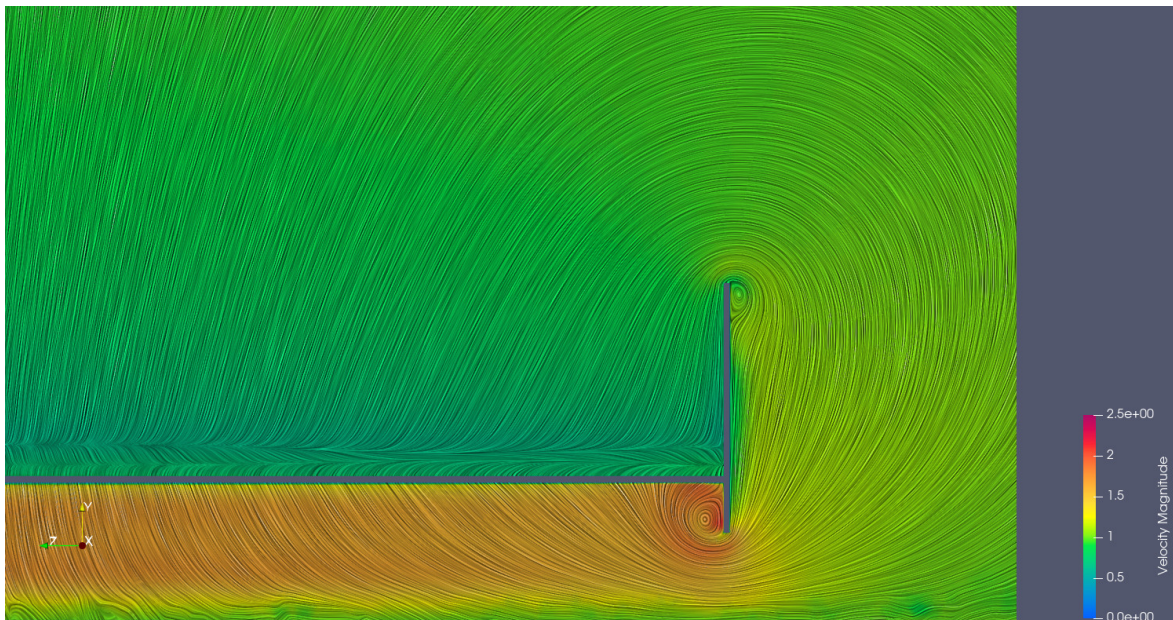


Figura F.13: Detalhe da visualização de velocidade do escoamento no plano yz mostrando os vórtices de ponta de asa do caso h60



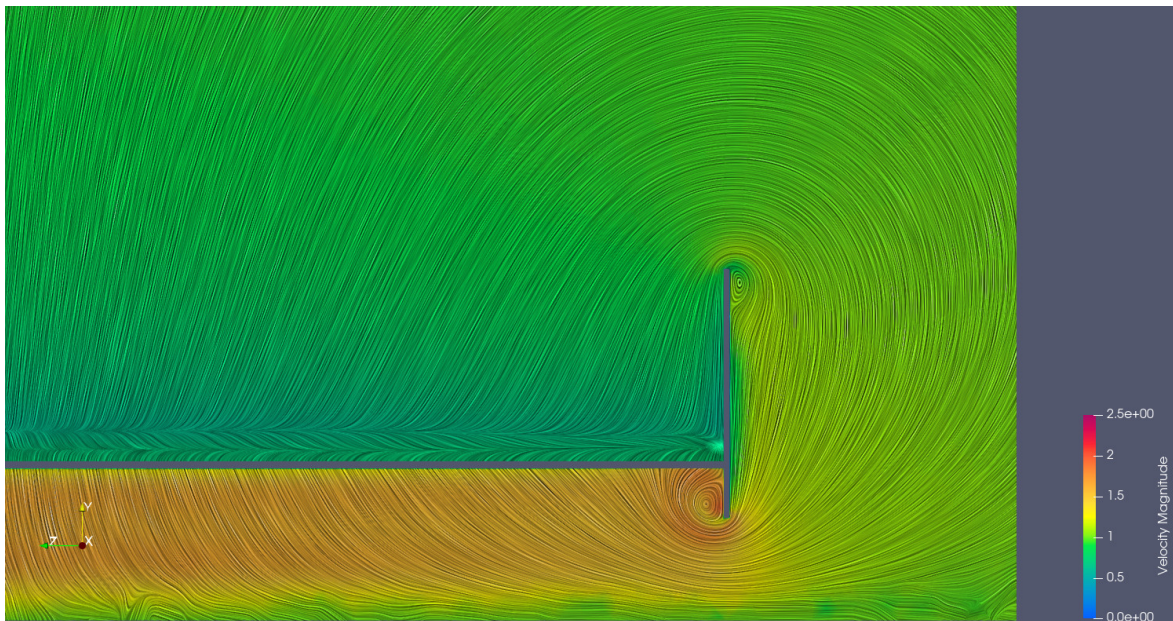


Figura F.14: Detalhe da visualização de velocidade do escoamento no plano yz mostrando os vórtices de ponta de asa do caso h70

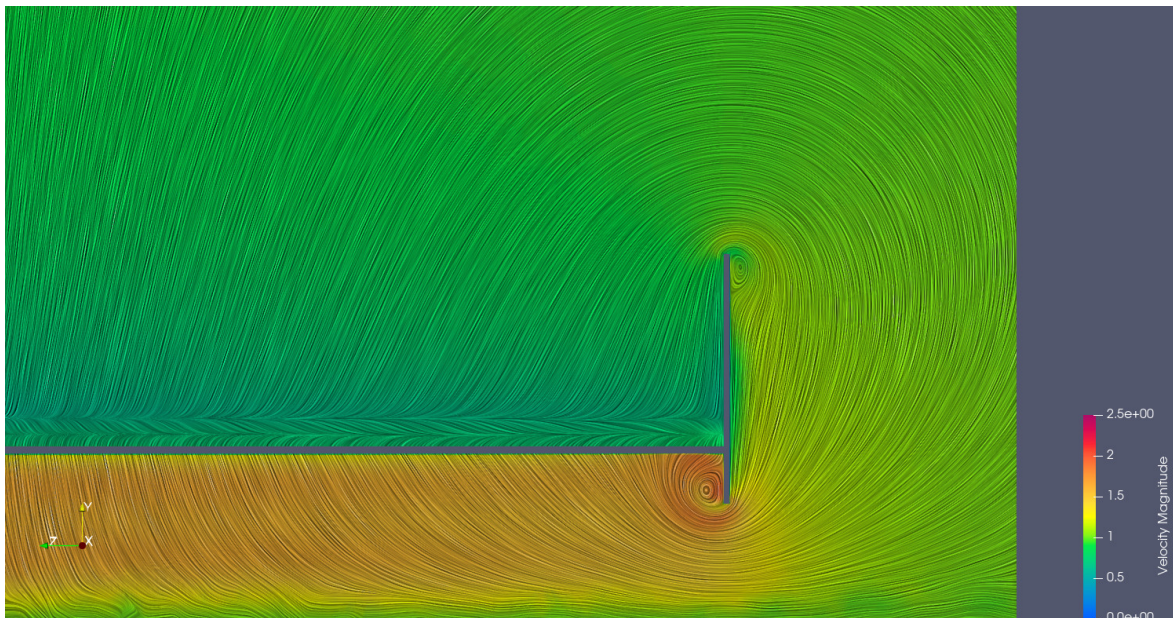


Figura F.15: Detalhe da visualização de velocidade do escoamento no plano yz mostrando os vórtices de ponta de asa do caso h80



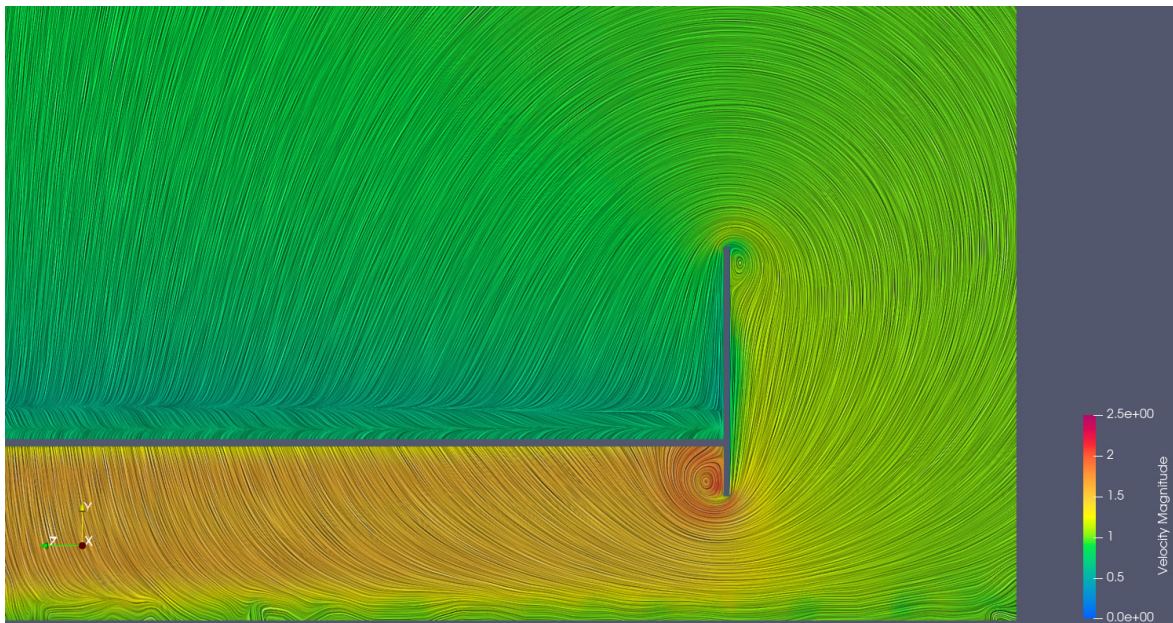


Figura F.16: Detalhe da visualização de velocidade do escoamento no plano yz mostrando os vórtices de ponta de asa do caso h85

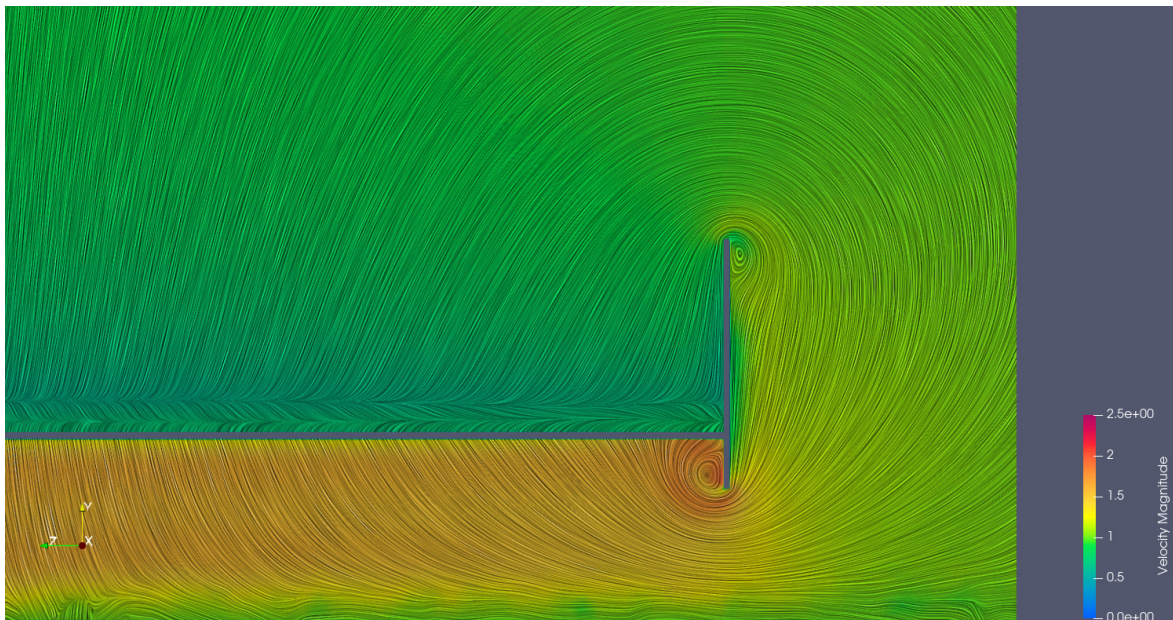


Figura F.17: Detalhe da visualização de velocidade do escoamento no plano yz mostrando os vórtices de ponta de asa do caso h90



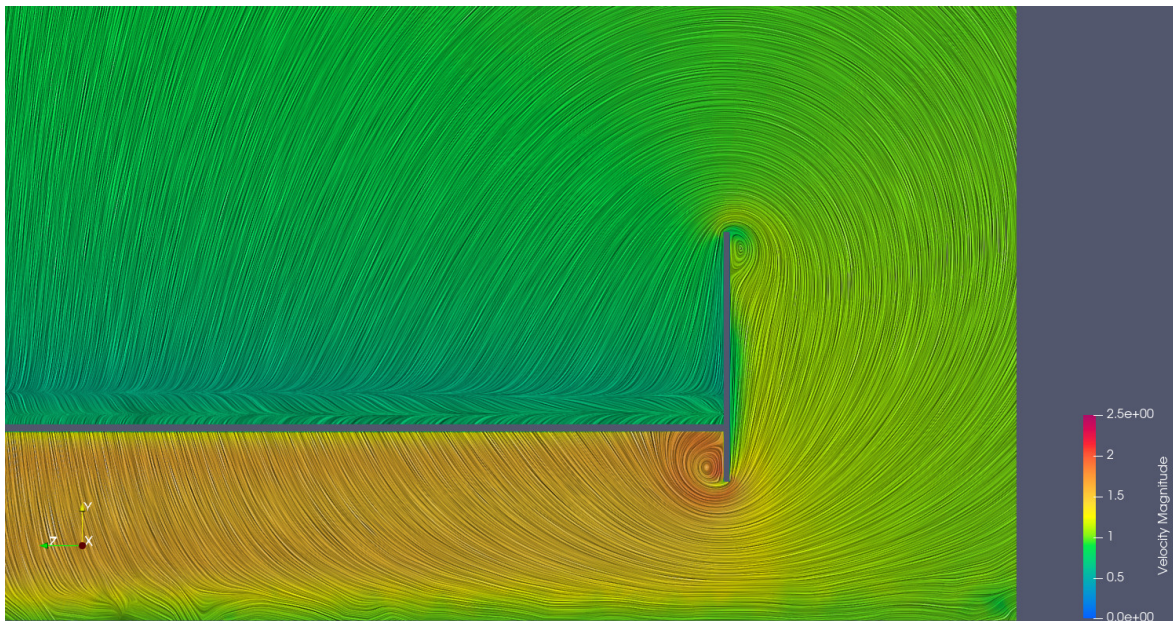


Figura F.18: Detalhe da visualização de velocidade do escoamento no plano yz mostrando os vórtices de ponta de asa do caso h95

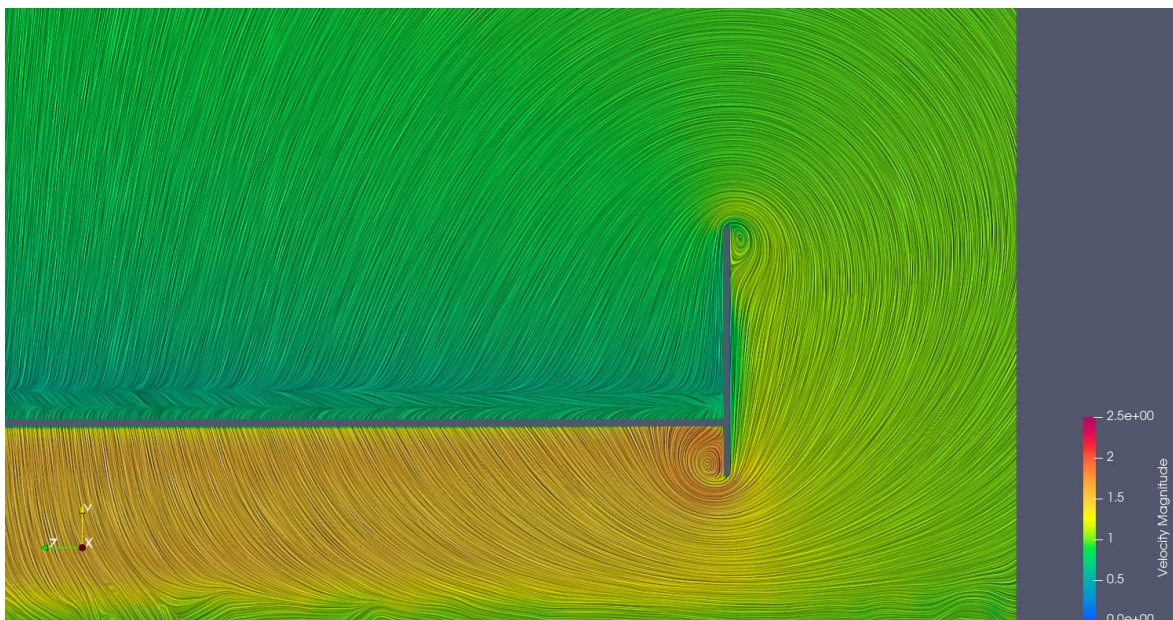


Figura F.19: Detalhe da visualização de velocidade do escoamento no plano yz mostrando os vórtices de ponta de asa do caso h100



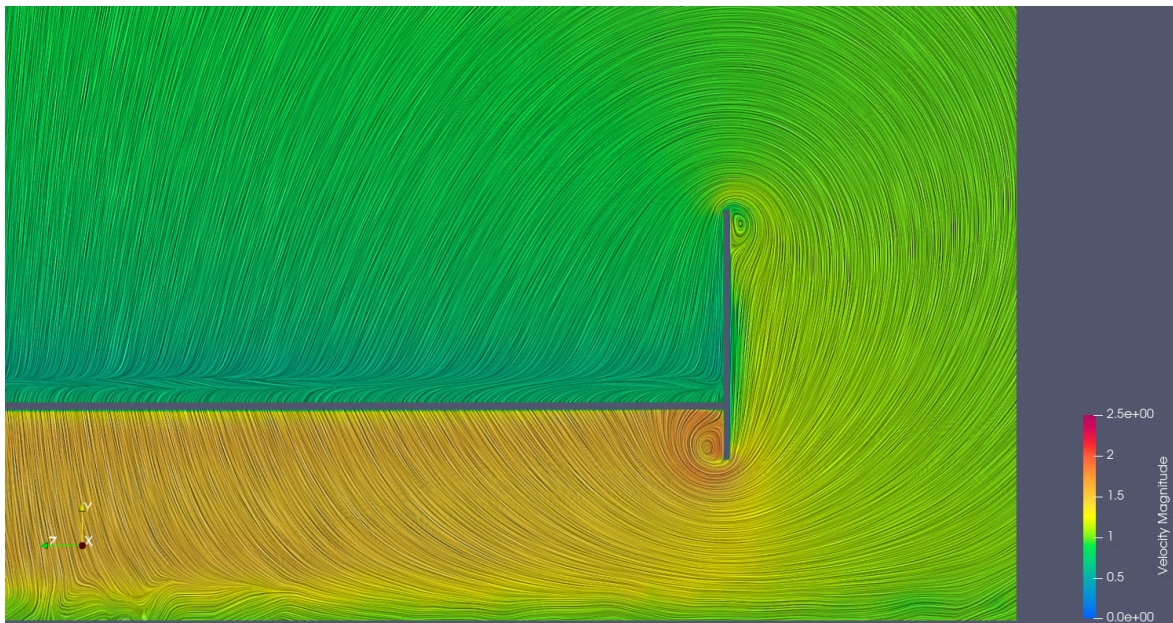


Figura F.20: Detalhe da visualização de velocidade do escoamento no plano yz mostrando os vórtices de ponta de asa do caso h110

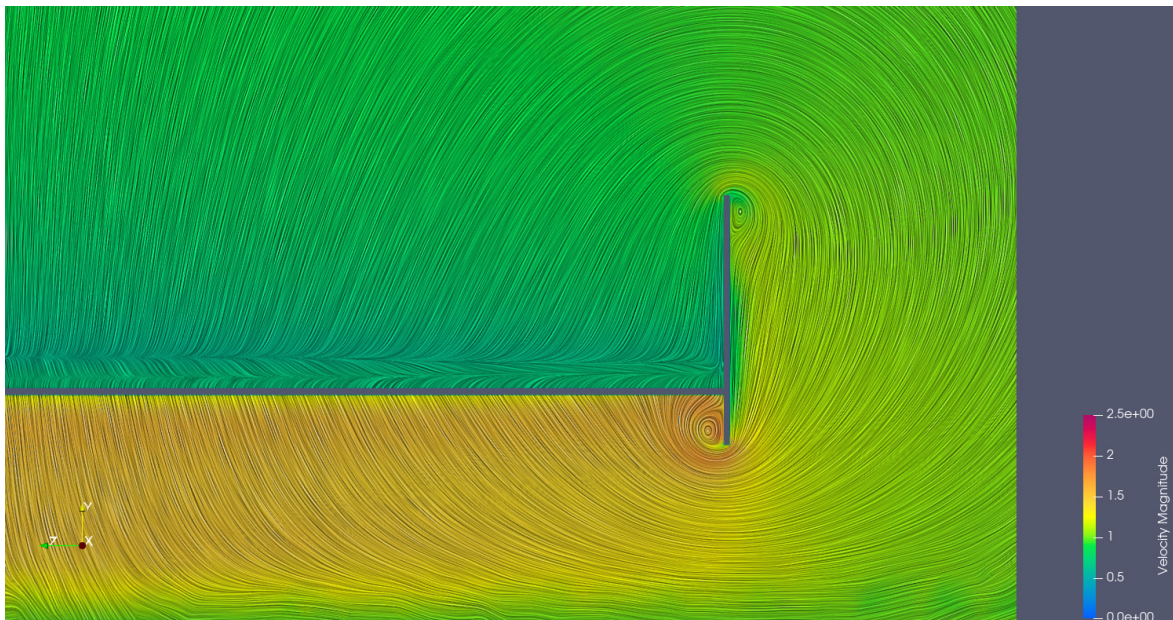


Figura F.21: Detalhe da visualização de velocidade do escoamento no plano yz mostrando os vórtices de ponta de asa do caso h120



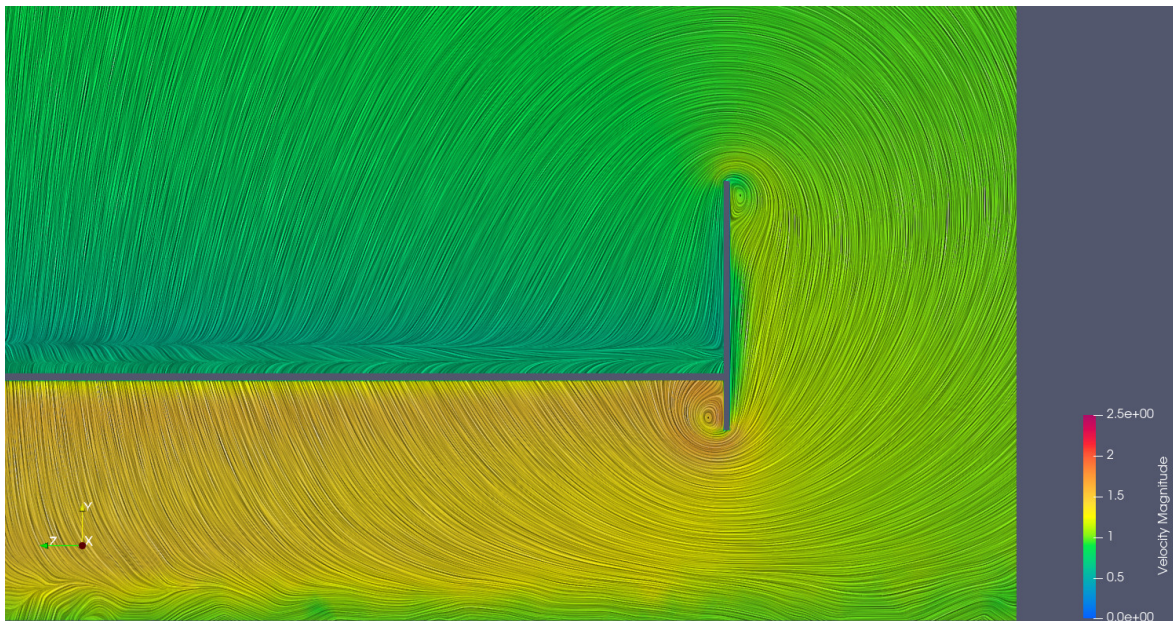


Figura F.22: Detalhe da visualização de velocidade do escoamento no plano yz mostrando os vórtices de ponta de asa do caso h130

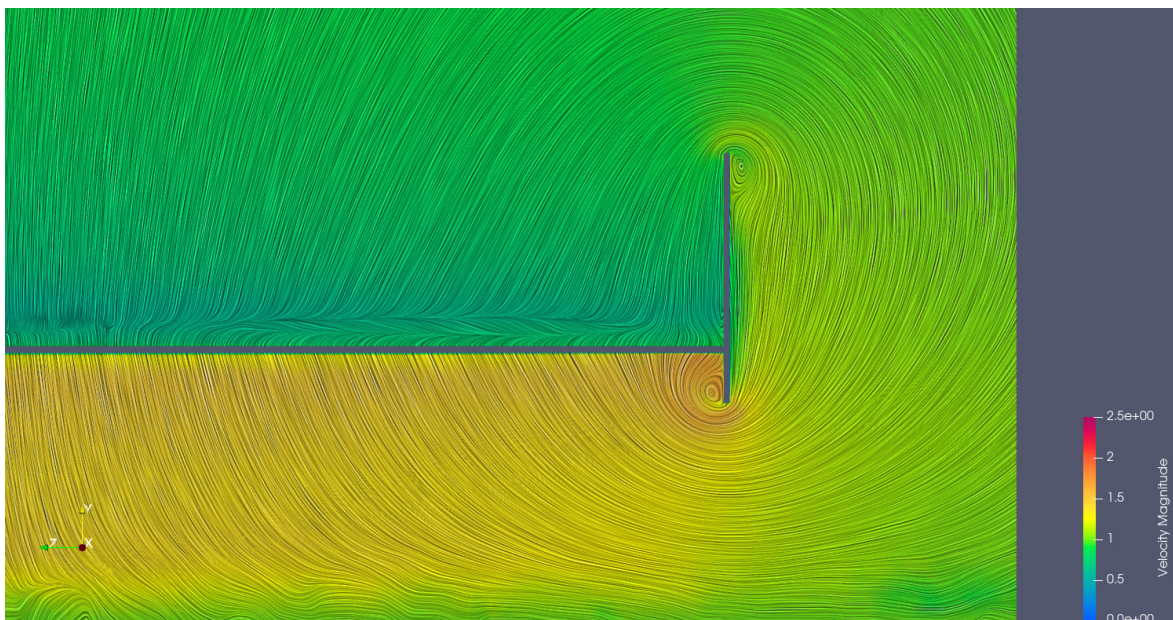


Figura F.23: Detalhe da visualização de velocidade do escoamento no plano yz mostrando os vórtices de ponta de asa do caso h150

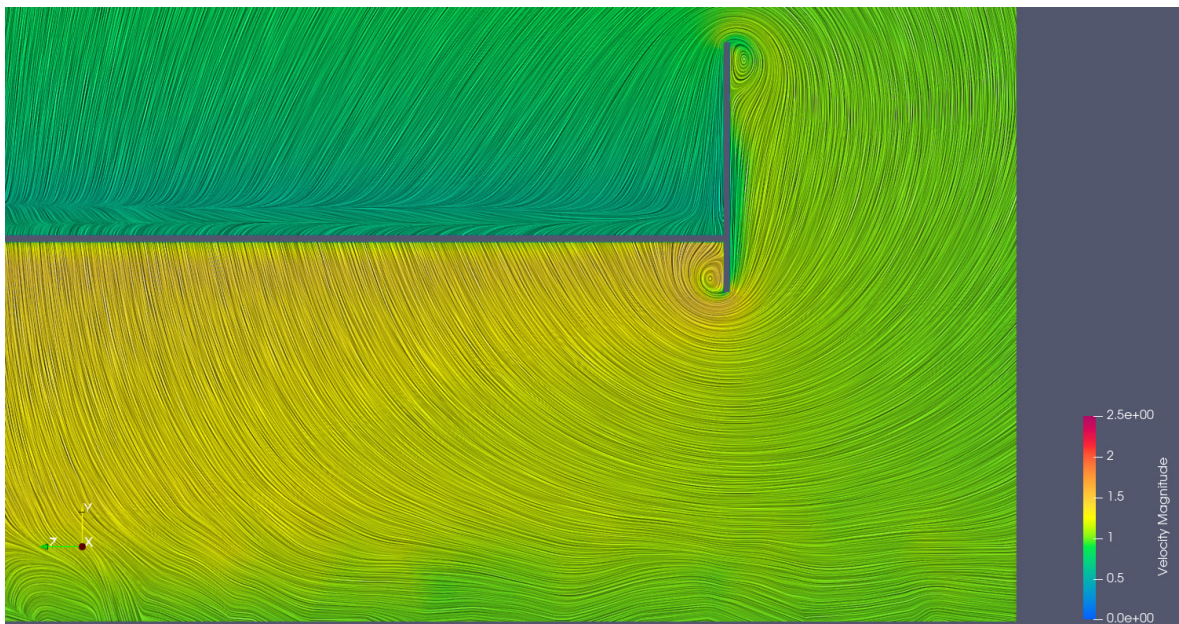


Figura F.24: Detalhe da visualização de velocidade do escoamento no plano yz mostrando os vórtices de ponta de asa do caso h225



# APÊNDICE G

## Visualizações 3D das linhas de corrente

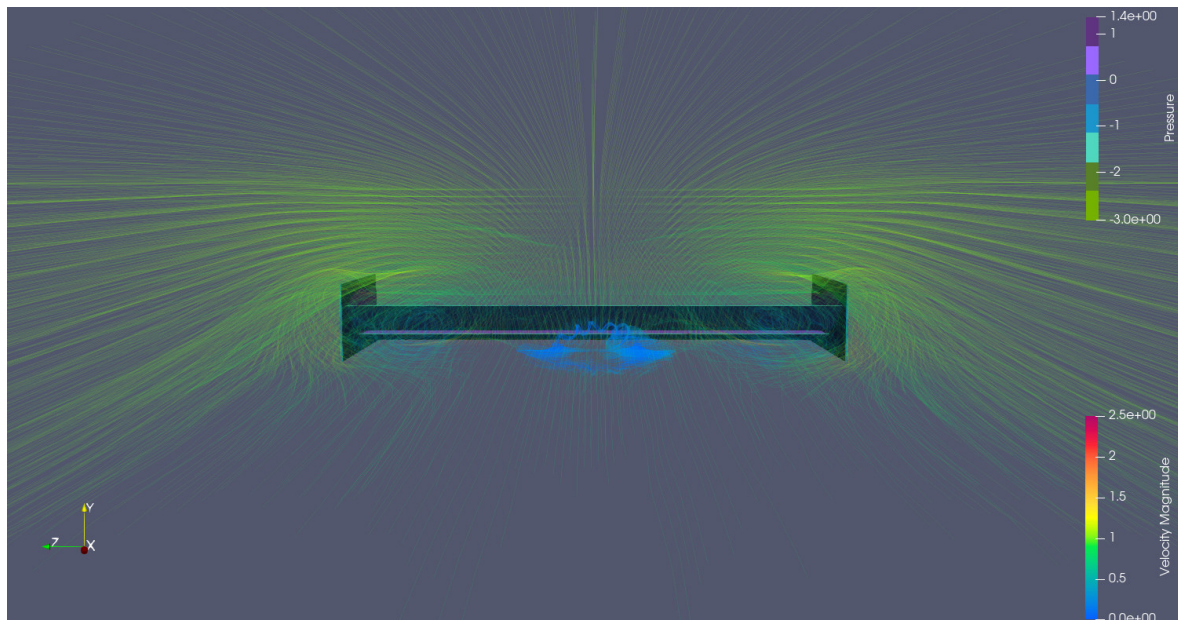


Figura G.1: Visualização 3D do escoamento mostrando os vórtices de ponta de asa do caso h17



