

UNIVERSIDADE ESTADUAL PAULISTA

"JÚLIO DE MESQUITA FILHO"

FACULDADE DE ENGENHARIA

CAMPUS DE SÃO JOÃO DA BOA VISTA

GABRIEL CLEMENTE CARRARI

**Aprendizado de Máquina para Modelagem Substituta em
Otimização Multiobjetivo de Asas**

São João da Boa Vista

2025

Gabriel Clemente Carrari

Aprendizado de Máquina para Modelagem Substituta em Otimização Multiobjetivo de Asas

Trabalho de Graduação apresentado ao Conselho de Curso de Graduação em Engenharia Aeronáutica do Campus de São João da Boa Vista, Universidade Estadual Paulista, como parte dos requisitos para obtenção do diploma de Graduação em Engenharia Aeronáutica .

Orientador: Prof^o Dr. Gabriel Pereira Gouveia da Silva

São João da Boa Vista

2025

C313a

Carrari, Gabriel

Aprendizado de Máquina para Modelagem Substituta em
Otimização Multiobjetivo de Asas / Gabriel Carrari. -- São João da
Boa Vista, 2025

73 p. : il., tabs.

Trabalho de conclusão de curso (Bacharelado - Engenharia
Aeronáutica) - Universidade Estadual Paulista (UNESP), Faculdade
de Engenharia, São João da Boa Vista

Orientador: Gabriel Pereira Gouveia da Silva

1. Aeronáutica. 2. Aprendizado do computador. 3. Otimização
matemática. 4. Aviões Asas. 5. Escoamento Potencial. I. Título.

**UNIVERSIDADE ESTADUAL PAULISTA “JÚLIO DE MESQUITA FILHO”
FACULDADE DE ENGENHARIA - CÂMPUS DE SÃO JOÃO DA BOA VISTA
GRADUAÇÃO EM ENGENHARIA AERONÁUTICA**

TRABALHO DE CONCLUSÃO DE CURSO

**APRENDIZADO DE MÁQUINA PARA MODELAGEM SUBSTITUTA EM
OTIMIZAÇÃO MULTI OBJETIVO DE ASAS**

Aluno: Gabriel Clemente Carrari

Orientador: Prof. Dr. Gabriel Pereira Gouveia da Silva

Banca Examinadora:

- Gabriel Pereira Gouveia da Silva (Orientador)
- Leandra Isabel De Abreu (Examinador)
- Marlon Rodrigues Garcia (Examinador)

Os formulários de avaliação e a ata da defesa, na qual consta a aprovação do trabalho, devidamente assinados pela banca encontram-se no prontuário eletrônico do aluno.

São João da Boa Vista, 02 de dezembro de 2025

AGRADECIMENTOS

Agradeço à Universidade Estadual Paulista (UNESP) pela formação acadêmica, pelo ambiente de aprendizado e por todo o suporte institucional que possibilitou o desenvolvimento deste trabalho.

O presente trabalho foi realizado com apoio da Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP), Brasil. Processo nº 2024/07804-2. Sendo assim, agradeço pelo apoio financeiro que viabilizou a realização desta pesquisa e contribuiu diretamente para o meu crescimento científico e profissional, aproximando-me de temas que posteriormente se mostraram de grande interesse e utilidade.

Manifesto minha sincera gratidão ao meu orientador, Prof. Gabriel Pereira Gouveia da Silva, pela orientação, paciência e pela confiança depositada em mim ao longo de todo o processo. Sua contribuição foi essencial para a conclusão deste trabalho. Expressando também minha gratidão a todos os professores que fizeram parte da minha trajetória acadêmica. Cada disciplina, orientação, conselho e desafio apresentado ao longo do curso contribuiu de maneira significativa para minha formação, ampliando minha visão e fortalecendo meu desenvolvimento como estudante e profissional.

Agradeço, igualmente, aos amigos que estiveram ao meu lado durante essa jornada, oferecendo apoio, motivação e companhia nos momentos de maior desafio.

À minha família, em especial aos meus pais, Elaine Cristina Clemente Carrari e Luis Eduardo Carrari, deixo meu profundo agradecimento por seu amor incondicional, incentivo constante e por sempre acreditarem no meu potencial, tornando possível toda essa experiência.

Por fim, e mais importante, agradeço a Deus por me conceder força, saúde e sabedoria, guiando-me para superar cada etapa deste caminho.

O presente trabalho contou com apoio financeiro, conforme processo nº 2024/07804-2, Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP), Brasil. As opiniões, hipóteses e conclusões ou recomendações expressas neste material são de responsabilidade dos autores e não necessariamente refletem a visão da FAPESP.

*“Imagination is more important than knowledge. Knowledge is limited. Imagination encircles
the world.”
(Albert Einstein)*

RESUMO

Projetos aeronáuticos são multidisciplinares, e otimizar uma tecnologia isoladamente pode afetar negativamente outras igualmente importantes. Por isso, esses projetos se beneficiam de otimizações multiobjetivo e multidisciplinares, viáveis desde as fases iniciais. No entanto, quanto mais complexos os modelos simulados no laço de otimização, maior o custo computacional. Uma solução eficiente é o uso de modelos substitutos, especialmente os baseados em aprendizado de máquina, que aprendem e extraem padrões úteis para previsão e otimização. Este trabalho teve como objetivo implementar modelos substitutos baseados em aprendizado de máquina para otimização aerodinâmica de asas subsônicas, com dados obtidos pelo método da rede de vórtices (VLM) e pela teoria da linha sustentadora não linear (N-LLT). Foi criado um banco com milhões de geometrias distintas, simuladas a partir de variações de parâmetros geométricos. Os modelos foram treinados para prever os coeficientes C_L , C_D e C_M , e utilizados em dois algoritmos de otimização: Dual Annealing e Algoritmo Genético. Os resultados do modelo substituto tiveram boa precisão nas previsões, com desempenho aerodinâmico próximo ao obtido nas otimizações com métodos iterativos, mas com redução de tempo de execução de cerca de 3 ordens de grandeza. A abordagem mostra grande potencial para aplicações em otimização de projeto, onde eficiência computacional é crítica.

PALAVRAS-CHAVE: otimização multiobjetivo; otimização multidisciplinar; modelos substitutos; aprendizado de máquina; aerodinâmica; asas subsônicas; rede de vórtices; dual annealing; algoritmo genético; projeto aeronáutico.

ABSTRACT

Aeronautical designs are inherently multidisciplinary, and optimizing a single technology in isolation can negatively impact other equally important ones. Therefore, such projects benefit from multi-objective and multidisciplinary optimization approaches, which are feasible even in early design stages. However, the more complex the models simulated within the optimization loop, the higher the computational cost. An efficient solution is the use of surrogate models, especially those based on machine learning, which learn and extract useful patterns for prediction and optimization. This work aimed to implement machine-learning-based surrogate models for the aerodynamic optimization of subsonic wings, using data generated by the Vortex Lattice Method (VLM) and the nonlinear lifting-line theory (N-LLT). A database with millions of distinct geometries was created, simulated from variations of geometric parameters. The surrogate models were trained to predict the aerodynamic coefficients C_L , C_D and C_M were used in two optimization algorithms: Dual Annealing and Genetic Algorithm. The surrogate models demonstrated high predictive accuracy, achieving aerodynamic performance close to that obtained through iterative simulation methods, while reducing execution time by approximately three orders of magnitude. The approach shows strong potential for application in design optimization, where computational efficiency is critical.

KEYWORDS: multi-objective optimization; multidisciplinary optimization; surrogate models; machine learning; aerodynamics; subsonic wings; vortex lattice method; vlm; dual annealing; genetic algorithm; aircraft design.

LISTA DE ILUSTRAÇÕES

Figura 1 – Desenho esquemático com vista frontal e em perspectiva para elucidar como se formam os vórtices de ponta de asa durante o voo.	18
Figura 2 – Desenho esquemático das forças atuantes em um perfil aerodinâmico sob efeito do downwash, responsável por reduzir o ângulo de ataque efetivo em uma asa tridimensional.	19
Figura 3 – Modelo de anéis de vórtice utilizado na formulação do VLM para superfícies sustentadoras finas.	20
Figura 4 – Representação da distribuição de circulação e dos vórtices ferradura em uma asa utilizando a LLT.	22
Figura 5 – Estrutura típica de uma rede neural do tipo MLP mostrando suas camadas e neurônios, juntamente com a influência que cada neurônio exerce sobre os outros.	23
Figura 6 – Fluxograma do algoritmo Dual Annealing utilizado para otimizações globais de objetivo único.	26
Figura 7 – Fluxograma do algoritmo NSGA-II utilizado nas otimizações para encontrar as fronteiras de Pareto.	28
Figura 8 – Diagrama esquemático da geometria geral de uma asa fixa com sua seção transversal (esquerda) e vista de cima (direita).	31
Figura 9 – 12 parâmetros PARSEC utilizados para descrever um perfil aerodinâmico.	33
Figura 10 – Perfis aerodinâmicos normalizados pela corda que foram utilizados nas análises de convergência.	35
Figura 11 – Efeito da escolha de valores pares para o número de seções da semi-envergadura ($SECT_{TESSu}$), com $OUT_{CLUSTER} = 1$ e $N_{AIRFOIL} = 40$, na qualidade dos resultados e no tempo de simulação.	36
Figura 12 – Efeito da escolha de valores ímpares para o número de seções da semi-envergadura ($SECT_{TESSu}$), com $OUT_{CLUSTER} = 1$ e $N_{AIRFOIL} = 40$, na qualidade dos resultados e no tempo de simulação.	36
Figura 13 – Influência da escolha de $OUT_{CLUSTER}$ na precisão dos resultados e no tempo de simulação.	37
Figura 14 – Influência da escolha de $N_{AIRFOIL}$ fixando $OUT_{CLUSTER} = 0.1$ e $SECT_{TESSu} = 7$ sobre a precisão e tempo de simulação.	38
Figura 15 – Vista em perspectiva da malha final utilizada nas simulações para uma configuração genérica de asa.	38
Figura 16 – Arquitetura genérica das redes neurais, sendo 18 entradas, 1 saída, N_1 neurônios na primeira camada oculta e N_2 na segunda camada oculta.	40
Figura 17 – Exemplo de aerofólio viável (esquerda) e aerofólio inviável (direita) gerados a partir de combinações de parâmetros PARSEC.	41

Figura 18 – Comparação entre as curvas reais dos aerofólios e as curvas ajustadas por polinômios para um aerofólio de geometria válida e para outro de geometria inválida.	41
Figura 19 – Dispersão das distâncias euclidianas calculadas para o intradorso e extradorso de cada um dos perfis aerodinâmicos do banco de dados PARSEC.	42
Figura 20 – Dispersão dos Erros Quadráticos Médios para o intradorso e extradorso de cada um dos perfis aerodinâmicos do banco de dados PARSEC.	42
Figura 21 – Comparação entre métodos considerados para estimativa de estol.	44
Figura 22 – Curvas de sustentação do modelo N-LLT implementado e XFLR5 comparadas.	44
Figura 23 – Curvas de polar de arrasto do modelo N-LLT implementado e XFLR5 comparadas.	45
Figura 24 – Curvas de momento de arfagem do modelo N-LLT implementado e XFLR5 comparadas.	45
Figura 25 – Influência da escolha do número de seções da envergadura ($N_{stations}$) na precisão e tempo de simulação.	46
Figura 26 – Influência da escolha do número de seções da envergadura ($N_{stations}$) na curva de sustentação.	47
Figura 27 – Influência da escolha do número de seções da envergadura ($N_{stations}$) na polar de arrasto.	47
Figura 28 – Influência da escolha do número de seções da envergadura ($N_{stations}$) na curva de momento de arfagem.	48
Figura 29 – Comparação da distribuição normalizada dos dados das saídas dos algoritmos VLM e N-LLT.	53
Figura 30 – Distribuição dos dados das saídas de estol da N-LLT implementada.	54
Figura 31 – Curvas de aprendizado do treinamento final para as saídas de sustentação realizado com o banco de dados VLM.	55
Figura 32 – Curvas de aprendizado do treinamento final para as saídas de momento de arfagem realizado com o banco de dados VLM.	56
Figura 33 – Curvas de aprendizado do treinamento final para as saídas de arrasto realizado com o banco de dados VLM.	56
Figura 34 – Aerofólio 1459 resultante das otimizações realizadas com VLM e sua rede neural.	58
Figura 35 – Aerofólios resultantes das otimizações conduzidas no OpenVSP (esquerda) e Rede Neural (direita).	59
Figura 36 – Curvas de aprendizado do treinamento final para as saídas de sustentação realizado com o banco de dados N-LLT.	61
Figura 37 – Curvas de aprendizado do treinamento final para as saídas de momento de arfagem realizado com o banco de dados N-LLT.	61
Figura 38 – Curvas de aprendizado do treinamento final para as saídas de arrasto realizado com o banco de dados N-LLT.	62

Figura 39 – Comparação dos aerofólios do banco de dados PARSEC obtidos nas otimizações realizadas com a N-LLT e sua rede neural. Aerofólio 1406 (esquerda) e 1331 (direita).	63
Figura 40 – Aerofólio resultante da otimização completa com a rede neural baseada na N-LLT.	65
Figura 41 – Comparação dos aerofólios do banco de dados PARSEC obtidos nas otimizações realizadas com a N-LLT e sua rede neural. Aerofólio 1329 (esquerda) e 1328 (direita).	66
Figura 42 – Fronteira de Pareto obtida a partir da otimização multiobjetivo com a rede VLM utilizando NSGA-II.	66
Figura 43 – Fronteira de Pareto obtida a partir da otimização multiobjetivo com a rede N-LLT utilizando NSGA-II.	67

LISTA DE TABELAS

Tabela 1 – Parâmetros de Design.	32
Tabela 2 – Parâmetros do aerofólio com valores mínimos, máximos, médias e desvios padrões.	34
Tabela 3 – Valores utilizados para descrever a malha nas simulações.	38
Tabela 4 – Combinações de Variáveis	39
Tabela 5 – Parâmetros de Design.	46
Tabela 6 – Combinações de Variáveis N-LLT	48
Tabela 7 – Mínimos, máximos, médias e desvios padrões para cada uma das saídas do VLM.	52
Tabela 8 – Mínimos, máximos, médias e desvios padrões para cada uma das saídas da N-LLT.	52
Tabela 9 – Número de neurônios (N1) e erro médio quadrático (RMSE) para cada saída.	54
Tabela 10 – Número de neurônios (N1 e N2) e RMSE para cada saída.	55
Tabela 11 – Comparação de resultados das otimizações com 6 parâmetros no VLM e Rede Neural.	57
Tabela 12 – Comparação de resultados das otimizações com 17 parâmetros no VLM e Rede Neural.	59
Tabela 13 – Número de neurônios (N1, N2 e N3) e RMSE para cada saída.	60
Tabela 14 – Comparação de resultados das otimizações com 5 parâmetros na N-LLT e Rede Neural.	63
Tabela 15 – Comparação de resultados das otimizações com 15 parâmetros no N-LLT e Rede Neural.	64
Tabela 16 – Comparação de resultados das otimizações com 5 parâmetros para previsão de estol na N-LLT e Rede Neural.	65

SUMÁRIO

1	INTRODUÇÃO	15
2	OBJETIVOS	17
3	FUNDAMENTAÇÃO TEÓRICA E REVISÃO BIBLIOGRÁFICA	18
3.1	Teoria de Asa Finita	18
3.2	Método Vortex Lattice (VLM)	20
3.3	Método Nonlinear Lifting-Line (N-LLT)	21
3.4	Redes Neurais do Tipo Multi-Layer Perceptron (MLP)	22
3.5	Algoritmo Dual Annealing	25
3.6	Algoritmo Genético NSGA-II	27
3.7	Revisão de trabalhos semelhantes	28
4	METODOLOGIA	30
4.1	Método VLM	30
4.1.1	Definição das Variáveis	30
4.1.2	Malha e Convergência	35
4.1.3	Banco de Dados	39
4.1.4	Arquitetura e Treinamento da Rede Neural	39
4.1.5	Algoritmo de Classificação de Aerofólios	40
4.2	Método N-LLT	43
4.2.1	Definição de Variáveis	45
4.2.2	Malha e Convergência	46
4.2.3	Banco de Dados	48
4.2.4	Arquitetura e Treinamento da Rede Neural	49
4.3	Otimização Multiobjetivo	49
5	RESULTADOS	52
5.1	Banco de Dados	52
5.2	Método VLM	54
5.2.1	Arquitetura e Treinamento da Rede Neural	54
5.2.2	Comparações entre Rede Neural e VLM	57
5.3	Método N-LLT	60
5.3.1	Arquitetura e Treinamento da Rede Neural	60
5.3.2	Comparações entre Rede Neural e N-LLT	62
5.4	Otimização Multiobjetivo	66
6	CONCLUSÕES	69

7	SUGESTÕES PARA TRABALHOS FUTUROS	70
	REFERÊNCIAS	71

1 INTRODUÇÃO

Embora o transporte aéreo seja uma das atividades humanas mais intensivas quanto à emissão de gases de efeito estufa, sua participação relativa nas emissões globais é atenuada pelo acesso mais restrito a esse meio de transporte em comparação com modais mais acessíveis (Gössling; Humpe, 2020). Entretanto, segundo dados da Agência Internacional de Energia da Organização para a Cooperação e Desenvolvimento Econômico (IEA-OECD), estima-se que a quantidade de dióxido de carbono emitida por passageiro por quilômetro percorrido seja cerca de quatro vezes maior para o transporte aéreo que para transporte por carros (International energy agency, 2009). Este fato torna o aumento das emissões globais bastante sensível ao constante crescimento do setor de transporte aéreo, fazendo com que haja um crescente conflito entre a aviação civil e as metas globais de descarbonização (Gössling; Humpe, 2020). Em face das alterações climáticas antrópicas, que representam uma ameaça ambiental e social, e desta maior sensibilidade do total de emissões em relação ao crescimento da aviação, operadores, fabricantes e agências regulatórias do setor de transporte aéreo civil estabeleceram uma meta internacional de neutralidade de carbono na aviação comercial até 2050 (Air transport action group, 2021; International civil aviation organization, 2022).

Embora as principais soluções buscadas para o cumprimento desta meta de descarbonização da aviação até 2050 sejam voltadas para a substituição da matriz energética por fontes mais limpas, como combustíveis sustentáveis de aviação (SAFs), aeronaves elétricas e propulsão a hidrogênio (International air transport association, 2023), também é possível reduzir o consumo energético e conseqüentemente a emissão de poluentes através da redução da potência requerida para voo, o que pode ser obtido através de um melhor projeto aerodinâmico (Anderson, 2010; Filippone, 2012). Nas fases iniciais de projeto, quando há muitas soluções possíveis para atender aos requisitos e restrições, é inviável a utilização de ferramentas de alta fidelidade de simulação que possuem elevado custo computacional, e as ferramentas mais adequadas para cálculos aerodinâmicos são ferramentas baseadas em escoamento potencial, como método de rede de vórtices (VLM), teoria da linha sustentadora (LLT) e método dos painéis (Anderson, 2010; Gudmundsson, 2013). Conforme avança-se no projeto, mais informações sobre a aeronave se tornam consolidadas e o uso de ferramentas de maior fidelidade passa a ser viável, uma vez que poucas configurações serão testadas e já não há margem para grandes modificações no projeto.

Devido aos altos custos de modificação do projeto em fases mais avançadas, há um benefício em realizar otimizações desde as fases iniciais de projeto. Projetos aeronáuticos, entretanto, são intrinsecamente multidisciplinares e complexos e a melhor solução de projeto do ponto de vista de uma característica pode prejudicar o desempenho global mais do que o beneficiar (Gudmundsson, 2013). Sendo assim, é necessário realizar uma abordagem de otimização multiobjetivo e multidisciplinar, o que torna esta atividade ainda mais intensiva do ponto de vista computacional. Uma maneira de reduzir a demanda por recursos computacionais em uma otimização multidisciplinar é a utilização de modelos substitutos representativos do sistema que

está sendo otimizado (Wang; Shan, 2007). Modelos substitutos são aproximações da física do fenômeno obtidos a partir de um grande número de resultados experimentais ou de simulações, e são formulados de tal maneira que seus resultados são calculados diretamente a partir de um conjunto de entradas, eliminando o tempo consumido no processamento de cálculos iterativos (Wang; Shan, 2007).

Embora haja uma miríade de técnicas de modelagem substituta na literatura, as técnicas de aprendizado de máquina que tem se desenvolvido rapidamente nos últimos anos se mostram bastante promissoras na resolução de problemas relacionados a escoamentos de fluidos, como é o caso do escoamento ao redor das asas e empenagens de uma aeronave (Brunton; Noack; Koumoutsakos, 2020). Neste contexto, o trabalho de conclusão de curso visou a implementação e validação de modelos substitutos baseados em aprendizado de máquina para cálculo das características aerodinâmicas subsônicas de uma asa tridimensional. Estes modelos poderão então ser utilizados de maneira mais eficiente em otimizações multidisciplinares da geometria de uma asa ou de empenagens.

Esta monografia final apresenta o desenvolvimento do projeto proposto, descrevendo desde a construção da base de dados e escolha das arquiteturas dos modelos até os procedimentos de treinamento, validação e análise de desempenho, de modo a demonstrar a eficácia e as limitações dos modelos substitutos obtidos.

2 OBJETIVOS

Este estudo teve como objetivo avaliar o desempenho de modelos substitutos baseados em aprendizado de máquina aplicados ao cálculo aerodinâmico subsônico tridimensional de asas. A meta central consistiu em desenvolver um banco de dados capaz de relacionar diversas combinações de parâmetros geométricos de asas tridimensionais com seus respectivos parâmetros de desempenho aerodinâmico. Esse banco de dados foi utilizado para treinar modelos de aprendizado de máquina, que podem atuar como substitutos ao *solver* de escoamento potencial tradicional. Além disso, foram conduzidas otimizações multiobjetivo utilizando tanto os modelos substitutos quanto o *solver* convencional, permitindo avaliar diferenças nos resultados obtidos e no custo computacional associado.

Com base nesses objetivos gerais, estabeleceram-se os seguintes objetivos específicos:

1. Gerar um banco de dados abrangente a partir de simulações VLM, contendo combinações variadas de parâmetros geométricos de asas tridimensionais e seus respectivos coeficientes aerodinâmicos.
2. Gerar um segundo banco de dados, também abrangente, utilizando o método N-LLT, explorando as mesmas variações geométricas e registrando seus coeficientes aerodinâmicos correspondentes.
3. Comparar os bancos de dados produzidos pelos métodos VLM e N-LLT, avaliando a coerência entre os resultados e identificando diferenças inerentes às formulações de cada modelo.
4. Desenvolver, treinar e validar modelos substitutos baseados em aprendizado de máquina capazes de prever coeficientes aerodinâmicos utilizando ambos os bancos de dados.
5. Avaliar o desempenho dos modelos substitutos em relação aos métodos originais VLM e N-LLT, quantificando diferenças de precisão e tempo computacional.
6. Realizar otimizações multiobjetivo empregando tanto os modelos substitutos quanto os *solvers* originais (VLM e N-LLT), de modo a comparar os resultados obtidos em cada abordagem.
7. Quantificar os ganhos de eficiência computacional obtidos pelo uso dos modelos substitutos em cenários típicos de otimização aerodinâmica.

3 FUNDAMENTAÇÃO TEÓRICA E REVISÃO BIBLIOGRÁFICA

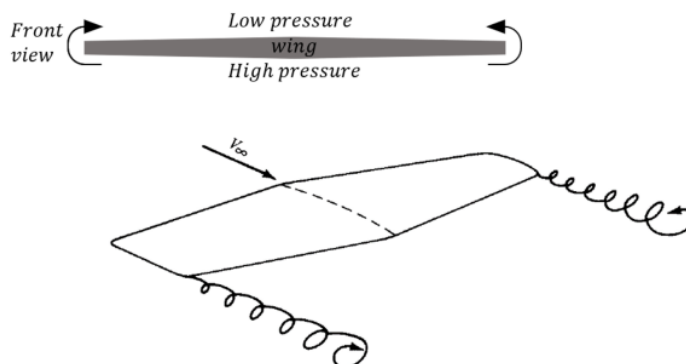
O presente capítulo tem como objetivo apresentar os fundamentos teóricos que sustentam o desenvolvimento deste trabalho, oferecendo uma visão concisa das principais formulações, modelos e algoritmos empregados ao longo da pesquisa. São introduzidas as bases da aerodinâmica de asas tridimensionais, incluindo a teoria de asa finita e os métodos computacionais VLM e N-LLT, assim como os conceitos essenciais de redes neurais do tipo MLP utilizadas como modelos substitutos. Além disso, descrevem-se os métodos de otimização aplicados: Dual Annealing e NSGA-II, destacando seus princípios gerais e suas características mais relevantes. Essa fundamentação fornece o suporte necessário para a compreensão dos capítulos seguintes, nos quais essas ferramentas são aplicadas à formulação e solução dos problemas de interesse.

3.1 TEORIA DE ASA FINITA

A teoria de asa finita descreve o comportamento aerodinâmico de asas tridimensionais, considerando os efeitos induzidos pelos vórtices de ponta que reduzem a eficiência em comparação ao caso bidimensional. A distribuição de circulação ao longo da envergadura determina grandezas como o coeficiente de sustentação tridimensional, o arrasto induzido e o momento aerodinâmico. Essa formulação fundamenta métodos computacionais como o VLM e o N-LLT, amplamente utilizados no projeto conceitual de aeronaves devido à sua boa relação entre precisão e custo computacional.

