

Universidade de Brasília - UnB
Faculdade UnB Gama - FGA
Engenharia Aeroespacial

Desenvolvimento de uma metodologia de otimização de aerofólios e asas empregando algoritmos genéticos e métodos de painéis

Autor: Gustavo Weffort Fernandes

Orientador: Prof. Dr. Manuel Nascimento Dias Barcelos Junior

Brasília, DF

2022



Gustavo Weffort Fernandes

Desenvolvimento de uma metodologia de otimização de aerofólios e asas empregando algoritmos genéticos e métodos de painéis

Monografia submetida ao curso de graduação em Engenharia Aeroespacial da Universidade de Brasília, como requisito parcial para obtenção do Título de Bacharel em Engenharia Aeroespacial.

Universidade de Brasília - UnB

Faculdade UnB Gama - FGA

Orientador: Prof. Dr. Manuel Nascimento Dias Barcelos Junior

Brasília, DF

2022

Gustavo Weffort Fernandes

Desenvolvimento de uma metodologia de otimização de aerofólios e asas empregando algoritmos genéticos e métodos de painéis/ Gustavo Weffort Fernandes.
– Brasília, DF, 2022-

217 p. : il. (algumas color.) ; 30 cm.

Orientador: Prof. Dr. Manuel Nascimento Dias Barcelos Junior

Trabalho de Conclusão de Curso – Universidade de Brasília - UnB
Faculdade UnB Gama - FGA , 2022.

1. Algoritmo genético. 2. Aerofólios e asas. I. Prof. Dr. Manuel Nascimento Dias Barcelos Junior. II. Universidade de Brasília. III. Faculdade UnB Gama. IV. Desenvolvimento de uma metodologia de otimização de aerofólios e asas empregando algoritmos genéticos e métodos de painéis

CDU xx:xxx:xxx.x

Gustavo Weffort Fernandes

Desenvolvimento de uma metodologia de otimização de aerofólios e asas empregando algoritmos genéticos e métodos de painéis

Monografia submetida ao curso de graduação em Engenharia Aeroespacial da Universidade de Brasília, como requisito parcial para obtenção do Título de Bacharel em Engenharia Aeroespacial.

Trabalho aprovado. Brasília, DF, 6 de Maio de 2022:

**Prof. Dr. Manuel Nascimento Dias
Barcelos Junior**
Orientador

Prof. Dr. Fábio Alfaia da Cunha
Convidado 1

**Prof. Me. Luís Fernando Balduino de
Sousa**
Convidado 2

**Prof. Dr. Victor Rafael Rezende
Celestino**
Convidado 3

Brasília, DF
2022

Agradecimentos

Aos meus pais, por todo o suporte incondicional, tanto na vida acadêmica quanto fora dela. Aos meus amigos e família, por todo o entusiasmo e auxílio. Ao meu orientador, por toda a paciência e boas dicas desde nosso projeto de iniciação científica. Aos meus colegas da Dragonfly e da Capital Rocket Team, pelos inúmeros momentos proporcionados que contribuíram para meu crescimento tanto como profissional quanto como pessoa. Aos professores que me cativaram. Aos funcionários que permitem a UnB funcionar. E, não menos importante, a todos os cientistas, artistas, engenheiros, inventores, escritores, músicos e programadores que, alguma vez, de alguma forma, já me inspiraram.

*"Qualquer tecnologia suficientemente avançada é indistinguível de magia."
(Arthur C. Clarke)*

Resumo

Neste trabalho é apresentada uma metodologia de otimização de aerofólios e asas utilizando algoritmos genéticos. A proposta foi desenvolver algoritmos acessíveis e abrangentes, escritos em Octave/MATLAB e Python, que podem ser configurados e adaptados de acordo com as preferências do usuário. Os algoritmos permitem a configuração das funções objetivo e restrições em termos de quatro coeficientes aerodinâmicos distintos (coeficiente de sustentação, coeficiente de arrasto, razão L/D e coeficiente de momento de arfagem), em várias combinações possíveis, e utilizam duas parametrizações geométricas para os aerofólios: a série NACA 4 dígitos e o método CST. No total há seis tipos de algoritmos distintos - três para aerofólios e três para asas e seus aerofólios. A avaliação dos aerofólios é realizada utilizando o XFOIL (solucionador em método de painéis para aerofólios), e a avaliação das asas é realizada com o APAME (solucionador em método de painéis para corpos tridimensionais), primariamente, e com o VSPAERO (solucionador em método de painéis e VLM para corpos tridimensionais), como uma opção secundária. Após testes de validação do algoritmo genético em si, testes dos parâmetros de configuração, e testes de replicação com bibliografia anterior, foram estabelecidos cinco estudos de caso, utilizando múltiplas configurações distintas (funções objetivo, condições de voo, entre outros) para fornecer exemplos de como os algoritmos poderiam ser aplicados. Todos os testes foram bem sucedidos dentro dos objetivos estipulados. Os códigos desenvolvidos foram postados no GitHub¹. Assim, o presente trabalho teve o propósito de demonstrar a eficiência e aplicabilidade da metodologia adotada, de modo que os algoritmos desenvolvidos possam ser utilizados por outrem com ou sem modificações.

Palavras-chaves: Algoritmo genético. Aerofólios e asas.

¹ (FERNANDES, 2022)

Abstract

This work presents a methodology of optimization of airfoils and wings through genetic algorithms. The plan was to develop accessible and in-depth algorithms which may be configured and adapted according to the user's preference. The optimization algorithms allow for the configuration of the objective functions and constraints in terms of four distinct aerodynamic coefficients (lift coefficient, drag coefficient, L/D ratio and pitch moment coefficient), in various possible combinations, and they employ two geometric parametrizations for the airfoils: the NACA 4-digit series and the CST. In total there are six different types of algorithms - three for airfoils and three for wings and their airfoils. The evaluation of the airfoils is done with XFOIL (panel method solver for airfoils), and the evaluation of the wings is done with APAME (panel method solver for three dimensional bodies), primarily, and with VSPAERO (panel method and VLM solver for three dimensional bodies), as a secondary option. After performing validation tests on the genetic algorithm itself, configuration parameters tests, and replicatio tests with previous bibliography, five case studies were defined, using multiple distinct configurations (objective functions, flight conditions, and others) to provide examples on how the algorithms could be utilized. All tests were successful within the stipulated objectives. The code developed was posted to GitHub². Therefore, the purpose of the present work was to demonstrate the efficiency and the applicability of the adopted methodology, so that the developed algorithms may be used by others with or without modifications.

Key-words: Genetic algorithms. Airfoils and wings.

² (FERNANDES, 2022)

Lista de ilustrações

Figura 1 – Problema da caixa de papelão como visto em (LEITHOLD, 1986)	36
Figura 2 – Antena utilizada na missão ST5 da NASA, como visto em (HORNBY et al., 2006)	37
Figura 3 – Esquemático do funcionamento básico de um algoritmo genético (fonte: autor)	45
Figura 4 – Exemplo de uma operação de <i>crossover</i> (YANG, 2010)	46
Figura 5 – Uma combinação de escoamento uniforme e fonte (ANDERSON, 2017)	49
Figura 6 – Uma combinação de escoamento uniforme e fonte (ANDERSON, 2017)	50
Figura 7 – Geometria e nomenclatura de um aerofólio discretizado em painéis (ANDERSON, 2017)	51
Figura 8 – Modelo em painéis 3D do F-16XL (ERICKSON, 1990)	55
Figura 9 – Representação de uma asa arbitrária em termos da teoria do VLM (ANDERSON, 2017)	56
Figura 10 – Nomenclatura de um aerofólio (fonte: autor)	57
Figura 11 – Forças e momentos agindo no aerofólio (fonte: autor)	57
Figura 12 – Um aerofólio montado em um túnel de vento (LARSON, 2021)	58
Figura 13 – O aerofólio de um planador ASW 28-18 visível como um corte transversal na asa (MÜNCH, 2021)	59
Figura 14 – Ilustração de um aerofólio NACA 2412 (ABBOTT; DOENHOFF, 1959)	61
Figura 15 – Exemplo de geometria gerado pelo CST, usando função <i>shape</i> unitária (fonte: autor)	62
Figura 16 – Exemplos de tipos distintos de geometria CST (fonte: autor)	63
Figura 17 – Contornos de extradorso gerados pela Eq. 2.32 ilustrando os efeitos das variáveis de <i>design</i> (fonte: autor)	64
Figura 18 – Geometria dos painéis no XFOIL (DRELA, 1989)	65
Figura 19 – Coordenadas locais dos painéis (DRELA, 1989)	67
Figura 20 – Supermarine Spitfire e suas asas elípticas (CROWN, 2021)	68
Figura 21 – Exemplos de tipos diferentes de razão de aspecto	69
Figura 22 – Uma asa elíptica gerando uma distribuição elíptica de sustentação (GUDMUNDSSON, 2014)	70
Figura 23 – Projeto de uma asa baseado em uma asa de referência, com geometria mais simples (RAYMER, 2018)	71
Figura 24 – Painéis de trilha em uma asa no APAME garantindo a condição de Kutta (FILKOVIC, 2008)	72
Figura 25 – Geometrias do E190-E2 no OpenVSP (ACHARD, 2019)	74
Figura 26 – Representação das ferramentas na forma de módulos (fonte: autor)	76

Figura 27 – Tela inicial do XFOIL V 6.99 (fonte: autor)	79
Figura 28 – Ilustração de uma seleção em roleta (RAO, 2009)	84
Figura 29 – Primeiro tipo de <i>crossover</i> no algoritmo NACA 4 dígitos (fonte: autor)	85
Figura 30 – Segundo tipo de <i>crossover</i> no algoritmo NACA 4 dígitos (fonte: autor)	85
Figura 31 – Terceiro tipo de <i>crossover</i> no algoritmo NACA 4 dígitos (fonte: autor)	86
Figura 32 – Um aerofólio CST com curvatura negativa (fonte: autor)	90
Figura 33 – Primeiro tipo de <i>crossover</i> no algoritmo CST (fonte: autor)	91
Figura 34 – Segundo tipo de <i>crossover</i> no algoritmo CST (fonte: autor)	92
Figura 35 – Terceiro tipo de <i>crossover</i> no algoritmo CST (fonte: autor)	92
Figura 36 – Quarto tipo de <i>crossover</i> no algoritmo CST (fonte: autor)	92
Figura 37 – Quinto tipo de <i>crossover</i> no algoritmo CST (fonte: autor)	92
Figura 38 – Locais das seções transversais da planta de uma asa genérica (fonte: autor)	98
Figura 39 – Exemplo de interpolação de aerofólios (fonte: autor)	98
Figura 40 – Tela do solucionador avulso do APAME (fonte: autor)	100
Figura 41 – Comparação da evolução dos melhores resultados em configurações distintas de mutação	113
Figura 42 – Comparação da evolução dos melhores resultados em configurações distintas de elitismo no algoritmo NACA 4 dígitos (fonte: autor)	114
Figura 43 – Médias dos coeficientes da população completa em cada iteração no algoritmo NACA 4 dígitos com elitismo desativado (fonte: autor)	115
Figura 44 – Comparação da evolução dos melhores resultados em configurações distintas de elitismo no algoritmo CST (fonte: autor)	117
Figura 45 – Médias dos coeficientes da população completa em cada iteração no algoritmo CST com elitismo desativado (fonte: autor)	118
Figura 46 – Aerofólios resultantes do caso 1 da primeira comparação	120
Figura 47 – Aerofólios resultantes do caso 2 da primeira comparação	120
Figura 48 – Comparação das plantas de asas resultantes do teste de replicação (fonte: autor)	121
Figura 49 – Comparação dos aerofólios resultantes do teste de replicação (fonte: autor)	122
Figura 50 – Aerofólios resultantes dos testes independentes do algoritmo NACA 4 dígitos (fonte: autor)	134
Figura 51 – Aerofólios resultantes dos testes independentes do algoritmo CST (fonte: autor)	135
Figura 52 – Asas resultantes do algoritmo NACA 4 dígitos 3D para o primeiro estudo de caso	136
Figura 53 – Asas resultantes do algoritmo CST 3D para o primeiro estudo de caso	137

Figura 54 – Comparação do aerofólio inicial obtido com suas respectivas otimizações no segundo estudo de caso (fonte: autor)	138
Figura 55 – Aerofólios obtidos no terceiro estudo de caso (fonte: autor)	139
Figura 56 – Gráfico de α versus CM dos perfis CST obtidos no terceiro estudo de caso, traçado no XFLR5 (fonte: autor)	140
Figura 57 – Asas e aerofólios resultantes dos algoritmos CST, CST 3D e CST 3D específico para o quarto estudo de caso (fonte: autor)	142
Figura 58 – Asas e aerofólios resultantes do algoritmo NACA 4 dígitos para o quinto estudo de caso (fonte: autor)	144
Figura 59 – Aerofólio CST obtido em um trabalho anterior (FERNANDES; JÚNIOR, 2021)	146
Figura 60 – Menu de edição de seções de plantas de asa no OpenVSP (fonte: autor)	147
Figura 61 – Escoamento passando por entre duas linhas de corrente distintas (ANDERSON, 2017)	159
Figura 62 – Curva fechada em um dado escoamento para o cálculo da circulação (ANDERSON, 2017)	165
Figura 63 – Escoamento uniforme orientado a uma direção arbitrária (HOUGHTON et al., 2017)	167
Figura 64 – Escoamento surgindo de uma fonte (HOUGHTON et al., 2017)	168
Figura 65 – Escoamento em vórtice (HOUGHTON et al., 2017)	169
Figura 66 – Efeito do cumprimento da condição de Kutta no escoamento ao redor do aerofólio (HOUGHTON et al., 2017)	171
Figura 67 – Relações das velocidades do ar quando a condição de Kutta é satisfeita (HOUGHTON et al., 2017)	171
Figura 68 – Contornos de velocidade de uma simulação CFD (fonte: autor)	174
Figura 69 – Resultados do experimento de Reynolds referente a escoamentos laminares e turbulentos dentro de um tubo (WIKIPEDIA, 2021b)	175
Figura 70 – Experimento em túnel de vento demonstrando a separação do ar sobre um aerofólio (WIKIPEDIA, 2021a)	175
Figura 71 – Metade de uma distribuição cosseno para uso na geração de coordenadas de aerofólio (fonte: autor)	184
Figura 72 – Perfil CST com interseção de superfícies (fonte: autor)	185
Figura 73 – Perfil CST cujo extradorso tem mais de uma mudança de inclinação (fonte: autor)	186
Figura 74 – Perfil CST cujo extradorso tem mais de uma mudança de inclinação (fonte: autor)	186
Figura 75 – Perfil CST excessivamente fino (fonte: autor)	187
Figura 76 – Perfil NACA 2412 convertido para o CST	190

Lista de tabelas

Tabela 1 – Nomenclatura dos algoritmos deste trabalho (fonte: autor)	77
Tabela 2 – Resultados dos testes de validação com a função de De Jong	109
Tabela 3 – Resultados dos testes de validação com a função de Ackley	110
Tabela 4 – Resultados dos testes de validação com a função de Griewank	110
Tabela 5 – Configuração padrão do algoritmo NACA 4 dígitos para os testes de parâmetros	111
Tabela 6 – Resultados dos testes de tamanho populacional no algoritmo NACA 4 dígitos	112
Tabela 7 – Resultados dos testes de chance de mutação no algoritmo NACA 4 dígitos	112
Tabela 8 – Resultados dos testes de número de iterações no XFOIL no algoritmo NACA 4 dígitos	113
Tabela 9 – Resultados dos testes do mecanismo de elitismo no algoritmo NACA 4 dígitos	114
Tabela 10 – Configuração padrão do algoritmo CST para os testes de parâmetros	115
Tabela 11 – Resultados dos testes de tamanho populacional no algoritmo CST	116
Tabela 12 – Resultados dos testes de chance de mutação no algoritmo CST	116
Tabela 13 – Resultados dos testes de número de iterações no XFOIL no algoritmo CST	116
Tabela 14 – Resultados dos testes do mecanismo de elitismo no algoritmo CST	117
Tabela 15 – Resultados do caso 1 da primeira comparação (DINA et al., 2019), (fonte: autor)	119
Tabela 16 – Resultados do caso 2 da primeira comparação (DINA et al., 2019), (fonte: autor)	120
Tabela 17 – Dados aerodinâmicos dos aerofólios resultantes dos testes independentes do algoritmo NACA 4 dígitos (fonte: autor)	134
Tabela 18 – Dados aerodinâmicos dos aerofólios resultantes dos testes independentes do algoritmo CST (fonte: autor)	135
Tabela 19 – Dados aerodinâmicos das asas resultantes dos testes independentes do algoritmo NACA 4 dígitos (fonte: autor)	136
Tabela 20 – Dados aerodinâmicos das asas resultantes dos testes independentes do algoritmo CST 3D (fonte: autor)	137
Tabela 21 – Dados aerodinâmicos dos aerofólios obtidos no segundo estudo de caso (fonte: autor)	138
Tabela 22 – Dados aerodinâmicos dos aerofólios obtidos no terceiro estudo de caso (fonte: autor)	139

Tabela 23 – Dados aerodinâmicos dos aerofólios resultantes dos algoritmos CST e CST 3D específico para o quarto estudo de caso (fonte: autor)	141
Tabela 24 – Dados aerodinâmicos das asas resultantes dos algoritmos CST 3D e CST 3D específico para o quarto estudo de caso (fonte: autor)	142
Tabela 25 – Dados aerodinâmicos das asas obtidas no quinto estudo de caso (fonte: autor)	143
Tabela 26 – Configuração padrão do algoritmo CST específico para os testes de parâmetros	195
Tabela 27 – Resultados dos testes de tamanho populacional no algoritmo CST específico	195
Tabela 28 – Resultados dos testes de chance de mutação no algoritmo CST específico	195
Tabela 29 – Resultados dos testes de número de iterações no XFOIL no algoritmo CST específico	196
Tabela 30 – Resultados dos testes do mecanismo de elitismo no algoritmo CST específico	196
Tabela 31 – Configuração padrão do algoritmo NACA 4 dígitos 3D para os testes de parâmetros (versão Octave)	196
Tabela 32 – Resultados dos testes de tamanho populacional no algoritmo NACA 4 dígitos 3D	197
Tabela 33 – Resultados dos testes de chance de mutação no algoritmo NACA 4 dígitos 3D	197
Tabela 34 – Resultados dos testes do mecanismo de elitismo no algoritmo NACA 4 dígitos 3D	197
Tabela 35 – Configuração padrão do algoritmo CST 3D para os testes de parâmetros (versão Octave)	197
Tabela 36 – Resultados dos testes de tamanho populacional no algoritmo CST 3D .	197
Tabela 37 – Resultados dos testes de chance de mutação no algoritmo CST 3D . . .	198
Tabela 38 – Resultados dos testes do mecanismo de elitismo no algoritmo CST 3D	198
Tabela 39 – Configuração padrão do algoritmo CST 3D específico para os testes de parâmetros (versão Octave)	199
Tabela 40 – Resultados dos testes de tamanho populacional no algoritmo CST específico 3D	199
Tabela 41 – Resultados dos testes de chance de mutação no algoritmo CST 3D específico	199
Tabela 42 – Resultados dos testes do mecanismo de elitismo no algoritmo CST 3D específico	199
Tabela 43 – Aerofólios obtidos nos testes de tamanho populacional no algoritmo NACA 4 dígitos (fonte: autor)	201

Tabela 44 – Aerofólios obtidos nos testes de chance de mutação no algoritmo NACA 4 dígitos (fonte: autor)	201
Tabela 45 – Aerofólios obtidos nos testes de números de iterações no XFOIL no algoritmo NACA 4 dígitos (fonte: autor)	201
Tabela 46 – Aerofólios obtidos nos testes de mecanismo de elitismo no algoritmo NACA 4 dígitos (fonte: autor)	202
Tabela 47 – Parâmetros geométricos dos aerofólios obtidos nos testes de tamanho populacional no algoritmo CST (fonte: autor)	202
Tabela 48 – Parâmetros geométricos dos aerofólios obtidos nos testes de chance de mutação no algoritmo CST (fonte: autor)	202
Tabela 49 – Parâmetros geométricos dos aerofólios obtidos nos testes de números de iterações no XFOIL no algoritmo CST (fonte: autor)	202
Tabela 50 – Parâmetros geométricos dos aerofólios obtidos nos testes de mecanismo de elitismo no algoritmo CST (fonte: autor)	203
Tabela 51 – Parâmetros geométricos dos aerofólios obtidos nos testes de tamanho populacional no algoritmo CST específico (fonte: autor)	203
Tabela 52 – Parâmetros geométricos dos aerofólios obtidos nos testes de chances de mutação no algoritmo CST específico (fonte: autor)	203
Tabela 53 – Parâmetros geométricos dos aerofólios obtidos nos testes de números de iterações no XFOIL no algoritmo CST específico (fonte: autor)	203
Tabela 54 – Parâmetros geométricos dos aerofólios obtidos nos testes de números de iterações no XFOIL no algoritmo CST específico (fonte: autor)	204
Tabela 55 – Geometrias das plantas das asas obtidas nos testes de tamanho populacional no algoritmo NACA 4 dígitos 3D - parte 1 (fonte: autor)	204
Tabela 56 – Geometrias das plantas das asas obtidas nos testes de tamanho populacional no algoritmo NACA 4 dígitos 3D - parte 2 (fonte: autor)	204
Tabela 57 – Aerofólios das asas obtidas nos testes de tamanho populacional no algoritmo NACA 4 dígitos 3D (fonte: autor)	204
Tabela 58 – Geometrias das plantas das asas obtidas nos testes de chance de mutação no algoritmo NACA 4 dígitos 3D - parte 1 (fonte: autor)	204
Tabela 59 – Geometrias das plantas das asas obtidas nos testes de chance de mutação no algoritmo NACA 4 dígitos 3D - parte 2 (fonte: autor)	205
Tabela 60 – Aerofólios das asas obtidas nos testes de chance de mutação no algoritmo NACA 4 dígitos 3D (fonte: autor)	205
Tabela 61 – Geometrias das plantas das asas obtidas nos testes de mecanismo de elitismo no algoritmo NACA 4 dígitos 3D - parte 1 (fonte: autor)	205
Tabela 62 – Geometrias das plantas das asas obtidas nos testes de mecanismo de elitismo no algoritmo NACA 4 dígitos 3D - parte 2 (fonte: autor)	205

Tabela 63 – Aerofólios das asas obtidas nos testes de mecanismo de elitismo no algoritmo NACA 4 dígitos 3D - parte 2 (fonte: autor)	205
Tabela 64 – Geometrias das plantas das asas obtidas nos testes de tamanho populacional no algoritmo CST 3D - parte 1 (fonte: autor)	206
Tabela 65 – Geometrias das plantas das asas obtidas nos testes de tamanho populacional no algoritmo CST 3D - parte 2 (fonte: autor)	206
Tabela 66 – Parâmetros geométricos dos aerofólios da raiz obtidos nos testes de tamanho populacional no algoritmo CST 3D (fonte: autor)	206
Tabela 67 – Parâmetros geométricos dos aerofólios do meio obtidos nos testes de tamanho populacional no algoritmo CST 3D (fonte: autor)	206
Tabela 68 – Parâmetros geométricos dos aerofólios da ponta obtidos nos testes de tamanho populacional no algoritmo CST 3D (fonte: autor)	207
Tabela 69 – Geometrias das plantas das asas obtidas nos testes de chance de mutação no algoritmo CST 3D - parte 1 (fonte: autor)	207
Tabela 70 – Geometrias das plantas das asas obtidas nos testes de chance de mutação no algoritmo CST 3D - parte 2 (fonte: autor)	207
Tabela 71 – Parâmetros geométricos dos aerofólios da raiz obtidos nos testes de chance de mutação no algoritmo CST 3D (fonte: autor)	207
Tabela 72 – Parâmetros geométricos dos aerofólios do meio obtidos nos testes de chance de mutação no algoritmo CST 3D (fonte: autor)	208
Tabela 73 – Parâmetros geométricos dos aerofólios da ponta obtidos nos testes de chance de mutação no algoritmo CST 3D (fonte: autor)	208
Tabela 74 – Geometrias das plantas das asas obtidas nos testes de mecanismo de elitismo no algoritmo CST 3D - parte 1 (fonte: autor)	208
Tabela 75 – Geometrias das plantas das asas obtidas nos testes de mecanismo e elitismo no algoritmo CST 3D - parte 2 (fonte: autor)	208
Tabela 76 – Parâmetros geométricos dos aerofólios da raiz obtidos nos testes de mecanismo de elitismo no algoritmo CST 3D (fonte: autor)	208
Tabela 77 – Parâmetros geométricos dos aerofólios do meio obtidos nos testes de mecanismo de elitismo no algoritmo CST 3D (fonte: autor)	209
Tabela 78 – Parâmetros geométricos dos aerofólios da ponta obtidos nos testes de mecanismo de elitismo no algoritmo CST 3D (fonte: autor)	209
Tabela 79 – Geometrias das plantas das asas obtidas nos testes de tamanho populacional no algoritmo CST 3D específico - parte 1 (fonte: autor)	209
Tabela 80 – Geometrias das plantas das asas obtidas nos testes de tamanho populacional no algoritmo CST 3D específico - parte 2 (fonte: autor)	209
Tabela 81 – Parâmetros geométricos dos aerofólios da raiz obtidos nos testes de tamanho populacional no algoritmo CST 3D específico (fonte: autor)	210

Tabela 82 – Parâmetros geométricos dos aerofólios do meio obtidos nos testes de tamanho populacional no algoritmo CST 3D específico (fonte: autor)	210
Tabela 83 – Parâmetros geométricos dos aerofólios da ponta obtidos nos testes de tamanho populacional no algoritmo CST 3D específico (fonte: autor)	210
Tabela 84 – Geometrias das plantas das asas obtidas nos testes de chance de mutação no algoritmo CST 3D específico - parte 1 (fonte: autor)	210
Tabela 85 – Geometrias das plantas das asas obtidas nos testes de chance de mutação no algoritmo CST 3D específico - parte 2 (fonte: autor)	211
Tabela 86 – Parâmetros geométricos dos aerofólios da raiz obtidos nos testes de chance de mutação no algoritmo CST 3D específico (fonte: autor)	211
Tabela 87 – Parâmetros geométricos dos aerofólios do meio obtidos nos testes de chance de mutação no algoritmo CST 3D específico (fonte: autor)	211
Tabela 88 – Parâmetros geométricos dos aerofólios da ponta obtidos nos testes de chance de mutação no algoritmo CST 3D específico (fonte: autor)	211
Tabela 89 – Geometrias das plantas das asas obtidas nos testes de mecanismo de elitismo no algoritmo CST 3D específico - parte 1 (fonte: autor)	212
Tabela 90 – Geometrias das plantas das asas obtidas nos testes de mecanismo e elitismo no algoritmo CST 3D específico - parte 2 (fonte: autor)	212
Tabela 91 – Parâmetros geométricos dos aerofólios da raiz obtidos nos testes de mecanismo de elitismo no algoritmo CST 3D específico (fonte: autor)	212
Tabela 92 – Parâmetros geométricos dos aerofólios do meio obtidos nos testes de mecanismo de elitismo no algoritmo CST 3D específico (fonte: autor)	212
Tabela 93 – Parâmetros geométricos dos aerofólios da ponta obtidos nos testes de mecanismo de elitismo no algoritmo CST 3D específico (fonte: autor)	212
Tabela 94 – Parâmetros geométricos dos aerofólios obtidos no primeiro estudo de caso do algoritmo CST (fonte: autor)	213
Tabela 95 – Geometrias das plantas das asas obtidas no primeiro estudo de caso do algoritmo NACA 4 dígitos 3D - parte 1 (fonte: autor)	213
Tabela 96 – Geometrias das plantas das asas obtidas no primeiro estudo de caso do algoritmo NACA 4 dígitos 3D - parte 2 (fonte: autor)	213
Tabela 97 – Aerofólios das asas obtidas no primeiro estudo de caso do algoritmo NACA 4 dígitos 3D (fonte: autor)	214
Tabela 98 – Geometrias das plantas das asas obtidas no primeiro estudo de caso do algoritmo CST 3D - parte 1 (fonte: autor)	214
Tabela 99 – Geometrias das plantas das asas obtidas no primeiro estudo de caso do algoritmo CST 3D - parte 2 (fonte: autor)	214
Tabela 100 – Parâmetros geométricos dos aerofólios da raiz obtidos no primeiro estudo de caso do algoritmo CST 3D (fonte: autor)	214

Tabela 101–Parâmetros geométricos dos aerofólios do meio obtidos no primeiro estudo de caso do algoritmo CST 3D (fonte: autor)	215
Tabela 102–Parâmetros geométricos dos aerofólios da ponta obtidos no primeiro estudo de caso do algoritmo CST 3D (fonte: autor)	215
Tabela 103–Parâmetros geométricos dos aerofólios CST obtidos no segundo estudo de caso (fonte: autor)	215
Tabela 104–Parâmetros geométricos dos aerofólios CST obtidos no terceiro estudo de caso (fonte: autor)	216
Tabela 105–Geometrias das plantas das asas obtidas no quarto estudo de caso - parte 1 (fonte: autor)	216
Tabela 106–Geometrias das plantas das asas obtidas no quarto estudo de caso - parte 2 (fonte: autor)	216
Tabela 107–Parâmetros geométricos dos aerofólios da raiz obtidos no quarto estudo de caso(fonte: autor)	216
Tabela 108–Parâmetros geométricos dos aerofólios do meio obtidos no quarto estudo de caso (fonte: autor)	217
Tabela 109–Parâmetros geométricos dos aerofólios da ponta obtidos no quarto estudo de caso (fonte: autor)	217
Tabela 110–Geometrias das plantas das asas obtidas no quinto estudo de caso - parte 1 (fonte: autor)	217
Tabela 111–Geometrias das plantas das asas obtidas no quinto estudo de caso - parte 2 (fonte: autor)	217
Tabela 112–Aerofólios das asas obtidas no quinto estudo de caso (fonte: autor) . .	217

Lista de abreviaturas e siglas

AoA	<i>Angle of Attack</i> (Ângulo de Ataque)
API	<i>Application Programming Interface</i> (Interface de Programação de Aplicações)
CFD	<i>Computational Fluid Dynamics</i> (Dinâmica dos Fluidos Computacional)
CST	<i>Class and Shape Function Transformation</i> (Transformação de funções classe e formato)
NACA	<i>National Advisory Committee for Aeronautics</i> (Comitê Nacional para Aconselhamento sobre Aeronáutica)
NASA	<i>National Aeronautics and Space Administration</i> (Administração Nacional da Aeronáutica e Espaço)
NLF	<i>Natural Laminar Flow</i> (Escoamento laminar natural)
VLM	<i>Vortex Lattice Method</i> (Método da Treliça de Vórtices)

Lista de símbolos

L	Força de sustentação
q_∞	Pressão dinâmica do vento relativo
S	Área alar
C_L	Coefficiente de sustentação
ρ_∞	Densidade do fluido do vento relativo
V_∞	Velocidade do vento relativo
D	Força de arrasto
C_D	Coefficiente de arrasto
L/D	Razão sustentação-arrasto
M	Momento de arfagem
l	Comprimento de referência
C_M	Coefficiente de momento de arfagem
ψ	Função de corrente
u	Velocidade na direção x
v	Velocidade na direção y
ξ	Vetor vorticidade
\mathbf{V}	Vetor velocidade
ϕ	Potencial de velocidade
Γ	Circulação, intensidade do escoamento em vórtice
L'	Força de sustentação por unidade de envergadura
Λ	Intensidade de uma fonte ou sumidouro
$\lambda(s)$	Intensidade de uma linha de fonte por unidade de comprimento
$\gamma(s)$	Intensidade de uma linha de vórtice por unidade de comprimento, abs- cissa normalizada no CST

Re	Número de Reynolds
μ_∞	Viscosidade dinâmica do fluido do escoamento relativo
α	Ângulo de ataque
m	Curvatura máxima do aerofólio NACA 4 dígitos expresso em uma porcentagem da corda
p	Local da curvatura máxima do aerofólio NACA 4 dígitos, a partir do bordo de ataque, expresso em décimos de corda
t	Espessura máxima do aerofólio NACA 4 dígitos, expresso em uma porcentagem da corda
c	Comprimento de corda
$C(x/c)$	Função <i>class</i>
$S(x/c)$	Função <i>shape</i>
ΔZ_{TE}	Espessura do bordo de fuga
A_i	Pesos da função <i>shape</i>
BP_n	Polinômio de Bernstein
β	Ângulo do bordo de fuga
ζ	Ordenada normalizada no CST
$\Delta \xi$	Espessura de bordo de fuga normalizada
AR	Razão de aspecto
b	Envergadura
\in	Pertence
\mathbb{N}	Conjunto dos números naturais
\mathbb{Z}	Conjunto dos números inteiros
\mathbb{R}	Conjunto dos números reais
o	Configuração dos algoritmos: buscar um valor específico de um coeficiente aerodinâmico
$\hat{}$	Configuração dos algoritmos: maximizar ou minimizar um coeficiente aerodinâmico

$!$	Configuração dos algoritmos: não considerar um coeficiente aerodinâmico
c	Configuração dos algoritmos: buscar coeficiente de momento igual em múltiplas condições de voo (valor médio)
k	Configuração dos algoritmos: buscar coeficiente de momento igual em múltiplas condições de voo (valor especificado)
q	Configuração dos algoritmos: buscar força aerodinâmica específica ou momento de arfagem específico
$\#$	Configuração dos algoritmos: maximizar ou minimizar uma força aerodinâmica
C_i	Valor de um coeficiente aerodinâmico arbitrário
C_{tgt}	Valor alvo de um coeficiente aerodinâmico arbitrário
f_*	Mínimo global das funções de validação
\mathbf{x}_*	Genótipo que corresponde ao mínimo global nas funções de validação

Sumário

1	INTRODUÇÃO	35
1.1	Contextualização	35
1.2	Motivação	39
1.3	Objetivos	40
1.3.1	Objetivo geral	40
1.3.2	Objetivos específicos	40
1.4	Metodologia	41
1.5	Estruturação do trabalho	41
2	REVISÃO BIBLIOGRÁFICA	43
2.1	Conceitos básicos de algoritmos de otimização	43
2.2	Algoritmos genéticos	44
2.3	Aerodinâmica	47
2.3.1	Forças e momentos	48
2.3.2	Método de painéis	48
2.3.2.1	Painéis de fonte	49
2.3.2.2	Painéis de vórtice	53
2.3.3	Método de painéis 3D	54
2.3.4	Método da treliça de vórtices (VLM)	55
2.4	Aerofólios	57
2.4.1	NACA 4 dígitos	60
2.4.2	CST	61
2.5	XFOIL	65
2.6	Asas	67
2.7	APAME	71
2.8	VSPAERO	73
3	METODOLOGIA	75
3.1	Estrutura do algoritmo NACA 4 dígitos	77
3.1.1	Representação dos aerofólios dentro dos algoritmos	77
3.1.2	Geração da população inicial	78
3.1.3	Avaliação	79
3.1.4	Atribuição de aptidão	81
3.1.4.1	Alcançar um valor específico (opção 'o')	82
3.1.4.2	Alcançar um valor máximo/mínimo (opção '^')	83
3.1.4.3	Alcançar um mesmo valor em múltiplas condições de voo (opções 'c' e 'k')	83

3.1.5	Seleção para reprodução	84
3.1.6	<i>Crossover</i>	85
3.1.7	Mutação	86
3.1.7.1	Mutação de perfis simétricos ($m = 0$ e $p = 0$)	86
3.1.7.2	Mutação de perfis assimétricos ($m \neq 0$ e $p \neq 0$)	86
3.1.8	Aplicação de elitismo	87
3.2	Estrutura do algoritmo CST	87
3.2.1	Representação dos aerofólios dentro do algoritmo	87
3.2.2	Geração da população inicial	89
3.2.3	Avaliação	90
3.2.4	Atribuição de aptidão e seleção para reprodução	91
3.2.5	<i>Crossover</i>	91
3.2.5.1	Ambos os perfis são assimétricos	91
3.2.5.2	Ambos os perfis são simétricos	93
3.2.5.3	Um perfil é simétrico e o outro é assimétrico	93
3.2.6	Mutação	93
3.3	Estrutura do algoritmo CST específico	94
3.3.1	Representação dos aerofólios dentro do algoritmo	94
3.3.2	Geração da população inicial	94
3.3.3	Avaliação, atribuição de aptidão, seleção para reprodução, <i>crossover</i> e mu- tação	95
3.4	Estrutura do algoritmo NACA 4 dígitos 3D	95
3.4.1	Representação das asas dentro do algoritmo	95
3.4.2	Geração da população inicial	99
3.4.3	Avaliação	100
3.4.4	Atribuição de aptidão e seleção para reprodução	101
3.4.5	<i>Crossover</i>	102
3.4.5.1	<i>Crossovers</i> gerais das características	102
3.4.5.2	Trocar os tipos de asas	103
3.4.6	Mutação	104
3.5	Estrutura do algoritmo CST 3D	104
3.5.1	Representação das asas dentro do algoritmo	104
3.5.2	Geração da população inicial	106
3.5.3	Avaliação, atribuição de aptidão e seleção para reprodução	107
3.5.4	<i>Crossover</i>	107
3.5.5	Mutação	107
3.6	Estrutura do algoritmo CST 3D específico	107
3.6.1	Representação das asas dentro do algoritmo	107
3.6.2	Geração da população inicial	107

3.6.3	Avaliação, atribuição de aptidão, seleção para reprodução, <i>crossover</i> e mutação	108
3.7	Validações do algoritmo genético	108
3.7.1	Função de De Jong	109
3.7.2	Função de Ackley	110
3.7.3	Função de Griewank	110
3.8	Testes de parâmetros	111
3.8.1	Algoritmo NACA 4 dígitos	111
3.8.2	Algoritmo CST	115
3.9	Comparações com bibliografia	118
3.9.1	Aerofólios	119
3.9.2	Asas	120
3.10	Estudos de caso	122
3.10.1	Primeiro estudo de caso: testes independentes	122
3.10.2	Segundo estudo de caso: detalhando aerofólios	126
3.10.3	Terceiro estudo de caso: coeficientes de momento constantes	127
3.10.4	Quarto estudo de caso: testes integrados	128
3.10.5	Quinto estudo de caso: comparação com o VLM	131
4	RESULTADOS E DISCUSSÕES	133
4.1	Primeiro estudo de caso: testes independentes	133
4.1.1	Algoritmo NACA 4 dígitos	133
4.1.2	Algoritmo CST	134
4.1.3	Algoritmo NACA 4 dígitos 3D	135
4.1.4	Algoritmo CST 3D	136
4.2	Segundo estudo de caso: detalhando aerofólios	138
4.3	Terceiro estudo de caso: coeficientes de momento constantes	139
4.4	Quarto estudo de caso: testes integrados	141
4.5	Quinto estudo de caso: comparação com o VLM	143
5	CONCLUSÕES	145
	REFERÊNCIAS	149
	APÊNDICES	155
	APÊNDICE A – ANÁLISE DIMENSIONAL	157
	APÊNDICE B – FUNÇÃO DE CORRENTE	159

	APÊNDICE C – POTENCIAL DE VELOCIDADE	161
	APÊNDICE D – EQUAÇÃO DE LAPLACE	163
	APÊNDICE E – CIRCULAÇÃO	165
	APÊNDICE F – ESCOAMENTOS FUNDAMENTAIS	167
	APÊNDICE G – CONDIÇÃO DE KUTTA	171
	APÊNDICE H – VISCOSIDADE, CAMADA LIMITE E O NÚMERO DE REYNOLDS	173
	APÊNDICE I – MODELOS DOS ARQUIVOS DE ENTRADA PARA O XFOIL	177
I.1	Algoritmo NACA 4 dígitos	177
I.2	Algoritmo CST e CST específico	178
	APÊNDICE J – MODELOS DOS ARQUIVOS DE ENTRADA PARA O APAME	179
	APÊNDICE K – FORMATAÇÃO DAS COORDENADAS NO AL- GORITMO CST	183
	APÊNDICE L – CHECAGENS DE QUALIDADE NO ALGORITMO CST (2D E 3D)	185
	APÊNDICE M – CST REVERSO	189
M.1	Sistema linear simples	189
M.2	Método dos mínimos quadrados	190
	APÊNDICE N – INTERPOLAÇÃO DE AEROFÓLIOS	193
	APÊNDICE O – TESTES DE PARÂMETROS RESTANTES	195
O.1	Algoritmo CST específico	195
O.2	Algoritmo NACA 4 dígitos 3D	196
O.3	Algoritmo CST 3D	197
O.4	Algoritmo CST 3D específico	198
	APÊNDICE P – AEROFÓLIOS E ASAS RESULTANTES DOS TES- TES DE PARÂMETROS	201
P.1	Algoritmo NACA 4 dígitos	201
P.2	Algoritmo CST	202
P.3	Algoritmo CST específico	203

P.4	Algoritmo NACA 4 dígitos 3D	204
P.5	Algoritmo CST 3D	206
P.6	Algoritmo CST 3D específico	209
APÊNDICE Q – AEROFÓLIOS E ASAS RESULTANTES DOS ES-		
TUDOS DE CASO		213
Q.1	Primeiro estudo de caso: testes independentes	213
Q.1.1	Algoritmo CST	213
Q.1.2	Algoritmo NACA 4 dígitos 3D	213
Q.1.3	Algoritmo CST 3D	214
Q.2	Segundo estudo de caso: detalhando aerofólios	215
Q.3	Terceiro estudo de caso: coeficientes de momento constantes	216
Q.4	Quarto estudo de caso: testes integrados	216
Q.5	Quinto estudo de caso: comparação com o VLM	217

1 Introdução

Este capítulo de introdução tem o papel de apresentar os desenvolvimentos deste trabalho de conclusão de curso. Primeiro, para prover contexto, é feita uma discussão a respeito dos elementos principais do trabalho e de pesquisas anteriores. A seguir, são apresentados os objetivos e por último expõe-se a organização do documento.

1.1 Contextualização

A engenharia é uma das ciências mais vitais da humanidade. Com o intuito de melhor compreender e controlar o ambiente ao redor, e com o desejo de simplificar as tarefas, fossem elas banais ou graves, a humanidade desenvolveu suas ferramentas para alcançar esses objetivos. Aliada à matemática, a engenharia pôde apenas evoluir à medida que foram sendo desenvolvidas novas tecnologias e, por sua vez, novas áreas do conhecimento. A engenharia possibilitou a observação detalhada dos céus, permitiu a locomoção por terra, água e ar, e proporcionou uma rede de comunicação global que há apenas cinquenta anos atrás teria sido considerada como uma utopia exagerada exceto pelos menos céticos. No entanto, deixando de lado a multidisciplinaridade da engenharia por um momento e resumindo seus conceitos, pode-se descrever seu propósito primordial como, simplesmente, a solução de problemas - afinal, tudo o que há de artificial (isto é, que não foi fornecido pronto pela natureza) surgiu de alguma necessidade.

Quando se trata de solução de problemas, há poucas razões para não desejar a melhor solução possível. O processo que busca esse objetivo é nomeado como otimização e, apesar de a otimização ser bastante associada a algoritmos de computador, o conceito de problema de otimização já existia antes da invenção do computador digital, assim como engenheiros já existiam mesmo antes que a palavra que os nomeia. Por exemplo, matemáticos gregos resolveram problemas de otimização envolvendo geometria Euclidiana, e, em seu *Principia Mathematica*, Isaac Newton solucionou um problema que consistia em determinar o corpo simétrico de revolução que geraria o menor arrasto possível dentro de um escoamento de fluido (YANG, 2010). Focando nesse aspecto do objetivo de minimizar o arrasto, é visível que esse problema é, fundamentalmente, não muito distante do que se realiza em projetos aeronáuticos contemporâneos; com a diferença, claro, dos detalhes do problema em mãos e dos métodos de solução empregados.

Assim, fica claro que uma das formas de se resolver um problema de otimização é por meios analíticos e, na verdade, todo estudante, independentemente do nível, já solucionou ao menos um problema de otimização usando esse método. Como exemplo, a Fig. 1 expõe o problema da caixa de papelão, uma questão clássica nos cursos de cálculo.

A proposta é, dado um pedaço quadrado de papelão, determinar o lado x dos quadrados cortados nos cantos de modo que maximize o volume da caixa sem tampa que será formada depois que o papelão for dobrado.

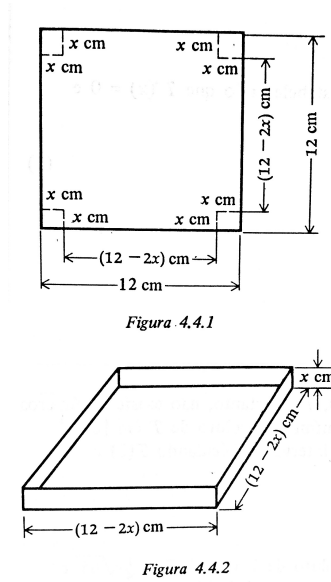


Figura 1 – Problema da caixa de papelão como visto em (LEITHOLD, 1986)

No entanto, métodos analíticos são aplicáveis apenas até certo nível de complexidade do problema em mãos. A natureza não se acomoda sempre dentro dos modelos analíticos desenvolvidos nas diversas disciplinas da engenharia. Sendo assim, cabe aos métodos numéricos e aos algoritmos computacionais auxiliar nessa tarefa. Algoritmos de otimização são processos iterativos que trabalham visando minimizar ou maximizar uma dada função objetivo selecionando e manipulando sistematicamente os parâmetros de projeto. O domínio contendo todas as soluções possíveis para uma dada configuração de entrada é definido como espaço de busca e o intuito é encontrar dentro dele o ótimo global, que é a melhor solução possível para o problema. Há uma quantidade de algoritmos de otimização distintos, cada qual com suas respectivas vantagens e fraquezas, o que torna cada tipo melhor adaptado a diferentes classes de problema. Algoritmos baseados em gradientes, que se enquadram dentro da categoria dos algoritmos determinísticos, são alguns dos mais populares. São classificados como determinísticos porque cada passo em direção à melhor solução é calculado de forma exata. Como conceito matemático, o gradiente contribui diretamente com esse comportamento por indicar a direção e sentido nos quais uma determinada função terá seu maior incremento (ou menor, caso seja usado o negativo do gradiente). Portanto, métodos baseados em gradientes convergem rápido às suas soluções. No entanto, algoritmos baseados em gradientes podem ter a tendência de ficarem presos em ótimos locais - isto é, resultados que são os melhores em comparação aos resultados diretamente próximos, mas que são distintos do ótimo global. Há também a necessidade do emprego de funções contínuas para que o método funcione, e sua impraticabilidade aumenta à medida que cresce o número de variáveis de *design*. Apesar desses

problemas, algoritmos baseados em gradientes ainda têm muitas aplicações reais devido ao seu comportamento rápido e direto, o qual é aperfeiçoado por adaptações que mitigam suas falhas. Por exemplo, (SOEMARWOTO, 1998) realiza otimizações de aerofólios por meio de *design* inverso usando gradientes auxiliados pelo método variacional, o qual facilita o processo de otimização quando há muitas variáveis de *design* (SOEMARWOTO, 1997). Também já existem métodos sendo desenvolvidos para mapear dados discretos de modo a convertê-los em funções contínuas para torná-los compatíveis com os gradientes (GÓMEZ-BOMBARELLI et al., 2018).

Existem conceitos fora da matemática que inspiraram a criação de algoritmos de otimização. Os chamados algoritmos evolucionários são baseados na evolução natural das espécies. Um exemplo de uso de algoritmos evolucionários é a antena de banda X utilizada na missão ST5 da NASA. A antena é ilustrada na Fig. 2, e seu formato peculiar se deve aos múltiplos requisitos de missão considerados. Em comparação a métodos tradicionais de *design*, a antena evoluída tem vantagens nos quesitos de consumo de energia, tempo de fabricação, complexidade e desempenho (HORNBY et al., 2006).

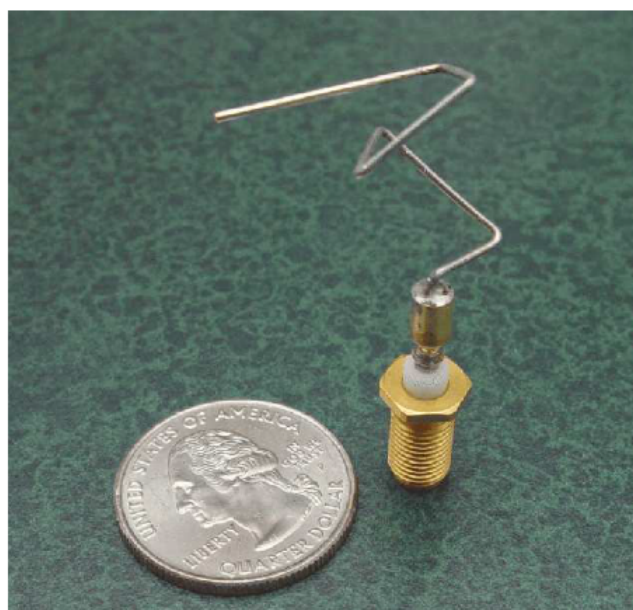


Figura 2 – Antena utilizada na missão ST5 da NASA, como visto em (HORNBY et al., 2006)

Entre os algoritmos evolucionários, os algoritmos genéticos, escolhidos como ferramenta de otimização deste trabalho, estão entre os mais populares. O fundamento dos algoritmos genéticos é o uso das variáveis de *design* como uma analogia aos cromossomos dos indivíduos de uma população, sendo estes cromossomos manipulados de acordo com a aptidão dos indivíduos. Por não depender da continuidade das funções objetivo, o algoritmo genético tem implementação simples em diversas aplicações. Por exemplo, (GROENWOLD; STANDER; SNYMAN, 1999) faz a otimização estrutural de treliças com o intuito de minimizar seus pesos; (JUN, 2009) aplica a otimização ao problema de

roteamento de veículos; (JIN; CUI; FANG, 1997) faz a busca de um modelo aprimorado da representação da dispersão eletromagnética para aperfeiçoar a eficiência de radares; (PONTEROSSO; FOX, 2000) otimiza o uso de *geogrids* para a estabilização de taludes. Esses quatro estudos de otimização foram todos realizados usando algoritmos genéticos e, com uma busca rápida, pode-se encontrar muitos outros exemplos, variando da engenharia eletrônica (CHENG; HE, 1999) à bioinformática (SZUSTAKOWSKI; WENG, 2000). Claramente também a engenharia aeronáutica se beneficia enormemente dessa técnica. Em (RAM et al., 2014) é feita a otimização de um perfil NACA 0012 em uma condição de voo com ângulo de ataque de 3° , que resulta em um ligeiro aumento no arrasto, mas também em um aumento de cerca de 10% no coeficiente de sustentação. Em (FLYNN; SHERMAN, 1995) é feita a otimização de painéis estruturais da fuselagem de uma aeronave, objetivando reduzir o peso, mas garantindo a resistência dos componentes a flambagem. Em (YANG; YANG, 2012) apresenta-se, como problema de otimização, o desenvolvimento de um plano de manutenção de aeronaves, almejando coordenar todas as tarefas de manutenção necessárias garantindo a melhor eficiência em termos de tempo e recursos financeiros.

A solução de problemas de otimização é de grande importância na engenharia. Isso se dá pelo fato de que todo projeto de engenharia tem seus *trade-offs* referentes. É preciso compreender e coordenar as relações de compromisso entre os parâmetros envolvidos e a complexidade desse processo é fundamentado não apenas na quantidade de variáveis, mas também no nível de severidade dos requisitos que devem ser cumpridos para que o projeto seja funcional e também seguro. Por exemplo, (FATHALLAH; HELAL, 2016) apresenta a situação de um projeto de submarino. Mais especificamente, o casco, que deve ser forte o suficiente para suportar as altas pressões da água nas profundezas, mas que também deve comportar todos os sistemas do submarino com o máximo de carga útil possível, deve permitir bom desempenho de locomoção do submarino e deve também, evidentemente, ser o mais leve o possível. Projetos aeronáuticos não são submetidos a condições tão implacáveis como essa, que resultam em fatores de segurança de valores como 1,5 ou 2, mas o potencial de complexidade dos *trade-offs* não deve ser subestimado.

Um dos melhores exemplos disso na engenharia aeronáutica são os aerofólios. Aerofólios, também chamados de perfis, são definidos como a seção transversal de asas ou outras superfícies aerodinâmicas como empenagens e *canards*, e têm geometrias desenhadas de modo que, quando movendo-se através do ar, gerem forças de sustentação, procurando também minimizar o arrasto. É necessário ter ciência dessas propriedades do aerofólio ao longo de toda a extensão de ângulos de ataque nos quais a aeronave voará, incluindo as características de momento de arfagem. Ainda no âmbito da aerodinâmica, é preciso também conhecer o comportamento de estol do aerofólio e, caso o projeto requira, por exemplo, deve-se averiguar as características de transição de fluxo laminar para turbulento, o número de Mach crítico, entre outras informações possivelmente relevantes. Fora

isso, aerofólios devem ser escolhidos também em função de dados não aerodinâmicos. As seções de asa devem ter espaço o suficiente para estruturas, mecanismos de trens de pouso e de superfícies de controle, tanques de combustível, entre outros. Portanto, nota-se como apenas um (supostamente) pequeno ramo de um projeto aeronáutico pode se expandir em um complexo problema de otimização, o qual se aprofunda ainda mais prosseguindo dos aerofólios, o caso bidimensional, para as asas, o caso tridimensional. Asas englobam tudo o que foi mencionado a respeito dos aerofólios com o adicional de efeitos exclusivos ao espaço tridimensional - um exemplo notável são os vórtices de ponta de asa, relacionados ao arrasto induzido, causados pela movimentação de ar na direção da envergadura da asa.

Independentemente do tipo de algoritmo de otimização empregado, é essencial que haja alguma forma de avaliar os objetos de estudo. Como o intuito é otimizar aerofólios e asas em função de suas características aerodinâmicas, deve-se implementar um método para obter tais características. Métodos numéricos computacionais são a ferramenta adequada, e, dentre eles, o método de painéis será o escolhido. O método de painéis soluciona numericamente o fluxo de ar ao redor de um corpo discretizando sua superfície em um número finito de painéis. Apesar de não ser tão preciso quanto o CFD, o método de painéis fornece resultados suficientemente realistas para estágios de *design* preliminar e, focando em *software* já bem estabelecido no cenário da engenharia, seus códigos são robustos e rápidos, características essenciais para qualquer processo iterativo. O *software* usado neste trabalho para os algoritmos de aerofólios é o XFOIL, possivelmente o mais popular dos códigos de métodos de painéis disponíveis. Graças à sua versatilidade e seus aprimoramentos, como correções de efeitos de compressibilidade e de viscosidade, o XFOIL segue sendo popular mesmo após quase quarenta anos desde o seu lançamento. Outros exemplos de *software* de métodos de painéis para o caso bidimensional são o *Eppler Airfoil Code (PROFILE)* e o XFLR5 que, inclusive, é baseado no XFOIL. Para os casos tridimensionais, há novamente o XFLR5, e outros programas como o Panukl, o VSPAERO e o APAME - sendo este último a escolha primária para os algoritmos de asas.

Com isso, o intuito deste trabalho é desenvolver um algoritmo genético para a otimização de aerofólios e asas, cujo escopo conterà apenas os requisitos aerodinâmicos. Os algoritmos serão escritos em Octave/MATLAB e em Python.

1.2 Motivação

Como mencionado anteriormente, algoritmos de otimização são enormemente úteis na engenharia devido aos problemas de *trade-offs*. Com a devida configuração, um algoritmo assim pode se tornar uma poderosa ferramenta em estágios de *design* preliminar. Em verdade, tamanha é a popularidade dessa metodologia que já existem outros trabalhos feitos dessa forma, como (DINA et al., 2019), e códigos de MATLAB de otimização de

aerofólios estão disponíveis na internet (SAAD, 2018). No entanto, os autores de artigos nem sempre disponibilizam seus códigos, e aqueles que são acessíveis, como o referenciado acima, podem não ter uma implementação tão clara ao público geral, e podem não ser gerais o suficiente em suas funções objetivo para nuances distintas de problemas de otimização. Portanto, a ferramenta proposta neste trabalho servirá não apenas para provar a aplicabilidade da metodologia, como também para uso de projetistas em contextos reais. O emprego exclusivo de *software open-source* enfatiza essa filosofia, assim como a disponibilidade dos códigos em duas linguagens diferentes - Octave/MATLAB e Python.

Ponderando, por exemplo, no contexto da Faculdade UnB Gama, existem equipes de engenharia aeronáutica, a Dragonfly e a Mamutes do Cerrado, que poderiam ser beneficiadas por essa ferramenta de otimização. Em verdade, este trabalho surgiu especificamente dentro do contexto da Dragonfly, para uso no projeto do planador autolanzável. Adicionalmente, essa metodologia não é limitada apenas à engenharia aeronáutica. Rotores de turbinas eólicas são, fundamentalmente, asas com seus respectivos aerofólios projetadas para uma aplicação diferente daquela de aeronaves. Pode-se pensar o mesmo para aletas de foguetes e também nas asas de veículos como os *Space Shuttles*. Aerofólios de automóveis de corrida compartilham de muitos dos objetivos aerodinâmicos das asas de aviões, e os hidrofólios de veículos marítimos são projetados, obviamente, para garantirem o controle necessário do veículo ao mesmo tempo que se procura minimizar a força de resistência da água. Desse modo, novamente pensando no contexto da FGA, temos que os algoritmos de otimização de aerofólios e asas também poderiam ser benéficos a equipes como a Capital Rocket Team, a Fórmula Gama Racing e a Kraken.

1.3 Objetivos

1.3.1 Objetivo geral

Desenvolver uma metodologia de otimização confiável e adaptável, baseada em algoritmos genéticos e métodos de painéis, para a otimização de aerofólios e asas de uso (primariamente) aeronáutico.

1.3.2 Objetivos específicos

- Estudar Octave e Python;
- Estudar algoritmos genéticos;
- Estudar método de painéis para aplicações bidimensionais e tridimensionais;
- Aprimorar os algoritmos de otimização escritos anteriormente, melhorando a eficiência e adicionando novas funções objetivo;

- Aperfeiçoar a implementação da parametrização dos perfis CST (*Class and Shape Function Transformation*);
- Implementar o problema tridimensional (asas) ao algoritmo.

Estes últimos três itens explicam-se devido ao fato de que o presente trabalho é uma expansão de um PIBIC tratando da otimização de aerofólios, usando uma metodologia igual à empregada aqui. A publicação resultante desse PIBIC é (FERNANDES; JÚNIOR, 2021).