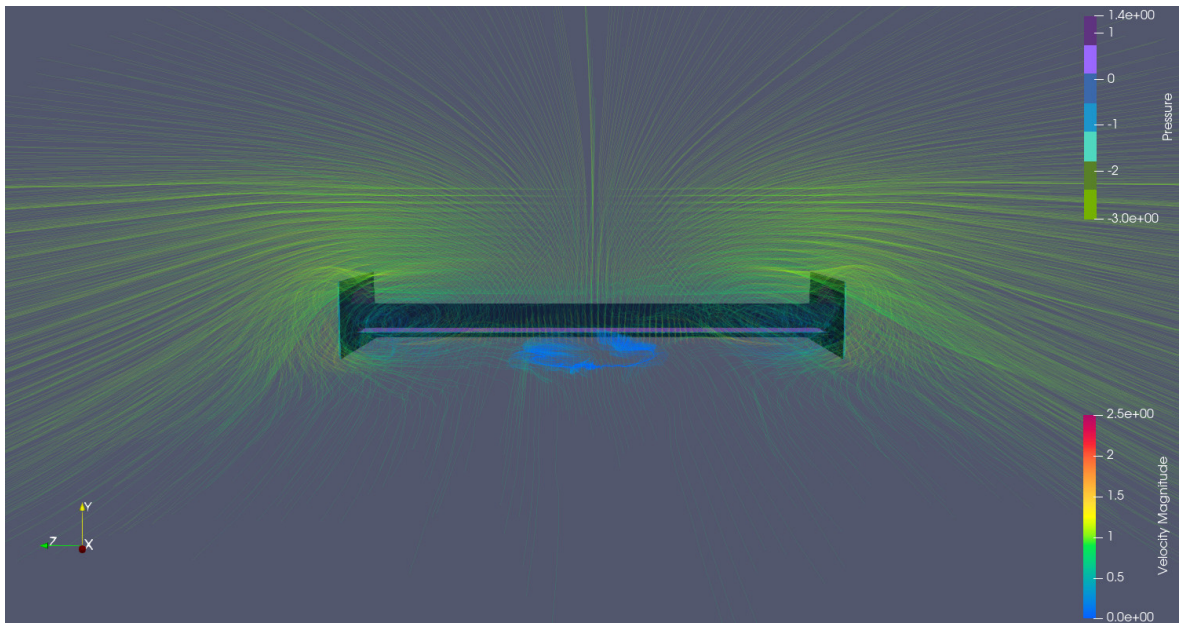


Figura G.2: Visualização 3D do escoamento mostrando os vórtices de ponta de asa do caso h20

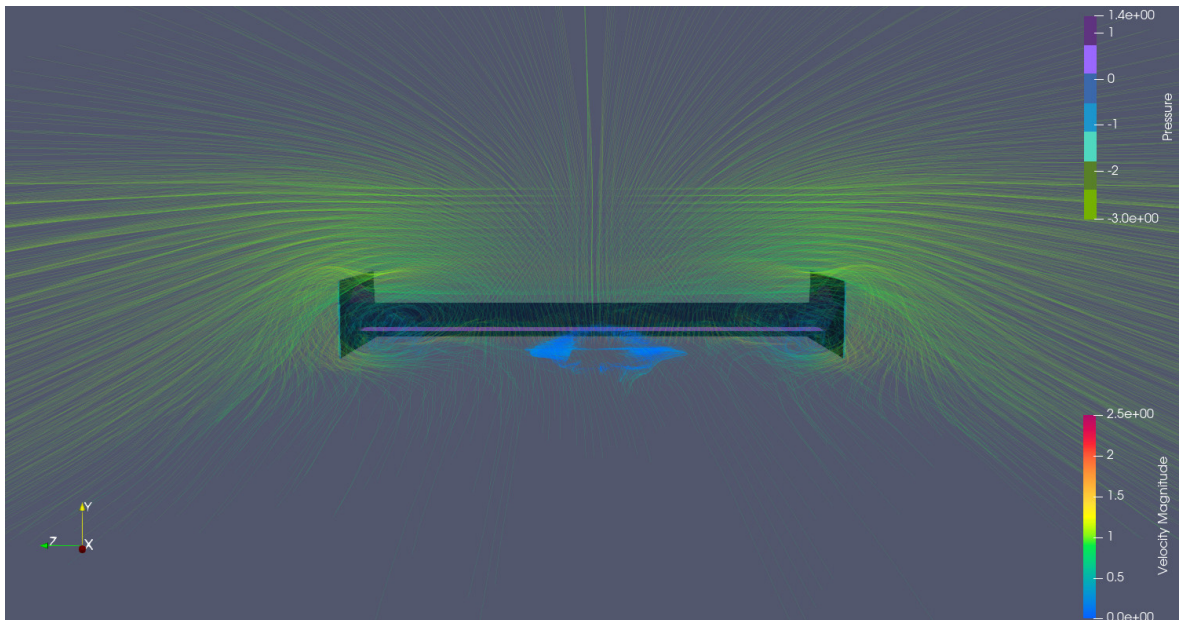


Figura G.3: Visualização 3D do escoamento mostrando os vórtices de ponta de asa do caso h22

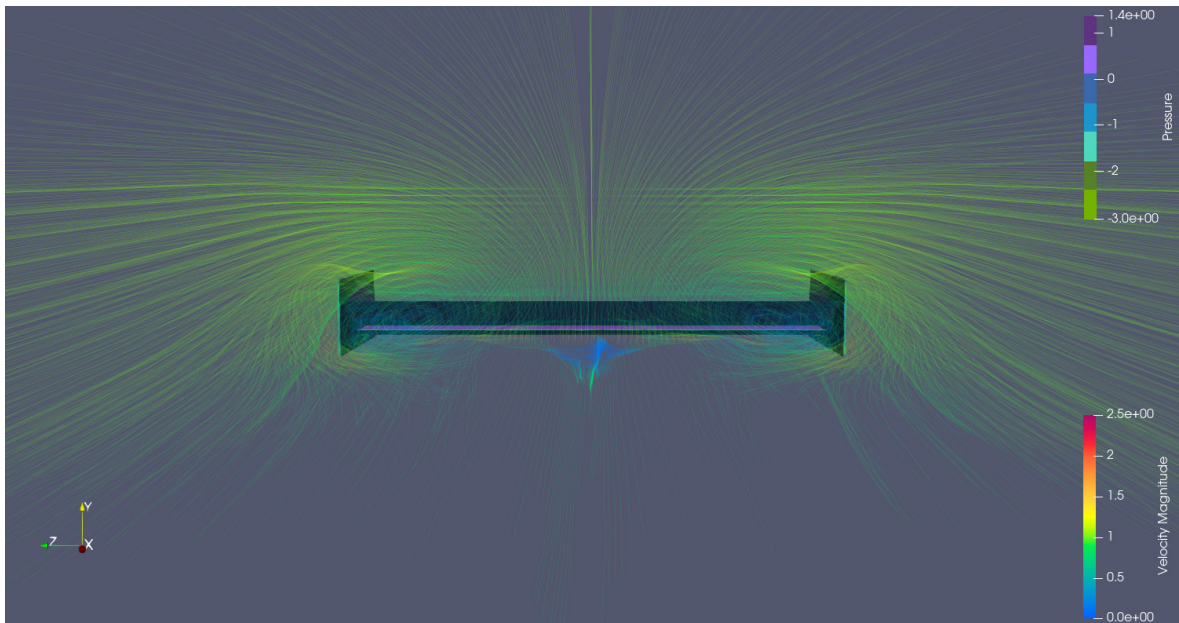


Figura G.4: Visualização 3D do escoamento mostrando os vórtices de ponta de asa do caso h25

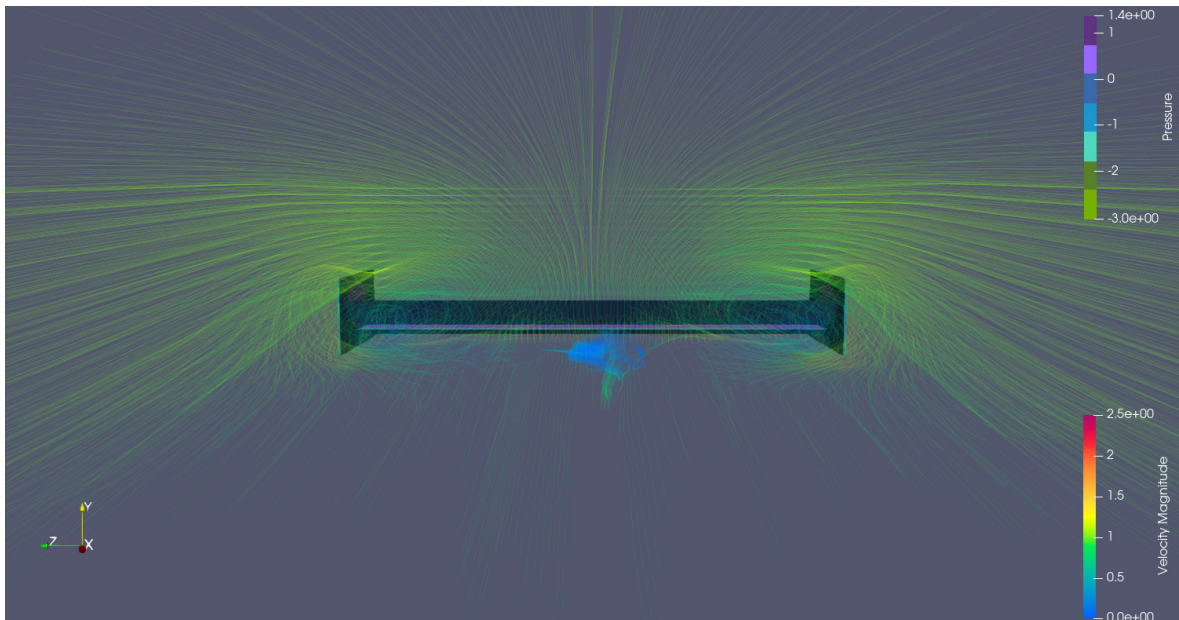


Figura G.5: Visualização 3D do escoamento mostrando os vórtices de ponta de asa do caso h27



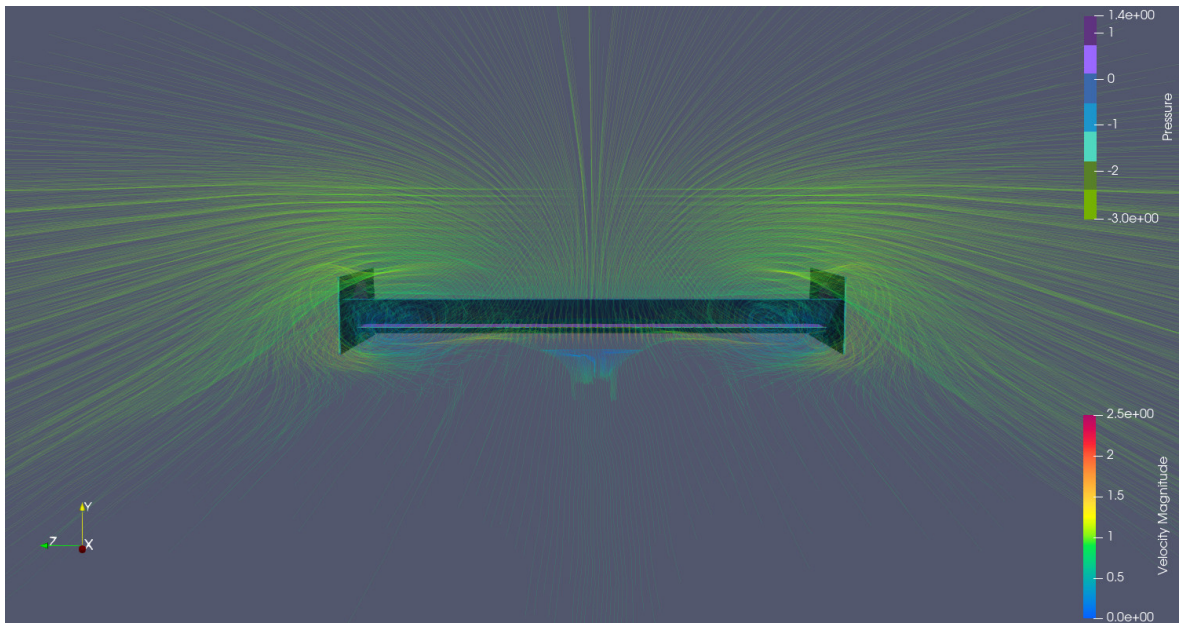


Figura G.6: Visualização 3D do escoamento mostrando os vórtices de ponta de asa do caso h30

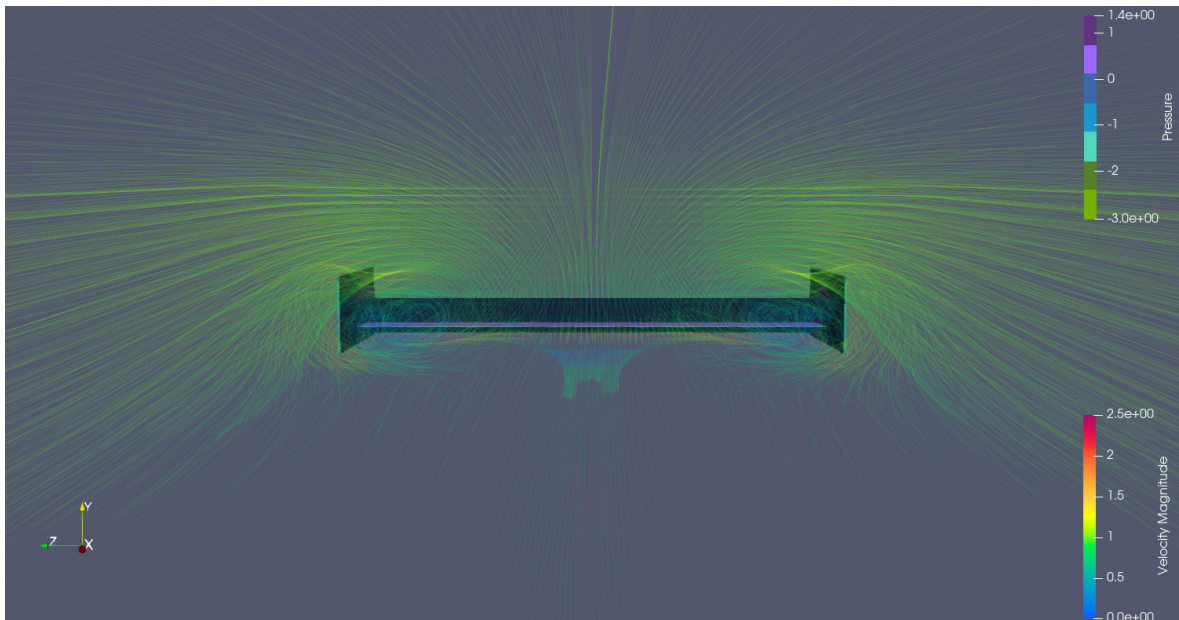


Figura G.7: Visualização 3D do escoamento mostrando os vórtices de ponta de asa do caso h32

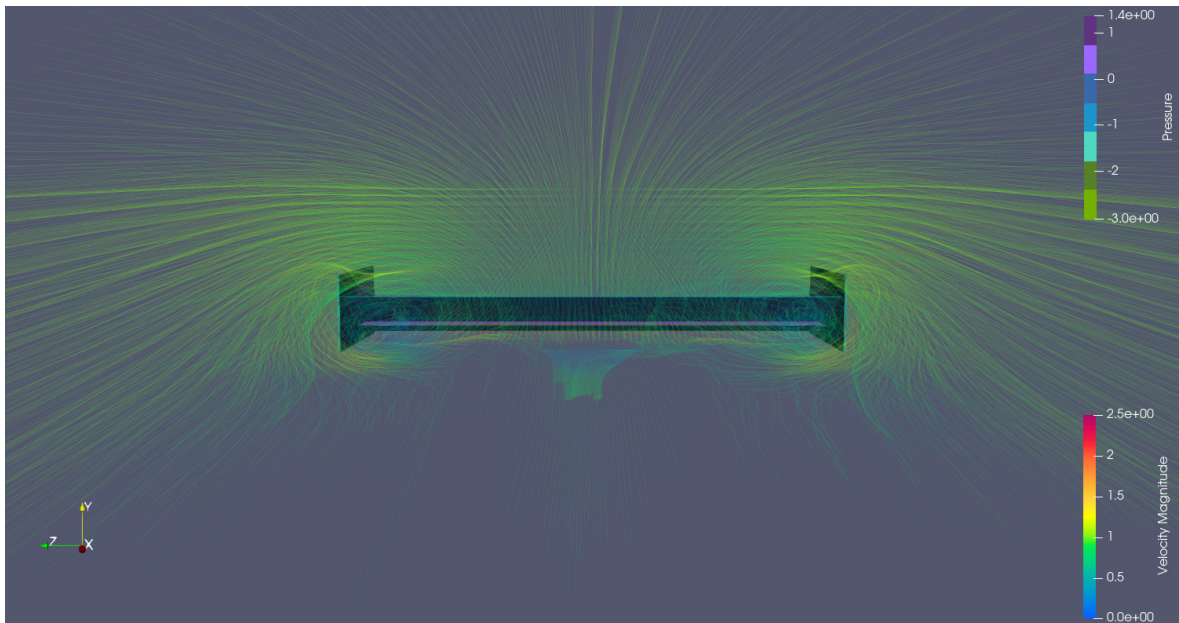


Figura G.8: Visualização 3D do escoamento mostrando os vórtices de ponta de asa do caso h35

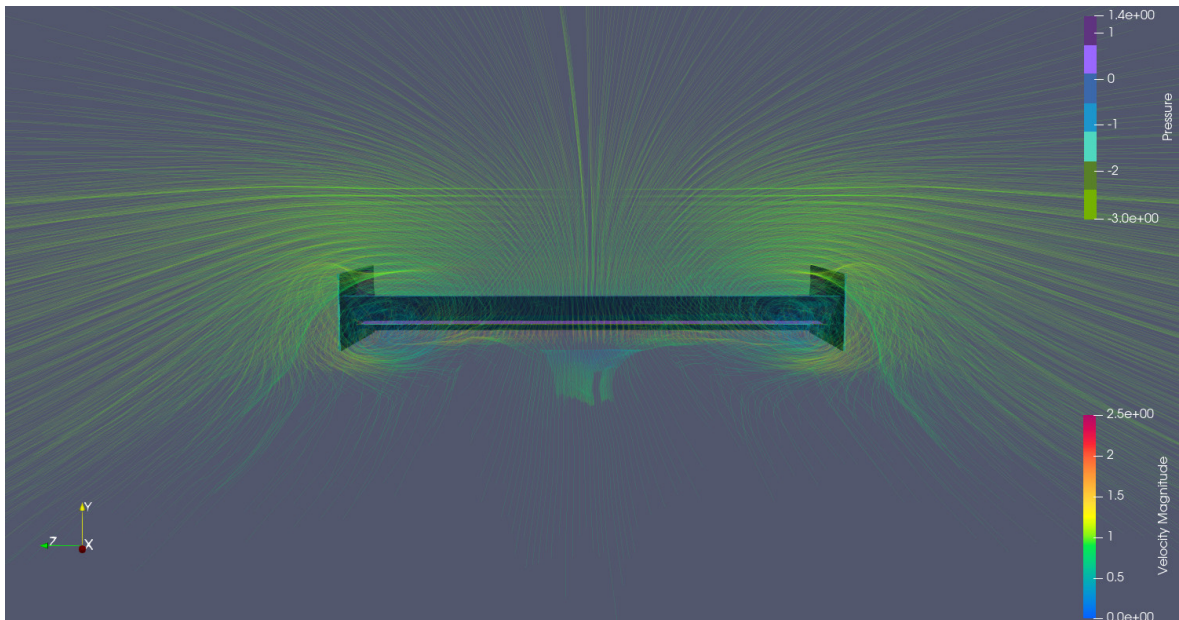


Figura G.9: Visualização 3D do escoamento mostrando os vórtices de ponta de asa do caso h37



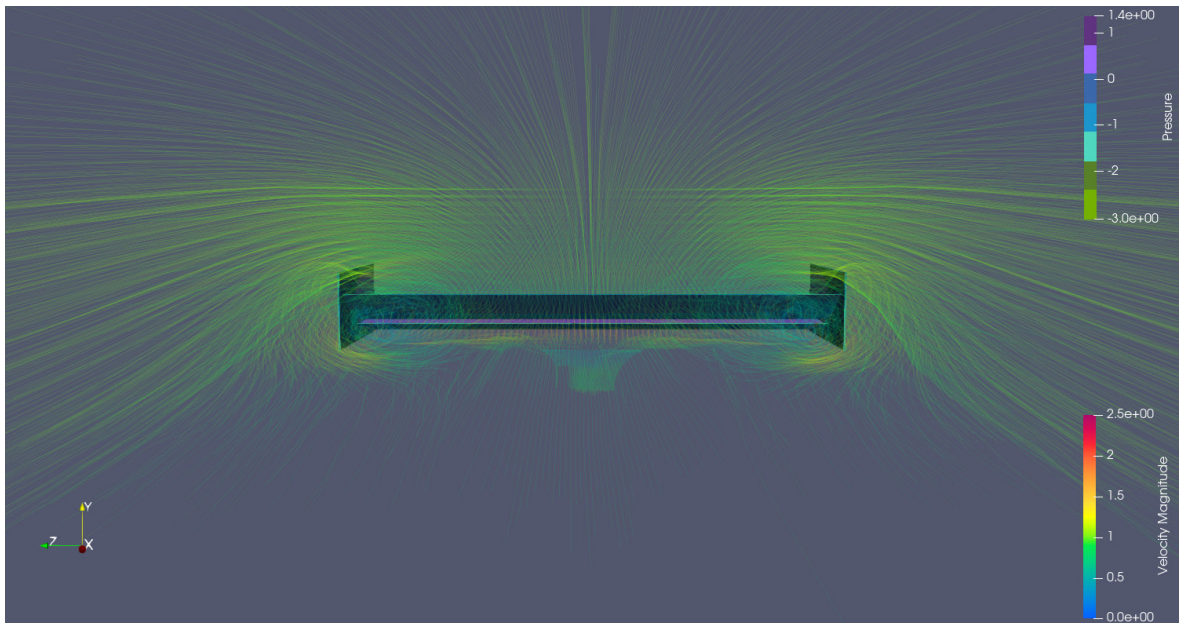


Figura G.10: Visualização 3D do escoamento mostrando os vórtices de ponta de asa do caso h40

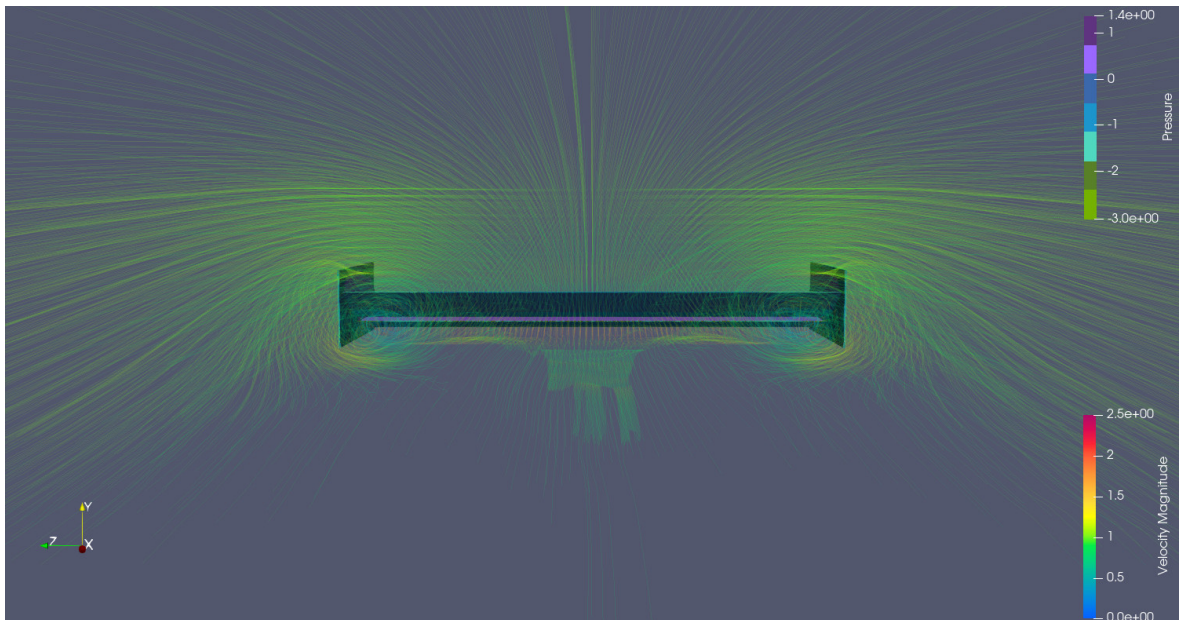


Figura G.11: Visualização 3D do escoamento mostrando os vórtices de ponta de asa do caso h45

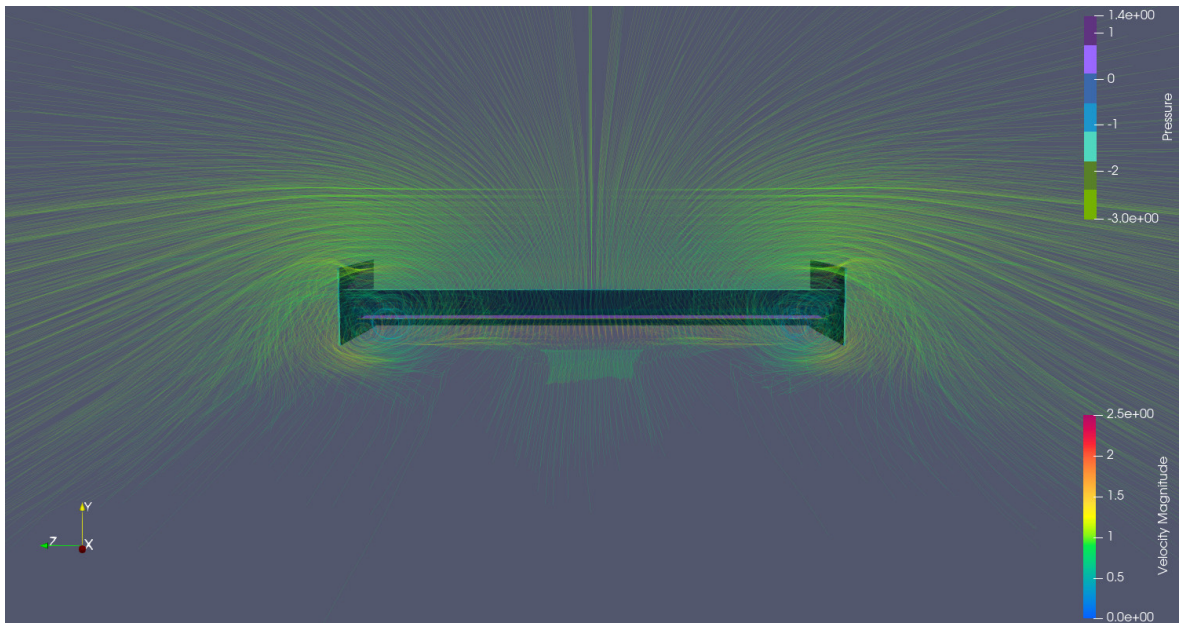


Figura G.12: Visualização 3D do escoamento mostrando os vórtices de ponta de asa do caso h50

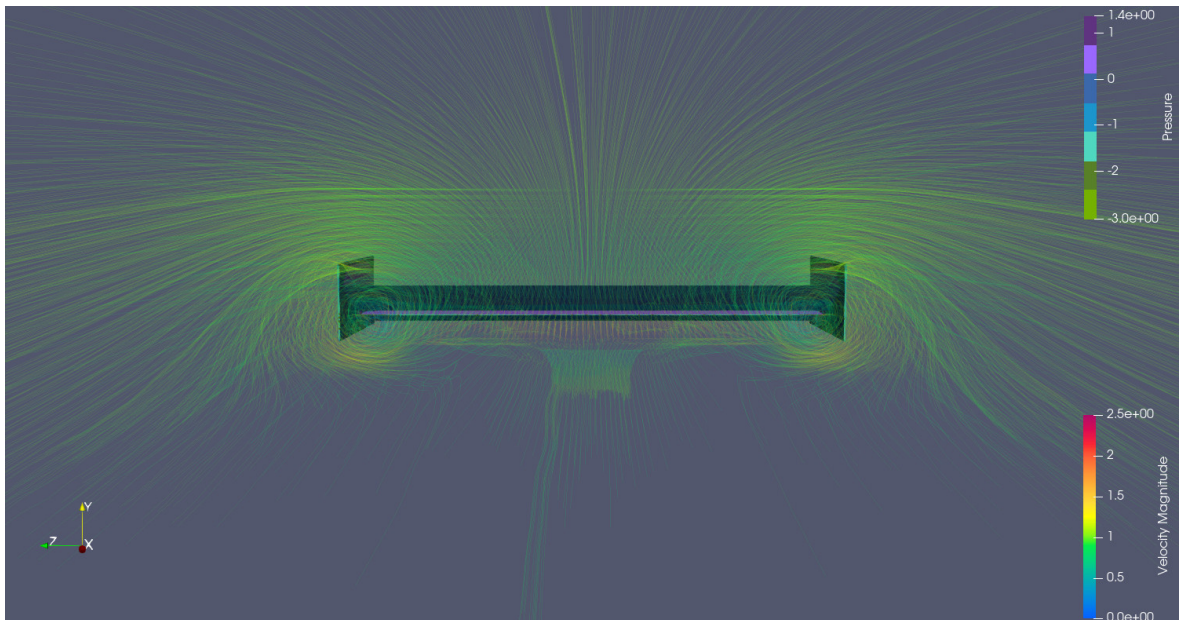


Figura G.13: Visualização 3D do escoamento mostrando os vórtices de ponta de asa do caso h60



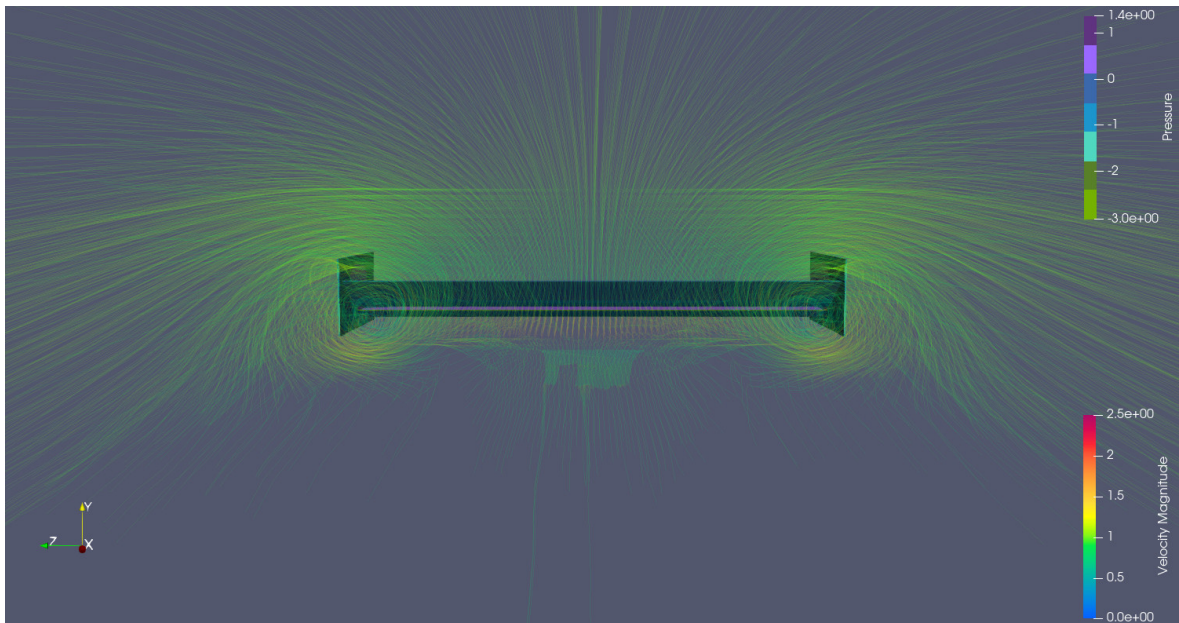


Figura G.14: Visualização 3D do escoamento mostrando os vórtices de ponta de asa do caso h70