A formação de sustentação em uma asa está diretamente ligada à existência de uma região de alta pressão na superfície inferior e de baixa pressão na superfície superior. Essa diferença de pressão, quando combinada à geometria da asa e ao ângulo de ataque, gera o campo de forças aerodinâmicas responsável pela sustentação. Entretanto, próximo às pontas da asa, o ar de alta pressão da face inferior escoar lateralmente em direção à face superior, onde a pressão é menor. Esse enrolamento do escoamento dá origem aos vórtices de ponta de asa, que se estendem a jusante e formam uma esteira helicoidal persistente, a qual pode ser vista na Figura 1.

Figura 1 – Desenho esquemático com vista frontal e em perspectiva para elucidar como se formam os vórtices de ponta de asa durante o voo.



Fonte: Adaptado de Anderson (2010).

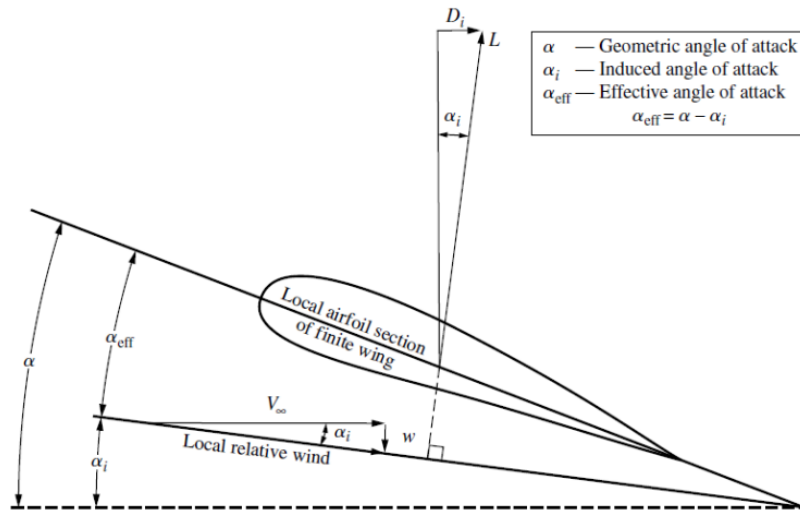
Os vórtices gerados nas pontas induzem na vizinhança da asa uma velocidade descendente, denominada downwash, usualmente representada pelo termo w . Essa velocidade induzida altera a direção do escoamento relativo percebido pela asa, de modo que o vetor velocidade local deixa de ser horizontal (como no caso bidimensional) e passa a possuir um componente vertical orientado para baixo.

Como consequência, o ângulo de ataque efetivamente percebido pelo aerofólio, chamado ângulo de ataque efetivo, representado por α_{eff} , é reduzido em relação ao ângulo geométrico. Essa relação é dada por:

$$\alpha_{eff} = \alpha - \alpha_i \quad (1)$$

onde α_i é o ângulo de ataque induzido, diretamente associado ao downwash. A Figura 2 ilustra essa decomposição entre o ângulo geométrico, o ângulo induzido e o ângulo efetivo, bem como a inclinação da força aerodinâmica resultante.

Figura 2 – Desenho esquemático das forças atuantes em um perfil aerodinâmico sob efeito do downwash, responsável por reduzir o ângulo de ataque efetivo em uma asa tridimensional.



Fonte: Anderson (2010).

Devido à redução do ângulo de ataque efetivo, uma asa tridimensional produz menos sustentação do que um aerofólio bidimensional sob o mesmo ângulo de ataque geométrico. Além disso, a inclinação da força aerodinâmica causada pelo downwash gera um componente horizontal adicional, responsável pelo arrasto induzido. Em regime subsônico, o arrasto induzido pode ser modelado por uma expressão quadrática na sustentação. A polar de arrasto utilizada em análises preliminares é expressa por

$$C_D = C_{D0} + K_2 C_L^2 = C_{D0} + \frac{C_L^2}{\pi e AR}, \quad (2)$$

onde C_D é o coeficiente de arrasto total, C_{D0} representa o arrasto parasita, C_L é o coeficiente de sustentação, K_2 é o fator associado ao termo quadrático referente ao arrasto induzido, e é o fator de eficiência de Oswald, AR é o alongamento (*aspect ratio*) da asa e π é a constante

matemática.

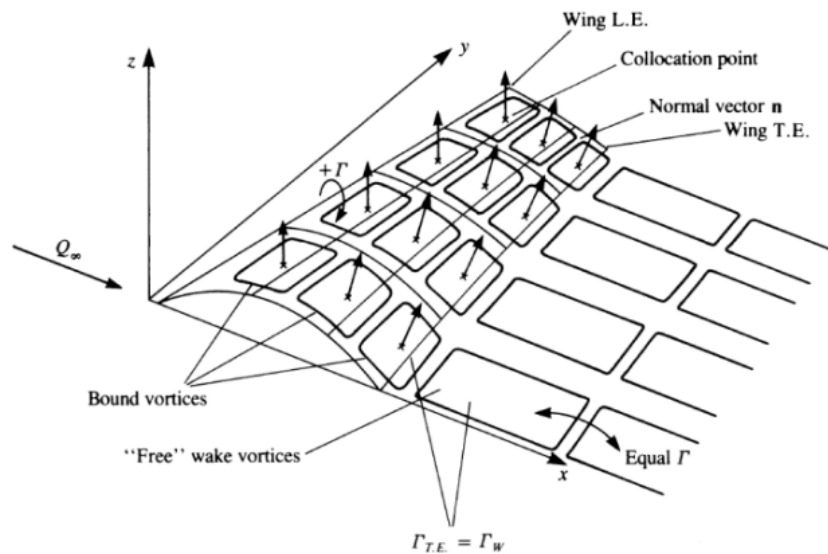
Essa formulação deixa evidente que o arrasto induzido cresce rapidamente com o aumento de C_L , tornando-se dominante em condições de voo de baixa velocidade ou elevada sustentação. Por esse motivo, configurações com maior alongamento ou o emprego de dispositivos como *winglets* reduzem a intensidade dos vórtices de ponta e, portanto, diminuem o arrasto induzido através do termo K_2 , contribuindo diretamente para o aumento do desempenho aeronáutico.

3.2 MÉTODO VORTEX LATTICE (VLM)

O Método Vortex Lattice (VLM) é uma formulação baseada no escoamento potencial incompressível e destina-se à estimativa da distribuição de sustentação em superfícies tridimensionais de espessura desprezível. Sua derivação fundamenta-se nas Equações de Laplace para o potencial de velocidade e no modelo de vorticidade distribuída, oferecendo uma abordagem computacionalmente leve para o cálculo de cargas aerodinâmicas em asas de alongamento moderado a elevado (Katz; Plotkin, 2001).

A primeira etapa do VLM consiste em representar a asa por uma superfície média (*camber surface*) com a mesma geometria em planta, porém de espessura nula. Essa superfície é discretizada em painéis quadriláteros ou retangulares, dentro dos quais são posicionados anéis de vórtice (*vortex rings*) de circulação constante desconhecida, respeitando o Teorema de Helmholtz que garante a conservação da circulação ao longo da estrutura (Katz; Plotkin, 2001). A Figura 3 ilustra esse arranjo clássico de anéis de vórtice utilizados para modelar a superfície sustentadora.

Figura 3 – Modelo de anéis de vórtice utilizado na formulação do VLM para superfícies sustentadoras finas.



Fonte: Katz e Plotkin (2001).

Em cada painel, define-se um ponto de controle localizado tipicamente no quarto de corda, posição em que os efeitos da circulação são mais bem representados. A velocidade induzida

pela rede completa de anéis de vórtice é então avaliada nesses pontos de controle, e aplica-se a condição de não penetração do escoamento: a componente normal da velocidade relativa entre escoamento e superfície deve ser nula. Essa condição gera um sistema linear cujo resultado fornece os valores de circulação associados a cada anel.

3.3 MÉTODO NONLINEAR LIFTING-LINE (N-LLT)

A Teoria de Linha Sustentadora (*Lifting-Line Theory* — LLT), originalmente formulada por Prandtl, constitui um dos pilares da aerodinâmica de asas de grande alongamento. Nessa formulação, a asa é representada por uma linha elástica ao longo do semiplano médio, sobre a qual se distribui uma circulação $\Gamma(y)$. A circulação corresponde à intensidade rotacional do escoamento ao redor da asa, sendo proporcional à sustentação produzida pela seção e diretamente ligada ao teorema de Kutta–Joukowski. Cada elemento dessa circulação é modelado por um vórtice em forma de ferradura, cujas extremidades se estendem a jusante, induzindo uma velocidade descendente (w_i) no escoamento.

A Figura 4 apresenta, de forma esquemática, essa representação por vórtices ferradura, ilustrando a filamentação do vórtice ligado à asa e seus prolongamentos. A partir dessa construção, a LLT estabelece que o ângulo de ataque efetivo em cada estação da asa é reduzido pela componente de velocidade induzida:

$$\alpha_{eff}(y) = \alpha_{geom}(y) - \alpha_i(y)$$

onde $\alpha_i(y)$ é o ângulo induzido associado ao *downwash* resultante da distribuição de circulação. A relação entre $\Gamma(y)$ e $\alpha_i(y)$ é obtida por meio da integração da influência dos vórtices ao longo da envergadura, levando à formulação integral de Prandtl. Essa equação é comumente resolvida por discretização, seja por séries de Fourier, seja por métodos numéricos diretos.

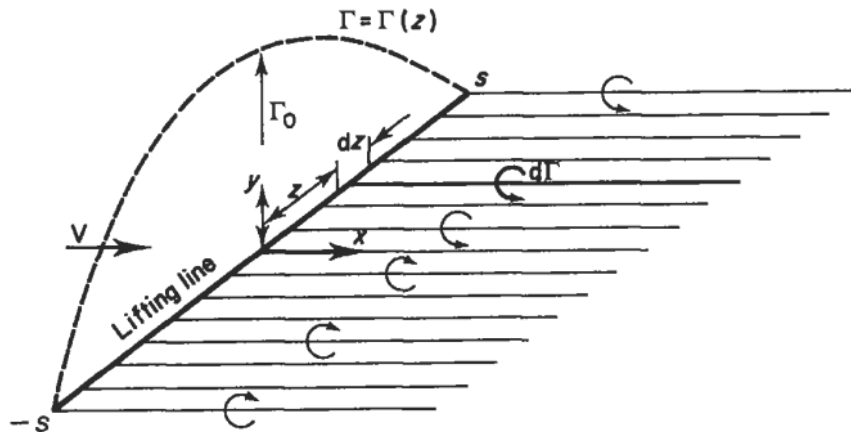
A LLT clássica assume que o coeficiente de sustentação seccional (C_ℓ) varia linearmente com o ângulo de ataque, além de considerar escoamento potencial de pequenas perturbações, operação em ângulos moderados afastados do estol e ausência de interações não lineares significativas ao longo da envergadura. Essas hipóteses tornam a LLT extremamente eficiente e amplamente empregada, mas também limitam sua aplicabilidade quando a asa opera próxima ao estol ou quando as curvas seccionais apresentam comportamento não linear relevante.

O método *Nonlinear Lifting-Line Theory* (N-LLT) estende a LLT clássica ao incorporar, no cálculo seccional, a polar completa de sustentação das seções 2D. Em vez de assumir a relação linear

$$C_\ell = a(\alpha - \alpha_{L0}),$$

o N-LLT utiliza uma relação não linear ($C_\ell = f(\alpha)$) obtida de dados experimentais, de simulações CFD ou de ferramentas específicas como o XFOIL. Aqui, o termo CFD (*Computational Fluid Dynamics*) refere-se a técnicas numéricas que resolvem as equações de Navier–Stokes para obter características aerodinâmicas com maior fidelidade. Já o XFOIL é um software

Figura 4 – Representação da distribuição de circulação e dos vórtices ferradura em uma asa utilizando a LLT.



Fonte: Houghton e Carpenter (2003).

amplamente utilizado para análise aerodinâmica 2D de aerofólios, capaz de fornecer curvas de sustentação, arrasto e momento com baixo custo computacional.

Com isso, a solução deixa de ser direta e passa a depender de um processo iterativo: assume-se uma distribuição inicial de circulação, calcula-se o *downwash* correspondente, obtém-se o ângulo de ataque efetivo α_{eff} , determina-se o coeficiente de sustentação seccional a partir da polar não linear e atualiza-se a circulação. O procedimento é repetido até convergência.

Essa abordagem permite capturar efeitos significativos como variações não lineares de C_l com o ângulo de ataque, comportamento próximo ao estol e a resultante redistribuição de incidência ao longo da envergadura, aumentando substancialmente a fidelidade em regimes onde a LLT linear deixa de ser válida.

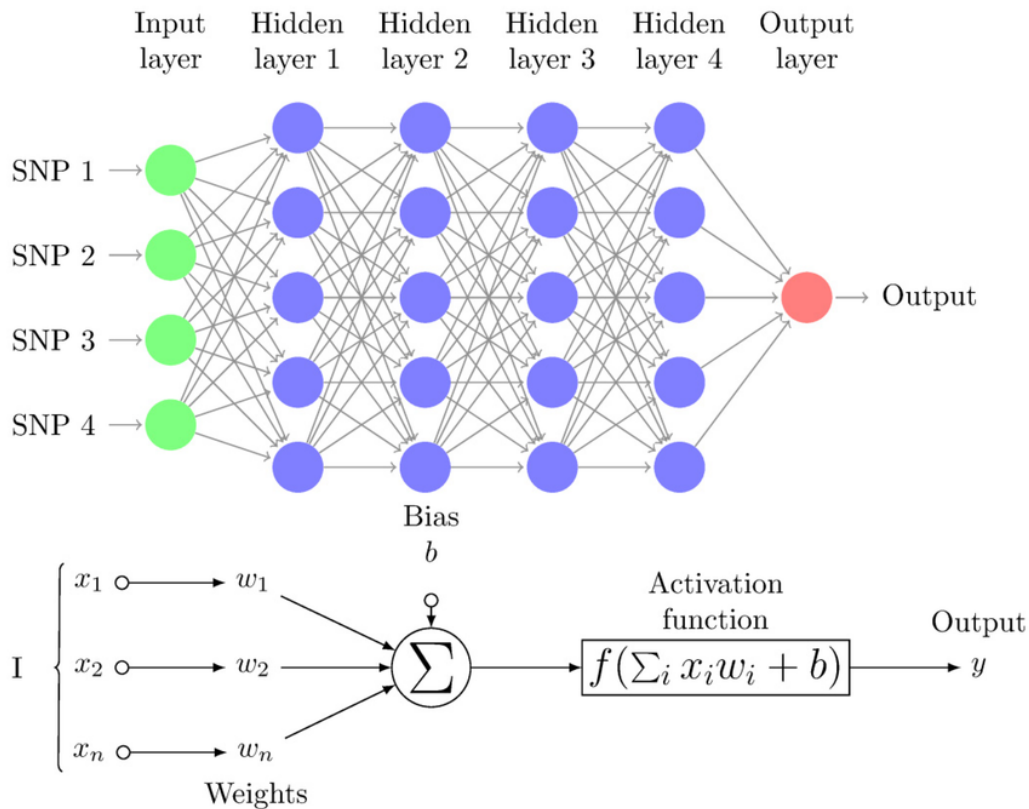
Apesar de mais custoso que a formulação clássica, o N-LLT permanece computacionalmente leve, tornando-se adequado para análises preliminares e para problemas envolvendo múltiplas avaliações aerodinâmicas, como otimizações multiobjetivo. No presente trabalho, o N-LLT é empregado para estimar C_L , $C_{L_{max}}$ e demais parâmetros aerodinâmicos de forma eficiente. Onde $C_{L_{max}}$ representa o valor máximo do coeficiente de sustentação da asa antes do início do estol, sendo um parâmetro fundamental para o dimensionamento e para a análise de desempenho em baixas velocidades (Karali *et al.*, 2021).

3.4 REDES NEURAIS DO TIPO MULTI-LAYER PERCEPTRON (MLP)

O *Multi-Layer Perceptron* (MLP) é um modelo de rede neural artificial composto por camadas totalmente conectadas, incluindo uma camada de entrada, uma ou mais camadas ocultas e uma camada de saída. Cada camada contém unidades de processamento denominadas neurônios, que recebem sinais numéricos, processam-nos e transmitem a informação para as camadas seguintes (Figura 5).

Cada neurônio recebe um vetor de entradas $\mathbf{x} = [x_1, x_2, \dots, x_n]$, associado a um conjunto de pesos treináveis $\mathbf{w} = [w_1, w_2, \dots, w_n]$. O neurônio calcula uma combinação linear desses valores,

Figura 5 – Estrutura típica de uma rede neural do tipo MLP mostrando suas camadas e neurônios, juntamente com a influência que cada neurônio exerce sobre os outros.



Fonte: Pérez-Enciso e Zingaretti (2019).

acrescida de um termo adicional denominado *bias*, b :

$$z = \sum_{i=1}^n x_i w_i + b.$$

Esse valor interno é então transformado por uma função de ativação não linear, como *ReLU*, sigmoide ou *tanh*:

$$y = f(z).$$

A presença dessas funções não lineares é fundamental, pois permite que a rede represente relações complexas entre as variáveis. Sem elas, mesmo múltiplas camadas resultariam em um modelo linear.

Os dados utilizados para treinar um MLP são normalmente divididos em dois subconjuntos: o **conjunto de treinamento**, empregado para ajustar os pesos da rede, e o **conjunto de teste**, composto por dados que a rede nunca vê durante o treinamento e que servem para avaliar sua capacidade de generalização.

Cada exemplo do conjunto de dados contém:

- um **vetor de entrada** (as variáveis do problema),
- um **rótulo** (*label*), que representa a **saída esperada**.

No contexto aerodinâmico, os rótulos podem corresponder a grandezas como C_L , C_D , $C_{L_{\max}}$ ou outras variáveis fornecidas pelo *solver*. O objetivo do treinamento é fazer com que a saída produzida pela rede $\hat{y}$ se aproxime do rótulo verdadeiro y .

A **função de erro** (ou função de custo), como o erro quadrático médio (MSE), é calculada por:

$$E = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2.$$

Esse erro quantifica a discrepância entre a saída prevista pela rede e a saída real.

Treinamento e retropropagação

O processo de treinamento consiste no ajuste sucessivo dos pesos e *biases* para minimizar o erro. Isso ocorre por meio do algoritmo de retropropagação (*backpropagation*), que calcula o gradiente do erro em relação aos pesos e os atualiza utilizando um otimizador, como o gradiente descendente.

A cada iteração, a rede:

1. recebe um conjunto de entradas e produz uma saída $\hat{y}$;
2. calcula o erro comparando $\hat{y}$ ao rótulo y ;
3. propaga esse erro de volta pelas camadas;
4. ajusta os pesos para reduzir erros futuros.

Esse procedimento é repetido por diversas épocas até atingir convergência ou até que o erro deixe de melhorar.

Overfitting e underfitting

Durante o treinamento, é importante monitorar o comportamento do erro no conjunto de validação:

- **Overfitting:** ocorre quando a rede aprende excessivamente os detalhes e ruídos do conjunto de treinamento, perdendo capacidade de generalização. O erro de treinamento diminui enquanto o erro de validação aumenta.
- **Underfitting:** ocorre quando o modelo não consegue aprender adequadamente a relação entre entrada e saída, apresentando erros elevados em ambos os conjuntos.

Curvas de aprendizado e métricas de desempenho auxiliam na identificação desses comportamentos.

MLP como modelo substituto

Quando treinado sobre bases de dados extensas e consistentes, o MLP pode reproduzir com alta fidelidade resultados obtidos por métodos numéricos mais custosos, como CFD ou métodos aerodinâmicos iterativos. Assim, o MLP atua como um *surrogate model*, permitindo milhares de avaliações em tempo muito reduzido — característica essencial para problemas de otimização aerodinâmica envolvendo buscas em espaços de projeto de alta dimensionalidade (Haykin, 2009).

3.5 ALGORITMO DUAL ANNEALING

O Dual Annealing é um método de otimização global inspirado no processo físico de recozimento (*annealing*), no qual um material é aquecido a altas temperaturas e posteriormente resfriado de forma controlada. Esse procedimento permite que a estrutura interna do material explore diferentes configurações energéticas, eventualmente alcançando um estado de mínima energia. Em sua formulação computacional, esse processo é traduzido em uma busca estocástica no espaço de soluções, onde a “temperatura” controla a probabilidade de aceitar soluções piores ao longo das iterações.

No contexto deste trabalho, o vetor de variáveis de projeto é representado por $\mathbf{x}$, o qual contém os parâmetros geométricos que caracterizam cada asa avaliada. Assim, o objetivo da otimização é encontrar o vetor $\mathbf{x}$ que minimize a função objetivo associada ao desempenho aerodinâmico.

Assim como no recozimento simulado tradicional, o Dual Annealing utiliza um mecanismo probabilístico baseado na estatística de Boltzmann para permitir, ocasionalmente, a aceitação de soluções cujo valor de função objetivo é pior que o atual. Seja Δf definido como a diferença entre o valor da função objetivo da nova solução e o da solução corrente:

$$\Delta f = f(\mathbf{x}_{\text{nova}}) - f(\mathbf{x}_{\text{atual}}). \quad (3)$$

Quando $\Delta f < 0$, a nova solução representa uma melhoria e é automaticamente aceita. Contudo, quando $\Delta f > 0$, a solução é pior e sua aceitação depende de uma probabilidade P , dada por:

$$P = \exp\left(-\frac{\Delta f}{T}\right), \quad (4)$$

onde T é a temperatura atual do sistema. Quanto maior a temperatura, maior a probabilidade de aceitar soluções piores, o que favorece a exploração global do espaço de busca. À medida que o algoritmo avança, a temperatura diminui gradualmente, reduzindo a probabilidade de aceitar soluções desfavoráveis e conduzindo o processo para uma fase de exploração mais refinada em torno de regiões promissoras.

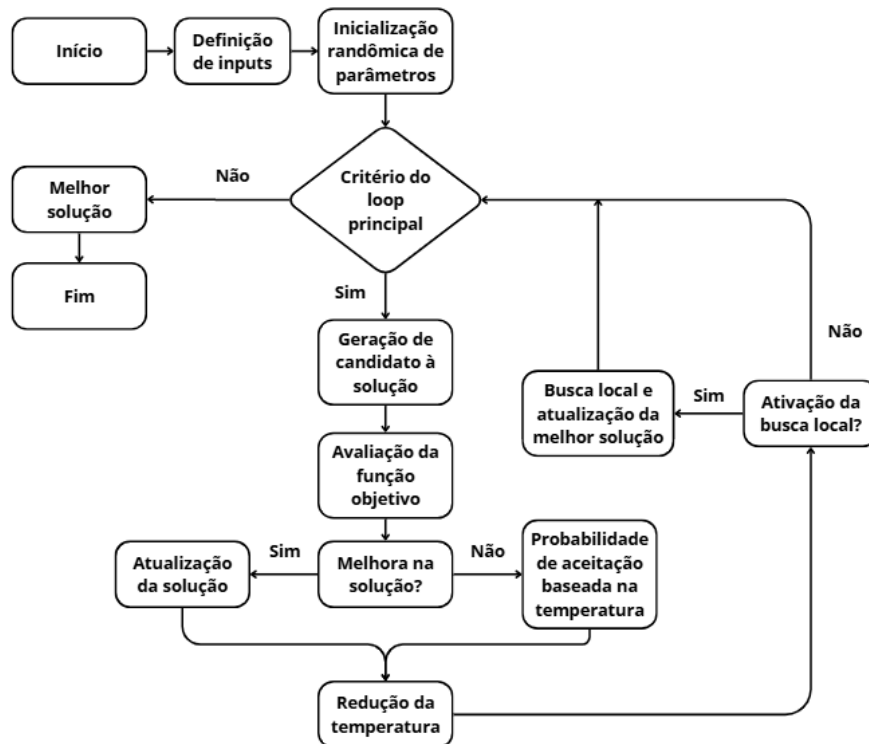
Além desse mecanismo fundamental, o Dual Annealing opera dentro de um laço principal cujo avanço é controlado por critérios específicos definidos na implementação do *SciPy*. Os critérios de parada incluem: (i) o número máximo de iterações globais (**maxiter**); (ii) o número máximo de avaliações da função objetivo (**maxfun**); (iii) a condição de reinicialização térmica

quando a temperatura atinge uma fração definida por `restart_temp_ratio`; e (iv) uma função de *callback* opcional, capaz de interromper o processo se retornar verdadeiro. Esses mecanismos asseguram que o processo de busca seja exploratório no início e cada vez mais direcionado à medida que avança, equilibrando eficiência e profundidade de busca.

Esse equilíbrio dinâmico entre exploração (altas temperaturas) e intensificação (baixas temperaturas) reduz significativamente a chance de o algoritmo ficar preso em mínimos locais, característica essencial em problemas multimodais, não suaves ou altamente não lineares. Além disso, o Dual Annealing combina esse mecanismo global com estágios opcionais de refinamento local, como o método L-BFGS-B, tornando-o especialmente robusto para otimizações aerodinâmicas complexas.

