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

METODOLOGIA PARSEC DE PARAMETRIZAÇÃO PARA MELHORIA DE PERFIL AERODINÂMICO POR ALGORITMOS PSO

Autor: Rafael Alves da Silva Neto
Orientador: Prof. Dr. Luciano Gonçalves Noleto, UnB/FGA

Brasília, DF
2016

Rafael Alves da Silva Neto

METODOLOGIA PARSEC DE PARAMETRIZAÇÃO PARA MELHORIA DE PERFIL AERODINÂMICO POR ALGORITMOS PSO

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

Universidade de Brasília - UnB

Faculdade UnB Gama - FGA

Orientador: Prof. Dr. Luciano Gonçalves Noleto, UnB/FGA

Coorientador: Prof$^{\underline{a}}$. Dr$^{\underline{a}}$. Josiane do Socorro Aguiar de Souza, UnB/FGA

Brasília, DF

2016

Rafael Alves da Silva Neto
METODOLOGIA PARSEC DE PARAMETRIZAÇÃO PARA MELHORIA DE PERFIL AERODINÂMICO POR ALGORITMOS PSO/ Rafael Alves da Silva Neto. – Brasília, DF, 2016-
102 p. : il. (algumas color.) ; 30 cm.

Orientador: Prof. Dr. Luciano Gonçalves Noleto, UnB/FGA

Trabalho de Conclusão de Curso – Universidade de Brasília - UnB
Faculdade UnB Gama - FGA , 2016.

1. Turbinas Hidrocinéticas,Parametrização, Algoritmos genéticos,Parsec, PSO. 2. Otimização, Aerofólios. I. Prof. Dr. Luciano Gonçalves Noleto, UnB/FGA. II. Universidade de Brasília. III. Faculdade UnB Gama. IV. METODOLOGIA PARSEC DE PARAMETRIZAÇÃO PARA MELHORIA DE PERFIL AERODINÂMICO POR ALGORITMOS PSO

CDU 02:141:005.6

Rafael Alves da Silva Neto

METODOLOGIA PARSEC DE PARAMETRIZAÇÃO PARA MELHORIA DE PERFIL AERODINÂMICO POR ALGORITMOS PSO

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

Trabalho aprovado. Brasília, DF, 30 de Junho de 2016:

Prof. Dr. Luciano Gonçalves Noleto, UnB/FGA
Orientador

Prof.Dr. Jhon Nero Vaz Goulart, UnB/FGA
Convidado 1

Prof.Dr. Fábio Alfaia da Cunha, UnB/FGA
Convidado 2

Brasília, DF
2016

Este trabalho é dedicado às mentes inquietas que almejam um futuro novo,
um futuro além do que é percetível aos olhos.

Agradecimentos

Precisamos sentir o vazio do deserto e o barulho das metrópoles para que entendamos o verdadeiro valor dos nossos esforços, dos nossos amigos, da nossa alma. Por tal, nesse pretexto, devo eu agradecer a todos que me ajudaram alcançar meu objetivo...digo, aproximar-me dele, pois essa grande viagem pelo conhecimento está apenas começando.

Perdi pessoas importantes, ganhei o apoio de outras, mas nunca me deixaram desamparado aqueles que realmente importam.

Deus, meu grande amigo, é quem merece todo o meu valor, e minha família que me proporcionou forças quando eu pensava que não as teria. E de fato, gostaria de agradecer a todo corpo docente do curso de engenharia de energia, pois todos participaram da construção do meu saber. Principalmente ao meu orientador Luciano Gonçalves Noleto e coorientadora Josiane do S. Aguiar que se permitiram ser apoio acadêmico.

Além, quero expressar meus agradecimentos à duas mentes brilhantes que me ajudaram na construção do meu trabalho, bem como do objeto de estudo, sendo eles o professor Antônio C. P. Brasil Junior e o professor Clóvis Campus, que me deram ideias e soluções para problemas inerentes ao campo de atuação que eu escolhi.

Agradeço aos colegas mestrandos do departamento de mecânica da Universidade de Brasília que me tiraram dúvidas e se mostraram solicitos quando requisitados. E especialmente, aos meus amigos que foram meu alívio para momentos fatídicos, e graça para momentos felizes.

Ao governo Federal, nos nome das intituições CNPQ e CAPES, que subsidiaram meu intercâmbio aos Estados Unidos, onde eu passei quase dois anos, alavancando meu espírito acadêmico e pessoal e que levarei para o resto da minha vida.

E a todos que me ajudaram e acompanharam, de perto ou de longe, acreditando em mim e nos meus resultados.

"As pessoas otimizam. Os investidores buscam criar portefólios que evitem excessivos riscos enquanto tentam um grande retorno de capital. A indústria foca na máxima eficiência no desenho e na operação dos processos de produção. Os engenheiros ajustam parâmetros para otimizar a performance dos seus projetos...a natureza se otimiza..."(NOCEDAL; WRIGHT, 2006)

Resumo

O sistema de geração eletromecânica se fundamenta principalmente nas características em que o rotor tem para "extrair" a energia do fluido de trabalho. Com isso, a engenharia de desenvolvimento de pás, como a geometria e sua construção, tem que ser efetivamente consolidadas para o aumento da eficiência global de uma turbina. Assim, este trabalho visa o aprofundamento nos preceitos mecânicos e computacionais inerentes a modelagem das pás, precisamente do hidrofólio da turbina hidrocinética geração 2 construída pelo LEA (Laboratório de Engenharia e Ambiente - UnB). O estudo tem o enfoque em projetar e analisar, por meio de estudos numéricos de parametrização e otimização de perfis, utilizando ferramentas computacionais como MATLAB R2016b e XFOIL, juntamente com o sistema operacional Windows 8. No caso, o estudo do modelo integra a junção da teoria de aerofólios e algoritmos de otimização "PSO", que é implementado de forma a se obter um máximo aproveitamento dos coeficiente aerodinâmicos da pá. Com isso, uma ótima geometria das pás da turbina G2, configurada pela parametrização PARSEC, é encontrada e comparada com resultados obtidos do hidrofólio original, usando o software de análise de perfis XFOIL para atesto do método matemático, a fim de comprovar a eficácia do uso de algoritmos para parametrizar perfis hidrodinâmicos e otimizar suas geometrias.

Palavras-chaves: Modelagem de pás. Turbinas hidrocinéticas. Parametrização e otimização. Parsec. PSO

Abstract

The electromechanical generating system is mainly based on the characteristics wherein the turbine has to "harvest" the energy of a working fluid. Thus, the engineering behind the blades, such as the geometry and its construction, must be effectively consolidated to increase the overall turbine efficiency. This work aims to go further into the mechanical and computational principles of modeling turbine blades, precisely the hydrokinetic turbine hydrofoil built by LEA (Laboratório de Engenharia e Ambiente - UNB). The study has a focus on designing and analyzing, through numerical studies of parameterization and optimization, a turbine blade, using computational tools such as MATLAB R2016b and XFOIL alongside Windows 8, as the operating system. In this case, part of the airfoil theory is combined with optimization algorithms (PSO), which are implemented to get a maximum utilization of the blade related to its aerodynamic coefficient CL over CD. Furthermore, an optimal turbine blade geometry, set by PARSEC parameter, is found and compared with results obtained from the original hydrofoil, using the software of profile analysis XFOIL to certify the mathematical method, proving the effectiveness of algorithms to parameterize hydrodynamic profiles and optimize their geometries.

Key-words: Blade Modeling. Hydrokicnetic Turbines. Parameterization and Optimization. Parsec. PSO.

Lista de ilustrações

Lista de tabelas

Lista de abreviaturas e siglas

CATIA V5R19 *Computer Aided Three-dimensional Interactive Application Version 5 Release 19*

SW *Swept area ou disco atuador*

VC *Volume de controle*

TG2 *Turbina Hidrocinética de geração 2*

PSO *Particle Swarm Optimization*

PARSEC *Parametric Section*

THC's *Turbinas Hidrocinéticas*

NACA *United States National Advisory Committee for Aeronautics*

TSR *Tip Speed Ratio ou Razão de velocidade na ponta da pá*

SR *Speed Ratio ou Razão de Velocidade*

SP *Shape parameter ou Parâmetro de Forma*

PDCA *Plan,Do,Check,Act ou Planeje,Faça,Analise,Produza*

LEA *Laboratório de Engenharia e Ambiente*

AG *Algoritmo Genético*

CFD *Computacional Fluid Dynamics*

RMRB *Representation by Minimium Residual Base*

Lista de símbolos

g	Aceleração pela gravidade
Cd	Coeficiente de arrasto ou coeficiente de dissipação
L_c	Comprimento de corda
N_p	Número de pás
b	Altura da lâmina
s	Espaçamento
Cl	Coeficiente de sustentação
a	Fator de interferência ou indução axial
a'	Fator de interferência ou indução tangencial
b	Envergadura da pá
A_p	Área de pá
L_c	Comprimento de corda
σ_r	Solidez local do rotor
D	Diâmetro
$\mathbf{N_p}$	Número de pás
A	Área
i	Ângulo de Incidência
L	Momento angular
$\dot{m}$	Fluxo de massa
n	Velocidade de rotação em revoluções por minuto
ω	Velocidade angula (rads por segundo)
dE	Empuxo
dT	Torque

CL	Coeficiente de Sustentação
CD	Coeficiente de Arrasto
CM	Coeficiente de Momento
p	Pressão
P	Potência
Q	Quociente de vazão
T	Torque
$\overrightarrow{U}$	Velocidade do frame
$\overrightarrow{V}$	Velocidade absoluta
$\overrightarrow{W}$	Velocidade relativa
$\overrightarrow{F}_t$	Força resultante transversal ou axial
$\overrightarrow{F}_Q$	Força resultante tangencial W Velocidade do escoamento

Sumário

1 Introdução

1.1 Contextualização

Apesar de muitos historiadores afirmarem que a Era Moderna se iniciou com a descoberta das máquinas à vapor e à combustão na primeira revolução industrial, a era tecnológica e de rápido crescimento na qualidade de vida se iniciou de fato com a descoberta do domínio da eletricidade. Embora os fenômenos elétricos têm sido estudados desde a antiguidade, os primeiros avanços na área foram feitos nos séculos XVII e XVIII e moldados a partir da segunda metade do século XIX e começo do século XX. As técnicas de transformação da energia eletromecânica em centrais por meio de dínamos e turbinas foram os primeiros avanços observados, permitindo assim, a explosão de um grande negócio em energia elétrica. Três nomes foram essenciais para esse processo, Thomas Edson, Nicola Tesla e George Westinghouse, sendo este último, criador do primeiro transformador e gerador em corrente alternada em 1887, tornando viável a comercialização e desenvolvimento da eletricidade a grandes centros urbanos e a longas distâncias (DINIZ, 2012). Porém, a centralização das fontes geradoras ao longo da história tornou o mercado saturado e encareceu a energia elétrica, além de que se mantiveram acorrentados aos governos, locações pontuais e grandes obras civis.

Os desenvolvimentos a cerca da geração prosseguiram durante os anos seguintes e resultaram em centenas de modelos para que a geração da energia fosse eficiente. Percebeu-se então que a fonte mais confiável para a produção em larga escala era a hídrica, "inesgotável, limpa e barata". Porém, nesse processo muitos desenvolvimentos, principalmente em relação as turbinas hidráulicas, foram abandonadas diante a grandes projetos em hidrelétricas e interesses econômicos. Contudo, um sistema peculiar veio chamar atenção dos pesquisadores, principalmente pela garantia de abastecimento energético de pequenas comunidades isoladas, além da fácil manutenção e baixo custo de instalação... a turbina hidrocinética (CPFL, 2015).

As turbinas hidrocinéticas trabalham com o escoamento das águas em um rio. Como moinhos d'água, esses dispositivos que utilizam a energia das correntes livres do fluxo do fluido para a "geração"de trabalho, faz dessa ciência uma das mais antigas que a humanidade desenvolveu, cerca de 120 A.C (MEHER-HOMJI, 2000).

Atualmente, dada a importância de uma estruturação descentralizada da capacidade de geração, principalmente pela inviabilidade econômica se mostrar aparente para a instalação de linhas de transmissão de energia elétrica a todos os consumidores, o estudo de fontes alternativas é ponto chave, tanto social quanto de desenvolvimento humano e econômico. Por isso, o emprego das máquinas hidrocinéticas não denota um conceito inovador, mas uma revisão dos modelos existentes, como uma alternativa para a geração sustentável e confiável (JUNIOR et al., 2007).

Apesar dos grandes benefícios, a modelagem e literatura a cerca das THC's não são tão populares e poucos artigos na literatura científica abordam o estudo da arte. Por outro lado, a modelagem matemática associada com teorias de otimização são crescentes em outras turbinas, como as eólicas, e alavancam o interesse nas THC's novamente, especificamente na aplicação em correntes marítimas e em comunidades ribeirinhas, que em princípio de estudo, mantém a mesma linha de modelagem que qualquer outra turbina de fluxo livre, apesar de algumas variáveis serem específicas, como a variabilidade das linhas de corrente e baixas velocidades (JUNIOR, 2010).

Sendo assim, Els et al. (2003) mostra que estruturação e modelagem de uma turbina hidrocinética seguem parâmetros bastantes peculiares a uma turbina eólica convencional, seguindo variáveis como número de pás, escoamento do fluido, tal qual a avaliação dos perfis das pás e cálculos em dinâmica dos fluidos. E nesses preceitos, os levantamentos numéricos computacionais compreendem em uma solução mais rápida e eficiente pois detém da capacidade de atender aos diversos parâmetros a partir de uma base de dados de experimentos antigos, possibilitando o alcance de novos projetos otimizados com o uso de ferramentas matemáticas e programação computacional.

E para a utilização desse método, há a necessidade de dominar a parametrização dos perfis de pá tendo em vista o uso de menor número de parâmetros possíveis para descrever dada geometria em âmbito computacional, que será abordado nesse trabalho. A programação deve com o algoritmo gerar variadas soluções em perfis, sendo que em uma dessas soluções, a configuração ótima deve estar inclusa. Adiante, por um método iterativo, o resultado final é encontrado pela junção entre parametrização, otimização e análise com algoritmos de otimização e simulações no programa XFOIL.

Assim, pelo estudo metodológico, o perfil deste trabalho buscou, a partir do conhecimento previamente documentado na bibliografia, uma forma de explorar a metodologia de parametrização e otimização das pás da THC. Pois, a partir do projeto de pesquisa em extensão, notou-se a necessidade de buscar um produto que satisfizesse a demanda de comunidades isoladas do Sistema Integrado Nacional de Energia (SIN), por energia elétrica. Por tal, a proposta de otimização de uma turbina que integre a teoria e projeto de máquinas hidráulicas é desenvolvida. A principal ferramenta adotada é, como dito, a prospecção de referencial teórico que pudesse embasar o trabalho e com isso fazer um planejamento apurado do que é necessário para o objetivo de construção de um algoritmo computacional que atendesse as especificações ou um máximo rendimento do projeto.

E assim, a partir da conexão entre a teoria, metodologia e planejamento inicial junto ao orientador, teve-se a inicialização dos cálculos referente as pás da turbina bem como dimensionamento hidrodinâmico e formulações matemáticas para a codificação em MATLAB. E por tal, o presente trabalho aborda o uso do método PARSEC e PSO para a otimização das pás de uma THC já existente, além de avaliar do desempenho da turbina hidrocinética otimizada em relação aos seus coeficientes aerodinâmicos.

1.2 Turbinas hidrocinéticas UnB

As primeiras idealizações a cerca do tema na universidade de Brasília foram feitas na década de 80, com o projeto de uma turbina piloto, no departamento de Engenharia Mecânica, exatamente em 1987. Desde então, vários projetos e máquinas foram desenvolvidas por alunos e professores, porém os protótipos que ganharam uma atenção maior foram os ditos "de geração" em parcerias com a ELETRONORTE no projeto PORAQUÊ e TUCUNARÉ. As turbinas eram feitas basicamente por meio de cálculos empíricos, e por meio da "tentativa e erro", sendo que a partir desse processo, houveram aprimoramentos ao longo do desenvolvimento pela técnica de PDCA.

A primeira turbina operacional, chamada de **Geração 1**, na qual os parâmetros podem ser visto na Tab. (1), esteve em trabalho em julho de 1995 em Correntina-BA, com o objetivo de fornecer energia reserva a um posto de saúde da região. Sempre com o intuito de desenvolver uma tecnologia viável e de rendimento considerável, a turbina compreendia parâmetros antes não aplicados, como nesse primeiro projeto, referentes a um rotor axial de duas pás com uma grade cônica de proteção frontal contra destroços flutuantes, e com um estator com pás diretrizes direcionando o fluxo da água que entra na turbina para melhorar o ângulo de ataque na hélice (LEA, 2007).

Tabela 1: Parâmetros de turbina Geração 1

Velocidade de escoamento (m/s)	**Número pás**	**Diâmetro (cm)**	**Coeficiente de solidez**	**Potência (W)**
2	6	80	0.3	1.5

Fonte: (LEA, 2007)

Após a constatação de que o sistema realmente era viável para aplicações em geração distribuídas, o estudo seguiu com a instalação de um difusor na turbina hidrocinética de **Geração 2**. Esta foi instalada no mesmo local em 2005 e em Maracá-AP, em Outubro de 2006. A teoria por trás dessa melhoria é que o difusor gera uma desaceleração do escoamento na saída da turbina criando uma região de baixa pressão, consequentemente, aumentando a velocidade do escoamento na entrada e aumento do coeficiente de potência da turbina.