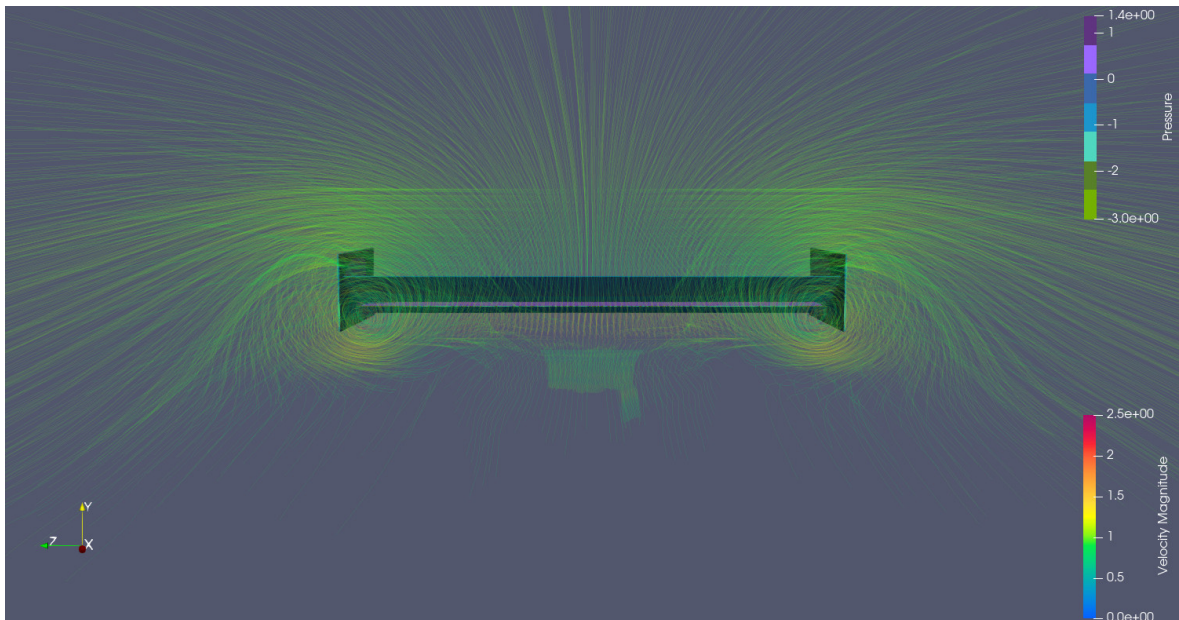


Figura G.15: Visualização 3D do escoamento mostrando os vórtices de ponta de asa do caso h80

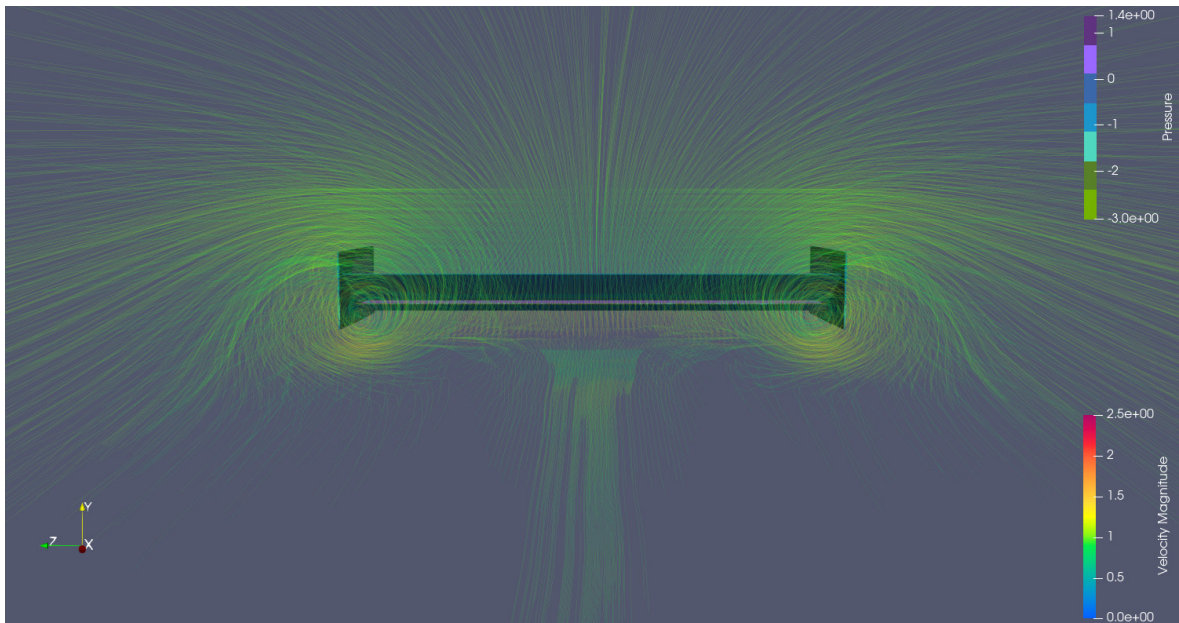


Figura G.16: Visualização 3D do escoamento mostrando os vórtices de ponta de asa do caso h85

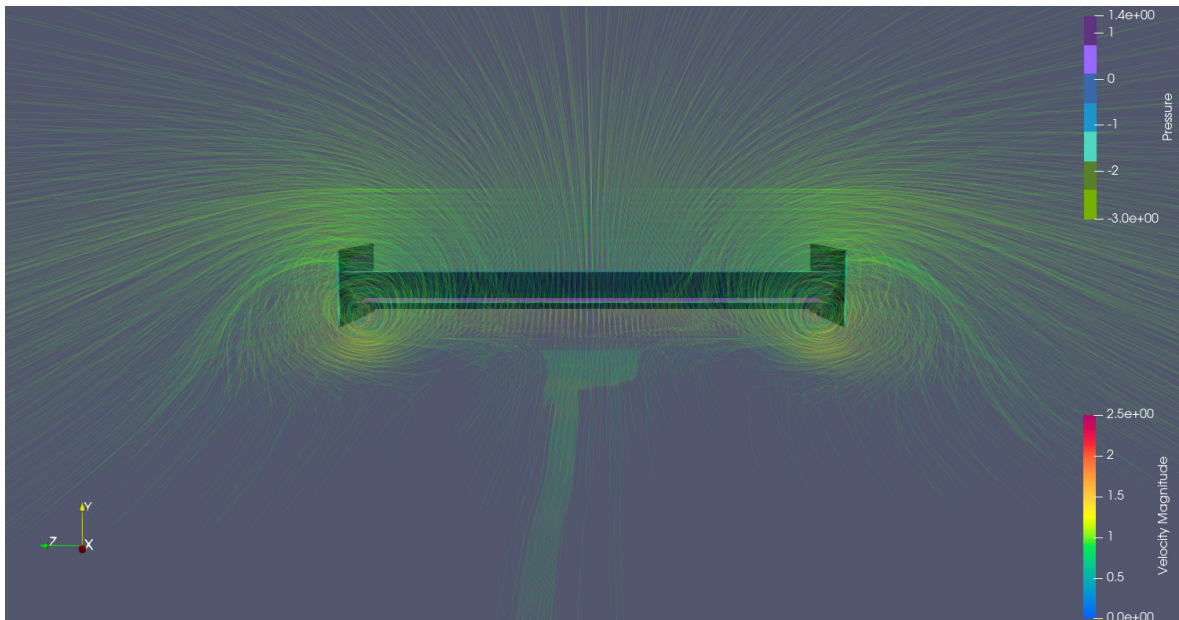


Figura G.17: Visualização 3D do escoamento mostrando os vórtices de ponta de asa do caso h90



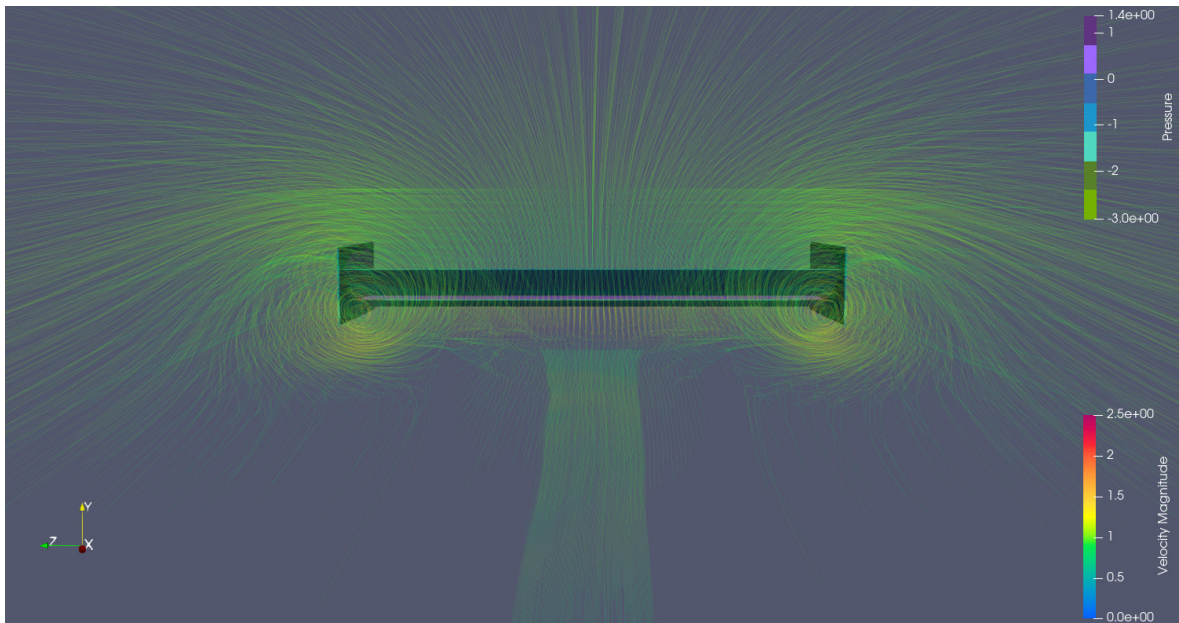


Figura G.18: Visualização 3D do escoamento mostrando os vórtices de ponta de asa do caso h95

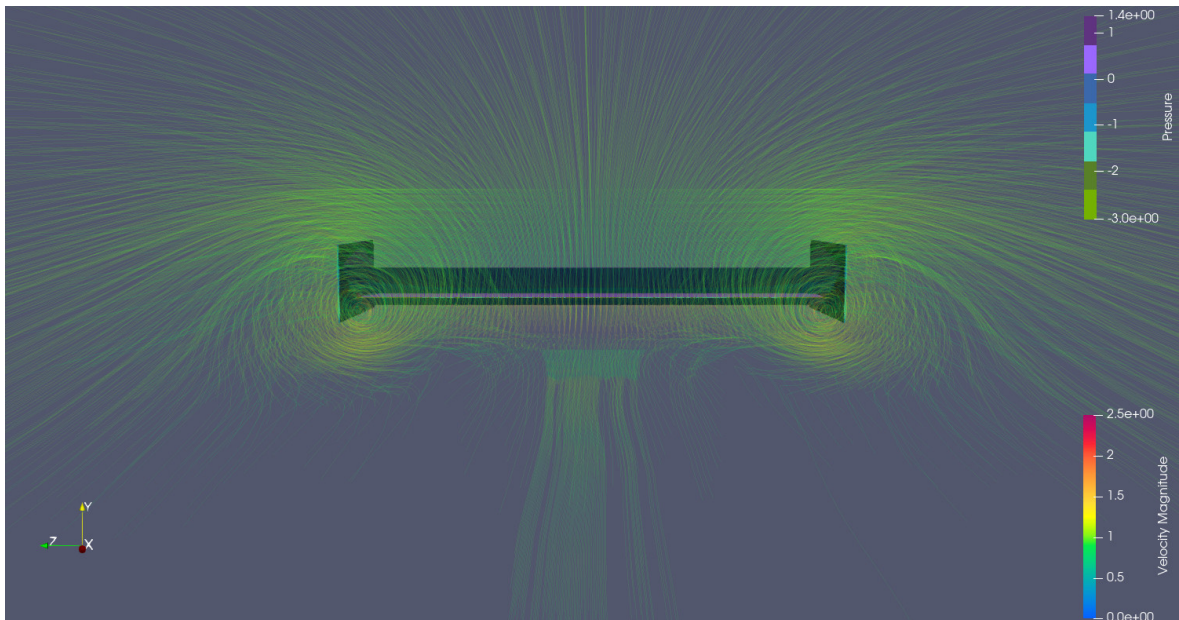


Figura G.19: Visualização 3D do escoamento mostrando os vórtices de ponta de asa do caso h100

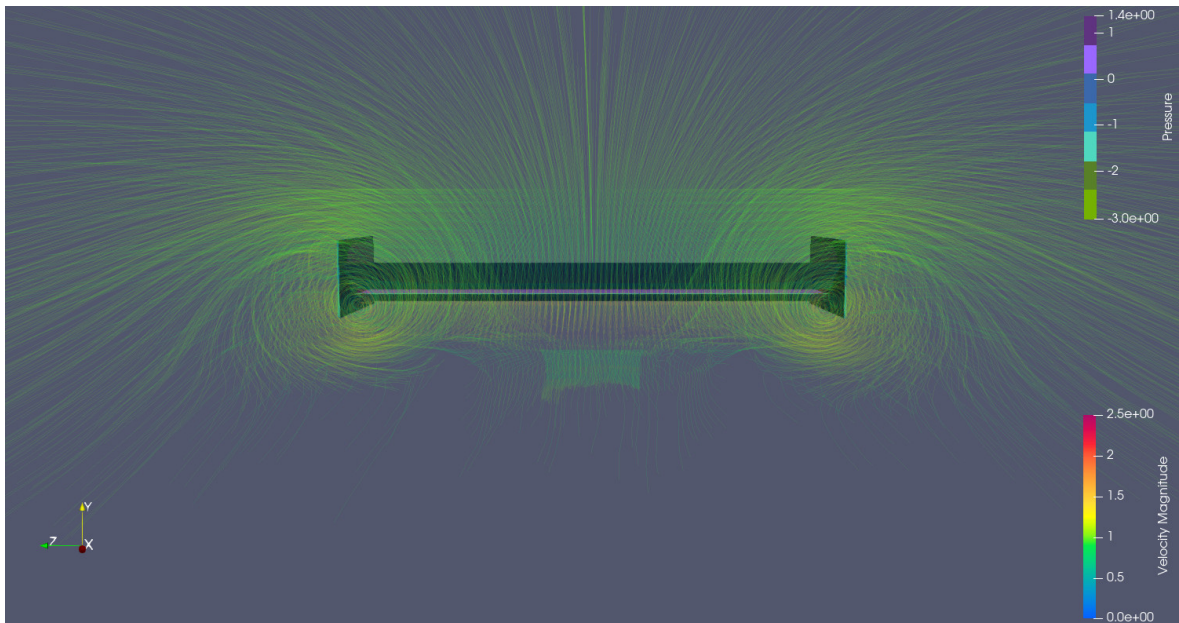


Figura G.20: Visualização 3D do escoamento mostrando os vórtices de ponta de asa do caso h110

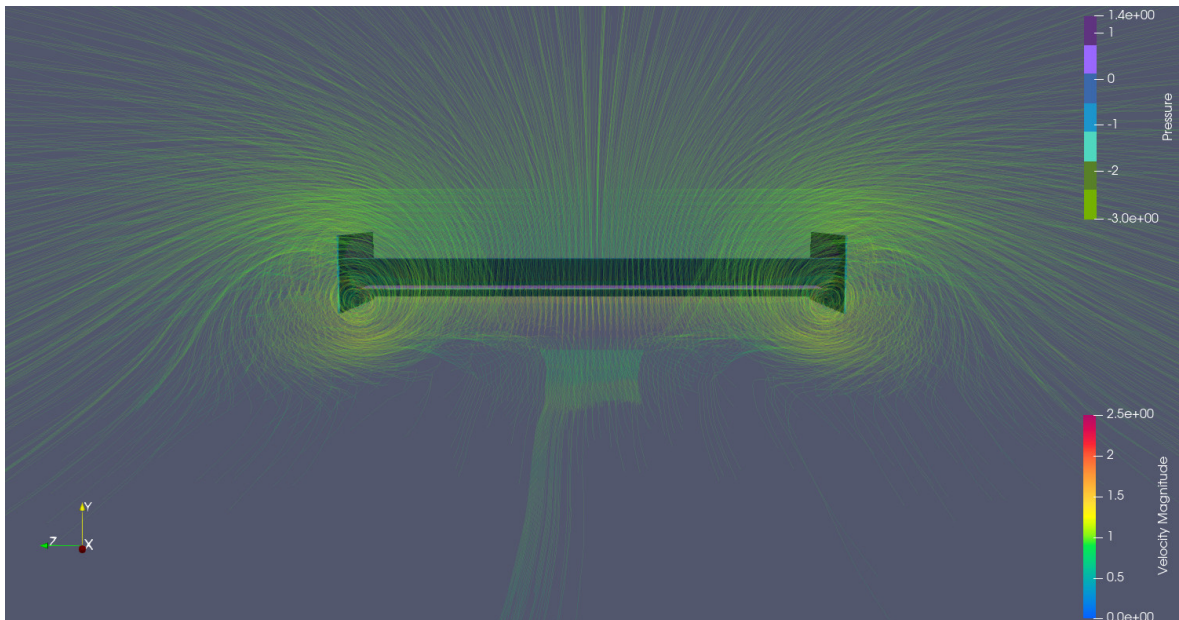


Figura G.21: Visualização 3D do escoamento mostrando os vórtices de ponta de asa do caso h120



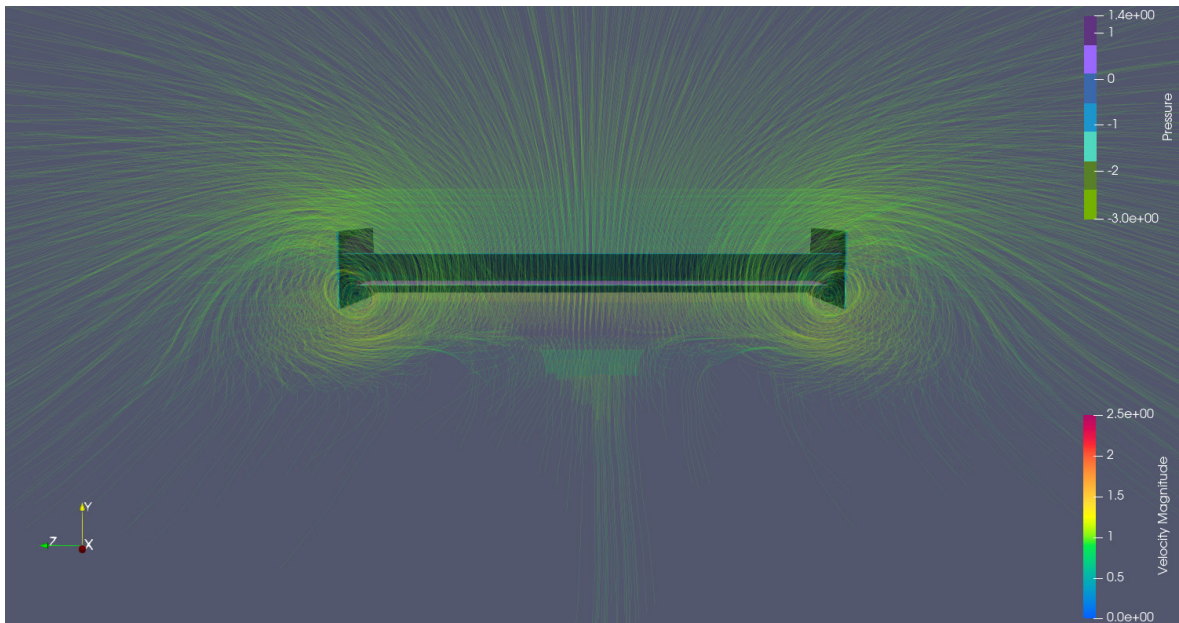


Figura G.22: Visualização 3D do escoamento mostrando os vórtices de ponta de asa do caso h130

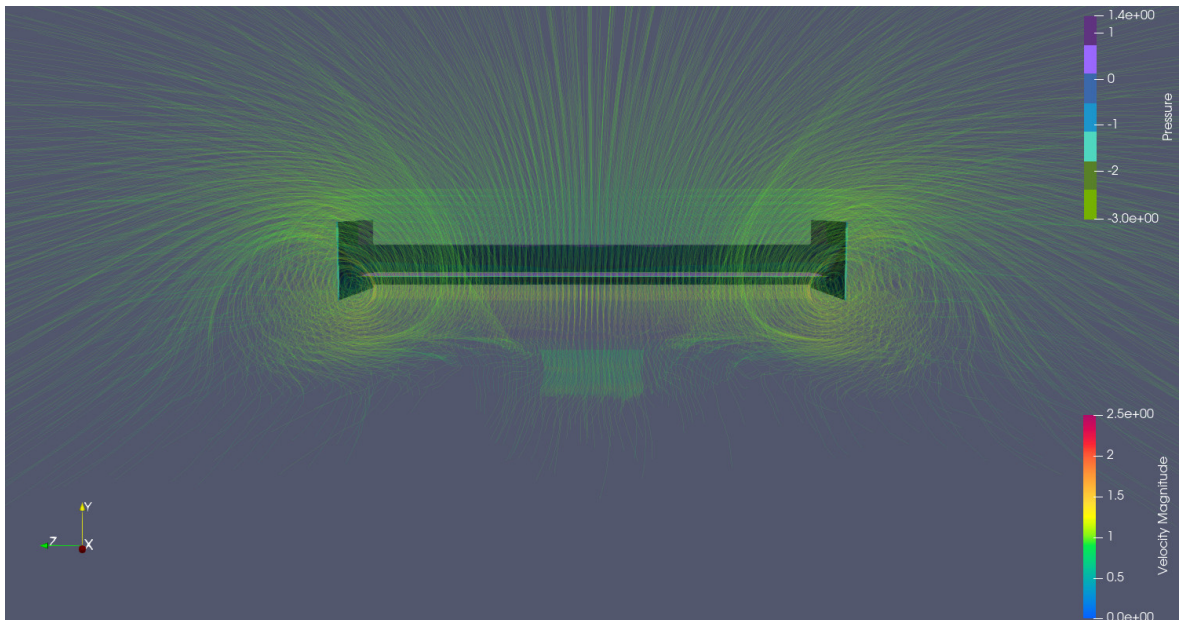


Figura G.23: Visualização 3D do escoamento mostrando os vórtices de ponta de asa do caso h150

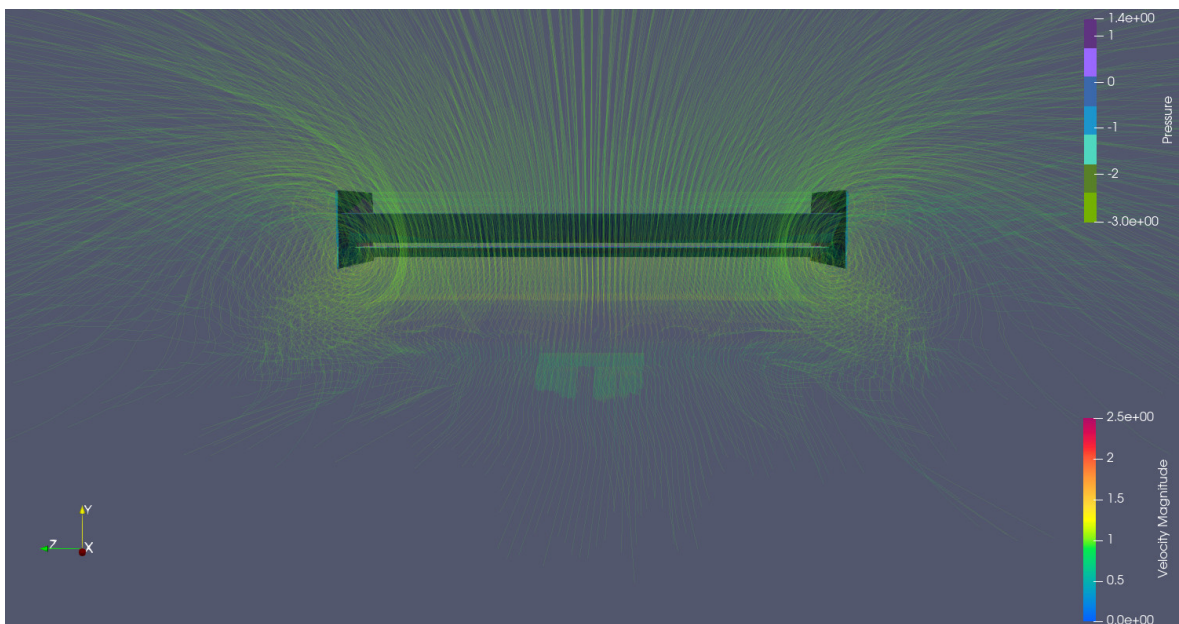


Figura G.24: Visualização 3D do escoamento mostrando os vórtices de ponta de asa do caso h225