1.4 Metodologia

A metodologia utilizada é construída de forma modular, de modo que o algoritmo de otimização, a plataforma de código na qual o algoritmo é programado e o simulador que provém dados do objeto de estudo podem todos ser alterados de acordo com o contexto. Dentro deste trabalho, os algoritmos genéticos podem ter suas funções objetivo configuradas com quatro coeficientes aerodinâmicos distintos. Os algoritmos de aerofólios utilizam dois tipos de parametrização geométrica: a série NACA 4 dígitos e o CST, ambas das quais também são utilizadas para definir os perfis dos indivíduos nos algoritmos de asas. Para demonstrar exemplos de aplicações de cada um dos algoritmos deste trabalho, são definidos cinco estudos de caso distintos, envolvendo diferentes condições de voo e diferentes funções objetivo.

1.5 Estruturação do trabalho

Este trabalho inicia com uma revisão bibliográfica acerca dos conhecimentos necessários. Será percorrido sobre algoritmos genéticos, incluindo seu mecanismo fundamental, formas de adaptá-lo a determinados tipos de problema e aprimoramentos de desempenho; e será percorrido sobre aerodinâmica - todos os aspectos relevantes a aerofólios e asas, abrangendo a teoria dos métodos de painéis.

Em seguida, é explicada a metodologia da pesquisa, definindo como exatamente os algoritmos são configurados tendo em vista as parametrizações geométricas, as funções objetivo e o *software* de simulação aerodinâmica empregado. São realizados testes de validação do algoritmo, para certificar-se que seu desempenho é adequado, são realizados testes de parâmetros dos algoritmos (como tamanho populacional e chance de mutação), e são realizados testes de replicação para realizar comparações com resultados de trabalhos anteriores. São especificados estudos de caso e funções objetivo de modo a testar os algoritmos como se fossem empregados em alguma situação de projeto.

O capítulo seguinte apresenta os resultados dos estudos de caso, sendo realizada uma breve discussão sobre como as parametrizações geométricas, as condições de voo e as funções objetivo afetam as geometrias obtidas. Por fim, os resultados e lições aprendidas são postos em perspectiva em relação aos objetivos iniciais e aos futuros desenvolvimentos possíveis.

2 Revisão Bibliográfica

Este capítulo tem o propósito de apresentar os conceitos relevantes ao trabalho em mãos. Será discorrido a respeito dos fundamentos de algoritmos genéticos e dos conceitos de aerodinâmica relevantes aos estudos aqui executados.

2.1 Conceitos básicos de algoritmos de otimização

O propósito da otimização, independentemente da aplicação, resume-se a aprimorar um dado objeto de estudo - geralmente por meio da maximização ou da minimização de alguma função objetivo. A função objetivo, por sua vez, é uma representação matemática das características a serem manipuladas pelos algoritmos. Por meio da função objetivo, o algoritmo explora o espaço de busca, que é uma representação abstrata de todas as soluções possíveis que o algoritmo pode oferecer (limitada apenas pelas respectivas restrições). O propósito dos algoritmos é explorar o espaço de busca para alcançar o ótimo global, que é a melhor solução possível para o problema.

Problemas de otimização se apresentam de muitas formas distintas, e portanto é essencial que se tenha ciência das características desses problemas de modo a empregar os algoritmos mais apropriados para cada um.

A definição dos problemas pode ser feita de modo que haja apenas um objetivo a ser alcançado, ou múltiplos. Tendo em vista que problemas reais de engenharia são multidisciplinares por natureza, requisitos conflitantes sempre são encontrados. Um exemplo simples dentro do contexto de aerofólios e asas é a maximização da razão L/D , que geralmente induz também ao aumento do módulo do coeficiente de momento, o que é prejudicial para a estabilidade da aeronave. Portanto, um projeto sensato realizaria a maximização do L/D ao mesmo tempo garantindo que o módulo do coeficiente de momento não aumente demasiadamente.

A definição de restrições nos algoritmos também deve ser considerada. Essas restrições buscam satisfazer requisitos do problema geralmente de forma a garantir que as variáveis de *design* de um dado problema permaneçam dentro de faixas aceitáveis definidas pelo usuário, e/ou também para garantir determinadas características tendo em vista possíveis combinações das variáveis de *design*. Por exemplo, para o primeiro caso, todas as variáveis de *design* deste trabalho contam com extensões de valores possíveis que cada variável pode tomar, e para o segundo caso, um exemplo é nos algoritmos de asas, que não permitem a ocorrência de cordas de ponta com comprimento maior que cordas de raiz.

Existem outros aspectos técnicos sobre a natureza de algoritmos de otimização que são úteis para suas formulações, os quais podem ser checados em bibliografia como (RAO, 2009) e (YANG, 2010).

2.2 Algoritmos genéticos

Determinados tipos de problema não são possíveis de solucionar por meios determinísticos. Por outro lado, adotar uma abordagem de força bruta também não é prático devido aos enormes custos computacionais e de tempo. Entre esses extremos se encontram os algoritmos meta-heurísticos, cujas duas características principais são a aleatoriedade e a seleção das melhores soluções (YANG, 2010). A aleatoriedade evita que o algoritmo dependa demasiadamente de seus pontos de partida e também que se prenda demais em ótimos locais, e a seleção das melhores soluções garante que o algoritmo convirja de fato para os resultados desejados ao invés de vagar arbitrariamente pelo espaço de busca. Como exemplo de algoritmo meta-heurístico estão os algoritmos genéticos.

Grande parte dos algoritmos meta-heurísticos são baseados na natureza. Algoritmos genéticos são fundamentados na teoria da evolução das espécies de Darwin - a sobrevivência dos mais aptos. Sendo assim, um algoritmo genético realiza o processo de otimização fazendo uma analogia ao transcurso da seleção natural, incluindo representações de mecanismos biológicos como reprodução, *crossover* e mutação (RAO, 2009).

O funcionamento básico de um algoritmo genético é apresentado abaixo e resumido na Fig. 3. A estrutura dos códigos deste trabalho foi baseada em (YARPIZ, 2020).

1. Geração da população inicial.
2. Avaliação de aptidão dos indivíduos.
3. Reprodução e *crossover*.
4. Mutação.

O algoritmo inicia com a geração da população inicial, constituída por N indivíduos. Cada indivíduo é codificado com parâmetros referentes ao problema em mãos, geralmente na forma de um vetor ou de uma *string*. Esse conjunto de parâmetros, denominado genótipo, costuma ser gerado de forma aleatória dentro da extensão de valores que forma o domínio do problema. Desse modo, todo indivíduo se torna uma solução possível para o problema de otimização, e o intuito do algoritmo genético é encontrar os indivíduos que melhor satisfazem a função objetivo. É importante enfatizar que o algoritmo genético não é exato, o que significa que sua tendência é convergir para um conjunto de indivíduos distintos com alta aptidão ao invés de uma única solução perfeita. Neste trabalho,

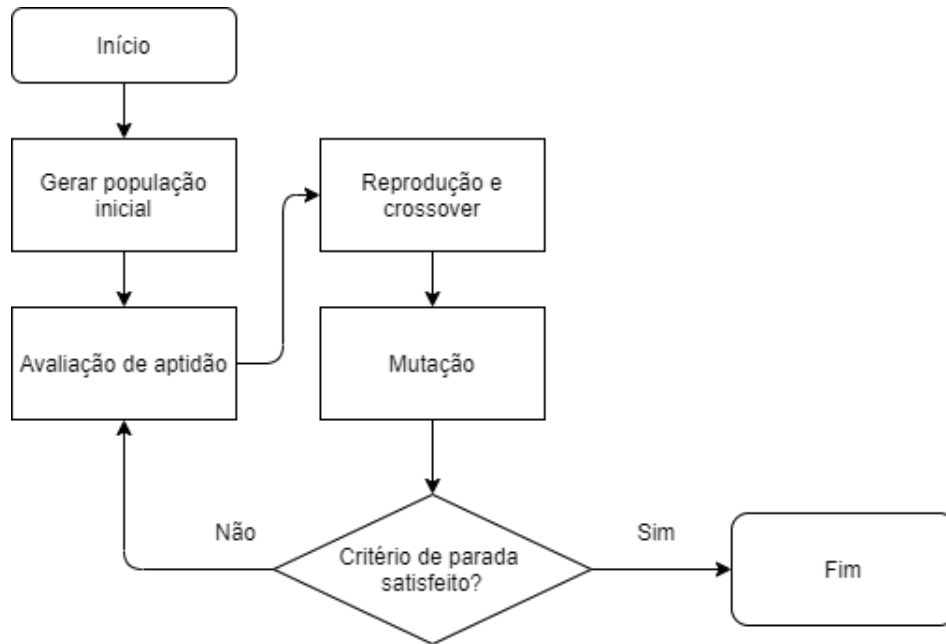


Figura 3 – Esquemático do funcionamento básico de um algoritmo genético (fonte: autor)

os indivíduos das populações dos algoritmos são aerofólios e asas, cujos genótipos são os parâmetros que definem suas geometrias.

A próxima etapa consiste na avaliação de cada um dos indivíduos em relação à função objetivo, o que marca o início do *loop* principal. A avaliação de aptidão deve representar o objetivo da otimização de modo a garantir que, nos passos seguintes, o algoritmo não tenda a fornecer soluções indesejáveis.

A seguir é feita a reprodução, a qual estabelece uma nova população de indivíduos a ser analisada na iteração seguinte. Baseando-se na inspiração biológica do algoritmo, a população original é tida como os pais e a população nova é denominada como os filhos. O primeiro estágio da reprodução é a seleção dos indivíduos, um processo tão importante quanto o *crossover* em si, pois determina quais genes serão passados para as gerações seguintes. Uma forma de se realizar a seleção é de forma completamente aleatória. Outro método comum, a seleção de roleta, é o escolhido para este trabalho. Seu funcionamento ainda é aleatório, mas com a diferença de que a probabilidade de cada indivíduo de ser selecionado para a reprodução é diretamente proporcional à sua aptidão. Isso garante que o algoritmo dirija-se às soluções desejadas, prezando por maior aptidão, mas sem ignorar completamente os indivíduos menos aptos, que podem possuir genes que contribuirão com a procura do ótimo global caso sejam recombinados em outros genótipos. Os pais selecionados, então, têm seus genes trocados por meio do *crossover*, os quais são atribuídos aos seus filhos. O mecanismo mais simples de *crossover* é a troca de cortes de comprimento arbitrário dos vetores de genótipo (como ilustrado na Fig. 4), mas é ideal que o *crossover* seja adaptado às características dos objetos de estudo. O esquema de *crossover* da Fig. 4 inclusive é adotado de forma igual ou similar em todos os algoritmos deste trabalho, o

que será detalhado no Capítulo 3.

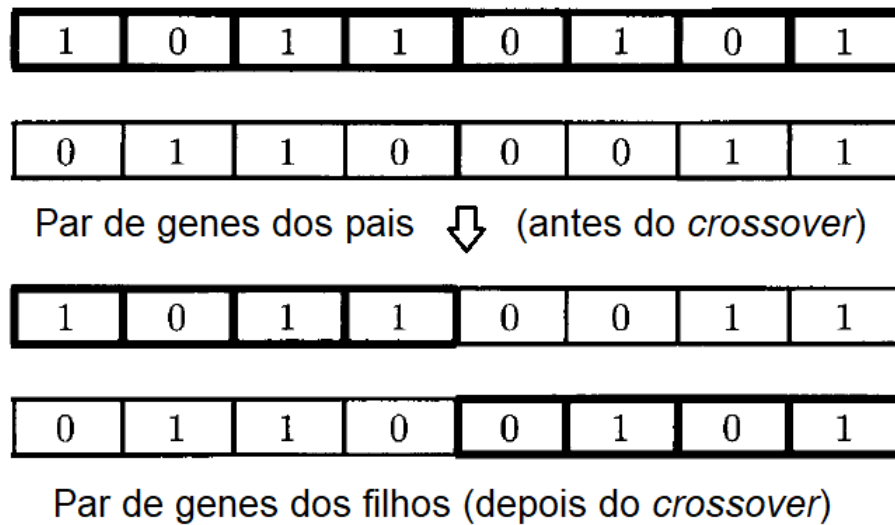


Figura 4 – Exemplo de uma operação de *crossover* (YANG, 2010)

O último estágio antes do final do *loop* é a mutação. Cada um dos indivíduos filhos têm uma chance de passar por mutação, a qual altera seus genes. A proposta da mutação é poder gerar, ocasionalmente, mais um pouco de variabilidade genética após o passo 2, o que pode ser benéfico dependendo da configuração do algoritmo. No entanto, é necessário precaução com esse processo porque a inserção demasiada de aleatoriedade nos genes de populações já estabelecidas pode dificultar a busca pelo ótimo global. A quantidade de genes alterados e a extensão de valores possíveis após a alteração são, novamente, adaptadas de acordo com o problema em mãos. Após isso, a população de filhos substitui a população original, dos pais, e o algoritmo segue o mesmo procedimento a partir do estágio de avaliação dos indivíduos. Desse modo, o algoritmo segue em *loop*, iterativamente, até que algum critério de parada seja satisfeito. O critério mais simples é o cumprimento de número máximo de iterações especificado pelo programador, mas também pode-se implementar, por exemplo, uma parada que seja acionada quando é detectado que as melhores soluções ao longo das iterações variam pouco (ou seja, um critério de convergência). Para os algoritmos deste trabalho, é utilizada a abordagem do número máximo de iterações.

Pode-se notar que o algoritmo genético contém um grau considerável de aleatoriedade. Os detalhes de seus funcionamentos variam de acordo com as intenções do programador, mas os processos de geração da população inicial, *crossover* e de mutação são, fundamentalmente, mecanismos aleatórios. O que torna o algoritmo genético efetivo é a importância da aptidão dos indivíduos em relação à função objetivo no momento de seleção para a reprodução, como explicado anteriormente.

Como a solução ótima é construída por meio de *crossovers* dos indivíduos, é impor-

tante que haja uma gama suficiente de variação genética quando o algoritmo é iniciado, o que implica que o desempenho do algoritmo genético é beneficiado por populações maiores - populações pequenas podem fazer com que o algoritmo convirja prematuramente em soluções distintas do ótimo global. No entanto, maiores populações trazem a desvantagem de requerir um tempo maior de execução. Essa é a fraqueza do algoritmo genético: é mais lento do que os algoritmos determinísticos por causa da repetição dos operadores genéticos para centenas ou milhares de indivíduos diferentes em várias iterações. Relações de compromisso como essas não são limitadas apenas ao tamanho populacional. É de extrema importância também a correta configuração dos outros parâmetros do algoritmo, como as probabilidades de seleção e as chances de mutação, para que seu comportamento seja como desejado.

O desempenho do algoritmo pode ser aprimorado com mecanismos de elitismo, que consistem em garantir que ao menos um dos indivíduos mais aptos de uma iteração seja copiado, inalterado, para a iteração seguinte. Esse procedimento é de grande utilidade, especialmente, quando o algoritmo converge para um determinado conjunto de genes e, por conta de seus mecanismos aleatórios, seus melhores resultados começam a oscilar em relação à função objetivo. O elitismo evita que o algoritmo regrida em relação aos seus melhores resultados obtidos em uma determinada sessão. Nos algoritmos deste trabalho há a implementação de elitismo, o que será detalhado no Capítulo 3.

2.3 Aerodinâmica

A mecânica dos fluidos trata do estudo da física referente ao comportamento de corpos líquidos e gasosos. Inserida nessa área do conhecimento está a aerodinâmica, que limita o estudo ao ar atmosférico. Desse modo, a aerodinâmica é uma ciência fundamental a praticamente qualquer veículo que locomova-se mergulhado na atmosfera da Terra - desde veículos terrestres, a veículos marítimos (especialmente aqueles movidos a vela), a, obviamente, aeronaves. E, de fato, o estudo da aerodinâmica é relevante mesmo àquilo que não se move, como, por exemplo, construções urbanas ([JęDRZEJEWSKI; POĆWIERZ; ZIELONKO-JUNG, 2017](#)).

Tratando da engenharia aeronáutica, o propósito da aerodinâmica é descrever os escoamentos ao redor das aeronaves de modo a prever as forças e momentos gerados sobre as mesmas. Isso permite o dimensionamento correto da aeronave, prezando por bom desempenho e segurança. As subseções seguintes apresentarão conceitos teóricos da área de aerodinâmica, enfatizando tópicos a respeito de asas e aerofólios de uso aeronáutico.

2.3.1 Forças e momentos

A força essencial que se busca em um projeto aeronáutico é a de sustentação, definida como

$$L = q_{\infty} S C_L \quad (2.1)$$

Onde S é a área de referência (por exemplo, a área da asa), C_L é o coeficiente de sustentação e q_{∞} é a pressão dinâmica, definida como

$$q_{\infty} = \frac{1}{2} \rho_{\infty} V_{\infty}^2 \quad (2.2)$$

Onde ρ_{∞} é a densidade e V_{∞} é a velocidade, ambas referentes ao vento relativo e imperturbado, indicado pelo subscrito ∞ .

O arrasto aerodinâmico é definido como

$$D = q_{\infty} S C_D \quad (2.3)$$

Onde C_D é o coeficiente de arrasto.

Finalmente, o momento causado pelas reações do escoamento é dado por

$$M = q_{\infty} S l C_M \quad (2.4)$$

Onde C_M é o coeficiente de momento e l é um comprimento de referência (por exemplo, a corda de um aerofólio). O tipo específico de momento a ser considerado neste trabalho é o momento de arfagem, o qual atua de modo a aumentar ou diminuir o ângulo de ataque. Como padrão, momentos (e coeficientes de momento) que induzem ao aumento do ângulo de ataque têm valores positivos, e aqueles que induzem à redução do ângulo de ataque têm valores negativos.

As Equações 2.1, 2.3 e 2.4 são aparentemente simples, o que é na verdade uma perspectiva ilusória. A complexidade inerente às reações do escoamento ao interagir com as geometrias sólidas está implícita nos coeficientes adimensionais C_L , C_D e C_M , os quais podem ser obtidos por meios analíticos, numéricos ou experimentais.

2.3.2 Método de painéis

Esta subseção trata dos métodos de painéis, métodos numéricos de simulação aerodinâmica que se popularizaram a partir dos anos 60. O texto a seguir discorre a respeito das duas classificações principais, os painéis de fonte e painéis de vórtice, sem ainda entrar no âmbito da formulação viscosa.

2.3.2.1 Painéis de fonte

Os métodos de painéis são baseados na teoria de escoamento potencial, o qual se caracteriza por fluidos invíscidos e incompressíveis. Os fundamentos de escoamentos potenciais relevantes a este trabalho estão nos Apêndices B a G, e recomenda-se que o leitor leia esses apêndices caso queira entender mais a fundo a teoria básica por trás dos solucionadores utilizados neste trabalho. De todo modo, em suma, escoamentos potenciais podem ser descritos por meio de funções de corrente (Ap. B) e funções de potencial (Ap. C), as quais fornecem dados sobre a velocidade do fluido, que, por sua vez, permite o cálculo de outros dados aerodinâmicos relevantes como a distribuição de coeficientes de pressão. Ambas essas funções são importantes porque são soluções da equação de Laplace (Ap. D) e, por essa equação ser linear, as soluções das funções de corrente e de potencial para escoamentos fundamentais (Ap. F) podem ser combinadas para formar fluxos mais complexos.

A Fig. 5, por exemplo, mostra uma combinação de escoamento uniforme com uma fonte, a qual resulta em linhas de corrente que a princípio progredem em linha reta e em seguida curvam-se para fora do ponto onde a fonte está localizada. Se a fonte fosse substituída por um sumidouro, as linhas de corrente, ao se aproximarem dele, seriam puxadas em sua direção, ainda seguindo da esquerda para a direita, mas mais próximas umas das outras. Na mesma figura, ψ é a função de corrente, Λ é a taxa de fluxo da fonte/sumidouro, e r e ϕ são as coordenadas polares de um determinado ponto em relação ao centro da fonte.

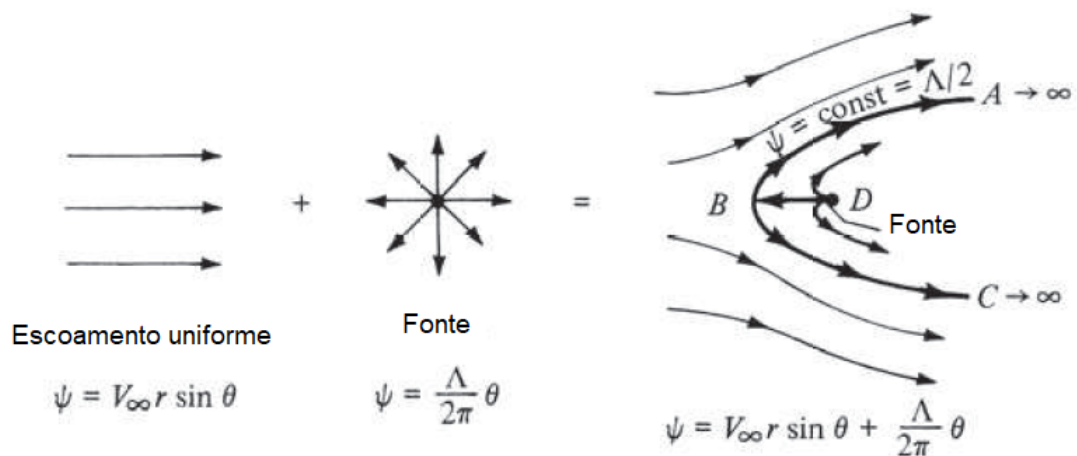


Figura 5 – Uma combinação de escoamento uniforme e fonte (ANDERSON, 2017)

Inicia-se, portanto, a discussão sobre como aplicar esses escoamentos em casos reais. Para começar, uma correlação clara que pode ser feita é que o padrão do escoamento combinado da Fig. 5 poderia ser tomado como o escoamento ao redor do bordo de ataque de um aerofólio. No ponto de espessura máxima, bastaria inserir um sumidouro para

que o escoamento fosse inclinado para dentro de novo, de modo a continuar a seguir o contorno do aerofólio, e assim em diante. Conclui-se, assim, que combinação de fontes e sumidouros é capaz de representar o escoamento ao redor de um dado contorno bidimensional. A capacidade de detalhamento dessa metodologia, evidentemente, é beneficiada com o aumento de fontes e sumidouros, e com o posicionamento dos mesmos ao longo da superfície do corpo, o que permite a modelagem de corpos assimétricos, ao contrário do que foi apresentado na Fig. 5 e no parágrafo anterior. Essa configuração, mostrada na Fig. 6, é chamada de folha de fontes.

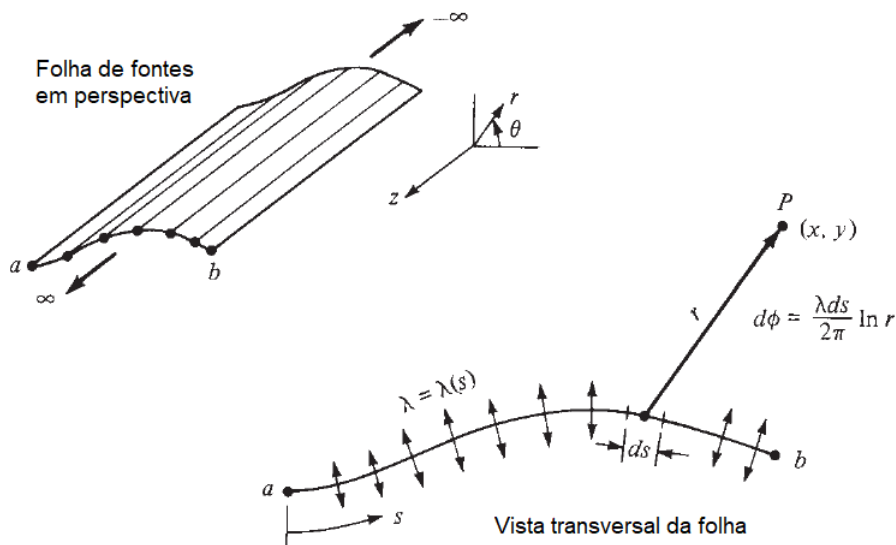


Figura 6 – Uma combinação de escoamento uniforme e fonte (ANDERSON, 2017)

Assume-se que a folha contém infinitas fontes, cada uma com intensidade infinitesimal. Sendo s a direção da folha, temos que $\lambda(s)$ é a intensidade de fonte, por unidade de comprimento, ao longo de s . Cada fragmento ds da folha induz um potencial $d\phi$ sobre um dado ponto arbitrário P , localizado a uma distância r de ds . Tendo em vista a Eq. F.4, esse potencial infinitesimal é

$$d\phi = \frac{\lambda ds}{2\pi} \ln r_{p,j} \quad (2.5)$$

O potencial completo sobre o ponto P , isto é, induzido pela folha completa, é, portanto,

$$\phi_P = \int_a^b \frac{\lambda ds}{2\pi} \ln r_{p,j} \quad (2.6)$$

Como não há uma solução fechada para $\lambda(s)$, adota-se uma abordagem numérica. A folha de fontes é discretizada em painéis. Cada painel tem, ao longo de seu comprimento,

intensidade λ constante, e as intensidades podem variar de painel para painel. Pode-se, então declarar o problema: dada uma geometria qualquer discretizada em n painéis, procura-se determinar os valores de λ em cada painel de modo que o escoamento ao redor da geometria seja satisfeito, com a superfície da geometria tornando-se uma linha de corrente.

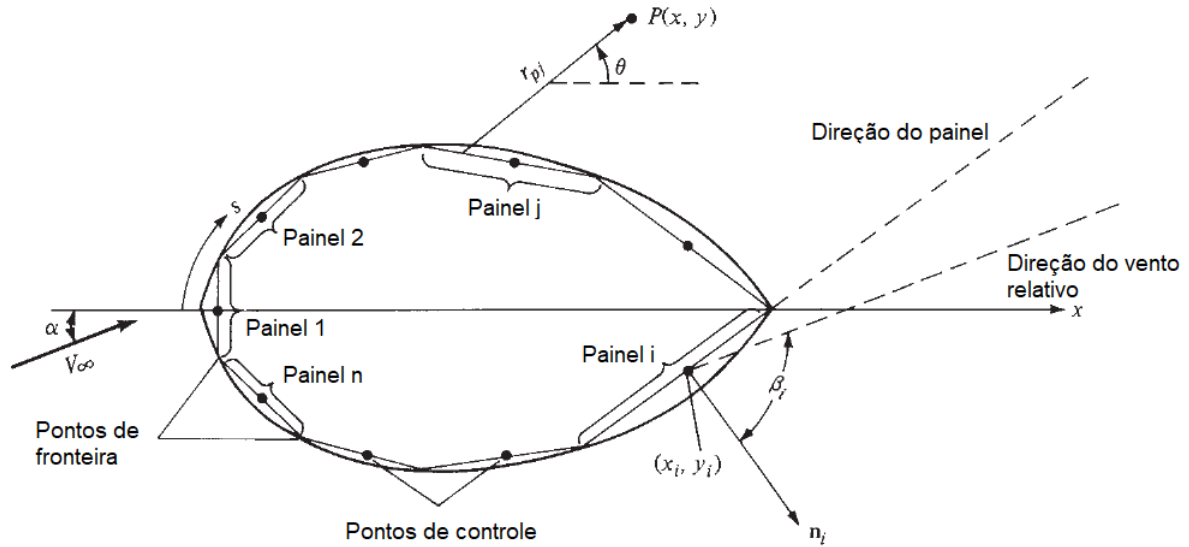


Figura 7 – Geometria e nomenclatura de um aerofólio discretizado em painéis (ANDERSON, 2017)

Os pontos de controle dos painéis são definidos no meio do comprimento dos mesmos, e estes têm o propósito de assegurar a condição de contorno de que a velocidade normal em um painel deve ser igual a zero. Isso garante que o ar não atravessará o contorno da geometria para dentro ou para fora. A partir de agora, adota-se a geometria de um aerofólio, como mostra a Fig. 7, e o ponto de referência é trocado de P para o ponto de controle do painel i , de modo que r torna-se a distância entre os pontos de controle dos painéis i e j . Então, o potencial de velocidade no painel i induzido por todos os outros painéis é igual a

$$\phi_i = \sum_{j=1}^n \frac{\lambda_j}{2\pi} \int_j \ln r_{i,j} ds_j \quad (2.7)$$

Aplica-se a condição de contorno. A velocidade normal ao painel i é uma soma das contribuições do vento relativo e do escoamento induzido pela fonte. A Eq. 2.8 representa o cumprimento da condição de contorno no painel i .

$$V_{\infty, n, i} + V_{n, i} = 0 \quad (2.8)$$

A velocidade $V_{n, i}$ é a contribuição da fonte, a qual pode ser calculada tendo em vista que as componentes de velocidade podem ser obtidas pela diferenciação do potencial

de velocidade. Portanto, derivando a Eq. 2.7 na direção normal, temos

$$V_{n,i} = \frac{\partial \phi_i}{\partial n_i} = \sum_{j=1}^n \frac{\lambda_j}{2\pi} \int_j \frac{\partial}{\partial n_i} (\ln r_{i,j}) ds_j \quad (2.9)$$

Quando $i = j$ a Eq. 2.9 encontra uma singularidade pois $r = 0$. No entanto, pode-se provar que, nessas condições, o potencial induzido pelo painel sobre ele mesmo é igual a $\lambda_i/2$. A Eq. 2.9 então é reescrita como

$$V_{n,i} = \frac{\lambda_i}{2} + \sum_{j=1, j \neq i}^n \frac{\lambda_j}{2\pi} \int_j \frac{\partial}{\partial n_i} (\ln r_{i,j}) ds_j \quad (2.10)$$

A componente normal $V_{\infty, n, i}$ vem da velocidade do vento relativo:

$$V_{\infty, n} = V_{\infty} \cos \beta_i \quad (2.11)$$

E então, inserindo as Equações 2.10 e 2.11 na Eq. 2.8, o resultado é

$$V_{\infty} \cos \beta_i + \frac{\lambda_i}{2} + \sum_{j=1, j \neq i}^n \frac{\lambda_j}{2\pi} \int_j \frac{\partial}{\partial n_i} (\ln r_{i,j}) ds_j = 0 \quad (2.12)$$

Substituindo a integral na Eq. 2.12 por valores $I_{i,j}$, ela é reescrita como

$$V_{\infty} \cos \beta_i + \frac{\lambda_i}{2} + \sum_{j=1, j \neq i}^n \frac{\lambda_j}{2\pi} I_{i,j} = 0 \quad (2.13)$$

Aplicando a Eq. 2.13 para cada um dos painéis, obtém-se um sistema linear com n equações e n incógnitas, sendo estas últimas as intensidades das fontes dos painéis.

Cumprido esse passo é possível calcular as velocidades tangentes aos painéis. Como o ar é tratado como invíscido, a velocidade diretamente na superfície do aerofólio é não nula. Novamente, essa velocidade tem contribuições tanto da corrente livre quanto do escoamento induzido pelo painel:

$$V_{\infty, t, i} + V_{t, i} = V_i \quad (2.14)$$

A componente referente ao painel é obtida, mais uma vez, por meio da derivação da função de potencial na direção desejada:

$$V_{t, i} = \frac{\partial \phi_i}{\partial s} = \sum_{j=1}^n \frac{\lambda_j}{2\pi} \int_j \frac{\partial}{\partial s} (\ln r_{i,j}) ds_j \quad (2.15)$$

E a componente referente à corrente livre é

$$V_{\infty,s} = V_{\infty} \sin \beta_i \quad (2.16)$$

Substituindo as Equações 2.15 e 2.16 na Eq. 2.14, temos

$$V_{t,i} = V_{\infty} \sin \beta_i + \sum_{j=1}^n \frac{\lambda_j}{2\pi} \int_j \frac{\partial}{\partial s} (\ln r_{i,j}) ds_j \quad (2.17)$$

Que fornece as velocidades tangentes em todos os painéis, as quais, por sua vez, habilita o cálculo dos coeficientes de pressão:

$$C_{p,i} = 1 - \left(\frac{V_i}{V_{\infty}} \right)^2 \quad (2.18)$$

Em nota, a Eq. 2.15 também tem uma singularidade quando $i = j$. Ela é solucionada com o fato de que, nessas condições, a velocidade induzida pelo painel sobre ele próprio é nula, pois, como visto na Fig. 6, as fontes atuam apenas na direção normal à superfície. Adicionalmente, é relevante notar que a contribuição do escoamento uniforme (Eq. F.2) foi contabilizada ao longo da resolução do problema, como visto nas Equações 2.11 e 2.16, por exemplo. Uma solução alternativa seria incluir no potencial da velocidade, desde o início, o escoamento uniforme:

$$\phi_i = ux + vy + \sum_{j=1}^n \frac{\lambda_j}{2\pi} \int_j \ln r_{i,j} ds_j$$

Onde $u = V_{\infty} \cos \alpha$ e $v = V_{\infty} \sin \alpha$. Essa estratégia resultaria nas mesmas conclusões que as alcançadas pela abordagem explicada.

2.3.2.2 Painéis de vórtice

Os métodos de painéis de fonte não são capazes de estimar forças de sustentação porque, como explicado nos Apêndices E e F, a sustentação é uma consequência direta da circulação, e a circulação é não nula apenas na presença de vórtices. Portanto, desenvolveu-se um tipo distinto de método de painéis com o intuito de satisfazer essa necessidade. O método de painéis de vórtice são fundamentalmente similares aos de fonte, com a diferença que a folha de fontes é convertida em uma folha de vórtices com intensidade $\gamma(s)$, analogamente à Fig. 6.

Novamente assume-se um aerofólio discretizado em painéis. Seguindo a mesma estratégia da Eq. 2.5 à Eq. 2.7, desta vez usando o potencial do escoamento em vórtice (Eq. F.8), temos que

$$\phi_i = - \sum_{j=1}^n \frac{\gamma_j}{2\pi} \int_j \theta ds_j \quad (2.19)$$

É a contribuição ao potencial no painel i gerado por todos os painéis, onde θ é o ângulo entre os pontos de controle dos painéis i e j .

Como antes, é preciso garantir o cumprimento da condição de contorno de que a componente normal da velocidade do ar em cada painel é nula, como expressa a Eq. 2.8. A contribuição da vorticidade dos painéis é

$$V_{n,i} = \frac{\partial \phi_i}{\partial n} = - \sum_{j=1}^n \frac{\gamma_j}{2\pi} \int_j \frac{\partial \theta_{i,j}}{\partial n_i} ds_j \quad (2.20)$$

E a contribuição da corrente livre é dada pela Eq. 2.11. Inserindo-a na Eq. 2.8 com a Eq. 2.20, temos

$$V_\infty \cos \beta_i - \sum_{j=1}^n \frac{\gamma_j}{2\pi} \int_j \frac{\partial \theta_{i,j}}{\partial n_i} ds_j = 0 \quad (2.21)$$

Substituindo a integral da Eq. 2.21 por um valor $J_{i,j}$ ela se torna

$$V_\infty \cos \beta_i - \sum_{j=1}^n \frac{\gamma_j}{2\pi} J_{i,j} = 0 \quad (2.22)$$

A Eq. 2.22 é similar o suficiente à Eq. 2.13 para induzir o projetista a já solucionar o sistema de equações. No entanto, por estarem incorporados à geração de sustentação, os painéis de vórtice precisam também cumprir a condição de Kutta para garantir a veracidade de seus resultados. A forma comum de se fazer isso é introduzindo uma nova equação que garante que os dois painéis mais próximos ao bordo de fuga cancelem suas vorticidades mutuamente:

$$\gamma_{TE,1} + \gamma_{TE,2} = 0 \quad (2.23)$$

A adição da Eq. 2.23 torna o sistema sobredeterminado, o que exige a remoção de uma das equações do sistema formado pela Eq. 2.22. Qual será a equação a ser removida é uma decisão que cabe ao projetista.

2.3.3 Método de painéis 3D

Toda a teoria utilizada no método de painéis pode ser readaptada para o regime tridimensional. Novamente, a estratégia é discretizar a geometria em um número finito de painéis, onde a cada um deles é atribuída uma singularidade (fontes/sumidouros e/ou vórtices), e o escoamento é solucionado determinando-se os valores dessas singularidades por meio de um sistema linear de equações algébricas. A Fig. 8 apresenta um exemplo de uma aeronave discretizada em painéis 3D, incluindo painéis de trilha.

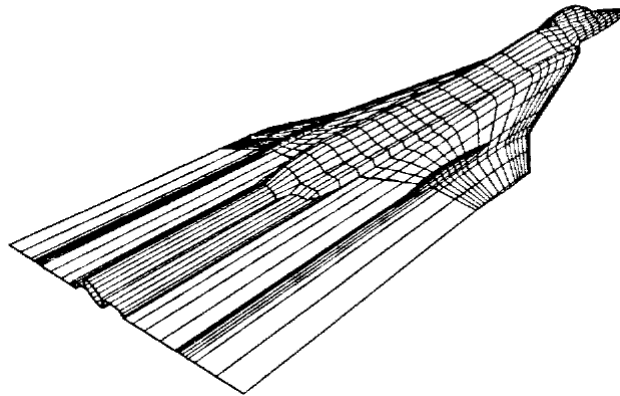


Figura 8 – Modelo em painéis 3D do F-16XL (ERICKSON, 1990)

Igualmente à sua versão bidimensional, o método de painéis 3D também é uma ferramenta poderosa de *design* preliminar, permitindo a estimativa de dados aerodinâmicos de aeronaves inteiras em questão de minutos ou mesmo segundos. O método trabalha com escoamentos invíscidos, incompressíveis e irrotacionais, mas correções de compressibilidade e estimativas de camada limite podem ser implementadas. Nota-se que arrasto induzido e arrasto de onda podem ser ambos estimados com confiabilidade mesmo sem correções, pois ambos estes fenômenos podem ser calculados dentro da teoria de escoamentos invíscidos.

2.3.4 Método da treliça de vórtices (VLM)

O VLM (*Vortex Lattice Method*) é uma expansão da teoria da linha sustentadora de Prandtl. A descrição detalhada dessa teoria está fora do escopo deste trabalho, e pode ser checada em literatura como (ANDERSON, 2017), mas, em suma, a proposta é transformar uma asa em uma superposição de um número infinito de filamentos de vórtice com intensidade $d\Gamma$. Com esses dados, é possível obter soluções analíticas para a distribuição de Γ ao longo da envergadura, o que, como se mostra no Ap. E, permite determinar a distribuição de sustentação e outros dados relacionados à mesma, como o *downwash* e o arrasto induzido.

A teoria da linha sustentadora, no entanto, deixa de fornecer bons resultados para asas com baixa razão de aspecto, com enflechamento, e asas em delta (explicações sobre geometrias de asas serão apresentadas na Seção 2.6). A partir desses impasses, surgiu o VLM, uma adaptação da teoria da linha sustentadora. Ao invés de transformar uma asa em uma linha com uma distribuição de vórtices, admite-se que a asa é uma superfície de espessura desprezível, composta de um número de linhas sustentadoras distribuídas ao longo da direção da corda. A Fig. 9 mostra essa representação.

A intensidade dos vórtices na direção x é dada por $\delta(x, y)$ e a intensidade dos

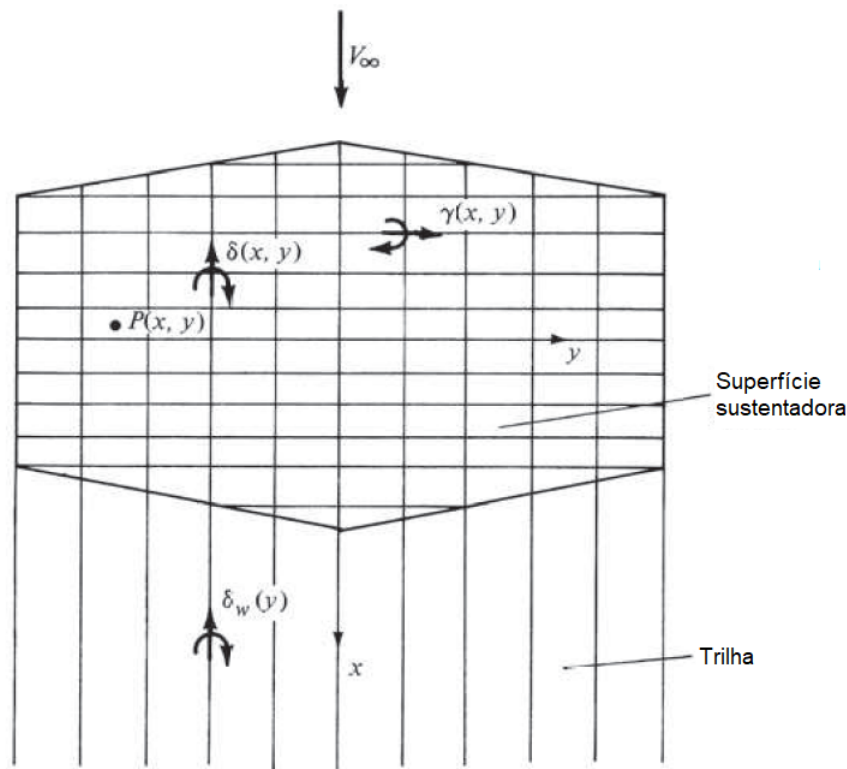


Figura 9 – Representação de uma asa arbitrária em termos da teoria do VLM (ANDERSON, 2017)

vórtices na direção y é dada por $\gamma(x, y)$. Sendo assim, a força da superfície completa é uma soma de $\gamma(x, y)$ com $\delta(x, y)$. Todos esses vórtices induzem velocidades normais ao plano xy , denominadas como $w(x, y)$, e a chave do VLM se baseia em encontrar os valores de $\gamma(x, y)$ e $\delta(x, y)$ de modo que w seja sempre nulo em toda a superfície da asa - afinal, o ar escoar ao redor da asa, e não através dela.

A solução encontrada para o método de painéis foi uma solução numérica porque, como visto na Subseção 2.3.2, não há uma solução fechada para $\lambda(s)$. No desenvolvimento do VLM ocorre uma situação similar - também não há soluções analíticas para $\delta(x, y)$, $\gamma(x, y)$ e $w(x, y)$, portanto, adota-se uma solução numérica. A superfície é discretizada em um número finito de painéis com seus respectivos pontos de controle, nos quais é imposta a condição de contorno $w(x, y) = 0$. Também como no método de painéis, os valores de $\delta(x, y)$ e $\gamma(x, y)$ podem ser tratados como constantes nos painéis ou não, o que afeta a precisão dos resultados. Correções de compressibilidade e de viscosidade também podem ser aplicadas.

A teoria principal utilizada nos solucionadores selecionados para este trabalho trata o ar como invíscido, o que é uma idealização com o objetivo de simplificar os cálculos. No mundo real, a viscosidade dos fluidos é responsável por uma série de comportamentos característicos substanciais à dinâmica dos fluidos. Mais detalhes sobre alguns dos efeitos

da viscosidade sobre o escoamento de fluidos podem ser encontrados no Ap. H.

2.4 Aerofólios

A obtenção de dados confiáveis a respeito dos fenômenos físicos referentes a um projeto de engenharia é essencial. É importante, também, que a obtenção desses dados seja de menor complexidade possível para facilitar o trabalho do projetista - especialmente considerando contextos de *design* iterativo e/ou projetos com muitas relações de compromisso, o que exige repetir cálculos e análises. O projeto da asa de uma aeronave é um exemplo de uma situação com requisitos conflitantes que afetam a aeronave como um todo. Afortunadamente, logo no início do século XX foi reconhecida a importância dos aerofólios para a análise da asa completa. Aerofólios, ou perfis, são as seções transversais de asas, empenagens, *canards* e outras superfícies aerodinâmicas, e sua nomenclatura básica é apresentada na Fig. 10.

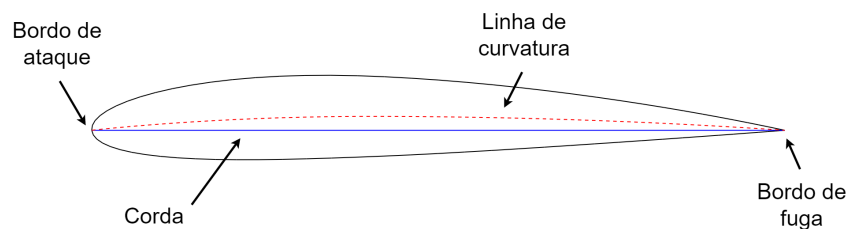


Figura 10 – Nomenclatura de um aerofólio (fonte: autor)

O bordo de ataque e o bordo de fuga são os pontos da frente e de trás do aerofólio, respectivamente, e a corda é a distância entre ambos os bordos. A distribuição de curvatura é representada na forma da linha de curvatura, ou linha média, a qual é igual a zero caso o perfil seja simétrico. As superfícies de cima e de baixo do aerofólio são nomeadas, respectivamente, extradorso e intradorso.

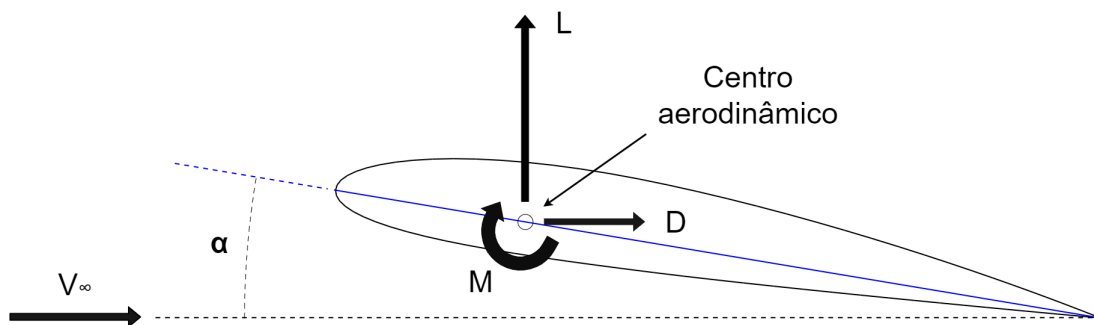


Figura 11 – Forças e momentos agindo no aerofólio (fonte: autor)

As propriedades aerodinâmicas principais de um aerofólio são apresentadas na Fig. 11. A sustentação L e o arrasto D são forças que agem, respectivamente, perpendicular-

mente e paralelamente à direção de voo, e o momento de arfagem M age de modo a alterar o ângulo de ataque α . O ângulo de ataque é o ângulo de inclinação do aerofólio (tomando a corda como referência) em relação ao vento relativo V_∞ . O centro aerodinâmico, normalmente próximo ao ponto de $1/4$ da corda em voo subsônico, é o ponto onde o momento é geralmente independente do ângulo de ataque. Dentro da análise das forças e momentos nos aerofólios existe também o centro de pressão, no qual o momento resultante é igual a zero. O centro de pressão, no entanto, se move com a variação do ângulo de ataque, o que justifica a preferência para o centro aerodinâmico quando se deseja calcular as forças e os momentos no aerofólio. Em nota, é importante esclarecer que esses conceitos não são exclusivos a aerofólios - outras partes de uma aeronave também geram contribuições de sustentação, arrasto e momento, as quais afetam as posições do centro aerodinâmico e do centro de pressão da aeronave completa.

Sendo um objeto bidimensional, a análise do aerofólio é menos complexa que a da asa, tanto analiticamente, quanto numericamente, quanto experimentalmente. Para este último, o teste experimental de aerofólios consiste em fabricar uma seção de asa uniforme (sem afilamento e com o mesmo aerofólio ao longo de toda a envergadura) e testá-la dentro de um túnel de vento cujas paredes encostem nas pontas da seção de asa. Isso efetivamente anula as componentes de escoamento ao longo da envergadura e faz com que a seção de asa "sinta" um fluxo apenas bidimensional, como os que escoariam sobre os aerofólios das Figuras 10 e 11. Um exemplo dessa configuração é mostrado na Fig. 12.

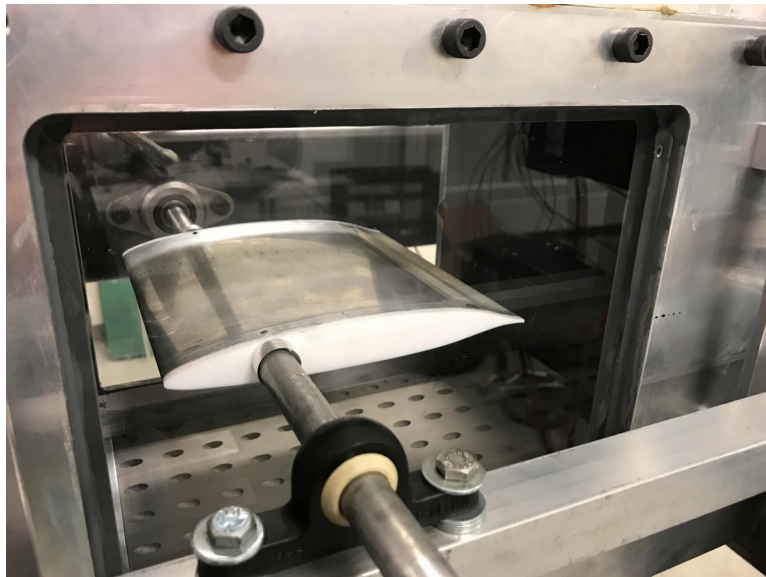


Figura 12 – Um aerofólio montado em um túnel de vento (LARSON, 2021)

Essa distinção de características do escoamento bidimensional e do tridimensional já permite deduzir que existem diferenças entre aerofólios e asas. No entanto, com a metodologia correta, aerofólios podem ser utilizados para prever as características das asas nas quais eles serão utilizados, e isso vai além dos coeficientes de sustentação, arrasto e

momento. As características de transição de fluxo laminar para turbulento são afetadas diretamente pelo formato do aerofólio. O comportamento no estol de uma dada asa também é ditado, em grande parte, pelos aerofólios - um projetista pode, por exemplo, adotar a torção aerodinâmica, que consiste em transicionar os aerofólios ao longo da envergadura de modo que os aerofólios da raiz e da ponta sejam melhores adaptados aos escoamentos em suas respectivas posições, fazendo com que a asa como um todo tenha características de estol mais amigáveis. Fora isso, o projeto de aerofólios é essencial para garantir condições de voo ótimas em regimes transônicos, supersônicos e hipersônicos, nos quais a presença de ondas de choque e fenômenos termodinâmicos proporcionam estudos de caso quase completamente distintos aos cenários subsônicos. Todos esses fatores e outros não mencionados também são cruciais a empenagens, *canards*, e quaisquer outras geometrias que usem aerofólios como seções transversais.



Figura 13 – O aerofólio de um planador ASW 28-18 visível como um corte transversal na asa (MüNCH, 2021)

Desse modo, tendo em vista a importância dos aerofólios para projetos aeronáuticos, e considerando o quão complexa a questão pode se tornar, fica claro o porquê de aerofólios serem adotados em problemas de otimização com tamanha frequência. Dentro desse quesito, a parametrização geométrica dos aerofólios a ser empregada é um dos aspectos mais importantes para a configuração do problema. Ao se selecionar um determinado método de parametrização, deve-se ter ciência de suas capacidades, limitações, e de que forma sua formulação se adapta ao algoritmo escolhido. Para este trabalho, dois métodos de parametrização de geometria foram escolhidos: a série NACA 4 dígitos e o CST. Os detalhes de ambos serão expostos nas subseções a seguir.

2.4.1 NACA 4 dígitos

Até dado momento na história, o desenvolvimento de aerofólios foi empírico em grande parte, de modo que seu *design* era basicamente personalizado à aeronave que se desejava projetar. No início dos anos 30, o NACA (*National Advisory Committee for Aeronautics*) iniciou uma série de investigações para estabelecer um método sistemático de projetar aerofólios, e o que foi constatado é que uma forma muito útil de se descrever a geometria de um aerofólio é fazê-lo em termos de uma distribuição de espessura e de uma distribuição de curvatura. A partir disso surgiu a série NACA 4 dígitos, a primeira das famílias de aerofólios desenvolvidas pelo NACA ao longo da primeira metade do século XX.

No início das investigações, notou-se que perfis eficientes da época como o Göttingen 398 e o Clark Y tinham distribuições de espessura bastante similares, o que pôde ser averiguado quando suas curvaturas eram zeradas, tornando-os perfis simétricos com espessuras máximas iguais (ABBOTT; DOENHOFF, 1959). A distribuição de espessura dos perfis NACA 4 dígitos foi projetada para se assemelhar a esses perfis, e sua formulação matemática é

$$\pm y_t = \frac{t}{0,20} (0,29690\sqrt{x} - 0,12600x - 0,35160x^2 + 0,28430x^3 - 0,10150x^4) \quad (2.24)$$

Onde t é o valor da espessura máxima em porcentagem da corda e x é o valor da ordenada à qual será associada a abscissa y_t . A Eq. 2.24 já é suficiente para traçar perfis simétricos. Para aerofólios com curvatura, é necessário um conjunto adicional de equações para traçar a linha média assimétrica. Essas equações são

$$y_c = \begin{cases} \frac{m}{p^2}(2px - x^2) & \text{se } x \leq p \\ \frac{m}{(1-p)^2} [(1-2p) + 2px - x^2] & \text{se } x > p \end{cases} \quad (2.25)$$

Onde m é a curvatura máxima em porcentagem da corda e p é a posição da curvatura máxima em décimos da corda. Sendo assim, temos que m , p e t são os parâmetros que definem a geometria dos aerofólios e também sua nomenclatura. Por exemplo, o perfil NACA 2412 tem $m = 2$, $p = 4$ e $t = 12$, o que indica que sua curvatura máxima é de 2% da corda, localizada a $0,4c$ a partir do bordo de ataque, e que sua espessura máxima é de 12% da corda.

Os aerofólios NACA 4 dígitos foram desenvolvidos há quase um século atrás, mas devido à disponibilidade de seus dados experimentais (ABBOTT; DOENHOFF, 1959) e suas características amigáveis de *performance*, como características suaves de estol e baixa suscetibilidade a imperfeições na superfície das asas que causem transição de fluxo (SCOTT, 2001), os perfis NACA 4 dígitos seguem em uso até hoje, desde aeronaves civis de

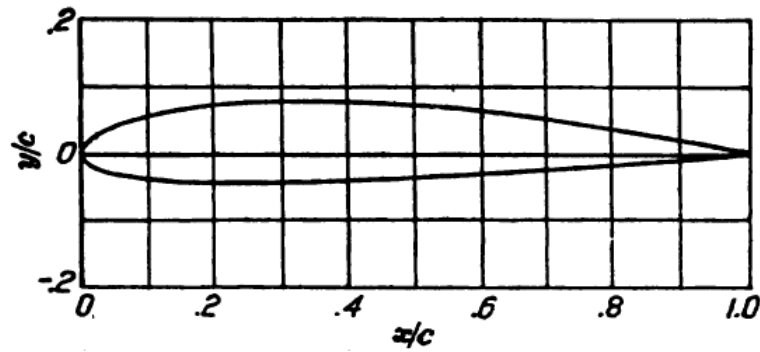


Figura 14 – Ilustração de um aerofólio NACA 2412 (ABBOTT; DOENHOFF, 1959)

pequeno porte a aviões de combate (LEDNICER, 2010). O NACA seguiu desenvolvendo outras séries de aerofólios, como os 5 dígitos e 6 dígitos, alguns dos quais também veem uso em projetos contemporâneos.

2.4.2 CST

Havendo a intenção de se melhorar os resultados obtidos pelos algoritmos, é necessário empregar métodos que permitam modificações mais minuciosas nas geometrias. Os aerofólios NACA 4 dígitos não cumprirão todas as ambições possíveis que um *designer* pode ter porque as equações que os geram cumprem apenas, e satisfatoriamente, o que elas foram projetadas para fazer. Sendo assim, no algoritmo genético de otimização de aerofólios foi implementada também outra forma de parametrização geométrica, denominada CST: *Class and Shape function Transformation*. Assumindo um plano de ordenadas x e abscissas z , temos que sua formulação fundamental é

$$\frac{z}{c}(x/c) = C(x/c) \cdot S(x/c) + \frac{x}{c} \cdot \frac{\Delta Z_{TE}}{c} \quad (2.26)$$

Onde c é o comprimento da corda, $C(x/c)$ é a função *class*, $S(x/c)$ é a função *shape* e ΔZ_{TE} é a altura do bordo de fuga em relação à horizontal, também denominado como espessura do bordo de fuga (KULFAN; BUSSOLETTI, 2006). A função *class* é

$$C(x/c) = \left(\frac{x}{c}\right)^{N1} \left[1 - \frac{x}{c}\right]^{N2} \quad (2.27)$$

Onde $N1$ e $N2$ são parâmetros geométricos. Por meio desses parâmetros a função *class* determina o formato geral da geometria que será representada e, na verdade, apenas com ela já é possível traçar geometrias¹. Assumindo que a função *shape* é unitária ($S(x/c) = 1$) e que não há espessura no bordo de fuga, a Eq. 2.26 gera o formato indicado na Fig. 15 quando $N1 = N2 = 1$.

¹ Todas as geometrias geradas nas demonstrações desta Subseção o foram feitas multiplicando-se as ordenadas por um valor de reescala igual a 0,5 para reduzir a altura das figuras.

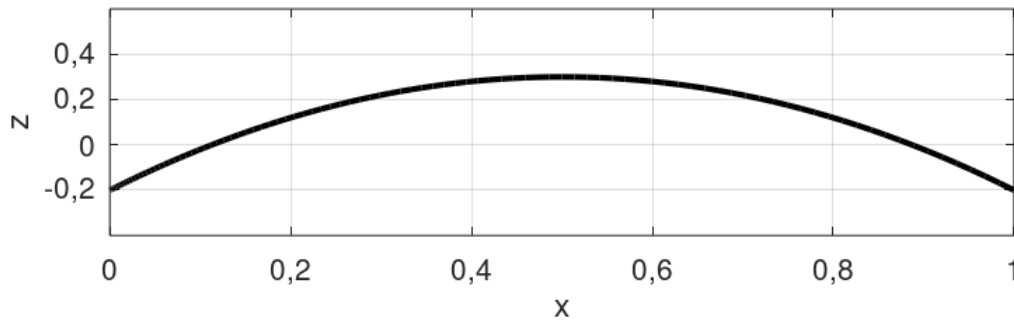


Figura 15 – Exemplo de geometria gerado pelo CST, usando função *shape* unitária (fonte: autor)

Gerando um conjunto de ordenadas igual às da Fig. 15, porém multiplicado por -1 de modo que se produza um formato simétrico, completa-se o contorno. A Fig. 16 apresenta mais exemplos dessa estratégia, agora variando os parâmetros $N1$ e $N2$. A Fig. 16a mostra um corpo ogival, a Fig. 16b mostra um aerofólio convencional, a Fig. 16c mostra uma cunha (ou cone) e a Fig. 16d mostra um projétil. É evidente que o CST pode ser aplicado como parametrização de outras geometrias que não sejam aerofólios - por exemplo, poderia-se descrever algo como uma fuselagem ou uma nacele de motor como um corpo de revolução ou em termos de seções transversais, sendo ambas as estratégias realizáveis por meio das geometrias bidimensionais do CST.