Sendo assim, com a grande verificação do aumento da eficiência, buscou-se uma máquina axial livre mais compacta, portátil e com um desempenho que atendesse os requisitos crescentes por qualidade energética. Surgiu então a turbina hidrocinética de **Geração 3**. Além de um perfil na superfície interna da carcaça que simula um difusor, o novo protótipo adequa um gerador ao núcleo do rotor, como mostrado na Fig. (1). Verificou-se também, que o uso de processos computacionais para simular a turbina em ambiente virtual era de grande ajuda no projeto final, sendo então marco da utilização dessa ferramenta.

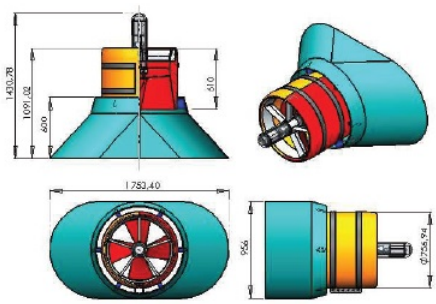


Figura 1: Turbina hidrocinética Geração 3 UnB

Fonte: (LEA, 2007)

Em teoria, essa turbina teria uma eficiência de 90%, contudo como será abordado nas sessões seguintes, o limite de Betz para máquinas de fluxo livre não pode ultrapassar $CP = 0.59$. O novo projeto tenta reduzir custos e tamanho dos difusores agregados, para se alcançar o limite de Betz, fazendo com que se encontre uma abertura entre a carcaça e o mesmo. Uma abertura entre a carcaça e o difusor faz com que haja um controle da camada limite interna, dispensando um longo difusor. O estudo foi empregado em parceria com a Ecole Nationale d'Arts e Metiers (ENSAM) de Paris, França, e abordou simulações numéricas (LEA, 2007).

Mais recente, em parceria entre UnB, escolas pública da região do Gama (CEMI) e CNPQ, criou-se um grupo para alavancar o interesse do público à engenharia. Auto nomeado de Pandoras do Cerrado, em alusão à mitologia grega, objetivou-se a criação de um protótipo de bancada de uma turbina hidrocinética com um estator feito em alumínio e materiais recicláveis (Fig. 2).

As pesquisas sobre THC's, em desenvolvimento na UnB, apresentam características pioneiras no Brasil e no mundo, além de apresentar resultados importantes para o desenvolvimento do uso sustentável da tecnologia. Portanto, mesmo batendo de frente em barreiras teóricas e de eficiência, o trabalho sobre THC's na universidade tenta explorar várias teorias para contornar tais impedimentos. No caso, seja pelo uso de difusores, seja pela inovação do desenho técnico da **geometria** (JUNIOR et al., 2007).

Atualmente, a otimização da geometria das pás do rotor vem sendo discutida em

Figura 2: Turbina hidrocinética de bancada - Pandoras do Cerrado

Fonte: Autor

diversos trabalhos de mestrado, inclusive nesse de graduação. Porém, esses avanços vem sendo trilhados a 15 anos na UnB e rendeu parcerias, em 2004, com a ELETRONORTE em termos de atividade para o desenvolvimento de uma turbina hidrocinética para comunidades isoladas na Amazônia, e ainda configura-se projetos de pesquisas promissores até para a indústria aeronáutica e de propulsão.

1.3 Problema em otimizações

Como já afirmado, as THC's detém de uma capacidade de geração ainda a ser pesquisada. E mesmo depois que a indústria aeronáutica e de turbo propulsores mitigaram durante anos a fórmula exata para o cálculo estrutural de uma pá de turbina, os processos computacionais ainda estão em crescimento e se mostram uma ferramenta importante nos processos de otimização. Existem várias variáveis complexas inclusas na "fórmula", e muito da idealização de um projeto envolve três etapas distintas: uma etapa preliminar, uma prospecção conceitual do problema e a avaliação detalhada do objeto, como abordado por Raymer (1999), que ainda mostra que em questão de otimização, existem duas vertentes fundamentais:

- Projeto inverso: Melhoria pelo erro.
- Otimização numérica: Cálculo numérico por algoritmos.

Porém, durante as pesquisas, muitos dos parâmetros avaliados eram de fato baseado em aprendizagem por erros de projetos anteriores, ou seja eliminavam os problemas a medida em que eles apareciam. Por tal, com a demanda crescente por máquinas mais eficientes e de baixo custo de produção, a otimização computacional passou a vigorar como ponto chave no desenvolvimento de turbinas pelo tempo utilizado e soluções reais (SOMOZA; MACQUART; MAHERI, 2014).

O projeto inverso, como já mencionado, é um processo demorado e bastante experimental. Os protótipos são testados exaustivamente com a finalidade de desenvolver um produto ótimo e que se assemelhe aos requisitos pré-estabelecidos. Atualmente, a indústria vem barrando projetos dessa categoria pelo fato de gastarem muitos recursos e muita força de trabalho. Podemos mencionar também que em alguns casos, para que se tenham processos verificados, a escala dos produtos tem que ser relativamente menor do que o respectivo modelo real, o que traduz em ainda mais erros.

Por outro lado, a otimização numérica traz a vantagem de ser relativamente mais rápida do que o anterior e sem gastos operacionais ou de recursos em grande escala. Pelo uso de algoritmos genéticos, o número de interações é despendido de acordo com uma base progressiva do conhecimento a cerca do produto a ser trabalhado. Contudo, para que a otimização seja efetiva, a implementação de dado produto deve condizer com o modelo empregado e o algoritmo desenvolvido. Com o objetivo de tentar representar, nesse caso um hidrofólio com o mínimo número de parâmetros/variáveis, faz do processo de parametrização importante para que não se utilize processamento computacional desnecessário.

2 Objetivos

2.1 Objetivo Geral

O objetivo geral do presente trabalho é propor uma modelagem para simulação numérica e otimização da pá da turbina hidrocinética de geração II fabricado pelo laboratório de engenharia e ambiente (LEA-UnB) em termos do coeficiente de sustentação e arrasto, visando gerar um hidrofólio para um ganho de potência gerada.

2.2 Objetivos Específicos

A seguir são apresentados os objetivos específicos do presente trabalho de conclusão de curso 2.

- Avaliar o método RMRB PARSEC (*Representation by minimal residual base*) de parametrização: Implementar e entender a metodologia na parametrização de perfis baseada em funções de base. Sendo assim, o desenvolvimento de algoritmo capaz de reduzir o número de parâmetros da representação do perfil sem grandes custos em eficiência e com um pequeno erro para a otimização é almejado. Esse estudo e implementação ajuda a conhecer os erros e efeitos do uso de ferramentas numéricas nos resultados.

- Combinar a metodologia de parametrização anterior com o método BSPLINE, na qual a base dos processos randômicos deste será os coeficientes encontrados na PARSEC. Este passo busca atender as características do processo de otimização.

- Integrar o algoritmo genético para a otimização do perfil (PSO) e o programa XFOIL: Após a implementação da parametrização, um algoritmo genético para otimizar o perfil selecionado é buscado. O algoritmo trabalhará de forma iterativa, selecionando de uma gama de funções base e processos randomicos o melhor design de pá, designando como objetivo as melhores características para Cd e Cl e assim otimizar a eficiência da pá.

- Comparação de polares: Por fim, apresentar a modelagem parcialmente completa da pá da turbina hidrocinética a fim de se obter um modelo para simulação no software ANSYS CFX (coordenadas cartesianas ótimas da pá). Além, verificar e comparar o resultado obtido computacionalmente com o perfil já usado na THC G2 do LEA, em termos de comparações aerodinâmicas.

2.2.1 Etapas da pesquisa

O resumo metodológico proporcionou um aprofundamento no método científico de forma que as etapas a serem trabalhadas fossem claramente organizadas. As etapas de pesquisa aqui usadas basearam em questões bem definidas para que os resultados fossem condizentes com a proposta. A partir de um projeto de extensão desenvolvido em parceria entre UnB e a comunidade público-privada, a proposta de otimização e construção de uma turbina alavancou os estudos. Assim, um detalhamento em fluxograma é apresentado na Fig. (3).

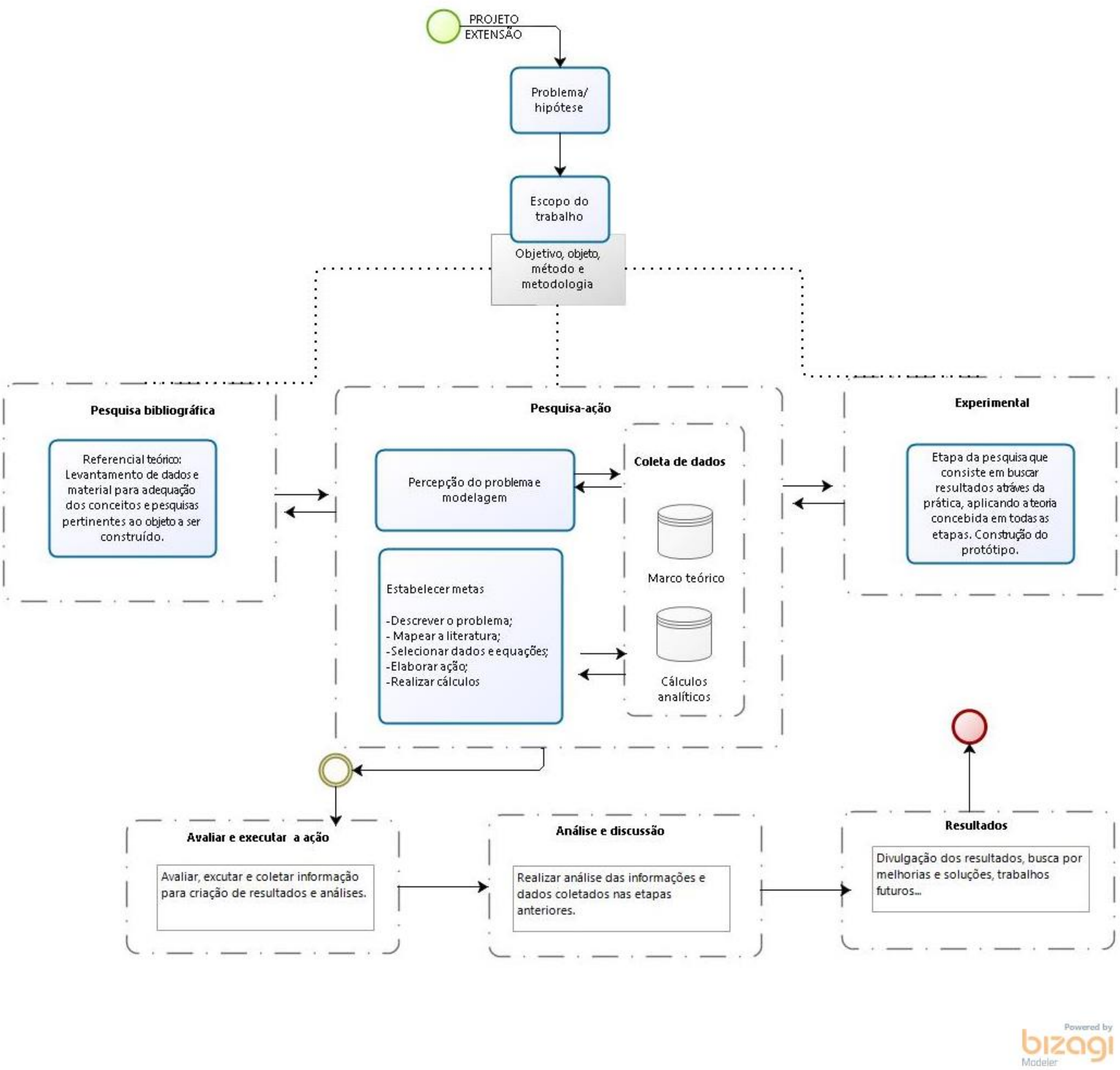


Figura 3: Etapas do plano metodológico.

Fonte: Autor

Além, as bases fundamentais para a construção do projeto seguiu um planejamento linear bastante utilizado na indústria de softwares. Essa linearização das etapas de trabalho fazem com que o estudo dos processos sejam bastante aproveitadas e bem compreendidas.

- Implementar o projeto adentrando no caráter físico da turbina;
- Selecionar as possíveis metodologias de parametrização e otimização do produto;
- Analisar e buscar possíveis softwares de validações físicas para integrar o escopo;
- Concatenar a parametrização com a otimização;
- Analisar projeto conceitual;
- Estabelecer conclusões finais;

2.3 Organização textual

Os capítulos que compreenderão esse trabalho seguem uma ordem de escala. Ou seja, para chegar ao resultado requerido, um apanhado geral teórico é empregado como base do desenvolvimento que irá se afunilando até o objetivo final, como pode ser melhor visualizado na Fig. (4).

1. Estudo sobre hidrofólios: Esse capítulo busca fazer um apanhado teórico e contextual sobre hidrofólios/aerofólios. Um estudo do cenário é importante para localizar os principais objetivos e situar o conhecimento a cerca do tema proposto. Sendo assim, tomando como base o afunilamento das ideias, adentrar na teoria do tema proposto traz ao contexto termos e definições que serão tratados ao longo do trabalho. Assim como, caracterizar as THC's e principais vantagens e desvantagens da mesma. A mecanicidade do fluido (água) no sistema a ser desenvolvido é um dos principais problemas a ser modelado, mesmo em ambiente computacional. Termos e compreensão física dos acontecimentos na turbina, mesmo que ela seja idealizada em muitos aspectos, para uma melhor otimização se faz necessários.
2. Estado da Arte: Nesse capítulo haverá um retorno à teoria apresentada anteriormente, porém voltada para o projeto em si. A conceitualização, tal qual as especificações de projeto serão concretizadas nesse passo para que os cálculos e estudos subsequentes estejam adequados. Além de uma retomada ao objeto a ser estudado como um todo e a teoria por trás da modelagem numérica.
3. Algoritmos de otimização: A partir desse ponto, o levantamento teórico já estará de certa forma concluído sobre o tema. O trabalho então se constituirá em estabelecer o caminho para o objetivo, bem quais os cálculos inerentes e análise do algoritmo

de otimização e parametrização. O objetivo, com isso, é o retorno de resultados e um roteiro de construção do ponto ótimo da pá da turbina.

4. Processo de modelagem e otimização: Nesse capítulo é apresentado o estudo e a comparação com o retorno do objetivo da parametrização junto a turbina real. O método RMRB, desenvolvido pelo LEA será um dos objetos de estudo, pois todo o desenvolvimento será baseado na confiabilidade em que a parametrização se estabelece. Há o adentramento nas parametrizações BSpline, bem como no processo de otimização por algoritmos genéticos e PSO.

5. Resultados da otimização: Nesse capítulo há a integração da literatura com os resultados obtidos, bem como a comparação e a análise dos resultados.

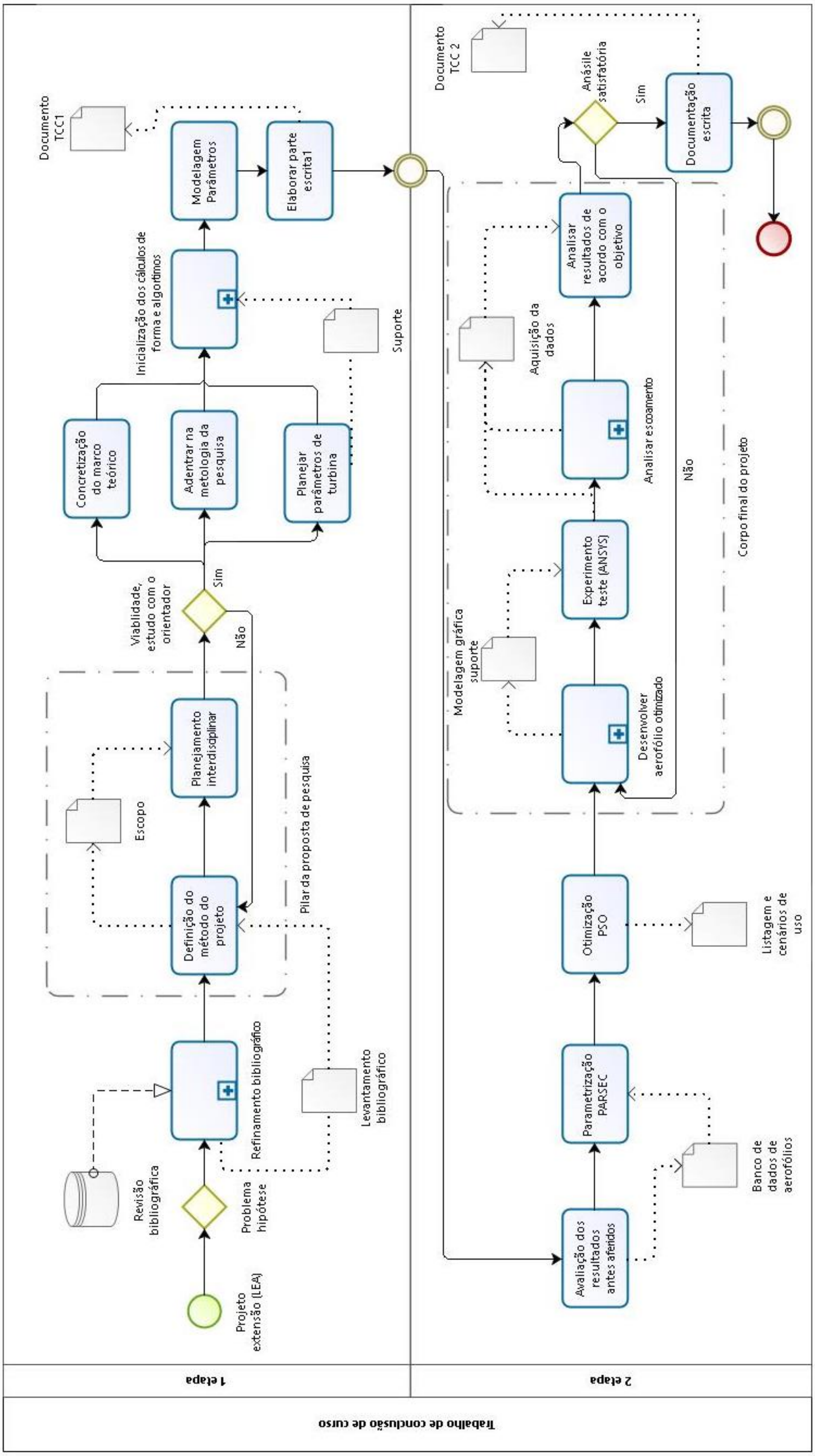


Figura 4: Modelagem do processo de otimização PARSEC/PSO.