# APÊNDICE H

## Visualizações de Pressão - Plano xy



Figura H.1: Visualização de pressão do escoamento no plano xy do caso h17



Figura H.2: Visualização de pressão do escoamento no plano xy do caso h20



Figura H.3: Visualização de pressão do escoamento no plano xy do caso h22



Figura H.4: Visualização de pressão do escoamento no plano xy do caso h25



Figura H.5: Visualização de pressão do escoamento no plano xy do caso h27

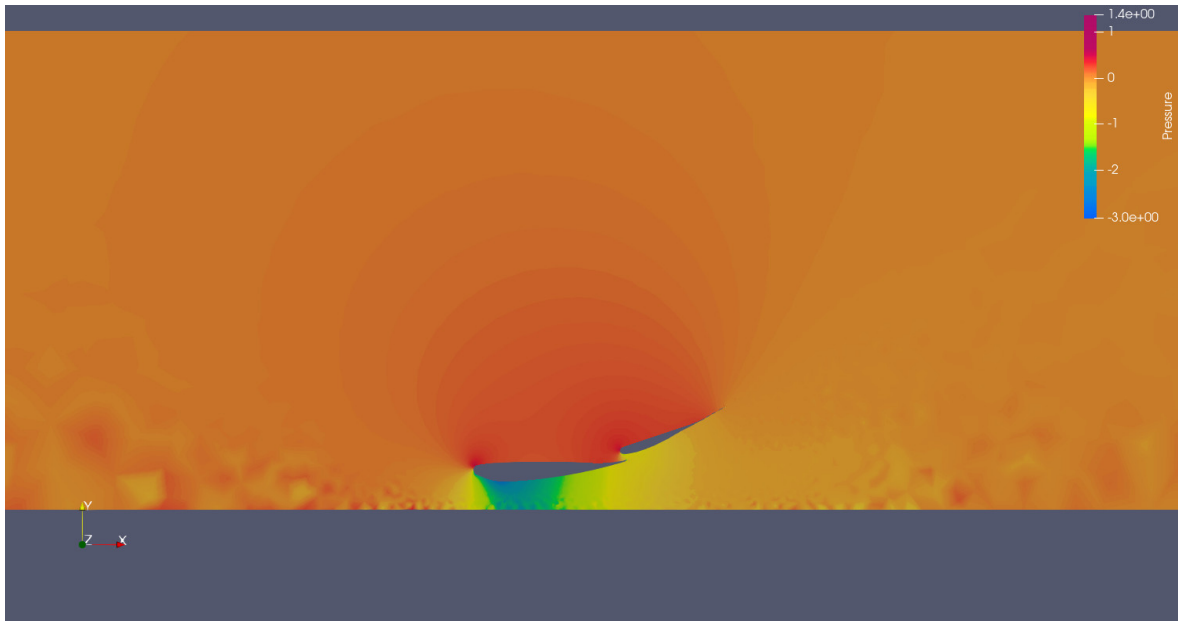


Figura H.6: Visualização de pressão do escoamento no plano xy do caso h30



Figura H.7: Visualização de pressão do escoamento no plano xy do caso h32





Figura H.8: Visualização de pressão do escoamento no plano xy do caso h35



Figura H.9: Visualização de pressão do escoamento no plano xy do caso h37

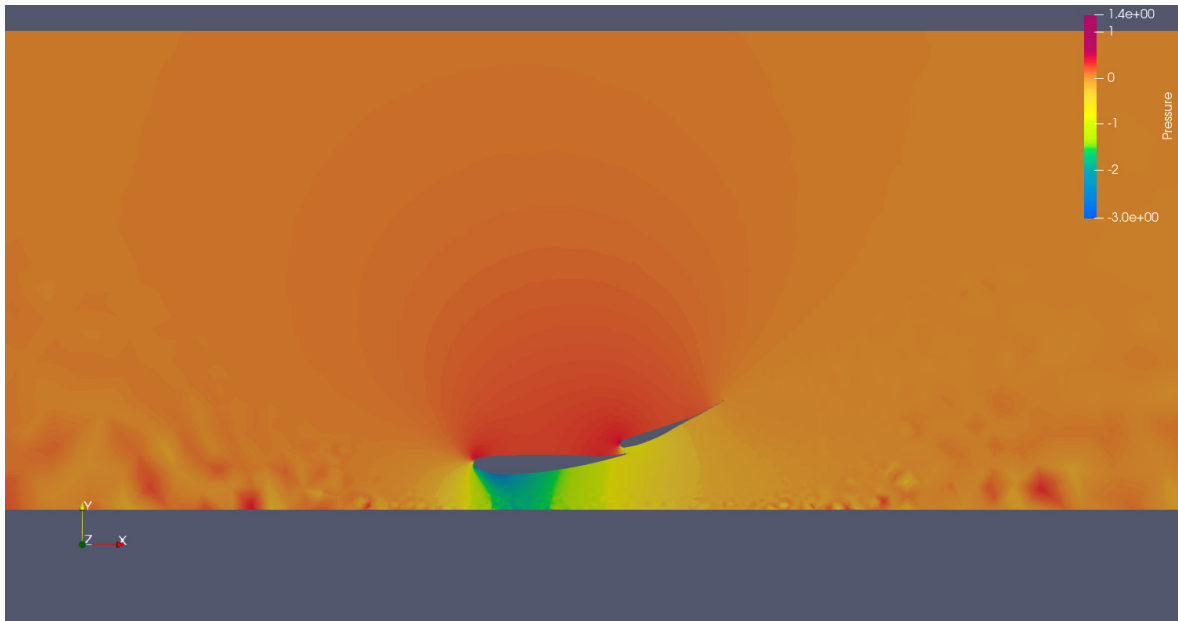


Figura H.10: Visualização de pressão do escoamento no plano xy do caso h40

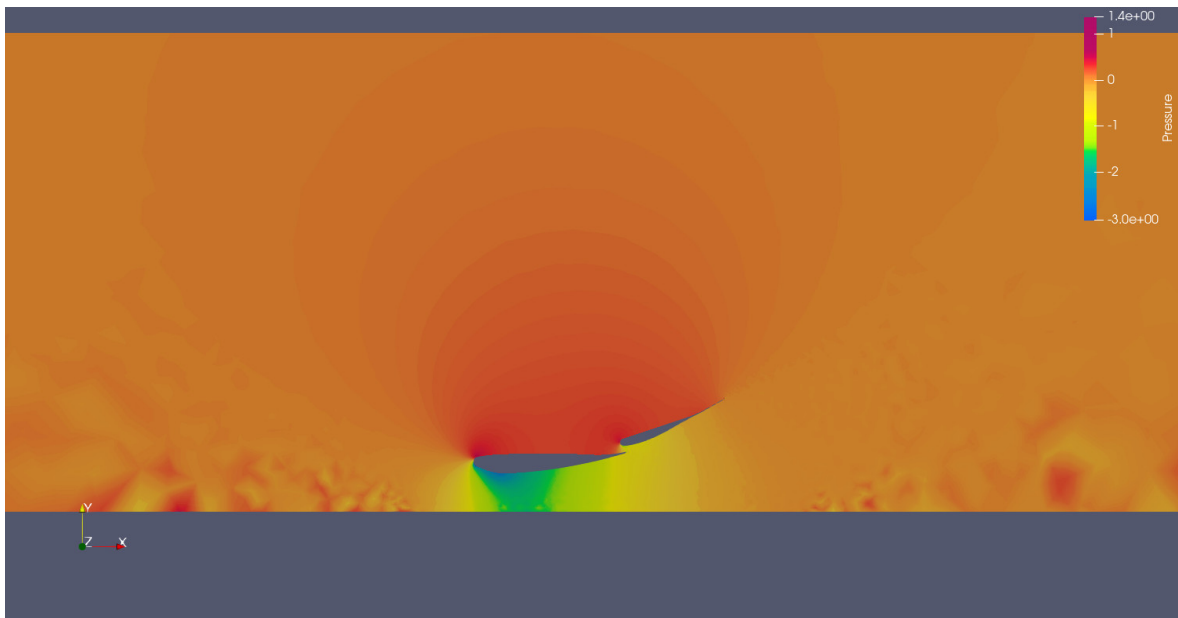


Figura H.11: Visualização de pressão do escoamento no plano xy do caso h45



Figura H.12: Visualização de pressão do escoamento no plano xy do caso h50

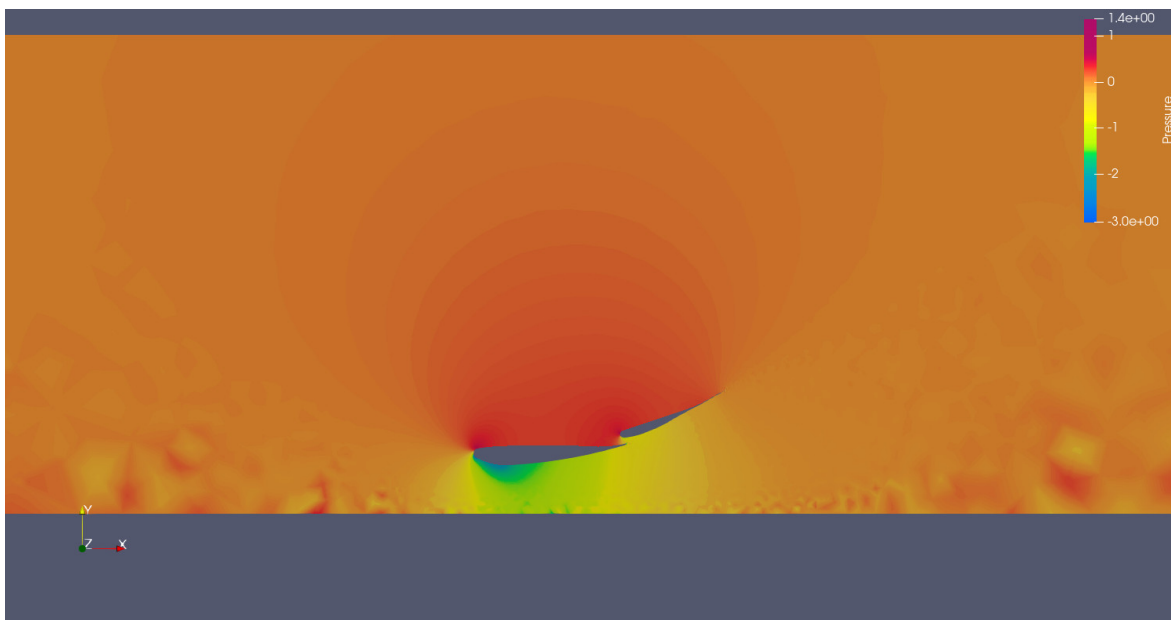


Figura H.13: Visualização de pressão do escoamento no plano xy do caso h60

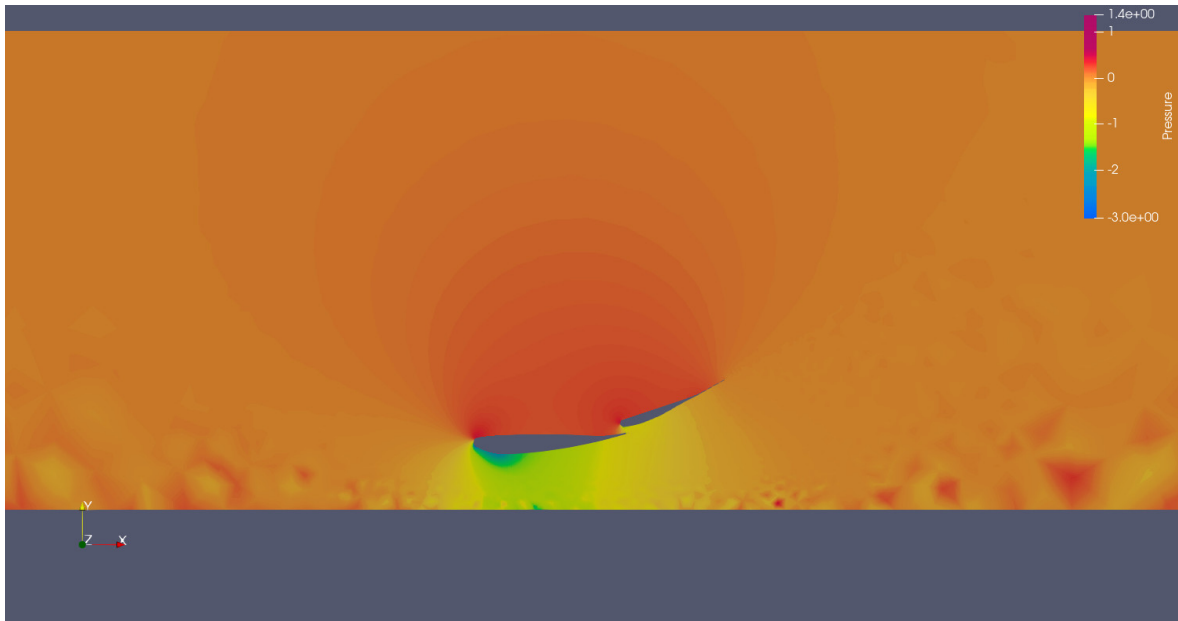


Figura H.14: Visualização de pressão do escoamento no plano xy do caso h70

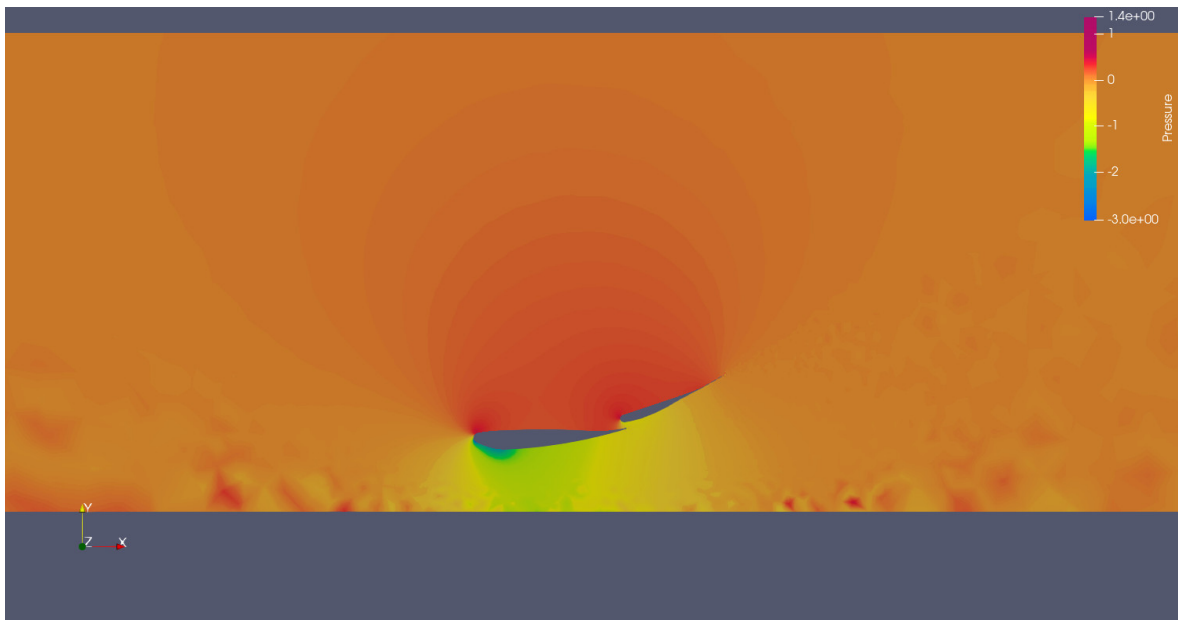


Figura H.15: Visualização de pressão do escoamento no plano xy do caso h80



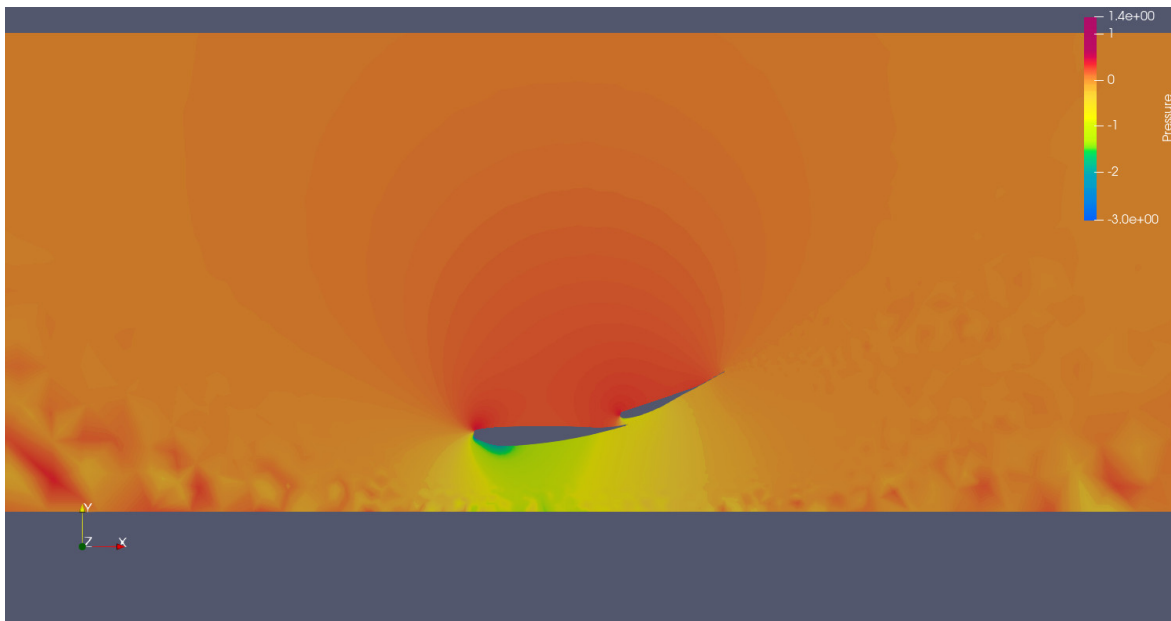


Figura H.16: Visualização de pressão do escoamento no plano xy do caso h85

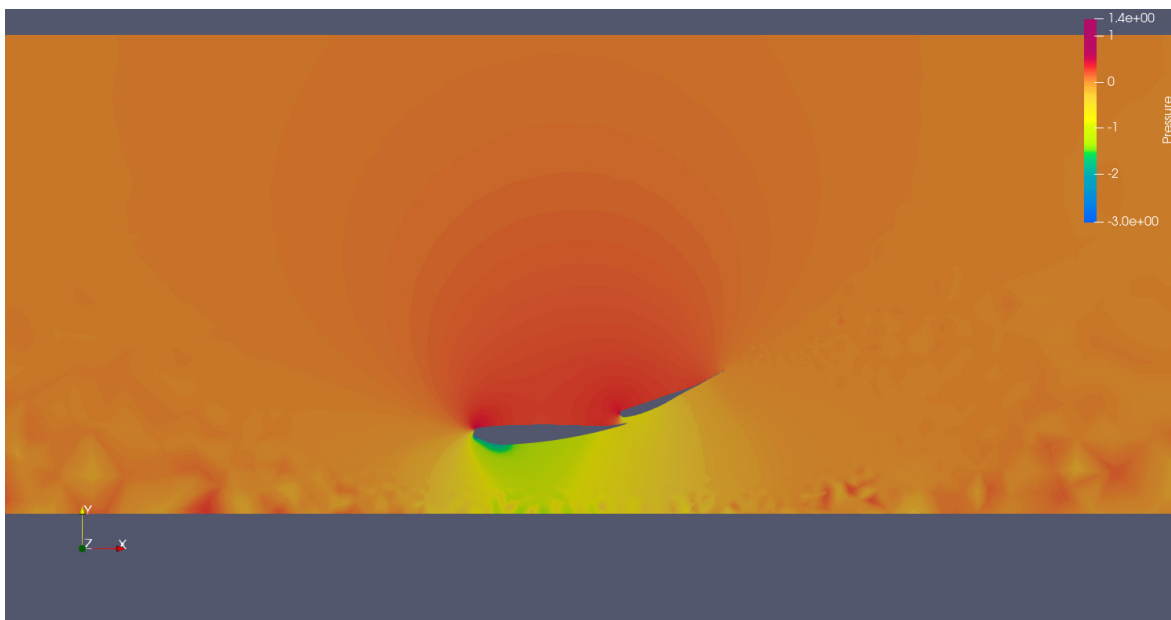


Figura H.17: Visualização de pressão do escoamento no plano xy do caso h90

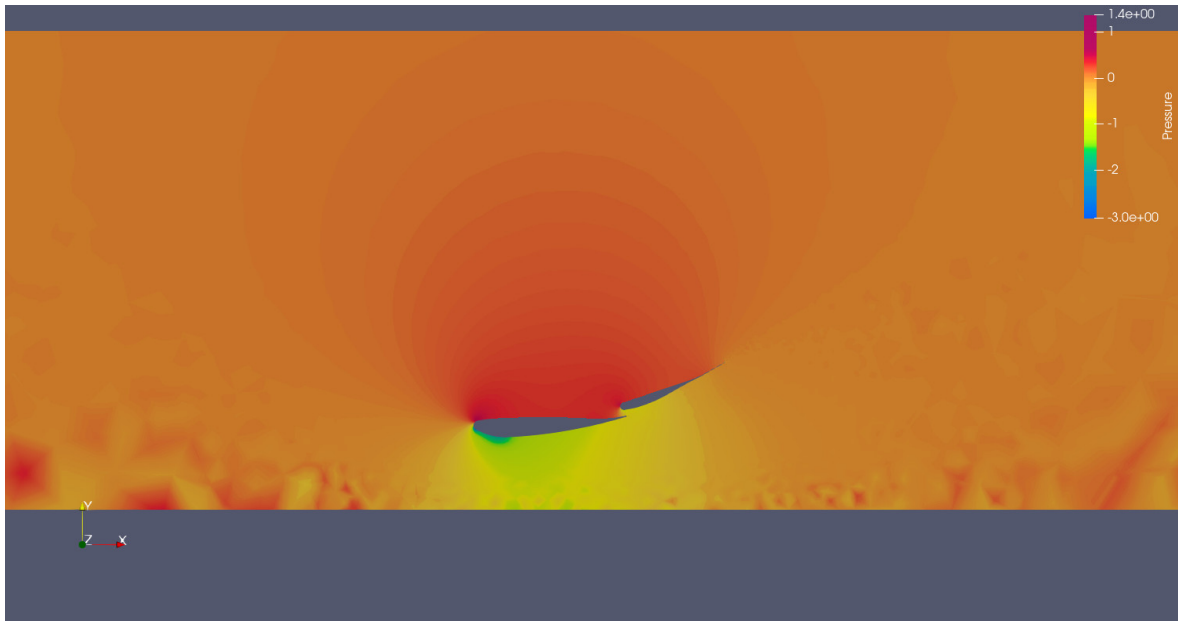


Figura H.18: Visualização de pressão do escoamento no plano xy do caso h95

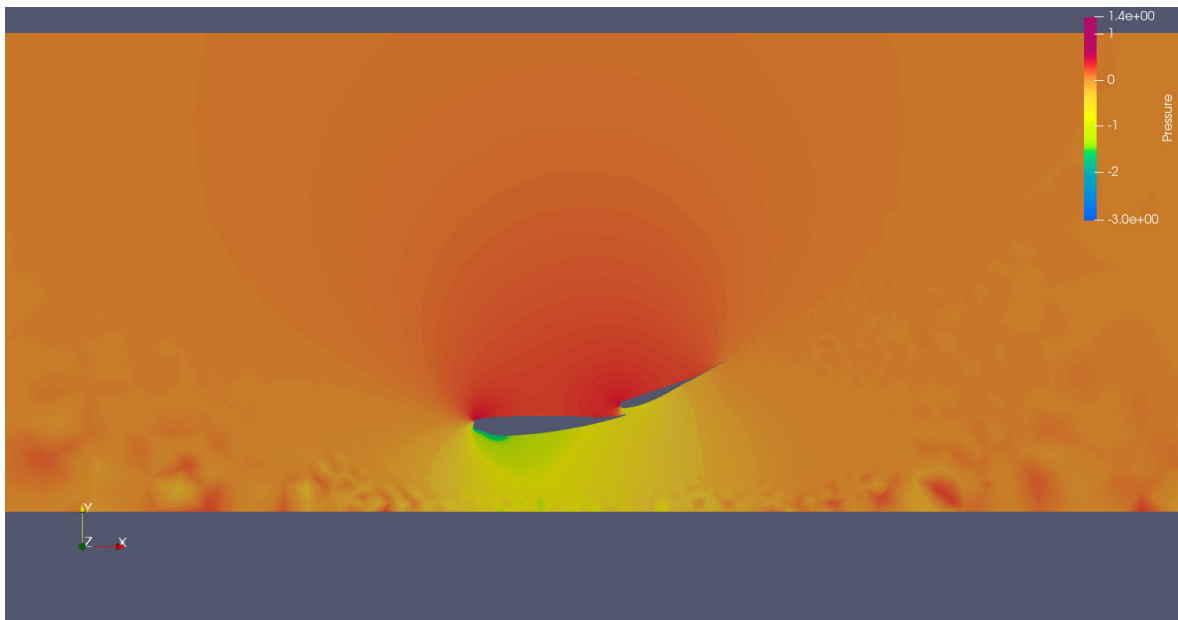


Figura H.19: Visualização de pressão do escoamento no plano xy do caso h100

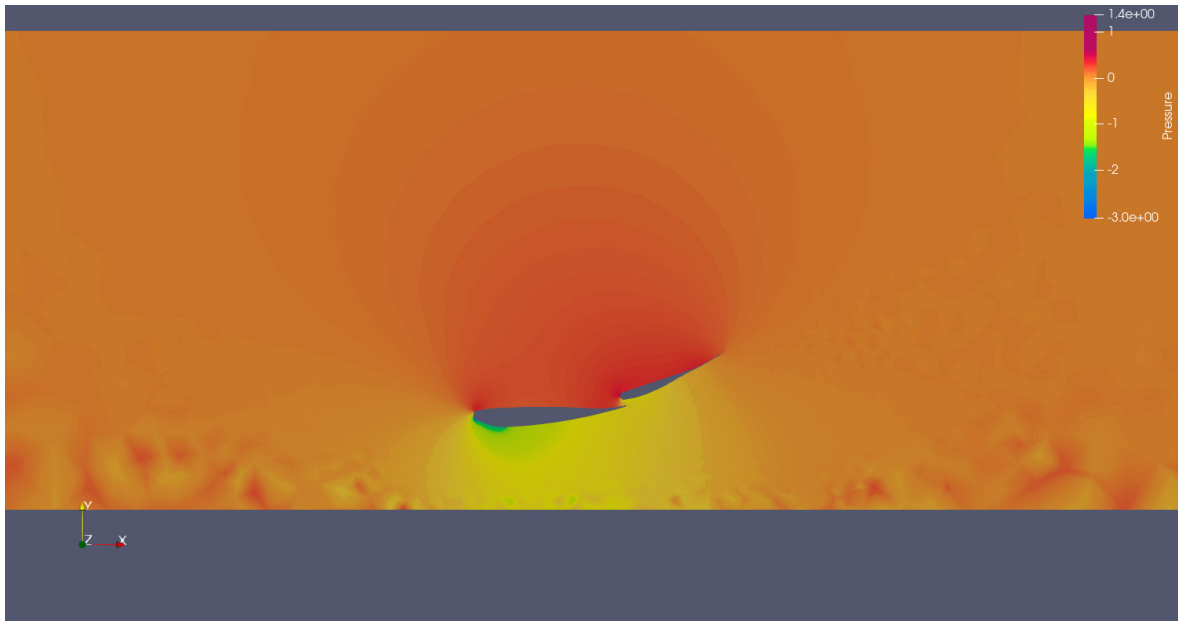


Figura H.20: Visualização de pressão do escoamento no plano xy do caso h110

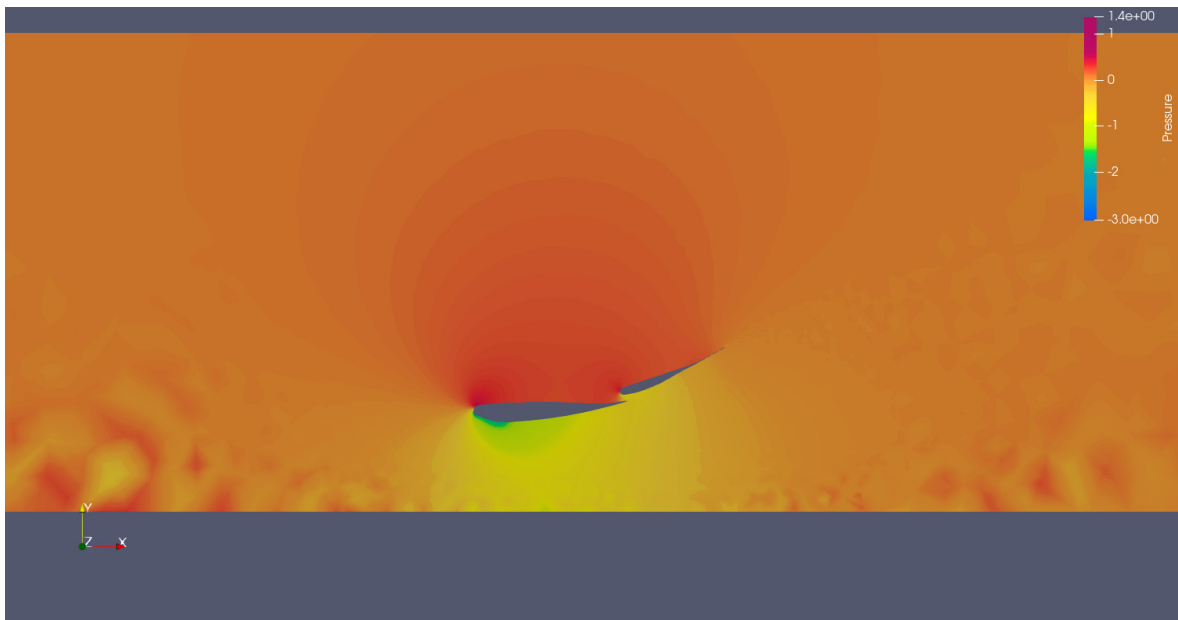


Figura H.21: Visualização de pressão do escoamento no plano xy do caso h120



Figura H.22: Visualização de pressão do escoamento no plano xy do caso h130

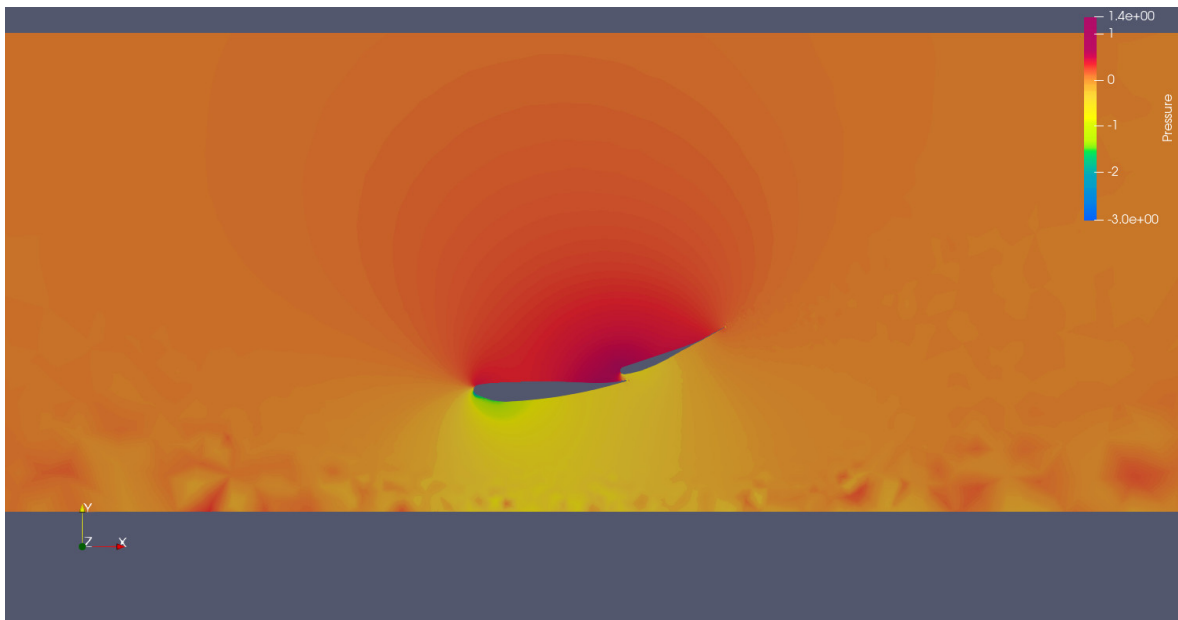


Figura H.23: Visualização de pressão do escoamento no plano xy do caso h150



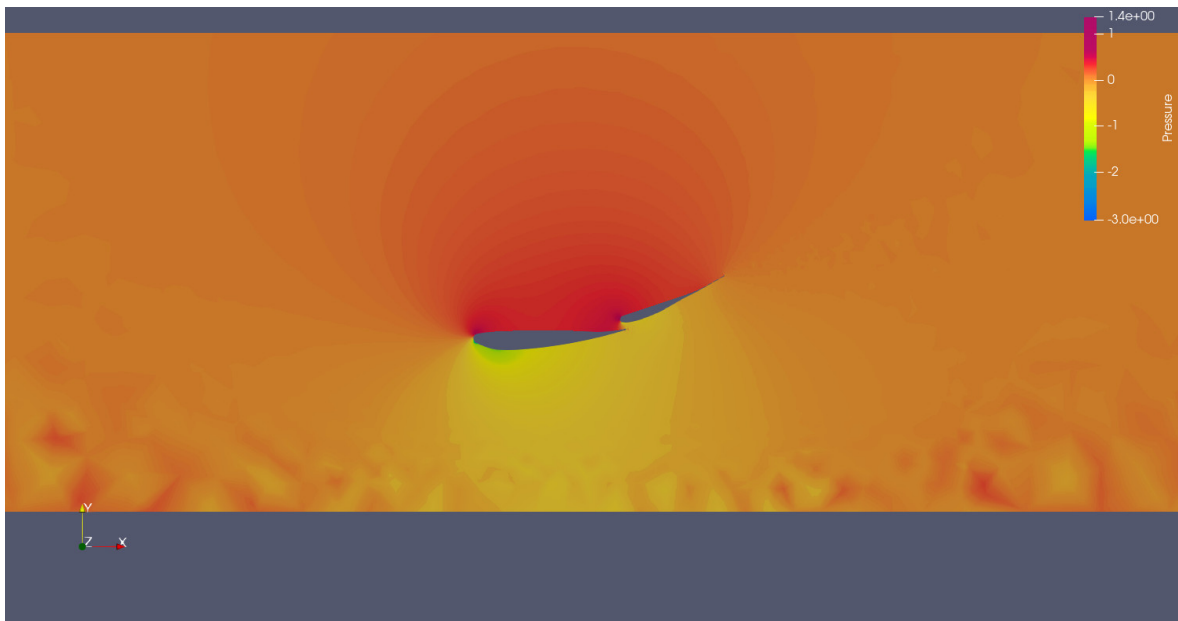


Figura H.24: Visualização de pressão do escoamento no plano xy do caso h225

# APÊNDICE I

## Visualizações de Pressão - Plano yz



Figura I.1: Visualização de pressão do escoamento no plano yz do caso h17



Figura I.2: Visualização de pressão do escoamento no plano yz do caso h20



Figura I.3: Visualização de pressão do escoamento no plano yz do caso h22



Figura I.4: Visualização de pressão do escoamento no plano yz do caso h25



Figura I.5: Visualização de pressão do escoamento no plano yz do caso h27





Figura I.6: Visualização de pressão do escoamento no plano yz do caso h30



Figura I.7: Visualização de pressão do escoamento no plano yz do caso h32



Figura I.8: Visualização de pressão do escoamento no plano yz do caso h35



Figura I.9: Visualização de pressão do escoamento no plano yz do caso h37



Figura I.10: Visualização de pressão do escoamento no plano yz do caso h40



Figura I.11: Visualização de pressão do escoamento no plano yz do caso h45



Figura I.12: Visualização de pressão do escoamento no plano yz do caso h50



Figura I.13: Visualização de pressão do escoamento no plano yz do caso h60





Figura I.14: Visualização de pressão do escoamento no plano yz do caso h70



Figura I.15: Visualização de pressão do escoamento no plano yz do caso h80



Figura I.16: Visualização de pressão do escoamento no plano yz do caso h85



Figura I.17: Visualização de pressão do escoamento no plano yz do caso h90



Figura I.18: Visualização de pressão do escoamento no plano yz do caso h95



Figura I.19: Visualização de pressão do escoamento no plano yz do caso h100



Figura I.20: Visualização de pressão do escoamento no plano yz do caso h110



Figura I.21: Visualização de pressão do escoamento no plano yz do caso h120





Figura I.22: Visualização de pressão do escoamento no plano yz do caso h130



Figura I.23: Visualização de pressão do escoamento no plano yz do caso h150



Figura I.24: Visualização de pressão do escoamento no plano yz do caso h225

# APÊNDICE J

## Tabela de Cálculo e Tratamento de Dados dos Perfis de Asa de Zerihan

Para os cálculos de redimensionalização das coordenadas dos pontos das asa frontal de Zerihan (2001), foi desenvolvida uma planilha de cálculos, essa planilha pode ser visualizada [neste link](https://docs.google.com/spreadsheets/d/1ctV5WwAIJTVW95AZgi9hZsi0dV2DGMCVaiOfMzl5h0k/edit?usp=sharing) ou em <https://docs.google.com/spreadsheets/d/1ctV5WwAIJTVW95AZgi9hZsi0dV2DGMCVaiOfMzl5h0k/edit?usp=sharing>.