O método L-BFGS-B (*Limited-memory Broyden-Fletcher-Goldfarb-Shanno with Bounds*) é um algoritmo de otimização local quase-Newton que aproxima a matriz Hessiana de forma eficiente em termos de memória, sendo particularmente adequado para problemas de grande dimensão. Diferentemente das versões clássicas do BFGS, o L-BFGS-B permite a imposição explícita de limites inferiores e superiores nas variáveis, o que o torna apropriado para otimizações com restrições de caixa (*box constraints*). No contexto do Dual Annealing, ele é utilizado como um refinador local aplicado após fases de exploração global, acelerando a convergência em regiões promissoras do espaço de busca. O fluxograma geral do algoritmo é apresentado na Figura 6.

Figura 6 – Fluxograma do algoritmo Dual Annealing utilizado para otimizações globais de objetivo único.



Fonte: Elaborado pelo autor.

3.6 ALGORITMO GENÉTICO NSGA-II

O NSGA-II (*Non-dominated Sorting Genetic Algorithm II*) é um dos algoritmos evolutivos mais utilizados em otimização multiobjetivo, sobretudo em problemas com objetivos conflitantes e superfícies de resposta altamente não lineares. Nesse método, cada solução é representada por um indivíduo, formado por um vetor contendo todos os parâmetros de projeto. O conjunto desses indivíduos compõe a população, que evolui ao longo das gerações por meio de operadores inspirados na seleção natural: seleção, cruzamento e mutação.

No processo de seleção, determina-se quais indivíduos têm maior probabilidade de gerar descendentes. O NSGA-II organiza a população em frentes de Pareto por meio do *non-dominated sorting*, um procedimento que classifica as soluções conforme o grau de dominância. Uma solução é dita *não dominada* quando nenhuma outra solução da população é simultaneamente melhor em todos os objetivos. Assim, o algoritmo agrupa primeiro todas as soluções não dominadas na primeira frente de Pareto, depois identifica aquelas que seriam não dominadas se a primeira frente fosse removida, formando a segunda frente, e assim por diante. As soluções da primeira frente recebem prioridade por representarem os melhores compromissos entre os objetivos.

Quando há empate entre indivíduos da mesma frente, utiliza-se a métrica de *crowding distance*, que quantifica o afastamento de cada solução em relação às suas vizinhas no espaço objetivo, favorecendo a manutenção da diversidade. A seleção é normalmente realizada por torneio binário: dois indivíduos são comparados, e vence aquele pertencente à melhor frente de Pareto; persistindo o empate, vence o que possui maior *crowding distance*. Dessa forma, o processo equilibra qualidade e diversidade ao longo das gerações.

O cruzamento (*crossover*) combina características de dois indivíduos selecionados, produzindo descendentes que herdam propriedades de ambos os pais. Esse mecanismo permite explorar novas regiões do espaço de projeto, preservando, ao mesmo tempo, atributos relevantes de soluções já bem avaliadas. Essa etapa é essencial para acelerar a busca em direção à fronteira de Pareto.

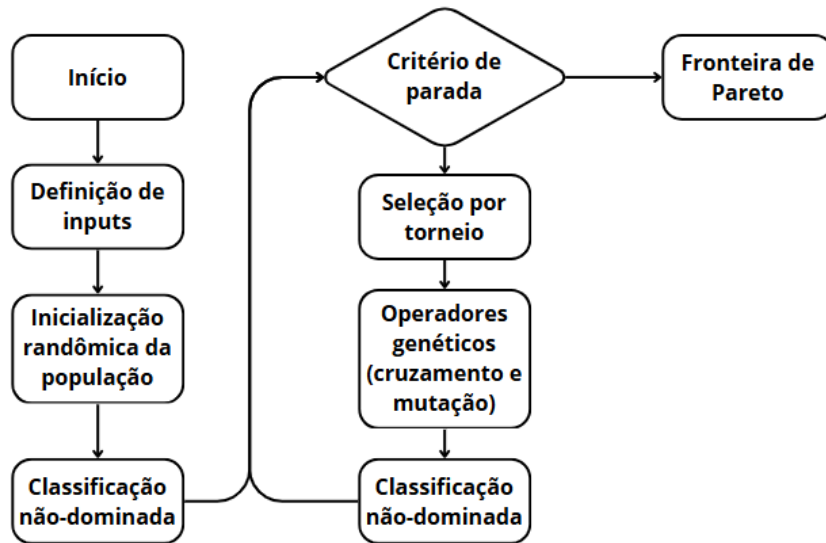
A mutação introduz pequenas perturbações aleatórias nos indivíduos, promovendo variabilidade genética e evitando a convergência prematura para regiões subótimas. Com isso, o algoritmo mantém capacidade de explorar o espaço de soluções e escapar de mínimos locais, preservando a diversidade populacional. O fluxograma geral do processo evolutivo pode ser observado na Figura 7.

A combinação de seleção baseada em frentes de Pareto, preservação de diversidade por *crowding distance*, operadores de cruzamento eficientes e mutação controlada permite ao NSGA-II aproximar de forma robusta e estável a fronteira de Pareto verdadeira. Por esses motivos, o método tornou-se amplamente adotado em problemas de otimização aerodinâmica, desempenho estrutural e diversas outras áreas da engenharia (Deb *et al.*, 2002; Deb, 2001).

$$f(x) = -C_{Lmax}$$

$$f(x) = C_{Mac}$$

Figura 7 – Fluxograma do algoritmo NSGA-II utilizado nas otimizações para encontrar as fronteira de Pareto.



Fonte: Elaborado pelo autor.

$$f(x) = - \left(\frac{L}{D} \right)_{max}$$

3.7 REVISÃO DE TRABALHOS SEMELHANTES

O estudo do comportamento aerodinâmico de asas e superfícies sustentadoras tem sido amplamente investigado nas últimas décadas, especialmente com o avanço de métodos numéricos de baixo custo computacional, como o *Vortex Lattice Method* (VLM) e teorias de linha de sustentação lineares e não-lineares (*Nonlinear Lifting-Line Theory*, N-LLT). Esses métodos são empregados em uma variedade de aplicações, desde a estimativa de cargas aerodinâmicas até a otimização de geometrias.

Trabalhos baseados no VLM são recorrentes devido ao bom equilíbrio entre fidelidade e custo computacional. Por exemplo, o programa AVL, desenvolvido por Mark Drela, usa um modelo de rede de vórtices estendido para superfícies sustentadoras e permite análise de estados de voo não lineares (Drela; Youngren, 2024). Além disso, Katz e Plotkin (2001) oferecem uma base teórica sólida em seu livro clássico sobre aerodinâmica de baixa velocidade (Katz; Plotkin, 2001). Na tese de mestrado de Tatar (2024), foi utilizado o VLM para calcular coeficientes aerodinâmicos em diferentes configurações de formação de UAVs (veículos aéreos não tripulados), que por sua vez foram utilizados para o treino de redes neurais artificiais (ANNs) (Tatar, 2024).

Por outro lado, métodos não lineares como a N-LLT têm ganhado destaque por incorporar efeitos tridimensionais mais realistas sem exigir o custo de simulações CFD completas (Karali *et al.*, 2021). Phillips e Snyder (2000) propuseram uma adaptação moderna da teoria de Prandtl usando elementos de vórtice tridimensional para resolver a linha de sustentação com curva de sustentação não linear. (Phillips; Snyder, 2000) Modelos mais recentes, como os de

Wickenheiser A. e Garcia (2011), generalizam essa abordagem para asas reconfiguráveis com curvatura arbitrária. (Wickenheiser a. Garcia, 2011)

Além das ferramentas aerodinâmicas, a modelagem por ANNs tem sido usada para reduzir o custo computacional em processos de otimização. Embora estudos específicos com MLPs para respostas aerodinâmicas ainda sejam emergentes, a literatura recente indica um crescimento no uso de redes neurais como modelos substitutos para simular respostas aerodinâmicas complexas com menor custo (Bàrcenas *et al.*, 2023; Suárez *et al.*, 2024).

Em otimizações multiobjetivo, algoritmos como o NSGA-II são valorizados por permitir a obtenção de múltiplas soluções de compromisso. O NSGA-II foi introduzido por Deb *et al.* (2002) no contexto da otimização multidisciplinar, o que motivou sua aplicação subsequente em problemas aeronáuticos. (Deb *et al.*, 2002)

No âmbito de otimização global, métodos como o *Dual Annealing* têm se destacado pela robustez na exploração de espaços de projeto complexos e por evitar mínimos locais, especialmente quando o modelo a ser otimizado não tem derivadas acessíveis. Isto pode ser visto no trabalho de Wang e Damodaran (2001) onde esse algoritmo foi utilizado com simulações CFD a fim de obter formas de asas otimizadas (Wang; Damodaran, 2001).

Assim, diversos trabalhos combinam métodos aerodinâmicos de baixa ordem com técnicas de otimização e aprendizado de máquina. Por exemplo, em estudos de projeto inicial de aeronaves, a combinação de VLM com modelos substitutos e algoritmos de otimização mostrou-se eficaz para reduzir significativamente o custo computacional sem sacrificar tanto a precisão.

Este trabalho busca preencher uma lacuna integrando VLM, N-LLT, MLPs e algoritmos de otimização (como NSGA-II e Dual Annealing) em um fluxo de processamento unificado, aplicando essas ferramentas para comparar diretamente métodos aerodinâmicos e estudar a influência geométrica nas métricas de desempenho e estabilidade.

4 METODOLOGIA

4.1 MÉTODO VLM

O OpenVSP é um software de geometria paramétrica desenvolvido pela NASA, capaz de gerar modelos tridimensionais completos de aeronaves e realizar análises aerodinâmicas preliminares por meio de ferramentas integradas, como o método VLM.

Inicialmente, desenvolveu-se um código em Python utilizando a API do OpenVSP, com o objetivo de integrar ambos os sistemas, permitindo a simulação de várias configurações de asa através do método VLM fornecido pelo *software*. Esses códigos foram disponibilizados em repositório aberto (disponível em <https://zenodo.org/records/16783676>) (Carrari; Silva, 2025).

4.1.1 Definição das Variáveis

Primeiramente, foi necessário definir quais variáveis seriam utilizadas para descrever adequadamente a asa e servir de entrada para o modelo subsequente de rede neural artificial. O formato e o tamanho da asa podem ser descritos por, no mínimo, três parâmetros independentes, que são combinações dos seguintes: alongamento (AR), envergadura (b), área (S), razão de afilamento (λ), corda média aerodinâmica ($\bar{c}$), corda na raiz (c_r) e corda na ponta (c_t). Cujas as relações podem ser dadas por:

$$\lambda = \frac{c_t}{c_r} \quad (1)$$

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

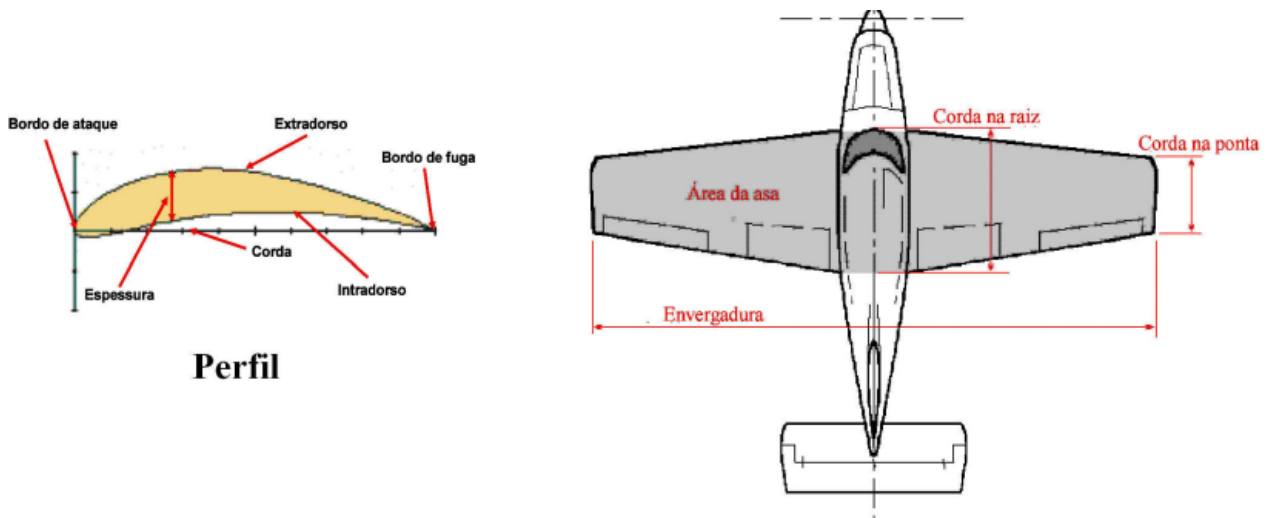
$$\bar{c} = \frac{2}{3} c_r \frac{1 + \lambda + \lambda^2}{1 + \lambda} \quad (3)$$

Os parâmetros geométricos utilizados para descrever asas fixas podem ser visualizados na Figura 8.

Dentre esses, foram selecionados o alongamento, a razão de afilamento e a corda média aerodinâmica. O alongamento e a razão de afilamento são medidas adimensionais que podem incluir e representar uma ampla gama de configurações de asa com pouca variação desses parâmetros. Já a corda média aerodinâmica pode ser utilizada para descrever a escala da asa, no entanto, não é necessário usá-la como entrada na rede, pois os resultados obtidos com diferentes cordas médias só diferem em escala, e esse efeito já é contemplado através da variação do número de Reynolds (Re), que também é um dos parâmetros de entrada da rede neural implementada neste projeto.

O número de Reynolds (Re) é um número adimensional que relaciona as forças inerciais às forças viscosas em um escoamento. Ele indica o regime do escoamento: laminar, transicional ou turbulento, e é fundamental para caracterizar o comportamento aerodinâmico de perfis e asas.

Figura 8 – Diagrama esquemático da geometria geral de uma asa fixa com sua seção transversal (esquerda) e vista de cima (direita).



Fonte: Rodrigues (2009).

No caso, V representa a velocidade do escoamento, ρ a densidade do fluido, μ a viscosidade do fluido e L o comprimento característico analisado.

$$Re = \frac{\rho V L}{\mu} \quad (4)$$

Além disso, as asas utilizadas no setor aeronáutico possuem outras características que não alteram significativamente seu formato em planta, mas influenciam o comportamento do escoamento ao seu redor. São esses: torção aerodinâmica, torção geométrica (θ), diedro (Γ), enflechamento (Λ) e, finalmente, o perfil aerodinâmico ou aerofólio, que será discutido posteriormente. Para simplificar a abordagem adotada neste projeto, são consideradas apenas asas sem torção aerodinâmica, ou seja, o mesmo aerofólio é utilizado ao longo de toda a asa.

Após a definição das variáveis, realizou-se uma pesquisa para estabelecer seus valores mínimos e máximos a serem utilizados no código, de modo a cobrir diferentes configurações de aeronaves. Para isso, foram analisados exemplos reais que representam extremos desses parâmetros. Por exemplo, planadores como o **Prue 215**, projetado por Irving Prue, têm alto alongamento (≈ 20), característica típica de aeronaves de alta eficiência aerodinâmica. Em contraste, aeronaves leves de aviação geral como o **Piper PA-28 Cherokee** possuem asas de corda quase constante e razão de alongamento mais baixa, servindo como limite inferior aproximado para esse parâmetro.

Em termos de razão de afilamento, aeronaves de combate como o **Lockheed Martin F-16 Fighting Falcon** utilizam asas muito afiladas, com razão de afilamento baixa ($\approx 0,2$), típicas de projetos de alta velocidade, enquanto aeronaves com asas retangulares como o **Cessna 150** têm razão de afilamento próxima de 1,0.

No que diz respeito ao enflechamento das asas, aeronaves militares como o **Sukhoi Su-15** apresentam altos ângulos de enflechamento para trás ($\approx 35^\circ$) para desempenho em altas velocidades, enquanto exemplos de aeronaves com enflechamento para frente (negativo), como

o **Grumman X-29**, alcançam valores como $\approx -33^\circ$, apesar de não serem comuns. Por esse motivo, foram estabelecidos limites de enfilechamento entre -25° e 35° no código.

Por fim, quanto ao ângulo de diedro, aeronaves utilitárias como o **Cessna 208 Caravan** apresentam leve *anhedral* (ângulos negativos em asas altas, usados para reduzir a estabilidade excessiva causada pelo efeito pendular da asa alta) e podem chegar próximo de -5° , enquanto aeronaves como o **McDonnell Douglas F-4 Phantom II** têm seções de asa com diedral positivo médio em torno de 5° para melhorar a estabilidade lateral.

Os valores definidos para os intervalos desses parâmetros estão detalhados na Tabela 1.

Tabela 1 – Parâmetros de Design.

Variável	Inferior	Superior
AR [-]	5	20
λ [-]	0.2	1
Λ [$^\circ$]	-25	35
Γ [$^\circ$]	-5	5
θ [$^\circ$]	-5	5
Re [-]	5×10^5	1×10^6

Fonte: Elaborado pelo autor.

A corda média aerodinâmica ($\bar{c}$) foi fixada em 7 metros, um valor médio encontrado em aeronaves, embora seu valor não interfira no treinamento e nos resultados da rede. As simulações foram realizadas com o ângulo de ataque variando entre -10° e 15° , com 10 pontos, e o número de Mach foi mantido fixo em zero, ignorando assim efeitos de compressibilidade, desprezíveis em regime de voo baixo subsônico. O número de Mach é definido como a razão entre a velocidade do escoamento e a velocidade do som no fluido:

$$M = \frac{V}{V_{\text{som}}} \quad (5)$$

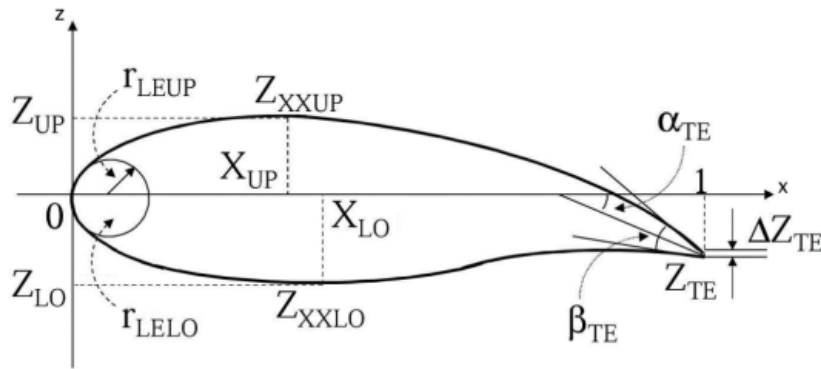
Conceitualmente, M^2 está relacionado à razão entre forças inerciais e forças elásticas no fluido: quando o fluido é comprimido e sua densidade aumenta, parte da energia cinética do escoamento é armazenada como energia potencial elástica.

A última variável restante e mais desafiadora de ser representada como entrada é o perfil aerodinâmico da asa. Existem diversos perfis padrão listados, além de perfis únicos e modificados por meio de otimizações e testes. Portanto, há uma dificuldade maior em abranger uma ampla variedade de perfis e, ainda mais, em como representá-los como parâmetros de entrada. Para lidar com isso, foi utilizado o método PARSEC, que consiste em 12 parâmetros para descrever um aerofólio (Hájek, 2007). Estes parâmetros podem ser vistos na Figura 9 e estão enumerados a seguir, sendo eles:

1. r_{LEUP} : raio de curvatura do bordo de ataque no extradorso;
2. r_{LELO} : raio de curvatura do bordo de ataque no intradorso;
3. X_{UP} : posição da distância vertical máxima do extradorso em relação a uma linha horizontal perpendicular ao bordo de ataque (referência horizontal);

4. Z_{UP} : distância vertical máxima do extradorso em relação à referência horizontal;
5. Z_{XXUP} : curvatura do extradorso em X_{UP} ;
6. X_{LO} : posição da distância vertical máxima do intradorso em relação à referência horizontal;
7. Z_{LO} : distância vertical máxima do intradorso em relação à referência horizontal;
8. Z_{XXLO} : curvatura do intradorso em X_{LO} ;
9. ΔZ_{TE} : espessura do bordo de fuga;
10. Z_{TE} : distância vertical do bordo de fuga em relação à referência horizontal;
11. α_{TE} : direção do bordo de fuga;
12. β_{TE} : ângulo de abertura do bordo de fuga.

Figura 9 – 12 parâmetros PARSEC utilizados para descrever um perfil aerodinâmico.



Fonte: Hájek (2007).

Para obter dados sobre perfis aerodinâmicos foi utilizado o banco gerado por Silva e Catalano (2019). Com isso foram obtidos 1499 perfis parametrizados com PARSEC que serão posteriormente utilizados para gerar geometrias de asas. Os parâmetros para a entrada da rede neural foram nomeados como na Tabela 2 onde são mostrados os valores mínimos e máximo em cada parâmetro dos 1499 aerofólios, bem como suas médias e desvios padrões (DP) a fim de avaliar a variedade de aerofólios contemplados. Os parâmetros que se referem a comprimentos foram adimensionalizados pela corda do aerofólio.

Com isso, a rede neural pode ser alimentada com 18 entradas utilizadas para descrever uma asa sem torção aerodinâmica e seu escoamento. As saídas geradas pelo OpenVSP e de interesse nesta pesquisa são os parâmetros aerodinâmicos que aproximam as curvas de coeficiente de sustentação (C_L), coeficiente de arrasto (C_D) e coeficiente de momento de arfagem em relação ao bordo de ataque da corda na raiz (C_M) que podem ser vistas nas equações a seguir.

$$C_L = C_{L0} + C_{L\alpha}\alpha \quad (6)$$

Tabela 2 – Parâmetros do aerofólio com valores mínimos, máximos, médias e desvios padrões.

Parâmetro	Mínimo	Máximo	Média	DP
$P_1 = r_{LEUP}/c$	0.0006	0.1565	0.02083	0.01566
$P_2 = r_{LELO}/c$	0.0005	0.2374	0.00949	0.01077
$P_3 = X_{UP}/c$	0.1702	0.5691	0.34555	0.05489
$P_4 = Z_{UP}/c$	0.0199	0.2318	0.09156	0.03159
$P_5 = Z_{XX_{UP}} \times c$	-2.5531	0.0161	-0.77033	0.39906
$P_6 = X_{LO}/c$	0.0533	3.3072	0.31845	0.12592
$P_7 = Z_{LO}/c$	-0.1935	1.0095	-0.03183	0.03995
$P_8 = Z_{XX_{LO}} \times c$	-1.8062	2.4212	0.32104	0.43654
$P_9 = \Delta Z_{TE}/c$	0.0	0.0419	0.00198	0.00309
$P_{10} = Z_{TE}/c$	0.0	0.0158	0.00092	0.00165
$P_{11} = \alpha_{TE} [^\circ]$	0.0	33.2179	3.92806	4.30490
$P_{12} = \beta_{TE} [^\circ]$	0.0	69.192	11.02431	6.86612

Fonte: Elaborado pelo autor.

$$C_D = C_{D0} + K_1 C_L + K_2 C_L^2 \quad (7)$$

$$C_M = C_{M0} + C_{M\alpha} \alpha \quad (8)$$

Assim, totalizam-se sete parâmetros de saída que deverão ser previstos pela rede neural: C_{L0} , $C_{L\alpha}$, C_{D0} , K_1 , K_2 , C_{M0} e $C_{M\alpha}$, descritos a seguir:

- C_{L0} – Coeficiente de sustentação quando o ângulo de ataque é zero, representando a sustentação inicial gerada exclusivamente pela forma e pelo perfil da asa, independentemente da incidência;
- $C_{L\alpha}$ – Taxa de variação do coeficiente de sustentação em relação ao ângulo de ataque (α), ou seja, a sensibilidade da sustentação a pequenas variações de α ;
- C_{D0} – Coeficiente de arrasto parasita, que representa o arrasto presente mesmo quando não há sustentação ($C_L = 0$), geralmente associado à forma e ao atrito superficial;
- K_1 – Coeficiente linear da variação do arrasto induzido em função de C_L , capturando parte da influência da sustentação sobre o arrasto total;
- K_2 – Coeficiente quadrático que representa o efeito do aumento não linear do arrasto induzido com o aumento de C_L ;
- C_{M0} – Coeficiente de momento de arfagem em relação ao bordo de ataque da corda na raiz, medido quando o ângulo de ataque é zero. Representa o momento inicial gerado apenas pela geometria e distribuição de sustentação da asa, sem influência da incidência;
- $C_{M\alpha}$ – Taxa de variação do momento de arfagem em relação ao ângulo de ataque, indicando como o equilíbrio de momentos se altera com mudanças em α .

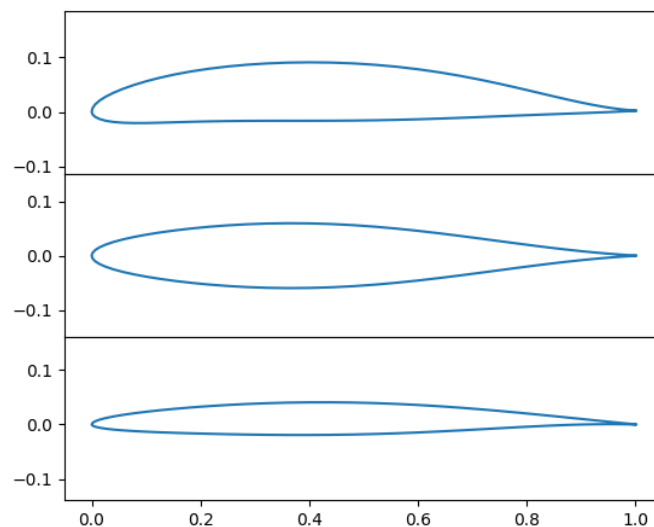
4.1.2 Malha e Convergência

Dado que o procedimento para gerar o banco de dados envolve muitas simulações, a precisão e o tempo de simulação são fatores de extrema importância. É necessário encontrar um equilíbrio que permita reduzir o tempo de simulação sem comprometer significativamente a precisão.