3 Hidrofólios: Descrição do problema

Os hidrofólios são componentes principais das turbinas hidrocinéticas, pois são os dispositivos que "trocam e modelam" a energia com o meio, utilizando a diferença no gradiente de pressão para gerar uma força de sustentação positiva no seu perfil, a fim de que esse gere torque, e consequentemente, trabalho (GRANT, 2009). Em termos teóricos, os hidrofólios são aerofólios empregados para ambientes onde o fluido de trabalho é a água.

Os primeiros estudos a cerca do tema para aplicações computacionais foram desenvolvidos por Nikolay Zhukovsky. Ele explorou o cálculo matemático pelas variáveis complexas e o fluxo potencial, na qual qualquer fluxo bi-dimensional poderia ser representada por uma função analítica de uma variável complexa, ou basicamente, representar um círculo no plano complexo, como se fosse um aerofólio no plano cartesiano (BERTIN; SMITH, 1998). Não obstante, vários processos ou modelagens computacionais, utilizam dessa teoria para avaliação dos projetos, chamado principalmente de modelagem em *"panels"* ou painéis. Entretanto, todas as formulações matemáticas levam para a adequar seleção de um perfil aerodinâmico em três características (verificadas no Tab. 2).

Tabela 2: Características fundamentais para aerofólios

Modelagem de aerofólios
1) Determinação da capacidade de sustentação
2) Determinação da correspondente força de arrasto
3) Determinação do momento resultante ao redor do centro aerodinâmico

E para que a base do sistema funcione, o conjunto de pás, ou seja, a turbina, deve priorizar as várias forças atuantes em conjunto, principalmente as ditas aerodinâmicas, que atuam conforme as características estruturais do projeto.

Dois parâmetros são principais, como abordado no quadro, as forças de sustentação (lift) que são aquelas perpendiculares a direção do escoamento do fluido de trabalho no perfil, e a de arrasto (drag), paralelo ao escoamento, e que diminuem a eficiência do processo. Todas esses parâmetros seguem principalmente a geometria das pás e o ângulo de ataque que traduzem o movimento do sistema em três subconjuntos, ou três velocidades - velocidade relativa, absoluta e tangencial (BURTON et al., 2001). Conceitos assim configuram o design característico de objetos que lidam com o escoamento de um fluido, e que juntamente com a teoria dos triângulos de velocidade, ajudam no projeto de qualquer turbina.

Os parâmetros estruturais também estabelecem grande valor na determinação dessas três características. A corda do aerofólio é a linha que liga suas extremidades (bordo de fuga e de ataque). A linha de esqueleto, mostrada também na Fig. (5), define a distribuiçãode arqueamento da linha de corda, na qual se encontra o arqueamento máximo e que é a maior diferença entre corda e o esqueleto.

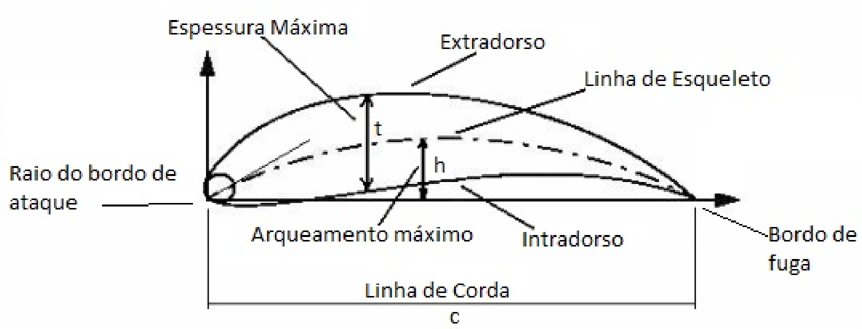


Figura 5: Características básicas de um perfil de pá.

Fonte: Autor

Outra característica importante na determinação do perfil de um hidrofólio é o raio de curvatura do bordo de ataque, inclusive que determina o primeiro parâmetro das coordenadas polares e PARSEC, desenvolvidos nesse trabalho. Esse raio é localizado na linha tangente à linha de esqueleto, e depende principalmente, da distribuição de espessura que é sobreposta à linha de esqueleto, de forma que metade da distribuição gera o extradorso e a outra metade gera o intradorso.

E por fim, a maior distância entre o extradorso e o intradorso é a espessura máxima do hidrofólio. Em testes, verificou-se que algumas pás fomentavam um desempenho melhor em baixas velocidades e baixo ângulo de ataque, porém dependiam de duas variáveis básicas que juntas são chamadas de razão de espectro para envergadura infinita, mostrada na Eq. (3.1) : A envergadura (b) e o comprimento de corda (L_c).

$$RA = \frac{b^2}{\mathrm{A_p}} = \frac{b}{L_c} \tag{3.1}$$

Uma razão elevada representa uma pá de grande envergadura e com uma corda pequena, e vice-versa. Modificando essa relação é possível reduzir o arrasto induzido na pá, além de viabilizar limites construtivos.

Com isso, na maioria dos processos de projeto e fabricação, os engenheiros atuam nos parâmetros apresentados, para a maximização do desempenho aerodinâmico e consequentemente a eficiência global do sistema. É possível fazer uma análise numérica sob a implementação de um modelo matemático para que a otimização seja mais rápida e relativamente eficiente em comparação com análises reais. Sendo assim, devidos a estes motivos, o presente trabalho busca determinar a melhor geometria da pá da turbina hidrocinética G2 pelo uso de algoritmos genéticos pela análises numéricas.

3.1 Máquinas de fluxo livre axial

Apesar de os estudos desse trabalho deterem de uma característica computacional, o desenvolvimento e percepção dos parâmetros os quais a turbina se insere se faz de grande relevância para o entendimento do processo de otimização, e até mesmo físico. Grande parte dos cálculos é baseado nas teorias de Kutta-Joukousky, método dos painéis, fluxo potencial, distribuição linear de vórtices e resolução de equações completas de Navier-Stokes. Contudo, elas são geridas indiretamente no presente trabalho, já que não configuram o objeto de estudo. Assim, esse capítulo busca fazer uma introdução a teoria por trás desse entendimento, na qual o software XFoil (caixa-preta) gere os resultados.

3.1.1 *Swept Area* ou disco atuador

A maneira como o sistema se comporta durante uma transformação de energia é complexa, principalmente com a utilização de turbinas. Contudo, o modelo teórico não leva em conta todas as variáveis inerente ao processo como atrito, ruído, transiência no fluxo do fluido. Mesmo assim, é possível analisar o processo de conversão apenas levando em consideração o balanço energético e balanço de massa do sistema. Para isso, modelos que simplificam essa configuração são aplicados, sendo que para essa análise, o conceito de **SW** é usado (Fig. 6).

É fato verificar que esse modelo equivale ao modelo do volume de controle, considerando um número infinito de pás (SILVA, 2013).

A área transversal infinitesimalmente após a montante do disco atuador é maior que a área do próprio disco e maior que a área anterior a jusante. A expansão do **VC** se dá pela verificação de que a vazão mássica é a mesma para toda região, mas a energia diminui após a passagem pela turbina. A massa de água que passa por uma sessão transversal de área em determinado período de tempo é caracterizado como "$\rho \mathbf{A}_s \mathbf{U}$ ", onde ρ é densidade do fluido, A_s é a área transversal e U é a velocidade do escoamento (BURTON et al., 2001). A vazão mássica para qualquer ponto ao longo do **VC** deve obedecer a condição de continuidade, como:

$$\rho \mathbf{A}_\infty \mathbf{U}_\infty = \rho \mathbf{A}_D \mathbf{U}_D + \rho \mathbf{A}_W \mathbf{U}_W \tag{3.2}$$

O disco atuador faz com que haja uma variação induzida de velocidade que deve ser imposta na velocidade do fluxo. Essa variação a qual modifica o escoamento é dado

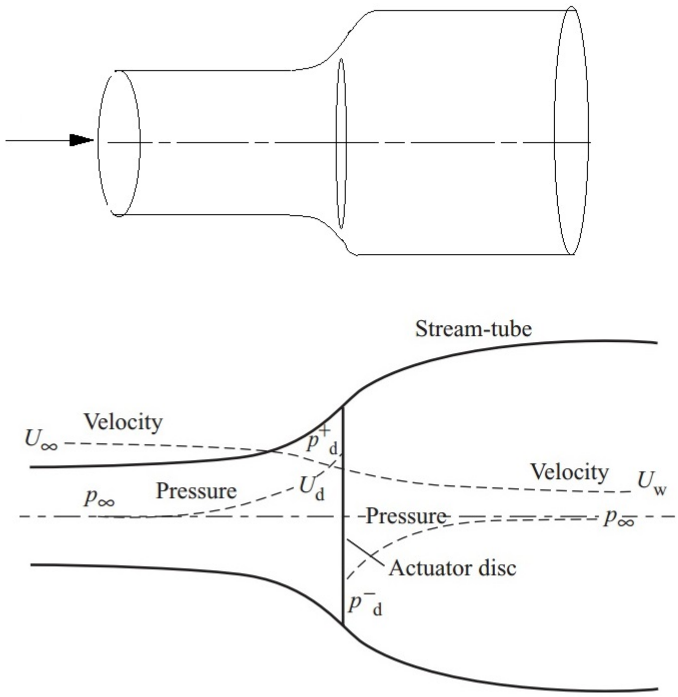


Figura 6: Extração de Energia no Disco atuador.

Fonte: (BURTON et al., 2001)

por $-a\mathbf{U}_\infty$, onde a é chamado de fator de indução de fluxo axial. Portanto, no disco atuador, podemos relacionar a velocidade como sendo:

$$\mathbf{U}_D = \mathbf{U}_\infty(1-a) \tag{3.3}$$

Analisando a Fig. (6), pode-se constatar que o fluido que passa pelo disco atuador sofre uma expansão e uma variação de velocidade $\mathbf{U}_\infty - \mathbf{U}_w$, e com isso uma taxa de variação na quantidade de movimento de:

$$\delta\mathbf{M} = (\mathbf{U}_\infty - \mathbf{U}_W)\rho\mathbf{A}_D\mathbf{U}_D \tag{3.4}$$

Sendo assim, a força é ocasionada pelas diferenças de pressão, pois o ar que circunscreve o disco se mantém em pressão atmosférica, tendo resultante nula. Com isso, relacionando a Eq. (3.4) em termos da variação de pressão, temos que:

$$(\mathbf{p^+}_D - \mathbf{p^-}_D)\mathbf{A}_D = (\mathbf{U}_\infty - \mathbf{U}_W)\rho\mathbf{A}_D\mathbf{U}_\infty(1-a) \tag{3.5}$$

E pela Eq. (3.5), sabendo que $\mathbf{U}_W = -a\mathbf{U}_\infty$, a força pode ser derivada como:

$$\mathbf{F} = (\mathbf{p^+}_D - \mathbf{p^-}_D)\mathbf{A}_D = 2\rho\mathbf{A}_D\mathbf{U}^2_\infty a(1-a) \tag{3.6}$$

Como a força descrita se concentra no disco atuador, a taxa de trabalho é de acordo com F$\mathbf{U}_D$:

$$\mathbf{P} = \mathbf{F}\mathbf{U}_D = (\mathbf{p^+}_D - \mathbf{p^-}_D)\mathbf{A}_D\mathbf{U}_D = 2\rho\mathbf{A}_D\mathbf{U}^3_\infty a(1-a)^2 \tag{3.7}$$

O coeficiente de potência então se formula por:

$$\mathbf{C}_P = \frac{\mathbf{P}}{1/2\rho\mathbf{A}_D\mathbf{U}^3_\infty} \tag{3.8}$$

Utilizando que o denominador é a representação da potência disponível do fluxo sem o disco atuador e o numerador, a potência retirada do mesmo podemos chegar no coeficiente de potência do sistema. Ou seja, desenvolvendo a equação pelas 3.7 e 3.8, têm-se:

$$\mathbf{C}_P = 4a(1-a)^2 \tag{3.9}$$

Definiu-se a potência e o torque pelas Eq. (3.6) e Eq. (3.7), porém em termos de velocidades e diferencial de pressão além do rotor. Outra forma de se determinar é pela característica do rotor, pela velocidade angular ω e o novo termo, fator de indução tangencial $\mathbf{a}'$.

$$\delta\mathbf{T} = \rho\delta\mathbf{A}_D\mathbf{U}_\infty(1-a)2\omega\mathbf{a}'\mathbf{r^2} \tag{3.10}$$

No qual A_D é a área do anel.

E por fim, a partir da Eq. (3.10), podemos formalizar a potência extraída da pá como sendo:

$$P = \int_0^R \rho(2\pi\mathrm{r})\mathbf{U}_\infty(1-a)2\omega^2\mathbf{a}'\mathbf{r}^2 dr \tag{3.11}$$

Deriva-se então a Eq. (3.9), ou resolvendo a Eq. (3.11) para encontrar o máximo:

$$\frac{\mathbf{C}_P}{da} = 4(1-a)(1-3a) = 0 \tag{3.12}$$

$$\mathbf{C}_{Pmax} = \tfrac{16}{27}$$

Esse quociente é conhecido como o coeficiente de Betz que define a proporção de eficiência máxima que máquinas de fluxo livre pode tirar de um escoamento ideal.

3.1.2 O limite de Betz

A mecanicidade de todo algoritmo ou programação na área estudada tem como base os preceitos numéricos e físicos. Para que se possa recriar "o mundo real" a base matemática deve estar sólida e o entendimento do mesmo ajuda a entender como "o mundo virtual" se integra nessa relação. E nesse contexto, leva-se em conta que a conversão de energia nunca poderá ser plena pelos preceitos termodinâmicos e físicos, ainda mais porque várias variáveis influenciam no resultado. Não é possível aproveitar toda a energia do fluxo de um fluido, pois a velocidade da mesma não pode ser zero após o rotor. Sendo assim, tem-se um limite máximo teórico que não é igual ao disponível. O limite de potência a ser extraído por máquinas de fluxo livre foi quantificado pelo físico alemão Albert Betz que deu nome ao coeficiente, tal como, fazendo uma analogia com o limite a ser extraído em máquinas térmicas, o Carnot (BURTON et al., 2001). Como verificado, o limite máximo teórico é de 59% da energia presente no meio, então caracterizar e otimizar as variáveis que limitam o ganho de potência, como o hidrofólio, é necessário no processo de otimização, principalmente lidando com coeficiente aerodinâmicos do mesmo.

3.2 Estudos aerodinâmicos

Máquinas de fluxo geralmente são muito complexas com um princípio de funcionamento básico, o rotor extrai a energia do fluido. Com isso, o estudo das características base é chave crítica no dimensionamento e formulação de projetos ótimos. As forças atuantes em uma turbina são basicamente representadas na Fig. (7) abaixo.

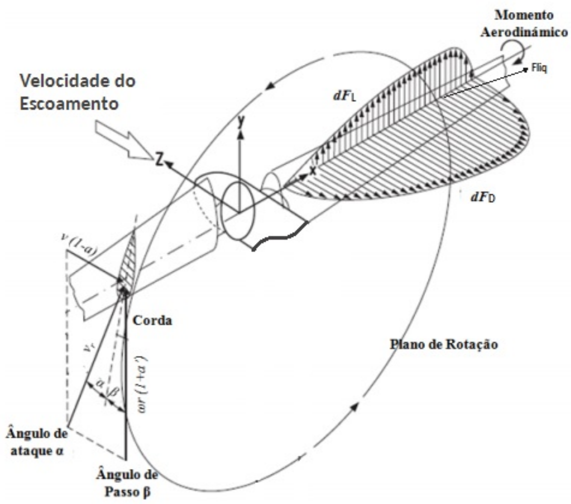


Figura 7: Forças de arrasto e sustentação em uma turbina.