Main Wing DATA	Main Wing / Chord length				Main Wing					
	Attack Angle (α) [deg]		14.1		Attack Angle (α) [deg]		Order	14.1		
	chord (c) [mm]		1.000		chord (c) [mm]			380.0000		
	Ride Height [mm]		0.000		Ride Height [mm]			0.0000		
Suction surface	Pressure surface	Suction		Pressure		Suction		Pressure		
x/c   y/c	x/c   y/c	x/c	y/c	x	y	x	y	x	y	
0.0000   0.0000	0.0000   0.0000	0.0000	0.0000	0.0000	0.000000000	17.389000000	1	0.000000000	17.389000000	
0.0006   -0.0044	0.0006   0.0006	0.0006	0.0044	0.0047	0.228000000	15.894000000	2	0.228000000	15.894000000	
0.0011   -0.0063	0.0011   0.0064	0.0011	0.0063	0.0064	0.418000000	14.972000000	3	0.418000000	14.972000000	
0.0029   -0.0098	0.0029   0.0102	0.0029	0.0098	0.0102	1.102000000	13.842000000	4	1.102000000	13.842000000	
0.0058   -0.0134	0.0058   0.0136	0.0058	0.0134	0.0136	2.204000000	12.274000000	5	2.204000000	12.274000000	
0.0087   -0.0156	0.0087   0.0159	0.0087	0.0156	0.0159	3.306000000	11.438000000	6	3.306000000	11.438000000	
0.0117   -0.0173	0.0117   0.0176	0.0117	0.0173	0.0176	4.448000000	10.792000000	7	4.448000000	10.792000000	
0.0146   -0.0188	0.0146   0.0184	0.0146	0.0188	0.0184	5.548000000	10.222000000	8	5.548000000	10.222000000	
0.0175   -0.0203	0.0175   0.0194	0.0175	0.0203	0.0194	6.650000000	9.852000000	9	6.650000000	9.852000000	
0.0205   -0.0217	0.0205   0.0194	0.0205	0.0217	0.0194	7.760000000	9.120000000	10	7.760000000	9.120000000	
0.0234   -0.0231	0.0234   0.0194	0.0234	0.0231	0.0194	8.892000000	8.588000000	11	8.892000000	8.588000000	
0.0263   -0.0245	0.0263   0.0204	0.0263	0.0245	0.0204	9.964000000	8.068000000	12	10.070000000	25.118000000	
0.0293   -0.0258	0.0293   0.0208	0.0293	0.0258	0.0208	11.134000000	7.562000000	13	11.210000000	25.270000000	
0.0322   -0.0270	0.0322   0.0212	0.0322	0.0270	0.0212	12.236000000	7.106000000	14	12.312000000	25.422000000	
0.0351   -0.0283	0.0351   0.0217	0.0351	0.0283	0.0217	13.338000000	6.612000000	15	13.452000000	25.612000000	
0.0410   -0.0306	0.0410   0.0225	0.0410	0.0306	0.0225	15.890000000	5.738000000	16	15.656000000	25.616000000	
0.0469   -0.0328	0.0469   0.0232	0.0469	0.0328	0.0232	17.822000000	4.902000000	17	17.898000000	25.618000000	
0.0528   -0.0348	0.0528   0.0239	0.0528	0.0348	0.0239	20.094000000	4.142000000	18	20.140000000	25.648000000	
0.0586   -0.0366	0.0586   0.0245	0.0586	0.0366	0.0245	22.286000000	3.458000000	19	22.382000000	25.678000000	
0.0704   -0.0398	0.0704   0.0256	0.0704	0.0398	0.0256	26.752000000	2.242000000	20	26.866000000	27.094000000	
0.0821   -0.0422	0.0821   0.0265	0.0821	0.0422	0.0265	31.198000000	1.330000000	21	31.312000000	27.438000000	
0.0939   -0.0444	0.0939   0.0272	0.0939	0.0444	0.0272	35.882000000	0.808000000	22	35.796000000	27.702000000	
0.1056   -0.0452	0.1056   0.0278	0.1056	0.0452	0.0278	40.128000000	0.190000000	23	40.242000000	27.930000000	
0.1174   -0.0457	0.1174   0.0282	0.1174	0.0457	0.0282	44.812000000	0.000000000	24	44.726000000	28.082000000	
0.1468   -0.0448	0.1468   0.0293	0.1468	0.0448	0.0293	55.794000000	0.342000000	25	55.898000000	28.500000000	
0.1762   -0.0430	0.1762   0.0303	0.1762	0.0430	0.0303	66.956000000	1.026000000	26	67.070000000	28.880000000	
0.2056   -0.0407	0.2056   0.0310	0.2056	0.0407	0.0310	78.128000000	1.900000000	27	78.242000000	29.148000000	
0.2350   -0.0379	0.2350   0.0314	0.2350	0.0379	0.0314	89.300000000	2.984000000	28	89.414000000	29.288000000	
0.2644   -0.0347	0.2644   0.0316	0.2644	0.0347	0.0316	100.472000000	4.180000000	29	100.586000000	29.374000000	
0.2938   -0.0309	0.2938   0.0315	0.2938	0.0309	0.0315	111.644000000	5.624000000	30	111.758000000	29.336000000	
0.3232   -0.0267	0.3232   0.0311	0.3232	0.0267	0.0311	122.816000000	7.220000000	31	122.892000000	29.184000000	
0.3526   -0.0219	0.3526   0.0305	0.3526	0.0219	0.0305	133.988000000	8.940000000	32	134.094000000	28.956000000	
0.3820   -0.0167	0.3820   0.0295	0.3820	0.0167	0.0295	145.160000000	11.020000000	33	145.236000000	28.576000000	
0.4114   -0.0110	0.4114   0.0283	0.4114	0.0110	0.0283	156.332000000	13.186000000	34	156.408000000	28.120000000	
0.4409   -0.0049	0.4409   0.0268	0.4409	0.0049	0.0268	167.504000000	15.504000000	35	167.580000000	27.550000000	
0.4703   0.0015	0.4703   0.0257	0.4703	0.0015	0.0257	178.676000000	18.050000000	36	178.752000000	27.132000000	
0.4997   0.0090	0.4997   0.0261	0.4997	0.0090	0.0261	189.848000000	20.786000000	37	189.924000000	27.284000000	
0.5291   0.0166	0.5291   0.0262	0.5291	0.0166	0.0262	201.020000000	23.674000000	38	201.096000000	28.082000000	
0.5409   0.0198	0.5409   0.0255	0.5409	0.0198	0.0255	205.542000000	24.890000000	39	205.542000000	28.576000000	

Figura J.1: Planilha de cálculo, dados relativos à Main Wing.

Main Wing DATA	Flap Wing / Chord lenght				Flap Wing					
	Attack Angle (α) [deg]				14.1		Attack Angle (α) [deg]		14.1	
	chord (c) [mm]				1.000		chord (c) [mm]		380.0000	
	Ride Height [mm]				0.000		Ride Height [mm]		0.0000	
Suction surface	Pressure surface				Suction		Order	Pressure		
x/c   y/c   x/c   y/c	x/c	y/c	x/c	y/c	x	y		x	y	
0.5643   0.0757   0.5643   0.0757	0.5643	0.0757	0.5643	0.0757	214.434	48.1320	1	214.434	48.1320	
0.5687   0.0633   0.5720   0.0663	0.5687	0.0633	0.5720	0.0663	216.108	41.4200	2	217.36	50.1600	
0.5744   0.0691   0.5785   0.0686	0.5744	0.0691	0.5785	0.0686	218.272	40.2040	3	219.83	51.0340	
0.5805   0.0597   0.5849   0.0608	0.5805	0.0597	0.5849	0.0608	220.59	40.0520	4	222.262	51.8700	
0.5869   0.0605   0.5914   0.0629	0.5869	0.0605	0.5914	0.0629	222.984	40.3590	5	224.732	52.6680	
0.5931   0.0614   0.5979   0.0651	0.5931	0.0614	0.5979	0.0651	225.378	40.6980	6	227.202	53.5040	
0.6248   0.0678   0.6302   0.1059	0.6248	0.0678	0.6302	0.1059	237.424	43.1300	7	239.478	57.6080	
0.6570   0.0779   0.6625   0.1167	0.6570	0.0779	0.6625	0.1167	249.86	46.9880	8	251.75	61.7120	
0.6896   0.0908   0.6948   0.1275	0.6896	0.0908	0.6948	0.1275	262.048	51.8700	9	264.024	65.8160	
0.7240   0.1065   0.7271   0.1383	0.7240	0.1065	0.7271	0.1383	275.12	57.8360	10	276.298	69.9200	
0.7573   0.1249   0.7594   0.1491	0.7573	0.1249	0.7594	0.1491	287.774	64.8280	11	288.572	74.0240	
0.7895   0.1434   0.7918   0.1600	0.7895	0.1434	0.7918	0.16	300.01	71.8580	12	300.884	78.1660	
0.8229   0.1617   0.8241   0.1709	0.8229	0.1617	0.8241	0.1709	312.702	78.8120	13	313.158	82.3080	
0.8566   0.1789   0.8567   0.1836	0.8566	0.1789	0.8567	0.1836	325.508	85.3480	14	325.546	87.1340	
0.8895   0.1967   0.8897   0.1996	0.8895	0.1967	0.8897	0.1996	338.01	92.1120	15	338.088	93.2140	
0.9230   0.2158   0.9232   0.2186	0.9230	0.2158	0.9232	0.2186	350.74	99.3700	16	350.816	100.4340	
0.9564   0.2345   0.9567   0.2372	0.9564	0.2345	0.9567	0.2372	363.432	106.4760	17	363.546	107.5020	
0.9669   0.2402   0.9670   0.2429	0.9669	0.2402	0.9670	0.2429	367.422	108.8420	18	367.48	109.8880	

Figura J.2: Planilha de cálculo, dados relativos à Flap Wing.



# APÊNDICE K

## Códigos em Python Desenvolvidos para Análise dos Dados de Simulação

### K.1 Monitorador e Gerador de Gráficos dos Coeficientes Aerodinâmicos ao Longo das Iterações

Listagem K.1: history\_scraper.py

```
1 import os
2 import csv
3 import numpy as np
4 import matplotlib.pyplot as plt
5
6
7 # Important folder PATHs
8 print('Aonde esse script ta rodando?')
9 print('1. PC do Matheus')
10 print('2. Cluster Amadea')
11 print("")
12 machine = input()
13 print("")
14 print("
    #####")
15 if machine == '1':
16     base_folder = r'C:\Users\mathe\Documents\Meus Arquivos\UnB\Projeto de
        Pesquisa\Cluster Mirror\pg2'
17     history_plotter_fig_plain_path = r'C:\Users\mathe\Documents\Meus
        Arquivos\UnB\Projeto de Pesquisa\Cluster Mirror\pg2\pg_files\
        results\History Plotter Figs\Plain'
18     history_plotter_fig_full_path = r'C:\Users\mathe\Documents\Meus
        Arquivos\UnB\Projeto de Pesquisa\Cluster Mirror\pg2\pg_files\
```

```

    results\History Plotter Figs\Full'
19     simulating_folder_name = r'Simulating'
20     results_file_path = r'C:\Users\mathe\Documents\Meus Arquivos\UnB\
    Projeto de Pesquisa\Cluster Mirror\pg2\pg_files\results'
21 else:
22     base_folder = r'/home/matheus/pg2'
23     history_plotter_fig_plain_path = r'/home/matheus/pg2/pg_files/results
    /History Plotter Figs/Plain'
24     history_plotter_fig_full_path = r'/home/matheus/pg2/pg_files/results/
    History Plotter Figs/Full'
25     simulating_folder_name = r'Simulating'
26     results_file_path = r'/home/matheus/pg2/pg_files/results'
27
28     folders_to_check = [17, 20, 22, 25, 27, 30, 32, 35, 37, 40, 45, 50, 60,
    70, 80, 85, 90, 95, 100, 110, 120, 130, 150, 225, 'inf']
29
30     # Separate integers and strings
31     integers = []
32     strings = []
33     for item in folders_to_check:
34         if isinstance(item, int):
35             integers.append(item)
36         else:
37             strings.append(item)
38
39     # Sort the integer list in ascending order
40     sorted_integers = sorted(integers)
41
42     # Sort the string list
43     sorted_strings = sorted(strings)
44
45     # Concatenate the sorted integer list with the sorted string list
46     sorted_folders = sorted_strings + sorted_integers
47
48     print('')
49
50
51     for case_number in sorted_folders:
52         folder_name = f'h{case_number}'
53         mesh_filename = f'mesh_h{case_number}.su2'
54         output_filename = 'output.dat'
55         folder_path = os.path.join(base_folder, folder_name)
56         output_filepath = os.path.join(folder_path, output_filename)
57         history_file = os.path.join(folder_path, 'history.csv')
58         mesh_file = os.path.join(folder_path, mesh_filename)
59
60         print(folder_name)
61         print("")
62         print(f"Checking folder: {folder_path}")

```

```

63
64 if os.path.exists(history_file):
65
66     target_string = "grid points before partitioning"
67
68     with open(output_filepath, "r") as file:
69         lines = file.readlines()
70         print(" ")
71         print(f"Checking for mesh file: {mesh_filename}")
72         if os.path.exists(mesh_file):
73             print(f"Mesh file: {mesh_filename} found in folder")
74         else:
75             print('Mesh file not found')
76         for lines_in_output, line in enumerate(lines):
77             if target_string in line:
78                 if lines_in_output + 1 < len(lines):
79                     next_line = lines[lines_in_output + 1]
80                     volume_mesh_elements = int(next_line.split(" ")
81                                             [0])
82                     print(f'Quantidade de volumes na malha: {
83                             volume_mesh_elements}')
84                 else:
85                     print("Error checking for the number of volume
86                             mesh elements")
87                 break
88             else:
89                 print("Number of volume mesh elements not found in the
90                             file.")
91
92         print(" ")
93         print(f"Checking file: {history_file}")
94         data_history = np.loadtxt(history_file, comments='', delimiter=',')
95
96         # Case name differentiation
97         case_h = str(case_number)
98         casename = f'h{case_number}'
99
100        # Get data from history file
101        iter_data = data_history[:, 2]
102        rms_data = data_history[:, 3]
103        cd_data = data_history[:, 4]
104        csf_data = data_history[:, 5]
105        cl_data = data_history[:, 6]
106
107        # Determine the number of values for mean calculation
108        value_pool = len(iter_data) // 5
109
110        # Calculate mean values from the last fifth part of the values

```

```

107 cd_mean = np.mean(cd_data[-value_pool:])
108 cl_mean = np.mean(cl_data[-value_pool:])
109 csf_mean = np.mean(csf_data[-value_pool:])
110
111 # Calculate the standard deviation
112 cd_std = np.std(cd_data[-value_pool:])
113 cl_std = np.std(cl_data[-value_pool:])
114 csf_std = np.std(csf_data[-value_pool:])
115
116 # Determine the number of iterations made and to make
117 iter_made = np.max(iter_data)+1
118 iter_max = 15000
119
120 # Determine the progress percentage made
121 sim_prog = (iter_made/iter_max) * 100
122 if sim_prog > 100:
123     sim_prog = 100
124
125 # Defining the static loading bar function
126 def print_loading_bar(progress):
127     bar_width = 45
128     filled_width = int(bar_width * progress / 100)
129     remaining_width = bar_width - filled_width
130     loading_bar = '[' + '█' * filled_width + ' ' *
131         remaining_width + ']'
132     print(loading_bar)
133
134 # Print Log
135 print('Iterations history file: history.csv found in folder')
136 print(f'{iter_made:.0f} iterations made')
137 print(f'{sim_prog:.2f}% done with the simulation based on
138     expected {iter_max:.0f} iterations')
139 print(loading_bar(sim_prog))
140 print(f'CD: {cd_mean:.5f}')
141 print(f'CL: {cl_mean:.5f}')
142 print(f'CSF: {csf_mean:.5f}')
143 print(f'CD: {cd_std:.5f}')
144 print(f'CL: {cl_std:.5f}')
145 print(f'CSF: {csf_std:.5f}')
146 print(f'CD: {cd_std / cd_mean:.5f}')
147 print(f'CL: {cl_std / cl_mean:.5f}')
148 print(f'CSF: {csf_std / csf_mean:.5f}')
149 print(f'CD: {cd_std_percent:.5f}')
150 print(f'CL: {cl_std_percent:.5f}')
151 print(f'CSF: {csf_std_percent:.5f}')
152 # Calculate the standard deviation as a percentage of the mean
153 cd_std_percent = np.abs((cd_std / cd_mean) * 100)

```



```

154     cl_std_percent = np.abs((cl_std / cl_mean) * 100)
155     csf_std_percent = np.abs((csf_std / csf_mean) * 100)
156
157     print('Desvio Padrão em formato percentual absoluto em relação
158           à média:')
159     print(f'CD: {cd_std_percent:.2f}%')
160     print(f'CL: {cl_std_percent:.2f}%')
161     print(f'CSF: {csf_std_percent:.2f}%')
162     print("")
163     print("
164           #####
165           ")
166     print("")
167
168     # Find the index of the data point where the mean is calculated
169     mean_index = len(iter_data) - value_pool
170
171     #####
172     ##### Plotting values #####
173     #####
174
175     # Plotting CD and CL
176     fig, ax1 = plt.subplots()
177
178     # Plotting CD data
179     color_cd = 'tab:red'
180     ax1.set_xlabel('Iterações')
181     ax1.set_ylabel('CD', color=color_cd)
182     ax1.plot(iter_data, cd_data, color=color_cd)
183     ax1.tick_params(axis='y', labelcolor=color_cd)
184
185     # Centering the CD axis on zero
186     cd_max = np.max(np.abs(cd_data))
187     ax1.set_ylim(-cd_max, cd_max)
188
189     # Creating a second y-axis for CL data
190     ax2 = ax1.twinx()
191
192     # Plotting CL data
193     color_cl = 'tab:blue'
194     ax2.set_ylabel('CL', color=color_cl)
195     ax2.plot(iter_data, cl_data, color=color_cl)
196     ax2.tick_params(axis='y', labelcolor=color_cl)
197
198     # Centering the CL axis on zero
199     cl_max = np.max(np.abs(cl_data))
200     ax2.set_ylim(-cl_max, cl_max)
201
202     # Creating a third y-axis for CSF data

```

```

200     ax3 = ax1.twinx()
201
202     # Plotting CSF data
203     color_csf = 'tab:green'
204     ax3.spines['right'].set_position(('outward', 60)) # Adjust
           position of the third y-axis
205     ax3.set_ylabel('CSF', color=color_csf)
206     ax3.plot(iter_data, csf_data, color=color_csf)
207     ax3.tick_params(axis='y', labelcolor=color_csf)
208
209     # Centering the CSF axis on zero
210     csf_max = np.max(np.abs(csf_data)*30)
211     ax3.set_ylim(-csf_max, csf_max)
212
213     # Adding a legend
214     lines = [plt.Line2D([], [], color=color_cd, label='CD'),
215             plt.Line2D([], [], color=color_cl, label='CL'),
216             plt.Line2D([], [], color=color_csf, label='CSF')]
217     plt.legend(handles=lines, loc='upper left')
218
219     # Displaying the mean values in a box
220     mean_box_text = f'CD MÃdio: {cd_mean:.5f}\nCL MÃdio: {cl_mean
           :.5f}\nCSF MÃdio: {csf_mean:.5f}'
221     plt.text(0.98, 0.95, mean_box_text, transform=ax1.transAxes,
222             verticalalignment='top', horizontalalignment='right',
223             bbox=dict(facecolor='white', edgecolor='black', boxstyle=
           'round,pad=0.3'))
224
225     # Adding a vertical line at the mean index
226     plt.axvline(x=iter_data[mean_index], color='grey', linestyle='-.')
           )
227     plt.axvline(x=0, color='black', linestyle='-')
228
229     # Setting the title and subtitle with enlarged font size
230     plt.title('EvoluÃo de CD, CL e CSF ao longo das iteraÃes',
           fontsize=15)
231     plt.suptitle(f'Caso {casename}', y=0.95, fontsize=30, fontweight=
           'bold')
232
233     # Show plot grid with vertical lines and bolden zero line
234     plt.grid(True, axis='both', which='major', linestyle='--',
           linewidth=0.5)
235     ax1.axhline(0, color='k', linewidth=1.5)
236     ax2.axhline(0, color='k', linewidth=1.5)
237     ax3.axhline(0, color='k', linewidth=1.5)
238
239     # Adjusting layout
240     plt.tight_layout(rect=[0, 0, 1.1, 1])
241

```

```

242     # Set the figure size
243     fig.set_size_inches(12, 5)
244
245     # Save the plot as a PNG file with casename in the file name
246     filename = f"convergence_CL_CD_CSF_{casename}.png"
247     #history_plotter_fig_plain_path = r'C:\Users\mathe\Documents\Meus
        Arquivos\UnB\Projeto de Pesquisa\Cluster Mirror\pg2\pg_files\
        results\History Plotter Figs\Plain'
248     #simulating_folder_name = r'Simulating'
249     if sim_prog >= 100:
250         save_fig_param = os.path.join(history_plotter_fig_plain_path,
            filename)
251         plt.savefig(save_fig_param, dpi=300)
252     else:
253         save_fig_param = os.path.join(history_plotter_fig_plain_path,
            simulating_folder_name, filename)
254         plt.savefig(save_fig_param, dpi=300)
255
256
257     # Displaying the plot
258     #plt.show()
259     plt.close()
260
261
262     #####
263     ##### Full Technincal Plot #####
264     #####
265
266     # Plotting CD and CL
267     fig, ax1 = plt.subplots()
268
269     # Plotting CD data
270     color_cd = 'tab:red'
271     ax1.set_xlabel('Iterações')
272     ax1.set_ylabel('CD', color=color_cd)
273     ax1.plot(iter_data, cd_data, color=color_cd)
274     ax1.tick_params(axis='y', labelcolor=color_cd)
275
276     # Centering the CD axis on zero
277     cd_max = np.max(np.abs(cd_data))
278     ax1.set_ylim(-cd_max, cd_max)
279
280     # Creating a second y-axis for CL data
281     ax2 = ax1.twinx()
282
283     # Plotting CL data
284     color_cl = 'tab:blue'
285     ax2.set_ylabel('CL', color=color_cl)
286     ax2.plot(iter_data, cl_data, color=color_cl)

```

```

287     ax2.tick_params(axis='y', labelcolor=color_cl)
288
289     # Centering the CL axis on zero
290     cl_max = np.max(np.abs(cl_data))
291     ax2.set_ylim(-cl_max, cl_max)
292
293     # Creating a third y-axis for CSF data
294     ax3 = ax1.twinx()
295
296     # Plotting CSF data
297     color_csf = 'tab:green'
298     ax3.spines['right'].set_position(('outward', 60)) # Adjust
           position of the third y-axis
299     ax3.set_ylabel('CSF', color=color_csf)
300     ax3.plot(iter_data, csf_data, color=color_csf)
301     ax3.tick_params(axis='y', labelcolor=color_csf)
302
303     # Centering the CSF axis on zero
304     csf_max = np.max(np.abs(csf_data)*30)
305     ax3.set_ylim(-csf_max, csf_max)
306
307     # Adding a legend
308     lines = [plt.Line2D([], [], color=color_cd, label='CD'),
309             plt.Line2D([], [], color=color_cl, label='CL'),
310             plt.Line2D([], [], color=color_csf, label='CSF')]
311     plt.legend(handles=lines, loc='upper left')
312
313     # Displaying the mean values in a box
314     mean_box_text = f'CD Médio: {cd_mean:.5f} Âš {cd_std_percent:.2f
           }%\nCL Médio: {cl_mean:.5f} Âš {cl_std_percent:.2f}%\nCSF
           Médio: {csf_mean:.5f} Âš {csf_std_percent:.2f}%'
315     plt.text(0.98, 0.95, mean_box_text, transform=ax1.transAxes,
316             verticalalignment='top', horizontalalignment='right',
317             bbox=dict(facecolor='white', edgecolor='black', boxstyle=
           'round,pad=0.3'))
318
319     # Adding a vertical line at the mean index
320     plt.axvline(x=iter_data[mean_index], color='grey', linestyle='-.'
           )
321     plt.axvline(x=0, color='black', linestyle='-')
322
323     # Setting the title and subtitle with enlarged font size
324     if sim_prog >= 100:
325         plt.title(f'EvoluÃ§Ã£o de CD, CL e CSF ao longo das
           iteraÃ§Ãµes', fontsize=15)
326     else:
327         plt.title(f'EvoluÃ§Ã£o de CD, CL e CSF ao longo de {sim_prog
           :.2f}% das iteraÃ§Ãµes planejadas', fontsize=15)

```



```

328 plt.suptitle(f'Case {casename}', y=0.95, fontsize=30, fontweight=
      'bold')
329
330 # Show plot grid with vertical lines and bolden zero line
331 plt.grid(True, axis='both', which='major', linestyle='--',
      linewidth=0.5)
332 ax1.axhline(0, color='k', linewidth=1.5)
333 ax2.axhline(0, color='k', linewidth=1.5)
334 ax3.axhline(0, color='k', linewidth=1.5)
335
336 # Adjusting layout
337 plt.tight_layout(rect=[0, 0, 1.1, 1])
338
339 # Set the figure size
340 fig.set_size_inches(12, 5)
341
342 # Save the plot as a PNG file with casename in the file name
343 filename = f"convergence_full_CL_CD_CSF_{casename}.png"
344 if sim_prog >= 100:
345     save_fig_param = os.path.join(history_plotter_fig_full_path ,
      filename)
346     plt.savefig(save_fig_param, dpi=300)
347 else:
348     save_fig_param = os.path.join(history_plotter_fig_full_path ,
      simulating_folder_name, filename)
349     plt.savefig(save_fig_param, dpi=300)
350
351
352 # Displaying the plot
353 #plt.show()
354 plt.close()
355
356
357 #####
358 ##### Update the csv results file #####
359 #####
360
361 def update_csv_file(filename, h_input, cl_simulation,
      cd_simulation, q, s, g):
362     rows = []
363     with open(filename, 'r') as file:
364         reader = csv.reader(file)
365         rows = list(reader)
366
367     header = rows[0]
368
369     cl_simulation_index = header.index('*CL - Simulation*')
370     cd_simulation_index = header.index('*CD - Simulation*')
371     cl_zerihan_index = header.index('*CL - Zerihan*')

```

```

372     delta_cl_index = header.index('*Delta CL*')
373     cd_zerihan_index = header.index('*CD - Zerihan*')
374     delta_cd_index = header.index('*Delta CD*')
375     delta_fy_kgf_index = header.index('*Delta Fy (Kgf)*')
376     delta_fx_kgf_index = header.index('*Delta Fx (Kgf)*')
377     delta_fy_pct_index = header.index('*Delta Fy (%)*')
378     delta_fx_pct_index = header.index('*Delta Fx (%)*')
379
380     for row in rows[1:]:
381         h_value = float(row[0])
382         if h_value == float(h_input):
383             row[cl_simulation_index] = f'{cl_simulation:.5f}'
384             row[cd_simulation_index] = f'{cd_simulation:.5f}'
385             delta_cl = np.abs(cl_simulation) - np.abs(float(row[
386                 cl_zerihan_index]))
387             row[delta_cl_index] = f'{delta_cl:.5f}'
388             delta_cd = np.abs(cd_simulation) - np.abs(float(row[
389                 cd_zerihan_index]))
390             row[delta_cd_index] = f'{delta_cd:.5f}'
391             delta_fy_kgf = delta_cl * q * s / g
392             row[delta_fy_kgf_index] = f'{delta_fy_kgf:.5f}'
393             delta_fx_kgf = delta_cd * q * s / g
394             row[delta_fx_kgf_index] = f'{delta_fx_kgf:.5f}'
395             delta_fy_pct = (np.abs(delta_cl))/(np.max([np.abs(
396                 cl_simulation), np.abs(float(row[cl_zerihan_index]
397                 ]))]))
398             row[delta_fy_pct_index] = f'{delta_fy_pct:.5f}'
399             delta_fx_pct = (np.abs(delta_cd))/(np.max([np.abs(
400                 cd_simulation), np.abs(float(row[cd_zerihan_index]
401                 ]))]))
402             row[delta_fx_pct_index] = f'{delta_fx_pct:.5f}'
403
404         with open(filename, 'w', newline='') as file:
405             writer = csv.writer(file)
406             writer.writerows(rows)
407
408     # Updating the results .csv file:
409     results_filename = 'all_results.csv'
410     results_file = os.path.join(results_file_path, results_filename)
411     cl_simulation = cl_mean
412     cd_simulation = cd_mean
413     u_sim = 30 # m/s
414     rho = 1.2886 # kg/m3
415     q = ((u_sim ** 2) * rho) / 2 # (u2 * rho) / 2
416     s = 0.405 # m2
417     g = 9.80665 # m/s2
418
419     update_csv_file(results_file, case_h, cl_simulation,
420         cd_simulation, q, s, g)

```

```

414
415
416
417
418     else:
419         if os.path.exists(mesh_file):
420             print(f"No history file found in folder {folder_name}.")
421             print("")
422             print('This probably means the simulation hasn\'t started yet
423                 , check the queue')
424             print("")
425             print("
426                 #####
427                 ")
428             print("")
429         else:
430             print(f"History file not found.")
431             print("")
432             print('This probably means the mesh file hasn\'t been
433                 uploaded and the simulation hasn\'t started yet')
434             print("")
435             print("
436                 #####
437                 ")
438             print("")
439
440     if machine == 1:
441         end = input('Aperte enter pra sair')
442     else:
443         end = 'end program'

```

## K.2 Gerador de Gráficos Comparativos com os Dados de Zerihan

Listagem K.2: results\_plotter.py

```

1 # Projeto de Graduação 2
2 # Matheus Ribeiro Vidal
3 # 17/06/2023
4
5 import numpy as np
6 import matplotlib.pyplot as plt
7
8 # Load data from the 'all_results.csv' file
9 data_results = np.loadtxt('all_results.csv', comments='*', delimiter=',')
10
11 # Extract the necessary data columns
12 h = data_results[:, 0]

```

```

13 h_c = h[:]/380
14 cl_zerihan = data_results[:, 2]
15 cd_zerihan = data_results[:, 3]
16 cl_simulation = data_results[:, 4] * -1
17 delta_cl = data_results[:, 5]
18 delta_fy_kgf = data_results[:, 6]
19 delta_fy_percent = data_results[:, 7] * 100
20 cd_simulation = data_results[:, 8]
21 delta_cd = data_results[:, 9]
22 delta_fx_kgf = data_results[:, 10]
23 delta_fx_percent = data_results[:, 11] * 100
24
25 # Filter out zero values of cl_simulation
26 nonzero_cl_indices = np.nonzero(cl_simulation)
27 h_c_nonzero = h_c[nonzero_cl_indices]
28 h_nonzero = h[nonzero_cl_indices]
29 cl_simulation_nonzero = cl_simulation[nonzero_cl_indices]
30
31 # Choosing figure sizes and figure folder path
32 fig_wid = 10
33 fig_hei = 12
34 folder_path = 'All Results Figs/'
35
36
37 #
38 #####
39 ##### Plotting the Coefficient Values
40 #####
41 #
42 #####
43
44 # Plot and compare the coefficient of lift (CL) data
45 print('CL')
46 for physics_cutoff_point in [10]:
47     plt.figure(figsize=(fig_wid, fig_hei))
48     plt.plot(h_c, cl_zerihan, 'bs', label='Experimental', markersize=3)
49     plt.plot(h_c, cl_zerihan, color='blue', linestyle='--', linewidth
50             =0.7)
51     plt.plot(h_c_nonzero, cl_simulation_nonzero, 'ro', label='SimulaÃ§Ã£o
52             ', markersize=3)
53     plt.plot(h_c_nonzero, cl_simulation_nonzero, color='red', linestyle='
54             --', linewidth=0.7)
55     # Perform linear regression for up to the physics cutoff point of
56     experimental CL
57     coefficients_exp_cl_1 = np.polyfit(h_c[:physics_cutoff_point],
58                                       cl_zerihan[:physics_cutoff_point], 1)