A *malha*, neste contexto, corresponde ao conjunto de discretizações que dividem a asa em elementos menores sobre os quais o método aerodinâmico realiza os cálculos. Em métodos como o VLM, a malha define como a superfície é fragmentada em painéis e linhas de vórtice, influenciando diretamente tanto a precisão quanto o custo computacional da análise. A malha utilizada na análise por VLM pode ser caracterizada por três parâmetros essenciais: $SECT_{TESSu}$, que representa o número de seções ao longo da semi-envergadura; $OUT_{CLUSTER}$, que determina o grau de agrupamento dessas seções ao longo da envergadura (com valores menores que 1 indicando maior agrupamento nas pontas e valores maiores que 1 indicando maior agrupamento na raiz); e $N_{AIRFOIL}$, que indica a quantidade de pontos usados para descrever o aerofólio.

Para realizar as análises de convergência utilizou-se do caso mais extremo para as simulações onde $AR = 5$ e $\lambda = 0.2$ que representam os casos mais críticos em simulações com o algoritmo VLM. Asas com baixo alongamento e baixas razões de afilamento representam situações extremas devido à maior variação do comportamento do escoamento em um menor comprimento, especialmente na formação e influência dos vórtices de ponta de asa no restante da asa. Os outros parâmetros foram fixados como $\Lambda = 0^\circ$, $\Gamma = 5^\circ$, $\theta = -5^\circ$, $Re = 7.5 \times 10^5$ e 3 perfis escolhidos aleatoriamente que podem ser vistos na Figura 10.

Figura 10 – Perfis aerodinâmicos normalizados pela corda que foram utilizados nas análises de convergência.



Fonte: Elaborado pelo autor.

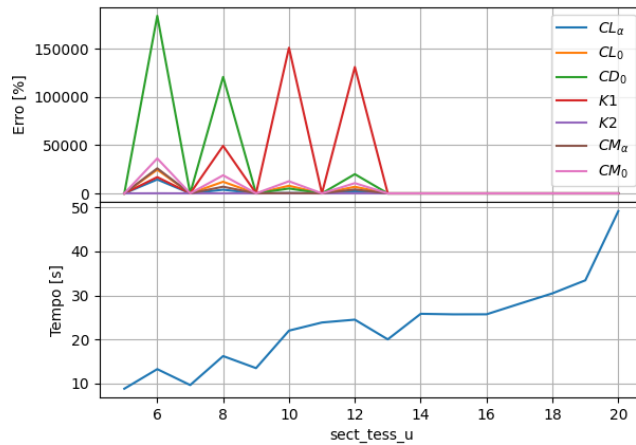
Diversas simulações foram realizadas no OpenVSP, onde os valores obtidos para precisão e tempo são uma média dos três casos de aerofólio. Para o cálculo do erro, os valores de referência

utilizados foram obtidos com uma simulação com $SECT_{TESSu} = 40$, $OUT_{CLUSTER} = 0.1$ e $N_{AIRFOIL} = 40$, representando uma simulação mais precisa e custosa.

Visando obter uma simulação com um tempo médio de no máximo 15 segundos, de modo a construir um extenso banco de dados no tempo disponível, foram realizadas diversas simulações para determinar os valores ideais de $SECT_{TESSu}$, $OUT_{CLUSTER}$ e $N_{AIRFOIL}$ para uma coleta eficiente de dados. A primeira série de simulações analisou o comportamento e a influência do parâmetro $SECT_{TESSu}$; para isso, fixou-se $OUT_{CLUSTER} = 1$ e $N_{AIRFOIL} = 40$.

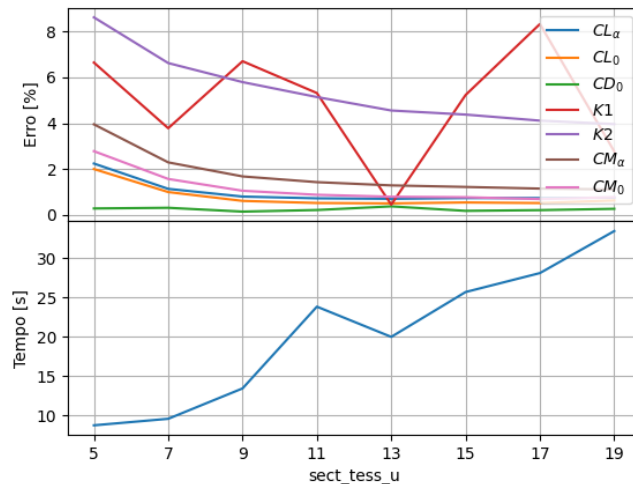
Observou-se que, para valores menores que 13 para $SECT_{TESSu}$, apenas valores ímpares convergiram; a partir de 13, todos os valores convergiram, conforme mostrado na Figura 11. Com isso priorizou-se a análise somente dos valores ímpares que podem ser vistos na Figura 12.

Figura 11 – Efeito da escolha de valores pares para o número de seções da semi-envergadura ($SECT_{TESSu}$), com $OUT_{CLUSTER} = 1$ e $N_{AIRFOIL} = 40$, na qualidade dos resultados e no tempo de simulação.



Fonte: Elaborado pelo autor.

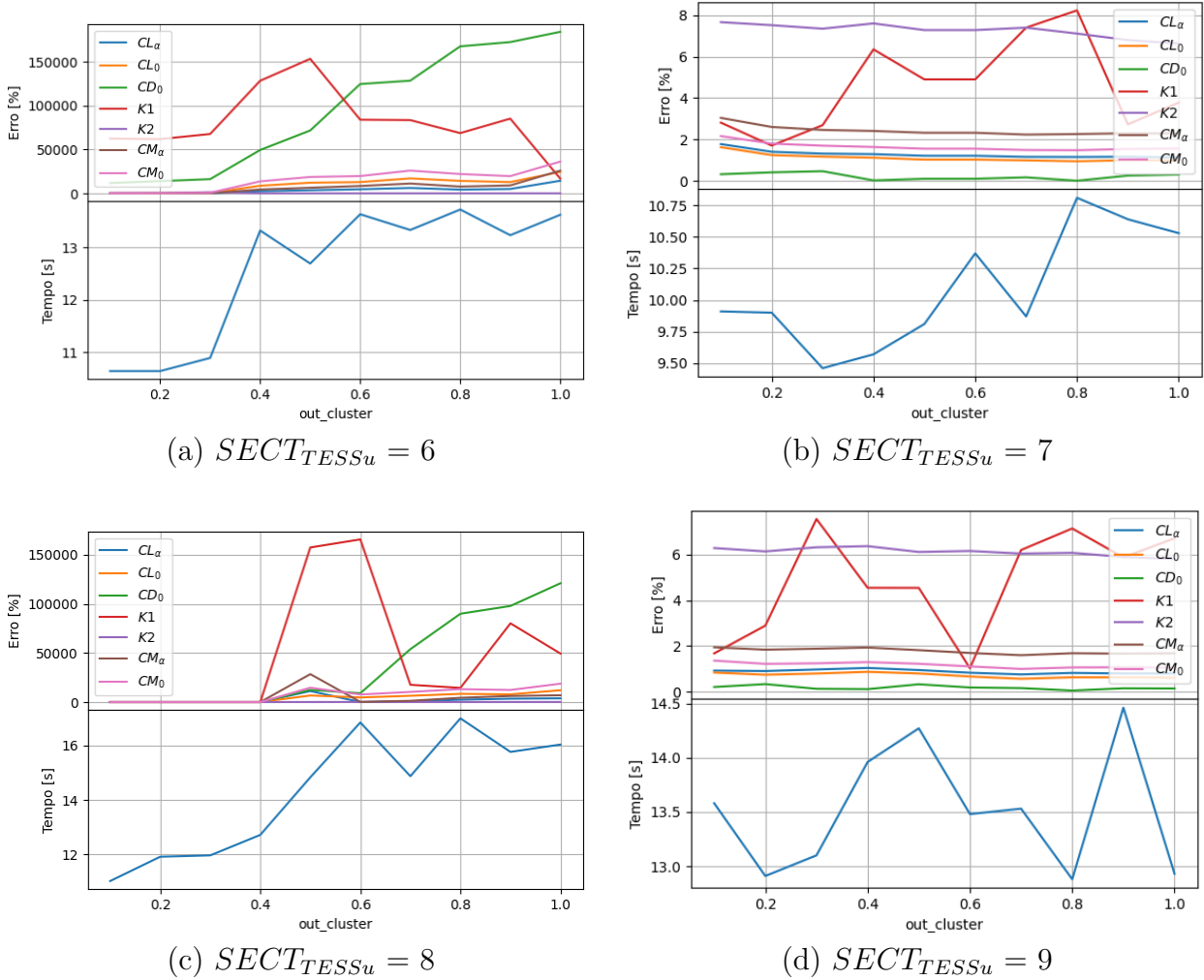
Figura 12 – Efeito da escolha de valores ímpares para o número de seções da semi-envergadura ($SECT_{TESSu}$), com $OUT_{CLUSTER} = 1$ e $N_{AIRFOIL} = 40$, na qualidade dos resultados e no tempo de simulação.



Fonte: Elaborado pelo autor.

Observou-se que no intervalo de $SECT_{TESSu}$ de 6 a 9 o tempo de simulação atende aos requisitos, com pouca variação no erro das saídas, embora $OUT_{CLUSTER} = 1$ não seja o ideal para precisão no cálculo dos vórtices de ponta de asa. Em seguida, foi realizada uma análise conjunta para entender como os parâmetros $SECT_{TESSu}$ e $OUT_{CLUSTER}$ influenciam na convergência. Para isso, foram realizadas simulações variando $OUT_{CLUSTER}$, com $N_{AIRFOIL} = 40$ e valores de $SECT_{TESSu}$ fixados em 6, 7, 8 e 9, conforme mostrado na Figura 13.

Figura 13 – Influência da escolha de $OUT_{CLUSTER}$ na precisão dos resultados e no tempo de simulação.



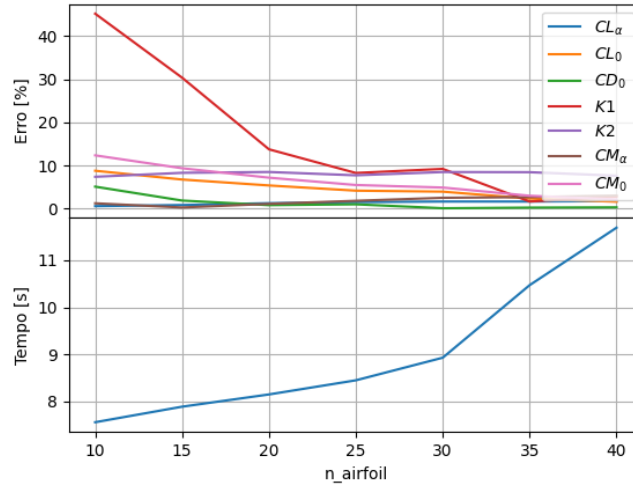
Fonte: Elaborado pelo autor.

Com isso, foi possível observar que o parâmetro $OUT_{CLUSTER}$ foi capaz de convergir os valores pares de $SECT_{TESSu}$ conforme se agrupam as seções nas pontas da asa e que sua variação não influencia significativamente o tempo de simulação, exceto nas simulações que não convergiram. Portanto, foi definido que as simulações com $OUT_{CLUSTER} = 0.1$ e $SECT_{TESSu} = 7$ representam a melhor relação tempo-precisão, com erros variando em uma faixa de 0% a 4%, exceto para as saídas K_1 , que teve um erro maior, mas que na maioria das vezes é negligenciado para estimativas da polar de arrasto quadráticas, e K_2 .

Por fim, foi analisada a influência do número de pontos do aerofólio nas simulações, como pode ser visto na Figura 14. O parâmetro $N_{AIRFOIL}$ mostrou ter relativa influência no tempo

de simulação. O valor de 30 para o número de pontos mostrou ter uma redução no tempo sem comprometer significativamente a precisão dos resultados.

Figura 14 – Influência da escolha de $N_{AIRFOIL}$ fixando $OUT_{CLUSTER} = 0.1$ e $SECT_{TESSu} = 7$ sobre a precisão e tempo de simulação.



Fonte: Elaborado pelo autor.

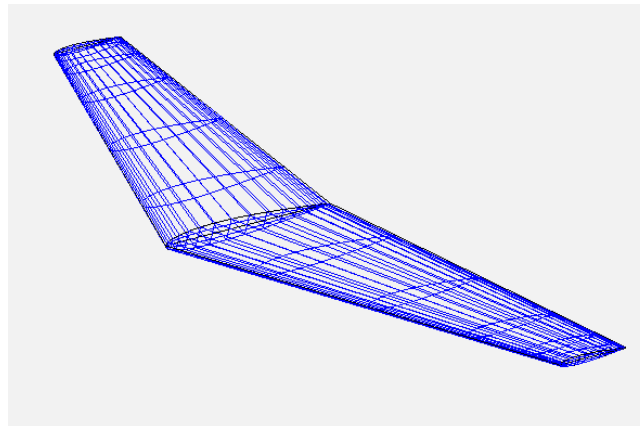
Ao fim desta análise de convergência, a malha menos refinada que garante independência dos resultados em relação a maiores refinamentos foi definida para ser usada em todas as simulações utilizadas para compilar o banco de dados que foi utilizado para treinar o modelo de rede neural. Os parâmetros finais podem ser vistos na Tabela 3. O tempo médio obtido para cada simulação é de aproximadamente 9 segundos. Na Figura 15 abaixo é possível ver um exemplo da malha definida em uma asa genérica.

Tabela 3 – Valores utilizados para descrever a malha nas simulações.

$OUT_{CLUSTER}$	$SECT_{TESSu}$	$N_{AIRFOIL}$
0.1	7	30

Fonte: Elaborado pelo autor.

Figura 15 – Vista em perspectiva da malha final utilizada nas simulações para uma configuração genérica de asa.



Fonte: Elaborado pelo autor.

4.1.3 Banco de Dados

Sabendo o tempo necessário para completar uma simulação, a quantidade de simulações para gerar o banco de dados pôde ser definida com base no tempo disponível. Assim, o número de simulações foi determinado como uma combinação de todos os valores das variáveis de entrada, que foram discretizados linearmente conforme a quantidade indicada na Tabela 4, gerando um banco de dados com 2.593.728 simulações. Uma vez que cada simulação leva aproximadamente 9 segundos, o tempo total de simulação é de aproximadamente 270 dias, porém essas simulações foram executadas em um *hardware* com 30 núcleos em paralelo, reduzindo o tempo total de simulação para aproximadamente 9 dias.

Tabela 4 – Combinações de Variáveis

Variável	Quantidade
AR	4
λ	4
Λ	4
Γ	3
θ	3
Re	3
Perfis	1499
Combinações	2.593.728

Fonte: Elaborado pelo autor.

4.1.4 Arquitetura e Treinamento da Rede Neural

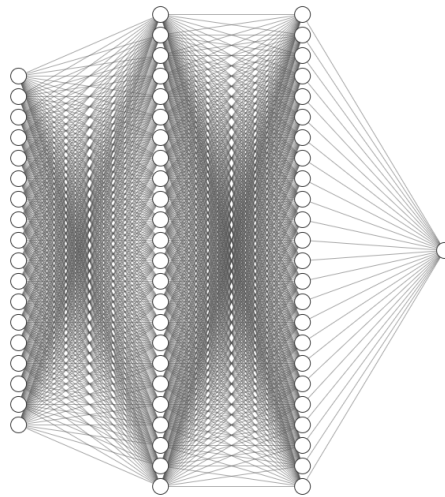
De posse dos dados, o próximo passo foi definir o modelo de rede neural a ser adotado, assim como a arquitetura final da rede, com o objetivo de obter um modelo preditivo com o menor erro possível. Esse objetivo pode ser alcançado utilizando-se bibliotecas de algoritmos de otimização, como o SciPy. Dada a complexidade do modelo e a grande quantidade de não linearidades esperadas, o uso de um algoritmo de otimização global não baseado em gradiente se faz necessário. Por esse motivo e pela sua robustez, o algoritmo `dual_annealing` foi escolhido para encontrar a rede que apresenta o menor erro. O algoritmo foi configurado utilizando uma temperatura inicial de 5230, razão de reinício de temperatura de 2×10^{-5} e número máximo de iterações de 1000. Os dados foram previamente normalizados entre 0 e 1 utilizando `MinMaxScaler` da biblioteca `scikit-learn` utilizando separadamente todas as entradas e todas as saídas e foram divididos em dados de treino e teste em uma proporção de 80% e 20%, respectivamente.

A rede neural foi treinada utilizando técnicas de regressão não linear como *Multi-Layer Perceptron* (MLP) com a biblioteca `scikit-learn` onde a função de ativação utilizada foi a ReLU por se mostrar melhor para o problema em questão comparado às outras opções. Portanto, a função objetivo a ser minimizada com o `dual_annealing` foi a raiz do erro médio quadrático (RMSE) associada à predição da rede treinada. As variáveis de projeto nesta otimização de hiperparâmetros foram os números de neurônios nas camadas intermediárias. Ao final, obteve-se a arquitetura da rede neural que minimiza o erro de predição, com o número de neurônios

definidos para cada camada, sendo que a quantidade de camadas também é um parâmetro a ser testado.

Definiu-se também que seria uma arquitetura de rede neural para cada uma das saídas, ou seja, cada rede terá suas 18 entradas e somente 1 saída, totalizando 7 redes neurais totalmente diferentes. Essa decisão se justifica, pois cada coeficiente aerodinâmico apresenta relações não-lineares distintas com as entradas, permitindo que cada rede capture essas dependências específicas sem interferência de outras saídas. Além disso, redes separadas oferecem maior flexibilidade para futuras atualizações, podendo-se refinar individualmente cada variável sem a necessidade de re-treinar todas as redes. Uma arquitetura genérica representativa de cada uma dessas redes é mostrada na Figura 16, onde somente o número de camadas intermediárias e a quantidade de neurônios em cada uma delas variam de acordo com qual saída a rede é treinada para gerar. Os treinamentos finais foram realizados utilizando o otimizador Adam com a taxa de aprendizado (*learning rate*) constante, `learning_rate_init` igual a 1×10^{-3} e `batch_size` de 200.

Figura 16 – Arquitetura genérica das redes neurais, sendo 18 entradas, 1 saída, N_1 neurônios na primeira camada oculta e N_2 na segunda camada oculta.



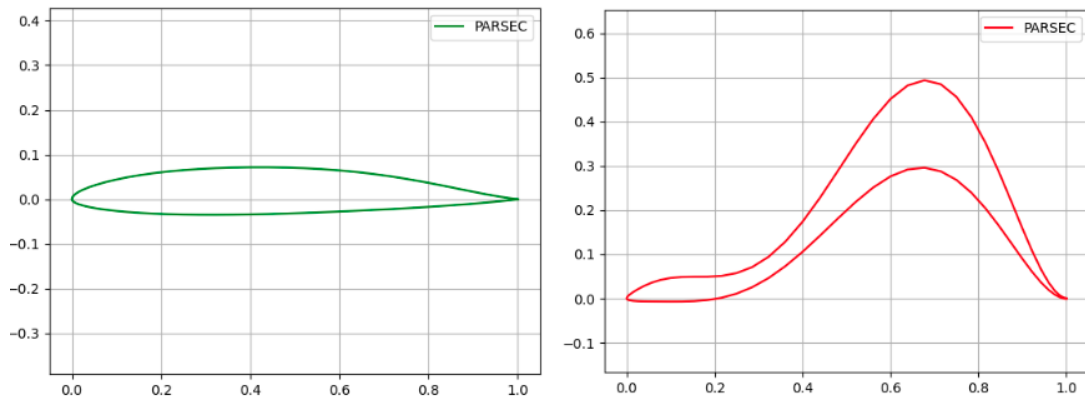
Fonte: Elaborado pelo autor.

4.1.5 Algoritmo de Classificação de Aerofólios

Para realizar uma otimização completa com os 17 parâmetros como variáveis de projeto (AR , λ , Λ , Γ , θ juntamente com os 12 parâmetros PARSEC), sem se limitar somente aos perfis do banco de dados, foi necessário criar um método de validação matemática dos aerofólios. Esse algoritmo tem como função filtrar as entradas da otimização, evitando simulações desnecessárias de combinações de parâmetros PARSEC que resultem em geometrias inviáveis, não capazes de convergir, o que atrasaria o processo de otimização. Exemplos dessas geometrias podem ser vistos na Figura 17.

O algoritmo consiste em, a partir dos 12 parâmetros PARSEC de cada perfil, gerar as coordenadas do extradorso e intradorso do aerofólio. Com essas coordenadas, calcula-se as

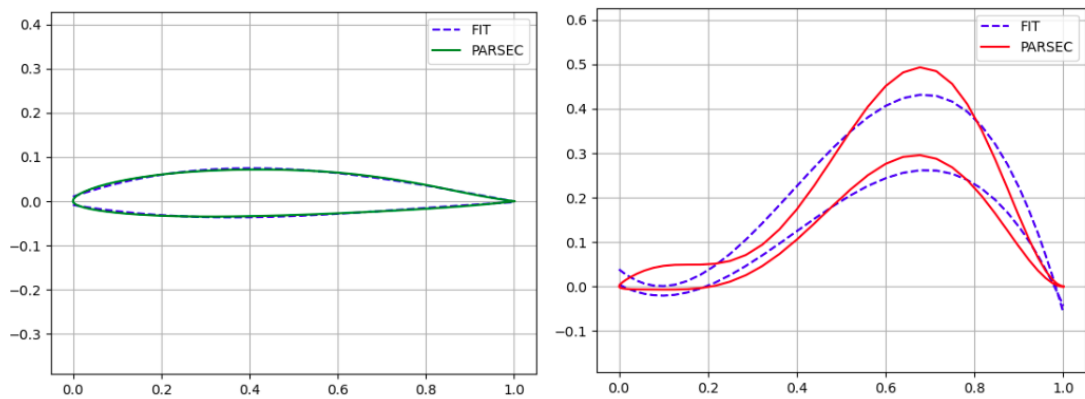
Figura 17 – Exemplo de aerofólio viável (esquerda) e aerofólio inviável (direita) gerados a partir de combinações de parâmetros PARSEC.



Fonte: Elaborado pelo autor.

curvas reais de arqueamento e espessura do perfil. Em seguida, essas curvas são ajustadas utilizando polinômios de terceiro grau. A partir dessas aproximações polinomiais para o arqueamento e a espessura, é possível obter novamente as coordenadas ajustadas do extradorso e intradorso. As curvas ajustadas podem ser visualizadas na Figura 18, onde é possível observar que com ajustes de terceiro grau, as curvas se aproximam das curvas reais do aerofólio em geometrias viáveis e se distanciam no caso de geometrias inviáveis.

Figura 18 – Comparação entre as curvas reais dos aerofólios e as curvas ajustadas por polinômios para um aerofólio de geometria válida e para outro de geometria inválida.



Fonte: Elaborado pelo autor.

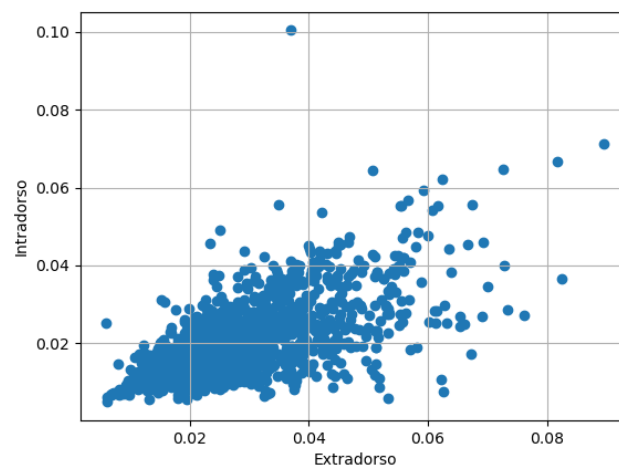
Se utilizando dessas duas curvas foi possível formular comparações a fim de classificar as combinações PARSEC em viáveis e não viáveis para as simulações. Para isso o banco de dados de perfis foi novamente utilizado para obter informações dessas comparações em geometrias previamente estabelecidas como viáveis. Com isso foram escolhidas duas métricas a serem comparadas:

- Distância Euclidiana: Mede a diferença absoluta entre os vetores de coordenadas do perfil ajustado e os vetores reais produzidos pelo PARSEC.

- Erro Quadrático Médio (MSE): Calcula o erro médio ao quadrado entre os perfis ajustados e reais.

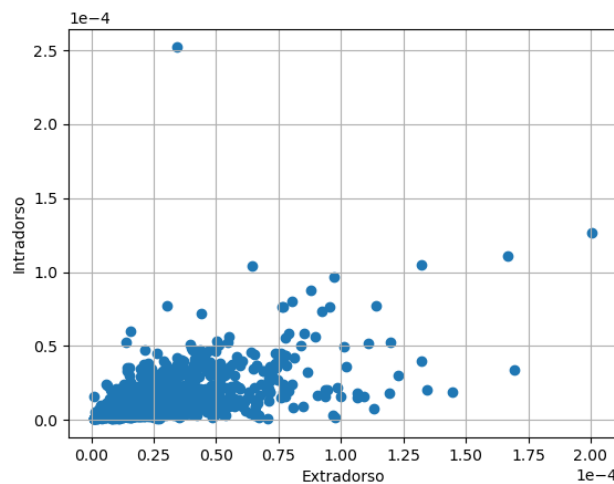
Com os dados dessas métricas para todos os perfis do banco de dados, escolheu-se um limite para a similaridade entre as curvas para definir o quão próximo o ajuste deve estar do perfil real. A escolha foi feita com base nas Figuras 19 e 20, que mostram a dispersão das métricas nos perfis do banco. Assim, os limites definidos foram 0,00021 para o MSE e 0,11 para a distância euclidiana, os quais compreendem praticamente todos os aerofólios.