Adaptado de Hau (2005)

Existem duas razões pelas quais uma turbina extrai energia dos fluidos que serão abordadas nesse capítulo:

1. A terceira lei de Newton afirma que para toda ação existe uma reação oposta de igual magnitude. Nesse conceito, a ação da água empurrando as pás do rotor fazem com que as mesmas sofram deflexão seguindo uma rota contrária ao escoamento. Se as pás não tiver uma angulação, elas apenas sofreram um estresse contrário a força do líquido, de forma axial, ou seja, empurradas no mesmo sentido do eixo do fluxo. Contudo, o ângulo de ataque (que será demonstrado) faz com que a água seja desviada em uma direção tangencial, produzindo uma componente de força de mesmo sentido.

2. O efeito de Bernoulli mostra que quanto mais rápido um fluido se move, menor é a pressão estática do mesmo. As pás de uma turbina são desenhadas para que uma unidade infinitesimal do fluido contorne o perfil hidrodinâmico mais rápido em um lado da pá do que em outro. Com isso, forças se configuram de forma a criar sustentação pela diferença de pressão entre "as faces"da pá.

E nesse contexto, o arrasto, comumente chamado de resistência do fluido ao movimento, é a força que atua contra o movimento das pás de qualquer turbina, fazendo que elas diminuam sua velocidade. Ele é uma característica primordial para qualquer objeto que se move e/ou rotaciona na água ou no ar. Com isso, quanto menos arrasto, maior é o coeficiente de potência de uma máquina geratriz. As turbinas devem ser projetadas de forma que sua linha do perfil seja muito eficiente, cortando o escoamento e mudando o ângulo das pás, geralmente construídas entre 5-20 graus, tendem em a ter menos arrasto do que ângulos maiores. O arrasto também aumenta com o aumento da velocidade do escoamento, conceito que é realmente importante para turbinas, pois as pás destas estão se movendo em rotações consideráveis (BERTIN; SMITH, 1998).

Em contra-partida, a força de sustentação se opõe a força de arrasto, fazendo com que as pás da turbina se movimentem, e com isso, quanto mais sustentação melhor o design das turbinas. Existem vários fatores que caracterizam e modificam essas forças, como o perfil, a velocidade, o ângulo de ataque e o torque requerido (KIDWIND, 2013):

1. Perfil: No design das pás, pequenas variações estruturais podem causar efeitos indesejáveis na potência. Geralmente uma base de dados para a construção de novos perfis é requerida, e nesse quesito o centro de pesquisas aeroespaciais (NACA), é um dos mais utilizados.

2. Velocidade: O estudo do triângulo de velocidade é fundamental para o cálculo de sustentação de uma pá. Com maiores velocidades, maior é o arrasto induzido e o estudo das interações das velocidades relativas e absolutas define os parâmetros de construção da pá.

3. Ângulo: O ângulo dita a magnitude da força de sustentação. E ele pode variar de acordo com o desejo do operador, além de ser ponto crucial para pontos como perda de sustentação pelo deslocamento da camada limite. Para a maioria dos aerofólios,o ângulo de ataque varia entre 5° a 20°, na qual varia entre a raíz a ponta da pá. A medida que se desloca a partir do ponto central da turbina, o ângulo decresce pois as velocidades na ponta da pá são maiores o que acarreta maiores perdas.

4. Torque: Esse variável é responsável pela "transmissão"da energia do escoamento do fluido para o eixo. Grandes projetos, requerem grandes pás para que o torque e a potência seja a máxima possível. Porém, aumentando as pás, aumenta-se as forças de arrasto e vórtices induzidos na esteira, consequentemente, perdas.

Com isso, alinhar todas essas variáveis, bem como otimiza-las, configura-se em preceitos fundamentais no estudo de perfis aerodinâmicos.

3.2.1 Torque e forças atuantes em elemento de pá

A força líquida, ou força atuante no sentido axial e tangencial, é o resultante vetorial da força de sustentação e a força de arrasto. Quanto maior a magnitude dessa força melhor o desempenho hidrodinâmico de uma turbina hidráulica. As forças axiais e tangenciais aerodinâmicas são expostas pelas Eqs. (3.13) e (3.14), já demonstradas na Fig. (7):

$$\vec{F_L} = \vec{L}\cos\phi + \vec{D}\sin\phi \tag{3.13}$$

$$\vec{F_D} = \vec{L}\sin\phi - \vec{D}\cos\phi \tag{3.14}$$

Nas quais, L representa a força de sustentação, D a força de arrasto. O ângulo ϕ representa a soma de α e β no triângulo de velocidades, e $\overrightarrow{W}$ a velocidade relativa do fluxo nas equações:

$$L = \frac{C_L \rho \mathbf{A} \vec{W}^2}{2} \tag{3.15}$$

$$D = \frac{C_D \rho \mathbf{A} \vec{W}^2}{2} \tag{3.16}$$

Os coeficientes de arrasto e de sustentação definidos nas Eqs. (3.15) e (3.16) são características do aerofólio estudado que são encontrados de acordo com a função do ângulo de ataque 'α' e o número de Reynolds (EGGLESTON; STODDARD, 1987). A partir do estudo dessa configuração de forças, pode-se projetar as melhores características da turbina para as condições de trabalho da região. Nessa premissa, voltando aos conceitos hidrodinâmicos se 'α' é pequeno, haverá uma sobrepressão na parte na parte inferior do perfil e uma depressão na parte superior que favorecerá a sustentação.

O perfil (aerofólio) possui em sua nomeclatura uma região chamada de intradorso, que o perfil de água se encontra sobrepressão e o extradorso que se encontram em depressão. Em baixo ângulo de ataque, a superfície superior sofre um gradiente de pressão adverso, porém ainda não suficiente para uma separação da camada limite, sendo que o arrasto é pequeno e a sustentação em geral ótima. Para grandes ângulos, o gradiente de pressão na superfície superior cresce, sendo que $\alpha = 15$ a 20 graus é o limite estando o escoamente completamente separado ou "estolado". A sustentação então cai acentuadamente e o arrasto aumenta (WHITE, 2010).

Contudo, para uma caracterização do escoamento e forças hidrodinâmicas, antes deve-se realizar a caracterização do escoamento pelo número de Reynolds. Para tal, deve-se especificar o "grau de turbulência" pela Eq. (3.17) para configuração do processo computacional.

$$R_e = \frac{\rho \mathrm{vL}}{\mu} \tag{3.17}$$

A força resultante líquida já mencionada é a força aplicada no centro hidrodinâmico da pá, ou seja somatória das resultantes de arrasto e sustentação. Podemos então pelas equações determinar a magnitude do vetor da Força liquida, que é a força resultante da componente tangencial (sustentação) e a componente axial (arrasto), como sendo:

$$F_{Liq} = \frac{C_W \rho \mathrm{AW}^2}{2} \tag{3.18}$$

Sendo que $\mathrm{C_W}$ é o resultado da combinação entre o coeficiente de arrasto e o coeficiente de sustentação, também chamado de coeficiente de resistência. Em termos gerais podemos mostrar $\mathrm{C_W}$ como:

$$F_{Liq}^2 = \mathrm{F_L}^2 + \mathrm{F_D}^2 \tag{3.19}$$

$$F_{Liq}^2 = \frac{C_L \rho \mathbf{A} \vec{W}^2}{2}^2 + \frac{C_D \rho \mathbf{A} \vec{W}^2}{2}^2 \tag{3.20}$$

$$F_{Liq} = \frac{(C_L^2 + C_D^2)^{1/2} \rho \mathbf{A} \vec{W}^2}{2} \tag{3.21}$$

Pelos cálculos até então apresentados, o módulo da velocidade relativa "W" presente nas equações e referente ao triângulo de velocidade é expresso como:

$$\mathbf{W} = \sqrt{U_\infty^2 (1-a)^2 + \mathrm{r}^2 \omega^2 (1+a')^2} \tag{3.22}$$

Sendo que,

$$\sin\phi = \frac{U_\infty(1-a)}{W} ...e... \cos\phi = \frac{\mathrm{r}\omega(1+a')}{W} \tag{3.23}$$

Além de que o ângulo de ataque é disposto como:

$$\alpha = \phi - \beta \tag{3.24}$$

Em sentido axial ao rotor, a velocidade absoluta da água sobre a pá é escrita como $Vx = U_\infty(1-a)$, enquanto no sentido tangencial é dado por $Vt = \omega\mathrm{r}(1-a')$, tomando em conta um elemento infinitesimal da pá. As forças de sustentação e arrasto, dF_L e dF_D respectivamente, são agora representados nas Eqs. (3.25) e (3.26) como:

$$dF_L = 1/2\rho\mathrm{W}^2\mathrm{C_L}(\mathrm{L_c}\mathrm{dr}) \tag{3.25}$$

$$dF_D = 1/2\rho\mathrm{W}^2\mathrm{C_D}(\mathrm{L_c}\mathrm{dr}) \tag{3.26}$$

No qual "Lc"é representado como sendo a corda das pás. E assim, pode-se representar os esforços axiais/empuxo "E", mostrado na Eq. (3.27), em um perfil:

$$dE = \mathbf{N_p}(d\vec{L}\cos\phi + d\vec{D}\sin\phi) = \frac{1}{2}\rho\mathrm{W}^2\mathbf{N_p}\mathbf{L_c}(C_L\cos\phi + C_D\sin\phi)\mathrm{dr} \tag{3.27}$$

E a força no sentindo tangencial, responsável pelo torque e consequentemente pela rotação ω, é dado pela Eq. (3.28). É fato observar que esta equação representa a forma de cálculo do torque pelos coeficientes de sustentação e arrasto, enquanto a Eq. (3.10) pela energia cinética.

$$dT = \mathbf{N_p}(d\vec{L}\sin\phi - d\vec{D}\cos\phi)\mathrm{r} = \frac{1}{2}\rho\mathrm{W}^2(\mathbf{N_p}\mathbf{L_c})\mathbf{r}(C_L\sin\phi - C_D\cos\phi)\mathrm{dr} \tag{3.28}$$

Sendo $\mathbf{N_p}$ o número de pás.

Todas esses equacionamentos são parte da teoria de aerofólios, precisamente do estudo do BEM (Blade Element Momentum). Elas ditam a dinâmica dos fluidos pela modelagem matemática, bem como estrutura a física por trás do estudo aerodinâmico.

3.2.2 Correção de Glauer: Fatores de Interferência ou indução (a e a')

Até então, pelos cálculos estabelecidos, fora introduzido termos ainda não explorados nesse estudo, mas que é fundamental para os cálculos de esforços e características hidrodinâmicas, o fator de interferência. Este fator adimensional estabelece a relação da mudança na velocidade axial e tangencial do fluido em uma turbina de fluxo livre. O fator (a) representa a redução de velocidade pelo aumento da pressão na região do disco atuador, enquanto (a′) representa o fator de mudança na direção do escoamento incidente, que adquire uma componente tangencial devido a redução de pressão na saída (GONZALEZ, 2012). Ambos coeficientes variam em uma faixa de 0 a 1, sendo eles que determinam a variação de velocidade do fluxo.

A determinação desses fatores é algo bastante complexo, pois as equações demonstradas não podem ser resolvidas analiticamente, mas resolvidas de forma iterativa por serem dependentes de variáveis como ϕ, C_L e C_D.

Muitos dos pesquisadores nesse trabalho, como Hau (2005), Gonzalez (2012) utilizam o método de cálculo número Newton-Rapson para a caracterização do perfil. Outra forma de cálculo dos parâmetros de interferência, mostrado por McCosker (2012), apresenta aferição de dados empíricos de perfis Naca TN 221 e a aproximação de Prandtl que apresenta o fenômeno de perda da base e na ponta das pás. Nessas localidades, o escoamento flui radialmente para cima do perfil, reduzindo a circulação do fluido e consequentemente o torque.

4 Processo de modelagem e otimização

O dimensionamento de uma pá deve seguir uma estrutura bem fundamentada e coesa com a literatura. E por isso, o uso de softwares que descrevem os processos físicos retornando resultados coerentes com testes físicos é adotado como ferramenta chave em processos de construção estrutural de novos perfis. A aquisição dos dados pelos cálculos, toma como auxílio o software XFOIL que é um software livre. Mas mesmo este sendo de grande valia para os cálculos teóricos, a análise dos perfis ainda deve ser empírica para um nível alto de confiabilidade.

Em suma, o projeto de modelagem computacional se baseia na solução de problemas em otimização de perfil de pá por meio teoria BEM (Blade Element Momentum) que consiste em um apanhado de equações integrais e derivativas sob o aerofólio de perfil já definido em função de ângulos de torsão, de incidência e de velocidade e conceitos da dinâmica dos fluidos. Para o atesto do método, atribui-se parâmetros comuns a turbinas hidrocinéticas encontradas na literatura, bem como as características de escoamento do fluido desenvolvidos pelo LEA. Esses parâmetros são desenvolvidos nas equações apresentadas no marco teórico e muitas delas resolvidas em softwares como XFOIL. Esses softwares são empregado de forma a apresentar diferentes valores de coeficientes de arrasto e sustentação para vários ângulos de ataque no perfil de pá e assim conhecer os parâmetros ótimos de construção do objeto de forma iterativa em conjunto com o regime dos algoritmos desenvolvidos.

Por fim, a modelagem teórica é feita obedecendo os padrões desenvolvidos na literatura e em projeto a cerca das parametrizações PARSEC e otimização PSO. Em resumo, por meio do banco de dados de aerofólios, a parametrização pela metodologia PARSEC, que será explorada na próxima seção, traduz as coordenadas cartesianas em coordenadas polares, exatamente em 11 parâmetros polares que descrevem qualquer aerofólio com uma boa precisão. Esses 11 parâmetros são então armazenados para serem base inicial de geração de novos aerofólios randomizados pelo processo de parametrização BSpline. Essa tradução ocorre uma vez na entrada do aerofólio a ser otimizado, e as novas geometrias criadas são baseadas nesses 11 parâmetros e formadas aleatoriamente, traduzidas uma a uma para coordenadas cartesianas novamente, a fim de serem analisadas pelo Xfoil a cada iteração.

Assim, com um espaço amostral de 25 aerofólios, cada um contendo os 11 parâmetros PARSEC de cada geometria, o processo de otimização PSO se inicia, retornando uma gama de iterações que convergem os perfis para o ponto ótimo de acordo com CL, CD e CM. Os perfis então são submetidos a comparações com modelos anteriores, já criados e testados pelo processo, até que se chegue a um resultado ótimo após 'n' iterações. Por fim, os 11 parâmetros ótimos avaliados são então traduzidos novamente para coordenadas cartesianas do aerofólio otimizado. E essa parte é determinante para o prosseguimento do trabalho que será baseado na construção e simulação, com um futuro uso do software ANSYS, fechando o ciclo de atividades.

4.1 Processos computacionais

Formulações algorítmicas para o desenvolvimento de otimizações, principalmente no projeto de novas turbinas, sejam elas para propulsão ou para geração de energia, vem atraindo interesses de investidores e pesquisadores da área. Contudo, para o bom desenvolvimento, vários parâmetros e variáveis devem ser levadas em consideração, bem como estudo dos processos físicos. Para tal, levando em consideração aerofólios/hidrofólios, objeto de investigação desse trabalho, a representação dos mesmos com a menor carga computacional e menor gama de parâmetros é o que define a eficiência do algoritmo, além de que a otimização do perfil, deve ser capaz de gerar uma grande variedade de aerofólios dentre os quais o ótimo deve estar incluso.

4.2 Métodos de parametrização

Existe um grande número de aerofólios definidos por pesquisadores em diferentes especialidades. Contudo, muitos desses foram definidos por expressões matemáticas, que por sua vez, definem famílias como NACA, GOE, Joukovsky, e que infelizmente, detém de características distintas (ROMÁN, 1993). Se os perfis não são membros de uma mesma família, é difícil a busca por uma parametrização que abranja todas elas. Sendo que a parametrização, por sua vez, tem a finalidade de reduzir parâmetros que descrevem o perfil do hidro/aerofólio para que sua modelagem e manipulação seja facilitada.

Para que haja uma modelagem da geometria de um aerofólio é necessário alguns parâmetros para descrever o perfil. Nesse preceito, vários processos e metodologias foram criadas ao longo do desenvolvimento de turbinas, principalmente no avanço da aeronáutica e, até mesmo, da industria automotiva. Uma das formas mais comum de parametrização é por meio de coordenadas cartesianas. Contudo, para melhores resultados, apesar de ser fácil a implementação, há o requerimento de um muitos pontos para descrever um perfil, principalmente no bordo de ataque e de fuga do perfil.

Com o avanço da tecnologia e programação, outros métodos populares surgiram na representação matemática que consiste na utilização de funções polinomiais analíticas. Um exemplo, que é investigado nesse trabalho, é a parametrização por curvas de Bezier associado com a representação spline, mostrado nos trabalhos de Consentino Gary B Holst

(1986). Resumidamente, este método consiste em representar a curva do aerofólio usando segmento de funções Bezier, reduzindo com isso o número de parâmetros ajustados, já que a representação pode ser modificada pelo usuário.