```



```

52 trendline_exp_cl_1 = np.polyval(coefficients_exp_cl_1, h_c[:
    physics_cutoff_point])
53 # Perform linear regression for up to the pyhsics cutoff point of
    experimental CL
54 coefficients_exp_cl_2 = np.polyfit(h_c[physics_cutoff_point:],
    cl_zerihan[physics_cutoff_point:], 1)
55 trendline_exp_cl_2 = np.polyval(coefficients_exp_cl_2, h_c[
    physics_cutoff_point:])
56 # Plot the two separate trendlines for experimental CL data
57 plt.plot(h_c[:physics_cutoff_point], trendline_exp_cl_1, '-.',
    linewidth=0.8, label='Trend (Experimental - Parte 1)', color='
    lightblue')
58 plt.plot(h_c[physics_cutoff_point:], trendline_exp_cl_2, '-.',
    linewidth=0.8, label='Trend (Experimental - Parte 2)', color='
    lightblue')
59 # Perform linear regression for up to the pyhsics cutoff point of
    simulation CL
60 coefficients_sim_cl_1 = np.polyfit(h_c_nonzero[:physics_cutoff_point
    ], cl_simulation_nonzero[:physics_cutoff_point], 1)
61 trendline_sim_cl_1 = np.polyval(coefficients_sim_cl_1, h_c_nonzero[:
    physics_cutoff_point])
62 # Perform linear regression for up to the pyhsics cutoff point of
    simulation CL
63 coefficients_sim_cl_2 = np.polyfit(h_c_nonzero[physics_cutoff_point
    :], cl_simulation_nonzero[physics_cutoff_point:], 1)
64 trendline_sim_cl_2 = np.polyval(coefficients_sim_cl_2, h_c_nonzero[
    physics_cutoff_point:])
65 # Plot the two separate trendlines for simulation CL data
66 plt.plot(h_c_nonzero[:physics_cutoff_point], trendline_sim_cl_1, '-.'
    , linewidth=0.8, label='Trend (SimulaÃ§Ã£o - Parte 1)', color='
    lightcoral')
67 plt.plot(h_c_nonzero[physics_cutoff_point:], trendline_sim_cl_2, '-.'
    , linewidth=0.8, label='Trend (SimulaÃ§Ã£o - Parte 2)', color='
    lightcoral')
68 # Calculate the midpoint x-coordinate
69 midpoint_x = (h_c[physics_cutoff_point - 1] + h_c[
    physics_cutoff_point]) / 2.0
70 # Plot the vertical line at the midpoint
71 plt.axvline(x=midpoint_x, color='gray', linestyle='--', linewidth
    =0.9, label='Desaparecimento da Bolha de RecirculaÃ§Ã£o')
72 plt.xlabel('h/c')
73 plt.ylabel('CL')
74 plt.xlim(0, 0.6)
75 plt.xticks(np.arange(0, 0.65, 0.05))
76 plt.ylim(1.6, 3.1)
77 plt.yticks(np.arange(1.6, 3.15, 0.05))
78 plt.title('Comparativo dos Resultados das SimulaÃ§Ãµes com os
    Experimentos de Zerihan (2001)')
79 plt.suptitle('Coeficiente de SustentaÃ§Ã£o', y=0.95, fontsize=30,

```

```

    fontweight='bold')
80 plt.legend()
81 plt.grid(True, linestyle=':')
82 plt.savefig(folder_path + 'coefficient_of_lift.png')
83 print(f'Coeficiente linear dos dados experimentais para CL, na parte
      1: {coefficients_exp_cl_1[0]}')
84 print(f'Coeficiente linear dos dados de simula~o para CL, na parte
      1: {coefficients_sim_cl_1[0]}')
85 print(f'Grau de Paralelismo, Parte 1: {1 - np.abs(
      coefficients_sim_cl_1[0]-coefficients_exp_cl_1[0])/np.abs(
      coefficients_exp_cl_1[0])}')
86 print(f'Raz~o da diferen~a entre os coeficientes e o coeficiente
      dos dados experimentais, Parte 1: {np.abs(coefficients_sim_cl_1
      [0]-coefficients_exp_cl_1[0])/np.abs(coefficients_exp_cl_1[0])}')
87 print(f'Coeficiente linear dos dados experimentais para CL, na parte
      2: {coefficients_exp_cl_2[0]}')
88 print(f'Coeficiente linear dos dados de simula~o para CL, na parte
      2: {coefficients_sim_cl_2[0]}')
89 print(f'Grau de Paralelismo, Parte 2: {1 - np.abs(
      coefficients_sim_cl_2[0]-coefficients_exp_cl_2[0])/np.abs(
      coefficients_exp_cl_2[0])}')
90 print(f'Raz~o da diferen~a entre os coeficientes e o coeficiente
      dos dados experimentais, Parte 2: {np.abs(coefficients_sim_cl_2
      [0]-coefficients_exp_cl_2[0])/np.abs(coefficients_exp_cl_2[0])}')
91
92 # print(f'Grau de Paralelismo, Parte 2, {physics_cutoff_point}: {1 -
      np.abs(coefficients_sim_cl_2[0]-coefficients_exp_cl_2[0])/np.abs(
      coefficients_exp_cl_2[0])}')
93
94 # plt.show()
95
96
97 # Filter out zero values of cd_simulation
98 nonzero_cd_indices = np.nonzero(cd_simulation)
99 h_c_nonzero = h_c[nonzero_cd_indices]
100 cd_simulation_nonzero = cd_simulation[nonzero_cd_indices]
101
102 # Plot and compare the coefficient of drag (CD) data
103 print('CD')
104 for physics_cutoff_point in [10]:
105     plt.figure(figsize=(fig_wid, fig_hei))
106     plt.plot(h_c, cd_zerihan, 'bs', label='Experimental', markersize=3)
107     plt.plot(h_c, cd_zerihan, color='blue', linestyle='--', linewidth
      =0.7)
108     plt.plot(h_c_nonzero, cd_simulation_nonzero, 'ro', label='Simula~o
      ', markersize=3)
109     plt.plot(h_c_nonzero, cd_simulation_nonzero, color='red', linestyle='
      --', linewidth=0.7)
110 # Perform linear regression for up to the physics cutoff point of

```

```

    experimental CD
111 coefficients_exp_cd_1 = np.polyfit(h_c[:physics_cutoff_point],
    cd_zerihan[:physics_cutoff_point], 1)
112 trendline_exp_cd_1 = np.polyval(coefficients_exp_cd_1, h_c[:
    physics_cutoff_point])
113 # Perform linear regression for after the pyhsics cutoff point of
    experimental CD
114 coefficients_exp_cd_2 = np.polyfit(h_c[physics_cutoff_point:],
    cd_zerihan[physics_cutoff_point:], 1)
115 trendline_exp_cd_2 = np.polyval(coefficients_exp_cd_2, h_c[
    physics_cutoff_point:])
116 # Plot the two separate trendlines for experimental CD data
117 plt.plot(h_c[:physics_cutoff_point], trendline_exp_cd_1, '-.',
    linewidth=0.8, label='Trend (Experimental - Parte 1)', color='
    lightblue')
118 plt.plot(h_c[physics_cutoff_point:], trendline_exp_cd_2, '-.',
    linewidth=0.8, label='Trend (Experimental - Parte 2)', color='
    lightblue')
119 # Perform linear regression for up to the pyhsics cutoff point of
    simulation CD
120 coefficients_sim_cd_1 = np.polyfit(h_c_nonzero[:physics_cutoff_point
    ], cd_simulation_nonzero[:physics_cutoff_point], 1)
121 trendline_sim_cd_1 = np.polyval(coefficients_sim_cd_1, h_c_nonzero[:
    physics_cutoff_point])
122 # Perform linear regression for after the pyhsics cutoff point of
    simulation CD
123 coefficients_sim_cd_2 = np.polyfit(h_c_nonzero[physics_cutoff_point
    :], cd_simulation_nonzero[physics_cutoff_point:], 1)
124 trendline_sim_cd_2 = np.polyval(coefficients_sim_cd_2, h_c_nonzero[
    physics_cutoff_point:])
125 # Plot the two separate trendlines for simulation CD data
126 plt.plot(h_c_nonzero[:physics_cutoff_point], trendline_sim_cd_1, '-.'
    , linewidth=0.8, label='Trend (SimulaÃ§Ã£o - Parte 1)', color='
    lightcoral')
127 plt.plot(h_c_nonzero[physics_cutoff_point:], trendline_sim_cd_2, '-.'
    , linewidth=0.8, label='Trend (SimulaÃ§Ã£o - Parte 2)', color='
    lightcoral')
128 midpoint_x = (h_c[physics_cutoff_point - 1] + h_c[
    physics_cutoff_point]) / 2.0
129 # Plot the vertical line at the midpoint
130 plt.axvline(x=midpoint_x, color='gray', linestyle='--', linewidth
    =0.9, label='Desaparecimento da Bolha de RecirculaÃ§Ã£o')
131 plt.xlabel('h/c')
132 plt.ylabel('CD')
133 plt.xlim(0, 0.6)
134 plt.xticks(np.arange(0, 0.65, 0.05))
135 plt.ylim(0.17, 0.32)
136 plt.yticks(np.arange(0.17, 0.33, 0.01))
137 plt.title('Comparativo dos Resultados das SimulaÃ§Ãµes com os

```

```

    Experimentos de Zerihan (2001)')
138 plt.suptitle('Coeficiente de Arrasto', y=0.95, fontsize=30,
    fontweight='bold')
139 plt.legend()
140 plt.grid(True, linestyle=':')
141 plt.savefig(folder_path + 'coefficient_of_drag.png')
142 print(f'Coeficiente linear dos dados experimentais para CD, na parte
    1: {coefficients_exp_cd_1[0]}')
143 print(f'Coeficiente linear dos dados de simulaÃ§Ã£o para CD, na parte
    1: {coefficients_sim_cd_1[0]}')
144 print(f'Grau de Paralelismo, Parte 1: {1 - np.abs(
    coefficients_sim_cd_1[0]-coefficients_exp_cd_1[0])/np.abs(
    coefficients_exp_cd_1[0])}')
145 print(f'RazÃ£o da diferenÃ§a entre os coeficientes e o coeficiente
    dos dados experimentais, Parte 1: {np.abs(coefficients_sim_cd_1
    [0]-coefficients_exp_cd_1[0])/np.abs(coefficients_exp_cd_1[0])}')
146 print(f'Coeficiente linear dos dados experimentais para CD, na parte
    2: {coefficients_exp_cd_2[0]}')
147 print(f'Coeficiente linear dos dados de simulaÃ§Ã£o para CD, na parte
    2: {coefficients_sim_cd_2[0]}')
148 print(f'Grau de Paralelismo, Parte 2: {1 - np.abs(
    coefficients_sim_cd_2[0]-coefficients_exp_cd_2[0])/np.abs(
    coefficients_exp_cd_2[0])}')
149 print(f'RazÃ£o da diferenÃ§a entre os coeficientes e o coeficiente
    dos dados experimentais, Parte 2: {np.abs(coefficients_sim_cd_2
    [0]-coefficients_exp_cd_2[0])/np.abs(coefficients_exp_cd_2[0])}')
150
151 # plt.show()
152
153
154 #
    #####

155 ##### Plotting the Force Values
    #####
156 #
    #####

157
158
159 ##### F in X Axis
    #####

160
161 # Create a Boolean mask to filter out rows with delta_fx_percent equal to
    100
162 mask = delta_fx_percent != 100
163
164 # Filter the data based on the mask
165 filtered_h_c = h_c[mask]

```



```

166 filtered_delta_fx_kgf = delta_fx_kgf[mask]
167 filtered_delta_fx_percent = delta_fx_percent[mask]
168 filtered_h = h[mask]
169
170 # Calculate the maximum absolute value of delta_fx_kgf
171 max_abs_fx_kgf = max(abs(filtered_delta_fx_kgf))
172
173 # Create a figure with two y-axes
174 fig, ax1 = plt.subplots(figsize=(fig_wid, fig_hei))
175 ax2 = ax1.twinx()
176
177 # Plot filtered_delta_fx_kgf on the first y-axis
178 line1 = ax1.plot(filtered_h_c, filtered_delta_fx_kgf, 'gv', label='\
    u0394Fx (kgf)', markersize=3)
179 line2 = ax1.plot(filtered_h_c, filtered_delta_fx_kgf, color='green',
    linestyle='--', linewidth=0.7)
180 ax1.set_xlabel('h/c')
181 ax1.set_ylabel('\u0394Fx (kgf)')
182 ax1.set_ylim(-max_abs_fx_kgf*1.1, max_abs_fx_kgf*1.1) # Set the y-axis
    limits centered around 0
183 ax1.set_xlim(0, 0.6)
184 ax1.set_xticks(np.arange(0, 0.65, 0.05))
185 ax1.grid(True, linestyle=':', linewidth=1, axis='x')
186
187 # Plot filtered_delta_fx_percent on the second y-axis
188 line3 = ax2.plot(filtered_h_c, filtered_delta_fx_percent, 'md', label='\
    u0394Fx (%)', markersize=3)
189 line4 = ax2.plot(filtered_h_c, filtered_delta_fx_percent, color='magenta',
    linestyle='--', linewidth=0.7)
190 ax2.set_ylabel('\u0394Fx (%)')
191 ax2.set_ylim(0, 100) # Set the y-axis limits from 0 to 100
192 ax2.set_yticks(np.arange(0, 101, 5))
193 ax2.grid(True, linestyle=':', linewidth=1)
194
195 # Get the minimum and maximum values for the y-axis data
196 min_val = np.min(np.abs(filtered_delta_fx_kgf))
197 max_val = np.max(np.abs(filtered_delta_fx_kgf))
198 min_val_percent = np.min(np.abs(filtered_delta_fx_percent))
199 max_val_percent = np.max(np.abs(filtered_delta_fx_percent))
200
201 min_index_fx = np.argmax(np.abs(filtered_delta_fx_percent))
202 max_index_fx = np.argmin(np.abs(filtered_delta_fx_percent))
203
204 min_h_fx = filtered_h[min_index_fx]
205 max_h_fx = filtered_h[max_index_fx]
206
207 # Create a text box with the minimum and maximum values
208 text_box = f'Min: {min_val:.3f} kgf ou {min_val_percent:.2f}% em h = {
    min_h_fx:.0f} mm\nMax: {max_val:.3f} kgf ou {max_val_percent:.2f}% em

```

```

    h = {max_h_fx:.0f} mm'
209 props = dict(boxstyle='round', facecolor='white', alpha=0.5)
210 ax1.text(0.985, 0.89, text_box, transform=ax1.transAxes, fontsize=10,
    verticalalignment='bottom', horizontalalignment='right', bbox=props)
211
212 # Combine the handles and labels from both axes
213 lines = line1 + line3
214 labels = [l.get_label() for l in lines]
215
216 # Set titles and legend
217 plt.suptitle('Comparativo dos Valores de \u0394Fx', y=0.95, fontsize=30,
    fontweight='bold')
218 plt.title('Diferença em Força Exercida no Eixo X Entre Simulações e
    os Experimentos de Zerihan (2001)')
219 plt.legend(lines, labels, loc='upper right')
220
221 # Save the figure
222 plt.savefig(folder_path + 'delta_fx.png')
223 # plt.show()
224
225
226 ##### F in Y Axis
    #####
227
228 # Create a Boolean mask to filter out rows with delta_fy_percent equal to
    100
229 mask = delta_fy_percent != 100
230
231 # Filter the data based on the mask
232 filtered_h_c = h_c[mask]
233 filtered_delta_fy_kgf = delta_fy_kgf[mask]
234 filtered_delta_fy_percent = delta_fy_percent[mask]
235 filtered_h = h[mask]
236
237 # Calculate the maximum absolute value of delta_fy_kgf
238 max_abs_fy_kgf = max(abs(filtered_delta_fy_kgf))
239
240 # Create a figure with two y-axes
241 fig, ax1 = plt.subplots(figsize=(fig_wid, fig_hei))
242 ax2 = ax1.twinx()
243
244 # Plot filtered_delta_fy_kgf on the first y-axis
245 line1 = ax1.plot(filtered_h_c, filtered_delta_fy_kgf, 'gv', label='\
    \u0394Fy (kgf)', markersize=3)
246 line2 = ax1.plot(filtered_h_c, filtered_delta_fy_kgf, color='green',
    linestyle='--', linewidth=0.7)
247 ax1.set_xlabel('h/c')
248 ax1.set_ylabel('\u0394Fy (kgf)')
249 ax1.set_ylim(-max_abs_fy_kgf*1.1, max_abs_fy_kgf*1.1) # Set the y-axis

```

```

        limits centered around 0
250 ax1.set_xlim(0, 0.6)
251 ax1.set_xticks(np.arange(0, 0.65, 0.05))
252 ax1.grid(True, linestyle=':', linewidth=1, axis='x')
253
254 # Plot filtered_delta_fy_percent on the second y-axis
255 line3 = ax2.plot(filtered_h_c, filtered_delta_fy_percent, 'md', label='\
        u0394Fy (%)', markersize=3)
256 line4 = ax2.plot(filtered_h_c, filtered_delta_fy_percent, color='magenta',
        , linestyle='--', linewidth=0.7)
257 ax2.set_ylabel('\u0394Fy (%)')
258 ax2.set_ylim(0, 100) # Set the y-axis limits from 0 to 100
259 ax2.set_yticks(np.arange(0, 101, 5))
260 ax2.grid(True, linestyle=':', linewidth=1)
261
262
263 # Get the minimum and maximum values for the y-axis data
264 min_val = np.min(np.abs(filtered_delta_fy_kgf))
265 max_val = np.max(np.abs(filtered_delta_fy_kgf))
266 min_val_percent = np.min(np.abs(filtered_delta_fy_percent))
267 max_val_percent = np.max(np.abs(filtered_delta_fy_percent))
268
269 min_index_fy = np.argmin(np.abs(filtered_delta_fy_percent))
270 max_index_fy = np.argmax(np.abs(filtered_delta_fy_percent))
271
272 min_h_fy = filtered_h[min_index_fy]
273 max_h_fy = filtered_h[max_index_fy]
274
275 # Create a text box with the minimum and maximum values
276 text_box = f'Min: {min_val:.3f} kgf ou {min_val_percent:.2f}% em h = {
        min_h_fy:.0f} mm\nMax: {max_val:.3f} kgf ou {max_val_percent:.2f}% em
        h = {max_h_fy:.0f} mm'
277 props = dict(boxstyle='round', facecolor='white', alpha=0.5)
278 ax1.text(0.985, 0.89, text_box, transform=ax1.transAxes, fontsize=10,
        verticalalignment='bottom', horizontalalignment='right', bbox=props)
279
280 # Combine the handles and labels from both axes
281 lines = line1 + line3
282 labels = [l.get_label() for l in lines]
283
284 # Set titles and legend
285 plt.suptitle('Comparativo dos Valores de \u0394Fy', y=0.95, fontsize=30,
        fontweight='bold')
286 plt.title('Diferen\u00e7a em For\u00e7a Exercida no Eixo Y Entre Simula\u00e7\u00f5es e
        os Experimentos de Zerihan (2001)')
287 plt.legend(lines, labels, loc='upper right')
288
289 # Save the figure
290 plt.savefig(folder_path + 'delta_fy.png')

```

```

291 # plt.show()
292
293 #
#####

294 ##### Plotting all Values over h
#####
295 #
#####

296
297
298 # Plot and compare the coefficient of lift (CL) data with 'h' as the x-
axis
299 plt.figure(figsize=(fig_wid, fig_hei))
300 plt.plot(h, cl_zerihan, 'bs', label='Experimental', markersize=3)
301 plt.plot(h, cl_zerihan, color='blue', linestyle='--', linewidth=0.7)
302 plt.plot(h_nonzero, cl_simulation_nonzero, 'ro', label='Simulaãģãčo',
markersize=3)
303 plt.plot(h_nonzero, cl_simulation_nonzero, color='red', linestyle='--',
linewidth=0.7)
304 plt.xlabel('h')
305 plt.ylabel('CL')
306 plt.xlim(0, 230)
307 plt.xticks(np.arange(0,240,10))
308 plt.ylim(1.6, 3.1)
309 plt.yticks(np.arange(1.6,3.15,0.05))
310 plt.title('Comparativo dos Resultados das Simulaãģãtes com os
Experimentos de Zerihan (2001)')
311 plt.suptitle('Coeficiente de Sustentaãģãčo', y=0.95, fontsize=30,
fontweight='bold')
312 plt.legend()
313 plt.grid(True, linestyle=':')
314 plt.savefig(folder_path + 'coefficient_of_lift_h.png')
315
316 # Plot and compare the coefficient of drag (CD) data with 'h' as the x-
axis
317 plt.figure(figsize=(fig_wid, fig_hei))
318 plt.plot(h, cd_zerihan, 'bs', label='Experimental', markersize=3)
319 plt.plot(h, cd_zerihan, color='blue', linestyle='--', linewidth=0.7)
320 plt.plot(h_nonzero, cd_simulation_nonzero, 'ro', label='Simulaãģãčo',
markersize=3)
321 plt.plot(h_nonzero, cd_simulation_nonzero, color='red', linestyle='--',
linewidth=0.7)
322 plt.xlabel('h')
323 plt.ylabel('CD')
324 plt.xlim(0, 230)
325 plt.xticks(np.arange(0,240,10))
326 plt.ylim(0.17, 0.32)

```



```

327 plt.yticks(np.arange(0.17,0.33,0.01))
328 plt.title('Comparativo dos Resultados das Simulações com os
           Experimentos de Zerihan (2001)')
329 plt.suptitle('Coeficiente de Arrasto', y=0.95, fontsize=30, fontweight='
           bold')
330 plt.legend()
331 plt.grid(True, linestyle=':')
332 plt.savefig(folder_path + 'coefficient_of_drag_h.png')
333
334 ##### F in X Axis
           #####
335
336 # Create a Boolean mask to filter out rows with delta_fx_percent equal to
           100
337 mask = delta_fx_percent != 100
338
339 # Filter the data based on the mask
340 filtered_h_c = h_c[mask]
341 filtered_delta_fx_kgf = delta_fx_kgf[mask]
342 filtered_delta_fx_percent = delta_fx_percent[mask]
343 filtered_h = h[mask]
344
345 # Calculate the maximum absolute value of delta_fx_kgf
346 max_abs_fx_kgf = max(abs(filtered_delta_fx_kgf))
347
348 # Create a figure with two y-axes
349 fig, ax1 = plt.subplots(figsize=(fig_wid, fig_hei))
350 ax2 = ax1.twinx()
351
352 # Plot filtered_delta_fx_kgf on the first y-axis
353 line1 = ax1.plot(filtered_h, filtered_delta_fx_kgf, 'gv', label='\u0394Fx
           (kgf)', markersize=3)
354 line2 = ax1.plot(filtered_h, filtered_delta_fx_kgf, color='green',
           linestyle='--', linewidth=0.7)
355 ax1.set_xlabel('h')
356 ax1.set_ylabel('\u0394Fx (kgf)')
357 ax1.set_ylim(-max_abs_fx_kgf*1.1, max_abs_fx_kgf*1.1) # Set the y-axis
           limits centered around 0
358 ax1.set_xlim(0, 230)
359 ax1.set_xticks(np.arange(0, 240, 10))
360 ax1.grid(True, linestyle=':', linewidth=1, axis='x')
361
362 # Plot filtered_delta_fx_percent on the second y-axis
363 line3 = ax2.plot(filtered_h, filtered_delta_fx_percent, 'md', label='\
           \u0394Fx (%)', markersize=3)
364 line4 = ax2.plot(filtered_h, filtered_delta_fx_percent, color='magenta',
           linestyle='--', linewidth=0.7)
365 ax2.set_ylabel('\u0394Fx (%)')
366 ax2.set_ylim(0, 100) # Set the y-axis limits from 0 to 100

```

```

367 ax2.set_yticks(np.arange(0, 101, 5))
368 ax2.grid(True, linestyle=':', linewidth=1)
369
370 # Get the minimum and maximum values for the y-axis data
371 min_val = np.min(np.abs(filtered_delta_fx_kgf))
372 max_val = np.max(np.abs(filtered_delta_fx_kgf))
373 min_val_percent = np.min(np.abs(filtered_delta_fx_percent))
374 max_val_percent = np.max(np.abs(filtered_delta_fx_percent))
375
376 min_index_fx = np.argmin(np.abs(filtered_delta_fx_percent))
377 max_index_fx = np.argmax(np.abs(filtered_delta_fx_percent))
378
379 min_h_fx = filtered_h[min_index_fx]
380 max_h_fx = filtered_h[max_index_fx]
381
382 # Create a text box with the minimum and maximum values
383 text_box = f'Min: {min_val:.3f} kgf ou {min_val_percent:.2f}% em h = {
    min_h_fx:.0f} mm\nMax: {max_val:.3f} kgf ou {max_val_percent:.2f}% em
    h = {max_h_fx:.0f} mm'
384 props = dict(boxstyle='round', facecolor='white', alpha=0.5)
385 ax1.text(0.985, 0.89, text_box, transform=ax1.transAxes, fontsize=10,
    verticalalignment='bottom', horizontalalignment='right', bbox=props)
386
387 # Combine the handles and labels from both axes
388 lines = line1 + line3
389 labels = [l.get_label() for l in lines]
390
391 # Set titles and legend
392 plt.suptitle('Comparativo dos Valores de \u0394Fx', y=0.95, fontsize=30,
    fontweight='bold')
393 plt.title('Diferen\u00e7a em For\u00e7a Exercida no Eixo X Entre Simula\u00e7\u00f5es e
    os Experimentos de Zerihan (2001)')
394 plt.legend(lines, labels, loc='upper right')
395
396 # Save the figure
397 plt.savefig(folder_path + 'delta_fx_h.png')
398 # plt.show()
399
400
401 ##### F in Y Axis
    #####
402
403 # Create a Boolean mask to filter out rows with delta_fy_percent equal to
    100
404 mask = delta_fy_percent != 100
405
406 # Filter the data based on the mask
407 filtered_h_c = h_c[mask]
408 filtered_delta_fy_kgf = delta_fy_kgf[mask]

```

```

409 filtered_delta_fy_percent = delta_fy_percent[mask]
410 filtered_h = h[mask]
411
412 # Calculate the maximum absolute value of delta_fy_kgf
413 max_abs_fy_kgf = max(abs(filtered_delta_fy_kgf))
414
415 # Create a figure with two y-axes
416 fig, ax1 = plt.subplots(figsize=(fig_wid, fig_hei))
417 ax2 = ax1.twinx()
418
419 # Plot filtered_delta_fy_kgf on the first y-axis
420 line1 = ax1.plot(filtered_h, filtered_delta_fy_kgf, 'gv', label='\u0394Fy
      (kgf)', markersize=3)
421 line2 = ax1.plot(filtered_h, filtered_delta_fy_kgf, color='green',
      linestyle='--', linewidth=0.7)
422 ax1.set_xlabel('h')
423 ax1.set_ylabel('\u0394Fy (kgf)')
424 ax1.set_ylim(-max_abs_fy_kgf*1.1, max_abs_fy_kgf*1.1) # Set the y-axis
      limits centered around 0
425 ax1.set_xlim(0, 230)
426 ax1.set_xticks(np.arange(0, 240, 10))
427 ax1.grid(True, linestyle=':', linewidth=1, axis='x')
428
429 # Plot filtered_delta_fy_percent on the second y-axis
430 line3 = ax2.plot(filtered_h, filtered_delta_fy_percent, 'md', label='\
      u0394Fy (%)', markersize=3)
431 line4 = ax2.plot(filtered_h, filtered_delta_fy_percent, color='magenta',
      linestyle='--', linewidth=0.7)
432 ax2.set_ylabel('\u0394Fy (%)')
433 ax2.set_ylim(0, 100) # Set the y-axis limits from 0 to 100
434 ax2.set_yticks(np.arange(0, 101, 5))
435 ax2.grid(True, linestyle=':', linewidth=1)
436
437
438 # Get the minimum and maximum values for the y-axis data
439 min_val = np.min(np.abs(filtered_delta_fy_kgf))
440 max_val = np.max(np.abs(filtered_delta_fy_kgf))
441 min_val_percent = np.min(np.abs(filtered_delta_fy_percent))
442 max_val_percent = np.max(np.abs(filtered_delta_fy_percent))
443
444 min_index_fy = np.argmin(np.abs(filtered_delta_fy_percent))
445 max_index_fy = np.argmax(np.abs(filtered_delta_fy_percent))
446
447 min_h_fy = filtered_h[min_index_fy]
448 max_h_fy = filtered_h[max_index_fy]
449
450 # Create a text box with the minimum and maximum values
451 text_box = f'Min: {min_val:.3f} kgf ou {min_val_percent:.2f}% em h = {
      min_h_fy:.0f} mm\nMax: {max_val:.3f} kgf ou {max_val_percent:.2f}% em

```

```

    h = {max_h_fy:.0f} mm'
452 props = dict(boxstyle='round', facecolor='white', alpha=0.5)
453 ax1.text(0.985, 0.89, text_box, transform=ax1.transAxes, fontsize=10,
    verticalalignment='bottom', horizontalalignment='right', bbox=props)
454
455 # Combine the handles and labels from both axes
456 lines = line1 + line3
457 labels = [l.get_label() for l in lines]
458
459 # Set titles and legend
460 plt.suptitle('Comparativo dos Valores de \u0394Fy', y=0.95, fontsize=30,
    fontweight='bold')
461 plt.title('Diferen\u00e7a em For\u00e7a Exercida no Eixo Y Entre Simula\u00e7\u00f5es e
    os Experimentos de Zerihan (2001)')
462 plt.legend(lines, labels, loc='upper right')
463
464 # Save the figure
465 plt.savefig(folder_path + 'delta_fy_h.png')
466 # plt.show()
467
468 #
    #####
469 ##### Plotting CL/CD over h/c
    #####
470 #
    #####

471 print('CL/CD')
472
473 # Calculate the CL/CD ratio for both simulation and experimental data
474 cl_cd_ratio_exp = cl_zerihan / cd_zerihan
475 cl_cd_ratio_sim = cl_simulation_nonzero / cd_simulation_nonzero
476
477 # Plot and compare the CL/CD ratio data
478 plt.figure(figsize=(fig_wid, fig_hei))
479 plt.plot(h_c, cl_cd_ratio_exp, 'bs', label='Experimental', markersize=3)
480 plt.plot(h_c, cl_cd_ratio_exp, color='blue', linestyle='--', linewidth
    =0.7)
481 plt.plot(h_c_nonzero, cl_cd_ratio_sim, 'ro', label='Simula\u00e7\u00e3o',
    markersize=3)
482 plt.plot(h_c_nonzero, cl_cd_ratio_sim, color='red', linestyle='--',
    linewidth=0.7)
483 # Calculate the midpoint x-coordinate
484 midpoint_x = (h_c[physics_cutoff_point - 1] + h_c[physics_cutoff_point])
    / 2.0
485 # Plot the vertical line at the midpoint
486 plt.axvline(x=midpoint_x, color='gray', linestyle='--', linewidth=0.9,
    label='Desaparecimento da Bolha de Recircula\u00e7\u00e3o')
```



```

487 plt.xlabel('h/c')
488 plt.ylabel('CL/CD')
489 plt.xlim(0, 0.6)
490 plt.xticks(np.arange(0, 0.65, 0.05))
491 plt.ylim(5, 12)
492 plt.yticks(np.arange(5, 13, 1))
493 plt.title('Comparativo da Razão CL/CD entre Simulações e os
      Experimentos de Zerihan (2001)')
494 plt.suptitle('Razão CL/CD', y=0.95, fontsize=30, fontweight='bold')
495 plt.legend()
496 plt.grid(True, linestyle=':')
497 plt.savefig(folder_path + 'cl_cd_ratio.png')
498
499 # Print CL/CD datapoints for experimental data
500 print('Experimental CL/CD datapoints:')
501 for i in range(len(h)):
502     print(f'{h[i]}, {cl_cd_ratio_exp[i]}')
503
504 # Print CL/CD datapoints for simulation data
505 print('\nSimulation CL/CD datapoints:')
506 for i in range(len(h_nonzero)):
507     print(f'{h_nonzero[i]}, {cl_cd_ratio_sim[i]}')
508
509
510 print('')
511 print(f'Done plotting the results, check {folder_path}')