Figura 19 – Dispersão das distâncias euclidianas calculadas para o intradorso e extradorso de cada um dos perfis aerodinâmicos do banco de dados PARSEC.



Fonte: Elaborado pelo autor.

Figura 20 – Dispersão dos Erros Quadráticos Médios para o intradorso e extradorso de cada um dos perfis aerodinâmicos do banco de dados PARSEC.



Fonte: Elaborado pelo autor.

Além das métricas comparativas percebeu-se que a necessidade de impor outros tipos de restrições utilizando-se das curvas reais e ajustadas, uma vez que não foi suficiente utilizar

somente as duas métricas para classificar os perfis aerodinâmicos. Assim, as primeiras derivadas das curvas do intradorso, extradorso, espessura e arqueamento em relação à coordenada x também foram avaliadas.

Para a validação determinou-se que o número de primeiras derivadas igual a zero para as curvas reais do intradorso, extradorso e arqueamento deve ser no máximo 2, e para a espessura deve ser igual a 1. Esta contagem está relacionada ao número de mudanças de sinal na primeira derivada, e os valores de referência foram adotados com base em características geométricas de aerofólios considerados aerodinamicamente válidos. Outros critérios importantes também foram considerados, como limitar a espessura máxima a $0.3c$, além de garantir que a posição da espessura máxima deve estar dentro do intervalo $0.25c$ e $0.5c$. Outro critério é que a posição do arqueamento máximo não ultrapasse a posição de $0.55c$, assim como o arqueamento máximo também não deve ser maior que $0.1c$. O aerofólio é considerado válido se todos os critérios forem atendidos.

Todos os critérios listados acima foram ajustados empiricamente no decorrer da pesquisa, conforme eram identificadas classificações incorretas de geometrias como aerofólios válidos ou inválidos. Na última versão do código, cerca de 90% dos aerofólios estão sendo classificados de maneira correta a partir do critério subjetivo da experiência do autor, o que constitui uma limitação deste estudo, cuja solução ainda se encontra em aberto. Desta forma, algumas geometrias ainda são validadas incorretamente como aerofólios, embora ainda sejam capazes de convergir nas simulações.

4.2 MÉTODO N-LLT

Como alternativa ao método VLM, empregou-se a Teoria da Linha Sustentadora Não Linear (N-LLT), com o objetivo de obter parâmetros relacionados ao estol da asa, tais como o $C_{L_{\max}}$ e o ângulo de ataque correspondente, $\alpha_{C_{L_{\max}}}$, os quais não são contemplados pelo VLM (Karali *et al.*, 2021). Todos os cálculos foram realizados a partir de curvas aerodinâmicas bidimensionais previamente determinadas para cada aerofólio, utilizando o algoritmo XFOIL por meio do software XFLR5 (Drela, 1989).

Antes da adoção definitiva do N-LLT, foram avaliadas diferentes possibilidades para estimar o efeito da geometria da asa sobre o $C_{L_{\max}}$ e o comportamento de estol. Essa análise incluiu métodos computacionais com diferentes níveis de fidelidade física e custo computacional. Considerando as limitações de tempo e os recursos computacionais disponíveis para este trabalho, foi necessário buscar um método que equilibrasse precisão adequada e viabilidade prática. A Figura 21 apresenta uma comparação entre as principais alternativas consideradas.

Diante dessa análise comparativa, o método selecionado foi a Teoria da Linha Sustentadora Não Linear (N-LLT). Ele apresentou o melhor equilíbrio entre custo computacional, disponibilidade das ferramentas, robustez dos resultados e compatibilidade com a geração de um banco de dados suficientemente amplo dentro do prazo disponível para o TCC. Dessa forma, o N-LLT tornou-se a alternativa mais adequada para estimar o comportamento de estol durante o

Figura 21 – Comparação entre métodos considerados para estimativa de estol.

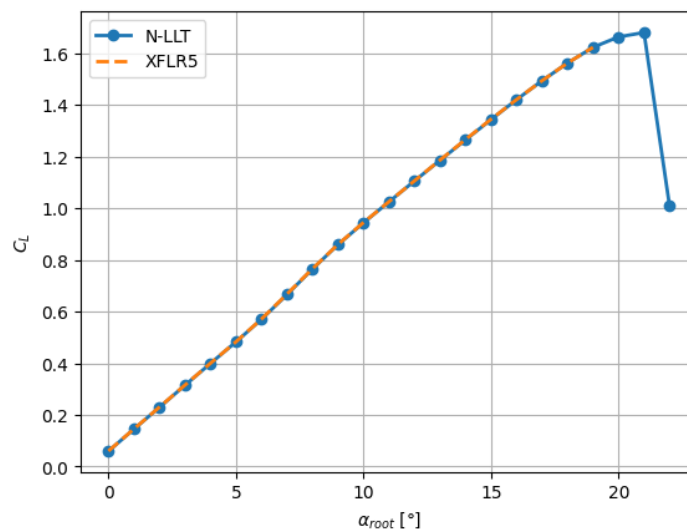
Método	Vantagens	Desvantagens
CFD (Open-FOAM)	Alta fidelidade física; captura efeitos viscosos, separação e estol com maior precisão; possibilidade de modelagem completa da geometria 3D.	Custo computacional extremamente elevado; inviável gerar um banco de dados abrangente no tempo disponível; alto esforço de configuração e malhaagem.
XFLR5 (XFOIL+LLT)	Fácil utilização; cálculos rápidos; fornece resultados bi e tridimensionais.	Dependência de XFOIL, que apresenta instabilidades e limitações para alguns aerofólios e condições de Reynolds.
N-LLT (versão implementada)	Baixo custo computacional; permite estimar $C_{L_{max}}$ e $\alpha_{C_{L_{max}}}$ com boa coerência física quando alimentado com curvas 2D confiáveis; execução rápida, possibilitando exploração ampla do espaço de projeto.	Acurácia dependente das curvas bidimensionais utilizadas; não captura efeitos tridimensionais complexos do estol, como separação progressiva na envergadura; ainda assume hipóteses simplificadoras da teoria de asas finitas.

Fonte: Elaborado pelo autor.

processo de otimização e foi implementado com base em Karali *et al.* (2021), dispensando o uso de *softwares* de terceiros.

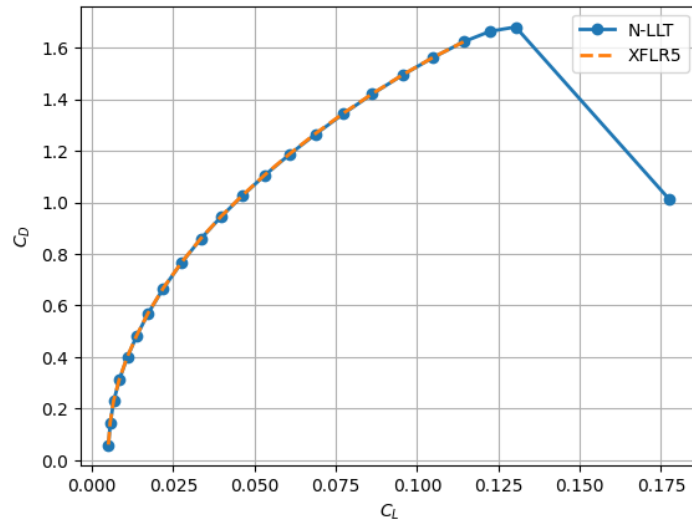
É possível visualizar as curvas geradas pelo algoritmo N-LLT comparado às curvas geradas pelo algoritmo LLT do XFLR5. As curvas de sustentação, polar de arrasto e momento de arfagem podem ser vistas nas Figuras 22, 23 e 24, respectivamente. Todas as curvas foram geradas para uma asa com $AR = 9$, $\lambda = 0.6$, $\Lambda = 10^\circ$, $\theta = -1^\circ$, $Re = 1 \times 10^6$ e aerofólio nº 0005 do banco de dados.

Figura 22 – Curvas de sustentação do modelo N-LLT implementado e XFLR5 comparadas.



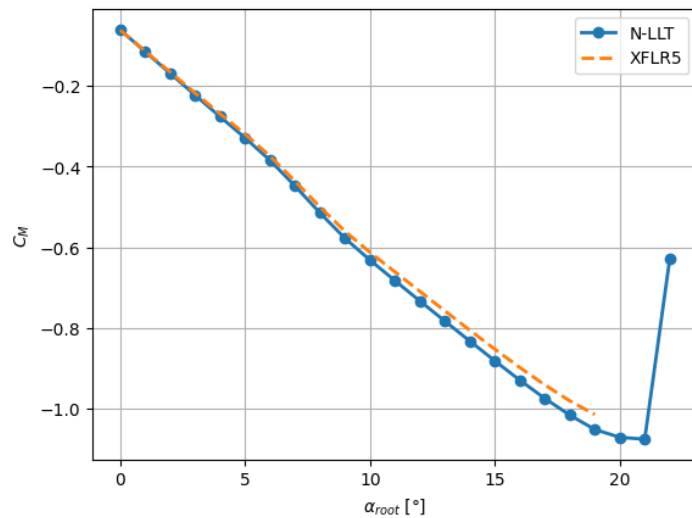
Fonte: Elaborado pelo autor.

Figura 23 – Curvas de polar de arrasto do modelo N-LLT implementado e XFLR5 comparadas.



Fonte: Elaborado pelo autor.

Figura 24 – Curvas de momento de arfagem do modelo N-LLT implementado e XFLR5 comparadas.



Fonte: Elaborado pelo autor.

4.2.1 Definição de Variáveis

De forma análoga ao procedimento adotado com o VLM, as variáveis de entrada foram definidas considerando as limitações da teoria N-LLT. Assim, o diedro foi removido da modelagem, uma vez que não é contemplado pela teoria. De modo semelhante, o enflechamento não exerce influência direta sobre as curvas de sustentação (C_L) ou sobre a polar de arrasto, embora afete as curvas de momento de arfagem (C_M).

Os parâmetros de projeto e seus respectivos limites estão apresentados na Tabela 5.

A descrição dos aerofólios seguiu a mesma abordagem empregada no método PARSEC, utilizando os limites apresentados na Tabela 2. A única diferença foi a anulação do parâmetro P_9 para todos os perfis do banco de dados de Silva e Catalano (2019), visto que valores diferentes de zero ocasionaram erros na geração das curvas bidimensionais no XFOIL.

Tabela 5 – Parâmetros de Design.

Variável	Inferior	Superior
AR [-]	5	20
λ [-]	0.2	1
Λ [°]	-25	35
θ [°]	-5	5
Re [-]	2×10^6	30×10^6

Fonte: Elaborado pelo autor.

Todos as saídas utilizadas no método VLM foram também utilizadas com a N-LLT com a adição das saídas $C_{L_{\max}}$ e $\alpha_{C_{L_{\max}}}$ totalizando 9 variáveis de saída e 16 variáveis de entrada.

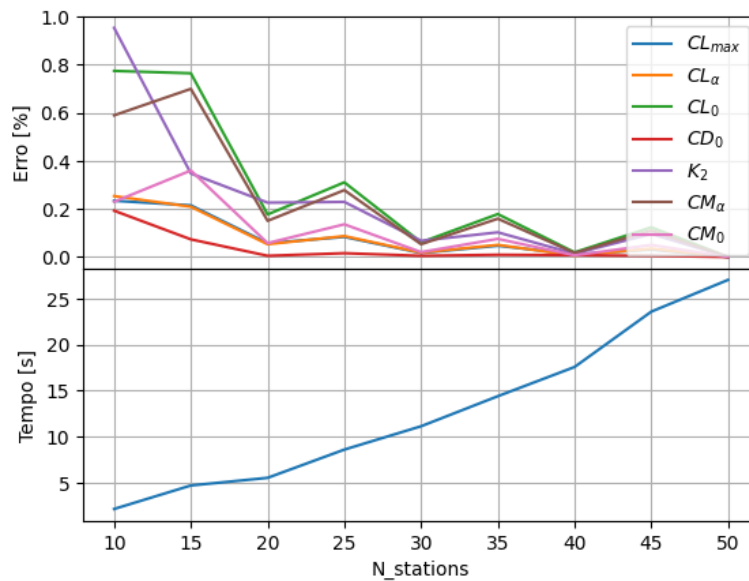
4.2.2 Malha e Convergência

A convergência da malha foi avaliada por meio do parâmetro $N_{stations}$, que representa o número de seções distribuídas ao longo da envergadura. A distribuição adotada foi cossenoidal, de modo a concentrar mais seções nas regiões próximas às pontas da asa, onde há maior variação de sustentação.

O número de pontos utilizados para descrever os perfis aerodinâmicos foi superdimensionado, uma vez que a análise bidimensional dos aerofólios é realizada previamente, e apenas as curvas finais obtidas são utilizadas no algoritmo. Dessa forma, o refinamento da malha dos aerofólios não afeta o tempo de simulação nas análises com o N-LLT. Assim, definiu-se um total de 100 pontos para a discretização dos perfis, buscando a melhor precisão possível nas simulações realizadas com o XFOIL.

A análise de convergência em função da variação de $N_{stations}$ é apresentada na Figura 25.

Figura 25 – Influência da escolha do número de seções da envergadura ($N_{stations}$) na precisão e tempo de simulação.

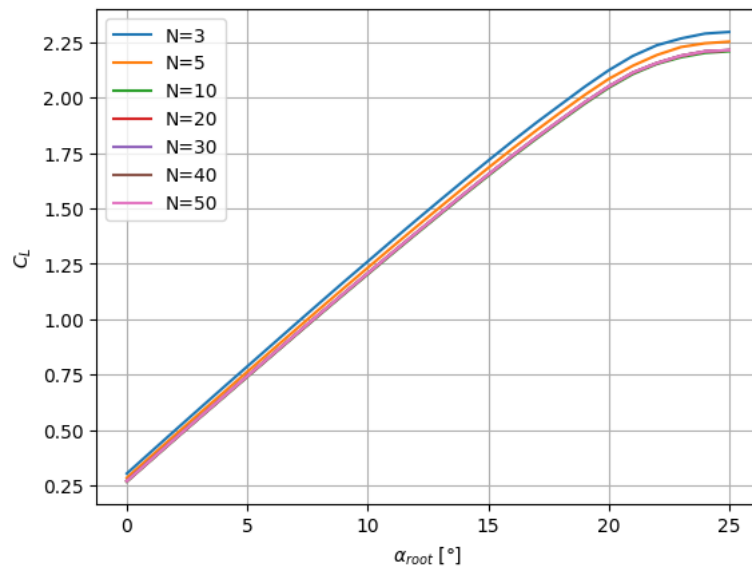


Fonte: Elaborado pelo autor.

O parâmetro K_1 também apresentou convergência e foi considerado nas análises, entretanto,

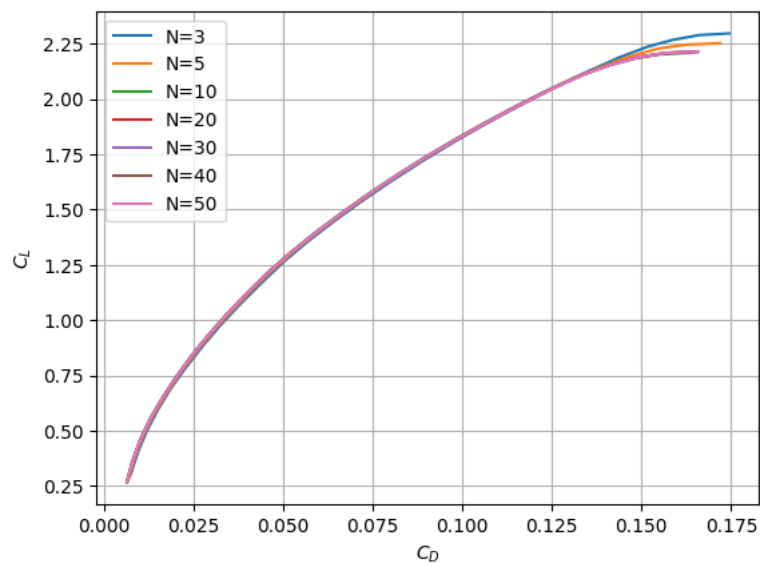
foi omitido da figura, pois sua ordem de grandeza diferiu significativamente dos demais parâmetros, o que inviabilizou uma comparação visual adequada. A variável K_1 manteve erros variando na faixa de 5% a 25%, porém o valor médio deste parâmetro sempre se aproxima de zero fazendo com que mesmo com essa ordem de grandeza, sua influência ainda seja mínima mantendo assim quase inalterado o formato das curvas, que pode ser comprovado através das Figuras 26, 27 e 28 que mostram a influência de $N_{stations}$ na sustentação, polar de arrasto e momento de arfagem, respectivamente.

Figura 26 – Influência da escolha do número de seções da envergadura ($N_{stations}$) na curva de sustentação.



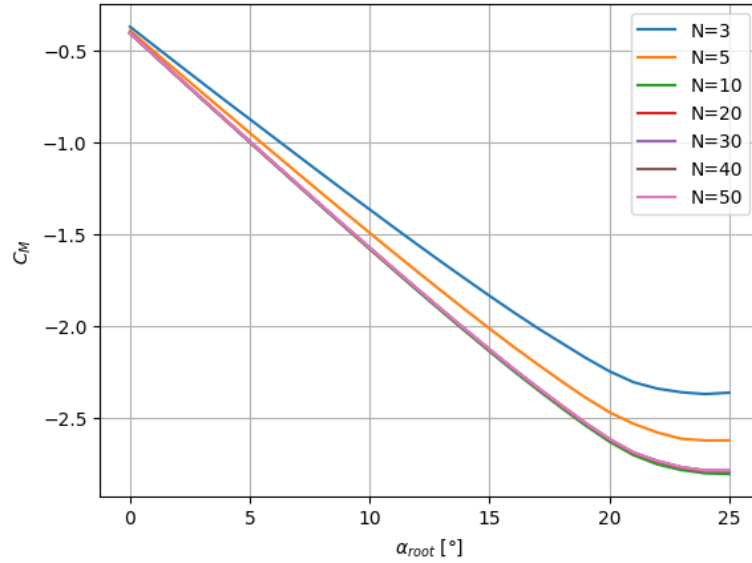
Fonte: Elaborado pelo autor.

Figura 27 – Influência da escolha do número de seções da envergadura ($N_{stations}$) na polar de arrasto.



Fonte: Elaborado pelo autor.

Figura 28 – Influência da escolha do número de seções da envergadura ($N_{stations}$) na curva de momento de arfagem.



Fonte: Elaborado pelo autor.

Sabendo a influência de $N_{stations}$ foi possível definir o valor ideal para realizar todas as simulações. Sendo o valor definido como $N_{stations} = 23$ com um tempo aproximado de execução de 9 segundos.

4.2.3 Banco de Dados

Sabendo o tempo necessário para completar uma simulação, a quantidade de simulações para gerar o banco de dados pôde ser definida com base no tempo disponível. Assim, o número de simulações foi determinado como uma combinação de todos os valores das variáveis de entrada, que foram discretizados linearmente conforme a quantidade indicada na Tabela 6, gerando um banco de dados com 1.435.200 simulações. Uma vez que cada simulação leva aproximadamente 9 segundos, o tempo total de simulação é de aproximadamente 150 dias, porém essas simulações foram executadas em um *hardware* com 30 núcleos em paralelo, reduzindo o tempo total de simulação para aproximadamente 5 dias.

Tabela 6 – Combinações de Variáveis N-LLT

Variável	Quantidade
AR	4
λ	4
Λ	3
θ	4
Re	5
Perfis	1495
Combinações	1.435.200

Fonte: Elaborado pelo autor.

Ao final uma filtragem foi realizada retirando *outliers*, e mantendo somente os casos com $C_{Lmax} \geq 9$ e $C_{D0} \geq 0$ resultando em um banco de dados com 1.405.020.

Para cada configuração as curvas foram geradas variando-se α_{root} de 0° a 30° com passo de 1° , com um pós processamento para refinamento de estol simulando novamente as configurações ao redor do estol com passo reduzido para 0.1° até que um novo C_{Lmax} fosse definido.

4.2.4 Arquitetura e Treinamento da Rede Neural

Nesta etapa, ainda utilizando uma arquitetura do tipo *Multilayer Perceptron* (MLP), adotou-se uma metodologia distinta para determinar a quantidade ideal de neurônios por camada oculta. Para cada uma das variáveis de saída foi realizada uma seleção randômica do número de neurônios, permitindo explorar diferentes capacidades representativas do modelo. As funções de ativação ReLU e tanh foram testadas manualmente e, ao final do processo, a configuração que apresentou o menor valor de RMSE foi selecionada. A função ReLU demonstrou o melhor desempenho geral, motivo pelo qual foi empregada nas redes finais.

A adoção da seleção randômica de neurônios por camada não teve como objetivo a busca exaustiva por uma arquitetura ótima, mas sim atender às restrições de tempo impostas ao desenvolvimento do trabalho. Assim, optou-se por uma abordagem estocástica simples, porém eficiente, na qual foram testadas 40 configurações aleatórias com uma, duas e três camadas ocultas para cada rede neural e com cada função de ativação. No total, nove redes distintas foram avaliadas, correspondendo às nove variáveis aerodinâmicas previstas no modelo. Essa estratégia permitiu explorar um espectro suficientemente variado de arquiteturas, mantendo o tempo de processamento em níveis viáveis para o projeto. Com isso as redes de 3 camadas intermediárias se apresentaram melhores do que as demais.

Considerando que o enflechamento exerce influência apenas nos coeficientes relacionados ao momento de arfagem, essa variável foi mantida como entrada exclusivamente nas redes destinadas ao cálculo de $C_{M\alpha}$ e C_{M0} . Assim, essas redes utilizam um total de 16 entradas (AR , λ , Λ , θ e os 11 parâmetros PARSEC). Para as demais variáveis aerodinâmicas, cuja relação com o enflechamento é negligenciável, foram utilizadas 15 entradas, omitindo-se Λ .

Os dados utilizados no treinamento foram normalizados no intervalo entre 0 e 1 por meio do `MinMaxScaler`. Foram empregados normalizadores distintos para cada rede, já que elas apresentam diferentes quantidades de variáveis de entrada. Além disso, os normalizadores foram separados entre entradas e saídas, garantindo que todas as variáveis fossem escalonadas de forma consistente. A divisão dos dados em conjuntos de treino e teste seguiu a mesma metodologia adotada para a rede VLM, destinando-se 80% dos dados para treino e 20% para teste. Os treinamentos finais foram realizados utilizando o otimizador Adam com a taxa de aprendizado constante e igual a 1×10^{-3} e `batch_size` de 200.

4.3 OTIMIZAÇÃO MULTIOBJETIVO

Após definir os modelos, a validação dos modelos substitutos foi realizada por meio de uma otimização multiobjetivo da asa, utilizando tanto o VLM e N-LLT quanto suas respectivas redes neurais, com o intuito de comparar a acurácia dos resultados e o tempo de processamento de ambos. Para isso, optou-se por uma otimização que leva em consideração dois objetivos principais:

a maximização da eficiência aerodinâmica máxima $(L/D)_{max}$, que melhora o desempenho geral da aeronave, e o coeficiente de arfagem ao redor do centro aerodinâmico (C_{Mac}), de modo a reduzir a necessidade de um grande volume de cauda para compensar o momento de arfagem.

A eficiência L/D varia com o ângulo de ataque, portanto para as otimizações foi utilizado o máximo L/D , cujo valor independe do ângulo, conforme mostrado em (Anderson, 1999). A equação utilizada é a seguinte:

$$\left(\frac{L}{D}\right)_{max} = \frac{1}{\sqrt{4K_2 C_{D0}}} \quad (9)$$

O C_{Mac} pode ser calculado a partir das saídas, ajustando a referência utilizada para o cálculo dos momentos. No OpenVSP (VLM), o coeficiente de momento é avaliado em relação ao ponto localizado a 0,25 da corda da raiz da asa. Como o C_M calculado em qualquer ponto diferente do centro aerodinâmico depende do ângulo de ataque, o uso de C_{Mac} torna-se vantajoso por ser independente do ângulo de ataque.

Para determinar o coeficiente de arfagem em torno do centro aerodinâmico, utilizou-se a Equação 10, conforme descrito em (Anderson, 1999):

$$C_{Mac} = C_{M0} - C_{L0} \frac{C_{M\alpha}}{C_{L\alpha}} \quad (10)$$

A função objetivo $f(x)$ combina a minimização de C_{Mac} e a maximização de $(L/D)_{max}$. Para alinhar ambos os objetivos ao formato de minimização, utiliza-se o sinal negativo para a eficiência aerodinâmica $(L/D)_{max}$.