Para especificar os detalhes da geometria é utilizada a função *shape*. Para entender como isso é realizado, primeiro ela é decomposta em componentes de formato. Por exemplo, assume-se que a função *shape* é a indicada na Eq. 2.28.

$$S(x/c) = S1(x/c) + S2(x/c) \quad (2.28)$$

Onde $S1(x/c)$ e $S2(x/c)$ são, respectivamente,

$$S1(x/c) = 1 - \frac{x}{c} \quad (2.29)$$

$$S2(x/c) = \frac{x}{c} \quad (2.30)$$

As Eq. 2.29 e 2.30, quando somadas, retornam novamente a função *shape* unitária. No entanto, ao atribuir pesos às funções $S1(x/c)$ e $S2(x/c)$, a função *shape* passa a alterar regiões específicas do contorno da geometria. Desse modo, multiplicando-as, respectivamente, por duas variáveis de *design* quaisquer A_1 e A_2 , temos que a equação 2.28 torna-se

$$S(x/c) = A_1 \left(1 - \frac{x}{c}\right) + A_2 \left(\frac{x}{c}\right) \quad (2.31)$$

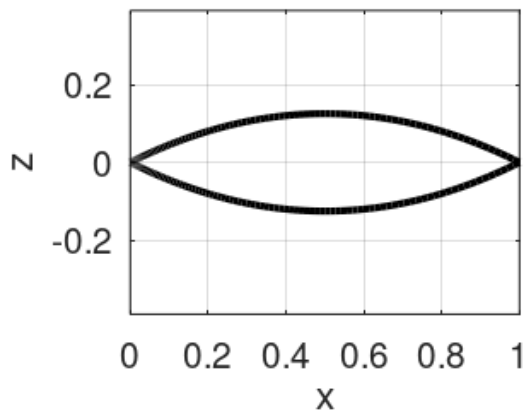
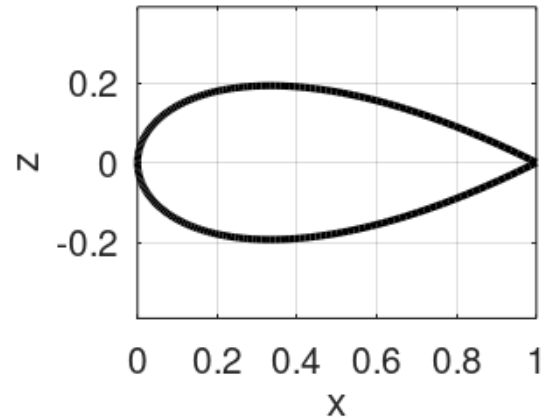
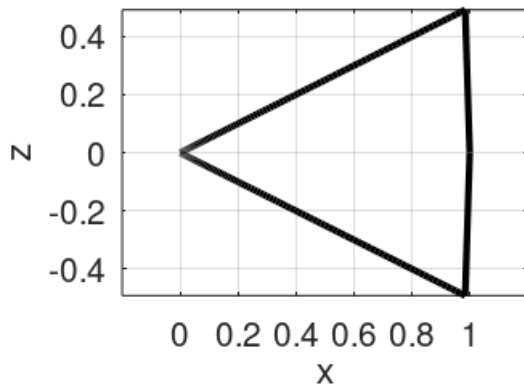
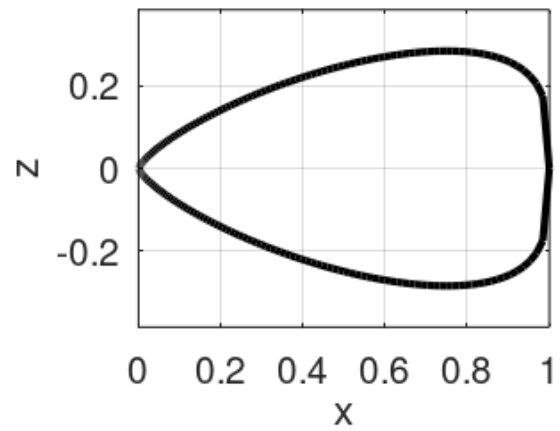
(a) Ogiva: $N1 = 1, N2 = 1$ (b) Aerofólio: $N1 = 0,5, N2 = 1$ (c) Cone: $N1 = 1, N2 = 1e - 10$ (d) Projétil: $N1 = 0,75, N2 = 0,25$

Figura 16 – Exemplos de tipos distintos de geometria CST (fonte: autor)

Inserindo a Eq. 2.31 na Eq. 2.26, junto com a função *class* (Eq. 2.27) configurada para gerar aerofólios, ou seja, com $N1 = 0,5$ e $N2 = 1$, e novamente ignorando a espessura do bordo de fuga, temos que a parametrização CST se torna

$$\frac{z}{c}(x/c) = \sqrt{\frac{x}{c}} \left(1 - \frac{x}{c}\right) \left[A_1 \left(1 - \frac{x}{c}\right) + A_2 \left(\frac{x}{c}\right) \right] \quad (2.32)$$

A Fig. 17 ilustra os efeitos dos pesos no traçado da geometria comparados à função *shape* unitária. Quando $A_1 = 2$ e $A_2 = 1$, o bordo de ataque se torna maior, e quando $A_1 = 1$ e $A_2 = 2$, o bordo de fuga fica mais inclinado.

Observando a Eq. 2.31, nota-se que os termos do lado esquerdo poderiam ser tomados como a expansão de um polinômio de Bernstein de primeira ordem multiplicado por variáveis de *design* A_r . O polinômio de Bernstein de grau n é descrito por meio da Eq. 2.33.

$$BP_n = \sum_{r=0}^n K_{r,n} x^r (1-x)^{n-r} \quad (2.33)$$

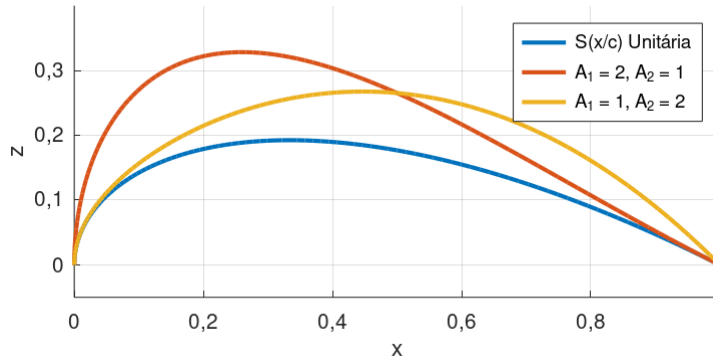


Figura 17 – Contornos de extradorso gerados pela Eq. 2.32 ilustrando os efeitos das variáveis de *design* (fonte: autor)

Onde $K_{r,n}$ são coeficientes binomiais definidos como

$$K_{r,n} = \binom{n}{r} = \frac{n!}{r!(n-r)!} \quad (2.34)$$

Com essas informações em mente, pode-se afirmar que o primeiro termo do polinômio afeta o bordo de ataque da geometria, enquanto que o último termo afeta o bordo de fuga. Caso o grau do polinômio seja maior que 1, a sua expansão produzirá termos intermediários que permitem a manipulação da geometria entre os bordos de ataque e de fuga. Sendo assim, define-se uma nova função *shape*, que se trata do polinômio de Bernstein multiplicado por variáveis de *design* A_r :

$$S(x/c) = \sum_{r=0}^n \left[A_r \left[K_{r,n} \left(\frac{x}{c} \right)^r \left(1 - \frac{x}{c} \right)^{n-r} \right] \right] \quad (2.35)$$

O primeiro peso é definido como

$$A_0 = S(0) = \sqrt{\frac{2R_{Le}}{c}} \quad (2.36)$$

Onde R_{Le} é o raio do bordo de ataque e o último peso é definido como

$$A_n = S(1) = \tan \beta + \frac{\Delta Z_{TE}}{c} \quad (2.37)$$

Onde β é o ângulo de inclinação do bordo de fuga. As variáveis de *design* intermediárias que surgem quando $n \geq 2$ têm valores unitários como padrão, mas podem ter outros valores atribuídos desde que $A_r \in \mathbb{R}$. Inserindo a Eq. 2.35 na Eq. 2.26, inserindo a Eq. 2.27 com os parâmetros corretos ($N1 = 0,5$ e $N2 = 1$) na Eq. 2.26 e adotando $\psi = x/c$, $\zeta = z/c$ e $\Delta\xi = \Delta Z_{te}/c$, temos a formulação CST para os aerofólios:

$$\zeta(\psi) = \sqrt{\psi(1-\psi)} \sum_{r=0}^n \left[A_r \left[K_{r,n} \psi^r (1-\psi)^{n-r} \right] \right] + \psi \Delta\xi \quad (2.38)$$

O CST, portanto, para uma das superfícies do aerofólio, tem $n + 1$ variáveis de design mais o ΔZ_{TE} . Se o aerofólio não for simétrico, deve-se especificar um conjunto de variáveis exclusivo para cada superfície. Os pesos intermediários da função *shape* são não-intuitivos, o que pode ser uma desvantagem dependendo das intenções do projetista, e que também já foi alvo de melhoras em estudos como (ZHU; QIN, 2014). No entanto, devido à sua ampla capacidade de parametrização, eficiência e robustez, o CST é um dos métodos mais populares para a otimização de aerofólios atualmente.

2.5 XFOIL

Os dois tipos de método de painéis apresentados na Subseção 2.3.2 podem ser combinados na forma do método de painéis de fonte/vórtice, no qual as fontes simulam a distribuição de espessura do aerofólio e os vórtices garantem as condições para a geração de sustentação. O XFOIL é, possivelmente, um dos exemplos mais conhecidos de emprego dessa metodologia, senão o *software* de método de painéis mais popular. Desenvolvido nos anos 80 como uma ferramenta para o projeto Daedalus do MIT, o XFOIL eventualmente foi lançado ao público geral como um *software open source*.

Como visto em (DRELA, 1989), a formulação fundamental, invíscida, do XFOIL é

$$\psi(x, y) = u_\infty y - v_\infty x + \frac{1}{2\pi} \int \gamma(s) \ln r(s; x, y) ds + \frac{1}{2\pi} \int \lambda(s) \theta(s; x, y) ds \quad (2.39)$$

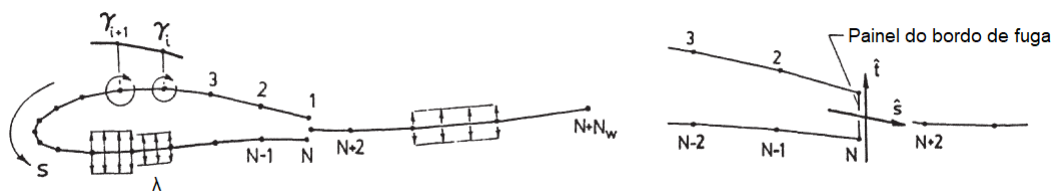


Figura 18 – Geometria dos painéis no XFOIL (DRELA, 1989)

A Eq. 2.39 é mais um exemplo da propriedade da equação de Laplace mencionada na Subseção 2.3.2.1 e explicada no Ap. F: a função de corrente induzida em um ponto qualquer com coordenadas (x, y) é uma soma dos efeitos do escoamento uniforme, do escoamento de vórtice e do escoamento de fonte/sumidouro. Discretizando a superfície em painéis, como na Fig. 18, a Eq. 2.39 se torna

$$\begin{aligned}
\psi(x, y) = & u_\infty y - v_\infty x + \frac{1}{4\pi} \sum_{j=1}^{N+N_w-1} \psi_j^\lambda(x, y) 2\lambda_j \\
& + \frac{1}{4\pi} \sum_{j=1}^{N-1} \left[\psi_j^{\gamma^+}(x, y) (\gamma_{j+1} + \gamma_j) + \psi_j^{\gamma^-}(x, y) (\gamma_{j+1} - \gamma_j) \right] \\
& + \frac{1}{4\pi} \left(\psi_N^\lambda(x, y) |\hat{s} \cdot \hat{t}| + \psi_N^{\gamma^+}(x, y) |\hat{s} \times \hat{t}| \right) (\gamma_1 - \gamma_N)
\end{aligned} \tag{2.40}$$

O método do XFOIL adiciona também painéis de fonte ao longo da esteira do aerofólio, os quais permitem a visualização do escoamento ao deixar o aerofólio. Adicionalmente, os cálculos foram idealizados, primariamente, para perfis com bordos de fuga retos, o que justifica a presença dos vetores \hat{s} e \hat{t} - normal e tangente ao bordo de fuga, respectivamente. O painel do bordo de fuga tem intensidades de fonte e de vórtice, respectivamente, iguais a

$$\sigma_{TE} = \frac{1}{2} (\gamma_1 - \gamma_N) |\hat{s} \cdot \hat{t}| \tag{2.41}$$

$$\gamma_{TE} = \frac{1}{2} (\gamma_1 - \gamma_N) |\hat{s} \times \hat{t}| \tag{2.42}$$

O que justifica a última linha da Eq. 2.40. A correção necessária para bordos de fuga pontiagudos é fornecida em (DRELA, 1989). As funções unitárias de corrente $\psi_j^{\gamma^+}(x, y)$, $\psi_j^{\gamma^-}(x, y)$ e $\psi_j^\lambda(x, y)$ são definidas como

$$\psi_j^{\gamma^+}(x, y) = \bar{x}_1 \ln r_1 - \bar{x}_2 \ln r_2 + \bar{x}_2 - \bar{x}_1 + \bar{y} (\theta_1 - \theta_2) \tag{2.43}$$

$$\psi_j^{\gamma^-}(x, y) = \left[(\bar{x}_1 + \bar{x}_2) \psi_j^{\gamma^+} + r_2^2 \ln r_2 - r_1^2 \ln r_1 + \frac{1}{2} (\bar{x}_1^2 - \bar{x}_2^2) \bar{y} \right] \frac{1}{\bar{x}_1 - \bar{x}_2} \tag{2.44}$$

$$\psi_j^\lambda(x, y) = \bar{x}_2 \theta_2 - \bar{x}_1 \theta_1 + \bar{y} \ln \frac{r_1}{r_2} \tag{2.45}$$

Onde os termos \bar{x} e \bar{y} são coordenadas locais a cada painel, como mostrado na Fig. 19. Finalmente, fazendo com que a função de corrente $\psi(x, y)$ seja igual a uma função de corrente constante ψ_0 , referente à linha de corrente contornando a superfície do aerofólio, a Eq. 2.40 torna-se

$$\sum_{j=1}^N a_{ij} \gamma_j - \psi_0 = -u_\infty y_i + v_\infty x_i - \sum_{j=1}^{N+N_p-1} b_{ij} \lambda_j \quad ; \quad 1 \leq i \leq N \tag{2.46}$$

Onde as matrizes de coeficientes a_{ij} e b_{ij} são definidas a partir das funções unitárias de corrente. A condição de Kutta é cumprida com

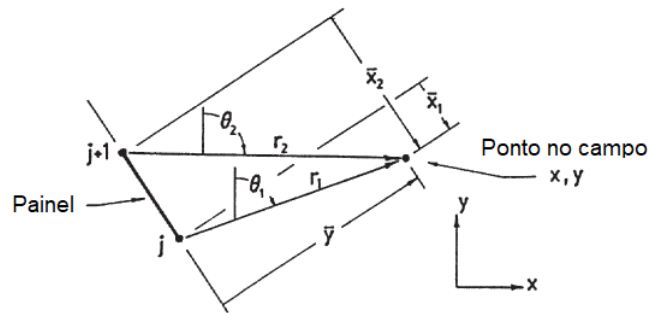


Figura 19 – Coordenadas locais dos painéis (DRELA, 1989)

$$\gamma_1 + \gamma_N = 0 \quad (2.47)$$

As Equações 2.46 e 2.47, assim, formam um sistema linear $(N + 1) \times (N + 1)$, com as incógnitas sendo a função de corrente ψ_0 e as intensidades de vórtice γ_i . Para a solução invíscida, os valores das intensidades de fonte λ_i são igualadas a zero.

Devido à base teórica fundamental do método de painéis, seu funcionamento sempre é, tecnicamente, invíscido. A consideração de efeitos viscosos do escoamento é inserida por meio de formulações adicionais. No caso do XFOIL, essa formulação é composta pelas equações padrões de momento compressível integral e de parâmetros de formato de energia cinética. Esse conjunto de equações é capaz de distinguir entre camadas limites laminares e turbulentas e estimar locais de transição. Mais detalhes a respeito dessa formulação são dados em (DRELA, 1989) e (DRELA; GILES, 1987).

2.6 Asas

Partindo para o domínio tridimensional, a asa toma o lugar como a estrutura aerodinâmica de interesse. Com a finalidade fundamental de proporcionar forças de sustentação à aeronave, as asas também devem cumprir outros requisitos que as fazem diferir completamente uma das outras à medida que se altera a função da aeronave a ser projetada.

Uma das características mais explícitas de uma dada asa é sua forma em planta, a qual permite a qualquer indivíduo, mesmo sem estudos em engenharia aeronáutica, deduzir o que a aeronave faz, a que velocidade ela voa, o quão rápido ela pode manobrar, entre outros aspectos. O formato mais simples é o retangular, de fácil construção. A seguir estão as asas trapezoidais ou afiladas, cujo comprimento de corda na raiz da asa é, geralmente, maior que o comprimento de corda na ponta da asa. Um exemplo de aeronave com asas afiladas é o EMB-314 Super Tucano. Enfatiza-se o uso da palavra "geralmente" na frase anterior porque existem exemplos de afilamentos inversos, como o Republic XF-

91. Para quantificar o afilamento, utiliza-se a razão de afilamento, definida como a corda da ponta dividida pela corda da raiz.

Mais exemplos de formatos comuns de planta de asa são as asas elípticas como as do Spitfire (Fig. 20) e asas em delta como as do Dassault Mirage. Asas em delta servem um propósito similar ao enflechamento. O enflechamento da asa se refere à angulação da asa em relação à fuselagem, de modo que as pontas sejam movidas em direção à cauda da aeronave ou, caso o enflechamento seja negativo, em direção ao nariz da aeronave. O propósito de asas em delta e asas enflechadas é de atrasar e enfraquecer a formação de ondas de choque na superfície da asa, as quais geram arrasto e, em casos extremos, tornam o voo completamente inviável. Por outro lado, essas geometrias reduzem a capacidade de geração de sustentação, o que torna essas aeronaves menos aptas para voo em baixas velocidades.



Figura 20 – Supermarine Spitfire e suas asas elípticas (CROWN, 2021)

Ainda tratando da forma da planta, existe ainda a razão de aspecto, definida como

$$AR = \frac{b^2}{S} \quad (2.48)$$

Onde b é a envergadura e S é a área alar. Em suma, a razão de aspecto, também denominada afilamento, é um valor grande quando a asa é longa e fina, e um valor pequeno quando a asa é curta e larga. Esse parâmetro aparentemente trivial contém em si informações fundamentais a respeito do desempenho de asas. Como mencionado anteriormente, o escoamento de ar sobre asas inclui uma componente na direção da envergadura, a qual flui das pontas para as raízes. O motivo para isso se dá porque o ar sob a asa tem pressão maior que o ar acima da asa, o que faz com que o ar procure atravessar para o lado de cima passando ao redor das pontas das asas. Essa passagem cria vórtices de ponta de asa que gera o fluxo de ar na direção da envergadura e esse fluxo, por sua vez, perturba os escoamentos no resto da asa e também causa a deflexão descendente do ar,

chamado *downwash*. O *downwash* efetivamente faz com que a asa "sinta" um ângulo de ataque menor do que o ângulo de ataque geométrico, o que não apenas significa que a asa fica com um coeficiente de sustentação menor que o do aerofólio no mesmo ângulo de ataque, como também gera uma componente de sustentação na mesma direção e sentido oposto ao movimento da aeronave. Esse é o arrasto induzido, exclusivo ao domínio tridimensional. É exatamente por esse fenômeno que uma asa sempre tem pior desempenho aerodinâmico do que o aerofólio com o qual ela é construída.

O aumento da razão de aspecto abranda diretamente esse efeito. Desse modo, caso o requisito principal do projeto de uma aeronave seja, por exemplo, altas razões de planeio ou voos de resistência, deve-se priorizar eficiência aerodinâmica, o que torna torna asas de alta razão de aspecto a escolha incontestável. Por isso o formato característico das asas de planadores e de UAVs de reconhecimento como o United 40 (Fig. 21a), capaz de voar missões de até 120 horas. Por outro lado, asas de baixa razão de aspecto são aptas a voo de alta velocidade, em regime transônico e acima, como o X-15 (Fig. 21b), projetado para alcançar Mach 7. Baixas razões de aspecto também são preferíveis caso um dos requisitos de projeto seja a resposta rápida a comandos de rolagem.



(a) Yabhon United 40 e suas asas de alta razão de aspecto em configuração tandem (ZIA-DEH, 2021)



(b) North American X-15 e suas asas de baixa razão de aspecto (LIASCH, 2018)

Figura 21 – Exemplos de tipos diferentes de razão de aspecto

A magnitude do arrasto induzido está diretamente associada também à distribuição de sustentação da asa. Tendo em vista que o arrasto induzido é causado pela geração de sustentação, há, para cada asa, uma distribuição de sustentação que supre exatamente as necessidades da aeronave, sem excessos, que causarão ineficiências. A distribuição de sustentação também é importante na perspectiva da estrutura da aeronave, pois sustentação excessiva na ponta de uma asa gera momentos fletores de grande magnitude na raiz. A distribuição ideal tem formato elíptico, o qual pode ser obtido, justamente, pela asa elíptica, como exemplificado na Fig. 22.

No entanto, além ter fabricação complexa, a asa elíptica e, por sua vez, a distribuição elíptica de sustentação, têm uma desvantagem gravíssima - para esse tipo de

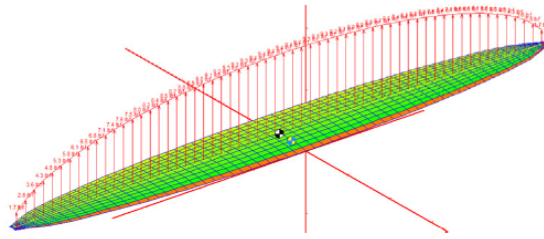


Figura 22 – Uma asa elíptica gerando uma distribuição elíptica de sustentação (GUDMUNDSSON, 2014)

distribuição, a asa tem o mesmo coeficiente de sustentação ao longo de toda a envergadura, o que significa que a asa completa perde a sustentação de uma vez quando alcançadas as condições de estol. O contraste a isso é a asa retangular, que, apesar de menos eficiente aerodinamicamente, tem comportamento amigável no estol, que começa na raiz da asa. Assim, com o escoamento ainda não separado nas pontas das asas, o piloto tem controle de rolagem. Desse modo, a esmagadora maioria dos projetistas preza por asas trapezoidais, as quais podem ter suas distribuições aproximadas à ideal, e aplicam metodologias para o bom comportamento no estol. Exemplos são a torção aerodinâmica, mencionada na Seção 2.4, e a torção geométrica ou *washout*, que consiste em torcer as pontas das asas de modo que tenham ângulos de incidência negativos em relação à raiz. O uso de *winglets* também é muito adotado para restringir os efeitos do arrasto induzido. *Winglets*, no entanto, são outro ponto de projeto com seus respectivos *trade-offs*: apesar de diminuir o arrasto induzido, eles também adicionam peso e arrasto parasita à aeronave.

Outros parâmetros relevantes à geometria de asas são, por exemplo, o diedro, que é a inclinação das asas em relação ao solo, afetando diretamente a estabilidade lateral da aeronave. O ângulo de incidência da asa em relação à fuselagem pode ser não nulo para impôr à asa ângulos de ataque mais propícios à sua *performance*. E isso tudo sem entrar no âmbito dos dispositivos hipersustentadores, que têm o propósito de aumentar a capacidade de geração de sustentação de aerofólios e asas. O projeto de flapes e slats é tão amplo que justificaria um estudo de otimização completamente distinto.

Ignorando por um momento os aerofólios, a modelagem da planta das asas é menos dependente de métodos como o CST devido ao fato de poder ser definida em termos de parâmetros gerais como envergadura, comprimentos de cordas, área alar, entre outros. Projetos de plantas de asas podem ser iniciados tendo em base geometrias simples, como asas retangulares e trapezoidais, e depois são incluídos refinamentos na geometria como curvaturas dos contornos da planta. Essa é a estratégia sugerida em (RAYMER, 2018), e a Fig. 23 é um exemplo. A geometria básica dessa asa pode ser definida em termos de apenas quatro parâmetros: corda da raiz, corda da ponta, envergadura e enflechamento (outras combinações de parâmetros envolvendo área alar, afilamento, razão de aspecto e outros

também poderiam ser utilizadas). Nota-se também que o CST é apto primariamente para a geração de geometrias que sejam utilizadas como seções transversais, portanto, para traçar curvas na planta da asa, é mais apropriado utilizar métodos como curvas de Bézier ou *B-splines*.

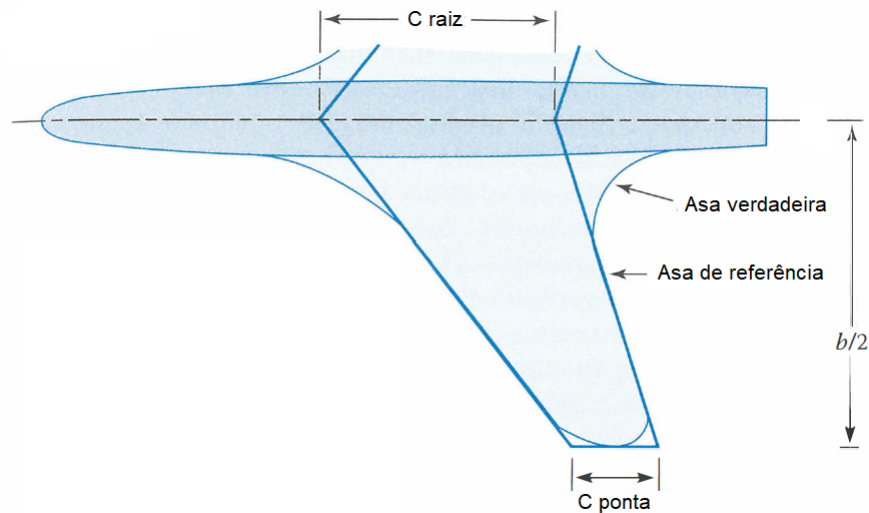


Figura 23 – Projeto de uma asa baseado em uma asa de referência, com geometria mais simples (RAYMER, 2018)

Para os algoritmos deste trabalho, as asas serão geradas com essa filosofia de *design* preliminar da planta em mente. As plantas serão geradas como retangulares, trapezoidais simples e trapezoidais duplas, geometrias que são comuns em muitas aeronaves e servem como boas aproximações para plantas mais complexas. Os aerofólios das asas serão gerados utilizando as parametrizações apresentadas anteriormente - a série NACA 4 dígitos e o CST.

2.7 APAME

O APAME (*Aircraft Panel Method*, Método de Painéis de Aeronaves) é um solucionador de escoamento potencial que utiliza o método de painéis 3D para realizar simulações aerodinâmicas. Desenvolvido como um trabalho de graduação (FILKOVIC, 2008), o APAME é uma alternativa a códigos mais pesados computacionalmente, e foi concebido também com o propósito de ser aplicável a problemas de otimização, o que pode ser averiguado explorando os *scripts* incluídos com o programa.

A geometria a ser simulada é transformada em uma malha de painéis que podem ter formato retangular ou triangular, cujos pontos de controle são posicionados no centro dos painéis. A equação básica para a solução do escoamento é

$$\sum_{i=1}^O a_i \mu_i + \sum_{i=1}^O b_i \sigma_i = 0 \quad (2.49)$$

Onde O é o número de painéis, μ_i são intensidades de dipolos, σ_i são intensidades de fontes, a_i são coeficientes de influência de dipolos e b_i são coeficientes de influência de fontes. As intensidades dos dipolos e das fontes são constantes em cada painel. Não existem singularidades de vórtice no APAME, ao contrário do XFOIL. A geração de sustentação no APAME é simulada por meio da distribuição contínua de dipolos contínuos, os quais podem se comportar como anéis de vórtice. Adicionalmente, superfícies sustentadoras possuem, em seus bordos fuga, painéis de trilha, os quais têm o propósito principal de impor a condição de Kutta (vide Fig. 24). Por esse motivo, não é possível estimar os coeficientes de sustentação de fuselagens ou corpos similares.

A eq. 2.49 é, então, reescrita na forma de um sistema de equações algébricas lineares:

$$\begin{bmatrix} a_{11} & a_{12} & \dots & a_{1O} \\ a_{21} & a_{22} & \dots & a_{2O} \\ \vdots & \vdots & \ddots & \vdots \\ a_{O1} & a_{O2} & \dots & a_{OO} \end{bmatrix} \begin{bmatrix} \mu_1 \\ \mu_2 \\ \vdots \\ \mu_O \end{bmatrix} = - \begin{bmatrix} b_{11} & b_{12} & \dots & b_{1O} \\ b_{21} & b_{22} & \dots & b_{2O} \\ \vdots & \vdots & \ddots & \vdots \\ b_{O1} & b_{O2} & \dots & b_{OO} \end{bmatrix} \begin{bmatrix} \sigma_1 \\ \sigma_2 \\ \vdots \\ \sigma_O \end{bmatrix} \quad (2.50)$$

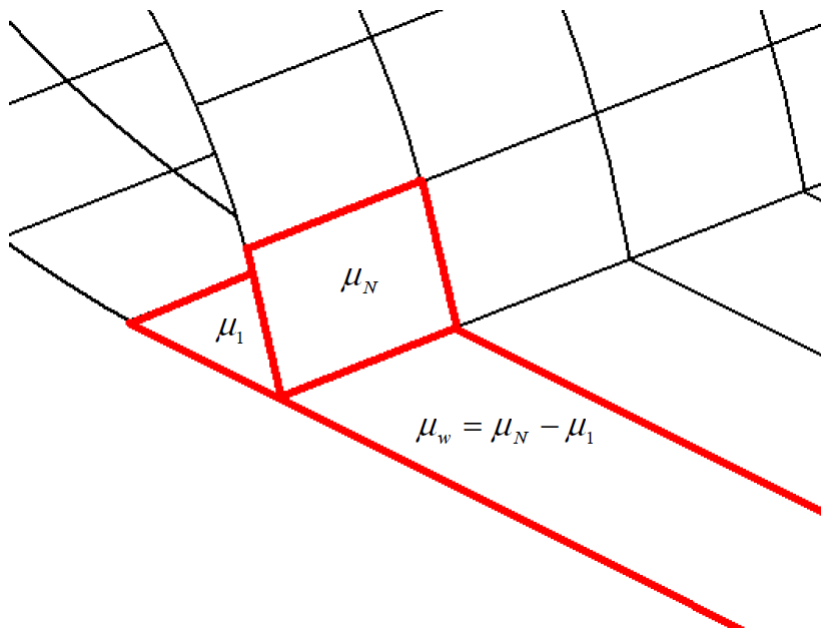


Figura 24 – Painéis de trilha em uma asa no APAME garantindo a condição de Kutta (FILKOVIC, 2008)

Aos valores de μ_i e σ_i são, à princípio, atribuídos valores unitários para o cálculo dos coeficientes de influência. Também sabendo que

$$\sigma_i = \vec{n}_i \cdot \mathbf{V}_\infty \quad (2.51)$$

Onde \vec{n}_i são os vetores normais dos painéis, e sabendo que b_i para os pontos de controle é igual a 0,5 quando analisados os painéis aos quais os mesmos pertencem, o lado direito da Eq. 2.50 torna-se definido, o que permite o cálculo de μ_i . Assim, pode-se determinar os dados relevantes para a simulação, como coeficientes adimensionais e distribuições de velocidade e pressão.

O APAME possui implementada a correção de Prandtl-Glauert, a qual contabiliza efeitos de compressibilidade em função do número de Mach (FILKOVIC, 2010). Ao contrário do XFOIL, não há correções implementadas para efeitos viscosos e, portanto, a estimativa de arrasto do APAME considera apenas arrasto induzido. Apesar disso, (FILKOVIC, 2008) mostra que grande parte dos demais resultados, como coeficientes de sustentação e momento, apresentam excelente correlação com simulações em CFD realizadas para validação do programa. Detalhes dessas validações, e mais detalhes sobre a formulação da solução do APAME podem ser encontradas em (FILKOVIC, 2008).

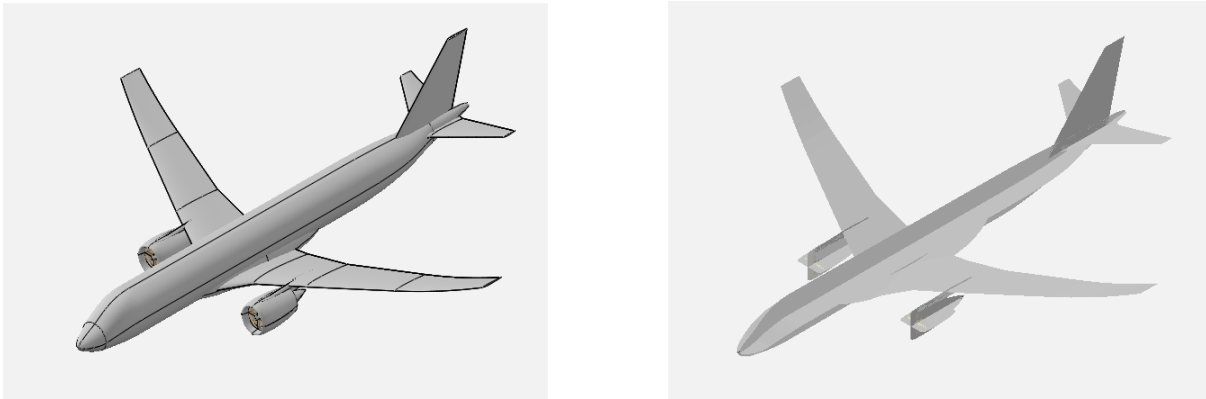
2.8 VSPAERO

O OpenVSP (*Open Vehicle Sketch Pad*, Bloco de Desenho Aberto de Veículos, em tradução livre) é um *software* desenvolvido pela NASA com funcionalidades para a modelagem paramétrica de aeronaves primariamente - o programa é, no entanto, versátil o suficiente para também permitir a modelagem de outros tipos de veículos. Originalmente de uso exclusivo da NASA, o OpenVSP tornou-se um projeto *Open Source* em 2012, recebendo atualizações desde então (MCDONALD; GLOUDEMANS, 2022). Entre as funcionalidades adicionais do OpenVSP que complementam a modelagem geométrica, existe o VSPAERO, um solucionador aerodinâmico de escoamento potencial, que pode fazer simulações usando método de painéis 3D e VLM. Neste trabalho, é usado o VSPAERO configurado com o VLM como uma demonstração da natureza modular da metodologia empregada neste trabalho, como será descrito em melhores detalhes no Cap. 3.

Utilizando a mesma teoria fundamental apresentada na Subseção 2.3.4, o VSPAERO é capaz de modelar também superfícies não sustentadoras, superfícies de controle e rotores. Um exemplo de conversão de um modelo tridimensional para a geometria VLM é apresentada na Fig. 25. Além de possuir correções para compressibilidade, o VLM do VSPAERO é também capaz de calcular arrasto viscoso, ao contrário do método de painéis 3D no mesmo programa. Os detalhes da teoria implementada no VSPAERO para ambos os métodos de solução, infelizmente, ainda não está disponível para o público² - no

² Pelo menos não em forma de documentação. Existe a possibilidade de investigar o código fonte e deduzir a teoria, mas isso está além do escopo deste trabalho.

momento da escrita deste trabalho há um manual de teoria sendo escrito, sem data prevista para publicação ([FERNANDES; MCDONALD, 2022](#)). Por hora, informação básica sobre o funcionamento do VSPAERO pode se encontrada nos *workshops* anuais, como ([KINNEY, 2020](#)).



(a) Modelo original

(b) Geometria VLM

Figura 25 – Geometrias do E190-E2 no OpenVSP ([ACHARD, 2019](#))

O OpenVSP (incluindo o VSPAERO) estão disponíveis também na forma de APIs em MATLAB e Python, as quais permitem gerar geometrias no formato OpenVSP e realizar simulações das mesmas. Para este trabalho, é feita a integração de uma das versões do algoritmo de asas em Python com a API do OpenVSP para gerar as geometrias dos indivíduos do algoritmo.

3 Metodologia

Este capítulo apresenta o funcionamento dos algoritmos desenvolvidos neste trabalho. Como demonstrado na Introdução, os algoritmos genéticos são de tal versatilidade que torna-se possível sua aplicação em variados tipos de problema. Independentemente do problema de otimização, no entanto, o funcionamento básico de todo algoritmo genético consiste no processo exposto na Seção 2.2 e resumido na Fig. 3. O presente Capítulo do texto explica como o problema de otimização de aerofólios e asas foi adaptado em algoritmos genéticos, apresentando as implementações referentes de cada um dos operadores genéticos, além de outros aspectos relevantes nos códigos. Os códigos em si estão disponíveis no GitHub: (FERNANDES, 2022).

Para fins de esclarecimento, os algoritmos em Python exigem os seguintes pacotes:

- *numpy*, importado como *np*, para a manipulação de vetores e incorporação de outras funções similares àquelas de Octave/MATLAB;
- *structure*, importado de *ypstruct*, para a inclusão de *structs* similares aos de Octave/MATLAB;
- *random* para o uso de geradores de números pseudo aleatórios, utilizados em diversos estágios dentro dos algoritmos;
- *time* para aferir os tempos de execução dos algoritmos;
- *matplotlib.pyplot*, importado como *plt*, para traçar gráficos e figuras;
- *openvsp* (a API no OpenVSP no Python), importado como *vsp*, para geração de geometrias de asas;
- *os* para manipular funções do sistema operacional, como apagar arquivos e convocar *prompts* de comando;
- *math* para utilizar funções matemáticas não disponíveis no Python base, e
- *locale* para alterar padrões como o caractere usado como separador decimal.

As especificações do computador usado para executar os algoritmos são as seguintes:

- CPU Intel Core i7-7500U
- Memória RAM 16 GB

- GPU NVIDIA GeForce 940MX
- Sistema operacional Windows 10 Home

É importante enfatizar que os requisitos de sistema podem ser mais baixos do que os apresentados acima, porque todo *software* utilizado é razoavelmente leve. A descrição de requisitos mínimos e/ou recomendáveis, no entanto, está fora do escopo deste trabalho. O sistema operacional também não necessariamente precisa ser o Windows - todo *software* utilizado neste trabalho também está disponível em Linux e macOS. De fato, a metodologia como um todo pode ser modificada e adaptada devido à sua natureza modular, como mostra a Fig. 26. Por exemplo, no caso dos algoritmos de otimização de aerofólios deste trabalho, o sistema operacional é o Windows 10, a plataforma de código é o Octave ou o Python, o algoritmo de otimização é o algoritmo genético, o simulador é o XFOIL e o objeto de estudo são os aerofólios. Adiante, no caso do algoritmo de otimização de asas, o objeto de estudo torna-se, justamente, as asas, e o simulador também deverá mudar, pois o XFOIL não é compatível com o caso tridimensional. Em uma outra situação hipotética, distinta, o usuário poderia optar em trocar apenas o algoritmo de otimização, ou a plataforma de código.

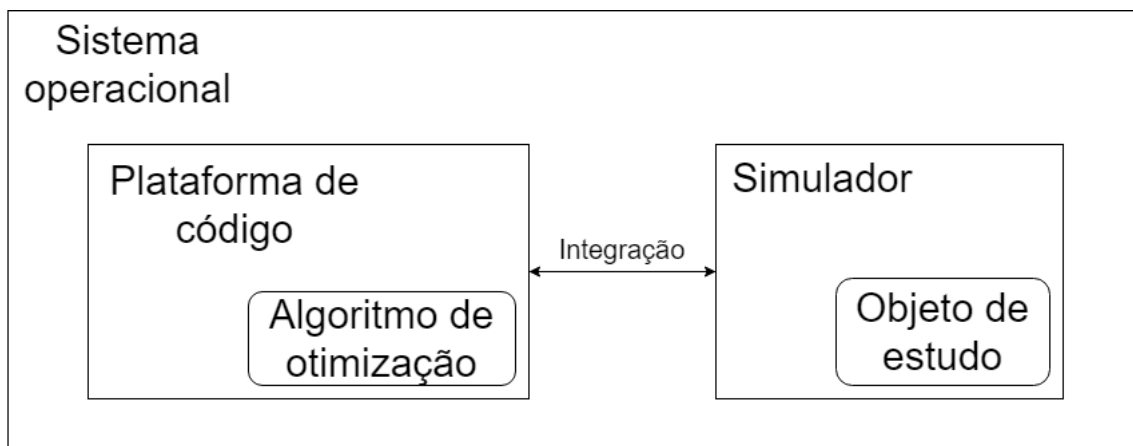


Figura 26 – Representação das ferramentas na forma de módulos (fonte: autor)

Daqui em diante, os nomes dos algoritmos seguirão a nomenclatura apresentada na Tab. 1. Tal nomenclatura faz referência a quatro características: o domínio geométrico (bidimensional ou tridimensional), qual dos dois métodos de parametrização geométrica é utilizado, o objeto a ser otimizado, e o espaço de busca, que pode ser livre (dentro de uma extensão de valores especificados pelo usuário) ou contido próximo a uma geometria única que se deseja otimizar.

Nome do algoritmo	Domínio	Objeto de estudo	Parametrização dos aerofólios	Espaço de busca
Algoritmo NACA 4 dígitos	2D	Aerofólios	Série NACA 4 dígitos	Livre
Algoritmo CST	2D	Aerofólios	CST	Livre
Algoritmo CST específico	2D	Aerofólios	CST	Baseado em uma geometria inicial
Algoritmo NACA 4 dígitos 3D	3D	Asas e seus aerofólios	Série NACA 4 dígitos	Livre
Algoritmo CST 3D	3D	Asas e seus aerofólios	CST	Livre
Algoritmo CST 3D específico	3D	Asas e seus aerofólios	CST	Baseado em uma geometria inicial

Tabela 1 – Nomenclatura dos algoritmos deste trabalho (fonte: autor)

3.1 Estrutura do algoritmo NACA 4 dígitos

3.1.1 Representação dos aerofólios dentro dos algoritmos

Antes de expor o funcionamento detalhado dos algoritmos é preciso explicar como um indivíduo-aerofólio dentro do algoritmo é representado dentro dos códigos.

A população no algoritmo é representada por um *struct* de tamanho N , onde N é o número de indivíduos. Cada indivíduo dentro desse *struct*, nomeado *pop*, tem informações sobre sua aptidão, suas características aerodinâmicas e suas especificações geométricas. Sua aptidão é salva no campo *score* e suas características aerodinâmicas são salvas no campo *aero*, o qual consiste em uma matriz $(dat.cases) \times 4$, onde *dat* é o *struct* de parâmetros de configuração dos algoritmos, e *dat.cases* é o número de condições de voo a serem consideradas no processo de otimização. Cada linha da matriz, da esquerda para a direita, contém quatro valores obtidos das simulações aerodinâmicas: o coeficiente de sustentação, o coeficiente de arrasto, a razão L/D e o coeficiente de momento. Sendo assim, a matriz *aero* pode ser compreendida como¹

$$pop.aero = \begin{bmatrix} C_{L,1}; C_{D,1}; L/D1; C_{M,1} \\ C_{L,2}; C_{D,2}; L/D2; C_{M,2} \\ \vdots \\ C_{L,dat.cases}; C_{D,dat.cases}; L/D_{dat.cases}; C_{M,dat.cases} \end{bmatrix} \quad (3.1)$$

Como visto na Subseção 2.4.1, um perfil NACA 4 dígitos é representado por três variáveis: m , p e t . Essas variáveis são adicionadas como campos no *struct* *pop* para

¹ O caractere usado como separador dos elementos dentro de um vetor é a vírgula em ambos o Octave e o Python. No entanto, neste texto, para evitar enganos com o separador de casas decimais, que também é a vírgula, todos os separadores de elementos serão substituídos pelo ponto e vírgula, como visto acima na definição do vetor *aero*. Os separadores não sofrem alterações nos *scripts* de código.

especificar a geometria de cada indivíduo. Todos os três valores são sempre números inteiros devido à padronização dos nomes dos aerofólios. O uso de números racionais também poderia ser adotado, pois as Equações 2.24 e 2.25 também são compatíveis com esse conjunto de números. Isso permitiria a ampliação do espaço de busca do algoritmo e a especificação mais detalhada dos perfis, mas também exigiria que o código gerasse as coordenadas e as importasse para o XFOIL, similarmente ao que é feito com os perfis CST.

Para o algoritmo NACA 4 dígitos, portanto, o *struct pop* contém os seguintes campos:

- m : curvatura máxima em porcentagem da corda ($m \in \mathbb{N}$);
- p : posição da curvatura máxima em décimos da corda ($p \in \mathbb{N}$);
- t : valor da espessura máxima em porcentagem da corda ($t \in \mathbb{N}$);
- $aero$: matriz com dados aerodinâmicos do aerofólio ($C_L, C_D, L/D, C_m \in \mathbb{R}, C_D > 0$), como visto na Eq. 3.1;
- $score$: aptidão do indivíduo em relação à função objetivo medida como um valor numérico ($score \in \mathbb{R}$).

3.1.2 Geração da população inicial

A geração de todos os indivíduos é realizada de forma completamente aleatória dentro de intervalos especificados. Esses intervalos podem variar de acordo com as intenções do usuário e, similarmente, serão configuradas de modos diferentes ao longo deste trabalho. Para os futuros testes de parâmetros, em específico, as extensões são as seguintes:

- m : definido de 0 a 9;
- p : definido de 0 a 9;
- t : definido de 10 a 30.

Portanto, o algoritmo NACA 4 dígitos lida com 3 variáveis. Uma restrição imposta neste estágio é referente aos perfis simétricos: caso o valor de m de um perfil seja igual a zero, o algoritmo atribui automaticamente também o valor zero ao seu campo p .

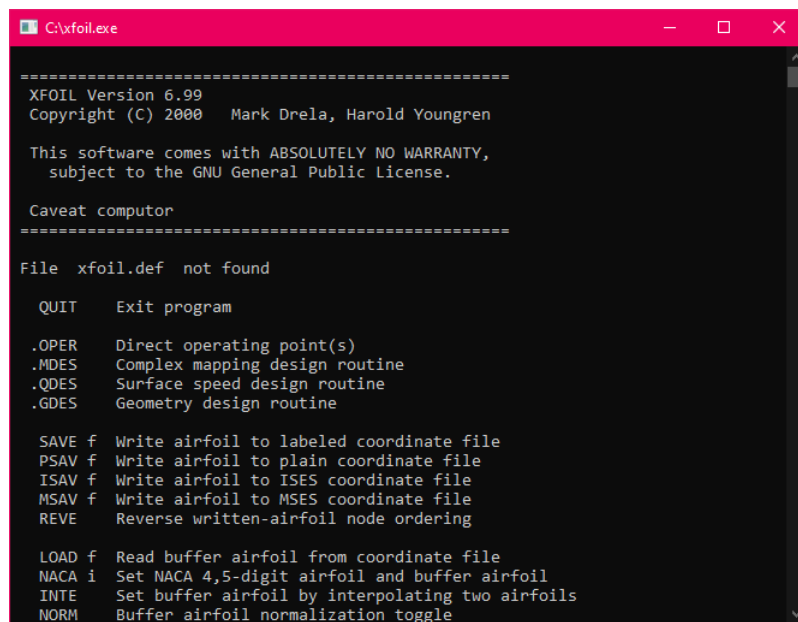
A extensão de t completa possível seria de 1 a 99. No entanto, a faixa especificada acima foi escolhida para priorizar geometrias mais realistas e diminuir o espaço de busca. Inclusive, na verdade, mesmo quando a extensão de t é a completa, o próprio algoritmo

tende a preferir aerofólios com espessuras dentro da extensão definida acima. É importante notar também que as extensões de m e p contêm valores que não são comumente utilizados em aerofólios reais (de 5 a 9), mas tais valores foram mantidos para experimentar que tipos de resultados o algoritmo pode obter.

3.1.3 Avaliação

Neste estágio o algoritmo convoca o XFOIL para simular os aerofólios. O XFOIL é um *software* baseado em *prompt* de comando, no qual o usuário insere as instruções desejadas. A Fig. 27 apresenta parte da tela inicial do XFOIL, ilustrando o cabeçalho e parte da lista de comandos. No final dessa lista, cortado da figura, está o local para digitar os comandos. É essa arquitetura do programa que permite que ele seja aberto por processos externos como, justamente, os algoritmos genéticos deste trabalho.

O XFOIL é executado uma vez para cada indivíduo na população. A sequência de comandos de XFOIL utilizada pelo algoritmo para simular um aerofólio-indivíduo i é descrita a seguir.



```
=====  
XFOIL Version 6.99  
Copyright (C) 2000 Mark Drela, Harold Youngren  
  
This software comes with ABSOLUTELY NO WARRANTY,  
subject to the GNU General Public License.  
  
Caveat computer  
=====  
File xfoil.def not found  
  
QUIT Exit program  
  
.OPER Direct operating point(s)  
.MDES Complex mapping design routine  
.QDES Surface speed design routine  
.GDES Geometry design routine  
  
SAVE f Write airfoil to labeled coordinate file  
PSAV f Write airfoil to plain coordinate file  
ISAV f Write airfoil to ISES coordinate file  
MSAV f Write airfoil to MSES coordinate file  
REVE Reverse written-airfoil node ordering  
  
LOAD f Read buffer airfoil from coordinate file  
NACA i Set NACA 4,5-digit airfoil and buffer airfoil  
INTE Set buffer airfoil by interpolating two airfoils  
NORM Buffer airfoil normalization toggle
```

Figura 27 – Tela inicial do XFOIL V 6.99 (fonte: autor)

Primeiro, é acessado o menu de opções de gráficos com o comando *PLOP* e a exibição de gráficos é desabilitada inserindo a letra *G*. A vantagem de desativar os gráficos do XFOIL durante seu uso nos algoritmos é impedir a aparição de telas *pop-up* a cada execução, o que é repetido para cada indivíduo na população, que pode ter tamanho de centenas, senão milhares de indivíduos. Sem essa configuração, o computador no qual o algoritmo é executado torna-se essencialmente inutilizável devido à aparição constante de janelas sobrepondo-se ao que quer que o usuário esteja utilizando. Terminado esse

passo, retorna-se ao menu principal do XFOIL apertando a tecla *enter*, a qual também é utilizada para sair dos outros menus.

A seguir começam os preparos para a análise propriamente dita do aerofólio. Primeiro, especifica-se o aerofólio por meio do comando *NACA NNNN*, onde *NNNN*, o nome do aerofólio, é retirado a partir dos campos *m*, *p* e *t* do indivíduo *i*. Nota-se, portanto, que o XFOIL tem um gerador embutido de aerofólios NACA 4 dígitos, o que torna desnecessária a geração de coordenadas para serem importadas para o programa, ao contrário do que será visto adiante no algoritmo CST.

Com o aerofólio definido, pode-se iniciar a configuração para realizar sua simulação. O primeiro comando dado é *OPER*, que abre o ambiente de análise direta do XFOIL. A seguir aciona-se o modo de análise viscosa por meio do comando *VISC Re*, onde *Re* é o número de Reynolds especificado pelo usuário. O comando *PACC* é usado a seguir para que o XFOIL salve um arquivo de polar, contendo os resultados das simulações, que serão lidos em seguida pelo algoritmo. No comando *PACC* também especifica-se o nome dessa polar, salvo como um arquivo de texto, que, neste trabalho, tem o nome "polar[n].txt", onde "[n]" é substituído pelo número da condição de voo referente - sendo assim, é feita uma simulação para cada condição de voo. Em seguida, se for da intenção do usuário, especifica-se o número de iterações com o comando *ITER I*, onde *I* é o número de iterações, e enfim é feita a simulação com o comando *ALFA a*, onde *a* é o ângulo de ataque especificado.

Após o final da simulação, os arquivos de polar são salvos, e o XFOIL é fechado. O algoritmo abre e lê os arquivos de polar. Caso a simulação tenha convergido, o algoritmo obtém os valores de C_L , C_D , L/D e C_m , os quais são inseridos, nessa ordem, no vetor *aero* do indivíduo *i*. Em caso de não convergência, o vetor *aero* é substituído por um único caractere *n*, o qual comunica ao algoritmo que esse aerofólio não poderá ser considerado nas atribuições de aptidão. Para o caso de múltiplas condições de voo, todas as simulações devem convergir, ou seja, se sequer uma simulação não convergir, é retornado o caractere *n*. Esse processo é repetido para todos os *N* indivíduos da população.

Todo esse conjunto de comandos é inserido no XFOIL por meio de um arquivo de texto nomeado "xfoil_input.txt", cujo modelo está apresentado no Apêndice I.1. O comando que abre o XFOIL especificando esse arquivo de *input*, no Octave, é

```
system('xfoil.exe < xfoil_input.txt');
```

E em Python é

```
os.system('xfoil.exe < xfoil_input.txt')
```

Isso exige que o XFOIL esteja localizado na mesma pasta que os *scripts* dos algoritmos.

3.1.4 Atribuição de aptidão

Todos os coeficientes aerodinâmicos salvos nas matrizes *aero* dos indivíduos podem ser utilizados como funções objetivo. Suas configurações são descritas a seguir.

Como mencionado brevemente na subseção 3.1.1, os algoritmos têm um outro *struct*, nomeado *dat*, que contém os parâmetros de configuração dos algoritmos e das simulações no XFOIL. Um dos campos desse *struct* é a matriz de caracteres *coeff_op*, com tamanho $(dat.cases) \times 4$, da qual cada casa é definida com um dos cinco caracteres seguintes:

- 'o': Buscar por um valor definido do coeficiente especificado (válido para todos os coeficientes);
- '^': Buscar valor máximo (C_L e L/D) ou mínimo (C_D) do coeficiente especificado;
- 'c': buscar valor constante arbitrário de C_M ao longo de múltiplas condições de voo;
- 'k': buscar valor constante especificado de C_M ao longo de múltiplas condições de voo;
- '!': Não usar na função objetivo.

Há também a matriz *coeff_val*, também de tamanho $(dat.cases) \times 4$, a qual especifica os valores dos coeficientes quando ao menos uma das configurações é definida como 'o'. As casas das matrizes *coeff_op* e *coeff_val* relacionam-se diretamente às casas das matrizes *aero*. Sendo assim, por exemplo, se o usuário configurasse a matriz *coeff_op* como

$$dat.coeff_op = \begin{bmatrix} 'o' ; '^' ; '!' ; 'k' \\ '!' ; '!' ; '!' ; '!' \end{bmatrix}$$

E a matriz *coeff_val* como

$$dat.coeff_val = \begin{bmatrix} 0,5 ; 0 ; 0 ; -0,05 \\ 0 ; 0 ; 0 ; 0 \end{bmatrix}$$

Trataria-se de um estudo de caso com duas condições de voo distintas. O algoritmo buscaria por um aerofólio com o coeficiente de sustentação igual a 0,5, com o menor arrasto possível, e que possua um coeficiente de momento constante, igual a $-0,05$, em ambas

as condições de voo. Os valores da segunda à última casa da primeira linha da matriz *coeff_val* são irrelevantes, justamente porque nas respectivas casas do vetor *coeff_op* foi especificada a opção '^', que não exige a definição de um valor alvo, ou foi especificado '!', o que ignora o uso daqueles respectivos coeficientes aerodinâmicos como forma de avaliação (neste exemplo, o L/D e o C_m).

Há ainda a matriz *coeff_F*, que especifica pesos para os requisitos e as funções objetivo. Novamente, tem tamanho $(dat.cases) \times 4$, onde cada casa corresponde às casas do vetor *aero* dos indivíduos. Os valores padrões dos pesos são 1 para todos os estudos de caso dos algoritmo deste trabalho, exceto quando especificado. Assim, cada um dos parâmetros da função objetivo é igualmente importante no processo de otimização.

Tendo em vista como as funções objetivo são configuradas pelo usuário, agora apresenta-se como o algoritmo avalia os indivíduos em base das mesmas. As aptidões são representadas em termos de valores numéricos salvos ao campo *score* do *struct pop*. Quanto maior o valor, mais apto é o indivíduo e vice-versa.

3.1.4.1 Alcançar um valor específico (opção 'o')

Assume-se um coeficiente C_i , referente ao indivíduo número i , e um coeficiente alvo C_{tgt} , sendo esses coeficientes qualquer um dos representados na matriz *aero*.

Se o coeficiente alvo for positivo:

$$\text{Apt}_{o', C_{tgt} > 0} = \begin{cases} \frac{C_{tgt}}{C_i}, & \text{se } C_i \geq C_{tgt} \\ \frac{C_i}{C_{tgt}}, & \text{se } C_i < C_{tgt} \end{cases} \quad (3.2)$$

Se o coeficiente alvo for negativo:

$$\text{Apt}_{o', C_{tgt} < 0} = \begin{cases} \frac{C_{tgt}}{C_i}, & \text{se } C_i \leq C_{tgt} \\ \frac{C_i}{C_{tgt}}, & \text{se } C_i > C_{tgt} \end{cases} \quad (3.3)$$

A distinção entre os casos de coeficiente alvo positivo e negativo é necessária para garantir que as pontuações de indivíduos com boa aptidão sejam sempre positivas.

Se o coeficiente alvo for nulo:

$$\text{Apt}_{o', C_{tgt} = 0} = \left(1 - \frac{C_i}{|C|_{max}} \right) \quad (3.4)$$

Onde $|C|_{max}$ é o valor máximo dos módulos dos coeficientes de todos os indivíduos.

As equações anteriores são válidas a qualquer um dos coeficientes da matriz *aero*. A exceção é para o coeficiente de arrasto: o algoritmo não permite que o C_D alvo seja menor ou igual a zero porque isso é fisicamente impossível.

É relevante notar que o algoritmo não impede a configuração de funções objetivo que levem a qualidades incomuns, como, por exemplo, coeficientes de sustentação negativos e coeficientes de momento positivos, os quais implicam em aerofólios com curvatura negativa. Averiguar as respostas dos algoritmos a essas condições seria um experimento interessante, que, no entanto, está fora do escopo deste trabalho.

3.1.4.2 Alcançar um valor máximo/mínimo (opção '^')

Quando aplicado ao coeficiente de sustentação e à razão L/D , a opção '^' configura o algoritmo para procurar aerofólios com os valores máximos dos mesmos.