```

# APÊNDICE L

## Código dos Arquivos de Configuração do SU2

### L.1 Caso modelo h $\alpha$

Listagem L.1: hx.cfg

```
1 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
2 %
3 % SU2 configuration file
4 % Case description: Tyrrell 026 Front Wing in Wind Tunnel -
5 % Author: Matheus Ribeiro
6 % Institution: Universidade de
7 % Date: 13/06/2023
8 % File Version 7.3.1 "Blackbird"
9 %
10 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
11
12 % ----- DIRECT, ADJOINT, AND LINEARIZED PROBLEM DEFINITION
13 % -----%
14 % Solver type (EULER, NAVIER_STOKES, RANS,
```

```

15 %          INC_EULER, INC_NAVIER_STOKES, INC_RANS,
16 %          NEMO_EULER, NEMO_NAVIER_STOKES,
17 %          FEM_EULER, FEM_NAVIER_STOKES, FEM_RANS, FEM_LES,
18 %          HEAT_EQUATION_FVM, ELASTICITY)
19 SOLVER= INC_RANS
20 %
21 % Specify turbulence model (NONE, SA, SA_NEG, SST, SA_E, SA_COMP,
    SA_E_COMP, SST_SUST)
22 KIND_TURB_MODEL= SST
23 %
24 % Mathematical problem (DIRECT, CONTINUOUS_ADJOINT, DISCRETE_ADJOINT)
25 % Defaults to DISCRETE_ADJOINT for the SU2*_AD codes , and to DIRECT
    otherwise.
26 MATH_PROBLEM= DIRECT
27 %
28 % System of measurements (SI, US)
29 % International system of units (SI): ( meters , kilograms , Kelvins ,
30 %          Newtons = kg m/s^2, Pascals = N/m
    ^2,
31 %          Density = kg/m^3, Speed = m/s ,
32 %          Equiv. Area = m^2 )
33 SYSTEM_MEASUREMENTS= SI
34 %
35 %
36 % _____ SOLVER CONTROL
    _____%
37 %
38 % Number of iterations for single-zone problems
39 ITER= 1
40 %
41 % Maximum number of inner iterations
42 INNER_ITER= 15000
43 %
44 % Maximum number of outer iterations (only for multizone problems)
45 OUTER_ITER= 1
46 %
47 % Maximum number of time iterations
48 TIME_ITER= 1
49 %
50 % Convergence field
51 CONV_FIELD= PRESSURE
52 %
53 % Min value of the residual (log10 of the residual)
54 CONV_RESIDUAL_MINVAL= -8
55 %
56 % Start convergence criteria at iteration number
57 CONV_STARTITER= 10
58 %
59 % Number of elements to apply the criteria

```

```

60 CONV_CAUCHY_ELEMS= 100
61 %
62 % Epsilon to control the series convergence
63 CONV_CAUCHY_EPS= 1E-10
64 %
65 % Iteration number to begin unsteady restarts
66 RESTART_ITER= 0
67 %
68 %% Time convergence monitoring
69 WINDOW_CAUCHY_CRIT = YES
70 %
71 % List of time convergence fields
72 CONV_WINDOW_FIELD = (TAVG_DRAG, TAVG_LIFT)
73 %
74 % Time Convergence Monitoring starts at Iteration WINDOW_START_ITER +
      CONV_WINDOW_STARTITER
75 CONV_WINDOW_STARTITER = 0
76 %
77 % Epsilon to control the series convergence
78 CONV_WINDOW_CAUCHY_EPS = 1E-3
79 %
80 % Number of elements to apply the criteria
81 CONV_WINDOW_CAUCHY_ELEMS = 10
82 %
83 % ----- TIME-DEPENDENT SIMULATION
      -----%
84 %
85 % Time domain simulation
86 TIME_DOMAIN= NO
87 %
88 % Unsteady simulation (NO, TIME_STEPPING, DUAL_TIME_STEPPING-1ST_ORDER,
89 %                      DUAL_TIME_STEPPING-2ND_ORDER, HARMONIC_BALANCE)
90 TIME_MARCHING= NO
91 %
92 % Time Step for dual time stepping simulations (s) — Only used when
      UNST_CFL_NUMBER = 0.0
93 % For the DG-FEM solver it is used as a synchronization time when
      UNST_CFL_NUMBER != 0.0
94 TIME_STEP= 0.00001
95 %
96 % Total Physical Time for dual time stepping simulations (s)
97 MAX_TIME= 2.0
98 %
99 % Unsteady Courant-Friedrichs-Lewy number of the finest grid
100 UNST_CFL_NUMBER= 1.0
101 %
102 %% Windowed output time averaging
103 % Time iteration to start the windowed time average in a direct run
104 WINDOW_START_ITER = 500

```



```

105 %
106 % Window used for reverse sweep and direct run. Options (SQUARE, HANN,
      HANN_SQUARE, BUMP) Square is default.
107 WINDOW_FUNCTION = SQUARE
108 %
109 % ----- DES Parameters
      -----%
110 %
111 % Specify Hybrid RANS/LES model (SA_DES, SA_DDES, SA_ZDES, SA_EDDES)
112 HYBRID_RANSLES= SA_DDES
113 %
114 % DES Constant (0.65)
115 DES_CONST= 0.65
116 %
117 % ----- INCOMPRESSIBLE FLOW CONDITION DEFINITION
      -----%
118 %
119 % Density model within the incompressible flow solver.
120 % Options are CONSTANT (default), BOUSSINESQ, or VARIABLE. If VARIABLE,
121 % an appropriate fluid model must be selected.
122 INC_DENSITY_MODEL= CONSTANT
123 %
124 % Solve the energy equation in the incompressible flow solver
125 INC_ENERGY_EQUATION = NO
126 %
127 % Initial density for incompressible flows
128 % (1.2886 kg/m3 by default (air), 998.2 Kg/m3 (water))
129 INC_DENSITY_INIT= 1.2886
130 %
131 % Initial velocity for incompressible flows (1.0,0,0 m/s by default)
132 INC_VELOCITY_INIT= ( 30.0, 0.0, 0.0 )
133 %
134 % Initial temperature for incompressible flows that include the
135 % energy equation (288.15 K by default). Value is ignored if
136 % INC_ENERGY_EQUATION is false.
137 INC_TEMPERATURE_INIT= 288.15
138 %
139 % Non-dimensionalization scheme for incompressible flows. Options are
140 % INITIAL_VALUES (default), REFERENCE_VALUES, or DIMENSIONAL.
141 % INC*_REF values are ignored unless REFERENCE_VALUES is chosen.
142 INC_NONDIM= INITIAL_VALUES
143 %
144 % Reference density for incompressible flows (1.0 kg/m3 by default)
145 INC_DENSITY_REF= 1.0
146 %
147 % Reference velocity for incompressible flows (1.0 m/s by default)
148 INC_VELOCITY_REF= 1.0
149 %
150 % Reference temperature for incompressible flows that include the

```

```

151 % energy equation (1.0 K by default)
152 INC_TEMPERATURE_REF = 1.0
153 %
154 % List of inlet types for incompressible flows. List length must
155 % match number of inlet markers. Options: VELOCITY_INLET, PRESSURE_INLET.
156 INC_INLET_TYPE= VELOCITY_INLET
157 %
158 % Damping coefficient for iterative updates at pressure inlets. (0.1 by
    default)
159 INC_INLET_DAMPING= 0.1
160 %
161 % List of outlet types for incompressible flows. List length must
162 % match number of outlet markers. Options: PRESSURE_OUTLET,
    MASS_FLOW_OUTLET
163 INC_OUTLET_TYPE= PRESSURE_OUTLET
164 %
165 % Damping coefficient for iterative updates at mass flow outlets. (0.1 by
    default)
166 INC_OUTLET_DAMPING= 0.1
167 %
168 % Epsilon^2 multiplier in Beta calculation for incompressible
    preconditioner.
169 BETA_FACTOR= 4.1
170 %
171 % ----- REFERENCE VALUE DEFINITION
    -----%
172 %
173 % Reference origin for moment computation (m or in) % Atualmente em um
    quarto de Asa
174 REF_ORIGIN_MOMENT_X = 0.00
175 REF_ORIGIN_MOMENT_Y = 0.00
176 REF_ORIGIN_MOMENT_Z = 0.00
177 %
178 % Reference length for moment non-dimensional coefficients (m or in)
179 REF_LENGTH= 0.380
180 %
181 % Reference area for non-dimensional force coefficients (0 implies
    automatic
182 % calculation) (m^2 or in^2)
183 REF_AREA= 0.405
184 %
185 % Aircraft semi-span (0 implies automatic calculation) (m or in)
186 SEMI_SPAN= 0.0
187 %
188 % ----- VISCOSITY MODEL
    -----%
189 %
190 % Viscosity model (SUTHERLAND, CONSTANT_VISCOSITY, POLYNOMIAL_VISCOSITY).
191 VISCOSITY_MODEL= CONSTANT_VISCOSITY

```

```

192
193 % Molecular Viscosity that would be constant (1.716E-5 by default)
194 MU_CONSTANT= 1.716E-5
195 %
196 % Sutherland Viscosity Ref (1.716E-5 default value for AIR SI)
197 MU_REF= 1.716E-5
198 %
199 % Sutherland Temperature Ref (273.15 K default value for AIR SI)
200 MU_T_REF= 273.15
201 %
202 % Sutherland constant (110.4 default value for AIR SI)
203 SUTHERLAND_CONSTANT= 110.4
204 %
205 % Temperature polynomial coefficients (up to quartic) for viscosity.
206 % Format -> Mu(T) : b0 + b1*T + b2*T^2 + b3*T^3 + b4*T^4
207 MU_POLYCOEFFS= (0.0, 0.0, 0.0, 0.0, 0.0)
208 %
209 % ----- DYNAMIC MESH DEFINITION
      -----%
210 %
211 % Type of dynamic surface movement (NONE, DEFORMING, MOVING_WALL,
212 % AEROELASTIC, AEROELASTIC_RIGID_MOTION EXTERNAL, EXTERNAL_ROTATION)
213 SURFACE_MOVEMENT= MOVING_WALL
214 %
215 % Moving wall boundary marker(s) (NONE = no marker, ignored for
      RIGID_MOTION)
216 MARKER_MOVING= ( ground )
217 %
218 % Coordinates of the motion origin
219 SURFACE_MOTION_ORIGIN= -4 0.0 0.0
220 %
221 % Translational velocity (m/s or ft/s) in the x, y, & z directions
222 SURFACE_TRANSLATION_RATE = 30.0 0.0 0.0
223 %
224 %
225 % Move Motion Origin for marker moving (1 or 0)
226 MOVE_MOTION_ORIGIN = 0
227 %
228 % ----- BOUNDARY CONDITION DEFINITION
      -----%
229 %
230 % Euler wall boundary marker(s) (NONE = no marker)
231 % Implementation identical to MARKER_SYM.
232 MARKER_EULER= ( wall )
233 %
234 % Navier-Stokes (no-slip), constant heat flux wall marker(s) (NONE = no
      marker)
235 % Format: ( marker name, constant heat flux (J/m^2), ... )
236 MARKER_HEATFLUX= ( wing, 0, ground, 0)

```

```

237 %
238 % Far-field boundary marker(s) (NONE = no marker)
239 MARKER_FAR= ( inlet , outlet )
240 %
241 % Inlet boundary type (TOTAL_CONDITIONS, MASS_FLOW)
242 %
243 %INLET_TYPE= TOTAL_CONDITIONS
244 %
245 % Inlet boundary marker(s) with the following formats (NONE = no marker)
246 % Total Conditions: ( inlet marker, total temp, total pressure ,
      flow_direction_x ,
247 %      flow_direction_y, flow_direction_z, ... ) where
      flow_direction is
248 %      a unit vector.
249 % Mass Flow: ( inlet marker, density , velocity magnitude, flow_direction_x
      ,
250 %      flow_direction_y, flow_direction_z, ... ) where
      flow_direction is
251 %      a unit vector.
252 % Inc. Velocity: ( inlet marker, temperature, velocity magnitude,
      flow_direction_x ,
253 %      flow_direction_y, flow_direction_z, ... ) where
      flow_direction is
254 %      a unit vector.
255 % Inc. Pressure: ( inlet marker, temperature, total pressure ,
      flow_direction_x ,
256 %      flow_direction_y, flow_direction_z, ... ) where
      flow_direction is
257 %      a unit vector.
258 %MARKER_INLET= ( INLET, 288.15, 101325.0, 1.0, 0.0, 0.0)
259 %
260 % Outlet boundary marker(s) (NONE = no marker)
261 % Compressible: ( outlet marker, back pressure (static thermodynamic),
      ... )
262 % Inc. Pressure: ( outlet marker, back pressure (static gauge in Pa), ...
      )
263 % Inc. Mass Flow: ( outlet marker, mass flow target (kg/s), ... )
264 %MARKER_OUTLET= ( OUTLET )
265 %
266 % _____ WALL FUNCTION DEFINITION
      _____%
267 %
268 % The von Karman constant, the constant below only affects the standard
      wall function model
269 WALLMODEL_KAPPA= 0.41
270 %
271 % The wall function model constant B
272 WALLMODEL_B= 5.5
273 %

```

```

274 % The y+ value below which the wall function is switched off and we
      resolve the wall
275 WALLMODEL_MINYPPLUS= 5.0
276 %
277 % [Expert] Max Newton iterations used for the standard wall function
278 WALLMODEL_MAXITER= 200
279 %
280 % [Expert] relaxation factor for the Newton iterations of the standard
      wall function
281 WALLMODEL_RELFAC= 0.5
282
283 % ----- SURFACES IDENTIFICATION
      -----%
284 %
285 % Marker(s) of the surface in the surface flow solution file
286 MARKER_PLOTTING = ( wing, wall, ground )
287 %
288 % Marker(s) of the surface where the non-dimensional coefficients are
      evaluated.
289 MARKER_MONITORING = ( wing )
290 %
291 % Viscous wall markers for which wall functions must be applied. (NONE =
      no marker)
292 % Format: ( marker name, wall function type -NO_WALL_FUNCTION,
      STANDARD_WALL_FUNCTION,
293 %          ADAPTIVE_WALL_FUNCTION, SCALABLE_WALL_FUNCTION,
      EQUILIBRIUM_WALL_MODEL,
294 %          NONEQUILIBRIUM_WALL_MODEL-, ... )
295 MARKER_WALL_FUNCTIONS= ( wing, NO_WALL_FUNCTION )
296 %
297 % Marker(s) of the surface where custom thermal BCs are defined.
298 MARKER_PYTHON_CUSTOM = ( NONE )
299 %
300 % Marker(s) of the surface that is going to be analyzed in detail (
      massflow, average pressure, distortion, etc)
301 MARKER_ANALYZE = ( wing )
302 %
303 % Method to compute the average value in MARKER_ANALYZE (AREA, MASSFLUX).
304 MARKER_ANALYZE_AVERAGE = AREA
305
306 % ----- COMMON PARAMETERS DEFINING THE NUMERICAL METHOD
      -----%
307 %
308 % Numerical method for spatial gradients (GREEN_GAUSS,
      WEIGHTED_LEAST_SQUARES)
309 NUM_METHOD_GRAD= GREEN_GAUSS
310
311 % Numerical method for spatial gradients to be used for MUSCL
      reconstruction

```



```

312 % Options are (GREEN_GAUSS, WEIGHTED_LEAST_SQUARES, LEAST_SQUARES).
      Default value is
313 % NONE and the method specified in NUM_METHOD_GRAD is used.
314 NUM_METHOD_GRAD_RECON = NONE
315 %
316 % CFL number (initial value for the adaptive CFL number)
317 CFL_NUMBER= 1.0
318 %
319 % Adaptive CFL number (NO, YES)
320 CFL_ADAPT= NO
321 %
322 % Parameters of the adaptive CFL number (factor-down, factor-up, CFL min
      value,
323 %                                CFL max value, acceptable linear
      solver convergence)
324 % Local CFL increases by factor-up until max if the solution rate of
      change is not limited,
325 % and acceptable linear convergence is achieved. It is reduced if rate is
      limited, or if there
326 % is not enough linear convergence, or if the nonlinear residuals are
      stagnant and oscillatory.
327 % It is reset back to min when linear solvers diverge, or if nonlinear
      residuals increase too much.
328 CFL_ADAPT_PARAM= ( 0.1, 2.0, 10.0, 1e10, 0.001 )
329 %
330 % Maximum Delta Time in local time stepping simulations
331 MAX_DELTA_TIME= 1E6
332 %
333 % Runge-Kutta alpha coefficients
334 RK_ALPHA_COEFF= ( 0.666667, 0.666667, 1.000000 )
335 %
336 % Objective function in gradient evaluation (DRAG, LIFT, SIDEFORCE,
      MOMENT_X,
337 %                                MOMENT_Y, MOMENT_Z,
      EFFICIENCY, BUFFET,
338 %                                EQUIVALENT_AREA,
      NEARFIELD_PRESSURE,
339 %                                FORCE_X, FORCE_Y, FORCE_Z,
      THRUST,
340 %                                TORQUE, TOTAL_HEATFLUX,
      CUSTOM_OBJFUNC
341 %                                MAXIMUM_HEATFLUX,
      INVERSE_DESIGN_PRESSURE,
342 %                                INVERSE_DESIGN_HEATFLUX,
      SURFACE_TOTAL_PRESSURE,
343 %                                SURFACE_MASSFLOW,
      SURFACE_STATIC_PRESSURE, SURFACE_MACH)
344 % For a weighted sum of objectives: separate by commas, add
      OBJECTIVE_WEIGHT and MARKER_MONITORING in matching order.

```

```

345 OBJECTIVE_FUNCTION= DRAG
346 %
347 % List of weighting values when using more than one OBJECTIVE_FUNCTION.
    Separate by commas and match with MARKER_MONITORING.
348 OBJECTIVE_WEIGHT = 1.0
349 %
350 % Expression used when "OBJECTIVE_FUNCTION= CUSTOM_OBIFUNC", any history/
    screen output can be used together with common
351 % math functions (sqrt, cos, exp, etc.). This can be used for constraint
    aggregation (as below) or to compute something
352 % SU2 does not, see TestCases/user_defined_functions/.
353 CUSTOM_OBIFUNC= 'DRAG + 10 * pow(fmax(0.4-LIFT, 0), 2) '
354
355 % ----- SLOPE LIMITER AND DISSIPATION SENSOR DEFINITION
    -----%
356 %
357 % Monotonic Upwind Scheme for Conservation Laws (TVD) in the flow
    equations.
358 %           Required for 2nd order upwind schemes (NO, YES)
359 MUSCL_FLOW= YES
360 %
361 % Slope limiter (NONE, VENKATAKRISHNAN, VENKATAKRISHNAN_WANG,
362 %           BARTH_JESPERSEN, VAN_ALBADA_EDGE)
363 SLOPE_LIMITER_FLOW= VENKATAKRISHNAN
364 %
365 % Monotonic Upwind Scheme for Conservation Laws (TVD) in the turbulence
    equations.
366 %           Required for 2nd order upwind schemes (NO, YES)
367 MUSCL_TURB= NO
368 %
369 % Slope limiter (NONE, VENKATAKRISHNAN, VENKATAKRISHNAN_WANG,
370 %           BARTH_JESPERSEN, VAN_ALBADA_EDGE)
371 SLOPE_LIMITER_TURB= VENKATAKRISHNAN
372 %
373 % Monotonic Upwind Scheme for Conservation Laws (TVD) in the adjoint flow
    equations.
374 %           Required for 2nd order upwind schemes (NO, YES)
375 MUSCL_ADJFLOW= YES
376 %
377 % Slope limiter (NONE, VENKATAKRISHNAN, BARTH_JESPERSEN, VAN_ALBADA_EDGE,
378 %           SHARP_EDGES, WALL_DISTANCE)
379 SLOPE_LIMITER_ADJFLOW= VENKATAKRISHNAN
380 %
381 % Monotonic Upwind Scheme for Conservation Laws (TVD) in the turbulence
    adjoint equations.
382 %           Required for 2nd order upwind schemes (NO, YES)
383 MUSCL_ADJTURB= NO
384 %
385 % Slope limiter (NONE, VENKATAKRISHNAN, BARTH_JESPERSEN, VAN_ALBADA_EDGE)

```

```

386 SLOPE_LIMITER_ADJTURB= VENKATAKRISHNAN
387 %
388 % Coefficient for the Venkats limiter (upwind scheme). A larger values
      decrease
389 %           the extent of limiting , values approaching zero cause
390 %           lower-order approximation to the solution (0.05 by default)
391 VENKAT_LIMITER_COEFF= 0.05
392 %
393 % Reference coefficient for detecting sharp edges (3.0 by default).
394 REF_SHARP_EDGES = 3.0
395 %
396 % Coefficient for the adjoint sharp edges limiter (3.0 by default).
397 ADJ_SHARP_LIMITER_COEFF= 3.0
398 %
399 % Remove sharp edges from the sensitivity evaluation (NO, YES)
400 SENS_REMOVE_SHARP = NO
401 %
402 % Freeze the value of the limiter after a number of iterations
403 LIMITER_ITER= 999999
404 %
405 % 1st order artificial dissipation coefficients for
406 %   the Lax&Auml;Friedrichs method ( 0.15 by default )
407 LAX_SENSOR_COEFF= 0.15
408 %
409 % 2nd and 4th order artificial dissipation coefficients for
410 %   the JST method ( 0.5, 0.02 by default )
411 JST_SENSOR_COEFF= ( 0.5, 0.02 )
412 %
413 % 1st order artificial dissipation coefficients for
414 %   the adjoint Lax&Auml;Friedrichs method ( 0.15 by default )
415 ADJ_LAX_SENSOR_COEFF= 0.15
416 %
417 % 2nd, and 4th order artificial dissipation coefficients for
418 %   the adjoint JST method ( 0.5, 0.02 by default )
419 ADJ_JST_SENSOR_COEFF= ( 0.5, 0.02 )
420
421 % ----- LINEAR SOLVER DEFINITION
      -----%
422 %
423 % Linear solver or smoother for implicit formulations:
424 % BCGSTAB, FGMRES, RESTARTED_FGMRES, CONJUGATE_GRADIENT (self-adjoint
      problems only), SMOOTHER.
425 LINEAR_SOLVER= FGMRES
426 %
427 % Same for discrete adjoint (smoothers not supported), replaces
      LINEAR_SOLVER in SU2 * AD codes .
428 DISCADJ_LIN_SOLVER= FGMRES
429 %
430 % Preconditioner of the Krylov linear solver or type of smoother (ILU,

```

```

        LU_SGS, LINELET, JACOBI)
431  LINEAR_SOLVER_PREC= ILU
432  %
433  % Same for discrete adjoint (JACOBI or ILU), replaces LINEAR_SOLVER_PREC
        in SU2*_AD codes.
434  DISCADJ_LIN_PREC= ILU
435  %
436  % Linear solver ILU preconditioner fill-in level (0 by default)
437  LINEAR_SOLVER_ILU_FILL_IN= 0
438  %
439  % Minimum error of the linear solver for implicit formulations
440  LINEAR_SOLVER_ERROR= 1E-6
441  %
442  % Max number of iterations of the linear solver for the implicit
        formulation
443  LINEAR_SOLVER_ITER= 5
444  %
445  % Restart frequency for RESTARTED_FGMRES
446  LINEAR_SOLVER_RESTART_FREQUENCY= 10
447  %
448  % Relaxation factor for smoother-type solvers (LINEAR_SOLVER= SMOOTHER)
449  LINEAR_SOLVER_SMOOTHER_RELAXATION= 1.0
450  %
451  % ----- FLOW NUMERICAL METHOD DEFINITION
        -----%
452  %
453  % Convective numerical method (JST, JST_KE, JST_MAT, LAX-FRIEDRICH, CUSP,
        ROE, AUSM,
454  %                               AUSMPLUSUP, AUSMPLUSUP2, AUSMPWPLUS, HLLC,
        TURKEL_PREC,
455  %                               SW, MSW, FDS, SLAU, SLAU2, L2ROE, LMROE)
456  CONV_NUM_METHOD_FLOW= JST
457  %
458  % Roe Low Dissipation function for Hybrid RANS/LES simulations (FD, NTS,
        NTS_DUCROS)
459  ROE_LOW DISSIPATION= FD
460  %
461  % Post-reconstruction correction for low Mach number flows (NO, YES)
462  LOW_MACH_CORR= NO
463  %
464  % Roe-Turkel preconditioning for low Mach number flows (NO, YES)
465  LOW_MACH_PREC= NO
466  %
467  % Use numerically computed Jacobians for AUSM+up(2) and SLAU(2)
468  % Slower per iteration but potentially more stable and capable of higher
        CFL
469  USE_ACCURATE_FLUX_JACOBIANS= NO
470  %
471  % Use the vectorized version of the selected numerical method (available

```

```

    for JST family and Roe).
472 % SU2 should be compiled for an AVX or AVX512 architecture for best
    performance.
473 USE_VECTORIZATION= NO
474 %
475 % Entropy fix coefficient (0.0 implies no entropy fixing , 1.0 implies
    scalar
476 %
    artificial dissipation)
477 ENTROPY_FIX_COEFF= 0.0
478 %
479 % Higher values than 1 (3 to 4) make the global Jacobian of central
    schemes (compressible flow
480 % only) more diagonal dominant (but mathematically incorrect) so that
    higher CFL can be used.
481 CENTRAL_JACOBIAN_FIX_FACTOR= 4.0
482 %
483 % Time discretization (RUNGE-KUTTA_EXPLICIT, EULER_IMPLICIT,
    EULER_EXPLICIT)
484 TIME_DISCRE_FLOW= EULER_IMPLICIT
485 %
486 % Use a Newton-Krylov method on the flow equations , see TestCases/rans/
    oneram6/turb_ONERAM6_nk.cfg
487 % For multizone discrete adjoint it will use FGMRES on inner iterations
    with restart frequency
488 % equal to "QUASI_NEWTON_NUM_SAMPLES".
489 NEWTON_KRYLOV= NO
490 %
491 %
492 % ----- TURBULENT NUMERICAL METHOD DEFINITION
    -----%
493 %
494 % Convective numerical method (SCALAR_UPWIND)
495 CONV_NUM_METHOD_TURB= SCALAR_UPWIND
496 %
497 % Time discretization (EULER_IMPLICIT, EULER_EXPLICIT)
498 TIME_DISCRE_TURB= EULER_IMPLICIT
499 %
500 % Reduction factor of the CFL coefficient in the turbulence problem
501 CFL_REDUCTION_TURB= 1.0
502 %
503 % ----- HYBRID PARALLEL (MPI+OpenMP) OPTIONS
    -----%
504 %
505 % An advanced performance parameter for FVM solvers , a large-ish value
    should be best
506 % when relatively few threads per MPI rank are in use (~4). However,
    maximum parallelism
507 % is obtained with EDGE_COLORING_GROUP_SIZE=1, consider using this value
    only if SU2

```



```

508 % warns about low coloring efficiency during preprocessing (performance
      is usually worse).
509 % Setting the option to 0 disables coloring and a different strategy is
      used instead,
510 % that strategy is automatically used when the coloring efficiency is
      less than 0.875.
511 % The optimum value/strategy is case-dependent.
512 EDGE_COLORING_GROUP_SIZE= 512
513 %
514 % Independent "threads per MPI rank" setting for LU-SGS and ILU
      preconditioners.
515 % For problems where time is spend mostly in the solution of linear
      systems (e.g. elasticity,
516 % very high CFL central schemes), AND, if the memory bandwidth of the
      machine is saturated
517 % (4 or more cores per memory channel) better performance (via a
      reduction in linear iterations)
518 % may be possible by using a smaller value than that defined by the
      system or in the call to
519 % SU2_CFD (via the -t/--threads option).
520 % The default (0) means "same number of threads as for all else".
521 LINEAR_SOLVER_PREC_THREADS= 0
522 %
523 % ----- PARTITIONING OPTIONS (ParMETIS)
      ----- %
524 %
525 % Load balancing tolerance, lower values will make ParMETIS work harder
      to evenly
526 % distribute the work-estimate metric across all MPI ranks, at the
      expense of more
527 % edge cuts (i.e. increased communication cost).
528 PARMETIS_TOLERANCE= 0.02
529 %
530 % The work-estimate metric is a weighted function of the work-per-edge (e
      .g. spatial
531 % discretization, linear system solution) and of the work-per-point (e.g.
      source terms,
532 % temporal discretization) the former usually accounts for >90% of the
      total.
533 % These weights are INTEGERS (for compatibility with ParMETIS) thus not
      [0, 1].
534 % To balance memory usage (instead of computation) the point weight needs
      to be
535 % increased (especially for explicit time integration methods).
536 PARMETIS_EDGE_WEIGHT= 1
537 PARMETIS_POINT_WEIGHT= 0
538 %
539 %
540 % ----- SCREEN/HISTORY VOLUME OUTPUT

```

```

    _____%
541 %
542 % Screen output fields (use 'SU2_CFD -d <config_file>' to view list of
    available fields)
543 SCREEN_OUTPUT= (INNER_ITER, PRESSURE, RMS_VELOCITY, SIDEFORCE, DRAG, LIFT
    )
544 %
545 % History output groups (use 'SU2_CFD -d <config_file>' to view list of
    available fields)
546 HISTORY_OUTPUT= (ITER, RMS_PRESSURE, RMS_VELOCITY, SIDEFORCE, DRAG, LIFT)
547 %
548 % Volume output fields/groups (use 'SU2_CFD -d <config_file>' to view
    list of available fields)
549 VOLUME_OUTPUT= (COORDINATES, SOLUTION, PRIMITIVE)
550 %
551 % Writing frequency for screen output
552 SCREEN_WRT_FREQ_INNER= 1
553 %
554 SCREEN_WRT_FREQ_OUTER= 1
555 %
556 SCREEN_WRT_FREQ_TIME= 1
557 %
558 % Writing frequency for history output
559 HISTORY_WRT_FREQ_INNER= 1
560 %
561 HISTORY_WRT_FREQ_OUTER= 1
562 %
563 HISTORY_WRT_FREQ_TIME= 1
564 %
565 % list of writing frequencies corresponding to the list in OUTPUT_FILES
566 OUTPUT_WRT_FREQ= 250, 250, 300
567 %
568 % Output the performance summary to the console at the end of SU2_CFD
569 WRT_PERFORMANCE= NO
570 %
571 % Overwrite or append iteration number to the restart files when saving
572 WRT_RESTART_OVERWRITE= YES
573 %
574 % Overwrite or append iteration number to the surface files when saving
575 WRT_SURFACE_OVERWRITE= YES
576 %
577 % Overwrite or append iteration number to the volume files when saving
578 WRT_VOLUME_OVERWRITE= YES
579 %
580 % _____ INPUT/OUTPUT FILE INFORMATION
    _____%
581 %
582 % Mesh input file
583 MESH_FILENAME= mesh_hx.su2

```

```

584 %
585 % Mesh input file format (SU2, CGNS)
586 MESH_FORMAT= SU2
587 %
588 % Mesh output file
589 MESH_OUT_FILENAME= mesh_hx.su2
590 %
591 % Restart flow input file
592 SOLUTION_FILENAME= solution_flow.dat
593 %
594 % Restart adjoint input file
595 SOLUTION_ADJ_FILENAME= solution_adj.dat
596 %
597 % Output tabular file format (TECPLOT, CSV)
598 TABULAR_FORMAT= CSV
599 %
600 % Files to output
601 % Possible formats : (TECPLOT_ASCII, TECPLOT, SURFACE_TECPLT_ASCII,
602 % SURFACE_TECPLT, CSV, SURFACE_CSV, PARAVIEW_ASCII, PARAVIEW_LEGACY,
603 % SURFACE_PARAVIEW_LEGACY, PARAVIEW, SURFACE_PARAVIEW, RESTART_ASCII,
604 % RESTART, CGNS, SURFACE_CGNS, STL_ASCII, STL_BINARY)
605 % default : (RESTART, PARAVIEW, SURFACE_PARAVIEW)
606 OUTPUT_FILES= (RESTART, PARAVIEW, SURFACE_PARAVIEW)
607 %
608 % Output file convergence history (w/o extension)
609 CONV_FILENAME= history
610 %
611 % Output file with the forces breakdown
612 BREAKDOWN_FILENAME= forces_breakdown.dat
613 %
614 % Output file restart flow
615 RESTART_FILENAME= restart_flow.dat
616 %
617 % Output file flow (w/o extension) variables
618 VOLUME_FILENAME= flow

```

## L.2 Caso hinf

Listagem L.2: hinf.cfg

```

1 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
2 %
3 % SU2 configuration file