Na mesma geração, o método Parsec, que é utilizado na proposta do projeto paralelamente à parametrização para dar robustez e enlace ao processo de otimização, utiliza 11 parâmetros como entrada do aerofólio.

Como no processo de otimização existe alguns preceitos que a metodologia de parametrização deve atender para ser discutida como viável e eficiente, atentou-se a objetivos apresentado por Kulfan, Bussoletti et al. (2006), eles são:

1. Gerar superfícies contínuas e mais próximas a realidade;
2. Seja, em termos computacionais, de processos rápidos, com uma certa acurácia e consistente em todo corpo do algoritmo;
3. Seja capaz de representar uma grande quantidade de aerofólios com poucos parâmetros;
4. Tenha uma consistência no processo de criação do perfil, além de ser capaz de alterar a geometria;

4.2.1 Parametrização PARSEC

A descrição de um aerofólio se baseia em uma série de polinômios nos quais parâmetros podem ser variados continuamente para que se chegue no extremo da função base. Porém, quanto mais famílias de aerofólios existem no processo, mais complicado se torna a versatilidade de uma função. Com isso, a parametrização PARSEC tenta recriar a maior quantidade de perfis sem uma perda substancial no perfil. Esse esquema foi criado por Sobieczky em 1998, e os parâmetros são:

1. Raio do bordo de ataque: r_{LE};
2. Posição do ponto máximo no extradorso abscissa: x_{up}
3. Posição do ponto máximo no extradorso ordenada:y_{up}
4. Curvatura no ponto máximo no extradorso: y_{xxup}
5. Posição do ponto mínimo no intradorso abiscissa: x_{lo}
6. Posição do ponto máximo no extradorso ordenada:y_{lo}
7. Curvatura no ponto de mínimo no intradorso: y_{xxlo}

8. Direção do bordo de fuga: α_{TE}
9. Ângulo do bordo de fuga: θ_{TE}
10. Ordenada do bordo de fuga: y_{TE}
11. Espessura do bordo de fuga: $\Delta\mathrm{y}_{TE}$

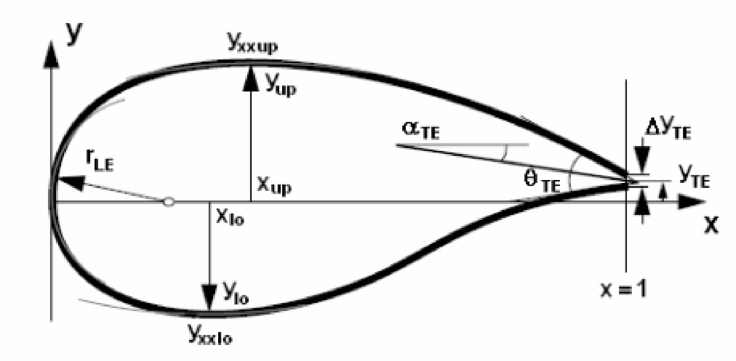


Figura 8: Representação da parametrização PARSEC

Adaptada de (SOBIECZY, 1998)

Constituir aerofólios pelas curvas de Bezier e/ou BSpline podem ser uma escolha sensata, porém a relação entre os parâmetros utilizados e as características físicas do perfil (como sustentação, arrasto e momentos) não é clara ou direta. E nesse quesito, a configuração Parsec trabalha com uma série de parâmetros geométricos que são altamente relacionadas as quantidades físicas do perfil, dando mais flexibilidade de estudo na parametrização.

Os perfis caracterizados pela parametrização PARSEC são definidos por duas funções polinomiais e uma condição de contorno. As duas funções descrevem o extradorso e o intradorso das curvas do perfil, e o contorno determina a condição do bordo de ataque.

$$y_e(x_e) = \sum_{k=1}^{6} \alpha_{ek} \mathbf{x_e}^{k-\frac{1}{2}} \tag{4.1}$$

$$y_i(x_i) = \sum_{k=1}^{6} \alpha_{ik} \mathbf{x_i}^{k-\frac{1}{2}} \tag{4.2}$$

$$\alpha_{i1} = -\alpha_{e1} \tag{4.3}$$

Os termos α_{ek} e α_{ik} são números reais que são encontrados de forma analítica por condições geométricas impostas ao perfil. Os limites de $x_e LE = 0$ até $x_e TE = 1$, são caracterizados pelo bordo de ataque (leading edge) e bordo de fuga (trailing edge), respectivamente. Com isso, a análise das funções passam pela posição máxima, região onde a derivada é nula, curvatura máxima, a posição do bordo de fuga, e a inclinação da mesma:

$$y_e max = \sum_{k=1}^{6} \alpha_{ek} \mathbf{x_{emax}}^{k-\frac{1}{2}} \qquad y_i min = \sum_{6}^{k=1} \alpha_{ik} \mathbf{x_{imin}}^{k-\frac{1}{2}} \tag{4.4}$$

$$y_e' max = 0 = \sum_{k=1}^{6} (k - \frac{1}{2}) \alpha_{ek} \mathbf{x_{emax}}^{k-\frac{3}{2}} \qquad y_i' min = 0 = \sum_{k=1}^{6} (k - \frac{1}{2}) \alpha_{ik} \mathbf{x_{imin}}^{k-\frac{3}{2}} \tag{4.5}$$

$$y_e'' max = \sum_{k=1}^{6} (k - \frac{1}{2})(k - \frac{3}{2}) \alpha_{ek} \mathbf{x_{emax}}^{k-\frac{5}{2}} \qquad y_i'' min = \sum_{6}^{k=1} (k - \frac{1}{2})(k - \frac{3}{2}) \alpha_{ik} \mathbf{x_{imin}}^{k-\frac{5}{2}} \tag{4.6}$$

$$\tan{(\alpha - \frac{1}{2}\beta)} = \sum_{k=1}^{6} (\frac{1}{2}) \alpha_{ek} \qquad \tan{(\alpha + \frac{1}{2}\beta)} = \sum_{k=1}^{6} (\frac{1}{2}) \alpha_{ik} \tag{4.7}$$

A condição do raio para o bordo de ataque é adicionado a curvatura do primeiro polinômio.

$$r_{LE} = -\frac{1}{2} \alpha_{e1} \tag{4.8}$$

Com isso, os 12 parâmetros α_{ek},α_{ik} são integrados ao conjunto dos 11 parâmetros geométricos que caracteriza essa parametrização $r_{LE}, x_{emax}, y_{emax}, y''_{emax}, x_{imin}, y_{imin}, y''_{imin}, y_{TE}, \alpha, \beta$ e a condição $\alpha_{i1} = -\alpha_{e1}$. Reorganizando os termos para formação de um sistema linear, as equações podem ser então resolvidas e os coeficiente gerados na matriz abaixo:

$$a_{e1} = \sqrt{2r_{LE}}$$
$$a_{i1} = -a_{e1}$$

$$\begin{bmatrix} a_{e2} \\ a_{e3} \\ a_{e4} \\ a_{e5} \\ a_{e6} \end{bmatrix} = \begin{bmatrix} \frac{3}{4}x_{emax}^{-\frac{1}{2}} & \frac{15}{4}x_{emax}^{\frac{1}{2}} & \frac{35}{4}x_{emax}^{\frac{3}{2}} & \frac{63}{4}x_{emax}^{\frac{5}{2}} & \frac{99}{4}x_{emax}^{\frac{7}{2}} \\ \frac{3}{2}x_{emax}^{\frac{1}{2}} & \frac{5}{2}x_{emax}^{\frac{3}{2}} & \frac{7}{2}x_{emax}^{\frac{5}{2}} & \frac{9}{2}x_{emax}^{\frac{7}{2}} & \frac{11}{2}x_{emax}^{\frac{9}{2}} \\ x_{emax}^{\frac{3}{2}} & x_{emax}^{\frac{5}{2}} & x_{emax}^{\frac{7}{2}} & x_{emax}^{\frac{9}{2}} & x_{emax}^{\frac{11}{2}} \\ \frac{3}{2} & \frac{5}{2} & \frac{7}{2} & \frac{9}{2} & \frac{11}{2} \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix}^{-1} \begin{bmatrix} y''_{emax} + \frac{1}{4}a_{e1}x_{emax}^{-\frac{3}{2}} \\ -\frac{1}{2}a_{e1}x_{emax}^{-\frac{1}{2}} \\ y_{emax} - a_{e1}x_{emax}^{\frac{1}{2}} \\ tan\left(\alpha - \frac{1}{2}\beta\right) - \frac{1}{2}a_{e1} \\ y_{TE} + \frac{1}{2}\Delta y_{TE} - a_{e1} \end{bmatrix}$$

$$\begin{bmatrix} a_{i2} \\ a_{i3} \\ a_{i4} \\ a_{i5} \\ a_{i6} \end{bmatrix} = \begin{bmatrix} \frac{3}{4}x_{imin}^{-\frac{1}{2}} & \frac{15}{4}x_{imin}^{\frac{1}{2}} & \frac{35}{4}x_{imin}^{\frac{3}{2}} & \frac{63}{4}x_{imin}^{\frac{5}{2}} & \frac{99}{4}x_{imin}^{\frac{7}{2}} \\ \frac{3}{2}x_{imin}^{\frac{1}{2}} & \frac{5}{2}x_{imin}^{\frac{3}{2}} & \frac{7}{2}x_{imin}^{\frac{5}{2}} & \frac{9}{2}x_{imin}^{\frac{7}{2}} & \frac{11}{2}x_{imin}^{\frac{9}{2}} \\ x_{imin}^{\frac{3}{2}} & x_{imin}^{\frac{5}{2}} & x_{imin}^{\frac{7}{2}} & x_{imin}^{\frac{9}{2}} & x_{imin}^{\frac{11}{2}} \\ \frac{3}{2} & \frac{5}{2} & \frac{7}{2} & \frac{9}{2} & \frac{11}{2} \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix}^{-1} \begin{bmatrix} y''_{imin} + \frac{1}{4}a_{i1}x_{imin}^{-\frac{3}{2}} \\ -\frac{1}{2}a_{i1}x_{imin}^{-\frac{1}{2}} \\ y_{imin} - a_{i1}x_{imin}^{\frac{1}{2}} \\ tan\left(\alpha + \frac{1}{2}\beta\right) - \frac{1}{2}a_{i1} \\ y_{TE} - \frac{1}{2}\Delta y_{TE} - a_{i1} \end{bmatrix}$$

4.2.2 Introdução a Parametrização BSPLINE

Os primeiros trabalhos com as curvas B-spline datam de cinquenta anos atrás no trabalho de Shoenberg (1988) e foi seguido pelo desenvolvimento de algoritmos capazes de recriar curvas e formas complexas. Desta maneira, provou ser efetiva para a representação do método e desenvolvimento de pesquisas utilizados constantemente na academia.

A curva B-spline é definida como uma combinação linear de pontos de controle P_i e bases de função B-spline $N_{i,k}(t)$. É caracterizada como uma forma geral das curvas de Bezier, podendo ser definida pela equação:

$$P(t) = \sum_{i=1}^{n+1} N_{i,k}(t)P_i \qquad t_{min} \geq t \geq t_{max} \tag{4.9}$$

Em termos, as curvas possuem uma característica em termos de centro de massa de pontos de massa em um espaço. Essa caracterização pode ser então visualizada, por exemplo, considerando quatro massas *m0,m1,m2 e m3*, como mostradas na Fig. (4.10), localizadas em pontos P0, P1, P2 e P3 (MIT, 2009). O centro de massa pode ser então representado pela equação:

$$P = \frac{m_0P_0 + m_1P_1 + m_2P_2 + m_3P_3}{m_0 + m_1 + m_2 + m_3} \tag{4.10}$$

Contudo, em vez de ser fixado em valores constantes, cada massa varia em funções de parâmetros t. Então, tem-se $m_0 = (1-t)^3$, $m_1 = 3t(1-t)^2$, $m_2 = 3t^2(1-t)$ e $m_3 = t^3$ na qual o valor da massa varia com t, assumindo diferenças no centro de massa. Pela Figura (10), pode-se verificar que t varia de 0 a 1, e a curva é varrida pelo centro de

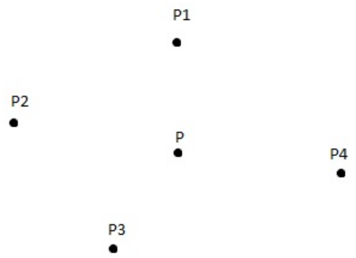


Figura 9: Centro de massa de quatro pontos.

Fonte: Autor

massa. Para qualquer valor de t, a soma das massas é igual a 1, o quê nos dá margem para simplificar a Eq. (4.10).

Quando t=0, $m_0 = 1$ e $m_1 = m_2 = m_3 = 0$, forçando a curva passar por P_0. Também, quando t=1, $m_3 = 1$ e $m_1 = m_2 = m_0 = 0$, o quê faz a curva passar pelo ponto P_3. Com isso, a curva é tangente a $P_0 - P_1$ e $P_3 - P2$, como pode ser visto na Fig. (11). O manuseio dessas propriedades faz com que essas curvas possam representar infinitas possibilidade de parametrização.

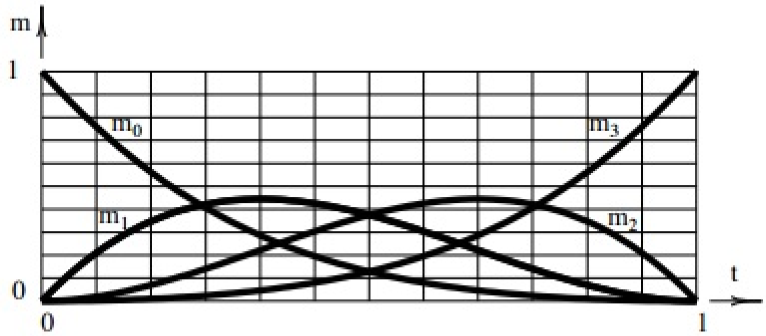


Figura 10: Curvas mistas de Bézier cúbica.

Fonte: (MIT, 2009)

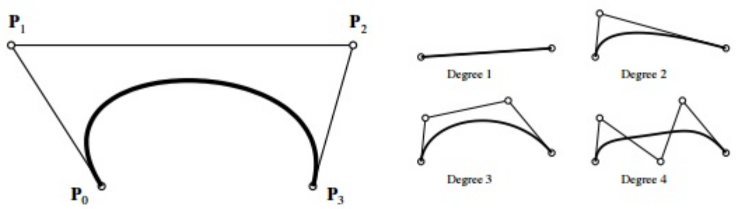


Figura 11: Curva de Bézier cúbica.

Fonte: (MIT, 2009)

A representação de Bézier tem algumas desvantagens. Uma delas é que o número de pontos de controle está diretamente relacionado com o grau da função. Com isso, para aumentar a complexidade da forma, que quer ser representada pela curva, requere o aumento do grau, ou satisfazendo a condição de continuidade entre segmentos consecutivos que compõe dada forma. A segunda desvantagem se baseia no fato de que mudando os pontos de controle, muda-se a curva como um todo, fazendo de algumas formas, um alto grau de complexidade. E por esses aspectos que a representação polinomial *B-spline* ou *basis-spline* é uma melhoria (MIT, 2009).

S-pline é, originalmente, uma ferramenta para desenho na qual consiste em uma tira flexível de metal, como uma régua, utilizada para desenhar curvas, fixando pesos a determinados pontos para dar forma as próprias curvas. E assim como no Bézier, a curva não passa pelos pontos de controle (pesos), porém difere-se por baseiar em um conjunto de segmentos de linhas suavizados que permite a continuidade das linhas.

As *splines* também possuem o príncipio básico das da interpolação de *Hermite* por partes, sendo que esta possui a característica de solucionar questões de definição de uma curva dado seus pontos extremos e as derivadas nestes pontos, e geralmente são empregadas para interpolação de pontos. Contudo, na interpolação de *Hermite* por partes, a derivada de cada ponto é imposta, enquanto na *spline* é estabelecida somente a condição de continuidade. Ou seja, dois segmentos consecutivos da curva apresentam derivadas de ordem k iguais.

Nesse apanhado de evolução paramétricas, a série B-spline é uma versão da *spline*, que por sua vez é uma versão melhorada das curvas de Bezier. A série implementa o controle local da curva e é composta por uma série de $m-2$ segmentos de curva $C_1, C_2 ... C_m$ definidos por $m+1$ pontos de controle $P_0, P_1 ... P_m$ para $m \geq 3$.