Ao contrário do caso anterior (opção 'o'), não é possível fazer uma comparação entre o coeficiente do indivíduo e o coeficiente alvo, simplesmente porque não é possível especificar qual seria o maior coeficiente possível nas dadas condições de voo. Desse modo, a aptidão é calculada em termos do maior coeficiente encontrado na população completa, denominado C_{max} . Para o coeficiente de sustentação e a razão L/D , a aptidão é dada por

$$\text{Apt}_{\wedge, C_L \text{ e } L/D} = \frac{C_i}{C_{max}} \quad (3.5)$$

Para o caso do arrasto, o objetivo é alterado para a minimização. Isso é feito com

$$\text{Apt}_{\wedge, C_D} = \left(1 - \frac{C_{D,i}}{C_{D,max}} \right) \quad (3.6)$$

3.1.4.3 Alcançar um mesmo valor em múltiplas condições de voo (opções 'c' e 'k')

Ambas as opções 'c' e 'k' são específicas para o coeficiente de momento, e funcionam utilizando as mesmas equações apresentadas na Subsubseção 3.1.4.1. O valor do coeficiente alvo C_{tgt} é igual ao especificado em *coeff_val(1;4)* quando é utilizada a opção 'c' e igual à média dos coeficientes de cada condição de voo quando é utilizada a opção 'k'. Com isso, obtém-se uma pontuação para cada condição de voo. Como o intuito é retornar um único valor, é feita uma média aritmética das aptidões encontradas.

Nota-se, portanto, com as equações apresentadas nesta Subseção, que a pontuação máxima de cada um dos parâmetros da função objetivo será sempre igual a 1. Isso garante que todas as funções contribuirão igualmente à aptidão do indivíduo quando todos os pesos são unitários. A aptidão total do indivíduo, representada pelo valor no campo *score*, é simplesmente a soma das aptidões avaliadas para cada coeficiente:

$$score = \sum_{i=1}^{dat.cases} Apt_{C_L,i} F_{C_L,i} + Apt_{C_D,i} F_{C_D,i} + Apt_{L/D,i} F_{L/D,i} + Apt_{C_M,i} F_{C_M,i} \quad (3.7)$$

Onde $F_{C_L,i}$, $F_{C_D,i}$, $F_{L/D,i}$ e $F_{C_M,i}$, especificados no vetor $coeff_F$, são os pesos para o C_L , C_D , L/D e C_M respectivamente, em cada condição de voo.

3.1.5 Seleção para reprodução

Todas as reproduções no algoritmo NACA 4 dígitos são feitas com dois indivíduos pais que geram, por meio da troca de seus genes, dois indivíduos filhos. A seleção dos pais é realizada por meio de uma seleção por roleta. A Fig. 28 apresenta um exemplo com seis indivíduos, cada um com sua respectiva aptidão. Aqueles mais aptos são representados por regiões maiores na roda, o que significa que os mesmos têm mais chances de serem selecionados. As chances de seleção de indivíduos menos aptos, similarmente, são menores, apesar de não nulas. Isso permite que esses indivíduos possam ainda passar seus genes adiante, os quais poderiam colaborar com a busca pelo ótimo global quando combinados com outros genes.

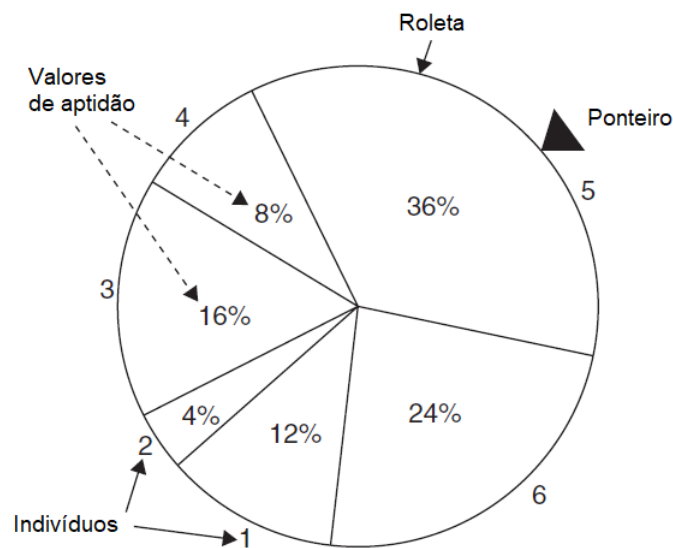


Figura 28 – Ilustração de uma seleção em roleta (RAO, 2009)

A implementação desse mecanismo em código é feito em base da soma cumulativa das aptidões de todos os indivíduos na população, que é salva em um vetor c . É gerado um número aleatório r , definido entre 0 e a soma de todas as aptidões. O número r define a seleção: o indivíduo selecionado é aquele referente ao índice do primeiro número no vetor c que é maior que r , lendo da esquerda para a direita.

Assim, tomando o exemplo da Fig. 28, o vetor c seria

$$c = [0,12; 0,16; 0,32; 0,40; 0,76; 1,00]$$

Se, por acaso, $r = 0,7$, o indivíduo selecionado seria o quinto, porque, lendo o vetor c da esquerda para a direita, o quinto número é o primeiro número maior que r .

3.1.6 Crossover

Tendo o par de pais selecionados, é gerado o par de filhos por meio do *crossover* dos genes dos pais. Os filhos são representados por outro *struct*, nomeado *chi*, que tem os mesmos campos que o *struct* *pop*. Para fins de explanação, o par de pais é representado pelos indivíduos $pop(i)$ e $pop(i+1)$ e o par de filhos é representado pelos indivíduos $chi(c)$ e $chi(c+1)$, onde i e c são índices arbitrários. Os genes trocados são as três variáveis de *design* dos aerofólios: os números m , p e t .

O algoritmo NACA 4 dígitos tem três tipos diferentes de *crossover*, fundamentalmente similares. O primeiro tipo consiste em fazer uma troca das variáveis t de cada um dos pais. O segundo tipo troca as variáveis p de maneira similar e o terceiro troca as variáveis m , também como no primeiro tipo. Os três tipos de *crossover* são ilustrados nas Figuras 29, 30 e 31 respectivamente.

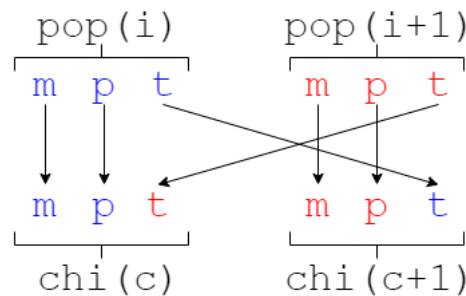


Figura 29 – Primeiro tipo de *crossover* no algoritmo NACA 4 dígitos (fonte: autor)

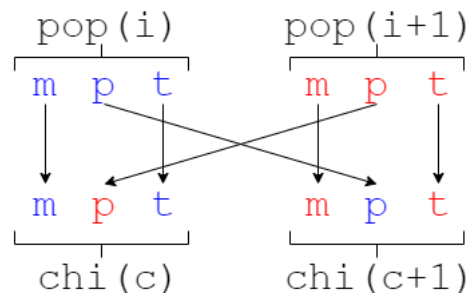


Figura 30 – Segundo tipo de *crossover* no algoritmo NACA 4 dígitos (fonte: autor)

Como visto na Subseção 3.1.2, se um perfil NACA 4 dígitos tem $m = 0$, ele obrigatoriamente deve ter $p = 0$. Portanto, quando ao menos um dos pais é um perfil simétrico, o único tipo de *crossover* permitido é o primeiro. Para perfis assimétricos, a designação do tipo de *crossover* é feita de forma aleatória, onde cada um dos três tipos tem chance igual de ser selecionado.

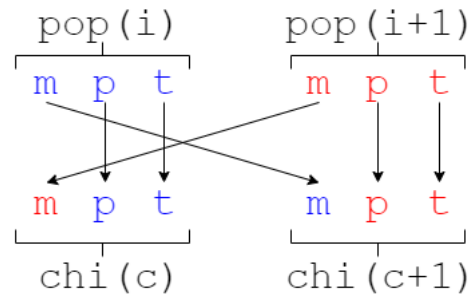


Figura 31 – Terceiro tipo de *crossover* no algoritmo NACA 4 dígitos (fonte: autor)

3.1.7 Mutação

A mutação altera apenas os genes da população de filhos, os quais serão passados para a iteração seguinte. Os indivíduos que passam por mutação são escolhidos de forma aleatória, em termos de uma chance de mutação definida no campo *mu* do *struct dat*. Após a seleção, os tipos de mutação são ditados, primeiramente, pela classificação do aerofólio em termos de sua simetria ou assimetria. Os detalhes serão apresentados a seguir.

3.1.7.1 Mutação de perfis simétricos ($m = 0$ e $p = 0$)

Para perfis simétricos podem ser aplicados dois tipos de mutação diferentes. O primeiro tipo mantém a sua espessura e o transforma em um aerofólio assimétrico, atribuindo ao mesmo valores aleatórios de m e p , ambos retirados de um intervalo de 1 a 9. O segundo tipo de mutação o mantém como simétrico e altera sua espessura, atribuindo aleatoriamente um novo valor para t baseado no intervalo 10 a 30 - o mesmo da geração da população inicial, como visto na Subseção 3.1.2.

A seleção do tipo de mutação é feito de forma aleatória, onde cada tipo tem chance igual de ser selecionado.

3.1.7.2 Mutação de perfis assimétricos ($m \neq 0$ e $p \neq 0$)

Quatro formas de mutação existem para perfis assimétricos no algoritmo, dentre as quais as três primeiras mantêm essa característica de assimetria. O primeiro tipo de mutação altera apenas a variável m , selecionando aleatoriamente um novo valor dentro do intervalo de 1 a 9. O segundo altera apenas a variável p da mesma forma feita no primeiro tipo de mutação. O terceiro tipo seleciona um novo valor aleatório para t dentro do intervalo 10 a 30. O quarto e último tipo de mutação de perfis assimétricos os transforma em perfis simétricos, mantendo sua espessura e alterando seus valores de ambos m e p para zero.

Similarmente ao caso de perfis simétricos, a designação do tipo de mutação para os perfis assimétricos não tem vieses, tendo cada tipo uma chance igual de ser selecionado aleatoriamente.

3.1.8 Aplicação de elitismo

No algoritmo NACA 4 dígitos, o elitismo funciona de modo a manter o melhor indivíduo presente em uma iteração. Ao final de cada iteração i , após o processo de mutação, o primeiro indivíduo da nova população é substituído pelo indivíduo mais apto avaliado na iteração atual. A opção de ativação do elitismo é incluída no campo *elite* do *struct dat*. Quando *elite*=1, o elitismo é ativado.

Este é o último estágio no *loop* principal do algoritmo. Havendo ou não atuação do mecanismo de elitismo, a população de pais é substituída pela população de filhos passando as informações do *struct chi* para o *struct pop*, e então o algoritmo segue a partir da avaliação dos indivíduos, como apresentada na Subseção 3.1.3. Todos os demais algoritmos deste trabalho empregam o mecanismo de elitismo da mesma forma e no mesmo local do *loop* principal, portanto, nas seguintes Seções tratando das estruturas dos outros algoritmos, as Subseções sobre o mecanismo de elitismo serão suprimidas para evitar redundância.

3.2 Estrutura do algoritmo CST

3.2.1 Representação dos aerofólios dentro do algoritmo

A representação dos perfis CST tem de ser menos rígida em comparação à dos perfis NACA 4 dígitos devido ao fato de que o grau do polinômio de Bernstein pode ser alterado, o que por sua vez altera o número de variáveis de *design*, como visto na Subseção 2.4.2. Para representação dos perfis CST em código são utilizados vetores $1 \times (BPn + 2)$, onde BPn é a variável nos códigos que representa o grau do polinômio de Bernstein ($BPn > 1$). Genericamente, portanto, um vetor de variáveis de design CST seria definido como

$$v = [R_{LE}; A_1; A_2; \dots; A_{n-1}; \beta; \Delta Z_{TE}]$$

Neste trabalho, como padrão, o algoritmo CST é configurado com o grau polinômio de Bernstein fixo em 4 em todos os casos, exceto quando especificado de outra forma. Assim, o algoritmo CST lida com 12 variáveis de *design*, ou $(dat.BPn + 2) \cdot 2$ para outros valores do polinômio de Bernstein. Para descrever o extradorso e o intradorso dos aerofólios, são utilizados os seguintes vetores:

$$v_{ex} = [R_{LE;ex}; A_{1;ex}; A_{2;ex}; A_{3;ex}; \beta_{ex}; \Delta Z_{TE;ex}]$$

$$v_{in} = [R_{LE;in}; A_{1;in}; A_{2;in}; A_{3;in}; \beta_{in}; \Delta Z_{TE;in}]$$

Para o algoritmo CST, portanto, o *struct pop* contém os seguintes campos:

- v_ex : vetor com as informações do extradorso ($v_ex \in \mathbb{R}$);
- v_in : vetor com as informações do intradorso ($v_in \in \mathbb{R}$);
- $symm$: valor especificando se o aerofólio é simétrico ($symm = 1$) ou assimétrico ($symm = 0$);
- $aero$: matriz com dados aerodinâmicos do aerofólio ($C_L, C_D, L/D, C_m \in \mathbb{R}, C_D > 0$), como visto na Eq. 3.1;
- $score$: aptidão do indivíduo em relação à função objetivo medida como um valor numérico ($score \in \mathbb{R}$).

Para o caso de perfis NACA 4 dígitos, o estabelecimento de perfis simétricos é simples porque, apesar de ser um processo aleatório, basta que a variável m receba o valor zero, o que não é uma ocorrência rara. No entanto, para perfis CST, a geração de perfis simétricos contando apenas com a geração aleatória dos valores de ambos os vetores torna a ocorrência de perfis simétricos muito rara. Desse modo, para o algoritmo CST, há incluída a opção de especificar se dado aerofólio é simétrico ou não utilizando-se o campo $symm$ do *struct pop*. Quando $symm = 1$, ao invés de gerar aleatoriamente os valores do vetor do intradorso, esses valores são tomados como exatamente iguais aos valores do vetor do extradorso. A proporção de perfis simétricos e assimétricos pode ser definida no algoritmo por meio do campo $dat.symm_op$. Quando $dat.symm_op = 1$, todos os aerofólios serão simétricos, quando $dat.symm_op = 0$, todos os aerofólios serão assimétricos, e valores intermediários alteram a proporção entre ambos os tipos (por exemplo: $dat.symm_op = 0,7$ fará com que cerca de 70 % da população seja composta de aerofólios simétricos, e cerca de 30 % seja composta de aerofólios assimétricos).

Ao contrário do que é recomendado em (KULFAN; BUSSOLETTI, 2006), o algoritmo permite o uso de raios de bordo de ataque diferentes com o intuito de permitir mais variação na geometria. Com essa configuração, é possível gerar mais tipos distintos de bordo de ataque, ao mesmo tempo também afetando diretamente o extradorso e o intradorso. A continuidade do contorno no bordo de ataque é garantida por meio da formatação dos pontos das coordenadas, que consiste em um contorno contínuo que segue do bordo de fuga do extradorso para o bordo de ataque e retorna ao bordo de fuga do intradorso. A formatação das coordenadas dos aerofólios CST é apresentada em maiores detalhes no Apêndice K.

As espessuras de bordo de fuga $\Delta Z_{TE,ex}$ e $\Delta Z_{TE,in}$ serão sempre nulas na versão básica do algoritmo CST, mas podem receber outros valores, como será apresentado na versão do algoritmo que otimiza aerofólios específicos. Detalhes de seu funcionamento serão apresentados na Seção 3.3, e a representação dos indivíduos naquele algoritmo é idêntica ao que foi apresentado nesta subseção.

3.2.2 Geração da população inicial

A geração inicial no algoritmo CST é feita como no algoritmo NACA 4 dígitos. Valores aleatórios, retirados de intervalos específicos, são atribuídos às variáveis dos indivíduos, como apresentado abaixo:

- R_{LE} : definido de 0,01 a 0,05 para o extradorso e de 0,01 a 0,03, ambos em passos de 0,005;
- A_r : definido de 0 a 0,3 para o extradorso e de 0 a 0,1 para o intradorso, ambos em passos de 0,025;
- β : definido de 5° a 20° para o extradorso e de $(20^\circ - \beta_{ex})$ a 20° para o intradorso, como números inteiros apenas.

Onde tais valores são adotados para os testes de parâmetros. A discretização dos intervalos dos raios do bordo de ataque e dos pesos intermediários tem o propósito de proporcionar boa variabilidade de geometrias sem entrar em detalhes extremamente minuciosos, o que permite melhor exploração do espaço de busca. Esse é o mesmo motivo pelo qual os ângulos do bordo de fuga são definidos apenas como números inteiros.

O lado inferior do intervalo para o ângulo do bordo de fuga do intradorso tem o intuito de evitar bordos de fuga demasiadamente finos. A definição é detalhada como

$$\beta_{ex} + \beta_{in} \geq L \quad (3.8)$$

Onde L é o ângulo mínimo de separação entre ambos os bordos de fuga, com o valor fixo de 20° . Manipulando essa expressão para isolar β_{in} , temos

$$\beta_{in} \geq 20^\circ - \beta_{ex} \quad (3.9)$$

Outra restrição imposta aos perfis CST trata da prevenção da geração de perfis com curvaturas negativas. Isso é realizado por meio dos pesos intermediários de ambas as superfícies do aerofólio. Por exemplo, a Fig. 32 apresenta um aerofólio arbitrário cujo extradorso e intradorso têm ambos os raios de bordo de ataque e ângulos de bordo de fuga iguais. No entanto, os pesos intermediários do extradorso são $A_{0,ex} = 0,1$, $A_{1,ex} = 0,1$ e $A_{2,ex} = 0,1$, enquanto que os pesos intermediários do intradorso são $A_{0,in} = 0,2$, $A_{1,in} = 0,2$ e $A_{2,in} = 0,1$. Essa configuração faz com que o aerofólio tenha curvatura negativa. Para garantir, portanto, que a curvatura seja sempre nula ou positiva, o algoritmo permite apenas a geração de aerofólios cuja soma dos pesos intermediários do extradorso seja maior ou igual à soma dos pesos intermediários do intradorso.

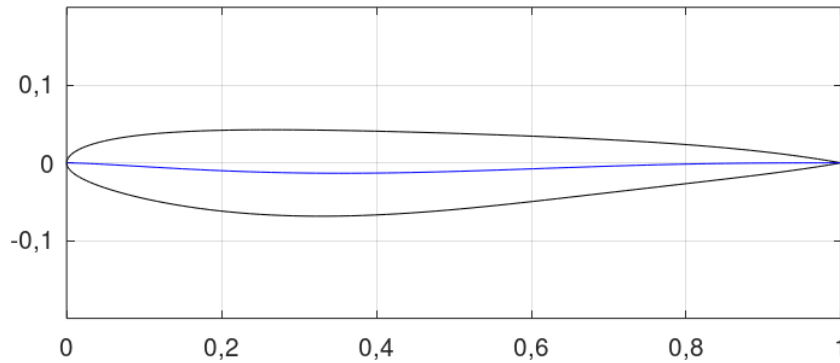


Figura 32 – Um aerofólio CST com curvatura negativa (fonte: autor)

Os perfis NACA 4 dígitos não requerem tratamentos especiais quando são gerados porque, mesmo que tenham características incomuns de espessura e de curvatura (por exemplo, espessuras muito grossas ou muito finas, ou valores de m e p iguais ou próximos a 9), as geometrias ainda possuem características que as fazem identificáveis como aerofólios. Os perfis CST não são assim devido à natureza aleatória da geração das variáveis de design de cada indivíduo - os contornos do extradorso e do intradorso podem ser gerados com características indesejáveis (ao ponto de a geometria sequer assemelhar-se a um aerofólio, mesmo que a função *class* esteja configurada para tal) ou até mesmo pode haver a interseção dos mesmos. As duas restrições expostas acima, dos ângulos de bordo de fuga e das somas dos pesos intermediários, não tratam desses problemas, e também não há uma forma de deduzir analiticamente as geometrias corretas dentro de todas as combinações possíveis de variáveis, visto que a atribuição dos valores é totalmente aleatória dentro dos intervalos especificados. Portanto, a geração dos perfis tem uma abordagem de tentativa e erro, baseada em uma função do algoritmo que faz checagens de qualidade nas geometrias. Para cada indivíduo, o algoritmo gera as características aleatoriamente, e então o aerofólio é avaliado com essa função. Caso ele passe, o algoritmo segue para a geração do próximo indivíduo. Caso ele não passe, o algoritmo gera novamente as suas variáveis de design. Os detalhes da checagem de qualidade do algoritmo CST estão descritos no Apêndice L.

3.2.3 Avaliação

O algoritmo CST usa a mesma estratégia apresentada na Subseção 3.1.3 para avaliar os indivíduos: a execução do XFOIL usando um arquivo de texto como entrada para instruções. O primeiro comando, novamente, desativa a aparição das janelas *pop-up* do XFOIL. Já o segundo passo, no entanto, é diferente para o caso do algoritmo CST. Para especificar o aerofólio a ser analisado, são carregadas coordenadas com o comando *LOAD n*, onde n é o nome do arquivo de coordenadas, salvo como um arquivo de texto formatado como apresentado no Apêndice K.

Todo o resto dos comandos necessários é igual ao caso do algoritmo NACA 4

dígitos, com a definição do arquivo de polar e a especificação do número de Reynolds e do ângulo de ataque. O modelo de arquivo de *input* para o algoritmo CST pode ser visto no Apêndice I.2.

3.2.4 Atribuição de aptidão e seleção para reprodução

Ambos esses estágios funcionam exatamente do mesmo modo que no algoritmo NACA 4 dígitos, como descrito, respectivamente, nas Subseções 3.1.4 e 3.1.5.

3.2.5 Crossover

A maior quantidade de variáveis de *design* dos perfis CST abre a possibilidade para mais tipos de *crossover*, o que permite uma variedade maior de combinações de genes da população. Todos os exemplos apresentados a seguir serão feitos com o CST configurado com o grau 4 do polinômio de Bernstein, que é o padrão adotado para o trabalho na maior parte dos algoritmos configurados com o CST, porém os mecanismos de *crossover* também são compatíveis com outros graus do polinômio.

Novamente, o *crossover* funciona de modo a gerar dois filhos $chi(c)$ e $chi(c+1)$ a partir dos genes de dois pais $pop(i)$ e $pop(i+1)$. Há vários tipos de *crossover* no algoritmo CST, sendo que a aplicação dos mesmos depende das classificações de simetria/assimetria dos aerofólios. Mais detalhes serão apresentados a seguir.

3.2.5.1 Ambos os perfis são assimétricos

O primeiro tipo de *crossover* para perfis assimétricos consiste em trocar inteiramente os vetores do extradorso e intradorso. Exclusivamente nesse tipo de *crossover* é feita uma checagem com o intuito de averiguar se o requisito de separação mínimo do bordo de fuga, apresentado na Subseção 3.2.2, é cumprido. Caso não seja, o ângulo do bordo de fuga do intradorso é alterado para o valor mínimo que cumpre o requisito. A decisão de alterar o intradorso ao contrário do extradorso se dá pelo fato que este último é mais decisivo na geração de sustentação (RAYMER, 2018).

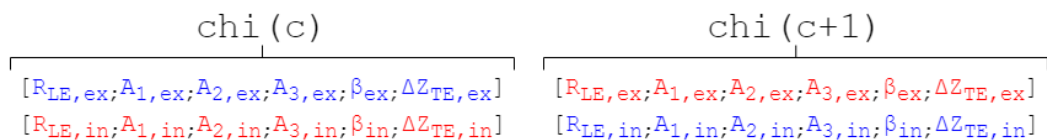


Figura 33 – Primeiro tipo de *crossover* no algoritmo CST (fonte: autor)

No segundo tipo de *crossover*, apenas os raios do bordo de ataque são trocados, como ilustrado na Fig. 34.

O terceiro tipo consiste em trocar apenas os pesos intermediários de ambas as superfícies. A Fig. 35 mostra esse procedimento.

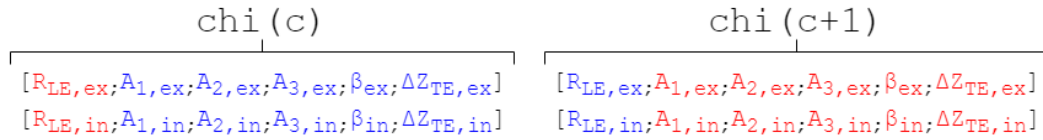


Figura 34 – Segundo tipo de *crossover* no algoritmo CST (fonte: autor)

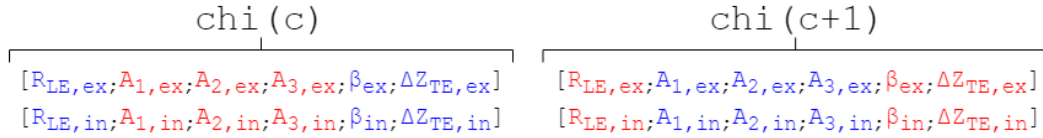


Figura 35 – Terceiro tipo de *crossover* no algoritmo CST (fonte: autor)

No quarto tipo também é feita a troca dos pesos intermediários, com a distinção de que são trocados alguns dos pesos ao invés de todos. Por exemplo, na Fig. 36, o indivíduo $chi(c)$ recebe, em seu extradorso, apenas o primeiro peso intermediário do indivíduo $pop(i+1)$, em contraste à Fig. 35, em que $chi(c)$ recebe todos os pesos intermediários de $pop(i+1)$. O mesmo vale para o vetor do intradorso. As quantidades de pesos intermediários que serão retiradas de cada um dos indivíduos pais é decidido de forma aleatória, portanto a Fig. 36 é apenas uma combinação possível. Adicionalmente, esse tipo de *crossover* não funciona quando o polinômio de Bernstein é configurado com grau 2. Nessa configuração, os vetores dos perfis têm apenas um peso intermediário, o que faz com que apenas o terceiro tipo de *crossover* faça sentido.

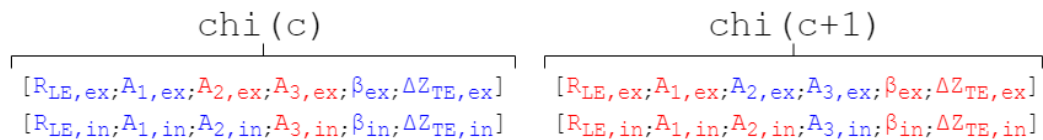


Figura 36 – Quarto tipo de *crossover* no algoritmo CST (fonte: autor)

O quinto e último tipo de *crossover* no algoritmo CST consiste em trocar os ângulos dos bordos de fuga dos perfis. A Fig. 37 ilustra esse procedimento.

A determinação do método de *crossover* a ser aplicado é realizada de forma aleatória.

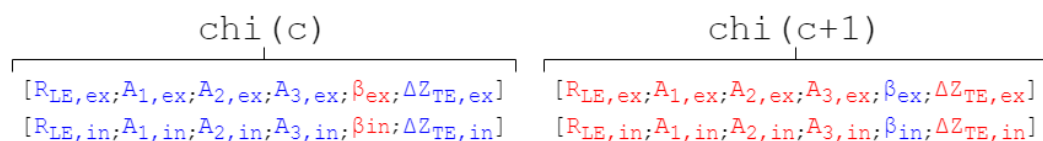


Figura 37 – Quinto tipo de *crossover* no algoritmo CST (fonte: autor)

3.2.5.2 Ambos os perfis são simétricos

Para perfis simétricos, são utilizados os mesmos esquemas de *crossover* utilizados para os perfis assimétricos, com a exceção do primeiro, que troca os vetores do extradorso e do intradorso completamente. Aerofólios simétricos têm ambos os seus vetores iguais, portanto, qualquer modificação feita no vetor do extradorso deve ser transmitida também ao vetor do intradorso. Por esse mesmo motivo, os esquemas de *crossover* têm ligeiras modificações de modo a ignorar modificações aos vetores de intradorso, mas ainda mantendo as devidas trocas para os vetores de extradorso.

3.2.5.3 Um perfil é simétrico e o outro é assimétrico

O último tipo de *crossover* consiste em transformar o aerofólio simétrico em um assimétrico e vice-versa. Assumindo, portanto, que o indivíduo $pop(i)$ seja o aerofólio simétrico e que o indivíduo $pop(i+1)$ seja o aerofólio assimétrico, o resultado deste *crossover* é um indivíduo $chi(c)$, cujos vetores de extradorso e intradorso são iguais ao vetor do extradorso de $pop(i+1)$, e um indivíduo $chi(c+1)$, que recebe o vetor do extradorso de $pop(i)$ e o vetor do intradorso de $pop(i+1)$.

3.2.6 Mutação

Novamente, o processo de mutação é realizado apenas em alguns dos indivíduos da população, os quais são selecionados em base da chance de mutação. Ao todo cinco tipos distintos de mutação podem ser aplicados aos perfis CST, dos quais todos os cinco são válidos para perfis assimétricos e três são válidos para perfis simétricos - o primeiro, o quarto e o quinto. Para perfis simétricos, as alterações são feitas apenas no extradorso e são copiadas para o intradorso.

O primeiro tipo altera apenas os raios do bordo de ataque. Pode ser atribuído um novo valor para ambas as superfícies ou para apenas uma delas. Independentemente, o novo valor é escolhido de forma aleatória dentro da mesma extensão usada no estágio de geração inicial (Subseção 3.2.2).

O segundo tipo de mutação altera os pesos intermediários do extradorso apenas, dentro de uma extensão próxima aos valores originais. Essa extensão tem seus limites inferior e superior definidos em termos do passo especificado para as extensões de valores na geração da população inicial. Desse modo, cada peso tem sua própria extensão de valores atribuída, e seus novos valores são escolhidos aleatoriamente de dentro dessas extensões. O terceiro e quarto tipo de mutação funcionam da mesma forma, mas, respectivamente, alterando os pesos intermediários apenas do intradorso, e alterando-os em ambas as superfícies. O intuito dessa abordagem, ao invés de usar as mesmas extensões definidas para a geração da população inicial, tem o intuito de evitar a perda exagerada de qualidade

do aerofólio, pois os pesos intermediários são os que mais têm potencial para isso devido aos seus maiores efeitos sobre as geometrias.

Enfim, o quinto e último modo de mutação altera os ângulos no bordo de fuga. O valor do ângulo no extradorso é escolhido aleatoriamente dentro da extensão de valores do estágio de geração da população inicial, e o valor do intradorso é atribuído de modo a ser a mínima separação permitida pelo requisito dos ângulos do bordo de fuga descrito na Subseção 3.2.1.

3.3 Estrutura do algoritmo CST específico

3.3.1 Representação dos aerofólios dentro do algoritmo

Todos os indivíduos dentro do algoritmo CST específico usam a mesma representação utilizada no algoritmo CST convencional, como descrito na Subseção 3.2.1.

3.3.2 Geração da população inicial

Ao contrário de ambos os algoritmos apresentados até agora, o algoritmo CST específico tem a finalidade de tomar uma geometria única de um aerofólio já existente e modificá-la moderadamente a fim de otimizá-la. Para que tais modificações sejam possíveis, deve-se expressar a geometria do aerofólio original em termos da parametrização CST. Isso é feito pelo CST reverso, cujo funcionamento é demonstrado no Apêndice M. Com esse passo cumprido, o algoritmo específico gera a população em termos do indivíduo original.

Para cada variável de *design* do aerofólio, especifica-se um valor máximo que pode ser subtraído da variável e um valor máximo que pode ser somado à variável. Ambos esses valores determinam a extensão de valores que essa variável pode tomar. Sendo assim, por exemplo, se o valor original de dada variável for 0,5, se o valor máximo de subtração for 0,1, e o valor máximo de adição também for 0,1, a extensão de valores possíveis para essa variável será de 0,4 a 0,6. Esse processo é realizado para cada uma das variáveis em ambos os vetores de extradorso e intradorso.

A configuração das extensões de valores do algoritmo CST específico para os testes de parâmetros é:

- R_{LE} : valor máximo de subtração igual a 0,01, valor máximo de adição igual a 0,01, valor mínimo possível igual a 0,005 para ambas as superfícies;
- A_r : valor máximo de subtração igual a 0,05, valor máximo de adição igual a 0,05 para ambas as superfícies;

- β : valor máximo de subtração igual a 5° , valor máximo de adição igual a 5° para o extradorso, valor máximo de subtração igual a $(L - \beta_{ex})$, valor máximo de subtração igual a 5° para o intradorso.

O valor mínimo possível dos raios do bordo de ataque são importantes para garantir que os mesmos sejam sempre positivos e não nulos. Para os ângulos do bordo de ataque, deve-se cumprir o requisito mínimo de separação do bordo de fuga, expressado pela Eq. 3.8. O ângulo mínimo de separação L para o algoritmo CST específico é definido como a soma dos ângulos do bordo de fuga de ambas as superfícies do aerofólio original e, portanto, seu valor muda de acordo com o aerofólio a ser otimizado. Outra distinção fundamental desta versão do algoritmo em comparação à convencional é que os intervalos definidos nas extensões dos valores das variáveis são contínuos ao invés de discretos. Isso permite uma modelagem ainda mais minuciosa da geometria para alcançar os objetivos dos processos de otimização, e não afeta negativamente a *performance* do algoritmo devido ao fato de que o espaço de busca se limita a geometrias bastante próximas àquelas especificadas.

Fora essas questões, o algoritmo CST específico também contém a checagem de curvaturas negativas e as demais checagens de qualidade das geometrias.

3.3.3 Avaliação, atribuição de aptidão, seleção para reprodução, *crossover* e mutação

Todos esses estágios funcionam da mesma forma que no algoritmo CST. Os detalhes desses respectivos estágios estão nas Subseções 3.2.3, 3.1.4, 3.1.5, 3.2.5 e 3.2.6. A única distinção na mutação se dá na atribuição dos novos valores de cada variável, que ocorre obedecendo a mesma lógica utilizada na geração da população inicial.

3.4 Estrutura do algoritmo NACA 4 dígitos 3D

3.4.1 Representação das asas dentro do algoritmo

A especificação da geometria de asas neste trabalho é realizada definindo-se suas seções transversais (os aerofólios) e a forma de sua planta. Neste algoritmo novamente é utilizada a série NACA 4 dígitos para a definição dos aerofólios.

A planta da asa, de forma simplificada, é uma geometria que pode ser completamente descrita em termos de poucos parâmetros. Neste trabalho, optou-se utilizar apenas os três seguintes tipos de planta: retangular, trapezoidal simples e trapezoidal dupla. Apesar de aparentemente limitante, essas geometrias são razoáveis para fins de otimização, pois são comuns em um grande número de aeronaves e são boas aproximações para outros formatos de asas mais complexos, como ilustrado na Fig. 23. As plantas de asas trape-

zoidais simples são definidas primariamente em termos de duas estações (raiz e ponta), enquanto que asas bitrapezoidais são definidas primariamente em termos de três estações (raiz, ponta e meio). A estação do meio, apesar do nome, não necessariamente fica posicionada precisamente no meio da asa, mas sempre se encontra entre a raiz e a ponta, e sua posição em relação à raiz é dada pela envergadura da primeira seção. Os aerofólios são definidos nessas estações. Para asas bitrapezoidais, define-se duas seções, uma da raiz ao meio e outra do meio à ponta.

O *struct pop* no algoritmo NACA 4 dígitos 3D é composto pelos seguintes campos:

- *type*: define se a asa será trapezoidal simples (*type*= 0) ou trapezoidal dupla (*type*= 1);
- *b*: envergadura completa ($b \in \mathbb{R}$);
- *b1*: Envergadura da primeira seção (asas trapezoidais duplas apenas) ($b1 \in \mathbb{R}$);
- *c_r*: corda da raiz ($c_r \in \mathbb{R}$);
- *c_m*: corda do meio (asas trapezoidais duplas apenas) ($c_m \in \mathbb{R}$);
- *c_t*: corda da ponta ($c_t \in \mathbb{R}$);
- *mac*: corda aerodinâmica média ($mac \in \mathbb{R}$);
- *S*: área alar ($S \in \mathbb{R}$);
- *sweep*: enflechamento (asas trapezoidais simples apenas) (*sweep* $\in \mathbb{R}$);
- *sweep1*: enflechamento da primeira seção (asas trapezoidais duplas apenas) (*sweep1* $\in \mathbb{R}$);
- *sweep2*: enflechamento da segunda seção (asas trapezoidais duplas apenas) (*sweep2* $\in \mathbb{R}$);
- *af_r*: vetor 1×3 que define o aerofólio da raiz ($af_r = [m; p; t]$) ($m, p, t \in \mathbb{N}$);
- *af_m*: vetor 1×3 que define o aerofólio do meio ($af_m = [m; p; t]$) (asas trapezoidais duplas apenas) ($m, p, t \in \mathbb{N}$);
- *af_t*: vetor 1×3 que define o aerofólio da ponta ($af_t = [m; p; t]$) ($m, p, t \in \mathbb{N}$);
- *tw_m*: torção geométrica da estação do meio (asas trapezoidais duplas apenas) ($tw_m \in \mathbb{Z}$);
- *tw_t*: torção geométrica da estação da ponta ($tw_t \in \mathbb{Z}$);
- *NODE*: matriz de nós da malha para o APAME (*NODE* $\in \mathbb{R}$);

- *PANEL*: matriz de painéis para o APAME ($PANEL \in \mathbb{Z}$);
- *sec_N*: número de seções transversais discretas na asa ($sec_N \in \mathbb{Z}$);
- *aero*: matriz com dados aerodinâmicos do aerofólio ($C_L, C_D, L/D, C_m \in \mathbb{R}, C_D > 0$), como visto na Eq. 3.1;
- *score*: aptidão do indivíduo em relação à função objetivo medida como um valor numérico ($score \in \mathbb{R}$).

Com essas informações, pode-se averiguar que asas trapezoidais simples têm 11 variáveis de *design* e asas trapezoidais duplas têm 17 variáveis de *design* neste algoritmo. Para fins de esclarecimento, *mac* e *S* não são variáveis de *design* porque são grandezas calculadas a partir da forma da planta para serem utilizadas como valores de referência nas simulações do APAME. Similarmente, *tw_m* não é uma variável de *design* porque sempre terá um valor calculado de modo a garantir uma distribuição linear da torção geométrica da ponta à raiz. Assim, por exemplo, se a torção geométrica na ponta é de -5° e sua envergadura da primeira seção *b1* é metade da envergadura completa *b*, a torção geométrica da estação do meio será calculada automaticamente e terá valor $-2,5^\circ$.

Não existe um tipo específico para definir asas retangulares, porque para uma asa ser retangular basta apenas que todas as suas cordas sejam iguais. Todos os campos que são exclusivos para determinado tipo de asa ficam vazios a não ser que o indivíduo seja uma asa daquele tipo (exemplo: o campo *sweep* de asas trapezoidais duplas fica vazio).

Existem requisitos de geometria impostos à planta de asa. O primeiro trata dos comprimentos de corda, para impedir a ocorrência de razões de aflamentos maiores que zero:

- $c_t \leq c_r$;
- $c_t \leq c_m$ (asas trapezoidais duplas apenas);
- $c_m \leq c_r$ (asas trapezoidais duplas apenas).

O segundo, exclusivo a asas trapezoidais duplas, refere-se à envergadura da primeira seção *b1*. Dependendo dos intervalos dos valores especificados para *b* e *b1*, é possível que a envergadura da primeira seção da asa fique igual ou maior que a envergadura completa, o que não tem sentido. Portanto, especifica-se um valor mínimo de separação entre *b* e *b1*, sempre positivo e não nulo, que garanta que $b1 < b$.

Apesar de haverem apenas dois aerofólios (ou três, dependendo do tipo da asa) definidos explicitamente pelos parâmetros geométricos, existe um número de aerofólios implícitos, localizados entre as estações. A existência desses aerofólios se dá pela natureza

da malha de painéis, e sua quantidade depende da densidade da malha utilizada. A Fig. 38 mostra a planta de uma asa trapezoidal simples arbitrária expressa em termos de suas seções transversais (ênfatiza-se que é uma vista do topo). Em preto são as seções localizadas nas estações da raiz e da ponta, e em vermelho são as seções localizadas entre as estações.

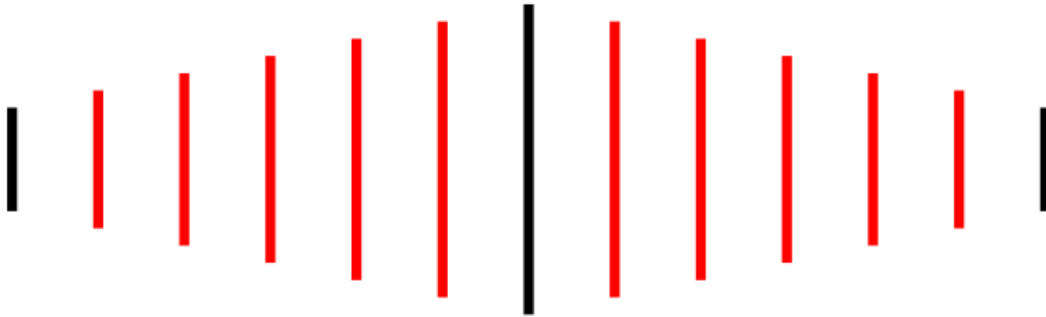


Figura 38 – Locais das seções transversais da planta de uma asa genérica (fonte: autor)

Esses aerofólios intermediários são gerados como interpolações lineares dos aerofólios localizados nas estações diretamente adjacentes. A Fig. 39 mostra um exemplo de interpolação, supondo que o perfil da raiz seja um NACA 4412, o perfil da ponta seja um NACA 0010, e o perfil interpolado esteja localizado precisamente na metade da semi-envergadura. O funcionamento da interpolação de aerofólios está detalhado no Apêndice N.

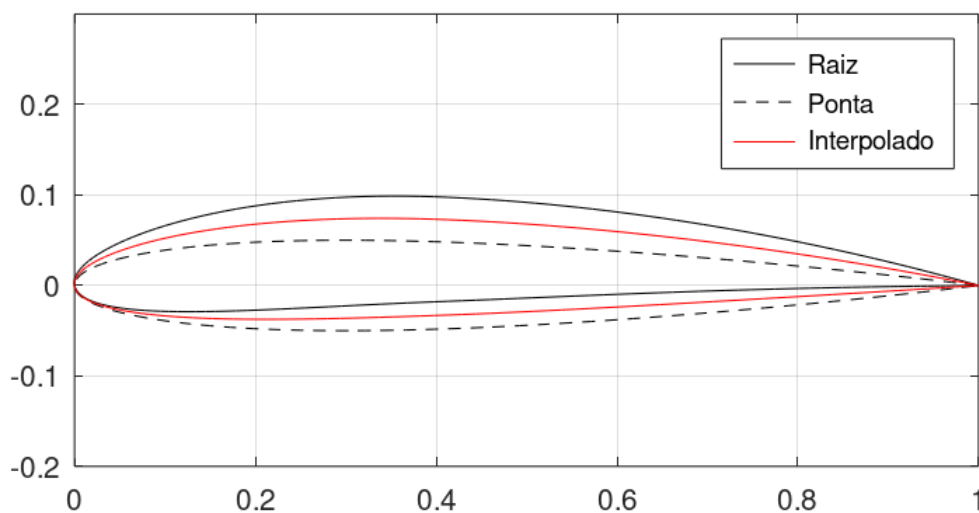


Figura 39 – Exemplo de interpolação de aerofólios (fonte: autor)

3.4.2 Geração da população inicial

Para os testes de parâmetros, os parâmetros geométricos das plantas das asas da população inicial são definidos em termos dos seguintes intervalos:

- b : definida de 10m a 15m em passos de 0,5m;
- $b1$: definida de 8m a 14m em passos de 0,5m, com uma separação mínima da ponta de 0,5m;
- c_r : definida de 1m a 2m em passos de 0,1m;
- c_m : definida de 0,5m a 2m em passos de 0,1m;
- c_t : definida de 0,5m a 2m em passos de 0,1m;
- $sweep$: definido de 0° a 15° em passos unitários;
- $sweep1$: definido de 0° a 15° em passos unitários;
- $sweep2$: definido de 0° a 15° em passos unitários.
- tw_t : definido de -5° a 0° em passos de $0,5^\circ$.

No *struct dat* deste algoritmo (e nos demais algoritmos de asas) há o campo *planf_op*, o qual permite especificar a proporção dos tipos de asas a serem tratadas no algoritmo. Esta configuração funciona da mesma forma que a opção *symm_op* do algoritmo CST, no sentido de que é possível escolher a proporção de tipos dependendo do valor escolhido, onde os extremos são $planf_op = 0$ para que todas as asas sejam trapezoidais simples e $planf_op = 1$ para que todas as asas sejam trapezoidais duplas. Para este algoritmo, a configuração escolhida, a não ser quando especificado de outra forma, é $planf_op = 0,5$ para que haja aproximadamente o mesmo número de asas de ambos os tipos na população.

Os parâmetros geométricos dos aerofólios das asas são definidos dentro dos seguintes intervalos:

- m : definido de 0 a 4 para todas as estações;
- p : definido de 0 a 4 para todas as estações;
- t : definido de 10 a 20 para todas as estações.

Como se pode ver, os intervalos são idênticos para aerofólios da raiz, do meio e da ponta. Isso, no entanto, foi uma escolha para este trabalho em específico. Caso deseje, o usuário pode especificar extensões distintas para cada estação. Nota-se também que os

intervalos de valores das variáveis dos perfis NACA são menores no algoritmo 3D do que no 2D. Essa diminuição foi realizada para compensar o aumento de variáveis de *design* da asa completa, além do fato de que os intervalos acima estão mais de acordo com aerofólios NACA 4 dígitos utilizados em aeronaves reais.

3.4.3 Avaliação

Para os algoritmos de asas, utiliza-se o APAME para realizar as simulações. Sendo mais exato, é utilizado o solucionador *.exe* avulso do programa, ao invés da versão GUI (*Graphical User Interface*, interface gráfica do usuário), o que permite que o programa seja convocado a partir dos algoritmos similarmente ao que é feito com o XFOIL. A única tela do solucionador avulso é a que está na Fig. 40. Como o próprio programa instrui, basta inserir o nome do arquivo de *input* da simulação, e os cálculos serão feitos.

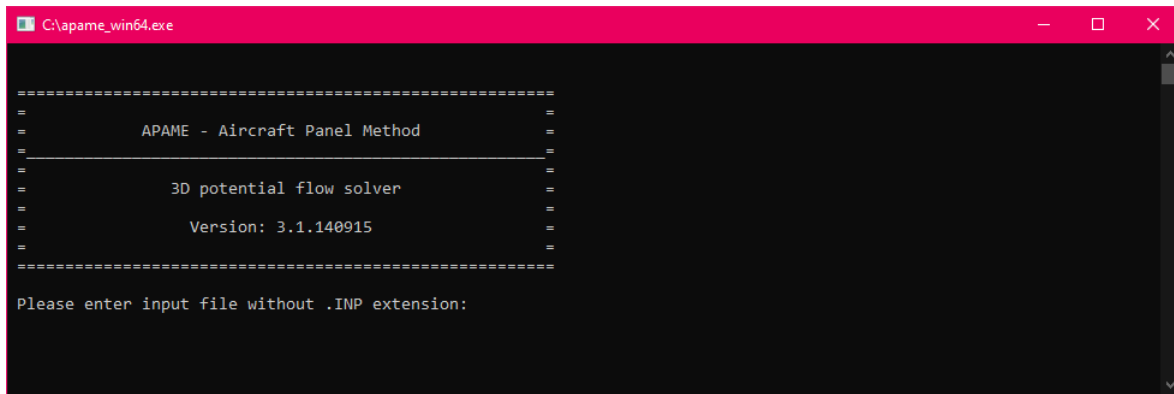


Figura 40 – Tela do solucionador avulso do APAME (fonte: autor)

No XFOIL, todas as configurações a respeito das geometrias dos aerofólios, das condições de voo e outros ajustes pertinentes às simulações eram realizados por meio da manipulação de menus. O APAME, em contraste, exige que o usuário insira um arquivo de entrada *.inp* já contendo todas as informações necessárias, as quais são, nessa ordem, dados sobre a condição de voo, valores de referência para a geometria, parâmetros do solucionador, opções dos resultados e a definição da geometria em si, definida em termos de uma malha de nós e painéis. O modelo desse arquivo de entrada está no Apêndice J. A edição e impressão do arquivo *.inp* é realizada automaticamente pelo algoritmo tendo em base todos os dados mencionados acima. O comando que executa o APAME, dando como entrada o arquivo *.inp*, é

```
system('apame_win64.exe < apame_cmd_input');
```

E em Python é

```
os.system('apame_win64.exe < apame_cmd_input')
```

Onde o arquivo "apame_cmd_input" é um arquivo de texto contendo o nome do arquivo .inp. A ausência da extensão de arquivo é apenas uma forma de sinalizar que esse arquivo em específico não precisa ser editado.

Após a simulação, o APAME imprime um arquivo de texto .log, no mesmo diretório onde os scripts do algoritmo e o executável do APAME estão salvos, contendo um breve registro do processo da simulação e, mais importante, os seus resultados. Os dados relevantes (os coeficientes de sustentação, arrasto e momento de arfagem) são lidos pelo algoritmo e são utilizados no passo seguinte.

Como nos algoritmos de aerofólios, também há a possibilidade de que a simulação não convirja. Novamente, nesses casos, o vetor *aero* é substituído pelo caractere *n* para indicar que esse indivíduo não poderá participar do resto do algoritmo devido à impossibilidade de avaliá-lo. Também como nos algoritmos de aerofólios, para estudos de caso com múltiplas condições de voo, é obrigatório que todas as simulações para um mesmo indivíduo converjam a fim de que o mesmo siga para os próximos estágios do algoritmo.

3.4.4 Atribuição de aptidão e seleção para reprodução

Como em todos os outros algoritmos apresentados até agora, a avaliação é realizada em termos de quatro coeficientes adimensionais distintos, com quatro configurações de função objetivo e restrições disponíveis para configurar o processo de otimização. Indo para o regime tridimensional, a diferença se dá na adição de duas restrições novas, referentes à força de sustentação, à força de arrasto, e ao momento de arfagem da asa. Assim, os algoritmos de asas têm as seguintes opções para configurar o processo de otimização:

- 'o': Buscar por um valor específico do coeficiente especificado (válido para todos os coeficientes);
- '^': Buscar valor máximo (C_L e L/D) ou mínimo (C_D) do coeficiente especificado;
- 'c': buscar valor constante arbitrário de C_M ao longo de múltiplas condições de voo;
- 'k': buscar valor constante especificado de C_M ao longo de múltiplas condições de voo;
- 'q': buscar valor específico da força de sustentação, fora de arrasto ou de momento de arfagem;
- '#': buscar valor máximo de força de sustentação ou mínimo de força de arrasto;
- '!': Não usar na função objetivo.

Onde, para a opção 'q', utiliza-se as equações da Subsubseção 3.1.4.1 para calcular as pontuações, e para a opção '#', utiliza-se as equações da Subsubseção 3.1.4.2.

A seleção para reprodução é a mesma que a utilizada nos algoritmos anteriores. Novamente, seus detalhes estão na Subseção 3.1.5.

3.4.5 Crossover

O *crossover* nos algoritmos de asas permitem a troca de múltiplos genes da planta e dos aerofólios ao mesmo tempo, e existem duas formas principais, as quais serão expostas no texto a seguir. Novamente, em todo *crossover*, para dois indivíduos pais selecionados $pop(i)$ e $pop(i+1)$ são gerados dois indivíduos filhos $chi(c)$ e $chi(c+1)$.

3.4.5.1 Crossovers gerais das características

Este tipo de *crossover* pode ser feito entre asas de tipos iguais e asas de tipos diferentes. Inicialmente, $chi(c)$ é copiado de $pop(i)$ e $chi(c+1)$ é copiado de $pop(i+1)$, e a seguir é feita a troca de genes específicos. As características a serem trocadas são definidas por meio de um vetor 1×12 , cujos elementos recebem, aleatoriamente, os valores 0 ou 1, onde aqueles com valor 1 indicam ao algoritmo quais genes deverão ser trocados. As casas 1 a 12 correspondem às respectivas características:

- Envergadura completa b ;
- Envergadura da primeira seção $b1$ (válido apenas se ambos os indivíduos forem asas trapezoidais duplas);
- Corda da raiz c_r ;
- Corda do meio c_m (válido apenas se ambos os indivíduos forem asas trapezoidais duplas);
- Corda da ponta c_t ;
- Aerofólio da raiz (vetor af_r);
- Aerofólio do meio (vetor af_m) (válido apenas se ambos os indivíduos forem asas trapezoidais duplas);
- Aerofólio da ponta (vetor af_t);
- Torção geométrica na ponta tw_t ;
- Enflechamento completo $sweep$ (válido apenas se ambos os indivíduos forem asas trapezoidais simples);
- Enflechamento da primeira seção $sweep1$ (válido apenas se ambos os indivíduos forem asas trapezoidais duplas);

- Enflechamento da segunda seção *sweep2* (válido apenas se ambos os indivíduos forem asas trapezoidais duplas).

Sendo assim, se o vetor de opção de *crossover*, depois de ter seus valores aleatoriamente atribuídos, ficar igual a

$$\text{cross_op_v} = [0;0;1;0;1;1;1;0;0;1;0;0]$$

Os genes a serem trocados serão a corda da raiz, a corda da ponta, o aerofólio da raiz, o aerofólio do meio, e o enflechamento completo. De acordo com os requisitos dos tipos de geometria apresentados nos itens acima, as trocas do aerofólio do meio e do enflechamento completo ocorrerão apenas, respectivamente, se ambas as asas forem trapezoidais duplas, e apenas se ambas forem trapezoidais simples. Eventos conflitantes como esse não são incomuns, mas são irrelevantes devido à natureza do processo de *crossover*, concebido precisamente para fornecer uma ampla gama de possibilidades de trocas de genes.

Os *crossovers* dos aerofólios podem ser realizados de modo a trocar completamente os aerofólios dos indivíduos, ou pode-se trocar características individuais. Seguindo o exemplo dado no parágrafo anterior, supõe-se que, como aerofólios da raiz, o indivíduo $chi(c)$ tem um NACA 1413 e o indivíduo $chi(c+1)$ tem um NACA 3215. Caso seja feita a troca dos aerofólios completos, o indivíduo $chi(c)$ recebe o NACA 3215 como aerofólio da raiz e o indivíduo $chi(c+1)$ recebe o NACA 1413. Nos demais tipos de *crossover* para os aerofólios, por outro lado, são adicionados mais alguns passos ao processo.

À princípio, ignora-se as asas e os aerofólios em mãos são tratados como no algoritmo 2D. Assim, é feita a troca de m , p ou t , onde a escolha de qual parâmetro será trocado é feita de forma aleatória. Supondo que t tenha sido o escolhido, o resultado são os perfis NACA 1415 e NACA 3213. Feito isso, basta retornar os aerofólios às asas, o que também é feito de forma aleatória - o perfil NACA 1415 será atribuído ao indivíduo $chi(c)$ e o perfil NACA 3213 será atribuído ao indivíduo $chi(c+1)$, ou o contrário, tendo cada uma dessas possibilidades 50% de chance de ocorrer. O processo deste exemplo vale para qualquer outro dos três parâmetros dos aerofólios NACA 4 dígitos, e é aplicado também para *crossovers* dos aerofólios do meio e da ponta.

3.4.5.2 Trocar os tipos de asas

Este tipo de *crossover* pode ser realizado apenas em asas que sejam de tipos distintos. Novamente, é feita uma cópia do indivíduo $pop(i)$ para o indivíduo $chi(c)$, e do indivíduo $pop(i+1)$ para o $chi(c+1)$.

Como exemplo, supõe-se que à princípio $chi(c)$ seja uma asa trapezoidal simples e $chi(c+1)$ seja uma asa trapezoidal dupla. As trocas e alterações realizadas são as seguintes:

- O indivíduo $chi(c)$ recebe de $chi(c+1)$ os valores dos campos $b1$, c_m , af_m , tw_m , $sweep1$ e $sweep2$;
- O indivíduo $chi(c+1)$ têm os campos $b1$, c_m , tw_m , $sweep1$ e $sweep2$ transformados em variáveis vazias ([] no Octave e *None* no Python) e o campo af_m transformado em um vetor de zeros. Também recebe o valor de $sweep$ do indivíduo $chi(c)$.

Independentemente do tipo de *crossover*, os requisitos de geometria da planta detalhados na Subseção 3.4.1 são impostos a todas as asas geradas durante as trocas de genes para garantir suas devidas características geométricas da planta.

3.4.6 Mutação

Novamente, os indivíduos que sofrerão mutação são escolhidos aleatoriamente em base da chance de mutação especificada pelo usuário. Quanto às operações em si, a mutação no algoritmo NACA 4 dígitos 3D funciona de modo a atribuir novos valores às variáveis. A planta e os aerofólios podem ser afetados, e a seleção do que exatamente será alterado é feita da mesma forma que é feita a escolha dos genes a serem trocados no estágio de *crossover* - estabelece-se um vetor 1×12 composto unicamente de zeros e uns, onde cada elemento recebe o valor de forma aleatória. Os campos com valor unitário indicam quais genes deverão sofrer mutação. Com os genes definidos, o algoritmo reatribui um novo valor a cada um deles, sempre respeitando as extensões de valores escolhidas pelo usuário.

Como no estágio de *croosver*, todas as operações de mutação são finalizadas com checagens da geometria da planta para cumprir os requisitos de comprimentos de corda e de separação das estações da ponta e do meio.

3.5 Estrutura do algoritmo CST 3D

3.5.1 Representação das asas dentro do algoritmo

As plantas das asas são tratadas da mesma forma que no algoritmo NACA 4 dígitos 3D, portanto, a única diferença é a definição dos aerofólios nas estações da raiz, meio e ponta. Como nos algoritmos CST 2D, a representação dos aerofólios se baseia na definição de um vetor com variáveis que descrevem o extradorso e um vetor com variáveis que descrevem o intradorso. Sendo assim, no algoritmo CST 3D, o *struct pop* é composto pelos seguintes campos:

- *type*: define se a asa será trapezoidal simples (*type*= 0) ou trapezoidal dupla (*type*= 1);

- b : envergadura completa ($b \in \mathbb{R}$);
- $b1$: Envergadura da primeira seção (asas trapezoidais duplas apenas) ($b1 \in \mathbb{R}$);
- c_r : corda da raiz ($c_r \in \mathbb{R}$);
- c_m : corda do meio (asas trapezoidais duplas apenas) ($c_m \in \mathbb{R}$);
- c_t : corda da ponta ($c_t \in \mathbb{R}$);
- mac : corda aerodinâmica média ($mac \in \mathbb{R}$);
- S : área alar ($S \in \mathbb{R}$);
- $sweep$: enflechamento (asas trapezoidais simples apenas) ($sweep \in \mathbb{R}$);
- $sweep1$: enflechamento da primeira seção (asas trapezoidais duplas apenas) ($sweep1 \in \mathbb{R}$);
- $sweep2$: enflechamento da segunda seção (asas trapezoidais duplas apenas) ($sweep2 \in \mathbb{R}$);
- v_{ex_r} : vetor com as informações do extradorso do perfil da raiz ($v_{ex} \in \mathbb{R}$);
- v_{in_r} : vetor com as informações do intradorso do perfil da raiz ($v_{ex} \in \mathbb{R}$);
- $symm_r$: valor especificando se o aerofólio da raiz é simétrico ($symm = 1$) ou assimétrico ($symm = 0$);
- v_{ex_m} : vetor com as informações do extradorso do perfil do meio ($v_{ex} \in \mathbb{R}$);
- v_{in_m} : vetor com as informações do intradorso do perfil do meio ($v_{ex} \in \mathbb{R}$);
- $symm_m$: valor especificando se o aerofólio do meio é simétrico ($symm = 1$) ou assimétrico ($symm = 0$);
- v_{ex_t} : vetor com as informações do extradorso do perfil da ponta ($v_{ex} \in \mathbb{R}$);
- v_{in_t} : vetor com as informações do intradorso do perfil da ponta ($v_{ex} \in \mathbb{R}$);
- $symm_t$: valor especificando se o aerofólio da ponta é simétrico ($symm = 1$) ou assimétrico ($symm = 0$);
- tw_m : torção geométrica da estação do meio (asas trapezoidais duplas apenas) ($tw_m \in \mathbb{Z}$);
- tw_t : torção geométrica da estação da ponta ($tw_t \in \mathbb{Z}$);
- $NODE$: matriz de nós da malha para o APAME ($NODE \in \mathbb{R}$);

- *PANEL*: matriz de painéis para o APAME ($PANEL \in \mathbb{Z}$);
- *sec_N*: número de seções transversais discretas na asa ($sec_N \in \mathbb{Z}$);
- *aero*: matriz com dados aerodinâmicos do aerofólio ($C_L, C_D, L/D, C_m \in \mathbb{R}$, $C_D > 0$), como visto na Eq. 3.1;
- *score*: aptidão do indivíduo em relação à função objetivo medida como um valor numérico ($score \in \mathbb{R}$).