Vale destacar que o balanceamento entre os dois objetivos, eficiência aerodinâmica e coeficiente de arfagem, é uma escolha de projeto. A importância relativa de cada um desses aspectos na otimização depende das prioridades estabelecidas para o projeto da aeronave. Se a eficiência aerodinâmica for priorizada, $(L/D)_{max}$ pode ter mais peso na função objetivo; por outro lado, se a estabilidade longitudinal for mais crítica, C_{Mac} terá maior relevância na otimização. Para fins de comparação dos resultados, os objetivos foram ponderados com a mesma importância, uma vez que o objetivo deste trabalho é apenas a prova de conceito do uso de um modelo substituto, e não a obtenção em si de uma asa otimizada para algum projeto em específico.

$$f(x) = C_{Mac} - \left(\frac{L}{D}\right)_{max} \quad (11)$$

Uma das principais vantagens do método apresentado neste trabalho reside na capacidade de obter resultados otimizados para novas funções objetivo em um tempo significativamente menor. Em processos de projeto aeronáutico, é comum que a função objetivo seja ajustada ou redefinida ao longo do desenvolvimento, seja para incorporar novos requisitos, seja para refinar o desempenho pretendido. Sem a aceleração proporcionada pelo uso do modelo substituto, a avaliação de múltiplas funções objetivo poderia se tornar inviável do ponto de vista computacional, comprometendo a capacidade de exploração do espaço de projeto.

Foram realizadas duas otimizações utilizando o VLM. A primeira considerou seis variáveis de projeto (AR , λ , Λ , Γ , θ e número do aerofólio), enquanto a segunda incluiu todas as 18 entradas

utilizadas pela rede neural, ampliando de forma significativa o conjunto de aerofólios possíveis. Em ambas as análises, as variáveis foram permitidas variar dentro dos intervalos definidos nas Tabelas 1 e 2. Adicionalmente, foi imposta uma restrição sobre os perfis aerodinâmicos, uma vez que diversas combinações dos parâmetros PARSEC não representam aerofólios fisicamente viáveis ou adequados ao uso. Posteriormente, os resultados da otimização realizada diretamente no VLM pelo OpenVSP foram comparados àqueles obtidos com a rede neural treinada, analisando-se o tempo de execução, a precisão dos coeficientes aerodinâmicos previstos e o desempenho global do modelo substituto.

De maneira análoga, também foram conduzidas duas otimizações utilizando o N-LLT. A primeira empregou cinco variáveis de entrada (AR , λ , Λ , θ e número do aerofólio), enquanto a segunda considerou 15 variáveis no total. Contudo, esta última não pôde ser realizada com a rotina original do N-LLT, uma vez que os dados bidimensionais dos aerofólios foram previamente coletados e não estão implementados nativamente no algoritmo, impossibilitando sua geração durante o processo iterativo. Uma otimização simples separada também foi realizada com o objetivo de explorar as características de estol do modelo, assim definiu-se outra função objetivo utilizando o coeficiente C_{Lmax} . A otimização foi realizada com *dual_annealing* e mesmos hiperparâmetros anteriormente citados.

$$f(x) = -C_{Lmax} \quad (12)$$

Por fim, realizou-se uma otimização multiobjetivo empregando ambas as redes neurais previamente treinadas, utilizando o algoritmo genético *NSGA-II*, disponível na biblioteca *pymoo*. Esse algoritmo foi selecionado devido à sua robustez, ampla utilização na literatura e suporte nativo a problemas com múltiplas funções objetivo, o que simplifica a implementação do processo de otimização. O objetivo principal dessa etapa foi obter a correspondente Fronteira de Pareto, permitindo evidenciar de forma clara os *trade-offs* envolvidos no problema de projeto.

Na primeira análise, baseada nos resultados fornecidos pelo VLM, utilizaram-se 18 parâmetros geométricos como variáveis de decisão, considerando os dois objetivos definidos na Equação 11. Para essa otimização, adotou-se uma população de 5000 indivíduos e um total de 400 gerações, além de uma taxa de mutação de 10%, valores considerados adequados para garantir diversidade populacional e convergência satisfatória sem comprometer o tempo de processamento.

No caso da abordagem utilizando a N-LLT, o mesmo algoritmo foi empregado, desta vez considerando a otimização completa das 15 variáveis disponíveis. Para alcançar convergência adequada em um espaço de projeto de maior complexidade, utilizou-se a mesma população, composta por 5000 indivíduos, distribuídos ao longo de 400 gerações e com taxa de mutação de 10%. Nessa formulação, foram tratados simultaneamente a minimização do coeficiente de momento no centro aerodinâmico, C_{Mac} , e a maximização do coeficiente de sustentação máximo, C_{Lmax} . A combinação desses objetivos está apresentada na Equação 13, que descreve formalmente a função a ser otimizada.

$$f(x) = C_{Mac} - C_{Lmax} \quad (13)$$

5 RESULTADOS

5.1 BANCO DE DADOS

O banco de dados do VLM e N-LLT geraram saídas que variam em certos intervalos, os quais são mostrados nas Tabelas 7 e 8, respectivamente. Através destes dados é possível observar que as simulações convergiram e geraram saídas próximas do esperado. Tanto os bancos de dados quanto os códigos e algoritmos utilizados foram disponibilizados em repositório aberto (Carrari; Silva, 2025).

Tabela 7 – Mínimos, máximos, médias e desvios padrões para cada uma das saídas do VLM.

Saída	Mínimo	Máximo	Média	DP
$C_{L\alpha}$	0,0711	0,1053	0,0900	0,0076
C_{L0}	-0,3421	1,3868	0,2572	0,2162
C_{D0}	0,0032	0,0094	0,0074	0,0005
K_1	-0,0033	0,0100	0,0004	0,0011
K_2	0,0113	0,0414	0,0220	0,0088
C_{M0}	-8,7359	5,3084	-0,2983	0,9554
$C_{M\alpha}$	-0,6207	0,3993	-0,0766	0,2308

Fonte: Elaborado pelo autor.

Tabela 8 – Mínimos, máximos, médias e desvios padrões para cada uma das saídas da N-LLT.

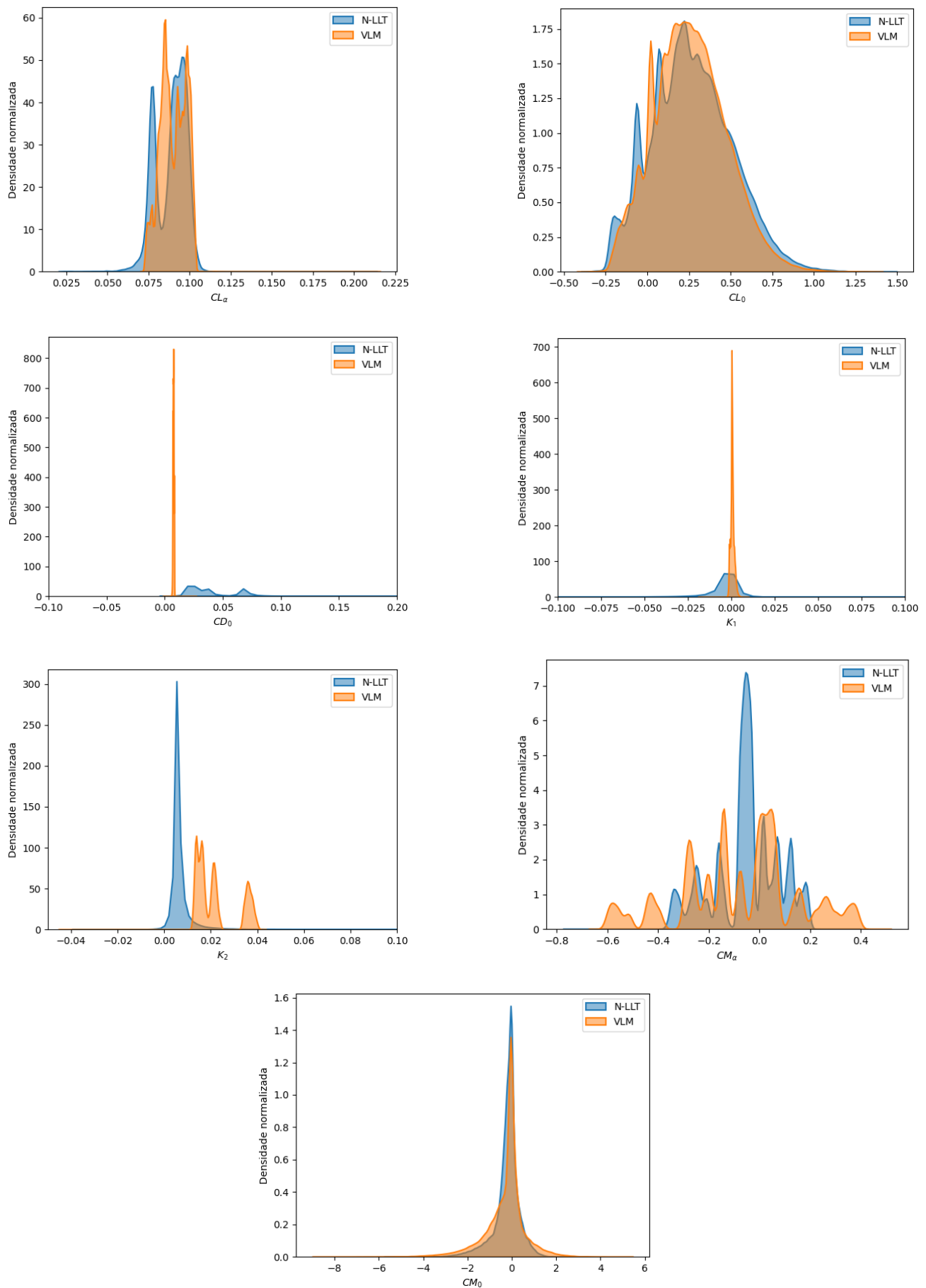
Saída	Mínimo	Máximo	Média	DP
C_{Lmax}	0,1431	2,5863	1,8349	0,2848
α_{CLmax}	9,000	30,000	21,653	4,259
$C_{L\alpha}$	0,0217	0,2126	0,0885	0,0094
C_{L0}	-0,3569	1,4576	0,2646	0,2397
C_{D0}	0,0007	1,1853	0,0409	0,0271
K_1	-0,8915	0,1118	-0,0049	0,0184
K_2	-0,0382	0,3029	0,0069	0,0069
C_{M0}	-5,6640	2,7455	-0,2212	0,5875
$C_{M\alpha}$	-0,7538	0,3776	-0,0524	0,1296

Fonte: Elaborado pelo autor.

A partir dos dados obtidos, observa-se que as saídas C_{M0} e $C_{M\alpha}$ foram as que apresentaram maior variação, o que pode indicar uma maior dificuldade na construção de modelos precisos para esses parâmetros. Também é possível notar que a saída K_1 se distribui em torno de zero em praticamente todos os casos, o que está em conformidade com a literatura, que frequentemente desconsidera sua influência nas formulações de polar de arrasto subsônica.

As demais saídas apresentaram resultados consistentes, com desvios padrão relativamente baixos e sem a presença evidente de *outliers*. As distribuições das variáveis de saída podem ser visualizadas na Figura 29.

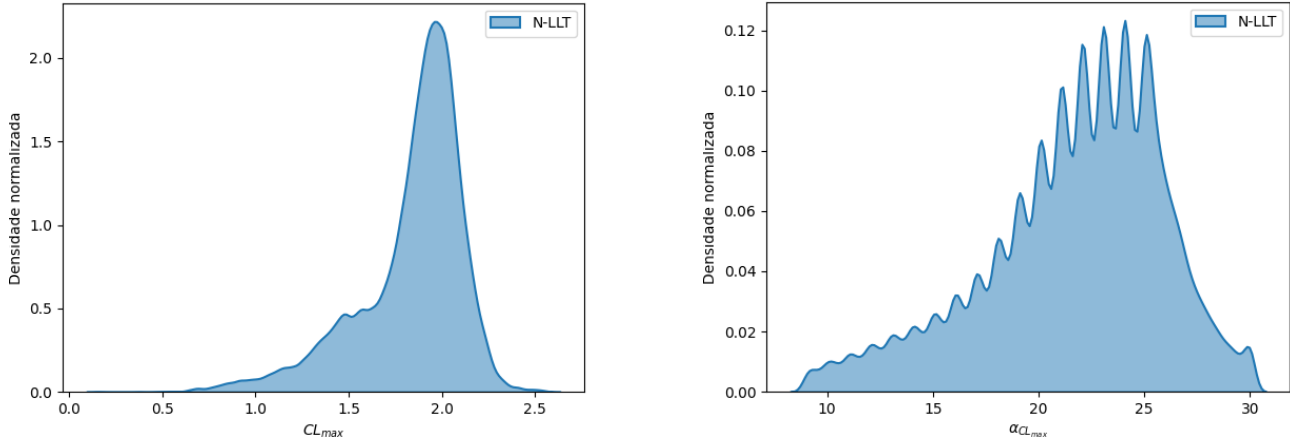
Figura 29 – Comparação da distribuição normalizada dos dados das saídas dos algoritmos VLM e N-LLT.



Fonte: Elaborado pelo autor.

As saídas que capturam o processo de estol estão representadas na Figura 30 geradas com a N-LLT. O aspecto ondulado da distribuição de ângulos de ataque correspondentes ao C_{Lmax} se deve ao fato da discretização ter sido feita com passo de 0.1° para se obter o C_{Lmax} .

Figura 30 – Distribuição dos dados das saídas de estol da N-LLT implementada.



Fonte: Elaborado pelo autor.

5.2 MÉTODO VLM

5.2.1 Arquitetura e Treinamento da Rede Neural

Cada uma das 7 redes neurais foi otimizada através do Dual Annealing. Primeiramente foi realizada uma otimização mais simples, com apenas uma camada oculta, onde a única variável de projeto foi a quantidade de neurônios desta camada (N_1), com o limite inferior sendo 10 neurônios e o limite superior sendo 250 neurônios. A função objetivo a ser minimizada foi o RMSE relacionado a previsão da rede. Nota-se que o Dual Annealing é um algoritmo estocástico, ou seja, utilizá-lo mais de uma vez com as mesmas entradas, pode gerar resultados diferentes, mas ambos com alta probabilidade de estar próximo do mínimo global da função objetivo, e um destes pode ser observado na Tabela 9 abaixo.

Tabela 9 – Número de neurônios (N_1) e erro médio quadrático (RMSE) para cada saída.

Saída	$C_{L\alpha}$	C_{L0}	C_{D0}	K_1	K_2	$C_{M\alpha}$	C_{M0}
N_1	168	184	192	157	226	194	233
RMSE	0,00119	0,00479	0,00192	0,00346	0,00155	0,00067	0,00367

Fonte: Elaborado pelo autor.

Uma segunda otimização foi feita, dessa vez cada rede neural com duas camadas ocultas, onde as variáveis de projeto N_1 e N_2 variam de 3 a 300 neurônios. Novamente o erro foi minimizado e chegou-se nos valores ótimos para os neurônios de cada camada mostrados na Tabela 10.

Tabela 10 – Número de neurônios (N_1 e N_2) e RMSE para cada saída.

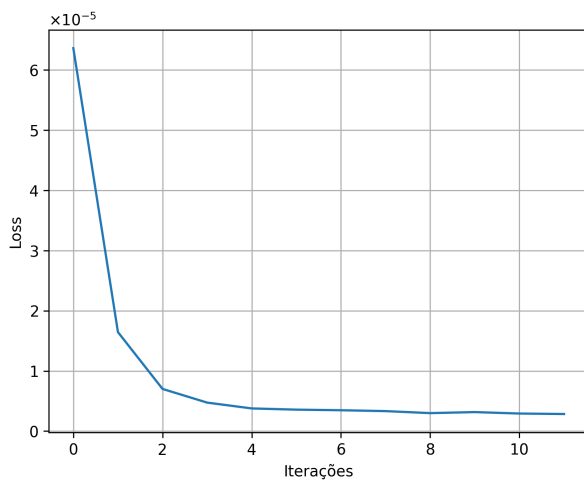
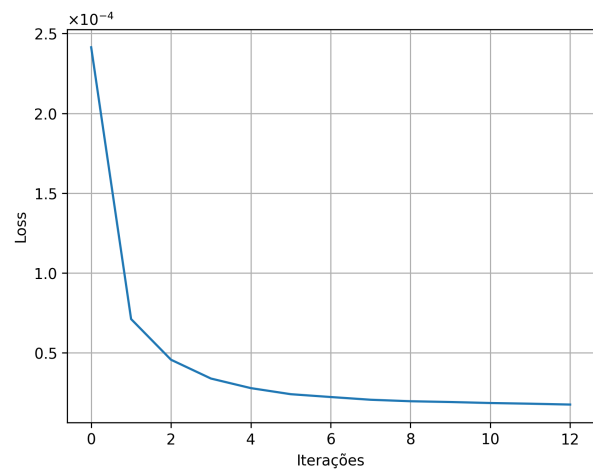
Saída	$C_{L\alpha}$	C_{L0}	C_{D0}	K_1	K_2	$C_{M\alpha}$	C_{M0}
N_1	293	223	257	273	281	222	213
N_2	171	273	259	117	16	278	187
RMSE	0,00116	0,00314	0,00177	0,00331	0,00149	0,00063	0,00280

Fonte: Elaborado pelo autor.

O tempo necessário para otimizar uma rede com duas camadas mostrou que uma rede com três camadas se torna impraticável devido à capacidade computacional e ao tempo disponível, já que o custo computacional aumenta drasticamente com o número de variáveis de projeto. Portanto, a otimização com duas camadas foi escolhida por se revelar mais eficiente, apesar da diferença ser pequena. É importante destacar que as otimizações foram realizadas treinando a rede neural com o banco de dados completo. Uma vez que uma configuração de rede que apresenta um desempenho melhor com um banco de dados reduzido não necessariamente terá o mesmo desempenho com o banco de dados completo. Em outras palavras, as otimizações foram feitas com todos os dados disponíveis para garantir a validade dos resultados, e por isso foram computacionalmente custosas.

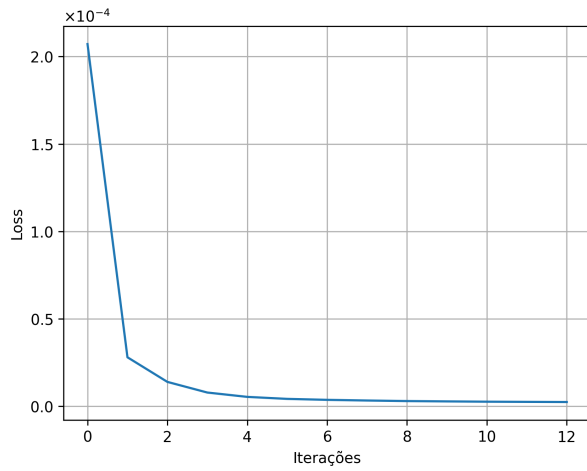
As Figuras 31, 32 e 33 apresentam as curvas de aprendizado das redes treinadas com o banco de dados completo, destacando o comportamento das perdas ao longo das épocas para todas as variáveis que descrevem sustentação, momento de arfagem e arrasto, respectivamente. Observa-se que todos os treinos apresentaram convergências estáveis com iterações próximas a 12.

Figura 31 – Curvas de aprendizado do treinamento final para as saídas de sustentação realizado com o banco de dados VLM.

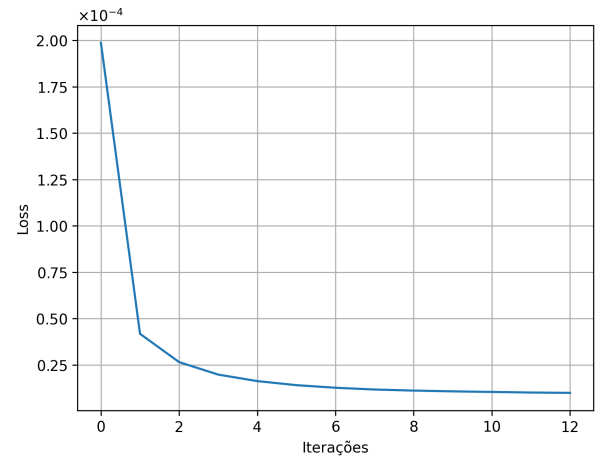
(a) Curva de aprendizado da saída $C_{L\alpha}$.(b) Curva de aprendizado da saída C_{L0} .

Fonte: Elaborado pelo autor.

Figura 32 – Curvas de aprendizado do treinamento final para as saídas de momento de arfagem realizado com o banco de dados VLM.



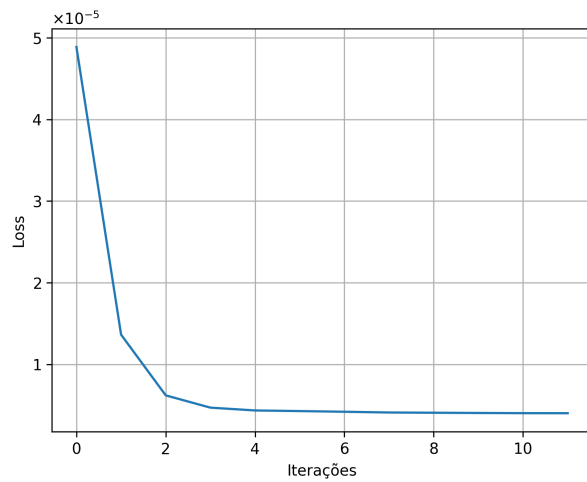
(a) Curva de aprendizado da saída $C_{M\alpha}$.



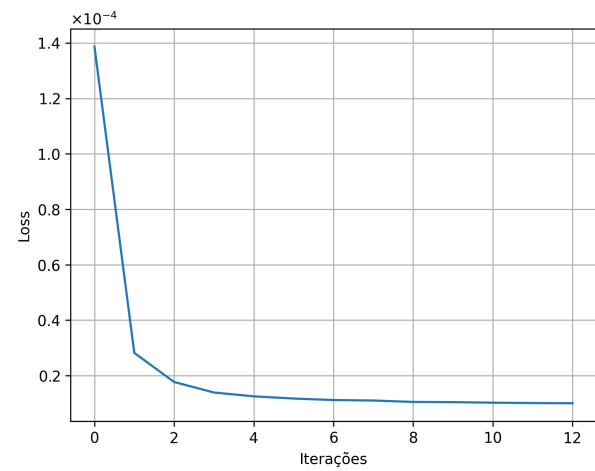
(b) Curva de aprendizado da saída C_{M0} .

Fonte: Elaborado pelo autor.

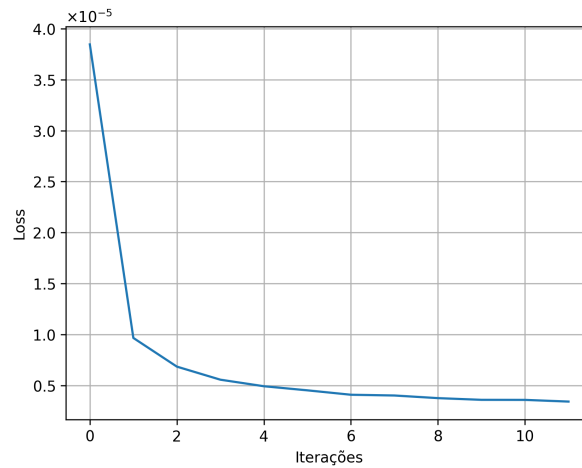
Figura 33 – Curvas de aprendizado do treinamento final para as saídas de arrasto realizado com o banco de dados VLM.



(a) Curva de aprendizado da saída C_{D0} .



(b) Curva de aprendizado da saída K_1 .



(c) Curva de aprendizado da saída K_2 .

Fonte: Elaborado pelo autor.

5.2.2 Comparações entre Rede Neural e VLM

Após o treinamento dos modelos, a Equação 11 foi empregada como função objetivo a ser minimizada, tanto por meio das redes neurais quanto com o uso do software OpenVSP, utilizando o algoritmo `dual_annealing` da biblioteca SciPy. O objetivo desse procedimento foi comparar os resultados das otimizações obtidos por ambas as abordagens, permitindo uma análise da acurácia e da eficiência dos modelos treinados.

Ambas as otimizações foram conduzidas com seis variáveis de projeto: alongamento (AR), afilamento (λ), enflechamento (Λ), diedro (Γ), torção (θ) e o número do perfil no banco de dados. Ao final, foram registrados o tempo total de execução, o valor ótimo da função objetivo e os correspondentes valores das variáveis de projeto no ponto ótimo.

Para validar os resultados da otimização realizada com os modelos treinados, uma verificação adicional foi conduzida no VLM, a fim de calcular o valor da função objetivo por meio do método VLM. Os resultados obtidos estão apresentados na Tabela 11.

Tabela 11 – Comparação de resultados das otimizações com 6 parâmetros no VLM e Rede Neural.

Parâmetro	VLM	Rede Neural	Teste VLM
AR	19,91	20,0	20,0
λ	0,433	0,474	0,474
Λ [°]	34,867	35,0	35,0
θ [°]	1,223	0,238	0,238
Γ [°]	3,416	-2,230	-2,230
Perfil	1459	1459	1459
Tempo [s]	60325,1	7,5	—
Func. Objetivo	-48,1567	-52,1950	-47,8888

Fonte: Elaborado pelo autor.