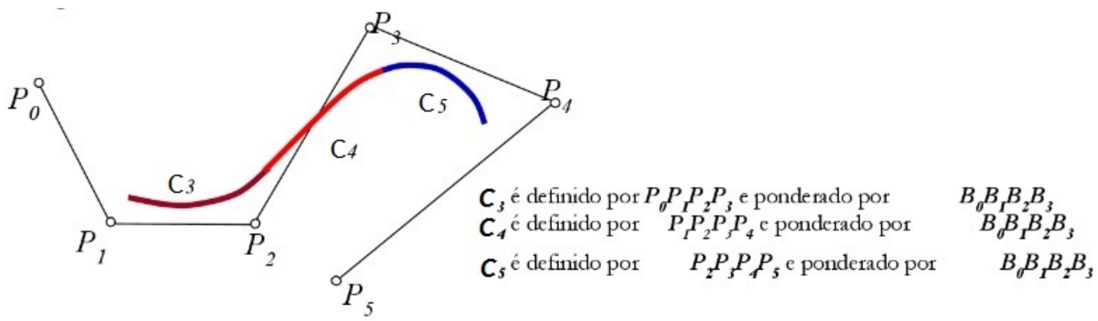


Figura 12: Curva de segmentos B-spline.

Modificado de Piegl e Tiller (1987)

Para construção de uma curva *B-spline*, precisa-se de um vetor de nós, das funções de base, por isso *"Basis"*, além de um conjunto de pontos de controle. A formulação de um segmento $C_i(u = t)$ B-spline cúbica, então é expresso segundo a relação demonstrada na Eq. (4.9), sendo *"i"* o índice global para os pontos de controle, assim como *"t"* o parâmetro global. Essa troca de parâmetros, deu-se pelo fato que em curvas de Bézier, o número de pontos de controle determina o grau da função base, enquanto em B-spline, *"k"* é quem determina por "k-1"identidades Piegl e Tiller (1987).

Assim, as funções de base são descritas como:

$$N_{i,1}(t) = 1 \qquad se \qquad t_i \geq u \geq t_{i+1} \qquad ou \qquad = 0 \tag{4.11}$$

e

$$N_{i,d}(t) = \frac{(t - t_i)N_{i,d-1}(t_i)}{t_{i,d-1} - t_i} + \frac{t_{i+d} - t)N_{i+1,d-1}(t)}{t_{i+d} - t_{i+1}} \tag{4.12}$$

Nota-se que:

1. $N_{i,0}(u = t)$ é uma função pulso, é sempre igual a zero a não ser no intervalo aberto $t \in [u_i, u_{i+1}]$;
2. Para $d > 0$, $N_{i,d}(u)$ é uma combinação linear de duas funçoes de base de grau $d - 1$;
3. O cálculo de um conjunto de funções de base requer a especificação de um vetor de nós U, e o grau p;
4. A Equação (4.12) pode ter quocientes 0/0. Ela é então compreendida como 0;
5. $N_{i,p}(u)$ são polinomiais por partes que representam todo o domínio real. Porém, geralmente apenas o intervalo $[u_i, u_{i+1}]$ que interessa;
6. O cálculo das funções de grau "d"gera um diagrama triangular na forma:

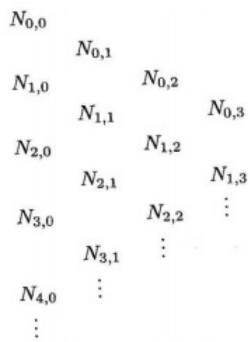


Figura 13: Matriz de construção de curva de segmentos B-spline.

Modificado de Piegl e Tiller (1987)

Como já demonstrado, a vantagem desse método é que a movimentação de um ponto de controle não afeta tanto a estrutura, no máximo um conjunto de 4 segmentos. Como mostrado na Fig. (14), mudando o controle 4, modifica-se apenas C4 e C5:

Para o cálculo de uma curva B-spline são necessários três etapas:

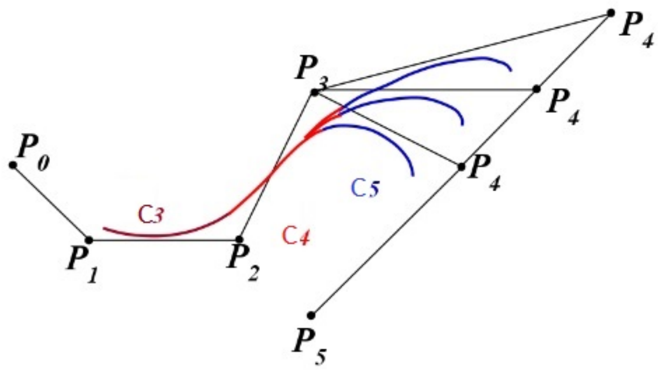


Figura 14: Modificação de curva pelos pontos de controle.

Modificado de Piegl e Tiller (1987)

1. Saber a distância que o nó u_i está no vetor de nós U;
2. Calcular as funções de bases que não sejam nulas;
3. Multiplicar as funções base não nulas pelos seus respectivos pontos de controle.

As curvas *B-spline* são caracterizadas de três maneiras: aberta, fechada ou *"clamped"*/amarrada mostradas na Fig. (15). Um exemplo de curva fechada é demonstrada no trabalho de Pereira (2014), com a adequação de nós e pontos de controle.

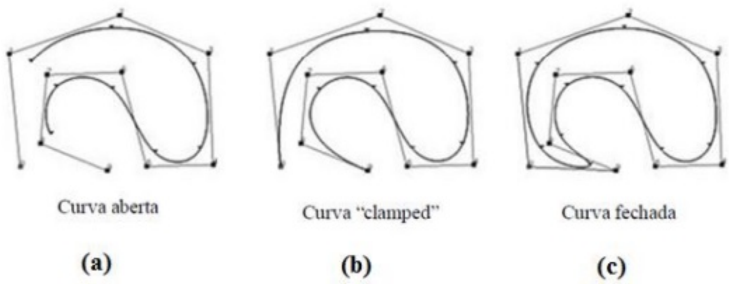


Figura 15: Classificação das curvas *B-spline*. (a)Curva aberta; (b)Curva "amarrada", (c) Curva fechada.

Fonte: (YAMAGUCHI, 2012)

Toma-se uma curva de grau 3, o vetor de nós que descreve-a sendo $U = (0 \quad 2 \quad 5 \quad 7 \quad 8 \quad 14 \quad 14.5 \quad 14.5 \quad 16)$ e os pontos de controle $P = (1;6)(0;3.5)(2;2.5)(3.5;0)(5;2)(7;3.5)(5;6)$.

Para construir a curva fechada deve-se então acrescentar mais 4 pontos de controles iguais aos 4 primeiros pontos, ou seja: $P = (1;6)(0;3.5)(2;2.5)(3.5;0)(5;2)(7;3.5)(5;6)(1;6)$ $(0;3.5)(2;2.5)(3.5;0)$ e acrescentar mais 6 novos nós, com distâncias iguais aos 6 primeiros vãos do vetor. Assim, calculam-se os valores dos vãos:

1. $\text{vão} = 2 - 0 = 2$

2. vão $= 5 - 2 = 3$
3. vão $= 7 - 5 = 2$
4. vão $= 8 - 7 = 2$
5. vão $= 14 - 8 = 6$
6. vão $= 14.5 - 14 = 0.5$

Deste modo, são acrescentados os novos nós respeitando os valores dos vãos encontrados, obtendo: $\mathbf{U} = (0 \quad 2 \quad 5 \quad 7 \quad 8 \quad 14 \quad 14.5 \quad 14.5 \quad 16 \quad 18 \quad 21 \quad 23 \quad 24 \quad 30 \quad 30.5)$ A curva então é representada na Fig. (16).

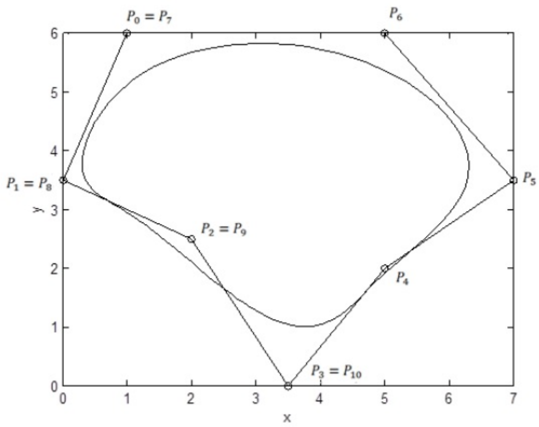


Figura 16: Curva B-spline fechada.

Fonte: (PEREIRA, 2014)

4.2.3 Metodologia Parsec-Bspline

O método Parsec poderia representar significativamente uma grande gama de aerofólios sem grandes perdas diretas na geometria se as condições de contorno fossem dadas. Com isso, antes é necessário calcular e manipular essas condições para os perfis desejados. Geralmente, como já dito, os perfis são representados por coordenadas cartesianas que indicam um determinado número de pontos no extradorso e no intradorso. Sendo assim, o método para reconhecer as condições de contorno do aerofólio se baseou no método dos mínimos quadrados.

Dado um grupo de pontos, como descrito, e suas coordenadas (X_{ej}, Y_{ej}) e (X_{ik}, Y_{ik}) para $j = 1...N_e$ e $k = 1...N_i$ com $n = N_e + N_i$, a representação do erro acometido representado pelos polinômios Parsec podem ser calculadas por:

$$e = \sum_{k=1}^{Ne} (Y_{ek} - y_{ek}(X_{ek}))^2 + \sum_{k=1}^{Ni} (Y_{ik} - y_{ik}(X_{ik}))^2 \tag{4.13}$$

Faz-se necessário minimizar o erro com os parâmetros que estamos buscando α_{ek}, α_{ik} para integrar o mais próximo polinômio para dado perfil. Com isso, para se minimizar, deriva-se o erro com respeito aos parâmetros:

$$\frac{\delta e}{\delta \alpha_{ek}, \alpha_{ik}} = 0 \tag{4.14}$$

Têm-se a relação:

$$\overline{Z_e} = \frac{1}{N_e} \sum_{k=1}^{N_e} Z_e \qquad \overline{Z_i} = \frac{1}{N_i} \sum_{k=1}^{N_i} Z_i \tag{4.15}$$

No qual Ze é qualquer valor considerado no extradorso e Zi é qualquer valor considerado no intradorso. Sendo assim, as equações podem ser reescritas:

$$\frac{\partial e}{\partial a_{e1}}: \; \left(N_e \overline{X_e} + N_i \overline{X_i}\right) a_{e1} + N_e \sum_{k=2}^{6} a_{ek} \overline{X_e^{k}} - N_i \sum_{k=2}^{6} a_{ik} \overline{X_i^{k}} = N_e \overline{X_e^{\frac{1}{2}} Y_e} - N_i \overline{X_i^{\frac{1}{2}} Y_i}$$

$$\frac{\partial e}{\partial a_{e2}}: \; a_{e1} \overline{X_e^{2}} + \sum_{k=2}^{6} a_{ek} \overline{X_e^{k+1}} = \overline{y_e X_e^{\frac{3}{2}}} \qquad \frac{\partial e}{\partial a_{i2}}: \; a_{i1} \overline{X_i^{2}} + \sum_{k=2}^{6} a_{ik} \overline{X_i^{k+1}} = \overline{Y_i X_i^{\frac{3}{2}}}$$

$$\frac{\partial e}{\partial a_{e3}}: \; a_{e1} \overline{X_e^{3}} + \sum_{k=2}^{6} a_{ek} \overline{X_e^{k+2}} = \overline{Y_e X_e^{\frac{5}{2}}} \qquad \frac{\partial e}{\partial a_{i3}}: \; a_{i1} \overline{X_i^{3}} + \sum_{k=2}^{6} a_{ik} \overline{X_i^{k+2}} = \overline{Y_i X_i^{\frac{5}{2}}}$$

$$\frac{\partial e}{\partial a_{e4}}: \; a_{e1} \overline{X_e^{4}} + \sum_{k=2}^{6} a_{ek} \overline{X_e^{k+3}} = \overline{Y_e X_e^{\frac{7}{2}}} \qquad \frac{\partial e}{\partial a_{i4}}: \; a_{i1} \overline{X_i^{4}} + \sum_{k=2}^{6} a_{ik} \overline{X_i^{k+3}} = \overline{Y_i X_i^{\frac{7}{2}}}$$

$$\frac{\partial e}{\partial a_{e5}}: \; a_{e1} \overline{X_e^{5}} + \sum_{k=2}^{6} a_{ek} \overline{X_e^{k+4}} = \overline{Y_e X_e^{\frac{9}{2}}} \qquad \frac{\partial e}{\partial a_{i5}}: \; a_{i1} \overline{X_i^{5}} + \sum_{k=2}^{6} a_{ik} \overline{X_i^{k+4}} = \overline{Y_i X_i^{\frac{9}{2}}}$$

$$\frac{\partial e}{\partial a_{e6}}: \; a_{e1} \overline{X_e^{6}} + \sum_{k=2}^{6} a_{ek} \overline{X_e^{k+5}} = \overline{Y_e X_e^{\frac{11}{2}}} \qquad \frac{\partial e}{\partial a_{i6}}: \; a_{i1} \overline{X_i^{6}} + \sum_{k=2}^{6} a_{ik} \overline{X_i^{k+5}} = \overline{Y_i X_i^{\frac{11}{2}}}$$

E assim, as últimas dez equações formam dois conjuntos de matrizes lineares que podem ser configuradas pela etapa demonstrada abaixo. As linhas de código que traduzem essas matrizes estão anexadas ao apêndice do documento.

Pode-se então separar os termos de acordo com os conjuntos correspondentes de forma que a resolução do sistema linear obedeça os padrões definidos:

$$a_{e1}\begin{bmatrix}\overline{X_e^2}\\\overline{X_e^3}\\\overline{X_e^4}\\\overline{X_e^5}\\\overline{X_e^6}\end{bmatrix}+\begin{bmatrix}\overline{X_e^3}&\overline{X_e^4}&\overline{X_e^5}&\overline{X_e^6}&\overline{X_e^7}\\\overline{X_e^4}&\overline{X_e^5}&\overline{X_e^6}&\overline{X_e^7}&\overline{X_e^8}\\\overline{X_e^5}&\overline{X_e^6}&\overline{X_e^7}&\overline{X_e^8}&\overline{X_e^9}\\\overline{X_e^6}&\overline{X_e^7}&\overline{X_e^8}&\overline{X_e^9}&\overline{X_e^{10}}\\\overline{X_e^7}&\overline{X_e^8}&\overline{X_e^9}&\overline{X_e^{10}}&\overline{X_e^{11}}\end{bmatrix}\begin{bmatrix}a_{e2}\\a_{e3}\\a_{e4}\\a_{e5}\\a_{e6}\end{bmatrix}=\begin{bmatrix}\overline{y_eX_e^{\frac{3}{2}}}\\\overline{Y_eX_e^{\frac{5}{2}}}\\\overline{Y_eX_e^{\frac{7}{2}}}\\\overline{Y_eX_e^{\frac{9}{2}}}\\\overline{Y_eX_e^{\frac{11}{2}}}\end{bmatrix}$$

$$a_{i1}\begin{bmatrix}\overline{X_i^2}\\\overline{X_i^3}\\\overline{X_i^4}\\\overline{X_i^5}\\\overline{X_i^6}\end{bmatrix}+\begin{bmatrix}\overline{X_i^3}&\overline{X_i^4}&\overline{X_i^5}&\overline{X_i^6}&\overline{X_i^7}\\\overline{X_i^4}&\overline{X_i^5}&\overline{X_i^6}&\overline{X_i^7}&\overline{X_i^8}\\\overline{X_i^5}&\overline{X_i^6}&\overline{X_i^7}&\overline{X_i^8}&\overline{X_i^9}\\\overline{X_i^6}&\overline{X_i^7}&\overline{X_i^8}&\overline{X_i^9}&\overline{X_i^{10}}\\\overline{X_i^7}&\overline{X_i^8}&\overline{X_i^9}&\overline{X_i^{10}}&\overline{X_i^{11}}\end{bmatrix}\begin{bmatrix}a_{i2}\\a_{i3}\\a_{i4}\\a_{i5}\\a_{i6}\end{bmatrix}=\begin{bmatrix}\overline{y_iX_i^{\frac{3}{2}}}\\\overline{Y_iX_i^{\frac{5}{2}}}\\\overline{Y_iX_i^{\frac{7}{2}}}\\\overline{Y_iX_i^{\frac{9}{2}}}\\\overline{Y_iX_i^{\frac{11}{2}}}\end{bmatrix}$$

$$\mathbf{a}_e=\begin{bmatrix}a_{e2}\\a_{e3}\\a_{e4}\\a_{e5}\\a_{e6}\end{bmatrix};\quad \mathbf{b}_e=\begin{bmatrix}\overline{X_e^2}\\\overline{X_e^3}\\\overline{X_e^4}\\\overline{X_e^5}\\\overline{X_e^6}\end{bmatrix};\quad \mathbf{c}_e=\begin{bmatrix}\overline{y_eX_e^{\frac{3}{2}}}\\\overline{Y_eX_e^{\frac{5}{2}}}\\\overline{Y_eX_e^{\frac{7}{2}}}\\\overline{Y_eX_e^{\frac{9}{2}}}\\\overline{Y_eX_e^{\frac{11}{2}}}\end{bmatrix};$$

$$\mathbf{A}_e=\begin{bmatrix}\overline{X_e^3}&\overline{X_e^4}&\overline{X_e^5}&\overline{X_e^6}&\overline{X_e^7}\\\overline{X_e^4}&\overline{X_e^5}&\overline{X_e^6}&\overline{X_e^7}&\overline{X_e^8}\\\overline{X_e^5}&\overline{X_e^6}&\overline{X_e^7}&\overline{X_e^8}&\overline{X_e^9}\\\overline{X_e^6}&\overline{X_e^7}&\overline{X_e^8}&\overline{X_e^9}&\overline{X_e^{10}}\\\overline{X_e^7}&\overline{X_e^8}&\overline{X_e^9}&\overline{X_e^{10}}&\overline{X_e^{11}}\end{bmatrix}$$