Portanto, adotando novamente $BPn = 4$ para os aerofólios em todas as estações, pode-se averiguar que asas trapezoidais simples têm 29 variáveis de *design* e asas trapezoidais duplas têm 44 variáveis de *design* neste algoritmo. Alternativamente, são $5 + (BPn_r + 2) \cdot 2 + (BPn_t + 2) \cdot 2$ variáveis de *design* para asas trapezoidais simples e $8 + (BPn_r + 2) \cdot 2 + (BPn_m + 2) \cdot 2 + (BPn_t + 2) \cdot 2$ variáveis de *design* para asas trapezoidais duplas, onde BPn_r , BPn_m e BPn_t são os graus do polinômio de Bernstein dos aerofólios da raiz, do meio e da ponta, respectivamente.

Os mesmos requisitos de geometria das plantas das asas utilizados no algoritmo NACA 4 dígitos 3D são impostos também neste algoritmo. As interpolações de aerofólios também são realizadas quando necessário.

3.5.2 Geração da população inicial

Os intervalos definidos para as variáveis de *design* das plantas são iguais àqueles definidos para o algoritmo NACA 4 dígitos 3D, os quais foram apresentados na Subseção 3.4.2. Também como naquele algoritmo, a proporção de ambos os tipos de planta é definido como $planf_op = 0,5$ no algoritmo CST 3D.

Os parâmetros geométricos dos aerofólios são definidos dentro das seguintes extensões, para todas as estações da asa:

- R_{LE} : definido de 0,01 a 0,05 para o extradorso e de 0,01 a 0,03, ambos em passos de 0,01;
- A_r : definido de 0 a 0,3 para o extradorso e de 0 a 0,1 para o intradorso, ambos em passos de 0,05;
- β : definido de 5° a 20° para o extradorso e de $(20^\circ - \beta_{ex})$ a 20° para o intradorso, como números inteiros apenas.

Novamente, esses valores serão utilizados para os testes de parâmetros, e serão alterados de acordo com a necessidade ao longo do resto deste trabalho. Assim como no algoritmo NACA 4 dígitos 3D, optou-se por reduzir o espaço de busca dos aerofólios como uma compensação para o aumento do número de variáveis de *design*.

3.5.3 Avaliação, atribuição de aptidão e seleção para reprodução

Estes estágios funcionam precisamente da mesma forma que no algoritmo NACA 4 dígitos 3D. Os detalhes foram apresentados nas Subseções 3.4.3 e 3.4.4.

3.5.4 Crossover

Os esquemas de *crossover* funcionam da mesma forma que no algoritmo NACA 4 dígitos 3D, como demonstrado na Subseção 3.4.5. A única distinção é que, no modo de *crossovers* gerais das características, as trocas de genes dos aerofólios é feito de modo a imitar precisamente os esquemas de *crossover* do algoritmo CST tradicional, da mesma forma que os *crossovers* do algoritmo NACA 4 dígitos 2D foram implementados sem alteração no algoritmo NACA 4 dígitos 3D.

3.5.5 Mutação

Como nos demais algoritmos, a ativação da mutação é feita para um dado número de indivíduos, sendo esse número baseado na chance de mutação especificada pelo usuário. A escolha dos parâmetros geométricos aos quais será imposta a mutação também é realizada como no algoritmo NACA 4 dígitos 3, onde utiliza-se um vetor 1×12 , onde cada elemento representa um parâmetro geométrico. Assim como naquele algoritmo, a mutação dos parâmetros da planta da asa também é feita de modo a atribuir novos valores dentro das extensões originais. Já a mutação dos aerofólios, quando são selecionados para o processo, é realizada como no algoritmo CST convencional, como demonstrado na Subseção 3.2.6.

3.6 Estrutura do algoritmo CST 3D específico

3.6.1 Representação das asas dentro do algoritmo

A representação dos indivíduos dentro deste algoritmo é feita da mesma forma que no algoritmo CST 3D convencional. Os detalhes foram apresentados na Subseção 3.5.1.

3.6.2 Geração da população inicial

Como em sua versão 2D, o algoritmo CST 3D específico opera de modo a otimizar uma única geometria especificada pelo usuário. Tanto a planta como os aerofólios podem ter suas geometrias modificadas para os fins de otimização. Novamente, o usuário configura os parâmetros geométricos fornecendo um valor máximo de subtração e um valor máximo de adição para cada variável de *design*. Para as plantas das asas, esses valores são os seguintes:

- b : valor máximo de subtração igual a 1m, valor máximo de adição valor a 1m, valor mínimo possível igual a 1m;
- $b1$: valor máximo de subtração igual a 1m, valor máximo de adição valor a 1m, valor mínimo possível igual a 1m separação mínima da ponta de 0,5m;
- c_r , c_r e c_r : valor máximo de subtração igual a 0,5m, valor máximo de adição igual a 0,5m, valor mínimo possível igual a 0,5m;
- $sweep$, $sweep1$ e $sweep2$: valor máximo de subtração igual a 5° , valor máximo de adição igual a 5°
- tw_t : valor máximo de subtração igual a 5° , valor máximo de adição igual a zero.

Similarmente para os aerofólios, também são impostos valores mínimos possíveis aos valores de envergaduras e cordas da planta, pois os mesmos devem sempre ser positivos e não nulos.

Para os aerofólio de todas as estações, os parâmetros são os seguintes:

- R_{LE} : valor máximo de subtração igual a 0,01, valor máximo de adição igual a 0,01, valor mínimo possível igual a 0,005 para ambas as superfícies;
- A_r : valor máximo de subtração igual a 0,1, valor máximo de adição igual a 0,1 para ambas as superfícies;
- β : valor máximo de subtração igual a 5° , valor máximo de adição igual a 5° para o extradorso, valor máximo de subtração igual a $(L - \beta_{ex})$, valor máximo de subtração igual a 5° para o intradorso.

Sendo estes valores adotados para os testes de parâmetros.

3.6.3 Avaliação, atribuição de aptidão, seleção para reprodução, *crossover* e mutação

Todos estes estágios funcionam da mesma forma que no algoritmo CST 3D. Os detalhes desses respectivos estágios estão disponíveis nas Subseções 3.4.3, 3.4.4, 3.5.4 e 3.5.5, onde, para este último, a única distinção se dá na mutação dos aerofólios, que obedece as mesmas operações do algoritmo CST específico 2D.

3.7 Validações do algoritmo genético

Algoritmos de otimização escritos do zero precisam ser validados para provar seu bom desempenho. Isso é feito com o uso de funções teste, que avaliam a capacidade e

eficiência do algoritmo de solucionar problemas de otimização. Sendo assim, foi usada a estrutura básica do algoritmo escrito para este trabalho, sem as adaptações para o tratamento de aerofólios e asas.

As funções de validação escolhidas são descritas nas subseções seguintes, juntamente aos respectivos resultados do algoritmo. Em todos os casos, os problemas são definidos em termos da dimensão n : cada indivíduo do algoritmo é um vetor numérico $1 \times n$, usado para calcular os resultados das funções teste. Os testes serão realizados sempre com as funções configuradas com dimensão 20, e a população será configurada com três tamanhos distintos: 400, 200 e 100 indivíduos, de modo que a dimensão escolhida seja, respectivamente, 5%, 10% e 20% do tamanho da população. O propósito dessa estratégia é mostrar como o algoritmo se comporta com cada função, já que a procura pelo ótimo global é facilitada com populações grandes e vice-versa. Cada indivíduo é definido como um vetor de tamanho n , composto por valores x_i , definidos dentro das extensões respectivas a cada função. A seleção para reprodução é feita da mesma forma que mostrado na Subseção 3.1.5, e o *crossover* é realizado como exemplificado na Fig. 4. A mutação é feita selecionando aleatoriamente uma casa dentro do vetor do indivíduo e reatribuindo um valor aleatório, novamente dentro da extensão referente a cada função.

Em todos os casos, a chance de mutação é 5%, e não há mecanismos de elitismo implementados. O número de iterações é contado até aquela em que o melhor resultado do algoritmo deixa de oscilar.

3.7.1 Função de De Jong

A função de De Jong mais simples é a Eq. 3.10.

$$f(x) = \sum_{i=1}^n x_i^n \quad (3.10)$$

Onde $i = 1, 2, \dots, n$ e os termos x_i são definidos no domínio $-5, 12 \leq x_i \leq 5, 12$. O mínimo global é $f_* = 0$ em $\mathbf{x}_* = (0, 0, \dots, 0)$. Os resultados dos testes com essa função estão na Tab. 2. Como esperado, o algoritmo chega muito mais próximo do mínimo global com a configuração de maior população, mas também não há muitas dificuldades quando o tamanho populacional diminui.

População	Resultado	Iterações até estabilização
400	0,0020	13
200	0,0343	12
100	0,0946	12

Tabela 2 – Resultados dos testes de validação com a função de De Jong

3.7.2 Função de Ackley

A função de Ackley é Eq. 3.11.

$$f(x) = -20 \exp \left[-\frac{1}{5} \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2} \right] - \exp \left[\frac{1}{n} \sum_{i=1}^n \cos(2\pi x_i) \right] + 20 + \exp[1] \quad (3.11)$$

Onde $i = 1, 2, \dots, n$ e os termos x_i são definidos no domínio $-32,768 \leq x_i \leq 32,768$. O mínimo global é $f_* = 0$ em $\mathbf{x}_* = (0, 0, \dots, 0)$. Os resultados dos testes com essa função estão na Tab. 3. Novamente, o trabalho do algoritmo é facilitado com populações maiores, mas, apesar da maior dificuldade imposta pela maior complexidade da função, o algoritmo segue tendendo ao mínimo global em todos os casos.

População	Resultado	Iterações até estabilização
400	0	47
200	3,8609	64
100	6,2076	91

Tabela 3 – Resultados dos testes de validação com a função de Ackley

3.7.3 Função de Griewank

A função de Griewank é Eq. 3.12.

$$f(x) = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos \left(\frac{x_i}{\sqrt{i}} \right) + 1 \quad (3.12)$$

Onde $i = 1, 2, \dots, n$ e os termos x_i são definidos no domínio $-600 \leq x_i \leq 600$. O mínimo global é $f_* = 0$ em $\mathbf{x}_* = (0, 0, \dots, 0)$. Os resultados dos testes com essa função estão na Tab. 4. As mesmas observações feitas para as funções anteriores podem ser aplicadas neste caso.

População	Resultado	Iterações até estabilização
400	0,1094	22
200	1,2752	46
100	6,9458	85

Tabela 4 – Resultados dos testes de validação com a função de Griewank

Pode-se concluir, portanto, que o algoritmo funciona como deve, e que seus resultados são confiáveis. Os algoritmos de otimização de aerofólios e asas usam a mesma estrutura básica dos algoritmos usados nestas validações, portanto, eles também são confiáveis em seus resultados.

3.8 Testes de parâmetros

O comportamento de algoritmos genéticos depende fortemente na configuração de parâmetros como tamanho populacional, chance de mutação, entre outros. A devida configuração desses parâmetros é a diferença entre um algoritmo eficiente e um que sequer consegue alcançar resultados razoavelmente bons. Portanto, a fim de escolher e justificar as configurações dos algoritmos deste trabalho, foram realizados testes com o intuito de averiguar os efeitos desses parâmetros sobre seus resultados.

As distinções entre as parametrizações geométricas utilizadas para os aerofólios (série NACA 4 dígitos e CST) e os domínios de busca (bidimensional e tridimensional) fazem com que a atuação dos operadores genéticos tenha suas próprias características em cada algoritmo. Portanto, o estudo dos parâmetros de configuração foi repetido para cada um dos seis algoritmos, de modo a aplicar as melhores configurações a cada um. Essas análises estarão nas Subseções abaixo e no Apêndice. Serão apresentadas algumas opções para cada parâmetro e, com os resultados, serão justificadas as escolhas das configurações para os estudos de caso, que serão apresentados adiante.

Para todas as análises a seguir, o algoritmo foi configurado com a função objetivo de maximizar a razão L/D . Portanto, a matriz *coeff_op* é definida como

```
dat.coeff_op = ['!'; '!'; '^'; '!']
```

E o vetor *coeff_val* é irrelevante. Todos os pesos são unitários.

3.8.1 Algoritmo NACA 4 dígitos

Para o algoritmo NACA 4 dígitos, durante estes testes, a configuração padrão do algoritmo é sempre a apresentada na Tab. 5. Adicionalmente, as condições de simulação foram configuradas com número de Reynolds igual a $1e6$ e ângulo de ataque zero em todos os casos.

População	Mutação	Iterações no XFOIL	Elitismo	Iterações no algoritmo
100	5%	50	Ativado	5

Tabela 5 – Configuração padrão do algoritmo NACA 4 dígitos para os testes de parâmetros

O primeiro teste trata do tamanho populacional. Os resultados estão na Tab. 6.

O tamanho populacional é, possivelmente, o parâmetro mais fundamental de um algoritmo genético e também um dos mais simples. Como as soluções ótimas são modeladas pelo algoritmo usando os genes dos indivíduos, fica claro que, quanto mais variabilidade

População	Perfil NACA	L/D	Tempo
100	9610	167,43	1 min e 47,62 s
300	9610	167,43	5 min e 1,19 s
500	9610	167,43	8 min e 29,86 s

Tabela 6 – Resultados dos testes de tamanho populacional no algoritmo NACA 4 dígitos

genética na população, mais combinações possíveis de genes existem, o que aumenta as chances de obter boas soluções. Por outro lado, o tamanho da população também é simples quanto ao seu *trade-off*: quanto maior a população, mais tempo o algoritmo demora até chegar ao fim de sua execução, o que já fica claro na Tab. 6. Essa tabela também é a primeira demonstração dos resultados do algoritmo e de sua consistência - em todos os casos o algoritmo chegou ao perfil com o maior L/D possível para as condições de voo dadas. No entanto, como explicado acima, populações baixas deixam o algoritmo demasiadamente à mercê do acaso, portanto, tamanhos maiores serão preferidos.

O segundo teste trata da chance de mutação. Os resultados estão na Tab. 7.

Mutação	Perfil NACA	L/D	Tempo
5%	9610	167,43	6 min e 3,53 s
10%	9610	167,43	4 min e 6,05 s
20%	9611	164,93	4 min e 1,25 s

Tabela 7 – Resultados dos testes de chance de mutação no algoritmo NACA 4 dígitos

Estes resultados são a primeira oportunidade para discutir o tamanho populacional padrão nos testes de parâmetros, igual a 100, como visto na Tab. 5. O motivo se dá pelo fato de que, devido ao mecanismo de elitismo, o algoritmo fica "preso" no melhor resultado possível até o final das iterações caso esse ótimo global seja encontrado. Com populações suficientemente grandes, logo na primeira iteração, o algoritmo pode já encontrar o ótimo global e mantê-lo até o final. Portanto, para que os efeitos da mutação sejam visíveis, é imposto ao algoritmo uma situação adversa na forma de uma população pequena, de modo a ver como os melhores resultados vão mudando ao longo das iterações. A Fig. 41 mostra exatamente isso, para cada uma das execuções apresentadas na Tab. 7. Enquanto que nas configurações de 5% e 10% de chance de mutação o algoritmo se manteve estável da segunda iteração até o final, na configuração de 20% de chance de mutação o algoritmo aprimorou o melhor resultado na última iteração. Esse comportamento foi, com quase total certeza, devido a algum gene novo introduzido durante o processo de mutação (não é possível afirmar com certeza absoluta porque esse resultado também pode ter sido efeito de recombinações de genes já existentes ao longo das iterações). De todo modo, nota-se que o comportamento do algoritmo é estável para a faixa de valores de mutação escolhida.

O terceiro teste trata do número de iterações no ambiente de simulação viscosa do

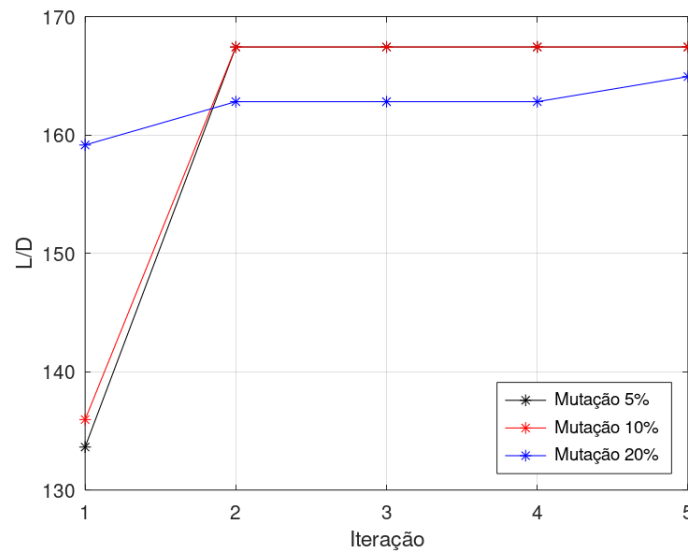


Figura 41 – Comparação da evolução dos melhores resultados em configurações distintas de mutação

XFOIL. Os resultados estão na Tab. 8.

Iterações no XFOIL	Perfil NACA	L/D	Tempo
10	9611	164,9300	1 min e 57,36 s
50	9610	167,4300	1 min e 56,60 s
100	9611	164,9300	1 min e 46,91 s

Tabela 8 – Resultados dos testes de número de iterações no XFOIL no algoritmo NACA 4 dígitos

As simulações no modo viscoso no XFOIL são um processo de natureza iterativa, que fornecem seus resultados mediante sua convergência. O grau de dificuldade que o XFOIL terá para convergir seus resultados depende diretamente da condição de voo e da geometria do aerofólio. Simulações com aerofólios a altos ângulos de ataque e/ou aerofólios com geometrias mais complexas geralmente exigem mais iterações para convergir. Simulações no XFOIL que não convergem não fornecem dados aerodinâmicos - portanto, todo aerofólio cujas simulações não convergem são ignorados no resto do algoritmo, porque é impossível avaliá-los sem possuir seus dados aerodinâmicos. Tendo tudo isso em mente, a configuração de número de iterações no XFOIL serve como uma forma de controlar o quão complexas as geometrias dos aerofólios resultantes serão. Adicionalmente, apesar de os resultados da Tab. 8 especificamente não demonstrarem isso, o aumento do número de iterações no XFOIL têm a tendência de aumentar o tempo de execução do algoritmo porque o estágio de avaliação demora mais devido a simulações mais longas.

O quarto e último teste trata do mecanismo de elitismo, cujos resultados estão na Tab. 9.

Ativação do mecanismo	Perfil NACA	L/D	Tempo
Desativado	9612	162,54	3 min e 24,87 s
Ativado	9610	167,43	3 min e 23,91 s

Tabela 9 – Resultados dos testes do mecanismo de elitismo no algoritmo NACA 4 dígitos

Para compreender de fato o que o mecanismo de elitismo altera no comportamento do algoritmo, é necessário visualizar como os melhores resultados vão mudando ao longo das iterações. A Fig. 42 apresenta essa visualização para as configurações de elitismo desativado e ativado. Para providenciar uma discussão melhor sobre este parâmetro, o algoritmo foi configurado com 10 iterações, o que permite mais variações do melhor resultado de cada iteração.

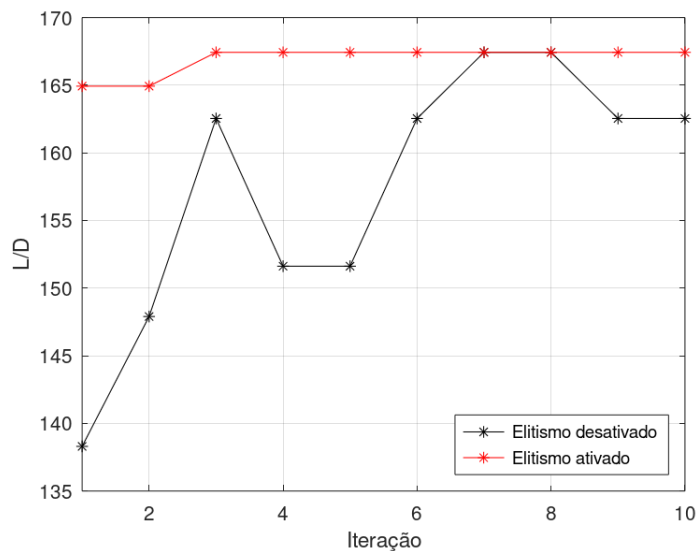


Figura 42 – Comparação da evolução dos melhores resultados em configurações distintas de elitismo no algoritmo NACA 4 dígitos (fonte: autor)

O intuito é que o algoritmo sempre siga em direção a resultados melhores. Devido à natureza aleatória do processo de seleção, *crossover* e mutação, pode ocorrer que os genes sejam misturados de forma a gerar outros aerofólios que não sejam iguais ou melhores àquele mais apto na iteração presente. Isso ocorre, por exemplo, na terceira para a quarta iteração quando o elitismo está desativado, ou nas últimas iterações. Por outro lado, com o elitismo ativado, o indivíduo mais apto de cada iteração é sempre passado para a próxima, de modo que, mesmo que o algoritmo não encontre imediatamente um resultado melhor, esse ótimo encontrado nunca retrocede. Além de garantir a permanência dos melhores resultados, o elitismo faz com que o melhor indivíduo sempre esteja presente no processo de *crossover*, o que significa que, com esses genes superiores, o algoritmo tem mais potencial para encontrar resultados ainda melhores. Portanto, o elitismo sempre estará ligado como padrão.

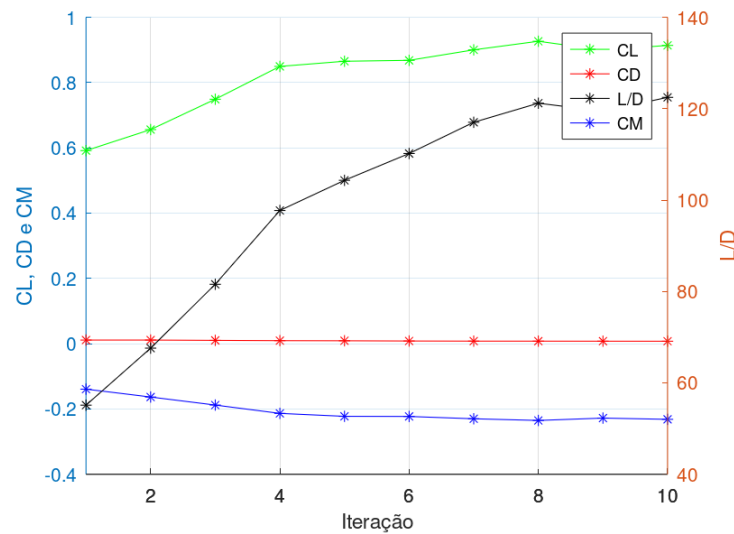


Figura 43 – Médias dos coeficientes da população completa em cada iteração no algoritmo NACA 4 dígitos com elitismo desativado (fonte: autor)

Em nota, deve-se mencionar que, apesar da oscilação dos melhores resultados, o algoritmo com elitismo desativado ainda teve comportamento desejável, sempre procurando aprimorar as soluções. Ele, inclusive, chegou a alcançar o ótimo global de $Re = 1e6$ e zero ângulo de ataque (o perfil NACA 9610), e apenas retrocedeu devido ao comportamento explicado no parágrafo anterior. Outra prova de seu bom desempenho é a Fig. 43, que mostra que a população como um todo tendeu a convergir para os melhores resultados.

Todos os aerofólios obtidos como resultados destes testes de parâmetros são apresentados no Apêndice P.1.

3.8.2 Algoritmo CST

Para o algoritmo CST, durante os testes de parâmetros, a configuração padrão do algoritmo é sempre a apresentada na Tab. 10. Novamente, a condição de voo é dada por $Re = 1e6$ e zero ângulo de ataque a não ser quando especificado de outra forma. Ao contrário dos aerofólios NACA 4 dígitos, não há espaço suficiente nas tabelas para apresentar a nomenclatura e/ou as características de cada perfil CST e, portanto, as mesmas serão omitidas. Esses dados, no entanto, estarão disponíveis no apêndice.

População	Mutação	Iterações no XFOIL	Elitismo	Iterações no algoritmo
500	5%	10	Ativado	5

Tabela 10 – Configuração padrão do algoritmo CST para os testes de parâmetros

O primeiro teste se refere ao tamanho populacional. Os resultados estão na Tab. 11.

População	L/D	Tempo
500	84,88	18 min e 3,15 s
1000	82,89	46 min e 26.73 s
1500	84,69	78 min e 18.63 s

Tabela 11 – Resultados dos testes de tamanho populacional no algoritmo CST

Enfatiza-se novamente que as soluções do algoritmo genético são construídas, majoritariamente, em base dos genes da população inicial, e que a variedade de genes disponível ao algoritmo é diretamente proporcional ao tamanho da população. A mudança para uma parametrização geométrica mais complexa resulta em um número maior de variáveis de *design*, como visto na Subseção 3.2.1. Isso resulta em um espaço de busca mais amplo, o que significa que existe um número consideravelmente maior de indivíduos possíveis, tanto bons quanto ruins.

O segundo teste trata da chance de mutação. Os resultados estão na Tab. 12.

Mutação	L/D	Tempo
5%	81,43	25 min e 33.60 s
10%	84,20	23 min e 49.77 s
20%	86,91	30 min e 40.52 s

Tabela 12 – Resultados dos testes de chance de mutação no algoritmo CST

As mesmas discussões a respeito deste parâmetro realizadas para o algoritmo NACA 4 dígitos valem para o caso do algoritmo CST, incluindo a questão do tamanho populacional reduzido e o comportamento do algoritmo sob a mutação. Novamente, essas chances de mutação escolhidas para os testes não afetam gravemente o comportamento do algoritmo.

O terceiro teste trata do número de iterações no ambiente de simulação viscosa do XFOIL. Os resultados estão na Tab. 13.

Iterações no XFOIL	L/D	Tempo
10	82,25	17 min e 39.63 s
50	84,20	22 min e 7,39 s
100	88,11	21 min e 30.63 s

Tabela 13 – Resultados dos testes de número de iterações no XFOIL no algoritmo CST

Como mencionado anteriormente, o XFOIL, inadvertidamente, acaba por atuar como um avaliador adicional à população pelo fato de que, sem convergência na simulação, um aerofólio não passa adiante as suas características. Para o caso do algoritmo CST, esse aspecto é ainda mais importante devido ao fato de que os perfis CST não têm a

mesma geração comportada como no algoritmo NACA 4 dígitos. A checagem de qualidade, apresentada na Subseção 3.2.2, atua de modo a garantir que as geometrias geradas tenham, ao menos, formatos de aerofólios, mas a tarefa de garantir que esses aerofólios tenham formatos realistas fica a cargo do resto do algoritmo. O estágio de avaliação é um dos mecanismos que faz isso, já que a tendência do XFOIL é necessitar de mais iterações na simulação para simular perfis mais complexos.

O quarto e último teste trata do mecanismo de elitismo. Os resultados estão na Tab. 14.

Ativação do mecanismo	L/D	Tempo
Desativado	82,44	38 min e 49.62 s
Ativado	93,40	43 min e 19.37 s

Tabela 14 – Resultados dos testes do mecanismo de elitismo no algoritmo CST

Para averiguar o comportamento do algoritmo, novamente é apresentado um gráfico ilustrando os melhores resultados do algoritmo em cada iteração, com as configurações de elitismo desativado e ativado. Esse gráfico está na Fig. 44.

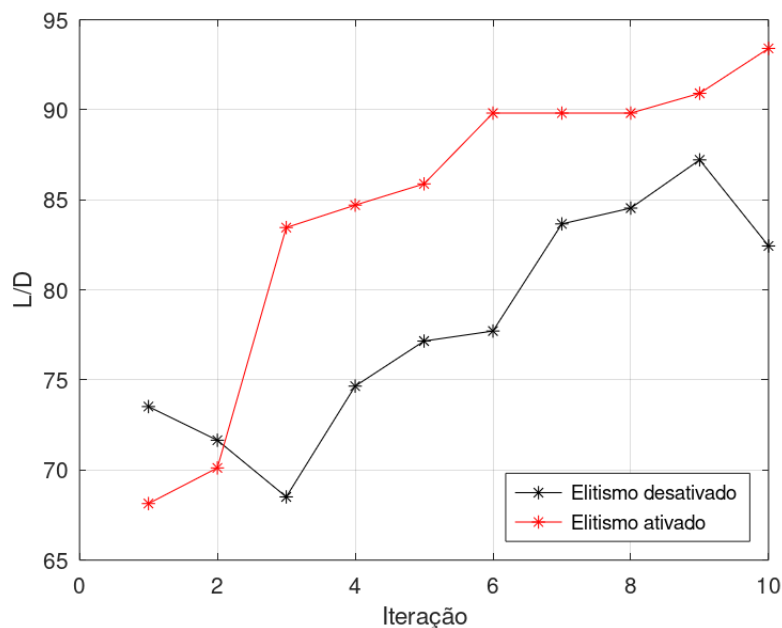


Figura 44 – Comparação da evolução dos melhores resultados em configurações distintas de elitismo no algoritmo CST (fonte: autor)

O comportamento do algoritmo com o elitismo desativado é curioso: apesar de iniciar com um resultado melhor do que no caso com o elitismo ativado, nas duas iterações seguintes os resultados pioram antes de serem aprimorados. Similarmente, o resultado da última iteração é inferior ao da iteração precedente. É relevante notar também que o algoritmo com o elitismo ativado obteve um resultado final superior. Isso demonstra como

a preservação dos melhores indivíduos ajuda o algoritmo a evoluir mais rapidamente. Finalmente, a Fig. 43 apresenta novamente o ponto importante que, apesar da oscilação dos melhores resultados, o algoritmo com elitismo desativado segue de fato manipulando as populações de modo a convergir todos à meta estipulada pela função objetivo.

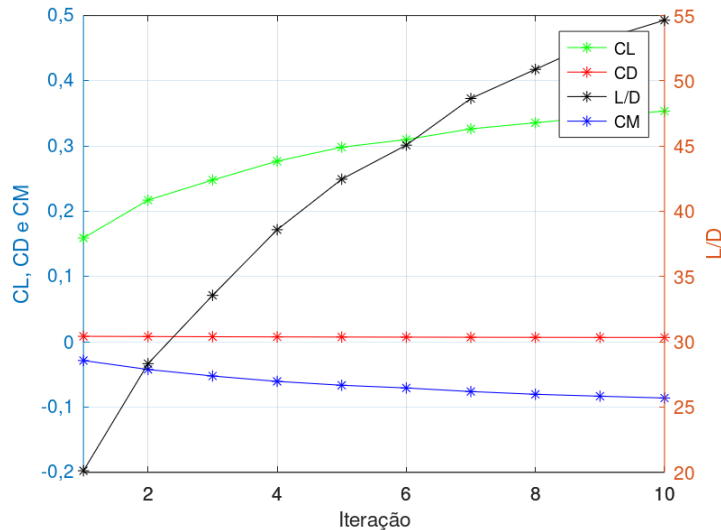


Figura 45 – Médias dos coeficientes da população completa em cada iteração no algoritmo CST com elitismo desativado (fonte: autor)

Em nota final, o tempo de execução dos algoritmos tende a variar de forma inesperada como, por exemplo, visto na Tab. 14 - em tese, os tempos de execução não deveriam variar entre esses testes, pois o mecanismo de elitismo não é tão complexo a ponto de tomar muito tempo e, mesmo que assim fosse, o esperado seria que o teste com o elitismo ativado demorasse mais do que o teste com o elitismo desativado, o que não foi o caso. O motivo para essas situações se deve a instabilidades do computador na fase de avaliação. As invocações do XFOIL, por vezes, são um pouco mais lentas do que o normal, o que consideravelmente aumenta o tempo de execução, já que o estágio de avaliação é o mais longo de todos. Essas lentidões podem ocorrer quando o algoritmo é executado ao mesmo tempo com outros programas, sejam eles processos de segundo plano mais pesados do sistema operacional, ou *software* que o usuário está utilizando.

Todos os aerofólios obtidos como resultados destes testes de parâmetros são apresentados no Apêndice P.2. O comportamento de todos os algoritmos é bastante similar tratando-se dos parâmetros de configuração, portanto, a fim de encurtar o caminho principal aos resultados deste trabalho, os demais testes de parâmetros também foram movidos para o apêndice - sendo mais exato, o Apêndice O.

3.9 Comparações com bibliografia

Antes de prosseguir aos estudos de caso propriamente ditos, o último passo para consolidar a confiabilidade destes algoritmos é realização de testes visando replicar resul-

tados de outros algoritmos de otimização apresentados em bibliografia publicada. Esse estudo será feito primeiramente para a otimização de aerofólios, e a seguir, para a otimização de asas completas.

3.9.1 Aerofólios

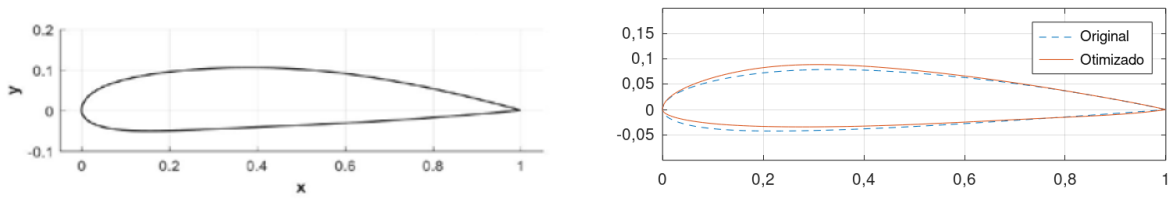
O primeiro teste de replicação é feito tendo como base o trabalho de (DINA et al., 2019), que contém uma metodologia essencialmente idêntica ao que é realizado com o algoritmo CST - isto é, emprega-se um algoritmo genético (programado justamente em MATLAB, no qual o Octave é baseado) para a otimização de aerofólios parametrizados geometricamente com CST, os quais são avaliados em simulações no XFOIL (no caso do algoritmo nomeado OPTX). É feita a parametrização em CST de um aerofólio NACA 4 dígitos genérico e esse aerofólio é otimizado em condições de voo específicas, definidas em termos de um número de Reynolds e um ângulo de ataque, com a função objetivo de maximização do L/D . Para comparação, o algoritmo CST específico deste trabalho é o mais apropriado para os testes.

O artigo não especifica qual é o aerofólio exato utilizado para as otimizações. Considerando os resultados das otimizações do OPTX, supõe-se, como aerofólio inicial para o algoritmo CST específico, um NACA 2412, configurado com polinômio de Bernstein de grau 6, o mesmo grau utilizado no artigo. Aqui serão refeitos os dois primeiros casos, que são condições de voo de número de Reynolds igual a, respectivamente, $1e5$ e $5e5$. O ângulo de ataque é nulo em ambos os casos e, como também utilizado no artigo, o tamanho populacional é de 40 indivíduos. As extensões dos valores das variáveis de *design* são as mesmas utilizadas nos testes de parâmetros do algoritmo CST específico. Para o primeiro caso, os resultados estão na Tab. 15, e as geometrias relevantes estão na Fig. 46.

Resultado	C_L	C_D	L/D
OPTX	0,40	0,016	25
CST específico	0,55	0,015	36,66
Variação	+35,5%	-6,25%	+46,64%

Tabela 15 – Resultados do caso 1 da primeira comparação (DINA et al., 2019), (fonte: autor)

Neste caso, o algoritmo CST específico ultrapassou com certa facilidade o resultado do artigo. A distinção entre esses resultados pode ser explicada por diferenças na parametrização geométrica empregada. Enquanto que, no algoritmo CST específico, foram utilizadas os valores dados na Subseção 3.3.2, o artigo não fornece os intervalos das variáveis. Adicionalmente, o algoritmo genético do *toolkit* do MATLAB utilizado no OPTX possui a taxa de *crossover*, uma configuração não presente nos algoritmos deste trabalho. A taxa de *crossover*, configurada como 0,4 no OPTX, define quantos indivíduos na



(a) Resultado do algoritmo OPTX (DINA et al., 2019) (b) Resultado do algoritmo CST específico (fonte: autor)

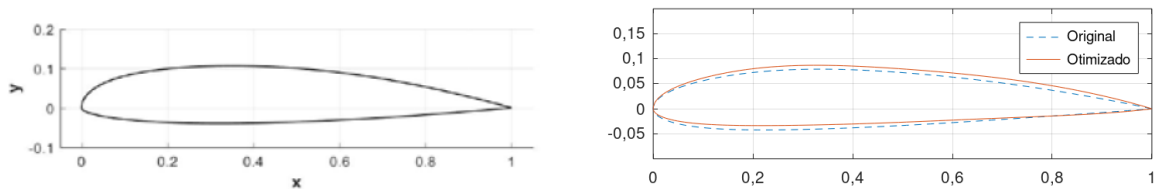
Figura 46 – Aerofólios resultantes do caso 1 da primeira comparação

população serão utilizados para o processo de reprodução, enquanto que o resto segue à iteração seguinte sem alterações. Os algoritmos deste trabalho, ao invés disso, possuem o mecanismo de elitismo (vide Subseção 3.1.8).

Para o segundo caso, os resultados estão na Tab. 16 e na Fig. 47.

Resultado	C_L	C_D	L/D
OPTX	0,395	0,0073	54,10
CST específico	0,397	0,0068	58,38
Variação	+0,5%	-6,84%	+7,91%

Tabela 16 – Resultados do caso 2 da primeira comparação (DINA et al., 2019), (fonte: autor)



(a) Resultado do algoritmo OPTX (DINA et al., 2019) (b) Resultado do algoritmo CST específico (fonte: autor)

Figura 47 – Aerofólios resultantes do caso 2 da primeira comparação

Neste caso, os resultados do algoritmo CST específico tiveram ótima correlação com o que foi obtido no artigo.

3.9.2 Asas

Para o caso de asas, o teste de replicação tomou como referência o trabalho de (KÖRPE; KANAT, 2019). Neste trabalho, é realizada a otimização de asas para VANTs (veículos aéreos não tripulados) por meio do SQL (*Sequential Quadratic Programming*, Programação Sequencial Quadrática). Três estudos de caso são realizados, dos quais o primeiro é utilizado para o presente teste.

O primeiro estudo de caso no artigo consiste em maximizar a razão $C_L^{3/2}/C_D$, a qual está diretamente relacionada com a capacidade de voos de longa duração de aeronaves movidas a hélice (GUDMUNDSSON, 2014). Os algoritmos deste trabalho não tratam da razão $C_L^{3/2}/C_D$, mas, para este teste em específico, uma pequena modificação no código foi feita para avaliar esse requisito. A geometria da asa é definida como uma planta trapezoidal simples, com perfis NACA 4 dígitos, cujas variáveis são definidas como:

- $4m \leq b \leq 8m$
- $0,2m \leq c_t \leq 0,4m$
- $0,2m \leq c_r \leq 0,4m$
- $2 \leq m \leq 8$
- $1 \leq p \leq 6$
- $8 \leq t \leq 16$

A velocidade do ar e o ângulo de ataque também são utilizados como variáveis de *design*, mas estas foram ignoradas do processo devido ao fato de que os algoritmos deste trabalho não poderem ser configurados dessa forma, nem serem modificados em tempo hábil para tal no contexto da execução deste trabalho. Para o teste de replicação, foi adotado já o valor obtido no artigo do ângulo de ataque para a condição de $C_L^{3/2}/C_D$ máximo, que é $5,56^\circ$. A velocidade foi desconsiderada por ser irrelevante nas simulações do APAME neste contexto. Também há restrições referentes ao peso da asa, mas, como estes algoritmos também não tratam desse aspecto, essas restrições serão ignoradas.

A planta de asa resultante é apresentada na Fig. 48 e os aerofólios são ilustrados na Fig. 49. Os aerofólios obtidos no algoritmo NACA 4 dígitos 3D para a raiz e a ponta, respectivamente, são um NACA 2408 e um NACA 3408.

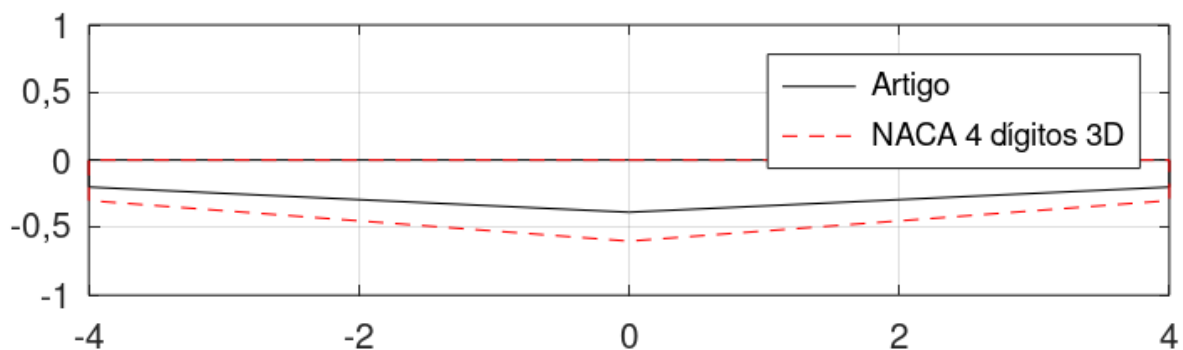


Figura 48 – Comparação das plantas de asas resultantes do teste de replicação (fonte: autor)

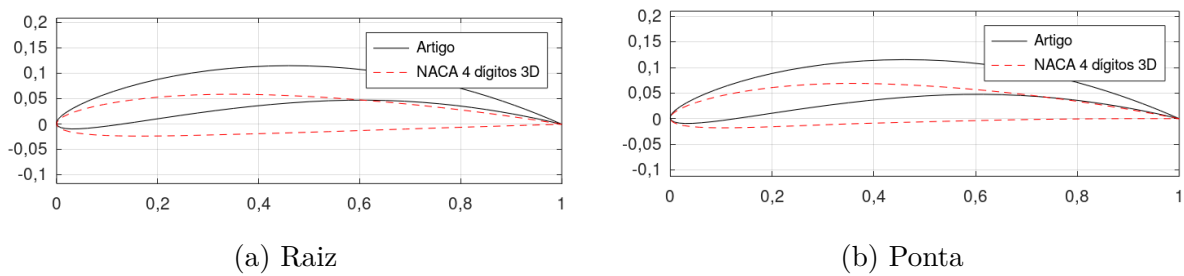


Figura 49 – Comparação dos aerofólios resultantes do teste de replicação (fonte: autor)

A geometria da planta é o resultado mais promissor deste teste, levando em conta que o algoritmo forneceu uma asa com a maior envergadura possível dentro da extensão definida e com uma razão de afilamento bastante similar (0,518 no resultado do artigo e 0,5 no resultado do algoritmo NACA 4 dígitos 3D). Os aerofólios, apesar de terem curvatura diferente, têm praticamente a mesma posição da curvatura máxima e a mesma espessura máxima: no artigo, os valores de m , p e t para o aerofólio (que é constante ao longo de toda a envergadura) são, respectivamente, 8, 5,24 e 8. Os valores de $C_L^{3/2}/C_D$ obtidos, no entanto, são discordantes. Para o artigo, foi obtido 30,79, e para o algoritmo, foi obtido 7,04. Assume-se que essa diferença seja explicada por distinções nos solucionadores empregados.

3.10 Estudos de caso

As últimas três Seções até aqui (validações dos algoritmos, testes de parâmetros e testes de replicação) tiveram o propósito de consolidar a confiabilidade dos algoritmos deste trabalho. Feito isso, segue-se agora para estudos de caso hipotéticos com a finalidade de demonstrar exemplos de aplicações dos algoritmos em análises preliminares de projeto. Serão realizados cinco estudos de caso distintos, cujas abordagens serão descritas a seguir. Todas as funções objetivo e restrições escolhidos são representações de objetivos hipotéticos para um dado projeto.

3.10.1 Primeiro estudo de caso: testes independentes

Os testes independentes consistirão das seguintes configurações e condições de voo:

1. Caso 1: obter $C_L = 0,2$ em zero ângulo de ataque;
2. Caso 2: obter $C_L = 0,2$ e minimizar C_D em zero ângulo de ataque;
3. Caso 3: obter $C_L = 0,2$ e minimizar C_D em zero ângulo de ataque com C_M constante nos ângulos de ataque de 0° e 4° .

Cada um desses testes será feito nos algoritmos NACA 4 dígitos (2D e 3D) e nos algoritmos CST (2D e 3D). Por hora ignora-se os algoritmos CST específicos, que terão papéis em alguns dos estudos de caso a seguir. O valor relativamente baixo do coeficiente de sustentação tem o propósito de ser fácil de obter, de modo que não afete demasiadamente a busca pelas demais características. A restrição de C_M constante não é feita em termos de um valor específico, portanto, se o aerofólio ou a asa tiver o mesmo C_M nas duas condições de voo (ou próximo o suficiente), o requisito será cumprido (de acordo com a opção 'c', como explicado na Subseção 3.1.4). O número de Reynolds será sempre igual a $1e6$, e o número de Mach, igual a zero². O intuito principal dos testes independentes é demonstrar como os resultados vão se tornando cada vez mais específicos à medida que se aumenta a quantidade de requisitos que a geometria deve cumprir.

É necessário enfatizar que os valores dos coeficientes de momento para um dado corpo aerodinâmico dependem do ponto de referência. O XFOIL utiliza o ponto de um quarto da corda como padrão, enquanto que o APAME utiliza a origem como referência. Em ambos os casos, nem sempre as configurações padrão são as mais apropriadas em uma dada aplicação, mas, para finalidades de demonstração do funcionamento destes algoritmos, são satisfatórias.

A configuração do algoritmo NACA 4 dígitos para este estudo de caso é:

- Tamanho populacional 500;
- Chance de mutação 5%;
- 5 iterações;
- 10 iterações no XFOIL;
- Elitismo ativado;
- m : definido de 0 a 9;
- p : definido de 0 a 9;
- t : definido de 10 a 30.

A configuração do algoritmo CST para este estudo de caso é:

- Tamanho populacional 1000;
- Chance de mutação 5%;

² O número de Mach, no contexto dos solucionadores empregados neste trabalho, tem o propósito principal de indicar se correções de compressibilidade devem ser aplicadas, e em qual intensidade. Quando $Mach < 0,3$, o ar pode ser tratado como incompressível.

- 5 iterações;
- 10 iterações no XFOIL;
- Elitismo ativado;
- R_{LE} : definido de 0,01 a 0,05 para o extradorso e de 0,01 a 0,03, ambos em passos de 0,005;
- A_r : definido de 0 a 0,3 para o extradorso e de 0 a 0,1 para o intradorso, ambos em passos de 0,025;
- β : definido de 5° a 20° para o extradorso e de $(20^\circ - \beta_{ex})$ a 20° para o intradorso, como números inteiros apenas.

A configuração do algoritmo NACA 4 dígitos 3D para este estudo de caso é:

- Tamanho populacional 500;
- Chance de mutação 5%;
- 5 iterações;
- Elitismo ativado;
- $dat.planf_op=0,5$ (aproximadamente 50% de asas trapezoidais e 50% de asas bitrapezoidais);
- b : definida de 10m a 15m em passos de 0,5m;
- $b1$: definida de 8m a 14m em passos de 0,5m, com uma separação mínima da ponta de 0,5m;
- c_r : definida de 1m a 2m em passos de 0,1m;
- c_m : definida de 0,5m a 2m em passos de 0,1m;
- c_t : definida de 0,5m a 2m em passos de 0,1m;
- $sweep$: definido de 0° a 15° em passos unitários;
- $sweep1$: definido de 0° a 15° em passos unitários;
- $sweep2$: definido de 0° a 15° em passos unitários.
- tw_t : definido de -5° a 0° em passos de 0,5°;
- m : definido de 0 a 4 para todas as estações;

- p : definido de 0 a 4 para todas as estações;
- t : definido de 10 a 20 para todas as estações.

A configuração do algoritmo CST 3D para este estudo de caso é:

- Tamanho populacional 1200;
- Chance de mutação 5%;
- 5 iterações;
- Elitismo ativado;
- $dat.planf_op=0,5$ (aproximadamente 50% de asas trapezoidais e 50% de asas bitrapezoidais);
- b : definida de 10m a 15m em passos de 0,5m;
- $b1$: definida de 8m a 14m em passos de 0,5m, com uma separação mínima da ponta de 0,5m;
- c_r : definida de 1m a 2m em passos de 0,1m;
- c_m : definida de 0,5m a 2m em passos de 0,1m;
- c_t : definida de 0,5m a 2m em passos de 0,1m;
- $sweep$: definido de 0° a 15° em passos unitários;
- $sweep1$: definido de 0° a 15° em passos unitários;
- $sweep2$: definido de 0° a 15° em passos unitários.
- tw_t : definido de -5° a 0° em passos de $0,5^\circ$;
- R_{LE} : definido de 0,01 a 0,05 para o extradorso e de 0,01 a 0,03, ambos em passos de 0,01;
- A_r : definido de 0 a 0,3 para o extradorso e de 0 a 0,1 para o intradorso, ambos em passos de 0,05;
- β : definido de 5° a 20° para o extradorso e de $(20^\circ - \beta_{ex})$ a 20° para o intradorso, como números inteiros apenas.

3.10.2 Segundo estudo de caso: detalhando aerofólios

Como mencionado na Subseção 3.2.1, todos os algoritmos CST (2D e 3D) permitem configurar a geometria dos aerofólios de acordo com o número de variáveis de *design*, o que permite ao projetista determinar o nível de detalhamento. Adicionalmente, o algoritmo CST específico, apresentado na Seção 3.3, tem o propósito de fazer o processo de otimização dentro de um espaço de busca próximo a uma única geometria. O objetivo deste teste é utilizar ambos estes fatos para um estudo de otimização. O primeiro passo será obter um aerofólio com $C_L = 0,5$ no algoritmo CST configurado com $BPn = 4$ e, a seguir, esse aerofólio será otimizado no algoritmo CST específico duas vezes: uma para minimizar o arrasto, e outra para induzir o coeficiente de momento a zero o quanto possível, procurando manter o C_L original em ambos os casos e usando $BPn = 6$. A condição de voo será definida por $Re = 1e6$ e ângulo de ataque zero.

A configuração do algoritmo CST para este estudo de caso é:

- Tamanho populacional 500;
- Chance de mutação 5%;
- 5 iterações;
- 10 iterações no XFOIL;
- Elitismo ativado;
- R_{LE} : definido de 0,01 a 0,05 para o extradorso e de 0,01 a 0,03, ambos em passos de 0,005;
- A_r : definido de 0 a 0,3 para o extradorso e de 0 a 0,1 para o intradorso, ambos em passos de 0,025;
- β : definido de 5° a 20° para o extradorso e de $(20^\circ - \beta_{ex})$ a 20° para o intradorso, como números inteiros apenas.

A configuração do algoritmo CST específico para este estudo de caso é:

- Tamanho populacional 150;
- Chance de mutação 5%;
- 5 iterações;
- 10 iterações no XFOIL;
- Elitismo ativado;

- R_{LE} : valor máximo de subtração igual a 0,01, valor máximo de adição igual a 0,01, valor mínimo possível igual a 0,005 para ambas as superfícies;
- A_r : valor máximo de subtração igual a 0,05, valor máximo de adição igual a 0,05 para ambas as superfícies;
- β : valor máximo de subtração igual a 5° , valor máximo de adição igual a 5° para o extradorso, valor máximo de subtração igual a $(L - \beta_{ex})$, valor máximo de subtração igual a 5° para o intradorso.

3.10.3 Terceiro estudo de caso: coeficientes de momento constantes

A maximização da razão L/D é um objetivo comum na otimização de corpos aerodinâmicos sustentadores. No entanto, a busca indiscriminada dessa meta induz, na grande maioria das vezes, a geometrias que possuem defeitos severos em relação a outras características aerodinâmicas. Focando no contexto de aerofólios, um desses defeitos se demonstra na instabilidade, representada pelo coeficiente de momento de arfagem. Aerofólios com coeficientes de sustentação excepcionalmente altos acabam por ter também módulos grandes do coeficiente de momento. Essa situação por si só já é prejudicial à estabilidade da aeronave, a qual é piorada adicionalmente com o fato de que os coeficientes de momento desses aerofólios também têm a tendência de variarem demasiadamente com a alteração do ângulo de ataque.

Este estudo de caso tem o objetivo de demonstrar como a aplicação de restrições ao coeficiente de momento pode melhorar a qualidade de resultados de otimizações para maximizar a razão L/D . O algoritmo CST será executado duas vezes visando maximizar o L/D , uma vez com a restrição de CM constante desativada e outra vez ativada. Este requisito será imposto na forma da opção ' k ' configurada para $CM = -0,05$. Serão três condições de voo, todas no mesmo número de Reynolds ($1e6$), com os seguintes ângulos de ataque: 3° , 6° e 9° . Busca-se maximizar o L/D no primeiro ângulo de ataque.

A configuração do algoritmo CST para este estudo de caso é:

- Tamanho populacional 1000;
- Chance de mutação 5%;
- 5 iterações;
- 10 iterações no XFOIL;
- Elitismo ativado;
- R_{LE} : definido de 0,01 a 0,05 para o extradorso e de 0,01 a 0,03, ambos em passos de 0,005;

- A_r : definido de 0 a 0,3 para o extradorso e de 0 a 0,1 para o intradorso, ambos em passos de 0,025;
- β : definido de 5° a 20° para o extradorso e de $(20^\circ - \beta_{ex})$ a 20° para o intradorso, como números inteiros apenas.

3.10.4 Quarto estudo de caso: testes integrados

Os testes integrados visam utilizar múltiplos algoritmos tendo em vista um objetivo final. Esse objetivo consiste em modelar uma asa de planador, com planta trapezoidal dupla, de modo a minimizar o arrasto em uma condição de voo qualquer em que aeronave esteja voando a 5° de ângulo de ataque. A estratégia é tomada seguindo o que foi apresentado na Seção 2.6: asas com distribuições elípticas são as que possuem maior eficiência aerodinâmica, e tais distribuições são obtidas quando os coeficientes de sustentação locais de cada seção transversal da asa são iguais ao longo da envergadura completa. A abordagem tomada é a seguinte:

1. Otimizar os perfis da raiz, do meio e da ponta para minimizar o arrasto, ao mesmo tempo garantindo $C_L = 0,8$ (algoritmo CST);
2. Gerar uma asa utilizando os aerofólios obtidos nos passos anteriores e otimizá-la no algoritmo CST 3D específico para minimizar o arrasto. Os únicos parâmetros alterados serão os comprimentos de corda, a envergadura da primeira seção e os enflechamentos.
3. Executar o algoritmo CST 3D para encontrar uma asa com o mesmo C_{Ls} da asa encontrada no passo 2, ao mesmo tempo minimizando o arrasto. A asa gerada no passo 3 é distinta daquela obtida no passo 2.

Além do ângulo de ataque, a condição de voo é definida por $Re = 1e6$ e Mach 0.

A configuração do algoritmo CST para este estudo de caso é:

- Tamanho populacional 700;
- Chance de mutação 5%;
- 5 iterações;
- 10 iterações no XFOIL;
- Elitismo ativado;
- R_{LE} (para o perfil da raiz): definido de 0,01 a 0,35 para o extradorso e intradorso, ambos em passos de 0,005;

- A_r (para o perfil da raiz): definido de 0,1 a 0,3 para o extradorso, em passos de 0,005. Para o intradorso, admite-se apenas o valor 0,1;
- R_{LE} (para o perfil do meio): definido de 0,005 a 0,02 para o extradorso e para o intradorso, ambos em passos de 0,005;
- A_r (para o perfil do meio): definido de 0 a 0,15 para o extradorso e intradorso, ambos em passos de 0,025;
- R_{LE} (para o perfil da ponta): definido de 0,005 a 0,01 para o extradorso e intradorso, ambos em passos de 0,005;
- A_r (para o perfil da ponta): definido de 0 a 0,2 para o extradorso e de 0 a 0,1 para o intradorso, ambos em passos de 0,025;
- β (todos os perfis): definido de 5° a 10° para o extradorso e de $(10^\circ - \beta_{ex})$ a 10° para o intradorso, como números inteiros apenas.

A configuração do algoritmo CST 3D para este estudo de caso é:

- Tamanho populacional 1200;
- Chance de mutação 5%;
- 5 iterações;
- Elitismo ativado;
- *dat.planf_op=1* (apenas asas bitrapezoidais);
- *b*: definida como 14m apenas;
- *b1*: definida de 11,5m a 12,5m em passos de 0,5m, com uma separação mínima da ponta de 0,1m;
- *c_r*: definida de 1,4m a 1,6m em passos de 0,05m;
- *c_m*: definida de 0,9m a 1,1m em passos de 0,05m;
- *c_t*: definida de 0,4m a 0,6m em passos de 0,05m;
- *sweep*: irrelevante neste caso (pois o algoritmo trata apenas de asas bitrapezoidais);
- *sweep1*: definido de -3° a 7° em passos unitários;
- *sweep2*: definido de 15° a 25° em passos unitários.
- *tw_t*: definido como nulo sempre;

- R_{LE} (todas as estações): definido de 0,01 a 0,05 para o extradorso e de 0,01 a 0,03, ambos em passos de 0,01;
- A_r (todas as estações): definido de 0 a 0,3 para o extradorso e de 0 a 0,1 para o intradorso, ambos em passos de 0,05;
- β (todas as estações): definido de 5° a 20° para o extradorso e de $(20^\circ - \beta_{ex})$ a 20° para o intradorso, como números inteiros apenas.