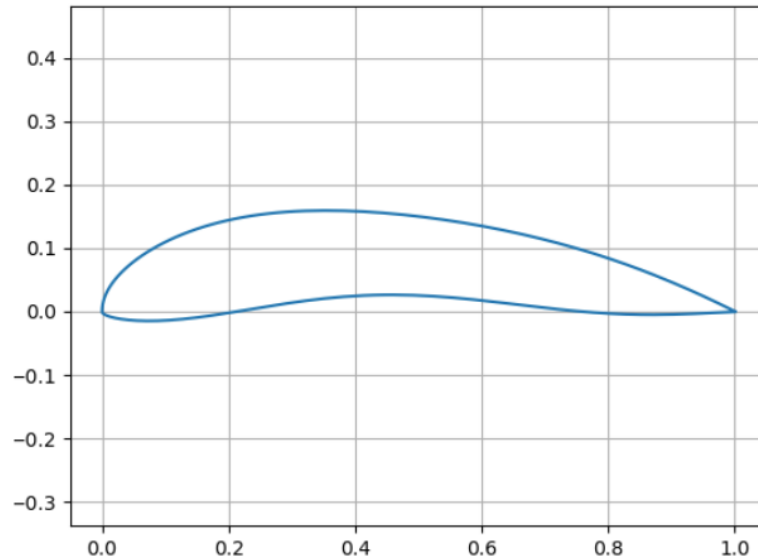
Observa-se que a rede neural treinada obteve valores muito próximos para a função objetivo, porém com um tempo de execução aproximadamente 8.043 vezes menor. É importante destacar que a função objetivo é altamente complexa e pode apresentar diversos mínimos locais. Por esse motivo, aliado à natureza inerentemente estocástica do algoritmo de otimização utilizado, cada execução tende a produzir resultados diferentes, fazendo com que os perfis aerodinâmicos obtidos pela rede neural nem sempre coincidam com os gerados pelo OpenVSP.

Para fins comparativos, foi incluído na Tabela 11 um caso específico em que ambos os métodos resultaram em perfis idênticos, permitindo a análise de possíveis mínimos equivalentes. A Figura 34 apresenta a imagem do aerofólio 1459, obtido pelas duas abordagens.

Notou-se também que as variáveis AR, λ e Λ convergiram consistentemente em todas as execuções. Em contraste, as variáveis Γ e θ não apresentaram convergência em nenhuma das execuções, o que pode indicar duas possibilidades: ou os modelos não conseguiram aprender adequadamente o comportamento do efeito dessas variáveis no desempenho aerodinâmico, ou tais variáveis possuem baixa ou nenhuma influência sobre a função objetivo.

Em uma segunda comparação, de natureza semelhante à anterior, foram realizadas otimizações mais complexas, totalizando 17 variáveis de projeto: AR, λ , Λ , Γ , θ , P_1 , P_2 , P_3 , P_4 , P_5 ,

Figura 34 – Aerofólio 1459 resultante das otimizações realizadas com VLM e sua rede neural.



Fonte: Elaborado pelo autor.

P_6 , P_7 , P_8 , P_9 , P_{10} , P_{11} e P_{12} . Ambas as otimizações foram conduzidas com o uso do algoritmo Dual Annealing e utilizaram a Equação 11 como função objetivo.

Por se tratar de uma função ainda mais complexa, os mesmos desafios observados na comparação anterior também se aplicam, resultando em soluções distintas a cada execução. Isso é especialmente evidente na geometria do aerofólio, cuja forma é altamente sensível às variações dos parâmetros PARSEC.

Optou-se por utilizar um método estocástico de otimização em detrimento de métodos determinísticos ou baseados em gradiente, devido à sua capacidade de evitar aprisionamento em mínimos locais e lidar com funções objetivo potencialmente não contínuas. No presente problema, algumas combinações de parâmetros não convergem nas simulações, impossibilitando garantir a continuidade da função objetivo, cenário em que métodos determinísticos ou de gradiente poderiam falhar ou apresentar desempenho inferior.

Reconhece-se que este tipo de algoritmo apresenta variação nos resultados devido à sua natureza aleatória, o que limita a comparação direta entre execuções. No entanto, para o objetivo desta prova de conceito, o método adotado mostrou-se adequado, permitindo reduzir significativamente o tempo de otimização e obter resultados de desempenho semelhantes, mesmo que as geometrias finais dos aerofólios não sejam idênticas entre as execuções.

Os resultados obtidos nas otimizações estão comparados na Tabela 12, juntamente com a validação do resultado final dos modelos treinados no OpenVSP. As geometrias dos aerofólios resultantes estão representadas na Figura 35.

Tabela 12 – Comparação de resultados das otimizações com 17 parâmetros no VLM e Rede Neural.

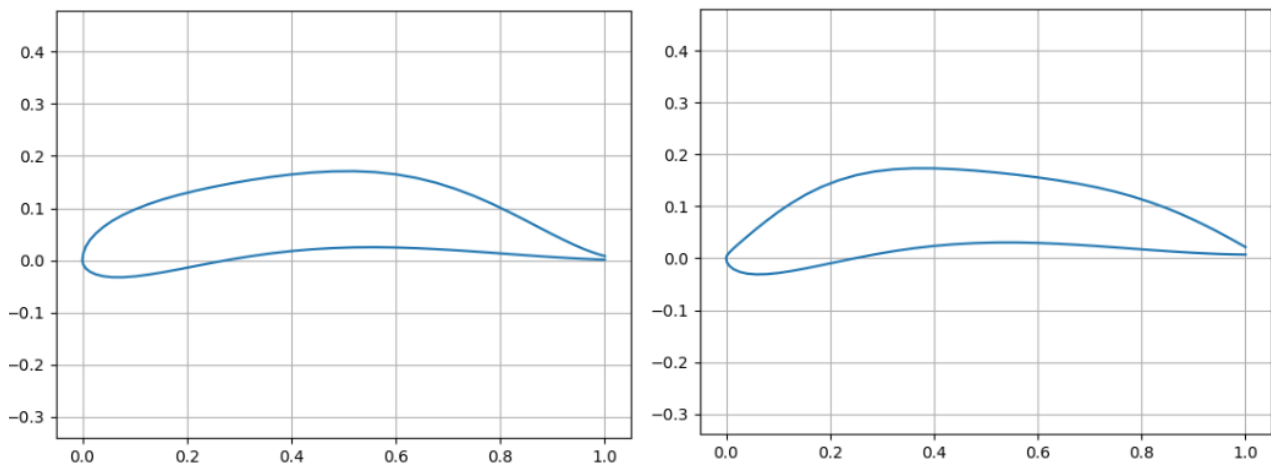
Parâmetro	VLM	Rede Neural	Teste VLM
AR	20,0	19,93	19,93
λ	0,52	0,47	0,47
Λ [°]	34,15	34,96	34,96
θ [°]	4,82	1,08	1,08
Γ [°]	-4,09	4,48	4,48
P_1	0,0533	0,0081	0,0081
P_2	0,0192	0,0189	0,0189
P_3	0,5076	0,3778	0,3778
P_4	0,1712	0,1733	0,1733
P_5	-1,2961	-1,1919	-1,1919
P_6	3,0956	2,8588	2,8588
P_7	0,7481	0,8818	0,8818
P_8	-0,4011	-1,6830	-1,6830
P_9	0,0070	0,0145	0,0145
P_{10}	0,0048	0,0142	0,0142
P_{11}	8,168	15,068	15,068
P_{12}	12,613	28,942	28,942
Tempo [s]	94982,7	24,5	—
Func. Objetivo	-49,518	-54,596	-48,493

Fonte: Elaborado pelo autor.

Mais uma vez, observaram-se resultados convergentes para as variáveis AR, λ e Λ em todas as execuções, enquanto Γ e θ não apresentaram convergência.

O modelo treinado foi capaz de alcançar resultados extremamente próximos aos obtidos com o OpenVSP, especialmente após a validação final, na qual o resultado da rede neural foi avaliado diretamente no software, mostrando que os modelos treinados foram capazes de completar seu objetivo de servir como um modelo substituto para o método VLM.

Figura 35 – Aerofólios resultantes das otimizações conduzidas no OpenVSP (esquerda) e Rede Neural (direita).



Fonte: Elaborado pelo autor.

As geometrias obtidas, apresentadas na Figura 35, exibem características visuais que se afastam dos perfis clássicos de aerofólios, um desafio ainda em aberto na área, devido à dificuldade de validar esse tipo de forma. Esse comportamento não é exclusivo deste trabalho e, além disso, as características geométricas dos aerofólios gerados também estão relacionadas às limitações do método VLM, que impede a aplicação de certas restrições durante o processo de otimização. Por exemplo, não é possível modelar o estol, o que dificulta impor limites ao arqueamento do perfil.

Nessa comparação, a rede neural mostrou-se aproximadamente 3876 vezes mais rápida que a otimização realizada diretamente no OpenVSP, demonstrando alta eficiência na estimativa de mínimos da função objetivo. Embora não atinja precisão absoluta em todos os casos, tal nível de exatidão não é estritamente necessário no contexto da otimização, especialmente nas fases iniciais de projeto.

5.3 MÉTODO N-LLT

5.3.1 Arquitetura e Treinamento da Rede Neural

Cada uma das nove redes neurais teve seus hiperparâmetros avaliados por meio de uma estratégia de seleção randômica. Inicialmente, investigaram-se arquiteturas mais simples, compostas por apenas uma camada oculta, nas quais a única variável ajustável era a quantidade de neurônios dessa camada, limitada entre 10 e 500 neurônios. Em seguida, foram testadas configurações mais complexas, contendo 2, 3 e 4 camadas ocultas, todas com o número de neurônios variando dentro do mesmo intervalo entre 10 e 500.

A comparação dos resultados indicou que a arquitetura com 3 camadas ocultas apresentou o melhor desempenho médio dentre as opções avaliadas. No entanto, observou-se que todas as arquiteturas testadas produziram valores de RMSE de mesma ordem de grandeza, sugerindo que o problema é relativamente bem comportado e que múltiplas configurações de rede são capazes de representá-lo de forma adequada. Os valores de RMSE obtidos para as diferentes arquiteturas podem ser observados na Tabela 13.

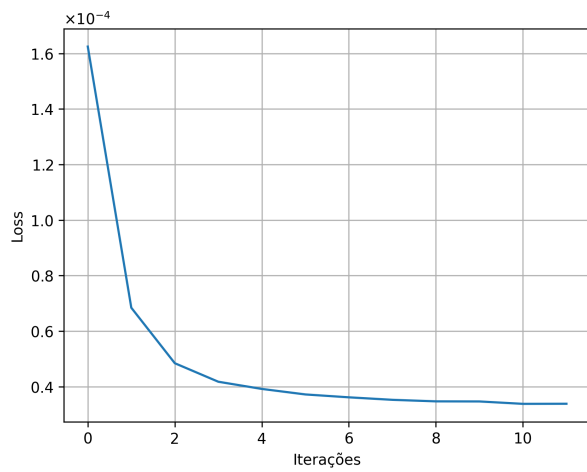
Tabela 13 – Número de neurônios (N_1 , N_2 e N_3) e RMSE para cada saída.

Saída	$C_{L\alpha}$	C_{L0}	C_{Lmax}	aC_{Lmax}	C_{D0}	K_1	K_2	$C_{M\alpha}$	C_{M0}
N_1	134	288	300	360	306	330	95	314	203
N_2	364	264	210	275	470	258	489	170	280
N_3	252	228	103	466	372	454	404	226	88
RMSE	0,00532	0,00437	0,00942	0,02592	0,00555	0,00721	0,00754	0,00364	0,00364

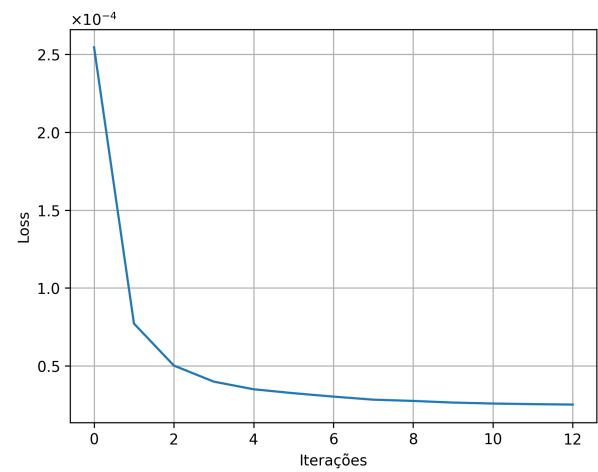
Fonte: Elaborado pelo autor.

As Figuras 36, 37 e 38 apresentam as curvas de aprendizado das redes treinadas com o banco de dados completo, destacando o comportamento das perdas ao longo das épocas para todas as variáveis que descrevem sustentação, momento de arfagem e arrasto, respectivamente. Observa-se que todos os treinos também apresentaram convergências estáveis com iterações próximas a 12.

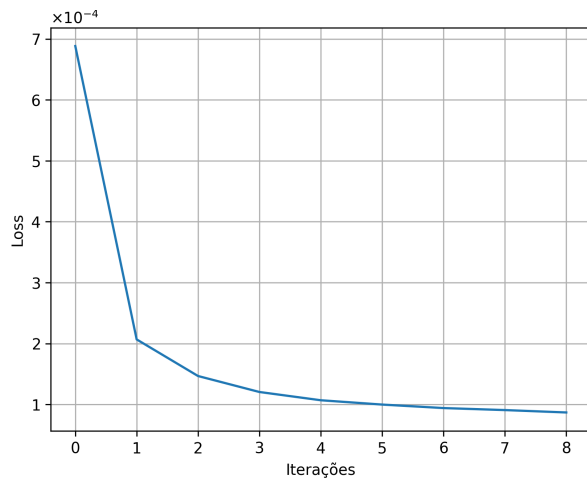
Figura 36 – Curvas de aprendizado do treinamento final para as saídas de sustentação realizado com o banco de dados N-LLT.



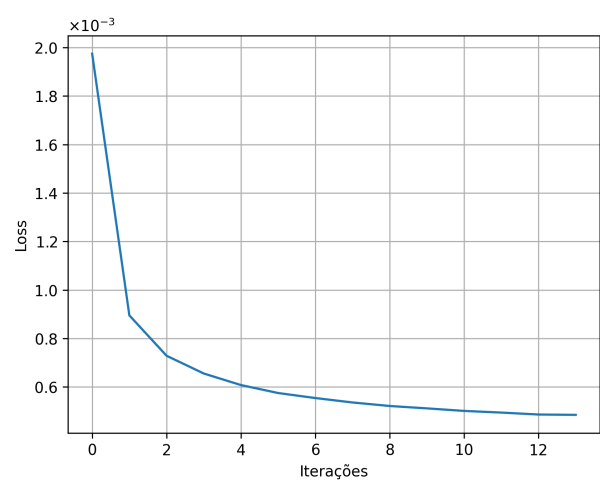
(a) Curva de aprendizado da saída $C_{L\alpha}$.



(b) Curva de aprendizado da saída C_{L0} .



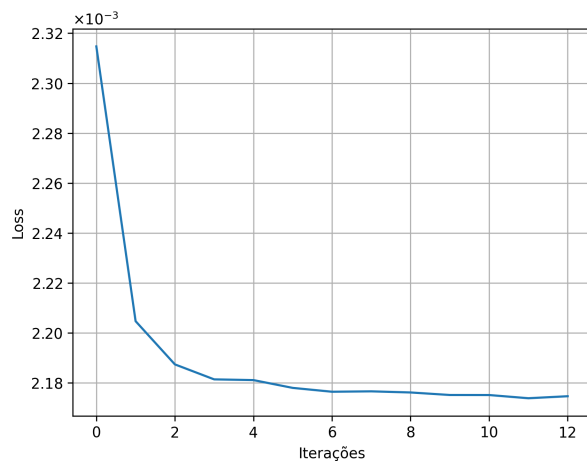
(c) Curva de aprendizado da saída C_{max} .



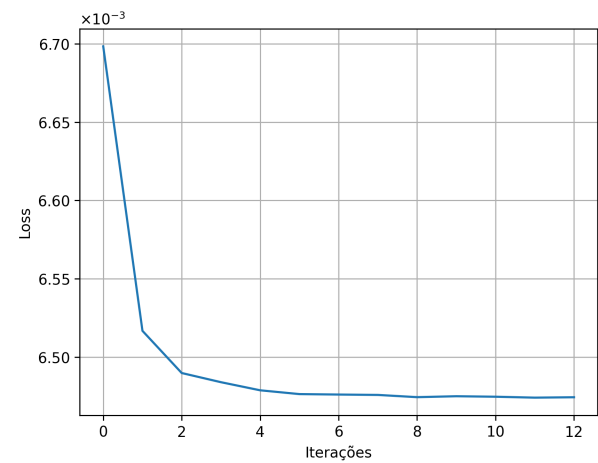
(d) Curva de aprendizado da saída αC_{Lmax} .

Fonte: Elaborado pelo autor.

Figura 37 – Curvas de aprendizado do treinamento final para as saídas de momento de arfagem realizado com o banco de dados N-LLT.



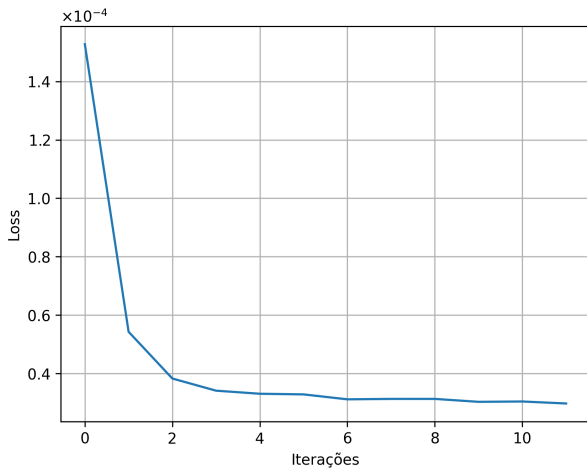
(a) Curva de aprendizado da saída $C_{M\alpha}$.



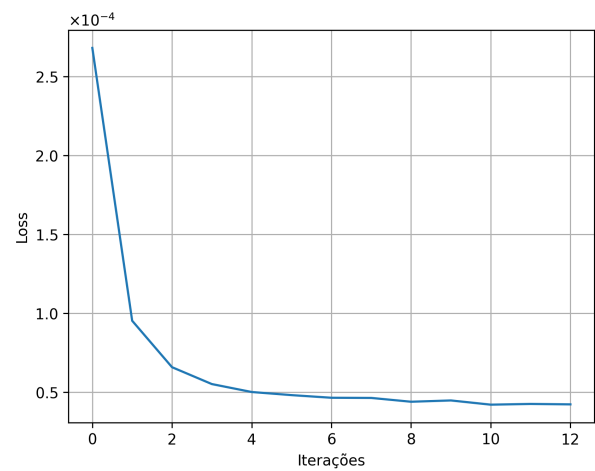
(b) Curva de aprendizado da saída C_{M0} .

Fonte: Elaborado pelo autor.

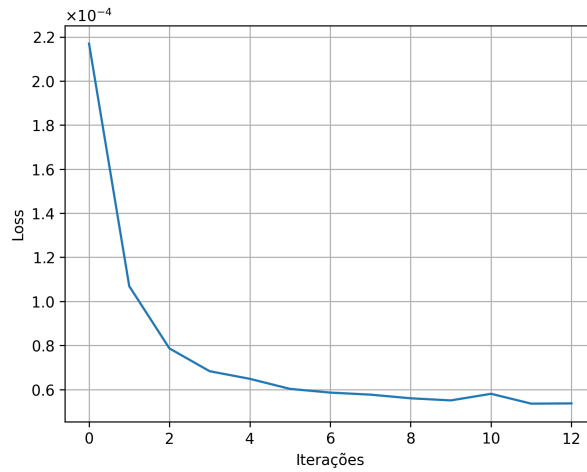
Figura 38 – Curvas de aprendizado do treinamento final para as saídas de arrasto realizado com o banco de dados N-LLT.



(a) Curva de aprendizado da saída C_{D0} .



(b) Curva de aprendizado da saída K_1 .



(c) Curva de aprendizado da saída K_2 .

Fonte: Elaborado pelo autor.

5.3.2 Comparações entre Rede Neural e N-LLT

Após o treinamento dos modelos baseados na N-LLT, a Equação 11 foi empregada como função objetivo a ser minimizada, tanto por meio das redes neurais quanto do próprio N-LLT, utilizando o algoritmo *dual_annealing* da biblioteca SciPy. O intuito desse procedimento, de maneira análoga ao realizado para o VLM, foi comparar os resultados das otimizações obtidos por ambas as abordagens, permitindo avaliar a acurácia, a eficiência computacional e o comportamento geral dos modelos substitutos.

As duas otimizações foram conduzidas considerando cinco variáveis de projeto: alongamento (AR), afilamento (λ), enfechamento (Λ), torção (θ) e o número do aerofólio no banco de dados. Ao final de cada execução, foram registrados o tempo total de processamento, o valor ótimo da função objetivo e os correspondentes valores das variáveis de projeto no ponto ótimo.

Para validar os resultados obtidos com os modelos treinados, foi realizada uma verificação adicional utilizando o N-LLT, calculando-se novamente o valor da função objetivo por meio desse método. Os resultados consolidados encontram-se na Tabela 14.

Tabela 14 – Comparação de resultados das otimizações com 5 parâmetros na N-LLT e Rede Neural.

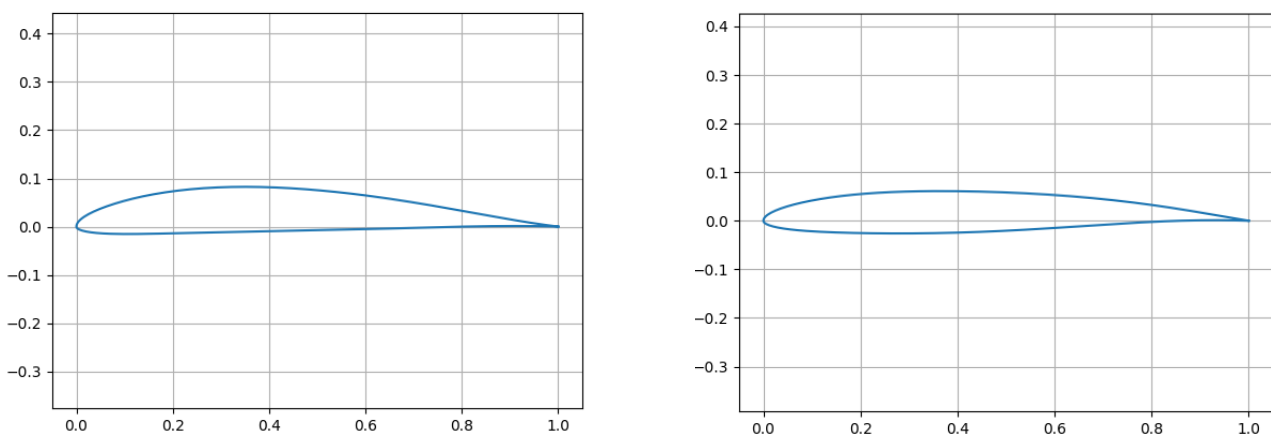
Parâmetro	N-LLT	Rede Neural	Teste N-LLT
AR	17,77	20,0	20,0
λ	0,91	1,0	1,0
Λ [°]	-12,12	32,67	32,67
θ [°]	1,77	0,66	0,66
Perfil	1406	1331	1331
Tempo [s]	12717,8	4,54	–
Func. Objetivo	-59,9920	-64,0269	-63,0032

Fonte: Elaborado pelo autor.

Observa-se que a rede neural foi capaz de obter um valor de função objetivo bastante próximo ao calculado diretamente pela N-LLT, porém com um tempo de execução aproximadamente 2.801 vezes menor. A função objetivo considerada é altamente complexa e apresenta múltiplos mínimos locais; assim, devido ao caráter estocástico do algoritmo de otimização empregado, diferentes execuções tendem naturalmente a convergir para soluções distintas. Em consequência, os perfis aerodinâmicos e os valores das variáveis de projeto obtidos via rede neural nem sempre coincidem com aqueles encontrados diretamente pelo N-LLT.

Nessa etapa, as otimizações resultaram em aerofólios distintos, impossibilitando a comparação direta de mínimos referentes a um mesmo perfil. Ainda assim, a Tabela 14 apresenta um caso em que a rede neural foi capaz de atingir um valor de função objetivo inferior ao encontrado pela otimização via N-LLT. A Figura 39 ilustra, os aerofólios 1406 e 1331, resultantes das otimizações por N-LLT e pela rede neural.

Figura 39 – Comparação dos aerofólios do banco de dados PARSEC obtidos nas otimizações realizadas com a N-LLT e sua rede neural. Aerofólio 1406 (esquerda) e 1331 (direita).



Fonte: Elaborado pelo autor.

Observou-se ainda que as variáveis AR e λ apresentaram comportamento convergente em todas as execuções, embora, para λ , os valores diferissem daqueles obtidos para o VLM. Em contraste, as variáveis Λ e θ não exibiram convergência consistente, o que pode indicar duas

possibilidades: os modelos substitutos não capturaram de forma satisfatória a influência dessas variáveis nas grandezas aerodinâmicas previstas, ou tais variáveis exercem pouca influência sobre a função objetivo, especialmente considerando as diferenças fundamentais nas formulações teóricas entre o VLM e o N-LLT. Dessa forma, desvios nos resultados das otimizações das duas teorias são esperados.

Por último a otimização completa foi realizada explorando todos os possíveis aerofólios, os resultados podem ser vistos na Tabela 15. Esses resultados foram avaliados utilizando o algoritmo N-LLT original para comparação final. O aerofólio resultante está representado na Figura 40

Tabela 15 – Comparação de resultados das otimizações com 15 parâmetros no N-LLT e Rede Neural.