As equações que representam os dorso para minimizar o método, podem ser escritos em uma forma mais compacta segundo a Eq. (4.16):

$$(N_e\overline{x_e}+N_i\overline{x_i})a_{e1}+N_eb^T{}_ea_e-N_ib^T{}_ia_i=N_e\overline{y_ex_e{}^{\frac{1}{2}}}-N_i\overline{y_ix_i{}^{\frac{1}{2}}} \tag{4.16}$$

A matriz então é estabelecidas pelas equações:

$$\mathrm{a}_{e1}\mathrm{b}_e+\mathrm{A}_e\mathrm{a}_e=\mathrm{c}_e \tag{4.17}$$

$$\mathrm{a}_{i1}\mathrm{b}_i+\mathrm{A}_i\mathrm{a}_i=\mathrm{c}_i \tag{4.18}$$

As Eqs. (4.17), (4.18) são então simplificadas de forma a isolar os coeficientes buscados, tornando-as nas Eqs. (4.19) e (4.20) .

$$\mathrm{a}_e=\mathrm{A}_e^{-1}(\mathrm{c}_e-\mathrm{a}_{e1}\mathrm{b}_e) \tag{4.19}$$

$$\mathrm{a}_i = \mathrm{A}_i^{-1}(\mathrm{c}_i + \mathrm{a}_{e1}\mathrm{b}_i) \tag{4.20}$$

Com isso, chega-se a Eq. (4.21):

$$\alpha_{e1} = \frac{N_e\overline{y_e X_e{}^{\frac{1}{2}}} - N_i\overline{y_i X_i{}^{\frac{1}{2}}} + N_i\mathbf{b}_i{}^T\mathbf{A}_i{}^{-1}\mathbf{c}_i - N_e\mathbf{b}_e{}^T\mathbf{A}_e{}^{-1}\mathbf{c}_e}{N_e\overline{x_e} + N_i\overline{x_i} - N_e\mathbf{b}_e{}^T\mathbf{A}_e{}^{-1}\mathbf{b}_e - N_i\mathbf{b}_i{}^T\mathbf{A}_i{}^{-1}\mathbf{b}_i} \tag{4.21}$$

Com os parâmetros α_{ek}, α_{ik} encontrados, a representação dos onze parâmetros (=($r_{LE}, x_{emax}, y_{emaxx}, y''_{emax}, x_{imin}, y_{imin}, y''_{imin}, y_{TE}, \Delta y_{TE}, \alpha, \beta$) que representam o perfil configurado, podem ser também encontrados. Assim, eles são modificados parametricamente pelo usuário para solução dos problemas de otimização.

Mas como já afirmado, esses parâmetros são representações de características físicas de um determinado aerofólio. E por tal, para a construção de novos perfis com base em 11 parâmetros específicos, seria necessário ter a descrição de funções base para cada 1 dos 11 parâmetros a fim de recriar novos outros 11 parâmetros viáveis. Portanto, a integração com uma parametrização que não envolvesse grandezas físicas, como a Bspline, foi adotado como base para a recriação de novos perfis. Após calculados, os valores são salvos em um vetor P(1) a P(11), nos quais são utilizados na função otimização como limite para geração base de novos aerofólios. Esse vetor, por sua vez, entra nas funções de otimização por uma variável global:

```
    if finished
        disp('Analise Xfoil terminado')

          P= parsec(hello.Airfoil.Extra,hello.Airfoil.Intra);
%As coordenadas do aerofolio sao salvas em memoria RAM nos enderecos
%especificados. Assim, essa linha chama  essas coordenadas para serem integradas
% na funcao PARSEC. A funcao entao codifica as mesmas em 11 parametros polares
% mostradas na janela de comando do Matlab.
            Global P
            strdata=num2str(P);
            st=strcat('      Parametros Parsec: ',strdata); disp(st);
            disp(' ');
```

Para melhor avaliação do leitor, parte dos códigos que descrevem ambas parametrizações Parsec e otimização se encontram no APÊNDICE.

Os valores Parsec então são usados como variáveis de entrada na configuração de novos perfis. Para tal, desevolve-se a parametrização baseada em função que descreve os aerofólios da família NACA. Escolheu-se tal abordagem pelo fato de que a parametrização PARSEC associa muito os parametros da família adotada.

```
    Globa P
```

```
...
tbnd=[P(3) P(6)] %Espessura min e max
...
%Parametros que serao matidos constantes
l=P(2); %localizacao da espessura maxima
r=P(1); %raio do bordo de ataque
...
tegap=P(9) %espacamento do bordo de fuga
...
%Limites para os designs baseados nos parametros Parsec
camb=[P(4) P(7)];  %min e max envergadura
cambloc=[P(5) P(2)*2];  %min e max local da max envergadura
teang=[P(10) P(11)];      %min e max para os angulos do bordo de fuga
nacamin=[camb(1);cambloc(1);tbnd(1);teang(1)];
nacamax=[camb(2);cambloc(2);tbnd(2);teang(2)];
```

A superfície do aerofólio no processo de otimização é definido pela curva Bspline. O bordo de ataque e o bordo de fuga são mantidos com pontos de controle fixos. E para manter a característica do bordo de ataque, os segundos pontos de controle são mantidos com a coordenada x fixa em 0. Portanto, para os "pontos que variem o perfil", eles são definidos como $pts + 2$ pontos de controle para o intradorso e extradorso, e $4 * pts - 2$ para o perfil total. Usar muitos pontos deixam a superfície mais desnivelada, com mais curvas intermediárias, então para o caso utiliza-se 4 pts. Sendo assim, tomando a Fig. (17), percebe-se que em termos de pontos variáveis, os pontos (2 3 4) possuem suas coordenadas x e y também variáveis, totalizando 12 coordenadas extradorso e intradorso, mais 2 coordenadas y do ponto (1). Somando, então 14 pontos de controle que dita a configuração dos novos perfis.

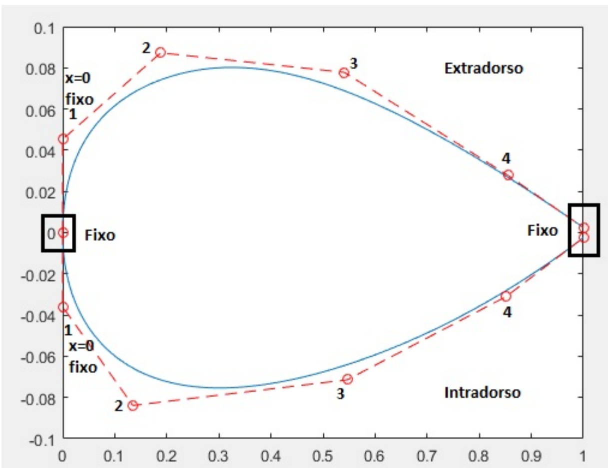


Figura 17: Constituição de perfil por 4 pontos de controle variáveis.

Fonte: Autor

Por fim, os pontos de controle são gerados aleatoriamente, depois da avaliação do perfil inicial e o processo de otimização se inicia. Eles são integrados em uma equação que define aerofólios da família NACA para serem base de equacionamento do processo de parametrização e geração dos pontos Bspline.

```
%Cria a populacao inicial de designs e velocidades comecando pelo aerofolio
%original parametrizado
DV=zeros(n,pop);
vel=zeros(n,pop);
objval=zeros(1,pop);
for i=1:pop
   %Cria designs de aerofolios NACA aleatoriamente
   nacaparams=nacamin+rand(4,1).*(nacamax-nacamin);
   %Encontra os pontos de controle para integrar aos designs
   DV(:,i)=matchfoils_f(nacaparams);

   %Configura a velocidade
   vel(:,i)=(rand(n,1)-0.5).*(xmax-xmin);  %A velocidade inicial pode ser positiva

   %Avalia os designs
   objval(i)=f(DV(:,i));
end
```

Nota-se que DV é o SWARM do processo PSO. A cada iteração, ele cede uma população "pop"de 25 partículas, sendo que cada partícula detém de 14 coordenadas dos pontos que formam os aerofólios. Ou seja, a cada iteração compreende-se 25 perfis diferentes a ser avaliado pelo programa XFOIL.

4.3 Métodos de otimização

A otimização é um importante processo no desenvolvimento da ciência e da natureza. Ela é um conceito que traduz direta ou indiretamente a capacidade de indivíduos ou processos, a partir de um conhecimento específico e aprendizado, de desenvolver características ótimas sob determinado requisito. E matematicamente relaciona a capacidade de conversão dos parâmetros inclusos no desenvolvimento da otimização em números que possam ser computadas por meio de uma inteligência artificial pela exploração do que hoje é chamado *machine learning* (DANTZIG, 1998).

A principal ressalva para a construção do interesse a cerca do processo se iniciou por volta dos anos 40, na qual uma conferência na Universidade de Chicago reuniu matemáticos, engenheiros, economistas e pesquisadores no intuito de desenvolver um método para planejar e melhorar ações militares e governamentais, como o rodízio de navios em um estaleiro e como a distribuição de comódites pelas indústrias. Um dos primeiros projetos para a otimização de sistemas foi pela utilização de programas lineares durante a

II Guerra Mundial que coincidiu com o desenvolvimento de computadores digitais, o que impulsionou mais o poder de uso de tais ferramentas de otimização (DANTZIG, 1998).

Contudo, para fazer uso dessa ferramenta é necessário a prospecção do objeto que será avaliado. É necessário que se conheça qual é a quantidade e objetivo do sistema, como por exemplo, o lucro da empresa, o tempo, a energia, ou qualquer quantidade ou combinação de quantidades que podem ser representadas por números. E geralmente, esses objetivos são funções de características independentes, ou variáveis, que como um todo representa a alma do problema (NOCEDAL; WRIGHT, 2006). A forma de indentificação dos objetivos e variáveis é chamada de modelagem, e por conta disso, várias metodologias de modelagens à cerca do processo de otimização foram elaboradas, principalmente no pós-guerra, destacando-se as redes neurais, os algoritmos genéticos (AG's) e os métodos baseado em gradientes (GM's).

Os algoritmos de otimização são iterativos, ou seja, eles precisam de um chute inicial para que a sequência de valores sejam estimadas conforme o requisito estabelecido pelo operador. Para que um algoritmo seja definido como funcional, ele porém deve seguir determinadas propriedades como verificadas por Nocedal e Wright (2006):

1. **Robustez:** O algoritmo precisa ter uma característica global, ou seja, deve ter uma boa performance em uma variada gama de problemas dentro do escopo do projeto, além de ter que atuar razoavelmente bem com muitos chutes iniciais;
2. **Eficiência:** Eles não devem requerer muita carga computacional ou tempo;
3. **Acurácia:** Os algoritmos devem ser capazes de identificar a solução do problema com precisão sem ser muito sensíveis aos erros decorrentes do código ou computacionais quando o algoritmo é implementado.

Por outro lado, esses objetivos podem ser duais entre sim, ou seja, ao mesmo tempo que melhora um requisito, piora outro, o que mostra o quão complicado é para fazer um algoritmo com uma boa qualidade. Por exemplo, para que haja uma rápida convergência para um problema um pouco mais complexo, talvez o requerimento necessite uma maior carga computacional. Além de que, um método muito robusto pode levar um tempo maior para compilar. E por isso, uma teoria matemática é conveniente para relacionar todos esses parâmetros de forma a otimizar também o processo de construção do algoritmo. Não é possível o entendimento numérico sem uma boa dose de teoria (NOCEDAL; WRIGHT, 2006).

4.3.1 Algoritmos Genéticos

Entre todos os métodos de desenvolvimento em otimização, os algoritmos genéticos são os mais estudadas e usadas pela sua flexibilidade, relativa simplicidade de implementação e eficácia em retornar resultados em ambientes adversos (SILVA, 2005). Em termos gerais, o algoritmo surgiu de forma a imitar o processo evolutivo da natureza, pelos mecanismos adaptativos e genéticos, apesar de não caracterizarem em sim uma forma de busca aleatória e independente como ocorre na natureza. Os AG's usam de suas especificações a busca dentre as regiões amostrais, onde é provável o encontro com o ponto ótimo.

Como já explicitado, os AG's utilizam de um processo iterativo no qual cada iteração é chamada de geração. A medida que as iterações ocorrem, os princípios da seleção natural por meio dos operadores genéticos são implementados, e assim gerando um número determinado de resultados para a próxima geração.

Os AG's utilizam do processamento de grupos de indivíduos ou cromossomos para chegar a um resultado. O cromossomo é uma estrutura de dados, geralmente vetores ou um segmento de números binários, reais ou combinações entre ambas, que em conjunto representam a possível solução do problema a ser otimizado (SILVA, 2005). O conjunto de todas as possíveis configurações que o cromossomo possa se retratado é chamado de *espaço de busca*. Com isso, se o cromossomo representa n parâmetros de uma função, então se caracteriza em um vetor com *dimensões*.

Ainda dentro dos algoritmos, denomina-se o termo genótipos o conjunto de genes que define a constituição genética de um indivíduo sobre estes genes que os operadores genéticos, que serão melhor explicados mais a frente, atuam. É importante dizer que estes vetores tem um tamanho limitado e não infinito. Geralmente, o genótipo é configurado em um vetor binário, no qual cada elemento dentro desse vetor representa a ausência ou a presença de uma determinada característica para a construção da função do indivíduo. E por meio da permutação, esses elementos são combinados formando as características "físicas"ou "reais"do indivíduo, configurando em seu fenótipo (SPEARS et al., 1993).

A representação por vetores binários é utilizada desde as premissas da otimização por algoritmos genéticos. Foi primeiramente descrita por John Henry Holland, um cientista da computação, que tentou formular um código que imitasse as técnicas naturais de evolução na Universidade de Michigan, a exemplo do cérebro humano, colonias de formigas, economia e até toda uma floresta tropical. Mas em 1975 que o Dr. Holland apresentou o primeiro algoritmo genético binário no livro "*Adaptation in Natural and Artificial Systems*". E ainda hoje é a representação mais utilizada pela fácil manipulação, além de ser simples de analisar teoricamente e seguir um determinado padrão como visto na Fig. (18).

Os operadores genéticos, representados na fig.18, são os agentes responsáveis pela mudança das gerações, que por meio de iterações, buscam melhorar as gerações seguintes segundo os requisitos estabelecidos. Pode-se dizer que esses operadores são o núcleo dos AG's e são caracterizados segundo o objetivo do programador para desempenhar determinado papel na "evolução" do código ou da otimização.

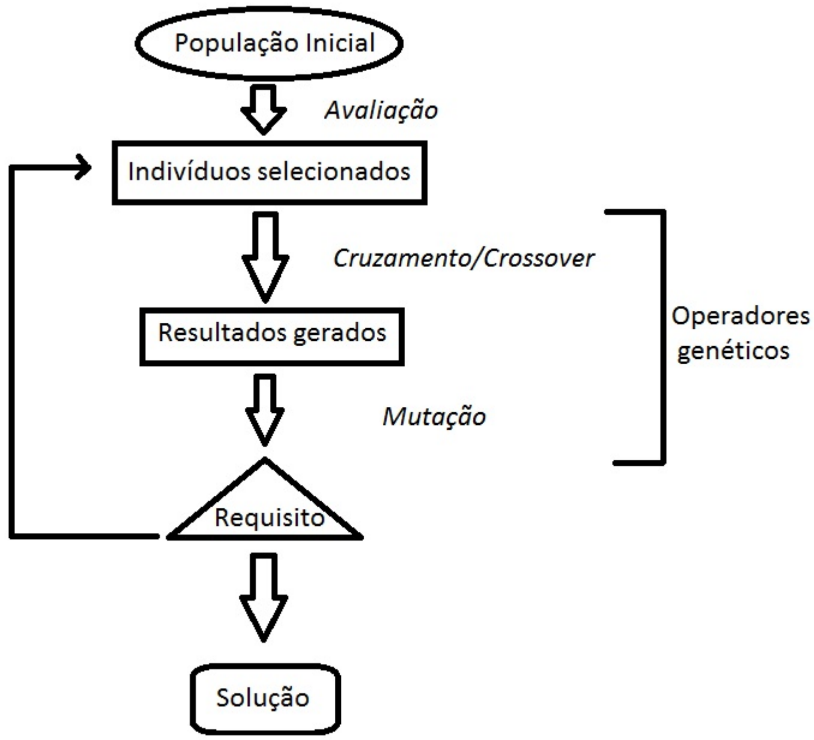


Figura 18: Estrutura base de um algoritmo genético.

Adaptado de (SILVA, 2005)

A escolha do grupo de indivíduos que será "adaptado" é a primeira etapa, sendo que "os filhos"são gerados a partir da recombinação dos cromossomos dos "pais", com a atuação exatamente dos operadores genéticos...o cruzamento, *crossover* e a mutação (RODRIGUES, 2007).