```

%

```

4 % Case description: Tyrrell 026 Front Wing in Wind
   Tunnel_____ %
5 % Author: Matheus Ribeiro
   Vidal_____ %
6 % Institution: Universidade de
   Brasilia_____ %
7 % Date: 28/01/2023
                                     %
8 % File Version 7.3.1 "Blackbird"
                                     %
9 %
   %
10 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
11
12 % _____ DIRECT, ADJOINT, AND LINEARIZED PROBLEM DEFINITION
   _____%
13 %
14 % Solver type (EULER, NAVIER_STOKES, RANS,
15 %             INC_EULER, INC_NAVIER_STOKES, INC_RANS,
16 %             NEMO_EULER, NEMO_NAVIER_STOKES,
17 %             FEM_EULER, FEM_NAVIER_STOKES, FEM_RANS, FEM_LES,
18 %             HEAT_EQUATION_FVM, ELASTICITY)
19 SOLVER= INC_RANS
20 %
21 % Specify turbulence model (NONE, SA, SA_NEG, SST, SA_E, SA_COMP,
   SA_E_COMP, SST_SUST)
22 KIND_TURB_MODEL= SST
23 %
24 % Mathematical problem (DIRECT, CONTINUOUS_ADJOINT, DISCRETE_ADJOINT)
25 % Defaults to DISCRETE_ADJOINT for the SU2*_AD codes , and to DIRECT
   otherwise .
26 MATH_PROBLEM= DIRECT
27 %
28 % System of measurements (SI , US)
29 % International system of units (SI): ( meters , kilograms , Kelvins ,
30 %                                     Newtons = kg m/s^2, Pascals = N/m
   ^2,
31 %                                     Density = kg/m^3, Speed = m/s ,
32 %                                     Equiv. Area = m^2 )
33 SYSTEM_MEASUREMENTS= SI
34 %
35 %
36 % _____ SOLVER CONTROL
   _____%
37 %
38 % Number of iterations for single-zone problems
39 ITER= 1

```

```

40 %
41 % Maximum number of inner iterations
42 INNER_ITER= 15000
43 %
44 % Maximum number of outer iterations (only for multizone problems)
45 OUTER_ITER= 1
46 %
47 % Maximum number of time iterations
48 TIME_ITER= 1
49 %
50 % Convergence field
51 CONV_FIELD= PRESSURE
52 %
53 % Min value of the residual (log10 of the residual)
54 CONV_RESIDUAL_MINVAL= -8
55 %
56 % Start convergence criteria at iteration number
57 CONV_STARTITER= 10
58 %
59 % Number of elements to apply the criteria
60 CONV_CAUCHY_ELEMS= 100
61 %
62 % Epsilon to control the series convergence
63 CONV_CAUCHY_EPS= 1E-10
64 %
65 % Iteration number to begin unsteady restarts
66 RESTART_ITER= 0
67 %
68 %% Time convergence monitoring
69 WINDOW_CAUCHY_CRIT = YES
70 %
71 % List of time convergence fields
72 CONV_WINDOW_FIELD = (TAVG_DRAG, TAVG_LIFT)
73 %
74 % Time Convergence Monitoring starts at Iteration WINDOW_START_ITER +
      CONV_WINDOW_STARTITER
75 CONV_WINDOW_STARTITER = 0
76 %
77 % Epsilon to control the series convergence
78 CONV_WINDOW_CAUCHY_EPS = 1E-3
79 %
80 % Number of elements to apply the criteria
81 CONV_WINDOW_CAUCHY_ELEMS = 10
82 %
83 % ----- TIME-DEPENDENT SIMULATION
      -----%
84 %
85 % Time domain simulation
86 TIME_DOMAIN= NO

```



```

87 %
88 % Unsteady simulation (NO, TIME_STEPPING, DUAL_TIME_STEPPING-1ST_ORDER,
89 %                      DUAL_TIME_STEPPING-2ND_ORDER, HARMONIC_BALANCE)
90 TIME_MARCHING= NO
91 %
92 % Time Step for dual time stepping simulations (s) — Only used when
    UNST_CFL_NUMBER = 0.0
93 % For the DG-FEM solver it is used as a synchronization time when
    UNST_CFL_NUMBER != 0.0
94 TIME_STEP= 0.00001
95 %
96 % Total Physical Time for dual time stepping simulations (s)
97 MAX_TIME= 2.0
98 %
99 % Unsteady Courant-Friedrichs-Lewy number of the finest grid
100 UNST_CFL_NUMBER= 1.0
101 %
102 %% Windowed output time averaging
103 % Time iteration to start the windowed time average in a direct run
104 WINDOW_START_ITER = 500
105 %
106 % Window used for reverse sweep and direct run. Options (SQUARE, HANN,
    HANN_SQUARE, BUMP) Square is default.
107 WINDOW_FUNCTION = SQUARE
108 %
109 % ----- DES Parameters
    -----%
110 %
111 % Specify Hybrid RANS/LES model (SA_DES, SA_DDES, SA_ZDES, SA_EDDES)
112 HYBRID_RANSLES= SA_DDES
113 %
114 % DES Constant (0.65)
115 DES_CONST= 0.65
116 %
117 % ----- INCOMPRESSIBLE FLOW CONDITION DEFINITION
    -----%
118 %
119 % Density model within the incompressible flow solver.
120 % Options are CONSTANT (default), BOUSSINESQ, or VARIABLE. If VARIABLE,
121 % an appropriate fluid model must be selected.
122 INC_DENSITY_MODEL= CONSTANT
123 %
124 % Solve the energy equation in the incompressible flow solver
125 INC_ENERGY_EQUATION = NO
126 %
127 % Initial density for incompressible flows
128 % (1.2886 kg/m^3 by default (air), 998.2 Kg/m^3 (water))
129 INC_DENSITY_INIT= 1.2886
130 %

```

```

131 % Initial velocity for incompressible flows (1.0,0,0 m/s by default)
132 INC_VELOCITY_INIT= ( 30.0, 0.0, 0.0 )
133 %
134 % Initial temperature for incompressible flows that include the
135 % energy equation (288.15 K by default). Value is ignored if
136 % INC_ENERGY_EQUATION is false.
137 INC_TEMPERATURE_INIT= 288.15
138 %
139 % Non-dimensionalization scheme for incompressible flows. Options are
140 % INITIAL_VALUES (default), REFERENCE_VALUES, or DIMENSIONAL.
141 % INC*_REF values are ignored unless REFERENCE_VALUES is chosen.
142 INC_NONDIM= INITIAL_VALUES
143 %
144 % Reference density for incompressible flows (1.0 kg/m3 by default)
145 INC_DENSITY_REF= 1.0
146 %
147 % Reference velocity for incompressible flows (1.0 m/s by default)
148 INC_VELOCITY_REF= 1.0
149 %
150 % Reference temperature for incompressible flows that include the
151 % energy equation (1.0 K by default)
152 INC_TEMPERATURE_REF = 1.0
153 %
154 % List of inlet types for incompressible flows. List length must
155 % match number of inlet markers. Options: VELOCITY_INLET, PRESSURE_INLET.
156 INC_INLET_TYPE= VELOCITY_INLET
157 %
158 % Damping coefficient for iterative updates at pressure inlets. (0.1 by
    default)
159 INC_INLET_DAMPING= 0.1
160 %
161 % List of outlet types for incompressible flows. List length must
162 % match number of outlet markers. Options: PRESSURE_OUTLET,
    MASS_FLOW_OUTLET
163 INC_OUTLET_TYPE= PRESSURE_OUTLET
164 %
165 % Damping coefficient for iterative updates at mass flow outlets. (0.1 by
    default)
166 INC_OUTLET_DAMPING= 0.1
167 %
168 % Epsilon2 multiplier in Beta calculation for incompressible
    preconditioner.
169 BETA_FACTOR= 4.1
170 %
171 % ----- REFERENCE VALUE DEFINITION
    -----%
172 %
173 % Reference origin for moment computation (m or in) % Atualmente em um
    quarto de Asa

```

```

174 REF_ORIGIN_MOMENT_X = 0.00
175 REF_ORIGIN_MOMENT_Y = 0.00
176 REF_ORIGIN_MOMENT_Z = 0.00
177 %
178 % Reference length for moment non-dimensional coefficients (m or in)
179 REF_LENGTH= 0.380
180 %
181 % Reference area for non-dimensional force coefficients (0 implies
    automatic
182 % calculation) (m^2 or in^2)
183 REF_AREA= 0.405
184 %
185 % Aircraft semi-span (0 implies automatic calculation) (m or in)
186 SEMI_SPAN= 0.0
187 %
188 % ----- VISCOSITY MODEL
    -----%
189 %
190 % Viscosity model (SUTHERLAND, CONSTANT_VISCOSITY, POLYNOMIAL_VISCOSITY).
191 VISCOSITY_MODEL= CONSTANT_VISCOSITY
192
193 % Molecular Viscosity that would be constant (1.716E-5 by default)
194 MU_CONSTANT= 1.716E-5
195 %
196 % Sutherland Viscosity Ref (1.716E-5 default value for AIR SI)
197 MU_REF= 1.716E-5
198 %
199 % Sutherland Temperature Ref (273.15 K default value for AIR SI)
200 MU_T_REF= 273.15
201 %
202 % Sutherland constant (110.4 default value for AIR SI)
203 SUTHERLAND_CONSTANT= 110.4
204 %
205 % Temperature polynomial coefficients (up to quartic) for viscosity.
206 % Format -> Mu(T) : b0 + b1*T + b2*T^2 + b3*T^3 + b4*T^4
207 MU_POLYCOEFFS= (0.0, 0.0, 0.0, 0.0, 0.0)
208 %
209 % ----- DYNAMIC MESH DEFINITION
    -----%
210 %
211 % Type of dynamic surface movement (NONE, DEFORMING, MOVING_WALL,
212 % AEROELASTIC, AEROELASTIC_RIGID_MOTION EXTERNAL, EXTERNAL_ROTATION)
213 SURFACE_MOVEMENT= NONE
214 %
215 % Moving wall boundary marker(s) (NONE = no marker, ignored for
    RIGID_MOTION)
216 % MARKER_MOVING= ( NONE )
217 %
218 % Coordinates of the motion origin

```

```

219 % SURFACE_MOTION_ORIGIN= -4 0.0 0.0
220 %
221 % Translational velocity (m/s or ft/s) in the x, y, & z directions
222 % SURFACE_TRANSLATION_RATE = 30.0 0.0 0.0
223 %
224 %
225 % Move Motion Origin for marker moving (1 or 0)
226 % MOVE_MOTION_ORIGIN = 0
227 %
228 % ----- BOUNDARY CONDITION DEFINITION
      % -----%
229 %
230 % Euler wall boundary marker(s) (NONE = no marker)
231 % Implementation identical to MARKER_SYM.
232 MARKER_EULER= ( NONE )
233 %
234 % Navier-Stokes (no-slip), constant heat flux wall marker(s) (NONE = no
      marker)
235 % Format: ( marker name, constant heat flux (J/m2), ... )
236 MARKER_HEATFLUX= ( wing, 0)
237 %
238 % Far-field boundary marker(s) (NONE = no marker)
239 MARKER_FAR= ( inlet , outlet )
240 %
241 % Inlet boundary type (TOTAL_CONDITIONS, MASS_FLOW)
242 %
243 %INLET_TYPE= TOTAL_CONDITIONS
244 %
245 % Inlet boundary marker(s) with the following formats (NONE = no marker)
246 % Total Conditions: (inlet marker, total temp, total pressure ,
      flow_direction_x ,
247 %          flow_direction_y, flow_direction_z, ... ) where
      flow_direction is
248 %          a unit vector.
249 % Mass Flow: (inlet marker, density , velocity magnitude, flow_direction_x
      ,
250 %          flow_direction_y, flow_direction_z, ... ) where
      flow_direction is
251 %          a unit vector.
252 % Inc. Velocity: (inlet marker, temperature, velocity magnitude ,
      flow_direction_x ,
253 %          flow_direction_y, flow_direction_z, ... ) where
      flow_direction is
254 %          a unit vector.
255 % Inc. Pressure: (inlet marker, temperature, total pressure ,
      flow_direction_x ,
256 %          flow_direction_y, flow_direction_z, ... ) where
      flow_direction is
257 %          a unit vector.

```

```

258 %MARKER_INLET= ( INLET, 288.15, 101325.0, 1.0, 0.0, 0.0)
259 %
260 % Outlet boundary marker(s) (NONE = no marker)
261 % Compressible: ( outlet marker, back pressure (static thermodynamic),
    ... )
262 % Inc. Pressure: ( outlet marker, back pressure (static gauge in Pa), ...
    )
263 % Inc. Mass Flow: ( outlet marker, mass flow target (kg/s), ... )
264 %MARKER_OUTLET= ( OUTLET )
265 %
266 % ----- WALL FUNCTION DEFINITION
    -----%
267 %
268 % The von Karman constant, the constant below only affects the standard
    wall function model
269 WALLMODEL_KAPPA= 0.41
270 %
271 % The wall function model constant B
272 WALLMODEL_B= 5.5
273 %
274 % The y+ value below which the wall function is switched off and we
    resolve the wall
275 WALLMODEL_MINYPPLUS= 5.0
276 %
277 % [Expert] Max Newton iterations used for the standard wall function
278 WALLMODEL_MAXITER= 200
279 %
280 % [Expert] relaxation factor for the Newton iterations of the standard
    wall function
281 WALLMODEL_RELFAC= 0.5
282
283 % ----- SURFACES IDENTIFICATION
    -----%
284 %
285 % Marker(s) of the surface in the surface flow solution file
286 MARKER_PLOTTING = ( wing )
287 %
288 % Marker(s) of the surface where the non-dimensional coefficients are
    evaluated.
289 MARKER_MONITORING = ( wing )
290 %
291 % Viscous wall markers for which wall functions must be applied. (NONE =
    no marker)
292 % Format: ( marker name, wall function type -NO_WALL_FUNCTION,
    STANDARD_WALL_FUNCTION,
293 %          ADAPTIVE_WALL_FUNCTION, SCALABLE_WALL_FUNCTION,
    EQUILIBRIUM_WALL_MODEL,
294 %          NONEQUILIBRIUM_WALL_MODEL-, ... )
295 MARKER_WALL_FUNCTIONS= ( wing, NO_WALL_FUNCTION )

```



```

296 %
297 % Marker(s) of the surface where custom thermal BCs are defined.
298 MARKER_PYTHON_CUSTOM = ( NONE )
299 %
300 % Marker(s) of the surface that is going to be analyzed in detail (
      massflow , average pressure , distortion , etc)
301 MARKER_ANALYZE = ( wing )
302 %
303 % Method to compute the average value in MARKER_ANALYZE (AREA, MASSFLUX).
304 MARKER_ANALYZE_AVERAGE = AREA
305
306 % ----- COMMON PARAMETERS DEFINING THE NUMERICAL METHOD
      -----%
307 %
308 % Numerical method for spatial gradients (GREEN_GAUSS,
      WEIGHTED_LEAST_SQUARES)
309 NUM_METHOD_GRAD= GREEN_GAUSS
310
311 % Numerical method for spatial gradients to be used for MUSCL
      reconstruction
312 % Options are (GREEN_GAUSS, WEIGHTED_LEAST_SQUARES, LEAST_SQUARES).
      Default value is
313 % NONE and the method specified in NUM_METHOD_GRAD is used.
314 NUM_METHOD_GRAD_RECON = NONE
315 %
316 % CFL number (initial value for the adaptive CFL number)
317 CFL_NUMBER= 1.0
318 %
319 % Adaptive CFL number (NO, YES)
320 CFL_ADAPT= NO
321 %
322 % Parameters of the adaptive CFL number (factor-down, factor-up, CFL min
      value ,
323 %
      CFL max value , acceptable linear
      solver convergence)
324 % Local CFL increases by factor-up until max if the solution rate of
      change is not limited ,
325 % and acceptable linear convergence is achieved. It is reduced if rate is
      limited , or if there
326 % is not enough linear convergence , or if the nonlinear residuals are
      stagnant and oscillatory .
327 % It is reset back to min when linear solvers diverge , or if nonlinear
      residuals increase too much.
328 CFL_ADAPT_PARAM= ( 0.1 , 2.0 , 10.0 , 1e10 , 0.001 )
329 %
330 % Maximum Delta Time in local time stepping simulations
331 MAX_DELTA_TIME= 1E6
332 %
333 % Runge-Kutta alpha coefficients

```

```

334 RK_ALPHA_COEFF= ( 0.66667, 0.66667, 1.00000 )
335 %
336 % Objective function in gradient evaluation (DRAG, LIFT, SIDEFORCE,
      MOMENT_X,
337 %                                     MOMENT_Y, MOMENT_Z,
      EFFICIENCY, BUFFET,
338 %                                     EQUIVALENT_AREA,
      NEARFIELD_PRESSURE,
339 %                                     FORCE_X, FORCE_Y, FORCE_Z,
      THRUST,
340 %                                     TORQUE, TOTAL_HEATFLUX,
      CUSTOM_OBJFUNC
341 %                                     MAXIMUM_HEATFLUX,
      INVERSE_DESIGN_PRESSURE,
342 %                                     INVERSE_DESIGN_HEATFLUX,
      SURFACE_TOTAL_PRESSURE,
343 %                                     SURFACE_MASSFLOW,
      SURFACE_STATIC_PRESSURE, SURFACE_MACH)
344 % For a weighted sum of objectives: separate by commas, add
      OBJECTIVE_WEIGHT and MARKER_MONITORING in matching order.
345 OBJECTIVE_FUNCTION= DRAG
346 %
347 % List of weighting values when using more than one OBJECTIVE_FUNCTION.
      Separate by commas and match with MARKER_MONITORING.
348 OBJECTIVE_WEIGHT = 1.0
349 %
350 % Expression used when "OBJECTIVE_FUNCTION= CUSTOM_OBJFUNC", any history/
      screen output can be used together with common
351 % math functions (sqrt, cos, exp, etc.). This can be used for constraint
      aggregation (as below) or to compute something
352 % SU2 does not, see TestCases/user_defined_functions/.
353 CUSTOM_OBJFUNC= 'DRAG + 10 * pow(fmax(0.4-LIFT, 0), 2) '
354
355 % ----- SLOPE LIMITER AND DISSIPATION SENSOR DEFINITION
      -----%
356 %
357 % Monotonic Upwind Scheme for Conservation Laws (TVD) in the flow
      equations.
358 %           Required for 2nd order upwind schemes (NO, YES)
359 MUSCL_FLOW= YES
360 %
361 % Slope limiter (NONE, VENKATAKRISHNAN, VENKATAKRISHNAN_WANG,
362 %               BARTH_JESPERSEN, VAN_ALBADA_EDGE)
363 SLOPE_LIMITER_FLOW= VENKATAKRISHNAN
364 %
365 % Monotonic Upwind Scheme for Conservation Laws (TVD) in the turbulence
      equations.
366 %           Required for 2nd order upwind schemes (NO, YES)
367 MUSCL_TURB= NO

```

```

368 %
369 % Slope limiter (NONE, VENKATAKRISHNAN, VENKATAKRISHNAN_WANG,
370 %           BARTH_JESPERSEN, VAN_ALBADA_EDGE)
371 SLOPE_LIMITER_TURB= VENKATAKRISHNAN
372 %
373 % Monotonic Upwind Scheme for Conservation Laws (TVD) in the adjoint flow
      equations.
374 %           Required for 2nd order upwind schemes (NO, YES)
375 MUSCL_ADJFLOW= YES
376 %
377 % Slope limiter (NONE, VENKATAKRISHNAN, BARTH_JESPERSEN, VAN_ALBADA_EDGE,
378 %           SHARP_EDGES, WALL_DISTANCE)
379 SLOPE_LIMITER_ADJFLOW= VENKATAKRISHNAN
380 %
381 % Monotonic Upwind Scheme for Conservation Laws (TVD) in the turbulence
      adjoint equations.
382 %           Required for 2nd order upwind schemes (NO, YES)
383 MUSCL_ADJTURB= NO
384 %
385 % Slope limiter (NONE, VENKATAKRISHNAN, BARTH_JESPERSEN, VAN_ALBADA_EDGE)
386 SLOPE_LIMITER_ADJTURB= VENKATAKRISHNAN
387 %
388 % Coefficient for the Venkats limiter (upwind scheme). A larger values
      decrease
389 %           the extent of limiting , values approaching zero cause
390 %           lower-order approximation to the solution (0.05 by default)
391 VENKAT_LIMITER_COEFF= 0.05
392 %
393 % Reference coefficient for detecting sharp edges (3.0 by default).
394 REF_SHARP_EDGES = 3.0
395 %
396 % Coefficient for the adjoint sharp edges limiter (3.0 by default).
397 ADJ_SHARP_LIMITER_COEFF= 3.0
398 %
399 % Remove sharp edges from the sensitivity evaluation (NO, YES)
400 SENS_REMOVE_SHARP = NO
401 %
402 % Freeze the value of the limiter after a number of iterations
403 LIMITER_ITER= 999999
404 %
405 % 1st order artificial dissipation coefficients for
406 %   the Lax&Friedrichs method ( 0.15 by default )
407 LAX_SENSOR_COEFF= 0.15
408 %
409 % 2nd and 4th order artificial dissipation coefficients for
410 %   the JST method ( 0.5, 0.02 by default )
411 JST_SENSOR_COEFF= ( 0.5, 0.02 )
412 %
413 % 1st order artificial dissipation coefficients for

```

```

414 % the adjoint Lax-Friedrichs method ( 0.15 by default )
415 ADJ_LAX_SENSOR_COEFF= 0.15
416 %
417 % 2nd, and 4th order artificial dissipation coefficients for
418 % the adjoint JST method ( 0.5, 0.02 by default )
419 ADJ_JST_SENSOR_COEFF= ( 0.5, 0.02 )
420
421 % ----- LINEAR SOLVER DEFINITION
422 % -----%
423 % Linear solver or smoother for implicit formulations:
424 % BCGSTAB, FGMRES, RESTARTED_FGMRES, CONJUGATE_GRADIENT (self-adjoint
425 % problems only), SMOOTHER.
426 %
427 % Same for discrete adjoint (smoothers not supported), replaces
428 % LINEAR_SOLVER in SU2_*_AD codes.
429 %
430 % Preconditioner of the Krylov linear solver or type of smoother (ILU,
431 % LU_SGS, LINELET, JACOBI)
432 %
433 % Same for discrete adjoint (JACOBI or ILU), replaces LINEAR_SOLVER_PREC
434 % in SU2_*_AD codes.
435 %
436 % Linear solver ILU preconditioner fill-in level (0 by default)
437 LINEAR_SOLVER_ILU_FILL_IN= 0
438 %
439 % Minimum error of the linear solver for implicit formulations
440 LINEAR_SOLVER_ERROR= 1E-6
441 %
442 % Max number of iterations of the linear solver for the implicit
443 % formulation
444 LINEAR_SOLVER_ITER= 5
445 %
446 % Restart frequency for RESTARTED_FGMRES
447 LINEAR_SOLVER_RESTART_FREQUENCY= 10
448 %
449 % Relaxation factor for smoother-type solvers (LINEAR_SOLVER= SMOOTHER)
450 LINEAR_SOLVER_SMOOTHER_RELAXATION= 1.0
451 % ----- FLOW NUMERICAL METHOD DEFINITION
452 % -----%
453 % Convective numerical method (JST, JST_KE, JST_MAT, LAX-FRIEDRICH, CUSP,
454 % ROE, AUSM,
455 %
456 % AUSMPLUSUP, AUSMPLUSUP2, AUSMPWPLUS, HLLC,

```

```

TURKEL_PREC,
455 % SW, MSW, FDS, SLAU, SLAU2, L2ROE, LMROE)
456 CONV_NUM_METHOD_FLOW= JST
457 %
458 % Roe Low Dissipation function for Hybrid RANS/LES simulations (FD, NTS,
NTS_DUCROS)
459 ROE_LOW DISSIPATION= FD
460 %
461 % Post-reconstruction correction for low Mach number flows (NO, YES)
462 LOW_MACH_CORR= NO
463 %
464 % Roe-Turkel preconditioning for low Mach number flows (NO, YES)
465 LOW_MACH_PREC= NO
466 %
467 % Use numerically computed Jacobians for AUSM+up(2) and SLAU(2)
468 % Slower per iteration but potentially more stable and capable of higher
CFL
469 USE_ACCURATE_FLUX_JACOBIANS= NO
470 %
471 % Use the vectorized version of the selected numerical method (available
for JST family and Roe).
472 % SU2 should be compiled for an AVX or AVX512 architecture for best
performance.
473 USE_VECTORIZATION= NO
474 %
475 % Entropy fix coefficient (0.0 implies no entropy fixing, 1.0 implies
scalar
476 % artificial dissipation)
477 ENTROPY_FIX_COEFF= 0.0
478 %
479 % Higher values than 1 (3 to 4) make the global Jacobian of central
schemes (compressible flow
480 % only) more diagonal dominant (but mathematically incorrect) so that
higher CFL can be used.
481 CENTRAL_JACOBIAN_FIX_FACTOR= 4.0
482 %
483 % Time discretization (RUNGE-KUTTA_EXPLICIT, EULER_IMPLICIT,
EULER_EXPLICIT)
484 TIME_DISCRE_FLOW= EULER_IMPLICIT
485 %
486 % Use a Newton-Krylov method on the flow equations, see TestCases/rans/
oneram6/turb_ONERAM6_nk.cfg
487 % For multizone discrete adjoint it will use FGMRES on inner iterations
with restart frequency
488 % equal to "QUASI_NEWTON_NUM_SAMPLES".
489 NEWTON_KRYLOV= NO
490 %
491 %
492 % ----- TURBULENT NUMERICAL METHOD DEFINITION

```



```

    _____%
493 %
494 % Convective numerical method (SCALAR_UPWIND)
495 CONV_NUM_METHOD_TURB= SCALAR_UPWIND
496 %
497 % Time discretization (EULER_IMPLICIT, EULER_EXPLICIT)
498 TIME_DISCRE_TURB= EULER_IMPLICIT
499 %
500 % Reduction factor of the CFL coefficient in the turbulence problem
501 CFL_REDUCTION_TURB= 1.0
502 %
503 % _____ HYBRID PARALLEL (MPI+OpenMP) OPTIONS
    _____%
504 %
505 % An advanced performance parameter for FVM solvers , a large-ish value
    should be best
506 % when relatively few threads per MPI rank are in use (~4). However,
    maximum parallelism
507 % is obtained with EDGE_COLORING_GROUP_SIZE=1, consider using this value
    only if SU2
508 % warns about low coloring efficiency during preprocessing (performance
    is usually worse).
509 % Setting the option to 0 disables coloring and a different strategy is
    used instead ,
510 % that strategy is automatically used when the coloring efficiency is
    less than 0.875.
511 % The optimum value/strategy is case-dependent.
512 EDGE_COLORING_GROUP_SIZE= 512
513 %
514 % Independent "threads per MPI rank" setting for LU-SGS and ILU
    preconditioners .
515 % For problems where time is spend mostly in the solution of linear
    systems (e.g. elasticity ,
516 % very high CFL central schemes), AND, if the memory bandwidth of the
    machine is saturated
517 % (4 or more cores per memory channel) better performance (via a
    reduction in linear iterations)
518 % may be possible by using a smaller value than that defined by the
    system or in the call to
519 % SU2_CFD (via the -t/--threads option).
520 % The default (0) means "same number of threads as for all else".
521 LINEAR_SOLVER_PREC_THREADS= 0
522 %
523 % _____ PARTITIONING OPTIONS (ParMETIS)
    _____ %
524 %
525 % Load balancing tolerance , lower values will make ParMETIS work harder
    to evenly
526 % distribute the work-estimate metric across all MPI ranks , at the

```

```

    expense of more
527 % edge cuts (i.e. increased communication cost).
528 PARMETIS_TOLERANCE= 0.02
529 %
530 % The work-estimate metric is a weighted function of the work-per-edge (e
    .g. spatial
531 % discretization, linear system solution) and of the work-per-point (e.g.
    source terms,
532 % temporal discretization) the former usually accounts for >90% of the
    total.
533 % These weights are INTEGERS (for compatibility with ParMETIS) thus not
    [0, 1].
534 % To balance memory usage (instead of computation) the point weight needs
    to be
535 % increased (especially for explicit time integration methods).
536 PARMETIS_EDGE_WEIGHT= 1
537 PARMETIS_POINT_WEIGHT= 0
538 %
539 %
540 % ----- SCREEN/HISTORY VOLUME OUTPUT
    -----%
541 %
542 % Screen output fields (use 'SU2_CFD -d <config_file>' to view list of
    available fields)
543 SCREEN_OUTPUT= (INNER_ITER, PRESSURE, RMS_VELOCITY, SIDEFORCE, DRAG, LIFT
    )
544 %
545 % History output groups (use 'SU2_CFD -d <config_file>' to view list of
    available fields)
546 HISTORY_OUTPUT= (ITER, RMS_PRESSURE, RMS_VELOCITY, SIDEFORCE, DRAG, LIFT)
547 %
548 % Volume output fields/groups (use 'SU2_CFD -d <config_file>' to view
    list of available fields)
549 VOLUME_OUTPUT= (COORDINATES, SOLUTION, PRIMITIVE)
550 %
551 % Writing frequency for screen output
552 SCREEN_WRT_FREQ_INNER= 1
553 %
554 SCREEN_WRT_FREQ_OUTER= 1
555 %
556 SCREEN_WRT_FREQ_TIME= 1
557 %
558 % Writing frequency for history output
559 HISTORY_WRT_FREQ_INNER= 1
560 %
561 HISTORY_WRT_FREQ_OUTER= 1
562 %
563 HISTORY_WRT_FREQ_TIME= 1
564 %

```

```

565 % list of writing frequencies corresponding to the list in OUTPUT_FILES
566 OUTPUT_WRT_FREQ= 250, 250, 300
567 %
568 % Output the performance summary to the console at the end of SU2_CFD
569 WRT_PERFORMANCE= NO
570 %
571 % Overwrite or append iteration number to the restart files when saving
572 WRT_RESTART_OVERWRITE= YES
573 %
574 % Overwrite or append iteration number to the surface files when saving
575 WRT_SURFACE_OVERWRITE= YES
576 %
577 % Overwrite or append iteration number to the volume files when saving
578 WRT_VOLUME_OVERWRITE= YES
579 %
580 % ----- INPUT/OUTPUT FILE INFORMATION
    % -----%
581 %
582 % Mesh input file
583 MESH_FILENAME= hinf.su2
584 %
585 % Mesh input file format (SU2, CGNS)
586 MESH_FORMAT= SU2
587 %
588 % Mesh output file
589 MESH_OUT_FILENAME= hinf.su2
590 %
591 % Restart flow input file
592 SOLUTION_FILENAME= solution_flow.dat
593 %
594 % Restart adjoint input file
595 SOLUTION_ADJ_FILENAME= solution_adj.dat
596 %
597 % Output tabular file format (TECPLOT, CSV)
598 TABULAR_FORMAT= CSV
599 %
600 % Files to output
601 % Possible formats : (TECPLOT_ASCII, TECPLOT, SURFACE_TECPLT_ASCII,
602 % SURFACE_TECPLT, CSV, SURFACE_CSV, PARAVIEW_ASCII, PARAVIEW_LEGACY,
    SURFACE_PARAVIEW_ASCII,
603 % SURFACE_PARAVIEW_LEGACY, PARAVIEW, SURFACE_PARAVIEW, RESTART_ASCII,
    RESTART, CGNS, SURFACE_CGNS, STL_ASCII, STL_BINARY)
604 % default : (RESTART, PARAVIEW, SURFACE_PARAVIEW)
605 OUTPUT_FILES= (RESTART, PARAVIEW, SURFACE_PARAVIEW)
606 %
607 % Output file convergence history (w/o extension)
608 CONV_FILENAME= history
609 %
610 % Output file with the forces breakdown

```

```
611 BREAKDOWN_FILENAME= forces_breakdown.dat
612 %
613 % Output file restart flow
614 RESTART_FILENAME= restart_flow.dat
615 %
616 % Output file flow (w/o extension) variables
617 VOLUME_FILENAME= flow
618 %
```