A configuração do algoritmo CST 3D específico para este estudo de caso é:

- Tamanho populacional 450;
- Chance de mutação 5%;
- 5 iterações;
- Elitismo ativado;
- b : valor original 14m e nunca modificado;
- $b1$: valor original 12m, com valor máximo de subtração igual a $0,5m$ e valor máximo de adição igual a $0,5m$;
- c_r : valor original 1,5m, com valor máximo de subtração igual a $0,1m$ e valor máximo de adição igual a $0,1m$;
- c_m : valor original 1m, com valor máximo de subtração igual a $0,1m$ e valor máximo de adição igual a $0,1m$;
- c_t : valor original 0,5m, com valor máximo de subtração igual a $0,1m$ e valor máximo de adição igual a $0,1m$;
- $sweep$: irrelevante neste caso (pois a asa é bitrapezoidal);
- $sweep1$: valor original 2° , com valor máximo de subtração igual a 5° e valor máximo de adição igual a 5° ;
- $sweep2$: valor original 20° , com valor máximo de subtração igual a 5° e valor máximo de adição igual a 5° ;
- tw_t : valor original 0° e nunca modificado;
- Parâmetros dos aerofólios serão os obtidos no primeiro passo deste estudo de caso, e nunca modificados.

3.10.5 Quinto estudo de caso: comparação com o VLM

Os resultados dos algoritmos de otimização de aerofólios e asas são diretamente afetados pelo método de solução empregado. Tendo isso em vista, este estudo de caso tem o objetivo de fazer uma comparação entre os resultados dos algoritmos de asas quando são configurados com solucionadores distintos. Será realizada a otimização de uma asa para maximizar o L/D no algoritmo NACA 4 dígitos 3D, o qual será executado duas vezes: uma vez configurado com o APAME (painéis 3D) e uma vez configurado com o VSPAERO (VLM). A condição de voo será definida pelo ângulo de ataque de 5° e Mach 0.

A configuração do algoritmo NACA 4 dígitos 3D para este estudo de caso é:

- Tamanho populacional 300 para o algoritmo com o APAME e 100 para o algoritmo com o VSPAERO;
- Chance de mutação 5%;
- 5 iterações;
- Elitismo ativado;
- $dat.planf_op=0,5$ (aproximadamente 50% de asas trapezoidais e 50% de asas bitrapezoidais);
- b : definida de 10m a 15m em passos de 0,5m;
- $b1$: definida de 8m a 14m em passos de 0,5m, com uma separação mínima da ponta de 0,5m;
- c_r : definida de 1m a 2m em passos de 0,1m;
- c_m : definida de 0,5m a 2m em passos de 0,1m;
- c_t : definida de 0,5m a 2m em passos de 0,1m;
- $sweep$: definido de 0° a 15° em passos unitários;
- $sweep1$: definido de 0° a 15° em passos unitários;
- $sweep2$: definido de 0° a 15° em passos unitários.
- tw_t : definido de -5° a 0° em passos de $0,5^\circ$;
- m : definido de 0 a 4 para todas as estações;
- p : definido de 0 a 4 para todas as estações;

- t : definido de 10 a 20 para todas as estações.

A API do OPENVSP tem um problema que muito provavelmente se trata de um vazamento de memória³. A API consome memória ao gerar asas para o algoritmo e não a libera em seguida, o que torna o processo de avaliação mais lento. Após algumas centenas de indivíduos, a avaliação de cada indivíduo torna-se lenta ao ponto de impedir o término do algoritmo em tempo hábil, requerindo-se horas para avaliar populações relativamente pequenas. Por esse motivo, no código com o VSPAERO, o algoritmo foi configurado para ter uma população de 100 indivíduos para manter o tempo de execução similar ao que já se observa nos demais testes e estudos de caso.

³ "Em ciência da computação vazamento de memória ou *memory leak* é um fenômeno que ocorre quando um programa de computador gerencia incorretamente alocações de memória de maneira que certa memória não é liberada quando não é mais necessária."(WIKIPEDIA, 2020)

4 Resultados e discussões

Este capítulo apresenta os resultados dos algoritmos de otimização dentro dos estudos de caso definidos anteriormente. Todos os resultados a seguir foram obtidos na versão em Octave do algoritmo exceto quando especificado de outra forma.

4.1 Primeiro estudo de caso: testes independentes

Como descrito na Subseção 3.10.1, os algoritmos NACA 4 dígitos e CST (ambos em suas respectivas versões 2D e 3D) foram executados três vezes, variando as funções objetivo e restrições. A descrição dos casos é repetida abaixo para facilitar sua referência:

1. Caso 1: obter $C_L = 0,2$ em zero ângulo de ataque;
2. Caso 2: obter $C_L = 0,2$ e minimizar C_D em zero ângulo de ataque;
3. Caso 3: obter $C_L = 0,2$ e minimizar C_D em zero ângulo de ataque com C_M constante nos ângulos de ataque de 0° e 4° .

4.1.1 Algoritmo NACA 4 dígitos

Os dados aerodinâmicos dos perfis NACA 4 dígitos obtidos neste teste estão na Tab. 17 e suas geometrias estão na Fig. 50. O requisito de $C_L = 0,2$ foi cumprido com bastante facilidade em todos os casos. Do NACA 1813 para o NACA 2210 houve uma redução de 0,5% do coeficiente de arrasto, e o NACA 1922 obteve coeficientes de momento razoavelmente próximos. Este último aerofólio é o primeiro exemplo claro neste trabalho da complexidade de otimizações com múltiplos requisitos - os três requisitos definidos entram em conflito entre si e a obtenção de soluções que satisfaçam perfeitamente todos os três torna-se mais complicada. Nesta solução em específico, o aerofólio não teve seu arrasto reduzido tanto quanto o NACA 2210, o que pode ser explicado, justamente, pelo maior número de requisitos a serem cumpridos. Outra explicação é a menor flexibilidade da parametrização da série NACA 4 dígitos. Outro aspecto relevante a se mencionar sobre o perfil resultante do caso 3 é sua maior espessura, o que também é uma explicação para o maior coeficiente de arrasto.

As restrições de obtenção de coeficientes de momento constantes são facilmente satisfeitas pelo algoritmo NACA 4 dígitos, pois esses aerofólios são notórios por suas características amigáveis de estabilidade - isso pode ser averiguado em material como (ABBOTT; DOENHOFF, 1959), nos quais os gráficos de ângulo de ataque *versus* coeficiente de momento são, na grande maioria das vezes, linhas quase completamente retas e

Aerofólio	C_L	C_D	L/D	C_M	Tempo
Caso 1: NACA 1813	0,1999	0,0056	35,1940	-0,0538	6 min e 33,87 s
Caso 2: NACA 2210	0,2022	0,0053	37,5140	-0,0341	7 min e 21,32 s
Caso 3: NACA 1922 (AoA 0°)	0,1980	0,0082	23,9420	-0,0544	12 min e 6,52 s
Caso 3: NACA 1922 (AoA 4°)	0,6262	0,0092	67,4780	-0,0495	-

Tabela 17 – Dados aerodinâmicos dos aerofólios resultantes dos testes independentes do algoritmo NACA 4 dígitos (fonte: autor)

horizontais¹ (o que implica em CMs praticamente constantes ao longo de toda a extensão de ângulos de ataque dos aerofólios). De todo modo, em qualquer um dos algoritmos deste trabalho, não é necessário que as funções ' c' ' e ' k' ' para o coeficiente de momento sejam cumpridas perfeitamente - basta que os valores não variem demasiadamente ao longo dos ângulos de ataque, e a solução será satisfatória em termos de sua estabilidade.

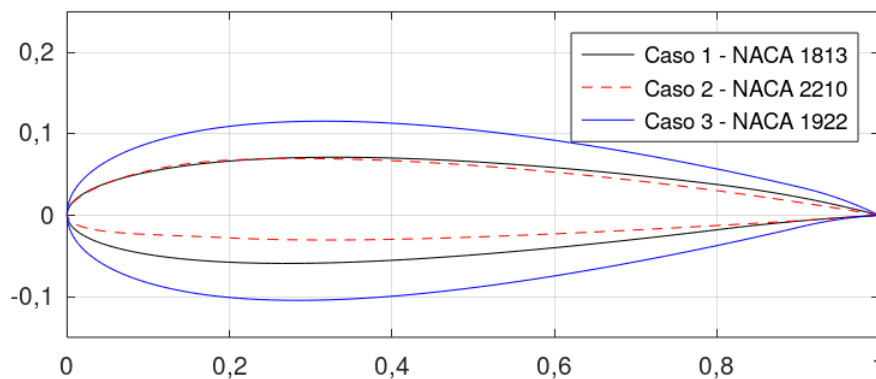


Figura 50 – Aerofólios resultantes dos testes independentes do algoritmo NACA 4 dígitos (fonte: autor)

4.1.2 Algoritmo CST

Os dados aerodinâmicos dos perfis CST obtidos neste teste estão na Tab. 18 e suas geometrias estão na Fig. 51. Logo de início, comparando com os resultados do teste anterior (vide Tab. 17), pode-se fazer a observação de que os perfis CST cumpriram os requisitos de otimização com maior facilidade do que os perfis NACA 4 dígitos. A restrição de $C_L = 0,2$ foi cumprida com ainda menos erro (com o perfil CST 1 alcançando-o perfeitamente), e a redução de arrasto foi bem sucedida em ambos os perfis dos casos 2 e 3, havendo uma redução de 49,43% do perfil CST 1a para o perfil CST 2a e uma redução de 6% do perfil CST 1a para o perfil CST 3a. Como mencionado na discussão anterior, o maior número de requisitos em um processo de otimização aumenta a sua complexidade. Desse modo, no caso do perfil CST 3a, o algoritmo teve mais dificuldade na minimização de arrasto devido aos demais requisitos conflitantes.

¹ Isso é devido à escala do eixo das ordenadas utilizado nos gráficos, no entanto.

Aerofólio	C_L	C_D	L/D	C_M	Tempo
Caso 1: CST 1a	0,2000	0,0089	22,4720	-0,0253	36 min e 48,41 s
Caso 2: CST 2a	0,2010	0,0045	43,8860	-0,0530	37 min e 47,99 s
Caso 3: CST 3a (AoA 0°)	0,1999	0,0083	23,8540	-0,0429	46 min e 28,50 s
Caso 3: CST 3a (AoA 4°)	0,6349	0,0084	75,5830	-0,0353	-

Tabela 18 – Dados aerodinâmicos dos aerofólios resultantes dos testes independentes do algoritmo CST (fonte: autor)

Tratando agora do coeficiente de momento, nota-se que houve uma variação de 17,71% entre os C_M s das duas condições de voo, enquanto que, no mesmo caso para o algoritmo NACA 4 dígitos, houve uma variação de 9%. Como mencionado antes, o algoritmo NACA 4 dígitos tem maior facilidade em cumprir esse requisito devido às características dos seus aerofólios. Portanto, para o caso do algoritmo CST, o processo de otimização é mais complexo para garantir o cumprimento de todos os requisitos e, sendo assim, o projetista deve ter atenção à forma como o algoritmo é configurado.

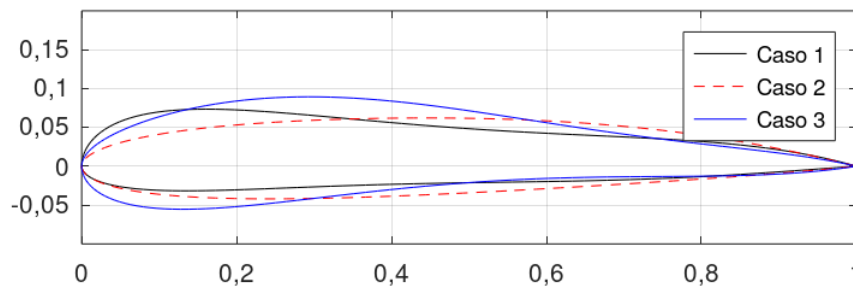


Figura 51 – Aerofólios resultantes dos testes independentes do algoritmo CST (fonte: autor)

Os dados das geometrias dos perfis obtidos neste teste estão disponíveis no Ap. Q.1.1.

4.1.3 Algoritmo NACA 4 dígitos 3D

Os dados aerodinâmicos das asas obtidas neste teste estão na Tab. 19 e suas geometrias estão na Fig. 52. Também aqui todos os resultados têm coeficientes de sustentação iguais a 0,2 ou muito próximos, e, em ambos os casos em que foi aplicada, a redução de arrasto foi bem sucedida. O requisito de momento constante também foi seguido satisfatoriamente.

Um aspecto interessante destes resultados é a tendência do algoritmo de preferir aerofólios mais finos na raiz e aerofólios mais espessos na ponta. Ocasionalmente, como no caso 1, o algoritmo também gera asas cujo perfil do meio é o mais espesso de todos. As plantas das asas variaram de modo notável e, em especial, o caso 2 produziu a asa com

Asa	C_L	C_D	L/D	C_M	Tempo
Caso 1: Asa 1	0,2000	0,0183	10,9290	-0,1834	28 min e 44,60 s
Caso 2: Asa 2	0,1932	0,0092	21,0000	-0,1271	30 min e 18,07 s
Caso 3: Asa 3 (AoA 0°)	0,1931	0,0093	9,1241	-0,0808	47 min e 50,65 s
Caso 3: Asa 3 (AoA 4°)	0,4642	0,0545	5,8031	-0,1099	-

Tabela 19 – Dados aerodinâmicos das asas resultantes dos testes independentes do algoritmo NACA 4 dígitos (fonte: autor)

a maior razão de aspecto - o que é lógico considerando o foco na redução de arrasto em conjunto com o requisito de C_L

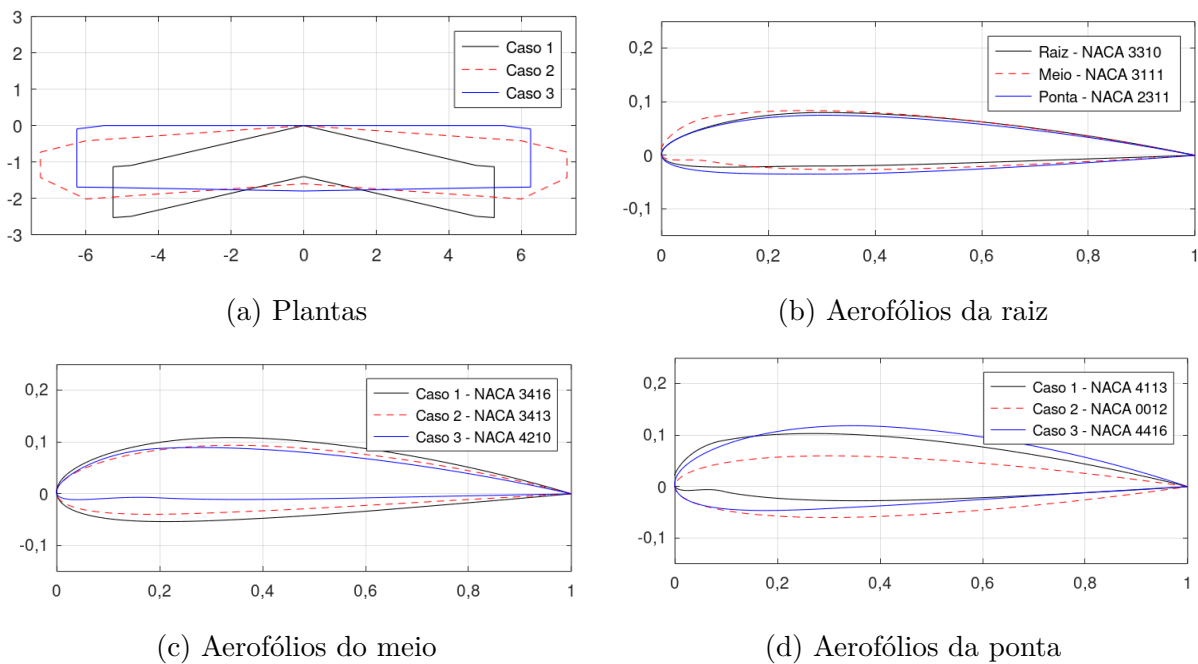


Figura 52 – Asas resultantes do algoritmo NACA 4 dígitos 3D para o primeiro estudo de caso

Os dados das geometrias das asas obtidas neste teste estão disponíveis no Ap. Q.1.2.

4.1.4 Algoritmo CST 3D

Os dados aerodinâmicos das asas obtidas neste teste estão na Tab. 20 e suas geometrias estão na Fig. 53. Novamente o requisito de $C_L = 0,2$ foi alcançado com facilidade em todos os casos. Curiosamente, a asa 3 obteve um coeficiente de arrasto menor do que aquele obtido pela asa 2, apesar do maior número de requisitos (isto é, o C_L específico, a redução de C_D e C_M s constantes). Nota-se também que o algoritmo CST 3D produziu asas com características de momento menos suaves que as das asas do algoritmo NACA 4 dígitos 3D, como se pode averiguar pelos valores dos coeficientes de momento.

Asa	C_L	C_D	L/D	C_M	Tempo
Caso 1: Asa 1	0,2000	0,0157	12,7390	-0,1402	91 min e 49,22 s
Caso 2: Asa 2	0,1939	0,0140	17,0090	-0,1136	86 min e 56,05 s
CAso 3: Asa 3 (AoA 0°)	0,2055	0,0111	18,5140	-0,1467	150 min e 50,56 s
Caso 3: Asa 3 (AoA 4°)	0,5416	0,0879	6,1615	-0,2293	-

Tabela 20 – Dados aerodinâmicos das asas resultantes dos testes independentes do algoritmo CST 3D (fonte: autor)

Apesar da parametrização geométrica diferente dos aerofólios, os resultados deste teste são similares àqueles no algoritmo NACA 4 dígitos 3D. Novamente, a tendência do processo de otimização foi fornecer aerofólios cuja espessura aumenta à medida que se segue para as pontas das asas. Similarmente, a planta de asa obtida no caso 1 tem como sua maior característica seus enflechamentos, e a asa obtida no caso 2 tem alta razão de aspecto, um resultado esperado tendo em vista o foco em redução de arrasto - uma razão de aspecto até notavelmente maior que a da asa obtida para o mesmo caso no algoritmo NACA 4 dígitos.

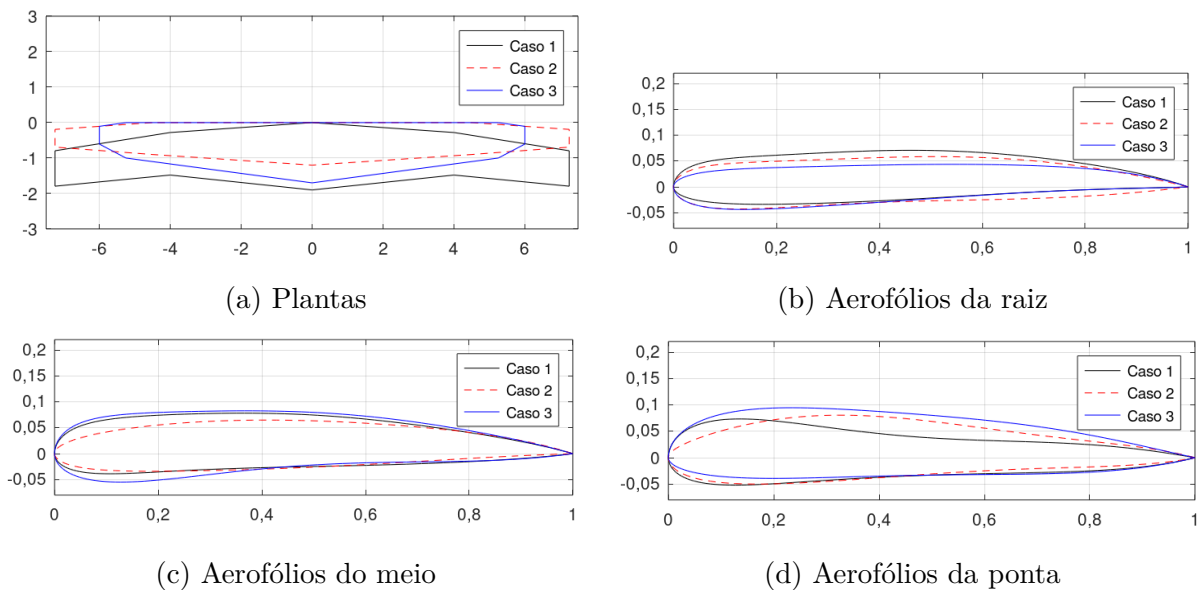


Figura 53 – Asas resultantes do algoritmo CST 3D para o primeiro estudo de caso

Em nota final para este estudo de caso, uma observação a ser dada, válida para todos os quatro algoritmos utilizados, é que processos de otimização como os realizados neste trabalho são naturalmente mais complicados devido à imposição de múltiplos requisitos que, na esmagadora maioria das vezes, são conflitantes entre si. Para o caso dos algoritmos deste trabalho, a consistência dos resultados é beneficiada por populações maiores e por maiores números de iterações (ambos dos quais, evidentemente, induzem a maiores tempos de execução dos códigos).

Os dados das geometrias das asas obtidas neste teste estão disponíveis no Ap.

Q.1.3.

4.2 Segundo estudo de caso: detalhando aerofólios

Três aerofólios CST foram obtidos neste teste: o primeiro originou-se a partir de uma busca para cumprir o objetivo de C_L específico, e os outros dois foram especificações da geometria do primeiro aerofólio tendo em vista outros objetivos de otimização adicionalmente ao requisito do C_L . Os dados aerodinâmicos estão apresentados na Tab. 21 e as geometrias dos aerofólios estão na Fig. 54.

Aerofólio	C_L	C_D	L/D	C_M
Perfil CST 1b (original)	0,5015	0,0067	74,2960	-0,1388
Perfil CST 2b (otimização de C_D)	0,5108	0,0054	93,0420	-0,1445
Perfil CST 3b (otimização de C_M)	0,5007	0,0068	73,2020	-0,1189

Tabela 21 – Dados aerodinâmicos dos aerofólios obtidos no segundo estudo de caso (fonte: autor)

Todo os perfis cumpriram devidamente o objetivo de $C_L = 0,5$, sendo o segundo o que mais se desviou do valor intencionado. O segundo perfil também obteve uma redução de 19,40% no coeficiente de arrasto, juntamente com um aumento de 25,23% na razão L/D , o que se dá tanto pelo aumento do C_L quanto pela redução no C_D . Nota-se que também houve um aumento no módulo do coeficiente de momento.

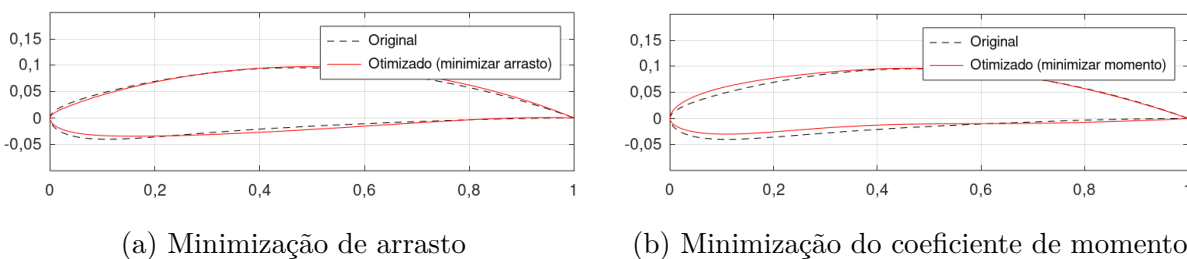


Figura 54 – Comparação do aerofólio inicial obtido com suas respectivas otimizações no segundo estudo de caso (fonte: autor)

O terceiro perfil, ignorando a redução desejada no módulo do coeficiente de momento (redução de 14,33%), tem características de sustentação e arrasto muito similares aos do perfil 1. Em especial, deve-se frisar um aspecto interessante destes resultados: os perfis 1 e 2 têm bordos de ataque cuja geometria é mais pronunciada para baixo devido ao fato de que seus raios do bordo de ataque do extradorso são menores que os raios do bordo de ataque do intradorso. Essa não é uma característica comum em aerofólios reais. Em contraste a isso, o perfil 3 perdeu completamente essa característica, adquirindo um bordo de ataque mais realista (tanto por não apontar tanto para baixo quanto por ser

mais arredondado) e, novamente, ainda mantendo em grande parte todas as outras características do perfil original. Esse resultado demonstra, como será enfatizado no estudo de caso seguinte, a importância da consideração do coeficiente de momento em processos de otimização devido à propensão desses requisitos imporem geometrias mais realistas e/ou características de *performance* mais realistas como resultados. Enfatiza-se também que os perfis 2 e 3 foram configurados com polinômios de Bernstein de grau 6, ao contrário do padrão 4, o que, devido à maior quantidade de variáveis de *design*, permite mais detalhamentos na geometria - isso sem dúvida auxiliou no cumprimento dos objetivos especificados neste estudo de caso.

Os dados das geometrias destes aerofólios estão disponíveis no Ap. Q.2.

4.3 Terceiro estudo de caso: coeficientes de momento constantes

Como exposto na Subseção 3.10.3, o algoritmo CST, neste teste, foi executado duas vezes para encontrar perfis que maximizassem a razão L/D na condição de voo de $Re = 1e6$ e ângulo de ataque de 3° . Na segunda surtida, o código foi configurado com o requisito adicional de buscar coeficientes de momento constantes no ângulo de ataque original e em mais dois adicionais. Os dados aerodinâmicos estão na Tab. 22 e as geometrias obtidas estão na Fig. 55

Aerofólio	C_L	C_D	L/D	C_M	Tempo
Perfil CST 1c	0,8347	0,0062	133,7700	-0,1238	54 min e 29,59 s
Perfil CST 2c (AoA 3°)	0,7126	0,0073	96,8210	-0,0600	56 min e 31,60 s
Perfil CST 2c (AoA 6°)	1,0102	0,0085	118,2900	-0,0500	-
Perfil CST 2c (AoA 9°)	1,2952	0,0108	119,2600	-0,0390	-

Tabela 22 – Dados aerodinâmicos dos aerofólios obtidos no terceiro estudo de caso (fonte: autor)

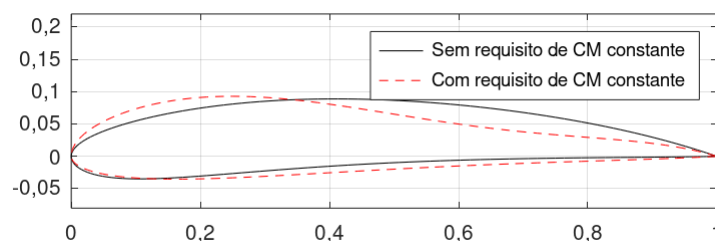


Figura 55 – Aerofólios obtidos no terceiro estudo de caso (fonte: autor)

Pode-se averiguar, claramente, que o perfil CST 2c obteve um L/D notoriamente menor do que o obtido pelo perfil CST 1c devido aos requisitos impostos para o coeficiente de momento. Similarmente, o perfil CST 1c possui um módulo de coeficiente de momento bastante alto, ao contrário do perfil CST 2c. É evidente como a capacidade de

geração de sustentação está diretamente relacionada às características de momento dos aerofólios. Enfatiza-se novamente que o ponto de referência para o cálculo dos coeficientes de momento é uma consideração importante. Como o XFOIL tem o ponto de quarto de corda como referência, o emprego do requisito de CM constante efetivamente faz com que o algoritmo tente forçar a realocação dos centros aerodinâmicos dos perfis para esse ponto.

Os dados dos CMs do perfil CST 2c podem induzir a questionamentos se o processo de otimização foi realmente efetivo - e de fato foi. Como mencionado na Subseção 4.1.1, não é necessário que todos os coeficientes de momento sejam absolutamente iguais, o que, em termos práticos, é quase impossível. Como prova, outra forma de se avaliar as características de momento do aerofólio é traçando um gráfico de seus coeficientes de momento ao longo de uma extensão de ângulos de ataque, como ilustrado na Fig. 56, obtida no XFLR5². Além dos dois perfis CST obtidos neste teste, também foi incluído um NACA 4 dígitos genérico para fins de comparação, pois, como já explicado, são aerofólios com excelentes qualidades de momento de arfagem. Nota-se que o perfil CST 1c tem módulos de CM bastante altos na faixa de pequenos ângulos de ataque, o que induz a uma grande tendência de reduzir o ângulo de ataque (movimento de nariz para baixo - picagem). Por outro lado, o perfil CST 2c tem características de momento muito similares às do perfil NACA 4 dígitos apresentado como exemplo.

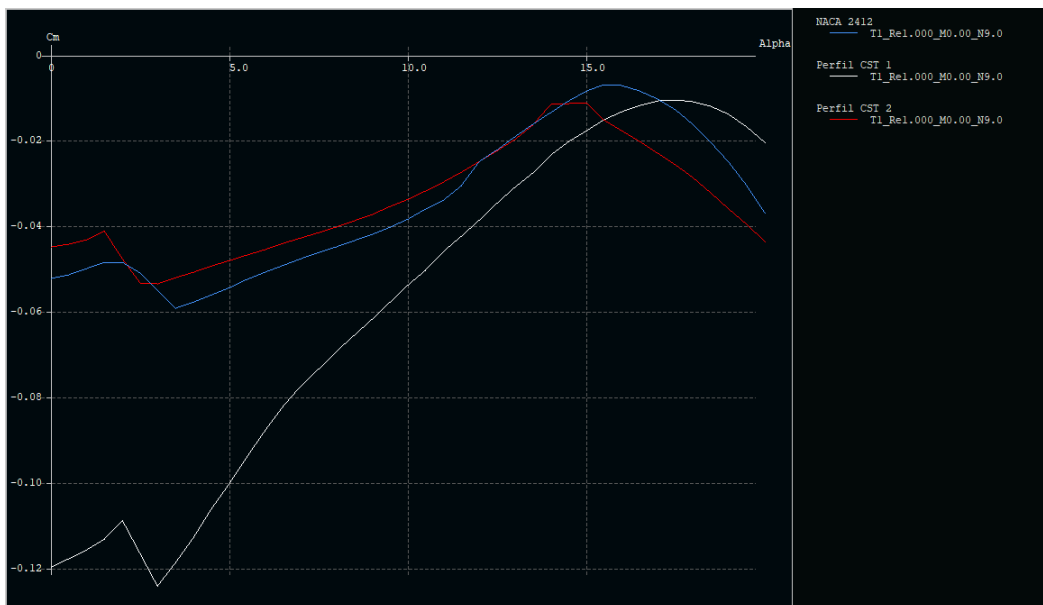


Figura 56 – Gráfico de α versus CM dos perfis CST obtidos no terceiro estudo de caso, traçado no XFLR5 (fonte: autor)

Em nota final, apesar das melhores características de estabilidade, o perfil CST 2c resultou em uma geometria incomum para aerofólios, com uma redução repentina da

² Como mencionado na Introdução, o XFLR5 é um solucionador aerodinâmico de aerofólios e asas. Para o caso 2D, o programa é baseado no código do XFOIL. Links de *download*, documentações e outras informações sobre o programa podem ser encontrados em <<http://www.xflr5.tech/xflr5.htm>>.

espessura na marca de cerca de metade da corda. Em um estudo de otimização mais detalhado, como para um projeto real, resultados como esse deveriam ser analisados com cautela para averiguar se outras características vitais de *performance* não seriam afetadas negativamente devido a esses aspectos geométricos incomuns.

Os dados das geometrias destes aerofólios estão disponíveis no Ap. Q.3.

4.4 Quarto estudo de caso: testes integrados

Para este estudo de caso, apresenta-se na Tab. 23 os dados aerodinâmicos dos aerofólios obtidos, na Tab. 24, os dados aerodinâmicos das asas obtidas, e, na Fig. 57, suas geometrias. Similarmente ao primeiro estudo de caso, os passos adotados para o presente estudo de caso são repetidos aqui para fácil consulta:

1. Otimizar os perfis da raiz, do meio e da ponta para minimizar o arrasto, ao mesmo tempo garantindo $C_L = 0,8$ (algoritmo CST);
2. Gerar uma asa utilizando os aerofólios obtidos nos passos anteriores e otimizá-la no algoritmo CST 3D específico para minimizar o arrasto. Os únicos parâmetros alterados serão os comprimentos de corda, a envergadura da primeira seção e os enflechamentos.
3. Executar o algoritmo CST 3D para encontrar uma asa com o mesmo C_L da asa encontrada no passo 2, ao mesmo tempo minimizando o arrasto. A asa gerada no passo 3 é distinta daquela obtida no passo 2.

Para fins de esclarecimento, especifica-se dois casos dentro deste estudo de caso: o primeiro refere-se à obtenção dos aerofólios separadamente, seguida pela obtenção da asa utilizando estes aerofólios (o teste integrado em si, ou seja, os dois primeiros passos), e o segundo caso refere-se ao uso do algoritmo CST 3D para a geração de uma asa, ou seja o terceiro passo - portanto, todos os aerofólios do segundo caso foram obtidos ao mesmo tempo que sua respectiva asa.

Aerofólio	C_L	C_D	L/D	C_M	Tempo
Raiz (caso 1)	0,8000	0,0068	117,6500	-0,0546	24 min e 7,15 s
Meio (caso 1)	0,7937	0,0084	93,5970	-0,0381	31 min e 12,88 s
Ponta (caso 1)	0,7923	0,0071	110,6600	-0,0368	23 min e 3,33 s
Raiz (caso 2)	0,0715	0,0079	9,0506	0,0007	90 min e 36.84 s
Meio (caso 2)	0,2014	0,0084	2,3976	-0,0539	90 min e 36.84 s
Ponta (caso 2)	0,0557	0,0102	5,4607	-0,0125	90 min e 36.84 s

Tabela 23 – Dados aerodinâmicos dos aerofólios resultantes dos algoritmos CST e CST 3D específico para o quarto estudo de caso (fonte: autor)

Asa	C_L	C_D	L/D	C_M	Tempo
Caso 1	0,54	0,0481	12,2270	-0,2848	35 min e 5.37 s
Caso 2	0,54	0,0952	5,6722	-0,2054	90 min e 36.84 s

Tabela 24 – Dados aerodinâmicos das asas resultantes dos algoritmos CST 3D e CST 3D específico para o quarto estudo de caso (fonte: autor)

Agora, compara-se as duas abordagens adotadas em cada caso. Logo de início, nota-se dois aspectos importantes. Em primeiro lugar, a abordagem do caso 1 demorou, no total, pouco menos de duas horas, o que foi mais do que o caso 2. Por outro lado, em segundo lugar, a estratégia de preparar os aerofólios antes de otimizar a planta da asa resultou em um sucesso maior no objetivo de minimização de arrasto: a asa do caso 1 tem um coeficiente de arrasto 49,47% menor que a asa do caso 2.

Todos os aerofólios do caso 1 são imensamente superiores aos respectivos perfis obtidos no caso 2, obtendo, por exemplo, razões L/D que chegam a ser duas ordens de grandeza maiores. Adicionalmente às melhores características aerodinâmicas, os perfis do caso 1 também têm geometrias mais "agradáveis", mais similares a aerofólios reais. Este estudo de caso é um claro exemplo da vantagem de se reduzir o escopo da busca em algoritmos de otimização - especialmente aqueles com múltiplos requisitos.

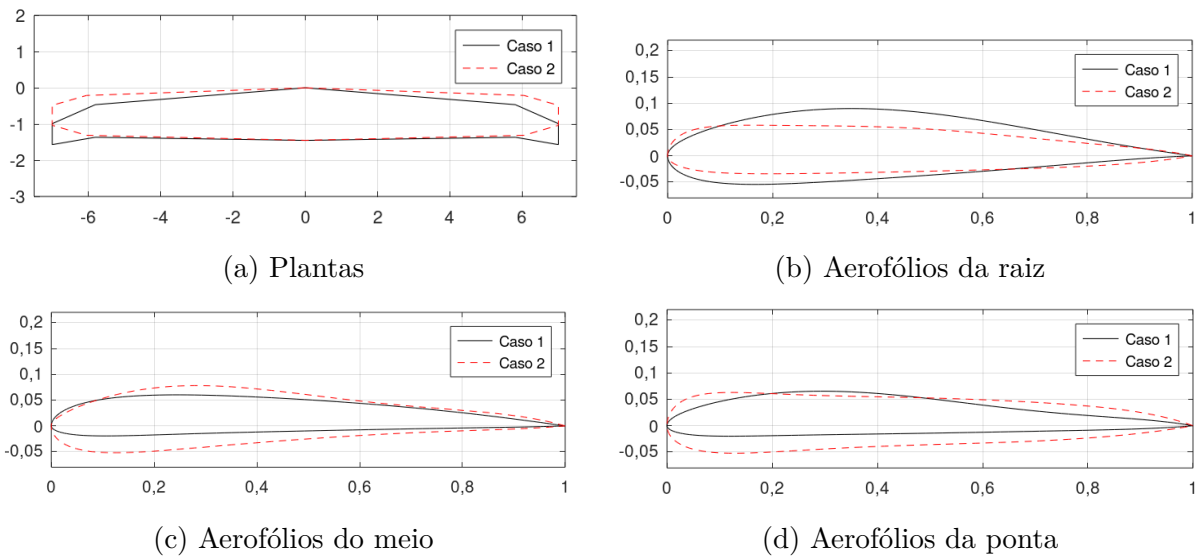


Figura 57 – Asas e aerofólios resultantes dos algoritmos CST, CST 3D e CST 3D específico para o quarto estudo de caso (fonte: autor)

Por fim, é errado afirmar que a abordagem do caso 2 é inútil - o projeto de aerofólios possui mudanças substanciais quando este é simulado num domínio 2D, e quando é simulado num domínio 3D já inserido no local ao qual pertence (como, no caso deste trabalho, as estações nas asas). O *design* de aerofólios considerando logo no princípio os efeitos tridimensionais do escoamento é uma estratégia inteligente e, com a devida configuração e preparo, pode fornecer ótimos resultados. Exemplos de aplicação dessa abordagem

são (HICKS; HENNET, 1978) e (JUN et al., 2006). Como vale para os outros estudos de caso deste trabalho, enfatiza-se a conclusão de que a qualidade dos resultados fornecidos para um processo de otimização é diretamente proporcional à adequação do algoritmo ao problema em mãos.

Os dados das geometrias destas asas e aerofólios estão disponíveis no Ap. Q.4.

4.5 Quinto estudo de caso: comparação com o VLM

As geometrias das duas asas encontradas neste teste estão na Fig. 58, sendo diretamente comparadas as plantas e os aerofólios de cada estação. Os dados aerodinâmicos das asas estão na Tab. 25. O caso 1 refere-se à otimização feita com o APAME e o caso 2 refere-se à otimização feita com o VSPAERO. Frisa-se que apesar das diferenças de linguagem e do solucionador (versão Octave configurada com o APAME e versão Python configurada com o VSPAERO), a estrutura dos algoritmos de otimização em si não possuem distinções, operando com exatamente os mesmos passos, os mesmos operadores genéticos (esquemas de *crossover* e mutação) e as mesmas parametrizações geométricas.

Asa	C_L	C_D	L/D	C_M	Tempo
Caso 1	0,2732	0,0169	16,1660	-0,0866	14 min e 34,01 s
Caso 2	0,7110	0,0211	33,6383	-0,6465	109 min e 55,96 s

Tabela 25 – Dados aerodinâmicos das asas obtidas no quinto estudo de caso (fonte: autor)

Os resultados obtidos apresentam tendências distintas. Quando se trata da maximização do L/D , o resultado natural a se obter são plantas com alta razão de aspecto, geralmente com pouco ou nenhum enflechamento. Foi precisamente um resultado como esse que foi obtido no caso 2, ao passo que a asa do caso 1 tem uma planta um pouco mais ampla. Também pode-se observar que o algoritmo com o APAME forneceu uma asa bitrapezoidal, com uma tendência também observada em alguns dos estudos de caso anteriores e nos testes de parâmetros, ao passo que o algoritmo com o VSPAERO resultou em uma planta trapezoidal simples. Já para os aerofólios, o algoritmo com o VSPAERO preferiu um perfil mais espesso na raiz, e o algoritmo com o APAME preferiu aerofólios simétricos na maioria da extensão da envergadura da asa. Em adição a isso, nota-se o tempo de execução dos códigos: a versão no Python com o VSPAERO demorou mais de sete vezes o tempo tomado pela versão no Octave com o APAME, o que ocorreu devido ao problema exposto na Subseção 3.10.5.

Repete-se aqui, novamente, as distinções fundamentais entre ambos os solucionadores: o APAME emprega um método de painéis 3D sem correções de viscosidade, enquanto o VSPAERO utiliza o VLM modificado para considerar efeitos de viscosidade. Desse modo, mesmo que sua análise completa de arrasto não seja precisa devido à sua natureza

invisível, era esperado que o APAME convergisse com maior facilidade às geometrias de planta descritas no parágrafo anterior. Nos trabalhos de validação do código realizados em (FILKOVIC, 2008), é analisado apenas o C_{D0} , o coeficiente de arrasto a sustentação zero, o que não esclarece o quão preciso o programa é na estimativa de arrasto induzido (o qual, lembrando, existe em escoamentos invíscidos para corpos sustentadores). No entanto, dito isso, a consistência dos resultados nesse aspecto poderia ser aprimorada por meio do refinamento da malha de painéis no APAME. Adicionalmente, nota-se que a asa do caso 1 tem *washout* ligeiramente maior que a asa do caso 2, o que poderia ser interpretado como uma forma do algoritmo de tentar se aproximar da distribuição elíptica de sustentação.

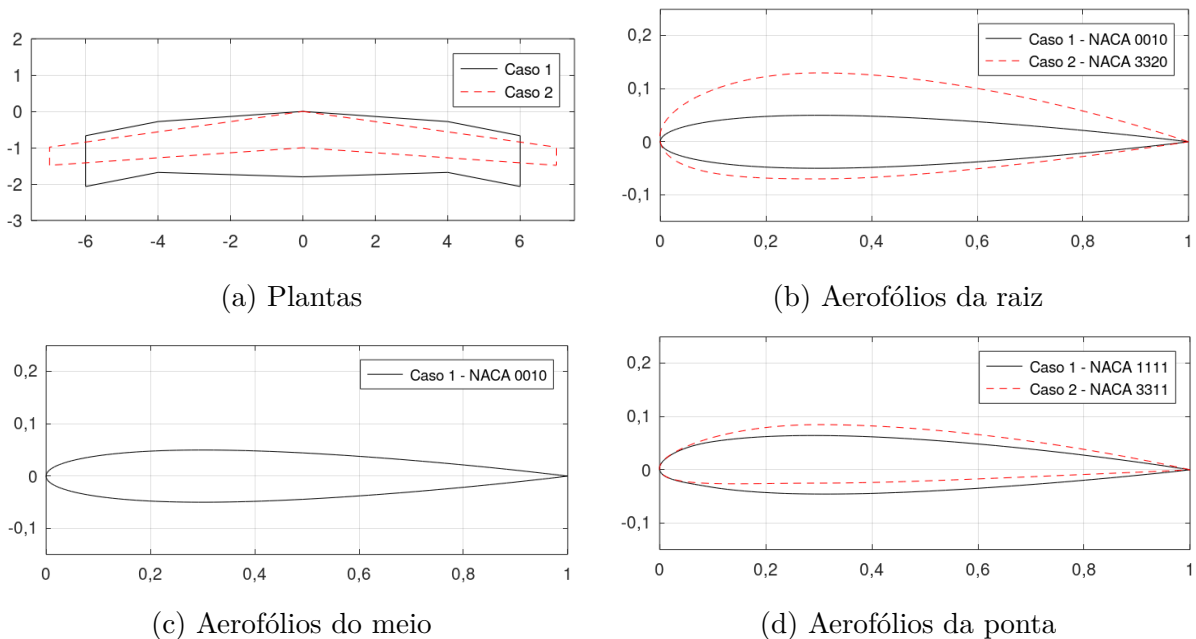


Figura 58 – Asas e aerofólios resultantes do algoritmo NACA 4 dígitos para o quinto estudo de caso (fonte: autor)

De todo modo, finalmente, este último estudo de caso teve o propósito principal de demonstrar exemplos de variações nos resultados de um algoritmo de otimização quando altera-se o solucionador. Adicionalmente às lições tomadas nos demais estudos de caso, enfatiza-se que o solucionador a ser empregado deve ser escolhido de modo a suprir eficazmente as necessidades do projetista, tendo em vista as limitações e requisitos de projeto, ambos dos quais variam de acordo com o estágio no qual o mesmo se encontra.

Os dados das geometrias destas asas e aerofólios estão disponíveis no Ap. Q.5.

5 Conclusões

A proposta deste trabalho foi o desenvolvimento de ferramentas de otimização voltadas para aerofólios e asas de uso aeronáutico. Este texto iniciou-se tratando da contextualização das questões relevantes em mãos, discorrendo sobre algoritmos de otimização, seus possíveis objetos de estudo e trabalhos anteriores que empregaram tais metodologias. A seguir, foi feita uma revisão bibliográfica a respeito de todos os conceitos necessários, referente tanto ao algoritmo genético em si quanto aos tópicos de aerodinâmica referentes ao *software* empregado. Depois, foi apresentada a metodologia detalhada dos estudos realizados, incluindo a estrutura dos códigos e as estratégias de integração do *software* de simulação e, finalmente, foram expostos os resultados dos algoritmos sob os estudos de caso definidos.

Tendo em vista as validações do algoritmo (Seção 3.7), os testes de parâmetros (Seção 3.8 e Apêndice O), os testes de replicação (Seção 3.9) e os estudos de caso (Capítulo 4), pode-se afirmar que há consistência em seus resultados independentemente do problema em mãos, tanto nas versões de aerofólios quanto nas versões de asas. O que foi apresentado neste documento é apenas uma parcela do que se pode realizar com esses algoritmos, havendo potencial para a solução de diversos problemas distintos que podem ser configurados de acordo com as intenções do usuário. A possibilidade de alterar funcionalidades no código (como funções objetivo, restrições, parametrizações geométricas e variáveis de *design*) apenas aumenta ainda mais a gama de possibilidades. Sendo assim, este trabalho serviu como mais um exemplo da efetividade de técnicas que, apesar de já terem décadas de idade, seguem ainda sendo muito relevantes em face de alternativas mais avançadas e complexas - especialmente considerando o enorme aumento da capacidade computacional desde que essas técnicas surgiram. Algoritmos genéticos não apenas são uma excelente introdução a algoritmos de otimização como também, com a devida implementação, são bem-sucedidos e confiáveis em suas aplicações, as quais podem ser as mais diversas devido à facilidade com a qual é possível readaptá-los aos mais diversos tipos de problema. Similarmente, o método de painéis é uma aplicação tão elegante quanto a teoria de escoamento potencial que o originou. Sua credibilidade para estimar escoamentos invíscidos é notória desde o princípio e, com as devidas correções de compressibilidade e viscosidade, obtém-se códigos imprescindíveis para projeto e *design*.

Dito isso, é fundamental ter em mente que algoritmos de otimização não são ferramentas perfeitas e que não devem ser utilizadas indiscriminadamente. A otimização é um processo que deve ser aliado às demais filosofias e abordagens de projeto ao invés de ser tratada como um caminho fácil para a solução de problemas. Esta segunda estratégia, mesmo que resolva o que se deseja, apenas piora outros problemas já existentes, ou faz

surgir problemas que sequer existiam antes. Enfatiza-se aqui, assim como foi feito em diversos pontos deste texto, que a engenharia é um balanço de perdas e ganhos e que simplesmente não existem soluções que cumpram absolutamente tudo o que se deseja para um dado projeto. Um exemplo disso foi no terceiro estudo de caso: de nada adianta gerar um aerofólio que tenha elevada eficiência aerodinâmica se suas características de estabilidade não forem adequadas para voo seguro. Outro exemplo é dado na Fig. 59. Oriundo de um trabalho anterior do autor (FERNANDES; JÚNIOR, 2021), esse aerofólio foi obtido otimizando-se para o L/D e nada mais, aplicando a otimização como um experimento ao invés de utilizá-la para um propósito realista. O objetivo foi cumprido, alcançando um L/D de 178,60 em $Re = 1e6$ e zero ângulo de ataque, mas, além dos problemas de estabilidade, esse perfil também seria muito difícil de construir e quaisquer esforços para inserir e organizar componentes internos (mecanismos de superfícies de controle, longarinas, trens de pouso, entre outros), independentemente da obtenção de sucesso ou não, seria um processo notoriamente frustrante. Em adição a este assunto, a otimização de aerofólios em especial nem sempre é a melhor opção a se tomar, devido a todo o estudo que deve ser realizado para averiguar a *performance* de uma geometria nova. A vantagem de se utilizar aerofólios existentes está em conhecer de antemão suas características aerodinâmicas, o que poupa tempo e recursos que seriam utilizados em simulações numéricas e experimentos em túneis de vento ou similares. Mas, novamente, isso não significa que deve-se abandonar por completo a otimização - basta apenas que a mesma seja realizada com cautela ao que se deseja para o projeto. O projetista pode, inclusive, adotar abordagens como os algoritmos CST específico (2D e 3D), que evita com que as geometrias obtidas afastem-se demasiadamente do que se tem originalmente.

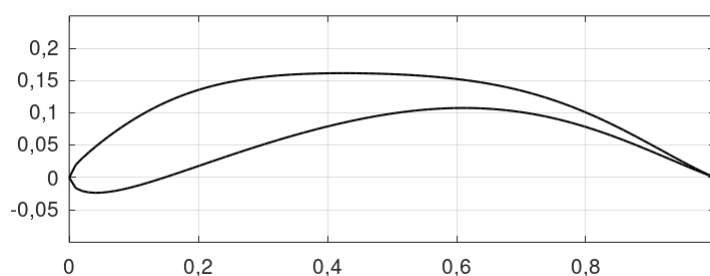


Figura 59 – Aerofólio CST obtido em um trabalho anterior (FERNANDES; JÚNIOR, 2021)

Existe uma gama extensa de fatores que poderiam ser importantes em outros estudos de otimização. Por exemplo, pode ser realizado o uso dos parâmetros das condições de voo como variáveis no problema. Otimizações para gerar geometrias NLF (*Natural Laminar Flow*, Escoamento laminar natural) também são extremamente relevantes em ambos os domínios 2D e 3D para a redução de arrasto. Características de estol também são um ponto importantíssimo para otimização. O *design* inverso utilizando a distribuição de coeficientes de pressão também é relativamente comum. Problemas de otimização de

asas são especialmente fascinantes devido ao imenso número de formas que podem ser montados.

Primeiramente, em contraste ao que foi realizado nesta metodologia, a parametrização das plantas das asas pode ser realizada priorizando dados adimensionais, utilizando parâmetros como razão de afilamento e razão de aspecto - essa abordagem é lógica precisamente devido ao emprego de coeficientes aerodinâmicos adimensionais. Similarmente, a planta pode, na verdade, ser definida em termos de várias combinações distintas dos parâmetros geométricos. Um excelente exemplo disso é no OpenVSP: a Fig. 60 mostra o menu de edição da planta de asas. O trecho sob *Section planform* (planta da seção¹) permite, na coluna da esquerda, escolher combinações de três parâmetros para definir a geometria. Um dado algoritmo de otimização de asas pode ser configurado para permitir essas escolhas, ou pode apresentar uma combinação distinta daquela escolhida para este trabalho. É relevante tomar atenção ao fato de que o emprego da API do VSPAERO, apesar do problema de memória descrito na Subseção 3.10.5, torna mais fácil a manipulação das geometrias das asas e a alteração dos parâmetros geométricos que as configuram - basta apenas montar corretamente as funções da API, enquanto que, utilizando o APAME, é preciso alterar fundamentalmente o funcionamento do gerador de malhas para modificar os parâmetros das asas.

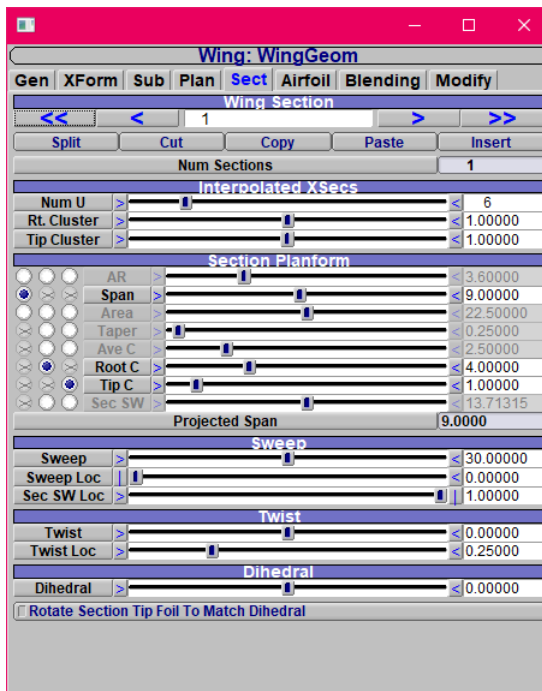


Figura 60 – Menu de edição de seções de plantas de asa no OpenVSP (fonte: autor)

A inclusão de maiores detalhes no formato da planta, como curvas e *splines*, também podem ser pertinentes para um processo de otimização de asas. Similarmente à ideia representada na Fig. 23, tendo um formato base da planta, pode-se especificar detalhes

¹ O termo "seções", neste caso, se refere a seções da planta, e não a seções transversais (os aerofólios)

para aprimorar as características da asa. Isso, no entanto, provavelmente tornaria necessário o uso de solucionadores mais precisos para que de fato as nuances dessas modificações na geometria sejam notadas. Ainda pode-se também incluir o projeto de *winglets*, o *design* de flapes e *slats* e o efeito de interações asa-fuselagem. Características de estabilidade e controle também são de grande importância, assim como a *performance* estrutural da asa.

Finalmente, e não menos importante, deve-se lembrar que existem inúmeros tipos de algoritmos de otimização distintos, cada um com suas respectivas vantagens e desvantagens. Algoritmos genéticos são confiáveis e de fácil implementação, mas não necessariamente os mais rápidos devido à sua estrutura dependente de populações com centenas de indivíduos. Essa questão da velocidade pode ser solucionada utilizando outros tipos de algoritmo ou até mesmo uma versão híbrida do algoritmo genético. Em (DUMAS; DRUEZ; LECERF, 2009), é desenvolvido um algoritmo híbrido, que utiliza ambos os métodos do algoritmo genético e métodos de gradiente, em conjunto, para a otimização de lâminas de motores *turbofan*. Nesse artigo em específico, o algoritmo trabalha alternando entre busca global (na qual o algoritmo genético é superior) e busca local (na qual os métodos em gradiente são superiores.). Os autores concluem que o algoritmo híbrido é mais rápido e é capaz de fornecer melhores resultados que o algoritmo genético independente. O cuidado adicional que se deve tomar com algoritmos híbridos é a configuração correta do código para que seus respectivos métodos de busca atuem precisamente onde precisem atuar, mas, com essa questão sob controle, a hibridização de algoritmos é uma solução inteligente, que já viu diversas aplicações.

Com isso, o autor espera que a metodologia aqui desenvolvida seja útil a quem necessite, tanto utilizando os algoritmos, quanto tomando-os como inspiração para testar suas próprias soluções.

Referências

- ABBOTT, I. H.; DOENHOFF, A. E. V. *Theory of Wing Sections: Including a Summary of Airfoil Data*. New York: Dover Publications, INC, 1959. Citado 4 vezes nas páginas 13, 60, 61 e 133.
- ACHARD, R. *VSP Hangar: E190-E2*. 2019. Acesso em: 31 mar. 2022. Disponível em: <<http://hangar.openvsp.org/vspfiles/404>>. Citado 2 vezes nas páginas 13 e 74.
- ANDERSON, J. D. *Fundamentals of Aerodynamics*. 6. ed. New York: McGraw-Hill, 2017. Citado 11 vezes nas páginas 13, 15, 49, 50, 51, 55, 56, 159, 165, 171 e 175.
- CHENG, C.; HE, S. Optimization of lc circuit parameters for obtaining maximum output of a copper vapor laser by a genetic algorithm. *Microwave and Optical Techonology Letters*, v. 22, p. 343–348, 1999. Citado na página 38.
- COATES, A. *Jane's World Sailplanes and Motor Gliders*. 2. ed. Londres: Jane's Publishing Company, 1980. Citado na página 199.
- CROWN. *Spitfire to fly over Buxton this weekend*. 2021. Acesso em: 30 out. 2021. Disponível em: <<https://www.buxtonadvertiser.co.uk/news/people/spitfire-to-fly-over-buxton-this-weekend-3299192>>. Citado 2 vezes nas páginas 13 e 68.
- DINA, A. et al. Using genetic algorithms to optimize airfoils in incompressible regime. *INCAS BULLETIN*, v. 11, p. 77–90, 2019. Citado 4 vezes nas páginas 17, 39, 119 e 120.
- DRELA, M. Xfoil: An analysis and design system for low reynolds number airfoils. In: . MIT Dept. of Aeronautics and Astronautics, Cambridge, Massachusetts: [s.n.], 1989. Citado 4 vezes nas páginas 13, 65, 66 e 67.
- DRELA, M.; GILES, M. B. Viscous-inviscid analysis of transonic and low reynolds number airfoils. *American Institute of Aeronautics and Astronautics Journal*, v. 25, p. 1347–1355, 1987. Citado na página 67.
- DUMAS, L.; DRUEZ, B.; LECERF, N. A fully adaptive hybrid optimization of aircraft engine blades. *Journal of Computational and Applied Mathematics*, v. 232, p. 54–60, 2009. Citado na página 148.
- ERICKSON, L. L. *Panel Methods - An Introduction*. MOffett Field, California, 1990. Citado 2 vezes nas páginas 13 e 55.
- FATHALLAH, E.; HELAL, M. Optimum structural design of deep submarine pressure hull to achieve minimum weight. In: . Civil Engineering Department, M.T.C. Kobry Elkobba - Cairo, Egypt: [s.n.], 2016. Citado na página 38.
- FERNANDES, G. W. *TCC2—Otimizacao-de-aerofolios-e-asas*. 2022. Acesso em: 25 abr. 2022. Disponível em: <<https://github.com/gustavo-011/TCC2---Otimizacao-de-aerofolios-e-asas>>. Citado 3 vezes nas páginas 9, 11 e 75.