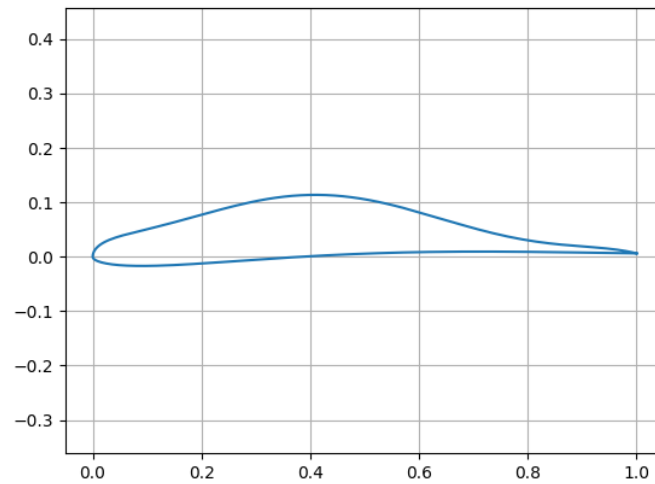
Parâmetro	Rede Neural	Teste N-LLT
AR	19,48	19,48
λ	0,96	0,96
Λ [°]	-12,05	-23,01
θ [°]	-0,17	-2,07
P_1	0,0198	0,0198
P_2	0,0039	0,0039
P_3	0,4094	0,4094
P_4	0,1138	0,1138
P_5	-2,0399	-2,0399
P_6	3,1625	3,1625
P_7	0,4244	0,4244
P_8	-1,1205	-1,1205
P_{10}	0,0062	0,0062
P_{11}	7,3138	7,3138
P_{12}	13,0864	13,0864
Tempo [s]	74,8	—
Func. Objetivo	-60,39	-65,66

Fonte: Elaborado pelo autor.

É importante destacar que, devido à natureza dos dados provenientes do N-LLT presentes no banco de dados, uma parcela significativa dos valores de C_{D0} e K_2 encontra-se muito próxima de zero, em contraste com os valores obtidos via VLM. Como consequência, as otimizações baseadas na N-LLT tendem a produzir pontos nos quais a função objetivo assume magnitudes substancialmente diferentes das observadas na otimização com VLM. Em diversos casos, a função objetivo atinge valores altamente negativos, resultado direto da presença desses coeficientes no denominador, o que amplifica a sensibilidade do cálculo e aumenta a variabilidade dos resultados. Essa característica torna a comparação direta entre os dois modelos particularmente desafiadora.

Diante dessa limitação, realizou-se uma segunda otimização especificamente direcionada a avaliar a capacidade preditiva do modelo N-LLT no regime de estol. Para isso, utilizou-se a formulação apresentada na Equação 12, restringindo o espaço de projeto aos cinco parâmetros selecionados. Os resultados dessa otimização encontram-se compilados na Tabela 16.

Figura 40 – Aerofólio resultante da otimização completa com a rede neural baseada na N-LLT.



Fonte: Elaborado pelo autor.

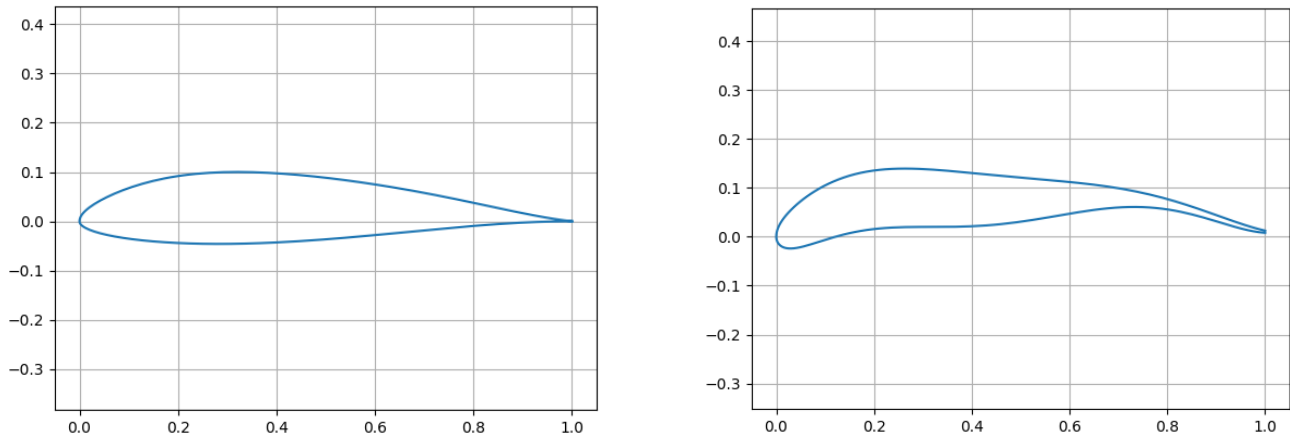
Tabela 16 – Comparação de resultados das otimizações com 5 parâmetros para previsão de estol na N-LLT e Rede Neural.

Parâmetro	N-LLT	Rede Neural	Teste N-LLT
AR	20,0	20,0	20,0
λ	0,2	0,2	0,2
Λ [°]	25,47	-24,31	-24,31
θ [°]	-2,47	-3,88	-3,88
Perfil	1329	1328	1328
Tempo [s]	8952,2	7,4	–
Func. Objetivo	-2,3459	-2,4124	-2,3300

Fonte: Elaborado pelo autor.

A otimização apresentou um comportamento significativamente mais estável e coerente, demonstrando que, quando o espaço de projeto é restringido aos aerofólios incluídos no treinamento, a rede neural apresenta desempenho satisfatório e resultados consistentes com o modelo N-LLT. Por outro lado, otimizações realizadas em um espaço de projeto livre, permitindo formas arbitrárias de aerofólios não presentes no conjunto de treinamento, não produziram resultados confiáveis, revelando limitações do modelo neuronal na extrapolação para geometrias de aerofólio não vistas. Os aerofólios resultantes dessa otimização podem ser visualizados na Figura 41.

Figura 41 – Comparação dos aerofólios do banco de dados PARSEC obtidos nas otimizações realizadas com a N-LLT e sua rede neural. Aerofólio 1329 (esquerda) e 1328 (direita).

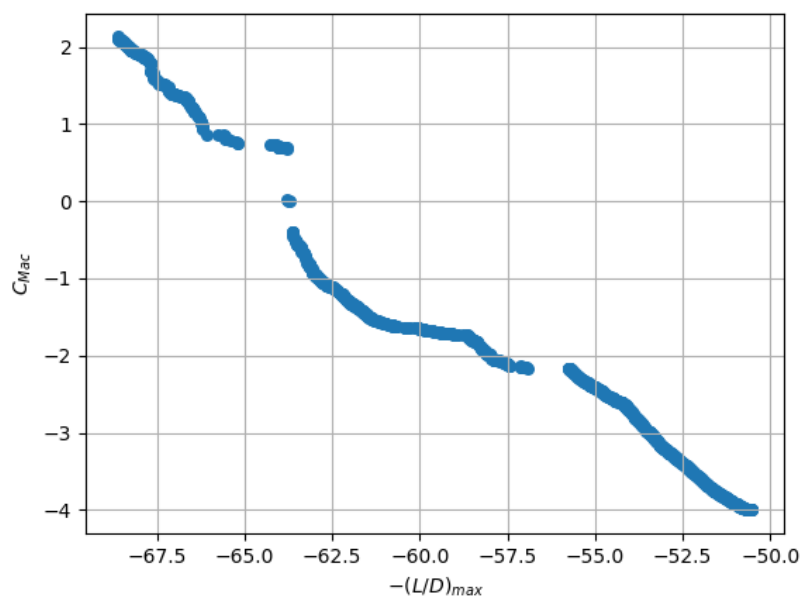


Fonte: Elaborado pelo autor.

5.4 OTIMIZAÇÃO MULTIOBJETIVO

Por fim, foi realizada uma otimização multiobjetivo utilizando o algoritmo genético *NSGA-II*. As Equações 9 e 10 foram utilizadas como funções objetivo, com o intuito de identificar os compromissos (*trade-offs*) existentes entre elas. Como resultado, foi obtida uma fronteira de Pareto, evidenciando os compromissos entre os objetivos considerados, a mesma pode ser vista na Figura 42.

Figura 42 – Fronteira de Pareto obtida a partir da otimização multiobjetivo com a rede VLM utilizando *NSGA-II*.



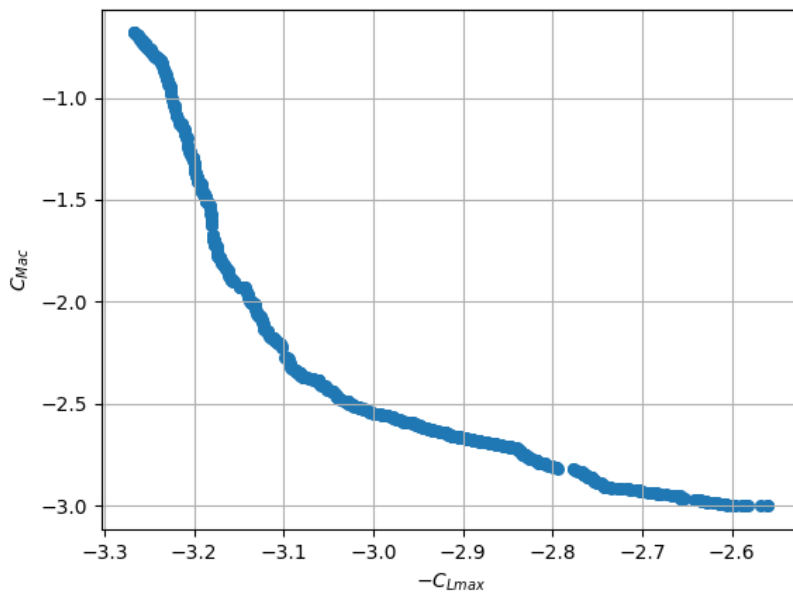
Fonte: Elaborado pelo autor.

Na Figura 42 observa-se uma clara tendência de comprometimento entre os dois objetivos,

evidenciada pela distribuição decrescente dos pontos ao longo da curva. À medida que se busca maximizar a eficiência aerodinâmica (valores mais negativos de $-(L/D)_{max}$), os valores de C_{Mac} tornam-se mais positivos, indicando um aumento na instabilidade longitudinal. Por outro lado, ao favorecer perfis com maior estabilidade (valores mais negativos de C_{Mac}), a eficiência aerodinâmica é prejudicada. Essa relação evidencia a existência de compromissos estruturais no projeto de asas, sendo essencial a escolha de um ponto na fronteira que atenda aos requisitos específicos da missão ou aplicação em questão.

A otimização multiobjetivo da rede N-LLT foi feita utilizando os mesmos algoritmos e os objetivos da Equação 13 resultando na Fronteira de Pareto da Figura 43. A densidade de pontos ao longo de ambas as curvas sugerem que os algoritmos foram eficazes em mapear a superfície de soluções não dominadas, proporcionando uma base sólida para a tomada de decisão no contexto da otimização.

Figura 43 – Fronteira de Pareto obtida a partir da otimização multiobjetivo com a rede N-LLT utilizando NSGA-II.



Fonte: Elaborado pelo autor.

No caso específico da fronteira de Pareto obtida com a rede N-LLT, observa-se uma tendência clara e bem definida entre C_{Lmax} e C_{Mac} , caracterizando o conflito estrutural entre desempenho em altas incidências e estabilidade longitudinal. À medida que os valores de C_{Lmax} aumentam ao longo da fronteira, os coeficientes de momento tornam-se progressivamente menos negativos, indicando que ganhos de sustentação máxima são acompanhados pela redução da estabilidade estática em arfagem.

Em contrapartida, as regiões da fronteira associadas a valores mais negativos de C_{Mac} apresentam menor capacidade de sustentação, refletindo configurações naturalmente mais estáveis, porém aerodinamicamente menos eficientes em ângulos elevados de ataque. A curvatura mais acentuada na extremidade direita da fronteira mostra que incrementos adicionais de C_{Lmax}

exigem sacrifícios cada vez maiores de estabilidade, delimitando o limite prático do espaço de projeto representado pela rede N-LLT.

A distribuição contínua e bem preenchida dos pontos demonstra que o algoritmo NSGA-II explorou o espaço de soluções de forma eficiente, mesmo diante da alta dimensionalidade da otimização. Esses resultados reforçam a consistência interna do modelo N-LLT e sua capacidade de capturar os efeitos combinados entre estol e estabilidade longitudinal.

6 CONCLUSÕES

Os métodos iterativos VLM e N-LLT foram implementados com sucesso, demonstrando capacidade para gerar bancos de dados extensos e consistentes destinados ao treinamento das redes neurais.

As redes neurais desenvolvidas demonstraram alta eficiência ao reduzir significativamente o tempo computacional das otimizações, alcançando economias de até três ordens de grandeza em comparação às simulações diretas. Em diversos casos, os resultados produzidos pelas redes foram bastante próximos daqueles obtidos pelo VLM, evidenciando que o modelo treinado foi capaz de capturar fenômenos aerodinâmicos globais com elevada fidelidade. A rede baseada no VLM mostrou-se particularmente eficaz em reproduzir as relações de compromisso entre C_{Mac} e $(L/D)_{max}$, como observado na fronteira de Pareto gerada, cuja distribuição apresentou comportamento físico coerente, refletindo a competição natural entre eficiência aerodinâmica e estabilidade longitudinal.

De maneira semelhante, a rede treinada com o modelo N-LLT demonstrou elevada capacidade de generalização, reproduzindo tendências aerodinâmicas fundamentais, incluindo a relação entre C_{Lmax} e C_{Mac} ao longo das configurações avaliadas. A fronteira de Pareto correspondente exibiu uma tendência clara e consistente, evidenciando o compromisso entre aumento da sustentação máxima e redução da estabilidade longitudinal. Esse comportamento reforça que o modelo N-LLT, mesmo com maior sensibilidade a não linearidades em condições próximas ao estol, foi devidamente aprendido pela rede neural.

É importante ressaltar que, devido à natureza estocástica dos algoritmos de otimização utilizados, bem como à complexidade das funções objetivo, que frequentemente apresentam múltiplos mínimos locais e potenciais descontinuidades, os resultados tendem a variar a cada execução. Por isso, as otimizações realizadas com as redes neurais nem sempre reproduzem exatamente os mesmos valores obtidos pelos modelos aerodinâmicos originais. Ainda assim, o fato de ambas as redes terem produzido otimizações extremamente próximas das soluções tradicionais demonstra uma capacidade sólida de aprender e generalizar padrões aerodinâmicos. Isso ficou particularmente evidente nas fronteiras de Pareto, cuja distribuição densa e bem estruturada dos pontos não dominados indicou que o algoritmo explorou adequadamente o espaço de soluções relevantes.

Esses resultados reforçam o potencial de algoritmos de aprendizado profundo, como o *Multi-Layer Perceptron*, em replicar métodos iterativos tradicionais, como VLM e N-LLT, funcionando como modelos substitutos em cenários que exigem grande número de simulações. Além de reduzir o custo computacional, essa abordagem possibilitou a realização de otimizações multiobjetivo de alta complexidade, permitindo análises mais abrangentes e decisões mais informadas sobre os compromissos de projeto. Assim, o uso de redes neurais em conjunto com algoritmos evolutivos mostra-se uma ferramenta promissora no contexto das otimizações aeroespaciais, abrindo caminho para o desenvolvimento de arquiteturas mais avançadas de projeto no futuro.

7 SUGESTÕES PARA TRABALHOS FUTUROS

A partir dos resultados obtidos e das limitações inerentes às metodologias utilizadas, diversas extensões podem ser exploradas para aprofundar e ampliar este trabalho. As seguintes sugestões representam possíveis direções de pesquisa futuras:

1. **Implementação de um framework completo de MDO baseado em redes neurais.** Desenvolver uma arquitetura de Otimização Multidisciplinar (*Multidisciplinary Design Optimization*, MDO) utilizando exclusivamente os coeficientes aerodinâmicos, estruturais e de estabilidade estimados pelas redes neurais treinadas neste trabalho. Esse sistema permitiria a avaliação rápida e integrada de desempenho, cargas estruturais, critérios de estabilidade e métricas de eficiência operacional, possibilitando otimizações de alto nível com custo computacional extremamente reduzido;
2. **Integração de CFD para aprimoramento da fidelidade dos bancos aerodinâmicos.** Expandir o processo de geração de dados substituindo ou complementando VLM e N-LLT por simulações CFD de maior fidelidade. Essa abordagem permitiria construir bases de treino mais realistas, especialmente em regimes não lineares, de alto ângulo de ataque, presença de separação ou interações tridimensionais mais complexas.
3. **Avaliação de arquiteturas mais avançadas de redes neurais.** Investigar o uso de arquiteturas modernas, como *Graph Neural Networks* (GNNs), *Transformers* ou métodos de *geometric deep learning*, para capturar relações geométricas e aerodinâmicas mais complexas, reduzindo o erro de predição e aumentando a capacidade de generalização;
4. **Extensão do modelo para asas mais complexas.** Incluir geometrias mais complexas, como asas com torção aerodinâmica, multifuncionais ou com múltiplas seções. Esse avanço permitiria analisar efeitos tridimensionais mais ricos e ampliaria significativamente o escopo de aplicações do método proposto;
5. **Desenvolvimento de uma interface gráfica interativa.** Criar uma ferramenta visual que integre a geração de geometria, simulação aerodinâmica, predição via redes neurais e otimização. Uma interface gráfica permitiria explorar o espaço de projeto em tempo real, ajustar parâmetros geometricamente e observar instantaneamente seus efeitos aerodinâmicos e de desempenho.

REFERÊNCIAS

- AIR TRANSPORT ACTION GROUP. **Waypoint 2050: Aviation — Benefits Beyond Borders**. 2. ed. Geneva, Switzerland, 2021. Disponível em: https://aviationbenefits.org/media/167417/w2050_v2021_27sept_full.pdf. Acesso em: 10 nov. 2025.
- ANDERSON, J. D. J. **Aircraft performance and design**. Boston, USA: McGraw-Hill, 1999. 580 p.
- ANDERSON, J. D. J. **Fundamentals of Aerodynamics**. 5. ed. New York, USA: McGraw-Hill, 2010.
- BÀRCENAS, O. U. E. *et al.* Surrogate aerodynamic wing modeling based on a multilayer perceptron. **Aerospace**, v. 10, n. 2, p. 149, 2023. DOI: 10.3390/aerospace10020149.
- BRUNTON, S. L.; NOACK, B. R.; KOUMOUTSAKOS, P. Machine learning for fluid mechanics. **Annual Review of Fluid Mechanics**, Annual Reviews, v. 52, n. 1, p. 477–508, 2020. DOI: 10.1146/annurev-fluid-010719-060214.
- CARRARI, G. C.; SILVA, G. P. G. da. **Machine Learning for Surrogate Modeling in Multi-Objective Wing Optimization**. [S.l.]: Zenodo, 2025. DOI: 10.5281/zenodo.17904575. Disponível em: <https://doi.org/10.5281/zenodo.17904575>. Acesso em: 11 dez. 2025.
- DEB, K. **Multi-Objective Optimization Using Evolutionary Algorithms**. New York, USA: John Wiley & Sons, 2001. 536 p. ISBN 978-0-471-87339-6.
- DEB, K. *et al.* A fast and elitist multiobjective genetic algorithm: Nsga-ii. **IEEE Transactions on Evolutionary Computation**, v. 6, n. 2, p. 182–197, 2002. DOI: 10.1109/4235.996017.
- DRELA, M. Xfoil: An analysis and design system for low reynolds number airfoils. *In*: MUELLER, T. J. (Ed.). **Low Reynolds Number Aerodynamics**. [S.l.]: Springer Berlin Heidelberg, 1989. p. 1–12. DOI: 10.1007/978-3-642-84010-4_1.
- DRELA, M.; YOUNGREN, H. **AVL – Athena Vortex Lattice β -Version User Primer**. Cambridge, UK, 2024. Disponível em: https://web.mit.edu/drela/Public/web/avl/AVL_User_Primer.pdf. Acesso em: 20 nov. 2025.
- FILIPPONE, A. **Advanced aircraft flight performance**. New York, USA: Cambridge University Press, 2012. v. 34. 664 p. (Cambridge Aerospace Series, v. 34). ISBN 9781107024007.
- GÖSSLING, S.; HUMPE, A. The global scale, distribution and growth of aviation: Implications for climate change. **Global Environmental Change**, Elsevier BV, v. 65, p. 102194, 2020. DOI: 10.1016/j.gloenvcha.2020.102194.
- GUDMUNDSSON, S. **General Aviation Aircraft Design: Applied Methods and Procedures**. Boston, USA: Butterworth-Heinemann, 2013. 1048 p. ISBN 9780123973085.
- HÁJEK, J. Parameterization of airfoils and its application in aerodynamic optimization. *In*: **Proceedings of the 16th Annual Conference of Doctoral Students – WDS**. Prague, Czech Republic: [s.n.], 2007. v. 7, p. 233–240.
- HAYKIN, S. S. **Neural Networks and Learning Machines**. 3. ed. Upper Saddle River, USA: Prentice Hall / Pearson, 2009. ISBN 9780131471399.

HOUGHTON, E. L.; CARPENTER, P. W. **Aerodynamics for Engineering Students**. 5. ed. Oxford, UK: Butterworth-Heinemann, 2003.

INTERNATIONAL AIR TRANSPORT ASSOCIATION. **Aircraft Technology Net Zero Roadmap**. Montreal, Canada, 2023. Disponível em: <https://www.iata.org/contentassets/8d19e716636a47c184e7221c77563c93/aircraft-technology-net-zero-roadmap.pdf>. Acesso em: 10 nov. 2025.

INTERNATIONAL CIVIL AVIATION ORGANIZATION. **Report of the High-Level Meeting on the Feasibility of a Long-Term Aspirational Goal for International Aviation CO₂ Emissions Reductions**. Montreal, Canada, 2022. Disponível em: <https://www.icao.int/sites/default/files/environmental-protection/Documents/DOC.10178.EN.pdf>. Acesso em: 10 nov. 2025.

INTERNATIONAL ENERGY AGENCY. **Transport energy and CO₂: Moving towards sustainability**. Paris, France: OECD Publishing, 2009. Disponível em: https://www.oecd.org/en/publications/transport-energy-and-co2-moving-towards-sustainability_9789264073173-en.html. Acesso em: 10 nov. 2025.

KARALI, H. *et al.* A new nonlinear lifting line method for aerodynamic analysis and deep learning modeling of small unmanned aerial vehicles. **International Journal of Micro Air Vehicles**, v. 13, 2021. DOI: 10.1177/17568293211016817. Disponível em: <https://doi.org/10.1177/17568293211016817>. Acesso em: 23 set. 2025.

KATZ, J.; PLOTKIN, A. **Low-Speed Aerodynamics**. 2. ed. Cambridge, UK: Cambridge University Press, 2001. 630 p.

PHILLIPS, W. F.; SNYDER, D. O. Modern adaptation of prandtl's classic lifting-line theory. **Journal of Aircraft**, v. 37, n. 4, p. 662–670, 2000. DOI: 10.2514/2.2649.

PÉREZ-ENCISO, M.; ZINGARETTI, L. A guide on deep learning for complex trait genomic prediction. **Genes**, v. 10, p. 553, 07 2019. DOI: 10.3390/genes10070553.

RODRIGUES, L. E. M. J. **Introdução ao Projeto de Aeronaves – Conceitos Fundamentais**. 2009. Revista Eletrônica AeroDesign Magazine. Disponível em: <https://irp-cdn.multiscreensite.com/c1ca6eb9/files/uploaded/v1apr1.pdf>. Acesso em: 20 nov. 2025.

SILVA, G. P. G. da; CATALANO, F. M. **Neural network metamodeling for aerodynamic optimization efficiency improvement**. São Carlos: EESC-USP, 2019. Disponível em: <https://soac.eesc.usp.br/index.php/SiPGEM/ivsipgem/paper/view/1715/1037>. Acesso em: 15 ago. 2025.

SUÁREZ, G. *et al.* Nonlinear surrogate model design for aerodynamic dataset generation based on artificial neural networks. **Aerospace**, v. 11, n. 8, p. 607, 2024. DOI: 10.3390/aerospace11080607.

TATAR, U. **Aerodynamic predictions for unmanned aerial vehicle formation flight using vortex lattice method and machine learning**. Dissertação (Mestrado) — Middle East Technical University, 2024. Disponível em: <https://hdl.handle.net/11511/111069>. Acesso em: 20 nov. 2025.

WANG, G. G.; SHAN, S. Review of metamodeling techniques in support of engineering design optimization. **Journal of Mechanical Design**, ASME International, v. 129, n. 4, p. 370–380,

2007. DOI: 10.1115/1.2429697. Disponível em: <https://doi.org/10.1115/1.2429697>. Acesso em: 10 nov. 2025.

WANG, X.; DAMODARAN, M. Aerodynamic shape optimization using computational fluid dynamics and parallel simulated annealing algorithms. **AIAA Journal**, v. 39, n. 8, p. 1500–1508, 2001. DOI: 10.2514/2.1474. Disponível em: <https://doi.org/10.2514/2.1474>. Acesso em: 20 nov. 2025.

WICKENHEISER A. M.; GARCIA, E. Extended nonlinear lifting-line method for aerodynamic modeling of reconfigurable aircraft. **Journal of Aircraft**, v. 48, n. 5, p. 1812–1817, 2011.