- FERNANDES, G. W.; JÚNIOR, M. N. D. B. A study on airfoil optimization with genetic algorithms. In: ASSOCIAÇÃO BRASILEIRA DE ENGENHARIA E CIÊNCIAS MECÂNICAS. *26th International Congress of Mechanical Engineering*. Santa Catarina, 2021. Citado 3 vezes nas páginas 15, 41 e 146.
- FERNANDES, G. W.; MCDONALD, R. *Literature/bibliography regarding the methods used by VSPAERO*. 2022. Acesso em: 11 mar. 2022. Disponível em: <https://groups.google.com/g/openvsp/c/M4RrLyia5Ek/m/sMJXHCdMAwAJ?utm_medium=email&utm_source=footer>. Citado na página 74.
- FILKOVIC, D. *Graduate Work*. 2008. Citado 5 vezes nas páginas 13, 71, 72, 73 e 144.
- FILKOVIC, D. Apame - aircraft panel method tutorial: Apame version 3.0. 2010. Citado 2 vezes nas páginas 73 e 181.
- FLYNN, R.; SHERMAN, P. D. Multicriteria optimization of aircraft panels: Determining viable genetic algorithm configurations. *International Journal of Intelligent Systems*, v. 10, p. 987–999, 1995. Citado na página 38.
- GROENWOLD, A. A.; STANDER, N.; SNYMAN, J. A. A regional algorithm for the discrete optimal design of truss structures. *International Journal for Numerical Methods in Engineering*, v. 44, p. 749–766, 1999. Citado na página 37.
- GUDMUNDSSON, S. *General Aviation Aircraft Design: Applied Methods and Procedures*. Oxford: Butterworth-Heinemann, 2014. Citado 3 vezes nas páginas 13, 70 e 121.
- GÓMEZ-BOMBARELLI, R. et al. Automatic chemical design using a data-driven continuous representation of molecules. *ACS Central Science*, v. 4, 2018. Citado na página 37.
- HICKS, R. M.; HENNET, P. A. Wing design by numerical optimization. *AIAA Aircraft Systems and Technology Conference*, v. 15, p. 407–412, 1978. Citado na página 143.
- HORNBY, G. S. et al. Automated antenna design with evolutionary algorithms. *Collection of Technical Papers - Space 2006 Conference*, v. 1, 2006. Citado 2 vezes nas páginas 13 e 37.
- HOUGHTON, E. L. et al. *Aerodynamics for Engineering Students*. 7. ed. Cambridge, MA 02139, United States: Editora HARBRA ltda, 2017. Citado 6 vezes nas páginas 15, 163, 167, 168, 169 e 171.
- JIN, D. X.; CUI, S. M.; FANG, D. G. An efficient representation for radar cattering with the use of the genetic algorithm. *Microwave and Optical Techonology Letters*, v. 15, p. 36–39, 1997. Citado na página 38.
- JUN, L. A genetic algorithm to logistics distribution vehicle routing problem with fuzzy due time. *Sixth International Conference on Fuzzy Systems and Knowledge Discovery*, p. 33–39, 2009. Citado na página 37.
- JUN, S. et al. Application of collaborative optimization using genetic algorithm and response surface method to an aircraft wing design. *Journal of Mechanical Science and Technology*, v. 20, p. 133–146, 2006. Citado na página 143.

- JęDRZEJEWSKI, M.; POĆWIERZ, M.; ZIELONKO-JUNG, K. The problem of airflow around building clusters in different configurations. *Archive of Mechanical Engineering*, v. 64, p. 401–418, 2017. Citado na página 47.
- KINNEY, D. *2020 VSP WS: VSPAERO Theory, Validation, and Features*. 2020. Acesso em: 31 mar. 2022. Disponível em: <<https://www.youtube.com/watch?v=-zQEmmr6ck>>. Citado na página 74.
- KULFAN, B. M.; BUSSOLETTI, J. E. "fundamental" parametric geometry representations for aircraft component shapes. In: . Seattle, Washington: [s.n.], 2006. Citado 3 vezes nas páginas 61, 88 e 190.
- KÖRPE, D. S.; KANAT, Aerodynamic optimization of a uav wing subject to weight, geometric, root bending moment, and performance constraints. *International Journal of Aerospace Engineering*, v. 2019, 2019. Citado na página 120.
- LARSON, D. L. *Class assignment leads to published research*. 2021. Acesso em: 1 nov. 2021. Disponível em: <<https://aerospace.illinois.edu/news/33220>>. Citado 2 vezes nas páginas 13 e 58.
- LEDNICER, D. *The Incomplete Guide to Airfoil Usage*. 2010. Acesso em: 17 set. 2021. Disponível em: <<https://m-selig.ae.illinois.edu/ads/aircraft.html>>. Citado na página 61.
- LEITHOLD, L. *O Cálculo com Geometria Analítica*. São Paulo: Editora HARBRA Ltda, 1986. v. 1. Citado 2 vezes nas páginas 13 e 36.
- LIASCH, J. *North American X-15: o avião mais veloz da história*. 2018. Acesso em: 30 out. 2021. Disponível em: <<http://culturaaeronautica.blogspot.com/2018/02/north-american-x-15-o-aviao-mais-veloz.html>>. Citado na página 69.
- LOPES, M. A. G. R. . V. L. da R. *Cálculo Numérico: Aspectos teóricos e computacionais*. 2. ed. São Paulo: MAKRON Books, 1996. Citado na página 190.
- MCDONALD, R. A.; GLOUDEMANS, J. R. Open vehicle sketch pad: An open source parametric geometry and analysis tool for conceptual aircraft design. *American Institute of Aeronautics and Astronautics*, 2022. Citado na página 73.
- MÜNCH, M. *ASW 28-18 ASW 28-18 E: ... A class of its own*. 2021. Acesso em: 30 out. 2021. Disponível em: <<https://www.alexander-schleicher.de/en/flugzeuge/asw-28-18/>>. Citado 2 vezes nas páginas 13 e 59.
- PONTEROSSO, P.; FOX, D. S. J. Optimization of reinforced soil embankments by genetic algorithm. *International Journal for Numerical and Analytical Methods in Geomechanics*, v. 24, p. 425–433, 2000. Citado na página 38.
- RAM, R. K. G. et al. Design optimization and analysis of naca 0012 airfoil using computational fluid dynamics and genetic algorithm. *Applied Mechanics and Materials*, v. 664, p. 111–116, 2014. Citado na página 38.
- RAO, S. S. *Engineering Optimization: Theory and Practice*. 4. ed. New Jersey: John Wiley & Sons, INC, 2009. Citado 3 vezes nas páginas 14, 44 e 84.

RAYMER, D. P. *Aircraft Design: A Conceptual Approach*. 6. ed. Reston, Virginia: American Institute of Aeronautics and Astronautics, 2018. Citado 4 vezes nas páginas 13, 70, 71 e 91.

SAAD, Z. *Optimization of an airfoil shape using Genetic algorithm(GA)*. 2018. Acesso em: 17 set. 2021. Disponível em: <<https://www.mathworks.com/matlabcentral/fileexchange/69095-optimization-of-an-airfoil-shape-using-genetic-algorithm-ga>>. Citado na página 40.

SCOTT, J. *NACA Airfoil Series*. 2001. Acesso em: 17 set. 2021. Disponível em: <<http://www.aerospaceweb.org/question/airfoils/q0041.shtml>>. Citado na página 60.

SELIG, M. *UIUC Airfoil Coordinates Database*. 2022. Acesso em: 25 mar. 2022. Disponível em: <https://m-selig.ae.illinois.edu/ads/coord_database.html>. Citado na página 199.

SOEMARWOTO, B. I. The variational method for aerodynamic optimization using the navier-stokes equations. *ICASE*, 1997. Citado na página 37.

SOEMARWOTO, B. I. Inverse aerodynamic shape design using the navier-stokes equations. *Inverse Problems in Engineering Mechanics*, v. 1, p. 437–446, 1998. Citado na página 37.

SZUSTAKOWSKI, J. D.; WENG, Z. Protein structure alignment using a genetic algorithm. *PROTEINS: Structure, Function, and Genetics*, v. 38, p. 428–440, 2000. Citado na página 38.

WIKIPEDIA. *Vazamento de memória*. 2020. Acesso em: 22 abr. 2022. Citado na página 132.

WIKIPEDIA. *Airflow separating from a wing at a high angle of attack*. 2021. Acesso em: 30 out. 2021. Disponível em: <[https://en.wikipedia.org/wiki/Flow_separation#/media/File:1915ca_abger_fluegel_\(cropped_and_mirrored\).jpg](https://en.wikipedia.org/wiki/Flow_separation#/media/File:1915ca_abger_fluegel_(cropped_and_mirrored).jpg)>. Citado 2 vezes nas páginas 15 e 175.

WIKIPEDIA. *Diagram from Reynolds's 1883 paper showing onset of turbulent flow*. 2021. Acesso em: 30 out. 2021. Disponível em: <https://en.wikipedia.org/wiki/Reynolds_number#/media/File:Reynolds_observations_turbulence_1883.svg>. Citado 2 vezes nas páginas 15 e 175.

YANG, X. *Engineering Optimization: An Introduction with Metaheuristic Applications*. New Jersey: John Wiley & Sons, Inc, 2010. Citado 4 vezes nas páginas 13, 35, 44 e 46.

YANG, Z.; YANG, G. Optimization of aircraft maintenance plan based on genetic algorithm. *Physics Procedia*, v. 33, p. 580–586, 2012. Citado na página 38.

YARPIZ. *Introduction to Genetic Algorithms - Practical Genetic Algorithms Series*. 2020. Acesso em: 18 jul. 2020. Disponível em: <<https://www.youtube.com/watch?v=Fdk7ZKJHFcI&list=PLgH3sgdvgO4TQagjfi1FmPcdBjllmk5s9>>. Citado na página 44.

ZHU, F.; QIN, N. Intuitive class/shape function parameterization for airfoils. *American Institute of Aeronautics and Astronautics Journal*, v. 52, p. 17–25, 2014. Citado na página 65.

ZIADEH, R. *Yabhon United 40 MALE UCAV*. 2021. Acesso em: 30 out. 2021. Disponível em: <https://twitter.com/rani_ziadeh/status/1364322396656779269>. Citado na página 69.

Apêndices

APÊNDICE A – Análise dimensional

Ao se utilizar a matemática como uma ferramenta para descrever fenômenos físicos, é preciso empregar unidades para atribuir significados físicos aos números manipulados e obtidos nos resultados. Tendo isso em vista, é necessário que as relações entre as unidades nas equações sejam cabíveis com a lógica, e que as unidades obtidas ao final dos cálculos estejam de acordo com as grandezas que se deseja obter. Um exemplo simples é a equação da velocidade escalar média:

$$V = \frac{\Delta S}{\Delta t} \quad (\text{A.1})$$

Onde V é a velocidade de uma partícula e ΔS é a distância percorrida por essa partícula dentro de um período Δt de tempo. Tomando as unidades do Sistema Internacional, sabe-se que as unidades de velocidade, distância e tempo são, respectivamente, metro por segundo (m/s), metro (m), e segundo (s). Assim, é intuitiva a ideia de que obtém-se uma grandeza de velocidade ao dividir uma grandeza de distância por uma grandeza de tempo. Igualmente intuitivo seria também entender que é incorreto realizar a soma de duas grandezas com unidades distintas - é impossível obter uma resposta coerente para, por exemplo, a pergunta "qual é o resultado de 5 pascais mais 5 volts?".

Existem inúmeras outras relações entre unidades que não são tão intuitivas quanto o exemplo da velocidade média, mas são facilmente deduzíveis com relações fundamentais da Física. Um exemplo vem de um desenvolvimento da segunda lei de Newton:

$$F = ma \quad (\text{A.2})$$

Onde F é uma força em Newtons (N), m é uma massa em quilogramas (kg) e a é uma aceleração em metros por segundo ao quadrado (m/s²). Portanto, em uma dada equação, uma grandeza de massa multiplicada por uma grandeza de aceleração é equivalente a uma grandeza de força.

Tomando tudo o que foi exposto até aqui, apresenta-se agora um exemplo. A equação abaixo é obtida inserindo-se a Eq. 2.2 na Eq. 2.1:

$$L = \frac{1}{2} \rho_{\infty} V_{\infty}^2 S C_L \quad (\text{A.3})$$

Para certificar-se que essa equação tem sentido físico, é feita uma análise dimensional. Tomando apenas o seu lado direito e substituindo as variáveis por suas respectivas unidades, temos

$$\left(\frac{\text{kg}}{\text{m}^3}\right) \left(\frac{\text{m}}{\text{s}}\right)^2 (\text{m}^2) \quad (\text{A.4})$$

Enfatiza-se que o coeficiente de sustentação é adimensional. Simplificando as unidades de metro, o resultado é

$$\text{kg} \frac{\text{m}}{\text{s}^2} \quad (\text{A.5})$$

O qual consiste de uma unidade de massa multiplicada por uma de aceleração. Como demonstrado acima, isso resulta em uma força (neste caso, em N), o que prova que a equação está correta - afinal, L é a força de sustentação.

APÊNDICE B – Função de corrente

Linhas de corrente representam a trajetória das partículas de um fluido, apresentando, em suas tangentes, as direções das velocidades das partículas. Em escoamentos incompressíveis, irrotacionais e em regime permanente as linhas de corrente relacionam-se diretamente à função de corrente, cuja definição é mostrada a seguir.

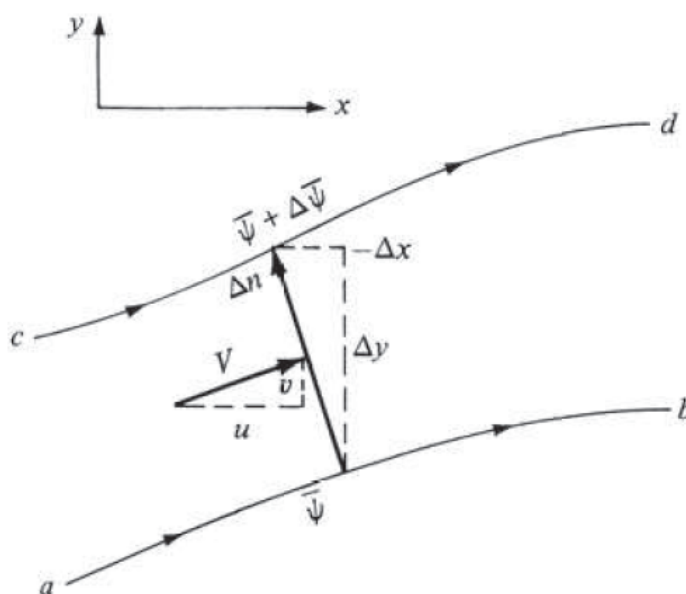


Figura 61 – Escoamento passando por entre duas linhas de corrente distintas (ANDERSON, 2017)

Assumem-se duas linhas de corrente. A primeira, ab , é definida por ψ e a outra, cd , por $\psi + \Delta\psi$. O objetivo é calcular o fluxo de massa de fluido passando entre ambas as linhas. Como ilustrado na Fig. 61, o vetor velocidade e a distância local entre as linhas pode ser decomposta. Assim, a quantidade de fluido passando através de Δn é

$$\Delta\psi = u \Delta y - v \Delta x \quad (\text{B.1})$$

Tomando limites tendendo a zero, a Eq. B.1 torna-se

$$d\psi = u dy - v dx \quad (\text{B.2})$$

O diferencial total da função de corrente é

$$d\psi = \frac{\partial\psi}{\partial x} dx + \frac{\partial\psi}{\partial y} dy \quad (\text{B.3})$$

Desse modo, comparando os termos das Equações B.2 e B.3, temos que

$$u = \frac{\partial \psi}{\partial y} \quad (\text{B.4})$$

$$v = -\frac{\partial \psi}{\partial x} \quad (\text{B.5})$$

A função de corrente é uma função escalar que contém dados da velocidade do fluido, cujas componentes são obtíveis a partir da derivação da função, como se vê nas Equações B.4 e B.5.

APÊNDICE C – Potencial de velocidade

Considera-se novamente um escoamento com as mesmas características daquele apresentado no apêndice. A definição da condição de irrotacionalidade de um fluido é

$$\xi = \nabla \times \mathbf{V} = 0 \quad (\text{C.1})$$

Onde ξ é o vetor vorticidade. Assumindo uma função escalar ϕ , temos que

$$\nabla \times (\nabla\phi) = 0 \quad (\text{C.2})$$

Comparando as Equações C.1 e C.2, se deduz, portanto, que

$$\mathbf{V} = \nabla\phi \quad (\text{C.3})$$

Define-se, assim, o potencial de velocidade ϕ , que, similarmente à função de corrente, também permite calcular as componentes da velocidade por meio da sua diferenciação:

$$u = \frac{\partial\phi}{\partial x} \quad (\text{C.4})$$

$$v = \frac{\partial\phi}{\partial y} \quad (\text{C.5})$$

APÊNDICE D – Equação de Laplace

A equação da continuidade para um fluido incompressível é

$$\nabla \cdot \mathbf{V} = \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0 \quad (\text{D.1})$$

Ao expandir o produto vetorial da equação C.1, obtém-se

$$\nabla \times \mathbf{V} = \frac{\partial u}{\partial z} \hat{j} + \frac{\partial v}{\partial x} \hat{k} - \frac{\partial u}{\partial y} \hat{k} - \frac{\partial v}{\partial z} \hat{i} \quad (\text{D.2})$$

Aplicando a condição de irrotacionalidade, da expansão apresentada na Eq. D.2 interessa apenas as componentes na direção z :

$$\xi_z = \frac{\partial v}{\partial x} - \frac{\partial u}{\partial y} = 0 \quad (\text{D.3})$$

Substituindo as Equações B.4 e B.5 na Eq. D.3, temos

$$\frac{\partial^2 \psi}{\partial x^2} + \frac{\partial^2 \psi}{\partial y^2} = 0$$

$$\nabla^2 \psi = 0 \quad (\text{D.4})$$

E substituindo as Equações C.4 e C.5 na Eq. D.1, temos

$$\frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2} = 0$$

$$\nabla^2 \phi = 0 \quad (\text{D.5})$$

Ambas as Equações D.4 e D.5 são equações de Laplace, para a qual ambas as funções ψ e ϕ são soluções (HOUGHTON et al., 2017).

APÊNDICE E – Circulação

Tomando um escoamento qualquer, assume-se um contorno fechado C de formato arbitrário fechando um trecho do escoamento, como ilustrado na Fig. 62.

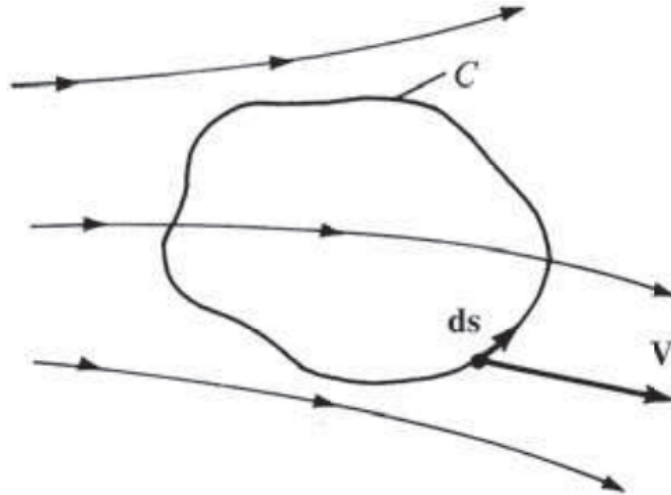


Figura 62 – Curva fechada em um dado escoamento para o cálculo da circulação (ANDERSON, 2017)

A circulação dentro desse contorno é dada por

$$\Gamma = - \oint_C \mathbf{V} \cdot d\mathbf{s} \quad (\text{E.1})$$

Onde o sinal negativo tem o propósito de inverter a convenção original do Cálculo, que considera positiva uma rotação no sentido anti-horário. Na aerodinâmica é mais conveniente que o sentido positivo seja horário. A circulação também pode ser definida em termos da vorticidade:

$$\Gamma = - \iint_S (\nabla \times \mathbf{V}) \cdot d\mathbf{S} \quad (\text{E.2})$$

O conceito de circulação está intimamente relacionado à geração de sustentação por meio do teorema de Kutta–Joukowski, definido como

$$L' = \rho_\infty V_\infty \Gamma \quad (\text{E.3})$$

Onde L' é a sustentação por unidade de envergadura. Traçando uma curva que contenha um corpo bidimensional qualquer, como um aerofólio, a circulação pode ser

calculada com a Eq. E.1 e, caso ela seja um valor finito não nulo, significa que esse corpo gera uma força de sustentação, calculada por meio da Eq. E.3.

APÊNDICE F – Escoamentos fundamentais

Para escoamentos sem separação, o comportamento do fluxo de ar pode ser descrito pela função de corrente e/ou o potencial de velocidade. A equação de Laplace é linear, o que permite a superposição de soluções. Isto é, a soma de soluções distintas da equação, referentes a tipos distintos de escoamento, também forma uma solução válida. Desse modo, é possível descrever escoamentos complexos em termos de escoamentos fundamentais, os quais serão apresentados a seguir.

O primeiro escoamento fundamental é o escoamento uniforme, definido em termos de ψ e ϕ nas Equações F.1 e F.2 respectivamente. O escoamento uniforme é definido por um fluxo de ar constante, como ilustrado na Fig. 63.

$$\psi = uy - vx \quad (\text{F.1})$$

$$\phi = ux + vy \quad (\text{F.2})$$

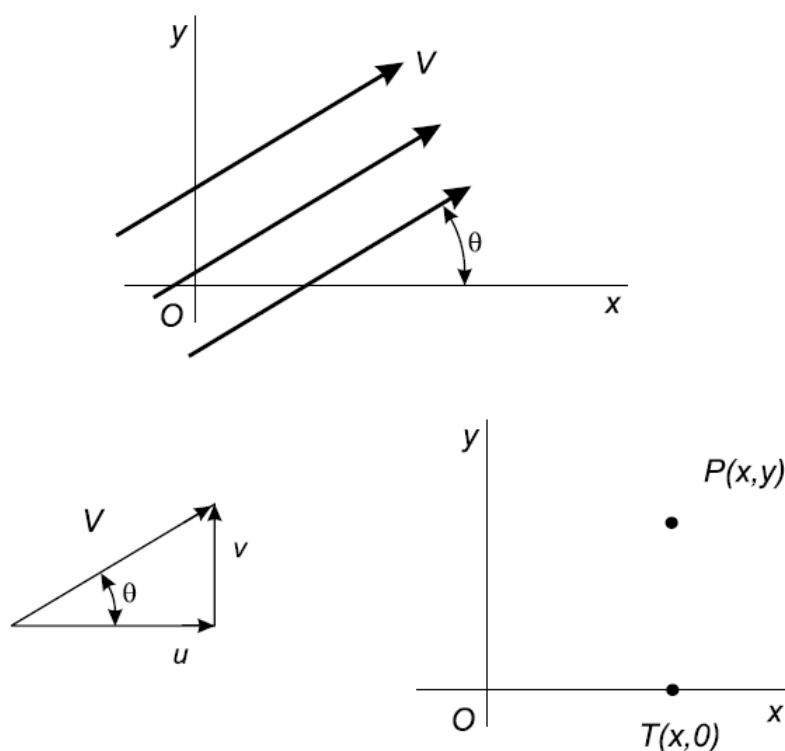


Figura 63 – Escoamento uniforme orientado a uma direção arbitrária (HOUGHTON et al., 2017)

A seguir estão os escoamentos em fontes e sumidouros. Uma fonte é definida por um ponto do qual se origina fluido a uma taxa constante Λ , o qual flui radialmente para fora da fonte. A fonte é matematicamente representada pelas Equações F.3 e F.4, e uma ilustração é apresentada na Fig. 64. Uma fonte é convertida em sumidouro quando Λ é negativo. Sumidouros funcionam da mesma forma que fontes, com a diferença de que o sentido do escoamento é invertido - isto é, o fluido segue radialmente ao centro do sumidouro, onde ele "some".

$$\psi = \frac{\Lambda}{2\pi}\theta \quad (\text{F.3})$$

$$\phi = \frac{\Lambda}{2\pi} \ln r \quad (\text{F.4})$$

As Equações F.3 e F.4 estão escritas em coordenadas polares. O raio r e o ângulo θ podem ser descritos em coordenadas cartesianas como

$$r = \sqrt{x^2 + y^2} \quad (\text{F.5})$$

$$\theta = \tan^{-1} \frac{y}{x} \quad (\text{F.6})$$

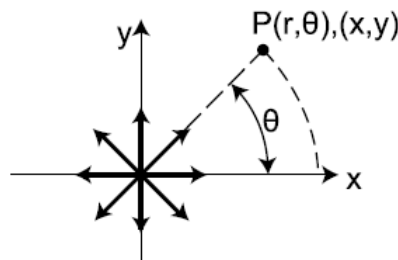


Figura 64 – Escoamento surgindo de uma fonte (HOUGHTON et al., 2017)

Finalmente, há o escoamento em vórtice, cuja intensidade é ditada por Γ , no qual o fluido flui em círculos concêntricos centrados em um eixo imaginário, como ilustrado na Fig. 65. As respectivas soluções da função de corrente e do potencial de velocidade são as Equações F.7 e F.8. Escoamentos em vórtice são especialmente importantes para a análise de fluxos que geram sustentação, porque, como explicado no Apêndice E, a circulação está diretamente associada à existência de forças de sustentação, e a circulação por sua vez só existe se o escoamento tiver vórtices. Essas noções são especialmente relevantes na implementação da teoria do método de painéis, como apresentado na Subseção 2.3.2.

$$\psi = \frac{\Gamma}{2\pi} \ln r \quad (\text{F.7})$$

$$\phi = -\frac{\Gamma}{2\pi} \theta \quad (\text{F.8})$$

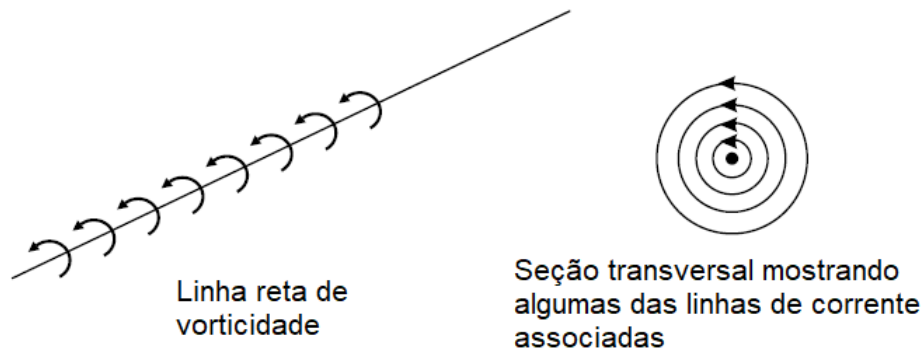


Figura 65 – escoamento em vórtice (HOUGHTON et al., 2017)

É importante notar que os escoamentos de fonte e sumidouro são, de fato, aplicáveis aos escoamentos estudados neste contexto, porque seus fluxos são incompressíveis em todos os pontos exceto seus centros. Fontes e sumidouros também geram escoamentos irrotacionais. Similarmente, o escoamento de vórtice também cumpre ambos os requisitos, exceto em seu centro. Adicionalmente, enfatiza-se a distinção entre a rotacionalidade de um fluido e as propriedades do escoamento em vórtice. A rotacionalidade, definida como o produto vetorial do operador nabla com o vetor velocidade do fluido (Eq. C.1), trata da rotação dos elementos de fluido ao longo de suas linhas de corrente. Por outro lado, o escoamento em vórtice enuncia apenas que as linhas de corrente têm formatos de círculos concêntricos, nos quais o fluido flui com velocidade constante.

APÊNDICE G – Condição de Kutta

Para um determinado corpo dentro de um escoamento que gera sustentação, existe um número infinito de soluções das equações do potencial de velocidade, cada uma correspondendo a um valor de circulação (ANDERSON, 2017). Desse modo, é necessária alguma limitação adicional que garanta que a solução teórica seja igual àquela fornecida pela natureza. Essa limitação é a condição de Kutta.

O lado esquerdo da Fig. 66 apresenta um aerofólio em um determinado ângulo de ataque. Os pontos S_F e S_A indicam pontos de estagnação do ar. A posição do ponto S_A no extradorso faz com que o ar curve-se ao redor do bordo de fuga. No entanto, porque o escoamento é invíscido, o ar adquire velocidades infinitamente grandes ao passar ao redor da geometria afiada do bordo de fuga. Essa situação se repetiria caso fosse imposto um valor de circulação que movesse o ponto S_A para o intradorso.

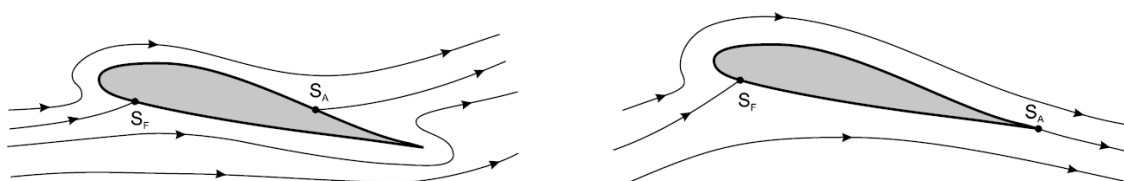


Figura 66 – Efeito do cumprimento da condição de Kutta no escoamento ao redor do aerofólio (HOUGHTON et al., 2017)

Desse modo, conclui-se que a única solução realista para o escoamento ao redor do aerofólio é aquela em que o ponto de estagnação S_A encontra-se precisamente no bordo de fuga, de modo que o ar flui suavemente para fora do mesmo, como visto no lado direito da Fig. 66. Para cumprir a condição, as velocidades do ar acima e abaixo do aerofólio anulam-se no bordo de fuga, caso este tenha um ângulo finito, ou tornam-se iguais, caso o bordo de fuga tenha formato pontiagudo. A Fig. 67 ilustra essas circunstâncias, assumindo V_1 como a velocidade do ar sobre o extradorso e V_2 como a velocidade do ar sob o intradorso.

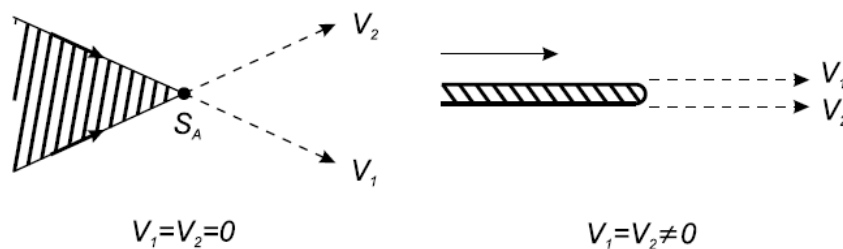


Figura 67 – Relações das velocidades do ar quando a condição de Kutta é satisfeita (HOUGHTON et al., 2017)

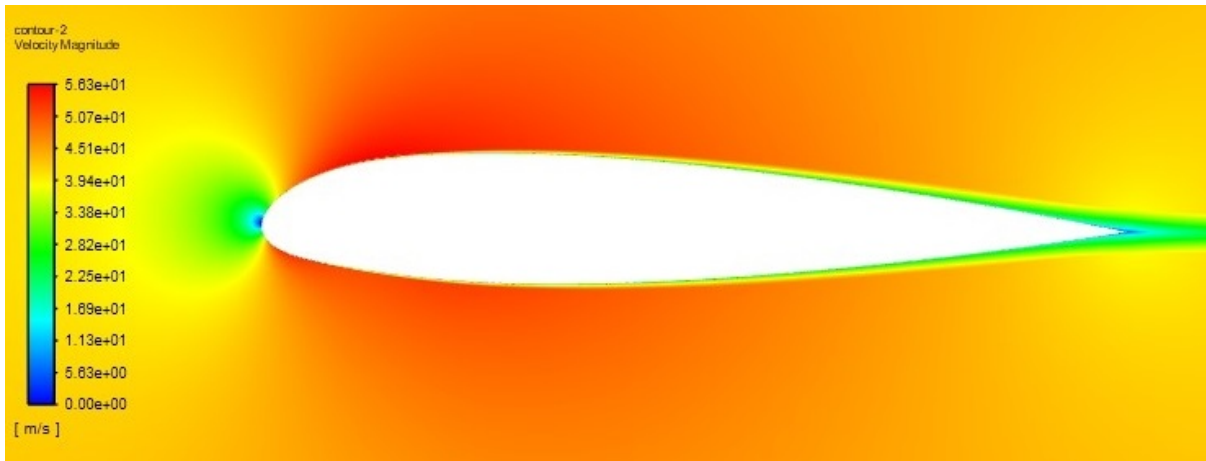
APÊNDICE H – Viscosidade, camada limite e o número de Reynolds

A viscosidade do ar faz com que o deslocamento do mesmo seja desacelerado quando este entra em contato com um corpo sólido. Diretamente na superfície desse corpo a velocidade do fluido é igual a zero, o que é determinado como a condição de não deslizamento. À medida que se afasta da superfície sólida, o fluido passa a ter velocidades cada vez maiores, até que se alcança um determinado ponto em que a velocidade do ar é igual à velocidade do vento relativo. Essa região com gradientes de velocidade, dominada por efeitos viscosos, é denominada camada limite. A Fig. 68a mostra um aerofólio NACA 23015 em uma simulação CFD, na qual a presença da camada limite fica clara, e a Fig. 68b apresenta uma vista em *zoom* na superfície do aerofólio, na qual é possível averiguar que a velocidade é de fato nula na superfície.

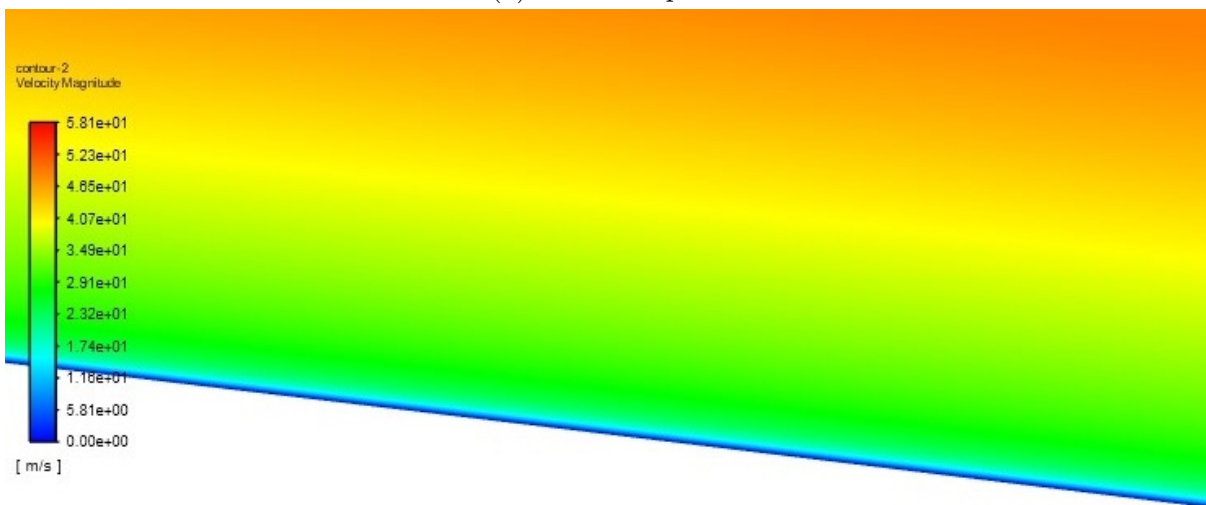
A aproximação de fluidos invíscidos é justificável porque, como visto na Fig. 68a, a camada limite é uma região fina ao longo da superfície do aerofólio, enquanto que o resto do escoamento ao redor é pouquíssimo afetado por efeitos viscosos. É por isso que a análise invíscida de aerofólios produz resultados realísticos para os coeficientes de sustentação, pois as diferenças de pressão ao redor do aerofólio que geram sustentação são causadas majoritariamente pelo escoamento fora da camada limite. Por outro lado, a estimativa sensata dos coeficientes de arrasto precisa incluir efeitos viscosos, pois eles são uma das principais contribuições para o arrasto sobre qualquer objeto dentro de um fluxo de fluido.

A viscosidade traz ainda consigo outra distinção fundamental na forma de escoamentos laminares e turbulentos. Escoamentos laminares se comportam similarmente aos invíscidos, isto é, com linhas de corrente uniformes e organizadas. Por outro lado, escoamentos turbulentos são caracterizados por movimento desordenado do fluido, com a mistura de linhas de corrente adjacentes. O primeiro a documentar cientificamente essas tendências foi Osborne Reynolds em 1883. Seu experimento consistiu em um aparato com um tubo de água limpa com um filete de água tingida no centro. O filete permitia a visualização do comportamento da água: em velocidades baixas, o filete mantinha-se imperturbado, ao passo de que, com o aumento da velocidade, o filete saía de sua trajetória uniforme, passando a misturar-se com a água transparente ao redor, na transição de escoamento laminar para turbulento. A Fig. 69 mostra esses comportamentos.

Com as propriedades do fluido é possível estimar o ponto de transição utilizando um coeficiente adimensional popularizado por Reynolds que mais tarde levou seu nome. O número de Reynolds é definido como



(a) Vista completa



(b) Vista em detalhe

Figura 68 – Contornos de velocidade de uma simulação CFD (fonte: autor)

$$Re = \frac{\rho_{\infty} V_{\infty} l}{\mu_{\infty}} \quad (\text{H.1})$$

Onde l é um comprimento característico e μ_{∞} é a viscosidade dinâmica do fluido. O número de Reynolds é uma razão entre as forças inerciais e forças viscosas do fluido, onde a primeira tende a transicionar o escoamento para turbulento e a segunda tende a mantê-lo em estado laminar.

A obtenção de informações a respeito das propriedades de escoamento laminar e turbulento e as tendências de transição são essenciais para o projeto ideal de superfícies aerodinâmicas. Camadas limite laminares geram menos arrasto do que camadas limite turbulentas e também têm mais facilidade de permanecer coladas à superfície. O fenômeno de separação, especialmente tratando do estol de aerofólios e asas, é dominado predominantemente por escoamento turbulento, caótico, como ilustrado na Fig. 70 - esse é o motivo pelo qual métodos de painéis tradicionais não serem compatíveis com separação.

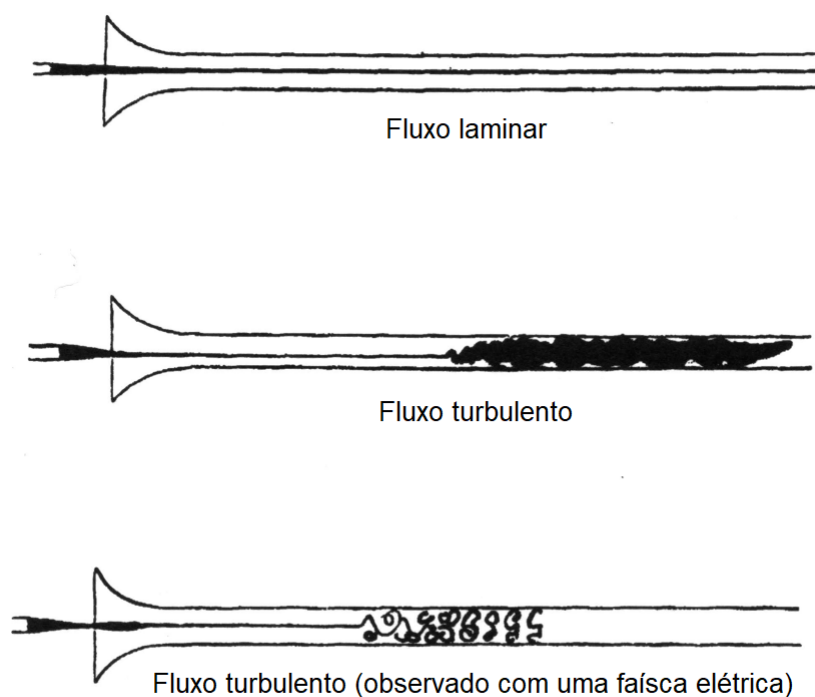


Figura 69 – Resultados do experimento de Reynolds referente a escoamentos laminares e turbulentos dentro de um tubo ([WIKIPEDIA, 2021b](#))



Figura 70 – Experimento em túnel de vento demonstrando a separação do ar sobre um aerofólio ([WIKIPEDIA, 2021a](#))

O número de Reynolds é também uma ferramenta poderosa no quesito de similaridade de escoamento. Um modelo em escala de uma aeronave em um túnel de vento terá os mesmos coeficientes aerodinâmicos que a aeronave real em voo desde que os números de Mach e de Reynolds no túnel de vento sejam iguais aos respectivos valores da aeronave real ([ANDERSON, 2017](#)). É essa a propriedade que permite testes válidos de modelos de aeronaves em tamanhos distintos àqueles previstos nos projetos, e é também uma demonstração da elegância dos coeficientes adimensionais.

APÊNDICE I – Modelos dos arquivos de entrada para o XFOIL

Os algoritmos utilizam arquivos de texto contendo todos os comandos necessários para a análise no XFOIL. Os modelos desses arquivos são apresentados abaixo. Deve-se enfatizar que as linhas vazias, indicadas com "[VAZIO]", também são importantes ao arquivo, porque elas são interpretadas no XFOIL como teclas *enter* (quebras de linha).

O XFOIL não é *case-sensitive* em seus comandos. No entanto, a distinção de caracteres maiúsculos e minúsculos é necessária na especificação de arquivos externos como, por exemplo, arquivos de coordenadas.

I.1 Algoritmo NACA 4 dígitos

O exemplo de modelo de *input* abaixo configura o XFOIL para simular um perfil NACA 2412 em um número de Reynolds igual a $1e5$ e em zero ângulo de ataque. Opcionalmente, foi especificado um número de iterações igual a 50.

```
PLOP
G
[VAZIO]
NACA 2412
OPER
VISC 100000
PACC
polar1.txt
[VAZIO]
ITER 50
ALFA 0.000000
QUIT
```

1.2 Algoritmo CST e CST específico

O exemplo de modelo de *input* abaixo configura o XFOIL para simular um dado perfil CST em um número de Reynolds igual a $1e5$ e em zero ângulo de ataque.

```
PLOP
G
[VAZIO]
LOAD coordenadas.dat
[VAZIO]
OPER
VISC 5000000
PACC
polar1.txt
[VAZIO]
ALFA 0.000000
[VAZIO]
QUIT
```

APÊNDICE J – Modelos dos arquivos de entrada para o APAME

Os arquivos de entrada no APAME são arquivos de texto com a extensão *.inp*, e contém informações a respeito das condições de voo, da geometria, e das opções desejadas para os resultados oferecidos ao final dos cálculos. O modelo é apresentado abaixo, onde o símbolo "#" indica o início de um comentário. Os detalhes do arquivo de entrada serão explicados por meio dos comentários.

```

APAME input file VERSION 3.1
# «Parâmetros do escoamento»
AIRSPEED 27778 # Velocidade
DENSITY 0.000000000001225 # Densidade
PRESSURE 0.101325 # Pressão atmosférica
# Correção de prandtl-glauert
# 0-Sem correção
# *-Mach number
MACH 0
CASE_NUM 1 # Número de casos
0 # Ângulos de ataque [graus]
0 # Ângulos de derrapagem [graus]

# «Valores de referência»
WINGSPAN 42400 # Envergadura
MAC 6000 # Corda média aerodinâmica
SURFACE 222000000 # Área alar
# Ponto de referência
ORIGIN *
0 0 0

# «Parâmetros do solucionador»

```

```
# Método de singularidade
# 0-Fonte/sumidouro constante
# 1-Dipolo constante
METHOD 0
ERROR 0.0000001 # Erro
COLLDIST 0.0000001 # Profundidade do ponto de colocação
FARFIELD 5 # Coeficiente de campo distante
# Cálculo do ponto de colocação
# 0-Aproximado
# 1-Preciso
COLLCALC 0
# Método de interpolação/ordem para cálculos de velocidade
# 0-Nodal
# 1-Primeiro
# 2-Segundo
VELORDER 1

# «Opções dos resultados»
# 0-Não
# 1-Sim
RESULTS 1
RES_COEF 1 # Coeficientes
RES_FORC 0 # Forças
RES_GEOM 0 # Geometria
RES_VELO 1 # Velocidade
RES_PRES 1 # Pressão
RES_CENT 0 # Pontos de centro
RES_DOUB 1 # Valores dos dipolos
RES_SORC 1 # Valores das fontes
RES_VELC 1 # Componentes da velocidade
```

```
RES_MESH 0 # Características da malha
RES_STAT 0 # Pressão estática
RES_DYNA 0 # Pressão dinâmica
RES_MANO 1 # Pressão manométrica

# «Geometria»
# Nós
NODES N # Onde N é o número de nós
# [coordenadas dos nós - x, y ,z]

PANELS P # Onde P é o número de painéis
# [Informações dos painéis]
```

Há três tipos de elementos no APAME: dois são elementos de superfície, que podem ser retangulares ou triangulares, e elementos de esteira, que são apenas retangulares. No caso de elementos de superfície retangulares, as informações dos painéis são dadas por nove valores separados por vírgulas, onde o primeiro valor identifica o tipo de elemento (neste caso, é igual a 1), os quatro seguintes indicam os nós que formam o elemento e os quatro últimos indicam os números dos painéis adjacentes. Para elementos de superfície triangulares, a organização é a mesma, com a diferença de que o valor de identificação é igual a 2, e os números de nós e painéis adjacentes são ambos iguais a 3. Os painéis de esteira são definidos por sete valores: o primeiro, novamente a identificação do tipo de painel, é igual a 10, os quatro valores seguintes informam os nós que formam o elemento e os dois últimos dois valores são os dois painéis de superfície adjacentes, aos quais se deve aplicar a condição de Kutta. Desse modo, é nítido que elementos de esteira são aplicados apenas a corpos geradores de sustentação.

A definição dos nós de cada painel deve ser feita de modo que, usando a regra da mão direita, a ordem dos nós faça com que o vetor normal da superfície aponte para fora. Estas e outras instruções estão disponíveis em ([FILKOVIC, 2010](#)).

APÊNDICE K – Formatação das coordenadas no algoritmo CST

Tendo em mente que todos os aerofólios têm corda unitária, a formatação das coordenadas é a seguinte: as coordenadas começam no bordo de fuga, no ponto $(1, 0)$, seguem ao longo do extradorso, chegam ao bordo de ataque, no ponto $(0, 0)$, e voltam ao bordo de fuga seguindo pelo intradorso. Essa formatação é resumida abaixo.

```

1 0
...
[Coordenadas do extradorso]
...
0 0
...
[Coordenadas do intradorso]
...
1 0

```

Desse modo, todo aerofólio gerado é um contorno fechado, pelo menos nessa configuração. O CST também permite a geração de perfis com bordo de fuga não fechado. Para isso, basta apenas atribuir valores não nulos às variáveis $\Delta Z_{TE,ex}$ e $\Delta Z_{TE,in}$, o que faria com que as ordenadas do primeiro e do último ponto fossem não nulas.

Outro aspecto fundamental que cabe à geração de coordenadas é a devida distribuição das abscissas. É vital que a concentração de pontos no bordo de ataque seja alta para que as nuances da curvatura sejam bem representadas quando o aerofólio é convertido em um conjunto de painéis quando importado ao XFOIL ou qualquer outro *software* de simulação. Esse requisito é cumprido por meio da metade de uma distribuição cosseno ao invés de uma distribuição linear. Um exemplo é apresentado na Fig. 71.

Em nota, a formatação de coordenadas apresentada deve ser cumprida por todo aerofólio que seja importado ao XFOIL. Isso vale para coordenadas obtidas prontas, por exemplo, de algum livro ou site, e também para coordenadas geradas por alguma parametrização geométrica. A distribuição de pontos descrita também é essencial para qualquer coordenada de aerofólio.

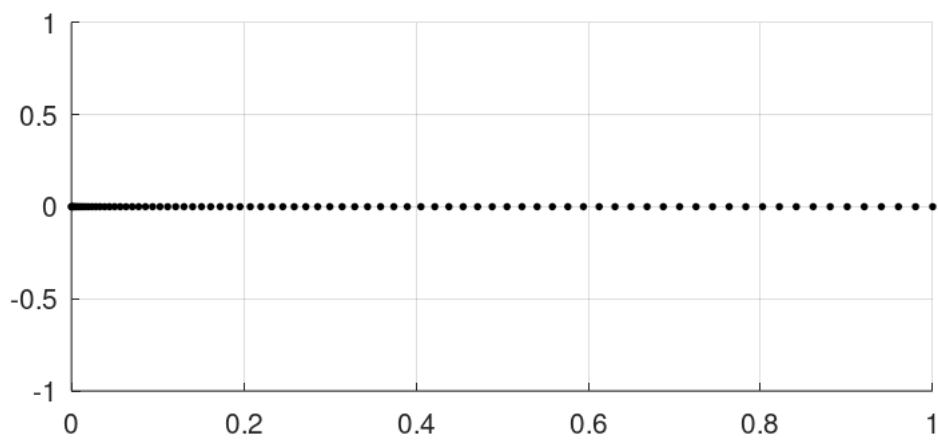


Figura 71 – Metade de uma distribuição cosseno para uso na geração de coordenadas de aerofólio (fonte: autor)

APÊNDICE L – Checagens de qualidade no algoritmo CST (2D e 3D)

A função de checagem de qualidade no algoritmo CST tem o propósito de evitar características indesejáveis nas geometrias dos perfis. Essa função atua nos estágios de geração inicial, *crossover* e mutação, os quais alteram as variáveis de design dos aerofólios. São quatro requisitos distintos, dos quais todos devem ser cumpridos pelo aerofólio. Sendo assim, caso um dado aerofólio falhe em qualquer uma das avaliações, o algoritmo já o toma como reprovado nas checagens e o processo de onde ele veio (geração inicial, *crossover* ou mutação) é refeito.

A primeira checagem é a mais fundamental para a própria praticabilidade do aerofólio. Não deve haver qualquer interseção das superfícies do aerofólio entre o bordo de ataque e o bordo de fuga. A Fig. 72 apresenta um exemplo que falha esse requisito, cujo aerofólio tem os seguintes valores para seus vetores de configuração:

$$v_ex = [0,01;0,1;0,1;0,1;10;0]$$

$$v_in = [0,01;0,1;-0,6;0,1;10;0]$$

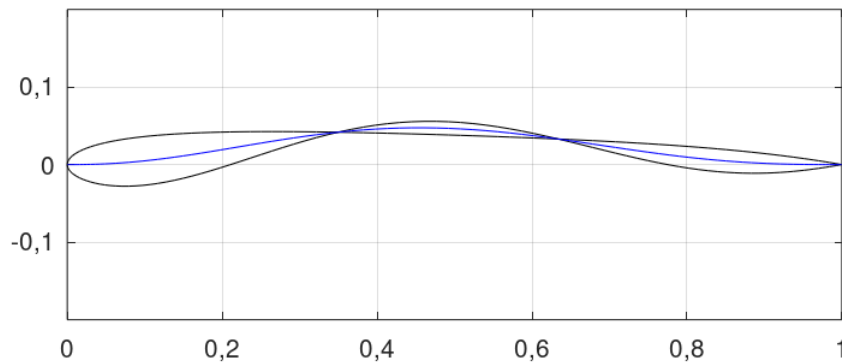


Figura 72 – Perfil CST com interseção de superfícies (fonte: autor)

A segunda checagem trata das mudanças de inclinação do extradorso. O contorno do extradorso pode ter, no máximo, uma mudança de inclinação ao longo de todo o seu comprimento. A Fig. 73 mostra um aerofólio que falha esse requisito, cujos vetores são definidos abaixo:

$$v_ex = [0,01;0,1;0;0,1;20;0]$$

$$v_in = [0,01;0,1;0,1;0,1;10;0]$$

A terceira checagem trata das mudanças de inclinação do intradorso. O contorno

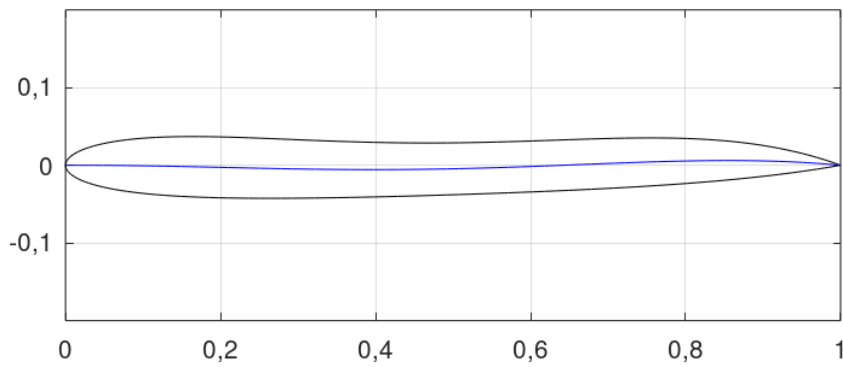


Figura 73 – Perfil CST cujo extradorso tem mais de uma mudança de inclinação (fonte: autor)

do intradorso pode ter, no máximo, duas mudanças de inclinação ao longo de todo o seu comprimento. A Fig. 74 mostra um aerofólio que falha esse requisito, cujos vetores são definidos abaixo:

$$v_ex = [0,01;0,1;0,1;0,1;10;0]$$

$$v_in = [0,01;0,1;-0,1;0,1;20;0]$$

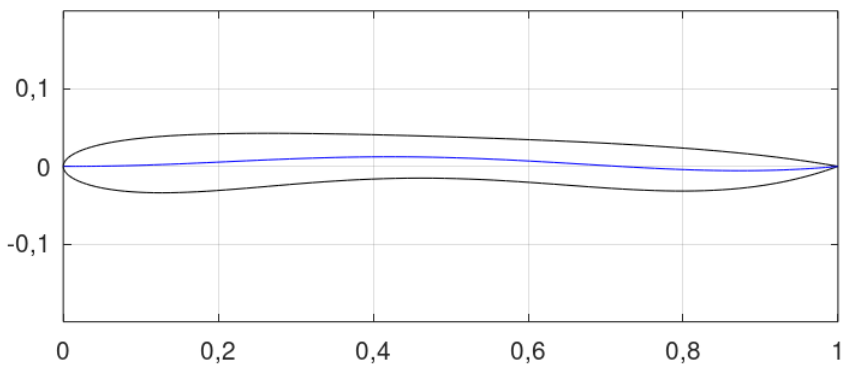


Figura 74 – Perfil CST cujo extradorso tem mais de uma mudança de inclinação (fonte: autor)

As duas últimas checagens foram implementadas investigando-se as geometrias de diversos aerofólios. É evidente que todos, ou a grande maioria, se enquadra dentro das características apresentadas nos últimos parágrafos.

A quarta e última checagem avalia a espessura para que os perfis não sejam finos demais. Para cumprir o requisito, a espessura média do aerofólio dividida pelo comprimento da corda deve ser maior do que 0.04. Esse valor foi determinado em termos de testes não documentados, averiguando qual valor resultaria em uma espessura mínima que fosse razoável, tendo em base aerofólios reais. A Fig. 75 apresenta um exemplo de aerofólio que falha esse requisito, e seus vetores de variáveis são apresentados abaixo.

$$v_ex = [0,005;0;0,1;0;5;0]$$

```
v_in = [0,005;0;0,1;0,5;0]
```

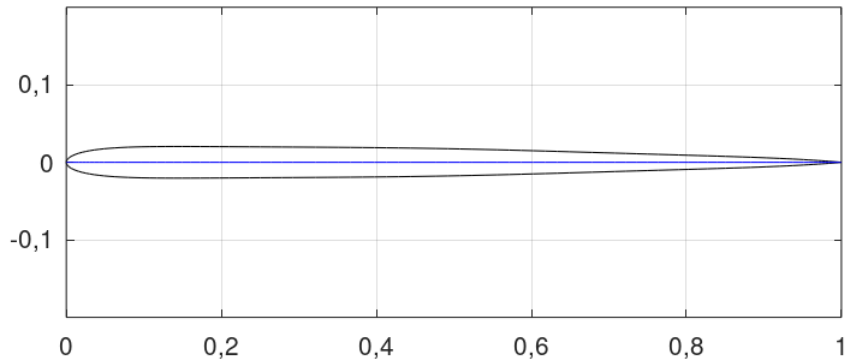


Figura 75 – Perfil CST excessivamente fino (fonte: autor)

Deve-se notar, no entanto, que certos destes requisitos podem ser alterados de acordo com as intenções do projetista. Por exemplo, aerofólios finos como o deste último exemplo são benéficos para aeronaves com regimes de voo transônico e supersônico, e, portanto, poderiam ser permitidos modificando-se o valor mínimo de espessura.

APÊNDICE M – CST reverso

A forma primária de uso do CST baseia-se na definição das variáveis de design, as quais, quando inseridas na formulação, geram uma dada geometria especificada na forma de coordenadas bidimensionais. Fazendo as devidas adaptações, é também possível realizar o caminho inverso: fornecer um conjunto de coordenadas e receber suas respectivas variáveis de *design* no CST, o que permite a manipulação da geometria desse aerofólio em aplicações como, justamente, algoritmos de otimização.

Existem duas formas de se realizar o cálculo inverso da geometria CST. Ambas serão expostas abaixo.

M.1 Sistema linear simples

A primeira variável obtida é a espessura do bordo de fuga, a qual é igual ao valor da ordenada exatamente no bordo de fuga. Com isso, restam mais $n + 1$ variáveis, que são os pesos A_r da função *shape*. A equação que representa a formulação do CST para aerofólios é repetida abaixo:

$$\zeta(\psi) = \sqrt{\psi} (1 - \psi) \sum_{r=0}^n \left[A_r \left[K_{r,n} \psi^r (1 - \psi)^{n-r} \right] \right] + \psi \Delta\xi \quad (2.38)$$

Da Eq. 2.38, $\zeta(\psi)$, ψ e $\Delta\xi$ são todos valores conhecidos. Assim, escolhendo $n + 1$ pontos linearmente espaçados ao longo do eixo x das coordenadas da superfície, forma-se um sistema linear de $n + 1$ equações e $n + 1$ incógnitas, facilmente solucionável. O processo é feito uma vez para cada uma de ambas as superfícies do perfil. Por exemplo, a Fig. 76 mostra uma geometria CST, configurada com o quarto grau do polinômio, gerada a partir de um aerofólio já existente.

Em nota, a distribuição dos pontos escolhidos para o sistema linear é fundamental para a boa correlação do CST com a geometria original. Se, por exemplo, os pontos escolhidos estiverem demasiadamente concentrados no bordo de ataque, o CST representará corretamente o bordo de ataque, mas possivelmente não será tão preciso no resto da geometria do aerofólio. Adicionalmente, geometrias mais complexas de aerofólios podem requerir graus maiores do polinômio de Bernstein, e não necessariamente todo aerofólio pode ser convertido para o CST, não importa o quão alto seja o grau do polinômio.

Cuidados também devem ser tomados quanto ao fato de o CST não gerar bordos de ataque cuja inclinação passa de 90° . Ou seja, caso esse requisito não seja cumprido, o CST não conseguirá acompanhar devidamente o bordo de ataque do aerofólio original. Neste

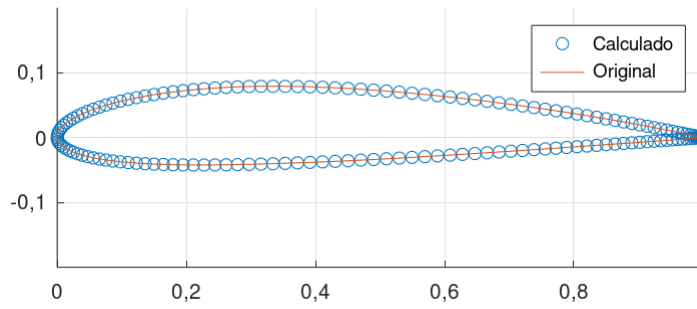


Figura 76 – Perfil NACA 2412 convertido para o CST

trabalho, esse problema foi abordado fazendo uma ligeira translação nas coordenadas do aerofólio quando necessário, de modo que o ponto com o menor valor da abscissa (caso haja algum valor menor que zero) sempre fique posicionado na origem (0,0).

M.2 Método dos mínimos quadrados

A situação em mãos trata-se de um problema sobredeterminado. O método descrito na seção anterior não é o mais eficiente devido ao fato de que os $n + 1$ pontos, escolhidos manualmente pelo usuário, nem sempre fornecem a melhor solução, levando a uma estratégia de tentativa e erro. Para situações como esta, o método dos mínimos quadrados é um dos mais aptos, o qual tem o propósito de obter a solução que melhor se correlaciona aos dados em mãos. Esta é a abordagem originalmente tomada em (KULFAN; BUSSOLETTI, 2006) para gerar geometrias CST a partir de aerofólios reais.

O método dos mínimos quadrados é baseado no sistema abaixo (LOPES, 1996):

$$\begin{cases} \left[\sum_{k=1}^m g_1(x_k)g_1(x_k) \right] \alpha_1 + \dots + \left[\sum_{k=1}^m g_n(x_k)g_1(x_k) \right] \alpha_n = \sum_{k=1}^m f(x_k)g_1(x_k) \\ \left[\sum_{k=1}^m g_1(x_k)g_2(x_k) \right] \alpha_1 + \dots + \left[\sum_{k=1}^m g_n(x_k)g_2(x_k) \right] \alpha_n = \sum_{k=1}^m f(x_k)g_2(x_k) \\ \vdots \\ \left[\sum_{k=1}^m g_1(x_k)g_n(x_k) \right] \alpha_1 + \dots + \left[\sum_{k=1}^m g_n(x_k)g_n(x_k) \right] \alpha_n = \sum_{k=1}^m f(x_k)g_n(x_k) \end{cases} \quad (\text{M.1})$$

Onde x_1, x_2, \dots, x_m e $f(x_1), f(x_2), \dots, f(x_m)$ são os pontos tabelados para os quais se deseja obter uma correlação, as funções $g_1(x), g_2(x), \dots, g_t(x_m)$ são as funções escolhidas pelo programador para correlacionar os dados, e $\alpha_1, \alpha_2, \dots, \alpha_t$ são as incógnitas. Passando para o domínio do problema da conversão de aerofólios para o CST, temos que os pontos x e $f(x)$ são as abscissas e ordenadas das coordenadas do aerofólio original, respectivamente, as funções $g(x)$ são funções retiradas da Eq. 2.38, e as incógnitas α são as variáveis de *design* no CST que busca-se obter.

Observa-se que, ao expandir o somatório na Eq. 2.38, obtém-se uma série de termos bastante similares ao que se deseja inserir nas equações M.1. Desse modo, como um exemplo, assumindo que o aerofólio será gerado com um grau 4 do polinômio de Bernstein, e utilizando $\psi = x$, as funções $g(x)$ para uso no sistema seriam as seguintes:

$$g_{r+1}(x) = \sqrt{x}(1-x) \left[K_{r,n} x^r (1-x)^{n-r} \right] \quad (\text{M.2})$$

$$g_6(x) = x \frac{1}{c} \quad (\text{M.3})$$

Enfatiza-se que, neste exemplo, $0 \leq r \leq 4$, portanto, a Eq. M.2 representa as funções $g_1(x)$ a $g_5(x)$ apenas.

Similarmente ao outro método, os valores de ΔZ_{te} para ambas as superfícies podem ser obtidos diretamente a partir das coordenadas do aerofólio. No entanto, como cada variável no CST tem efeito sobre a geometria completa, é necessário que o ΔZ_{te} seja considerado no sistema de equações para que os valores das demais incógnitas sejam calculadas corretamente. Sendo assim, temos que $\alpha_{r+1} = A_r$ para $0 \leq r \leq 4$ e $\alpha_6 = \Delta Z_{te}$ (nota: mesmo considerando ΔZ_{te} como uma incógnita, são utilizados os valores originais das coordenadas para futuros usos dos parâmetros CST, pois os valores calculados normalmente vêm com um pouco de erro. Sendo assim, o ΔZ_{te} como incógnita tem como único propósito garantir que o sistema forneça os valores corretos das outras incógnitas).

Com isso, estabelece-se um sistema linear de $n+2$ equações e $n+2$ incógnitas, que fornece as variáveis de *design* em formato CST de um dado aerofólio. Esse processo pode ser feito com graus diferentes do polinômio de Bernstein e, novamente, deve ser realizado uma vez para cada uma de ambas as superfícies do aerofólio. Os cuidados mencionados ao final da seção anterior também aplicam-se aqui.

APÊNDICE N – Interpolação de aerofólios

Nos algoritmos de asas, a geração e manipulação dos aerofólios é realizada apenas nas três estações principais (raiz, meio e ponta). No entanto, para o APAME, é necessário que haja mais seções transversais intermediárias entre essas estações para que a malha tenha uma resolução satisfatória. Essas seções intermediárias são obtidas por meio da interpolação linear das coordenadas dos aerofólios das estações diretamente adjacentes.

Admite-se dois aerofólios representados por suas respectivas coordenadas, as quais são armazenadas nas matrizes COO_1 e COO_2 . Cada uma dessas matrizes é dividida em duas - uma contendo apenas as coordenadas do extradorso e outra contendo apenas as coordenadas do intradorso. Assim, obtém-se as matrizes $COO_{1,ex}$, $COO_{1,in}$, $COO_{2,ex}$ e $COO_{2,in}$. As coordenadas do perfil interpolado são, então, obtidas com

$$\begin{cases} COO_{intp,ex} = COO_{1,ex}(1 - op) + COO_{2,ex} * op \\ COO_{intp,in} = COO_{1,in}(1 - op) + COO_{2,in} * op \end{cases} \quad (N.1)$$

Onde op ($0 \leq op \leq 1$) é o valor que determina a tendência da interpolação. Quanto mais próximo de zero, mais parecido com o perfil 1 será o perfil interpolado, e quanto mais próximo de 1, mais parecido com o perfil 2 será o perfil interpolado. Valores de $op = 0$ e $op = 1$ retornam aerofólios iguais ao perfil 1 e o perfil 2 respectivamente e, portanto, não são de interesse ao processo de interpolação.

É importante enfatizar que as Equações N.1 são modeladas tendo em mente que o formato das variáveis no Octave são matrizes, com as quais pode-se realizar diversas operações. Para uso em outras linguagens, deve-se carregar módulos/pacotes que permitam tais operações (como o *numpy* para o Python), ou fazer as operações com um elemento de cada vez.

APÊNDICE O – Testes de parâmetros restantes

O.1 Algoritmo CST específico

O algoritmo CST específico opera de modo a otimizar uma geometria única de um dado aerofólio. Para estes testes, o algoritmo foi configurado para otimizar um aerofólio NACA 2412. Os parâmetros geométricos para descrever sua geometria em termos da parametrização CST foram obtidos por meio do CST reverso de mínimos quadrados, e o grau do polinômio utilizado foi 6. Para o caso deste algoritmo, aumentar o grau do polinômio da função *shape* é uma forma de permitir que o algoritmo modele a geometria em mais detalhes, de modo que o resultado final seja ainda mais refinado em termos da função objetivo. Nessa mesma linha de pensamento, apesar do aumento do número de variáveis de *design*, o desempenho do algoritmo não é negativamente afetado devido ao fato de que o espaço de busca se limita a aerofólios com geometria similar ao original.

A configuração padrão deste algoritmo para os testes está na Tab. 26, e os resultados em si estão nas Tabelas 27, 28, 29 e 30.

População	Mutação	Iterações no XFOIL	Elitismo	Iterações no algoritmo
50	5%	10	Ativado	5

Tabela 26 – Configuração padrão do algoritmo CST específico para os testes de parâmetros

População	L/D	Variação	Tempo
50	77,21	+88,04%	1 min e 53.74 s
150	76,67	+86,72%	5 min e 33.45 s
300	78,08	+90,16%	10 min e 56.50 s

Tabela 27 – Resultados dos testes de tamanho populacional no algoritmo CST específico

Mutação	L/D	Variação	Tempo
5%	68,92	+67,85%	1 min e 53.63 s
10%	74,01	+80,24%	1 min e 54.46 s
20%	65,94	+60,59%	1 min e 50.89 s

Tabela 28 – Resultados dos testes de chance de mutação no algoritmo CST específico

Iterações no XFOIL	L/D	Variação	Tempo
10	78,69	+91,64%	3 min e 19.47 s
50	70,15	+70,84%	3 min e 19.73 s
100	71,59	+74,35%	3 min e 10.00 s

Tabela 29 – Resultados dos testes de número de iterações no XFOIL no algoritmo CST específico

Ativação do mecanismo	Variação	L/D	Tempo
Desativado	74,97	+82,58%	7 min e 26.44 s
Ativado	74,62	+81,73%	7 min e 13.70 s

Tabela 30 – Resultados dos testes do mecanismo de elitismo no algoritmo CST específico

Os parâmetros geométricos dos aerofólios obtidos nestes testes estão disponíveis no Apêndice P.2.

O.2 Algoritmo NACA 4 dígitos 3D

Para estes testes e todos os demais referentes aos algoritmos de asas, o ângulo de ataque nas simulações é nulo. Adicionalmente, como a simulação é invíscida, o número de Reynolds é irrelevante, e como estamos interessados apenas em coeficientes adimensionais, a velocidade do ar e suas demais propriedades também são irrelevantes. Similarmente aos algoritmos CST de aerofólios, dados sobre as geometrias das asas não serão fornecidos nestas tabelas e sim em um apêndice distinto devido à grande quantidade de parâmetros geométricos que definem cada asa.

A configuração padrão deste algoritmo para os testes está na Tab. 31, e os resultados em si estão nas Tabelas 32, 33 e 34.

População	Mutação	Elitismo	Iterações no algoritmo
300	5%	Ativado	5

Tabela 31 – Configuração padrão do algoritmo NACA 4 dígitos 3D para os testes de parâmetros (versão Octave)

População	L/D	Tempo
300	701,00	16 min e 7.58 s
500	1396,00	29 min e 1.95 s
700	1424,00	31 min e 36.35 s

Tabela 32 – Resultados dos testes de tamanho populacional no algoritmo NACA 4 dígitos 3D

Mutação	L/D	Tempo
5%	1023,00	9 min e 8.50 s
10%	663,00	10 min e 41.85 s
20%	753,00	9 min e 42.51 s

Tabela 33 – Resultados dos testes de chance de mutação no algoritmo NACA 4 dígitos 3D

Ativação do mecanismo	L/D	Tempo
Desativado	805,00	18 min e 46.57 s
Ativado	1535,00	20 min e 43.33 s

Tabela 34 – Resultados dos testes do mecanismo de elitismo no algoritmo NACA 4 dígitos 3D

As asas obtidas nestes testes são apresentadas no apêndice [P.4](#).

O.3 Algoritmo CST 3D

A configuração padrão deste algoritmo para os testes está na Tab. [35](#), e os resultados em si estão nas Tabelas [36](#), [37](#) e [38](#).

População	Mutação	Elitismo	Iterações no algoritmo
700	5%	Ativado	5

Tabela 35 – Configuração padrão do algoritmo CST 3D para os testes de parâmetros (versão Octave)

População	L/D	Tempo
700	912,00	33 min e 43.92 s
1200	512,50	65 min e 32.97 s
1700	1391,00	80 min e 56.23 s

Tabela 36 – Resultados dos testes de tamanho populacional no algoritmo CST 3D

Mutação	L/D	Tempo
5%	990,00	39 min e 40.39 s
10%	1007,00	41 min e 4.61 s
20%	1376,00	38 min e 29.32 s

Tabela 37 – Resultados dos testes de chance de mutação no algoritmo CST 3D

Ativação do mecanismo	L/D	Tempo
Desativado	728,00	81 min e 15.86 s
Ativado	1091,00	71 min e 0.18 s

Tabela 38 – Resultados dos testes do mecanismo de elitismo no algoritmo CST 3D

Os parâmetros geométricos das asas obtidas nestes testes são apresentados no apêndice P.5.

O.4 Algoritmo CST 3D específico

Este algoritmo é a versão análoga 3D do algoritmo CST específico, ou seja, seu propósito é fazer a otimização de uma asa específica alterando sua geometria dentro de extensões próximas aos seus valores iniciais. Para estes testes, a geometria inicial escolhida foi uma asa genérica de planador, com as seguintes dimensões da planta:

- $type = 1$, portanto, é uma asa trapezoidal dupla;
- $b = 14m$;
- $b1 = 12m$;
- $c_r = 2m$;
- $c_m = 1,5m$;
- $c_t = 0,5m$;
- $sweep1 = 5^\circ$;
- $sweep2 = 45^\circ$;
- $tw_t = 0^\circ$;

Os aerofólios das estações da raiz e do meio são perfis Wortmann FX 61-163, e o aerofólio da estação da ponta é um perfil Wortmann FX 60-126. As coordenadas de ambos

foram obtidas em (SELIG, 2022). A escolha desses perfis foi baseada em (COATES, 1980), que mostra que ambos esses perfis viram bastante uso em planadores como perfis da raiz e da ponta respectivamente.

A configuração padrão deste algoritmo para os testes está na Tab. 39, e os resultados em si estão nas Tabelas 40, 41 e 42. A asa original tem um L/D igual a 91,54.

População	Mutação	Elitismo	Iterações no algoritmo
150	5%	Ativado	5

Tabela 39 – Configuração padrão do algoritmo CST 3D específico para os testes de parâmetros (versão Octave)

População	L/D	Variação	Tempo
150	2566,00	+2703,14%	7 min e 32.82 s
300	3051,00	+3232,96%	18 min e 4.09 s
450	2780,00	+2936,92%	29 min e 20.56 s

Tabela 40 – Resultados dos testes de tamanho populacional no algoritmo CST específico 3D

Mutação	L/D	Variação	Tempo
5%	2032,00	2119,79%	10 min e 59.56 s
10%	2478,00	2607,01%	11 min e 35.58 s
20%	3055,00	3237,33%	8 min e 21.07 s

Tabela 41 – Resultados dos testes de chance de mutação no algoritmo CST 3D específico

Ativação do mecanismo	L/D	Variação	Tempo
Desativado	3201,00	3396,83%	25 min e 12.09 s
Ativado	3304,00	3509,35%	26 min e 6.81 s

Tabela 42 – Resultados dos testes do mecanismo de elitismo no algoritmo CST 3D específico

É relevante ter atenção ao fato de que o APAME é um solucionador invíscido, portanto, suas estimativas de L/D não condizem com a realidade na grande maioria dos casos. Precisamente por isso que, ao otimizar para a maximização do L/D , o algoritmo é capaz de encontrar valores exorbitantes, da ordem de grandeza de milhares. Dito isso, apesar de não ser realista, o problema de otimização imposto a estes testes de parâmetros é interessante do ponto de vista do processo de otimização em si, e abre espaço também

para a discussão da aplicabilidade de solucionadores a diferentes problemas tendo em vista suas virtudes e suas limitações.

Os parâmetros geométricos das asas obtidas nestes testes são apresentados no apêndice [P.6](#).

APÊNDICE P – Aerofólios e asas resultantes dos testes de parâmetros

A seguir estão disponíveis os dados das geometrias de todos os aerofólios e asas encontrados pelos algoritmos durante os testes de parâmetros de configuração. As numerações dos aerofólios e asas referem-se à ordem que os mesmos são obtidos nos testes. Desse modo, por exemplo, para o testes de tamanho populacional do algoritmo NACA 4 dígitos, os perfis 1, 2 e 3 são resultantes, respectivamente, dos testes realizados com tamanhos populacionais 100, 300 e 500. Para todo perfil CST apresentado, serão fornecidos seus vetores de extradorso e intradorso. Para as asas, serão fornecidos os parâmetros da planta e dos aerofólios, sempre seguindo as demais regras apresentadas.

P.1 Algoritmo NACA 4 dígitos

Aerofólio	NACA
1	9610
2	9610
3	9610

Tabela 43 – Aerofólios obtidos nos testes de tamanho populacional no algoritmo NACA 4 dígitos (fonte: autor)

Aerofólio	NACA
1	9611
2	9610
3	9611

Tabela 44 – Aerofólios obtidos nos testes de chance de mutação no algoritmo NACA 4 dígitos (fonte: autor)

Aerofólio	NACA
1	9610
2	9610
3	9611

Tabela 45 – Aerofólios obtidos nos testes de números de iterações no XFOIL no algoritmo NACA 4 dígitos (fonte: autor)

Aerofólio	NACA
1	9620
2	9610

Tabela 46 – Aerofólios obtidos nos testes de mecanismo de elitismo no algoritmo NACA 4 dígitos (fonte: autor)

P.2 Algoritmo CST

Vetor	R_{Le}	A_0	A_1	A_2	β	ΔZ_{TE}
$v_{ex,1}$	0,0150	0,2250	0,2250	0,3000	20,0000	0,0000
$v_{in,1}$	0,0100	0,0500	0,0750	0,0500	2,0000	0,0000
$v_{ex,2}$	0,0200	0,2250	0,2750	0,2750	19,0000	0,0000
$v_{in,2}$	0,0100	0,0500	0,0500	-0,0000	1,0000	0,0000
$v_{ex,3}$	0,0250	0,2250	0,2500	0,2750	20,0000	0,0000
$v_{in,3}$	0,0100	0,1000	0,0250	-0,0000	0,0000	0,0000

Tabela 47 – Parâmetros geométricos dos aerofólios obtidos nos testes de tamanho populacional no algoritmo CST (fonte: autor)

Vetor	R_{Le}	A_0	A_1	A_2	β	ΔZ_{TE}
$v_{ex,1}$	0,0250	0,2000	0,2000	0,2750	19,0000	0,0000
$v_{in,1}$	0,0100	0,0250	0,1000	-0,0000	3,0000	0,0000
$v_{ex,2}$	0,0150	0,2500	0,2250	0,3000	19,0000	0,0000
$v_{in,2}$	0,0100	0,0750	0,0500	-0,0000	1,0000	0,0000
$v_{ex,3}$	0,0150	0,1750	0,3000	0,2750	20,0000	0,0000
$v_{in,3}$	0,0100	-0,0000	0,0750	0,0500	1,0000	0,0000

Tabela 48 – Parâmetros geométricos dos aerofólios obtidos nos testes de chance de mutação no algoritmo CST (fonte: autor)

Vetor	R_{Le}	A_0	A_1	A_2	β	ΔZ_{TE}
$v_{ex,1}$	0,0100	0,2000	0,2250	0,2500	20,0000	0,0000
$v_{in,1}$	0,0100	0,0500	0,0750	0,1000	0,0000	0,0000
$v_{ex,2}$	0,0150	0,1000	0,2500	0,3000	19,0000	0,0000
$v_{in,2}$	0,0100	0,0250	0,0750	0,0250	9,0000	0,0000
$v_{ex,3}$	0,0350	0,1750	0,2250	0,3000	20,0000	0,0000
$v_{in,3}$	0,0100	0,0250	0,0750	0,0750	0,0000	0,0000

Tabela 49 – Parâmetros geométricos dos aerofólios obtidos nos testes de números de iterações no XFOIL no algoritmo CST (fonte: autor)

Vetor	R_{Le}	A_0	A_1	A_2	β	ΔZ_{TE}
$v_{ex,1}$	0,0200	0,1500	0,3000	0,2250	20,0000	0,0000
$v_{in,1}$	0,0100	0,0500	0,0500	0,0750	1,0000	0,0000
$v_{ex,2}$	0,0200	0,1750	0,3000	0,3000	20,0000	0,0000
$v_{in,2}$	0,0100	-0,0000	0,1000	0,0250	0,0000	0,0000

Tabela 50 – Parâmetros geométricos dos aerofólios obtidos nos testes de mecanismo de elitismo no algoritmo CST (fonte: autor)

P.3 Algoritmo CST específico

Vetor	R_{Le}	A_0	A_1	A_2	A_3	A_4	β	ΔZ_{TE}
$v_{ex,1}$	0,0158	0,1769	0,2006	0,1560	0,2767	0,2295	17,1325	-0,0000
$v_{in,1}$	0,0065	0,0812	0,0898	0,0707	0,0582	0,0786	0,6803	-0,0000
$v_{ex,2}$	0,0180	0,1709	0,2224	0,1610	0,2057	0,2346	14,4923	-0,0000
$v_{in,2}$	0,0058	0,0736	0,0575	0,0586	0,0385	0,0374	2,7447	-0,0000
$v_{ex,3}$	0,0248	0,1825	0,2529	0,1464	0,2827	0,2325	15,9124	-0,0000
$v_{in,3}$	0,0058	0,1168	0,0688	0,0721	0,1034	0,0343	10,5977	-0,0000

Tabela 51 – Parâmetros geométricos dos aerofólios obtidos nos testes de tamanho populacional no algoritmo CST específico (fonte: autor)

Vetor	R_{Le}	A_0	A_1	A_2	A_3	A_4	β	ΔZ_{TE}
$v_{ex,1}$	0,0157	0,1802	0,2142	0,2072	0,2788	0,2267	16,7254	-0,0000
$v_{in,1}$	0,0050	0,0847	0,0866	0,1235	0,0676	0,0870	1,1235	-0,0000
$v_{ex,2}$	0,0282	0,1690	0,2514	0,2065	0,2825	0,2230	16,9135	-0,0000
$v_{in,2}$	0,0050	0,0708	0,0469	0,1244	0,0697	0,0366	0,8332	-0,0000
$v_{ex,3}$	0,0229	0,1387	0,2127	0,1729	0,1971	0,2265	16,3341	-0,0000
$v_{in,3}$	0,0050	0,1135	0,0446	0,0628	0,0374	0,0930	4,3403	-0,0000

Tabela 52 – Parâmetros geométricos dos aerofólios obtidos nos testes de chances de mutação no algoritmo CST específico (fonte: autor)

Vetor	R_{Le}	A_0	A_1	A_2	A_3	A_4	β	ΔZ_{TE}
$v_{ex,1}$	0,0143	0,1761	0,2462	0,1468	0,2511	0,2165	17,0869	-0,0000
$v_{in,1}$	0,0051	0,0799	0,0479	0,0992	0,0327	0,1039	-0,1868	-0,0000
$v_{ex,2}$	0,0209	0,1606	0,2498	0,1758	0,2242	0,2290	16,5261	-0,0000
$v_{in,2}$	0,0051	0,1082	0,0740	0,0665	0,0387	0,1085	1,8743	-0,0000
$v_{ex,3}$	0,0220	0,2041	0,2404	0,1486	0,2740	0,2273	17,0499	-0,0000
$v_{in,3}$	0,0082	0,0869	0,0662	0,1002	0,0587	0,0726	2,2680	-0,0000

Tabela 53 – Parâmetros geométricos dos aerofólios obtidos nos testes de números de iterações no XFOIL no algoritmo CST específico (fonte: autor)

Vetor	R_{Le}	A_0	A_1	A_2	A_3	A_4	β	ΔZ_{TE}
$v_{ex,1}$	0,0115	0,1529	0,2083	0,1643	0,2600	0,2159	16,4611	-0,0000
$v_{in,1}$	0,0072	0,1002	0,0527	0,0661	0,0672	0,0694	9,3141	-0,0000
$v_{ex,2}$	0,0197	0,2057	0,2102	0,1901	0,2341	0,2186	15,1876	-0,0000
$v_{in,2}$	0,0050	0,0714	0,0549	0,1065	0,0596	0,0434	8,9338	-0,0000

Tabela 54 – Parâmetros geométricos dos aerofólios obtidos nos testes de números de iterações no XFOIL no algoritmo CST específico (fonte: autor)

P.4 Algoritmo NACA 4 dígitos 3D

Asa	Tipo	$b[m]$	$b1[m]$	$c_r[m]$	$c_m[m]$	$c_t[m]$
1	1	13,5	9,5	1,5	1,5	0,6
2	1	13,5	8	1,2	1,2	1,2
3	1	12,5	9	1,7	1,7	0,6

Tabela 55 – Geometrias das plantas das asas obtidas nos testes de tamanho populacional no algoritmo NACA 4 dígitos 3D - parte 1 (fonte: autor)

Asa	Tipo	$sweep[^\circ]$	$sweep1[^\circ]$	$sweep2[^\circ]$	$tw_t[^\circ]$
1	1	-	2	6	-4
2	1	-	2	6	-0.5
3	1	-	1	7	-1

Tabela 56 – Geometrias das plantas das asas obtidas nos testes de tamanho populacional no algoritmo NACA 4 dígitos 3D - parte 2 (fonte: autor)

Asa	Tipo	Perfil da raiz	Perfil do meio	Perfil da ponta
1	1	NACA 1418	NACA 4318	NACA 2417
2	1	NACA 0012	NACA 3419	NACA 1310
3	1	NACA 3112	NACA 2414	NACA 2318

Tabela 57 – Aerofólios das asas obtidas nos testes de tamanho populacional no algoritmo NACA 4 dígitos 3D (fonte: autor)

Asa	Tipo	$b[m]$	$b1[m]$	$c_r[m]$	$c_m[m]$	$c_t[m]$
1	1	10	9	1,1	0,9	0,8
2	1	10	8	1,9	1,9	0,5
3	1	11,5	10,5	1,8	1,6	1,6

Tabela 58 – Geometrias das plantas das asas obtidas nos testes de chance de mutação no algoritmo NACA 4 dígitos 3D - parte 1 (fonte: autor)

Asa	Tipo	<i>sweep</i> [°]	<i>sweep1</i> [°]	<i>sweep2</i> [°]	<i>tw_t</i> [°]
1	1	-	1	14	0
2	1	-	8	15	0
3	1	-	0	8	-1

Tabela 59 – Geometrias das plantas das asas obtidas nos testes de chance de mutação no algoritmo NACA 4 dígitos 3D - parte 2 (fonte: autor)

Asa	Tipo	Perfil da raiz	Perfil do meio	Perfil da ponta
1	1	NACA 1212	NACA 2116	NACA 4310
2	1	NACA 3210	NACA 0014	NACA 2210
3	1	NACA 0011	NACA 4118	NACA 0011

Tabela 60 – Aerofólios das asas obtidas nos testes de chance de mutação no algoritmo NACA 4 dígitos 3D (fonte: autor)

Asa	Tipo	<i>b</i> [m]	<i>b1</i> [m]	<i>c_r</i> [m]	<i>c_m</i> [m]	<i>c_t</i> [m]
1	1	13	9	1,7	1,5	1
2	1	13,5	9,5	1,7	1,3	1,3

Tabela 61 – Geometrias das plantas das asas obtidas nos testes de mecanismo de elitismo no algoritmo NACA 4 dígitos 3D - parte 1 (fonte: autor)

Asa	Tipo	<i>sweep</i> [°]	<i>sweep1</i> [°]	<i>sweep2</i> [°]	<i>tw_t</i> [°]
1	1	-	0	7	-5
2	1	-	2	6	-2

Tabela 62 – Geometrias das plantas das asas obtidas nos testes de mecanismo de elitismo no algoritmo NACA 4 dígitos 3D - parte 2 (fonte: autor)

Asa	Tipo	Perfil da raiz	Perfil do meio	Perfil da ponta
1	1	NACA 1411	NACA 4417	NACA 1413
2	1	NACA 4213	NACA 1420	NACA 3412

Tabela 63 – Aerofólios das asas obtidas nos testes de mecanismo de elitismo no algoritmo NACA 4 dígitos 3D - parte 2 (fonte: autor)

P.5 Algoritmo CST 3D

Asa	Tipo	$b[m]$	$b1[m]$	$c_r[m]$	$c_m[m]$	$c_t[m]$
1	1	13,5	11	1,6	1,3	0,6
2	1	12	9,5	1,7	1,7	1
3	1	13	10,5	1,5	1,5	0,8

Tabela 64 – Geometrias das plantas das asas obtidas nos testes de tamanho populacional no algoritmo CST 3D - parte 1 (fonte: autor)

Asa	Tipo	$sweep[^\circ]$	$sweep1[^\circ]$	$sweep2[^\circ]$	$tw_t[^\circ]$
1	1	-	0	6	-1,25
2	1	-	2	7	-0,75
3	1	-	0	5	-0,75

Tabela 65 – Geometrias das plantas das asas obtidas nos testes de tamanho populacional no algoritmo CST 3D - parte 2 (fonte: autor)

Vetor	R_{Le}	A_0	A_1	A_2	β	ΔZ_{TE}
$v_{ex,1}$	0,0400	0,2000	0,1000	0,0500	16,0000	0,0000
$v_{in,1}$	0,0400	0,2000	0,1000	0,0500	16,0000	0,0000
$v_{ex,2}$	0,0300	0,1500	0,2500	0,0500	13,0000	0,0000
$v_{in,2}$	0,0300	0,1500	0,2500	0,0500	13,0000	0,0000
$v_{ex,3}$	0,0200	0,2000	0,1500	0,1000	14,0000	0,0000
$v_{in,3}$	0,0300	0,1000	-0,0000	0,1000	7,0000	0,0000

Tabela 66 – Parâmetros geométricos dos aerofólios da raiz obtidos nos testes de tamanho populacional no algoritmo CST 3D (fonte: autor)

Vetor	R_{Le}	A_0	A_1	A_2	β	ΔZ_{TE}
$v_{ex,1}$	0,0300	0,3000	0,1500	0,0000	18,0000	0,0000
$v_{in,1}$	0,0100	0,1000	-0,0000	-0,0000	4,0000	0,0000
$v_{ex,2}$	0,0500	0,2000	0,1500	0,1000	17,0000	0,0000
$v_{in,2}$	0,0200	0,1000	-0,0000	-0,0000	6,0000	0,0000
$v_{ex,3}$	0,0200	0,3000	0,1500	0,1500	18,0000	0,0000
$v_{in,3}$	0,0300	0,1000	0,0500	0,0500	7,0000	0,0000

Tabela 67 – Parâmetros geométricos dos aerofólios do meio obtidos nos testes de tamanho populacional no algoritmo CST 3D (fonte: autor)

Vetor	R_{Le}	A_0	A_1	A_2	β	ΔZ_{TE}
$v_{ex,1}$	0,0100	0,0500	0,2000	0,1000	12,0000	0,0000
$v_{in,1}$	0,0100	0,1000	0,1000	0,0500	12,0000	0,0000
$v_{ex,2}$	0,0200	0,0000	0,2000	0,1500	20,0000	0,0000
$v_{in,2}$	0,0200	0,1000	0,1000	0,0500	17,0000	0,0000
$v_{ex,3}$	0,0100	0,0000	0,2500	0,1500	17,0000	0,0000
$v_{in,3}$	0,0100	0,0000	0,2500	0,1500	17,0000	0,0000

Tabela 68 – Parâmetros geométricos dos aerofólios da ponta obtidos nos testes de tamanho populacional no algoritmo CST 3D (fonte: autor)

Asa	Tipo	$b[m]$	$b1[m]$	$c_r[m]$	$c_m[m]$	$c_t[m]$
1	1	12,5	8	1,4	1,4	0,5
2	1	11	9,5	1	0,8	0,6
3	1	12	9	1,5	1,4	1,4

Tabela 69 – Geometrias das plantas das asas obtidas nos testes de chance de mutação no algoritmo CST 3D - parte 1 (fonte: autor)

Asa	Tipo	$sweep[^\circ]$	$sweep1[^\circ]$	$sweep2[^\circ]$	$tw_t[^\circ]$
1	1	-	0	7	-1,5
2	1	-	0	9	-0,5
3	1	-	1	5	-2,25

Tabela 70 – Geometrias das plantas das asas obtidas nos testes de chance de mutação no algoritmo CST 3D - parte 2 (fonte: autor)

Vetor	R_{Le}	A_0	A_1	A_2	β	ΔZ_{TE}
$v_{ex,1}$	0,0500	0,0500	0,1500	0,0000	17,0000	0,0000
$v_{in,1}$	0,0500	0,0500	0,1500	0,0000	17,0000	0,0000
$v_{ex,2}$	0,0200	0,3000	0,0000	0,0500	20,0000	0,0000
$v_{in,2}$	0,0100	0,1000	-0,0000	0,1000	5,0000	0,0000
$v_{ex,3}$	0,0100	0,2500	0,3000	0,0000	17,0000	0,0000
$v_{in,3}$	0,0200	-0,0000	0,1000	0,0500	4,0000	0,0000

Tabela 71 – Parâmetros geométricos dos aerofólios da raiz obtidos nos testes de chance de mutação no algoritmo CST 3D (fonte: autor)

Vetor	R_{Le}	A_0	A_1	A_2	β	ΔZ_{TE}
$v_{ex,1}$	0,0100	0,3000	0,3000	0,1000	19,0000	0,0000
$v_{in,1}$	0,0300	0,0500	0,0500	0,0500	9,0000	0,0000
$v_{ex,2}$	0,0100	0,2500	0,3000	0,1500	18,0000	0,0000
$v_{in,2}$	0,0100	0,2500	0,3000	0,1500	18,0000	0,0000
$v_{ex,3}$	0,0100	0,3000	0,2000	0,2500	18,0000	0,0000
$v_{in,3}$	0,0200	0,1000	0,1000	0,1000	4,0000	0,0000

Tabela 72 – Parâmetros geométricos dos aerofólios do meio obtidos nos testes de chance de mutação no algoritmo CST 3D (fonte: autor)

Vetor	R_{Le}	A_0	A_1	A_2	β	ΔZ_{TE}
$v_{ex,1}$	0,0100	0,3000	0,1500	0,2000	8,0000	0,0000
$v_{in,1}$	0,0200	0,0500	0,1000	0,0500	17,0000	0,0000
$v_{ex,2}$	0,0200	0,0000	0,3000	0,2500	13,0000	0,0000
$v_{in,2}$	0,0300	0,0500	0,0500	0,1000	11,0000	0,0000
$v_{ex,3}$	0,0200	0,0000	0,2000	0,1000	15,0000	0,0000
$v_{in,3}$	0,0100	0,1000	0,1000	-0,0000	10,0000	0,0000

Tabela 73 – Parâmetros geométricos dos aerofólios da ponta obtidos nos testes de chance de mutação no algoritmo CST 3D (fonte: autor)

Asa	Tipo	$b[m]$	$b1[m]$	$c_r[m]$	$c_m[m]$	$c_t[m]$
1	1	12	9,5	1,8	1,2	1,2
2	1	13	12	1,9	1,7	1,7

Tabela 74 – Geometrias das plantas das asas obtidas nos testes de mecanismo de elitismo no algoritmo CST 3D - parte 1 (fonte: autor)

Asa	Tipo	$sweep[^\circ]$	$sweep1[^\circ]$	$sweep2[^\circ]$	$tw_t[^\circ]$
1	1	-	2	6	-1,75
2	1	-	0	8	-1,75

Tabela 75 – Geometrias das plantas das asas obtidas nos testes de mecanismo e elitismo no algoritmo CST 3D - parte 2 (fonte: autor)

Vetor	R_{Le}	A_0	A_1	A_2	β	ΔZ_{TE}
$v_{ex,1}$	0,0200	0,3000	0,2000	0,1000	10,0000	0,0000
$v_{in,1}$	0,0300	0,0500	0,0500	0,1000	10,0000	0,0000
$v_{ex,2}$	0,0400	0,2000	0,2500	0,0000	15,0000	0,0000
$v_{in,2}$	0,0300	0,0500	0,1000	-0,0000	6,0000	0,0000

Tabela 76 – Parâmetros geométricos dos aerofólios da raiz obtidos nos testes de mecanismo de elitismo no algoritmo CST 3D (fonte: autor)

Vetor	R_{Le}	A_0	A_1	A_2	β	ΔZ_{TE}
$v_{ex,1}$	0,0400	0,3000	0,1500	0,2000	16,0000	0,0000
$v_{in,1}$	0,0300	-0,0000	0,1000	0,1000	8,0000	0,0000
$v_{ex,2}$	0,0100	0,2500	0,1500	0,1000	16,0000	0,0000
$v_{in,2}$	0,0300	0,1000	-0,0000	0,1000	4,0000	0,0000

Tabela 77 – Parâmetros geométricos dos aerofólios do meio obtidos nos testes de mecanismo de elitismo no algoritmo CST 3D (fonte: autor)

Vetor	R_{Le}	A_0	A_1	A_2	β	ΔZ_{TE}
$v_{ex,1}$	0,0200	0,0000	0,3000	0,2000	6,0000	0,0000
$v_{in,1}$	0,0200	0,1000	0,0500	0,0500	14,0000	0,0000
$v_{ex,2}$	0,0100	0,2500	0,0000	0,1500	16,0000	0,0000
$v_{in,2}$	0,0100	-0,0000	0,1000	0,0500	7,0000	0,0000

Tabela 78 – Parâmetros geométricos dos aerofólios da ponta obtidos nos testes de mecanismo de elitismo no algoritmo CST 3D (fonte: autor)

P.6 Algoritmo CST 3D específico

Asa	Tipo	$b[m]$	$b1[m]$	$c_r[m]$	$c_m[m]$	$c_t[m]$
1	1	13,2844	12,2844	1,6640	1,3386	0,5112
2	1	13,4071	12,4071	1,6299	1,6299	0,2000
3	1	13,1212	12,1212	2,2926	1,9783	0,3269

Tabela 79 – Geometrias das plantas das asas obtidas nos testes de tamanho populacional no algoritmo CST 3D específico - parte 1 (fonte: autor)

Asa	Tipo	$sweep[^\circ]$	$sweep1[^\circ]$	$sweep2[^\circ]$	$tw_t[^\circ]$
1	1	-	6,4500	42,6426	-1,9425
2	1	-	0,8942	44,0730	0
3	1	-	8,9912	40,1778	0

Tabela 80 – Geometrias das plantas das asas obtidas nos testes de tamanho populacional no algoritmo CST 3D específico - parte 2 (fonte: autor)

Vetor	R_{Le}	A_0	A_1	A_2	A_3	A_4	β	ΔZ_{TE}
$v_{ex,1}$	0,0208	0,4054	0,2748	0,4332	0,2654	0,2444	15,4491	0,0000
$v_{in,1}$	0,0090	0,2670	0,0182	0,3544	-0,0027	-0,0029	-8,7872	-0,0000
$v_{ex,2}$	0,0094	0,2606	0,1592	0,4153	0,1576	0,1636	20,1706	0,0000
$v_{in,2}$	0,0050	0,1025	0,0080	0,3071	0,1353	0,0114	-10,4680	-0,0000
$v_{ex,3}$	0,0158	0,3787	0,2618	0,3386	0,1426	0,2212	13,7894	0,0000
$v_{in,3}$	0,0123	0,0974	0,0677	0,2888	0,0870	0,0806	-6,1507	-0,0000

Tabela 81 – Parâmetros geométricos dos aerofólios da raiz obtidos nos testes de tamanho populacional no algoritmo CST 3D específico (fonte: autor)

Vetor	R_{Le}	A_0	A_1	A_2	A_3	A_4	β	ΔZ_{TE}
$v_{ex,1}$	0,0132	0,2859	0,1762	0,3539	0,1051	0,2157	18,9790	0,0000
$v_{in,1}$	0,0050	0,2161	0,0905	0,3386	0,1641	0,1281	-13,0198	-0,0000
$v_{ex,2}$	0,0138	0,3135	0,2543	0,3629	0,2652	0,1095	13,0325	0,0000
$v_{in,2}$	0,0050	0,2397	0,0509	0,3526	0,0866	0,0072	-6,9836	-0,0000
$v_{ex,3}$	0,0167	0,4168	0,1079	0,3208	0,1817	0,2246	15,4233	0,0000
$v_{in,3}$	0,0097	0,1010	-0,0148	0,3642	-0,0042	0,0716	-9,4366	-0,0000

Tabela 82 – Parâmetros geométricos dos aerofólios do meio obtidos nos testes de tamanho populacional no algoritmo CST 3D específico (fonte: autor)

Vetor	R_{Le}	A_0	A_1	A_2	A_3	A_4	β	ΔZ_{TE}
$v_{ex,1}$	0,0227	0,2103	0,2765	0,1999	0,3239	0,2407	9,0388	0,0000
$v_{in,1}$	0,0084	0,0609	0,0203	0,1991	-0,1582	-0,0017	-6,2212	-0,0000
$v_{ex,2}$	0,0189	0,2169	0,1463	0,2579	0,3008	0,2311	7,3091	0,0000
$v_{in,2}$	0,0157	0,1099	0,0623	0,1032	-0,0320	-0,0364	-4,7194	-0,0000
$v_{ex,3}$	0,0135	0,1695	0,2761	0,2769	0,2554	0,2496	10,9011	0,0000
$v_{in,3}$	0,0108	0,2127	0,0745	0,1813	-0,0357	-0,1222	-5,5570	-0,0000

Tabela 83 – Parâmetros geométricos dos aerofólios da ponta obtidos nos testes de tamanho populacional no algoritmo CST 3D específico (fonte: autor)

Asa	Tipo	$b[m]$	$b1[m]$	$c_r[m]$	$c_m[m]$	$c_t[m]$
1	1	13,0631	12,0631	1,5737	1,5737	0,5225
2	1	14,1908	12,2462	2,0649	1,7715	0,2000
3	1	13,6804	12,5351	2,3488	1,9866	0,2000

Tabela 84 – Geometrias das plantas das asas obtidas nos testes de chance de mutação no algoritmo CST 3D específico - parte 1 (fonte: autor)

Asa	Tipo	sweep[°]	sweep1[°]	sweep2[°]	tw_t[°]
1	1	-	9,1691	48,8973	-4,1459
2	1	-	3,0434	44,4229	-4,8999
3	1	-	6,6052	40,9557	-1,2129

Tabela 85 – Geometrias das plantas das asas obtidas nos testes de chance de mutação no algoritmo CST 3D específico - parte 2 (fonte: autor)

Vetor	R_{Le}	A_0	A_1	A_2	A_3	A_4	β	ΔZ_{TE}
$v_{ex,1}$	0,0264	0,3010	0,1902	0,4569	0,2337	0,1291	17,9752	0,0000
$v_{in,1}$	0,0050	0,1183	-0,0025	0,2821	0,0491	0,1147	-10,6245	-0,0000
$v_{ex,2}$	0,0220	0,3929	0,1831	0,4154	0,2263	0,2078	20,0618	0,0000
$v_{in,2}$	0,0050	0,2638	0,1073	0,2659	0,1617	0,0002	-14,1955	-0,0000
$v_{ex,3}$	0,0290	0,2888	0,2528	0,3969	0,1574	0,3034	18,9499	0,0000
$v_{in,3}$	0,0050	0,0969	0,1153	0,2361	-0,0068	-0,0261	-11,1286	-0,0000

Tabela 86 – Parâmetros geométricos dos aerofólios da raiz obtidos nos testes de chance de mutação no algoritmo CST 3D específico (fonte: autor)

Vetor	R_{Le}	A_0	A_1	A_2	A_3	A_4	β	ΔZ_{TE}
$v_{ex,1}$	0,0144	0,3451	0,2681	0,4912	0,2655	0,1817	20,6224	0,0000
$v_{in,1}$	0,0074	0,2562	0,0716	0,3339	0,1284	0,1237	-10,6904	-0,0000
$v_{ex,2}$	0,0092	0,2814	0,1494	0,4957	0,2193	0,2532	19,0682	0,0000
$v_{in,2}$	0,0122	0,2503	-0,0402	0,3181	0,1087	0,0351	-9,8194	-0,0000
$v_{ex,3}$	0,0198	0,2649	0,1116	0,3218	0,1765	0,1289	21,1857	0,0000
$v_{in,3}$	0,0050	0,2827	0,0016	0,2532	0,0108	0,1150	-14,0891	-0,0000

Tabela 87 – Parâmetros geométricos dos aerofólios do meio obtidos nos testes de chance de mutação no algoritmo CST 3D específico (fonte: autor)

Vetor	R_{Le}	A_0	A_1	A_2	A_3	A_4	β	ΔZ_{TE}
$v_{ex,1}$	0,0256	0,2237	0,1890	0,2282	0,2995	0,1740	9,8321	0,0000
$v_{in,1}$	0,0109	0,2306	0,0450	0,1127	-0,1180	-0,1166	-7,1553	-0,0000
$v_{ex,2}$,0141	0,2415	0,1820	0,1746	0,3017	0,1997	8,5702	0,0000
$v_{in,2}$	0,0050	0,1815	0,1508	0,1433	-0,0723	-0,0925	-3,6324	-0,0000
$v_{ex,3}$	0,0248	0,3264	0,1846	0,2643	0,1444	0,2014	8,2519	0,0000
$v_{in,3}$	0,0050	0,1313	0,0049	0,0685	-0,0283	-0,1128	-5,6980	-0,0000

Tabela 88 – Parâmetros geométricos dos aerofólios da ponta obtidos nos testes de chance de mutação no algoritmo CST 3D específico (fonte: autor)

Asa	Tipo	$b[m]$	$b1[m]$	$c_r[m]$	$c_m[m]$	$c_t[m]$
1	1	14,4196	12,9341	1,6238	1,2802	0,8523
2	1	13,4573	12,4573	1,6922	1,5210	0,2725

Tabela 89 – Geometrias das plantas das asas obtidas nos testes de mecanismo de elitismo no algoritmo CST 3D específico - parte 1 (fonte: autor)

Asa	Tipo	$sweep[^\circ]$	$sweep1[^\circ]$	$sweep2[^\circ]$	$tw_t[^\circ]$
1	1	-	4,7390	40,8630	0
2	1	-	4,3942	40,1520	-1,6892

Tabela 90 – Geometrias das plantas das asas obtidas nos testes de mecanismo e elitismo no algoritmo CST 3D específico - parte 2 (fonte: autor)

Vetor	R_{Le}	A_0	A_1	A_2	A_3	A_4	β	ΔZ_{TE}
$v_{ex,1}$	0,0175	0,4028	0,2431	0,4766	0,2635	0,2621	19,0779	0,0000
$v_{in,1}$	0,0050	0,1378	0,0917	0,3707	0,1176	0,1110	-12,6701	-0,0000
$v_{ex,2}$	0,0150	0,3542	0,3015	0,5047	0,2418	0,2262	21,3119	0,0000
$v_{in,2}$	0,0050	0,1534	0,0889	0,3415	0,0742	-0,0057	-15,4614	-0,0000

Tabela 91 – Parâmetros geométricos dos aerofólios da raiz obtidos nos testes de mecanismo de elitismo no algoritmo CST 3D específico (fonte: autor)

Vetor	R_{Le}	A_0	A_1	A_2	A_3	A_4	β	ΔZ_{TE}
$v_{ex,1}$	0,0284	0,4110	0,1217	0,3139	0,1423	0,2492	19,6929	0,0000
$v_{in,1}$	0,0050	0,2340	0,1090	0,3469	0,1573	0,1438	-13,9476	-0,0000
$v_{ex,2}$	0,0255	0,3337	0,2521	0,3988	0,1493	0,1539	12,2798	0,0000
$v_{in,2}$	0,0050	0,1256	0,0076	0,2135	0,1413	-0,0226	-3,3896	-0,0000

Tabela 92 – Parâmetros geométricos dos aerofólios do meio obtidos nos testes de mecanismo de elitismo no algoritmo CST 3D específico (fonte: autor)

Vetor	R_{Le}	A_0	A_1	A_2	A_3	A_4	β	ΔZ_{TE}
$v_{ex,1}$	0,0218	0,2030	0,2301	0,1538	0,2018	0,2213	8,3474	0,0000
$v_{in,1}$	0,0073	0,1928	-0,0069	0,1034	-0,0475	0,0419	-5,3323	-0,0000
$v_{ex,2}$	0,0177	0,2904	0,1975	0,3180	0,2444	0,3221	9,3622	0,0000
$v_{in,2}$	0,0050	0,1614	0,0560	0,0825	-0,0300	0,0461	-1,8877	-0,0000

Tabela 93 – Parâmetros geométricos dos aerofólios da ponta obtidos nos testes de mecanismo de elitismo no algoritmo CST 3D específico (fonte: autor)

APÊNDICE Q – Aerofólios e asas resultantes dos estudos de caso

Q.1 Primeiro estudo de caso: testes independentes

Q.1.1 Algoritmo CST

Vetor	R_{Le}	A_0	A_1	A_2	β	ΔZ_{TE}
$v_{ex,1a}$ 0,0450	0,1500	0,1000	0,1000	16,0000	0,0000	
$v_{in,1a}$ 0,0100	0,0500	0,0250	0,1000	4,0000	0,0000	
$v_{ex,2a}$ 0,0100	0,1500	0,1500	0,2250	14,0000	0,0000	
$v_{in,2a}$ 0,0100	0,1000	0,1000	0,0750	6,0000	0,0000	
$v_{ex,3a}$ 0,0200	0,2750	0,2500	0,0500	13,0000	0,0000	
$v_{in,3a}$ 0,0300	0,1000	0,0250	-0,0000	9,0000	0,0000	

Tabela 94 – Parâmetros geométricos dos aerofólios obtidos no primeiro estudo de caso do algoritmo CST (fonte: autor)

Q.1.2 Algoritmo NACA 4 dígitos 3D

Asa	Tipo	$b[m]$	$b1[m]$	$c_r[m]$	$c_m[m]$	$c_t[m]$
1	1	10,5	9,5	1,4	1,4	1,4
2	1	14,5	12	1,6	1,6	0,7
3	1	12,5	11	1,8	1,7	1,6

Tabela 95 – Geometrias das plantas das asas obtidas no primeiro estudo de caso do algoritmo NACA 4 dígitos 3D - parte 1 (fonte: autor)

Asa	Tipo	$sweep[^\circ]$	$sweep1[^\circ]$	$sweep2[^\circ]$	$tw_t[^\circ]$
1	1	-	13	4	-1
2	1	-	4	14	-1
3	1	-	0	7	-0,5

Tabela 96 – Geometrias das plantas das asas obtidas no primeiro estudo de caso do algoritmo NACA 4 dígitos 3D - parte 2 (fonte: autor)

Asa	Tipo	Perfil da raiz	Perfil do meio	Perfil da ponta
1	1	NACA 3310	NACA 3416	NACA 4113
2	1	NACA 3111	NACA 3413	NACA 0012
3	1	NACA 2311	NACA 4210	NACA 4416

Tabela 97 – Aerofólios das asas obtidas no primeiro estudo de caso do algoritmo NACA 4 dígitos 3D (fonte: autor)

Q.1.3 Algoritmo CST 3D

Asa	Tipo	$b[m]$	$b1[m]$	$c_r[m]$	$c_m[m]$	$c_t[m]$
1	1	14,5	8	1,9	1,2	1
2	1	14,5	9	1,2	0,9	0,5
3	1	12	10,5	1,7	1	0,5

Tabela 98 – Geometrias das plantas das asas obtidas no primeiro estudo de caso do algoritmo CST 3D - parte 1 (fonte: autor)

Asa	Tipo	$sweep[^\circ]$	$sweep1[^\circ]$	$sweep2[^\circ]$	$tw_t[^\circ]$
1	1	-	4	9	-0,5
2	1	-	0	4	-0,25
3	1	-	0	8	0

Tabela 99 – Geometrias das plantas das asas obtidas no primeiro estudo de caso do algoritmo CST 3D - parte 2 (fonte: autor)

Vetor	R_{Le}	A_0	A_1	A_2	β	ΔZ_{TE}
$v_{ex,1}$	0,0300	0,0500	0,3000	0,1500	19,0000	0,0000
$v_{in,1}$	0,0100	0,0500	0,1000	-0,0000	2,0000	0,0000
$v_{ex,2}$	0,0200	0,0500	0,2000	0,2000	14,0000	0,0000
$v_{in,2}$	0,0200	0,0500	0,0500	0,1000	7,0000	0,0000
$v_{ex,3}$	0,0100	0,0500	0,1500	0,1000	18,0000	0,0000
$v_{in,3}$	0,0200	0,0500	0,1000	-0,0000	2,0000	0,0000

Tabela 100 – Parâmetros geométricos dos aerofólios da raiz obtidos no primeiro estudo de caso do algoritmo CST 3D (fonte: autor)

Vetor	R_{Le}	A_0	A_1	A_2	β	ΔZ_{TE}
$v_{ex,1}$	0,0400	0,1000	0,3000	0,1500	16,0000	0,0000
$v_{in,1}$	0,0200	-0,0000	0,1000	0,0500	7,0000	0,0000
$v_{ex,2}$	0,0100	0,1500	0,2000	0,1500	18,0000	0,0000
$v_{in,2}$	0,0100	0,0500	0,1000	0,0500	2,0000	0,0000
$v_{ex,3}$	0,0500	0,1000	0,3000	0,2000	16,0000	0,0000
$v_{in,3}$	0,0300	0,1000	-0,0000	0,0500	8,0000	0,0000

Tabela 101 – Parâmetros geométricos dos aerofólios do meio obtidos no primeiro estudo de caso do algoritmo CST 3D (fonte: autor)

Vetor	R_{Le}	A_0	A_1	A_2	β	ΔZ_{TE}
$v_{ex,1}$	0,0500	0,1500	0,0000	0,1500	10,0000	0,0000
$v_{in,1}$	0,0300	0,0500	0,1000	0,0500	14,0000	0,0000
$v_{ex,2}$	0,0100	0,2500	0,2500	0,0500	15,0000	0,0000
$v_{in,2}$	0,0200	0,1000	0,1000	-0,0000	11,0000	0,0000
$v_{ex,3}$	0,0500	0,2500	0,1500	0,3000	12,0000	0,0000
$v_{in,3}$	0,0100	0,1000	0,0500	0,1000	14,0000	0,0000

Tabela 102 – Parâmetros geométricos dos aerofólios da ponta obtidos no primeiro estudo de caso do algoritmo CST 3D (fonte: autor)

Q.2 Segundo estudo de caso: detalhando aerofólios

Vetor	R_{Le}	A_0	A_1	A_2	β	ΔZ_{TE}
$v_{ex,1b}$	0,0150	0,2250	0,2250	0,3000	20,0000	0,0000
$v_{in,1b}$	0,0100	0,0500	0,0750	0,0500	2,0000	0,0000
$v_{ex,2b}$	0,0200	0,2250	0,2750	0,2750	19,0000	0,0000
$v_{in,2b}$	0,0100	0,0500	0,0500	-0,0000	1,0000	0,0000
$v_{ex,3b}$	0,0250	0,2250	0,2500	0,2750	20,0000	0,0000
$v_{in,3b}$	0,0100	0,1000	0,0250	-0,0000	0,0000	0,0000

Tabela 103 – Parâmetros geométricos dos aerofólios CST obtidos no segundo estudo de caso (fonte: autor)

Q.3 Terceiro estudo de caso: coeficientes de momento constantes

Vetor	R_{Le}	A_0	A_1	A_2	β	ΔZ_{TE}
$v_{ex,1c}$	0,0150	0,2250	0,2500	0,2500	19,0000	0,0000
$v_{in,1c}$	0,0150	0,0250	0,0250	-0,0000	1,0000	0,0000
$v_{ex,2c}$	0,0300	0,3000	0,2000	0,0000	17,0000	0,0000
$v_{in,2c}$	0,0100	0,0750	0,0500	0,0250	3,0000	0,0000

Tabela 104 – Parâmetros geométricos dos aerofólios CST obtidos no terceiro estudo de caso (fonte: autor)

Q.4 Quarto estudo de caso: testes integrados

Asa	Tipo	$b[m]$	$b1[m]$	$c_r[m]$	$c_m[m]$	$c_t[m]$
1	1	14	11,6147	1,4525	0,9001	0,5791
2	1	14	12,1	1,45	1,1	0,55

Tabela 105 – Geometrias das plantas das asas obtidas no quarto estudo de caso - parte 1 (fonte: autor)

Asa	Tipo	$sweep[^\circ]$	$sweep1[^\circ]$	$sweep2[^\circ]$	$tw_t[^\circ]$
1	1	-	4,5783	23,7687	0
2	1	-	2	16	0

Tabela 106 – Geometrias das plantas das asas obtidas no quarto estudo de caso - parte 2 (fonte: autor)

Vetor	R_{Le}	A_0	A_1	A_2	β	ΔZ_{TE}
$v_{ex,1}$	0,0150	0,2500	0,2750	0,1750	8,0000	0,0000
$v_{in,1}$	0,0250	0,1000	0,1000	0,1000	2,0000	0,0000
$v_{ex,2}$	0,0400	0,0000	0,3000	0,0000	12,0000	0,0000
$v_{in,2}$	0,0100	0,0500	0,1000	0,0500	10,0000	0,0000

Tabela 107 – Parâmetros geométricos dos aerofólios da raiz obtidos no quarto estudo de caso (fonte: autor)

Vetor	R_{Le}	A_0	A_1	A_2	β	ΔZ_{TE}
$v_{ex,1}$	0,0200	0,1500	0,1250	0,1500	8,0000	0,0000
$v_{in,1}$	0,0050	0,0000	0,0500	0,0000	2,0000	0,0000
$v_{ex,2}$	0,0100	0,3000	0,1500	0,0500	16,0000	0,0000
$v_{in,2}$	0,0300	0,0500	0,1000	-0,0000	5,0000	0,0000

Tabela 108 – Parâmetros geométricos dos aerofólios do meio obtidos no quarto estudo de caso (fonte: autor)

Vetor	R_{Le}	A_0	A_1	A_2	β	ΔZ_{TE}
$v_{ex,1}$	0,0100	0,2000	0,2000	0,0000	10,0000	0,0000
$v_{in,1}$	0,0050	0,0000	0,0750	0,0000	5,0000	0,0000
$v_{ex,2}$	0,0500	0,0000	0,2500	0,0500	20,0000	0,0000
$v_{in,2}$	0,0300	0,0500	0,1000	0,1000	10,0000	0,0000

Tabela 109 – Parâmetros geométricos dos aerofólios da ponta obtidos no quarto estudo de caso (fonte: autor)

Q.5 Quinto estudo de caso: comparação com o VLM

Asa	Tipo	$b[m]$	$b1[m]$	$c_r[m]$	$c_m[m]$	$c_t[m]$
1	1	12	8	1,8	1,4	1,4
2	1	14	-	1	-	0,5

Tabela 110 – Geometrias das plantas das asas obtidas no quinto estudo de caso - parte 1 (fonte: autor)

Asa	Tipo	$sweep[^\circ]$	$sweep1[^\circ]$	$sweep2[^\circ]$	$tw_t[^\circ]$
1	1	-	4	11	-3,5
2	1	8	-	-	-1,75

Tabela 111 – Geometrias das plantas das asas obtidas no quinto estudo de caso - parte 2 (fonte: autor)

Asa	Tipo	Perfil da raiz	Perfil do meio	Perfil da ponta
1	1	NACA 0010	NACA 0010	NACA 1111
2	1	NACA 3320	-	NACA 3311

Tabela 112 – Aerofólios das asas obtidas no quinto estudo de caso (fonte: autor)