O *crossover* representa a recombinação do material genético do grupo selecionado e faz a combinação das características principais e melhores dos indivíduos para o implemento na próxima geração. Em questão algorítmica, essa recombinação genética pode ainda ser divida em três caminhos:

1. A primeira forma e a mais simplista em realizar um *crossover* é a divisão dos cromossomos dos indivíduos em dois segmentos característicos, e assim as gerações seguintes recebem a combinação de um ponto escolhido no seguimento dos pais, permutando entre si representado na Fig. (19).

2. Outra forma de cruzamento genético, porém mais elaborado, é o de ponto duplo no qual tem as mesmas características do anterior, contudo são selecionado dois pontos, dividindo o cromossomo em três partes como na Fig. (20).

3. Há também o processo chamado de cruzamento em pontos aleatórios, o qual a permutação do material genético é feito sem um padrão específico dentro do segmento

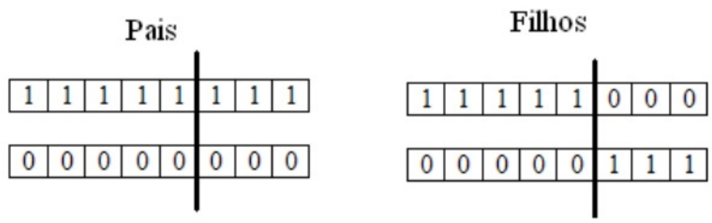


Figura 19: *Crossover* em ponto único escolhido.

Fonte: (RODRIGUES, 2007)

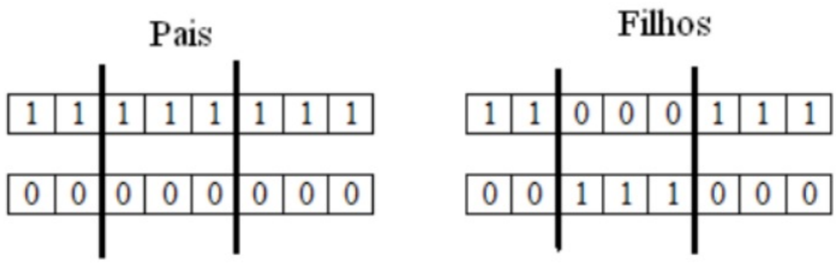


Figura 20: *Crossover* em ponto duplo.

Fonte: (RODRIGUES, 2007)

de cromossomo Fig. (21).

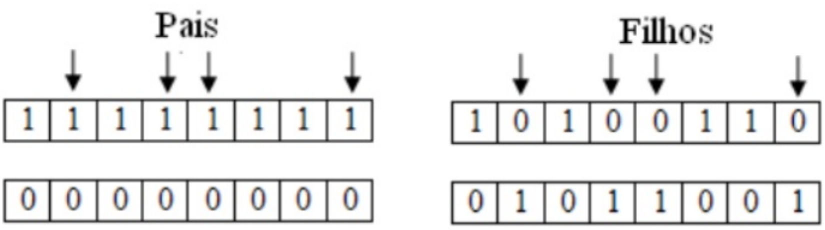


Figura 21: *Crossover* em ponto aleatório.

Fonte: (RODRIGUES, 2007)

4. E por fim, o processo chamado de cruzamento na codificação real fomenta a característica de resolver a solução de forma contínua sem a utilização da estrutura binária existente nos outros meios de *crossover*.

   Os operadores para esta condição atuam na especificação de um gene de cada vez, o que significa que o processo de cruzamento atuará em cada variável do problema independente do segmento do cromossomo. E sendo a caracterização do genes mais evidente nesse processo, uma das maneiras para gerar dois indivíduos com base nos genes dos anteriores é a média ponderada entre o valor dos genes dos pais (RODRIGUES, 2007). O problema relacionado a esse processo é que o ponto médio do intervalo pode ser polarizado levando a homogeneização precoce do grupo escolhido e a um resultado prematuro.

4.3.2 Otimização PSO (Particle Swarm Optimization)

Particle Swarm Optmization (PSO) é uma forma de otimização baseada na técnica estocástica de população e AG's, que foi desenvolvida por Dr. Eberhart e Dr. Kennedy em 1995, inspirados no comportamento de um bando de passáros ou cardume de peixes (HU, 2006).

A maioria das técnicas evolucionárias seguem os procedimentos:

1. Geração randômica de uma população inicial;
2. Cálculo do valor objetivo para cada algoritmo;
3. Reprodução da população baseado nos valores objetivos;
4. Se o requerimento é satisfeito, para. Se não, volta ao passo 2.

A metodologia PSO possui várias características que se assemelham com a proposta da computação "evolucionária", como os algoritmos genéticos (AG). Basicamente, o sistema é inicializado com uma população de possíveis soluções, e procura nesse universo, o ótimo, fazendo melhorias nas gerações. Contudo, contrário aos AG's, PSO não tem um algoritmo baseado na evolução dos operadores, as potenciais soluções, chamadas também de partículas, se movem dentro do espaço amostral seguindo aquelas partículas mais próximas da solução. Além de que, eles ainda possuem memória interna, parte fundamental nas iterações. Também, comparado aos algoritmos genéticos, o processo de comunicação entre as variáveis difere. Nos AG's, os cromossomos transferem informações uns com os outros, então toda a população se move como um grupo para o ponto ótimo, contrário ao PSO.

Comparados ao AG's, as vantagens do PSO são que eles possuem uma fácil adesão a problemas complexos. Não que sejam a melhor forma de resolver um problema, mas possuem poucos parâmetros param ser ajustados. E com esse sistema flexível, ele pode ser aplicado em várias áreas: em funções de otimização, em algoritmos de rede neural, sistemas *fuzzy*, dentre outras.

O seu funcionamento se baseia no seguinte princípio: um grupo de patos está, em uma busca randômica, procurando por comida em determinado espaço (um lago). Há apenas um local no espaço com o objetivo (a comida), mas os patos não sabem exatamente onde, porém eles sentem o quão próximos estão a cada iteração. E por isso, pergunta-se qual é a melhor estratégia para que se chegue ao ponto? Exatamente seguir o pato que está mais próximo da comida.

O algoritmo PSO é inicializado com um grupo de possíveis soluções, ou partículas geradas randomicamente. Ele aprende com o cenário e usa os processos iterativos para resolver os problemas de otimização. No algoritmo, cada solução, ou partícula, representa um "pato" no espaço. Todas as partículas têm valores de ajuste que são verificadas por uma função a ser otimizada, assim como possui velocidades e direções das partículas no espaço. Essas percorrem o espaço seguindo o melhor valor encontrado até dada iteração.

Em cada iteração, as partículas são ajustadas por dois "melhores" valores. O primeiro valor é a melhor solução que fora atingida até determinado momento e esse valor é salvo. No algoritmo, esse valor é chamado de *pbest*. O outro valor é o valor que é rastreado pelas partículas no "bando", refere-se ao obtido pela população até determinada iteração. É, em resumo, o valor ótimo global chamado de *gbest*. Com isso, após encontrar os dois valores, a partícula atualiza a sua velocidade e posição pelas equações características dos sistemas com PSO:

$$v() = v() + c1 * rand() * [pbest() - present()] + c2 * rand() * [gbest() - present()] \tag{4.22}$$

$$present() = present() + v() \tag{4.23}$$

- v() é a partícula velocidade;
- present() é a solução atual;
- pbest e gbest definidos anteriormente;
- rand() gera um número entre 0 e 1;
- c1 e c2 são fatores pré definidos no algoritmo. Representam fatores de aprendizagem.

Uma das vantagens do algoritmo PSO é que este utiliza valores reais para representar os problemas. Diferente dos AG's que precisam de um processo de tradução binária, ou operadores genéticos especiais. Como exemplo, pode-se dizer na resolução de uma equação de segunda ordem $f(x) = x1^2 + x2^2 + x3^2$, na qual as partículas podem ser representadas pelas variáveis $(x1, x2 \quad e \quad x3)$ e a função objetivo de f(x). E assim, pode-se utilizar o processo para encontrar o valor ótimo até que a máxima iteração seja atingida ou a condição de erro seja satisfeita.

Como já mostrado, não há muitos parâmetros no algoritmo, e existe um padrão no sequenciamento dos mesmos:

- **Dimensão das partículas:** Determinado pelo problema;
- **Faixa de atuação das partículas:** É também determinado pelo problema a ser otimizado;

- **Vmax:** Essa variável determina qual o tamanho da mudança que a variável sofrerá durante uma iteração. Geralmente a configuração desse parâmetro, segue como por exemplo, (x1,x2 e x3), na qual x1 está entre (-10, 10), logo Vmax=20;

- **Fatores de aprendizagem:** C1 e C2 geralmente é configurados para serem 2. Contudo, giram em torno de (0,4);

- **Condição de parada:** Representa o máximo número de iterações que o algoritmo PSO executa e o mínimo erro requerido;

- **Gbest e Pbest:** Valores de iteração determinados nos parágrafos acima. Os valores globais são mais rápidos, mas podem convergir a um ótimo local por alguns problemas. Pode-se então trabalhar nesses valores para que os valores globais achem mais rápidos os resultados e *pbest* para refina-los;

- **Peso de inércia** (*inertia weight*): É um parâmetro estudado por Shi e Eberhart (1999) que balanceia os valores globais e locais *(w)*. A Eq. (4.22) apresentada então se torna:

$$v() = v() * \mathbf{w} + c1 * rand() * [pbest() - present()] + c2 * rand() * [gbest() - present()] \quad (4.24)$$

Um grande peso de inércia facilita a busca global, enquanto um pequeno peso facilita uma busca local. Em respeito ao algoritmo, há um decréscimo linear dessa variável a medida que o curso do algoritmo se estabelece. Assim, o PSO tende em ter mais busca globais no inicio, enquanto tem mais buscas locais no final. Esse parâmetro é melhor discutido nos estudos de Shi e Eberhart (1999).

4.4 Software XFOIL

Projetos em aerofólios necessitam de um código em dinâmica do fluidos computacional, CFD, que contemple todos os parâmetros e especificações do projeto. E a escolha desses códigos estão diretamente relacionadas ao custo computacional que eles geram. Por exemplo, os códigos CFDs mais sofisticados e que detêm das equações completas de Navier-Stokes, apresentam grandes cargas de processamento. E com isso, processos de modelagem mais simples e com baixa carga computacional, na qual o objetivo é específico e limitada de aplicações e simulações.

O XFoil utiliza o método dos painéis com distribuições lineares de vórtices para a parte não-viscosa, e o método de integral da camada limite com duas equações para representar as camadas viscosas. Os pontos de transição da camada são determinados por meio de uma formulação de amplificação e^n. E o sistema de equações da camada limite, de transição e do escoamento potencial é solucionado pelo método de Newton global (DRELA, 1989).

Nesse contexto, o programa de cálculos de características aerodinâmicas XFOIL foi utilizado no processo computacional deste trabalho. Desenvolvido por Drela (1989), o programa possui em seu contexto soluções de subrotinas que formam um ambiente interativo para equacionamentos aerodinâmicos.

Têm-se 3 ambientes principais que descrevem o software:

1. Análise direta do escoamento sobre o aerofólio;
2. Para objeto inverso e misto;
3. Para manipulação de geometria de aerofólios bidimensionais, a escoamentos subsônicos, não-viscosos ou viscosos com baixo número de Reynolds

Pode-se verificar as rotinas computacionais do programa na Fig. (22). Assim, adapta-se para a versão apresentada nesse trabalho, como mostrado na Fig. (23).

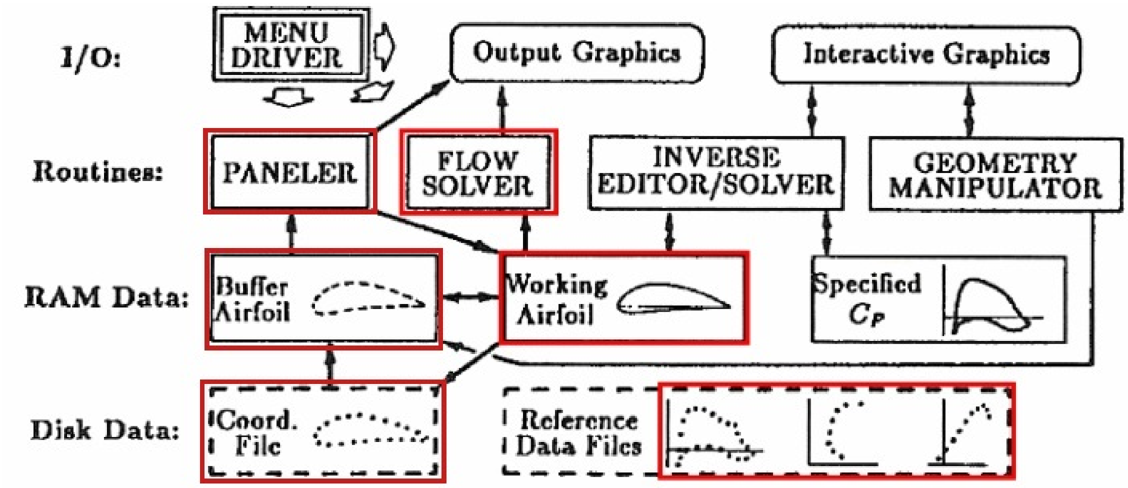


Figura 22: Fluxograma da rotina computacional original do programa XFOIL. Retirado de (DRELA, 1989)

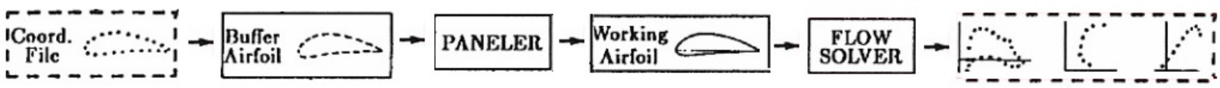


Figura 23: Fluxograma de rotina modificada do programa XFOIL. Adaptado de (DRELA, 1989)

Basicamente, as modificações no programa priorizaram a execução de variáveis específicas para o modelo do projeto, sem o modo iterativo do próprio programa (modo *batch*). O modo não-iterativo utiliza como entrada e saída arquivos que são ideais para modelos de otimização. Assim, modifica-se dois pontos principais:

1. Desativação de subrotinas que não representam o cálculo direto do escoamento;

2. Desativação da interface gráfica com o usuário.

Lida-se com o XFOIL de forma indireta por meio dos códigos de interface com o MATLAB. Por isso, as interfaces gráficas não representam grande valor no processo, pois esses representam apenas comandos de execução e resultados mostrados pelo programa. Sendo assim, há uma análise sistemática das subrotinas para focar em encontrar as rotinas de interesse, e criar um conjunto de saídas em arquivos .txt em substituição aos apresentados em tela.

E como o processo de otimização requer o XFOIL como código padrão para os cálculos do escoamento sobre uma geometria completamente definida, as rotinas de projeto inverso, de projeto misto e de modificação de geometria foram desativadas para melhor atender aos requisitos do objetivo do trabalho.

4.4.1 Modelo matemático Xfoil

O principal objetivo do desenvolvimento do software Xfoil foi diminuir os requerimentos computacionais enquanto mantivesse a funcionalidade de predizer escoamentos em número de Reynolds baixo (DRELA, 1989).

Basicamente, Xfoil trabalha com formulações viscosas e invíscidas para demonstrar a performance de um dado aerofólio em tais escoamentos, com a ajuda do método dos painéis e equações de distribuição potencial, como mostrado na Fig. (24).

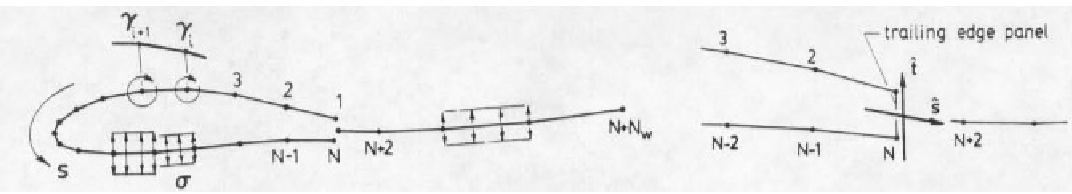


Figura 24: Método dos painéis aplicado a geometria de um aerofólio.

Fonte: (DRELA, 1989)

A formulação e o desenvolvimento de equações se concentra em 2 dimensões. E para esse caso, o campo de fluxo invíscido para um aerofólio é feito pela superposição do fluxo de correntes livres, efeito das vorticidades de força γ na superficíe da geometria e camada de origem de forças σ na superfície da geometria e esteira. Assim, retorna a Eq. (4.25):

$$\Psi(x,y) = \mathrm{u}_\infty \mathrm{y} - \mathrm{v}_\infty \mathrm{x} + \frac{1}{2\pi}\int \gamma(s)\ln \quad \mathrm{r}(\mathrm{s};x,y)\mathrm{d}s + \frac{1}{2\pi}\int \sigma(s)\theta(\mathrm{s};x,y)\mathrm{d}s \tag{4.25}$$

No qual, 's' é a coordenada ao longo do vórtice e as coordenadas fonte do aerofólio, 'r' é a magnitude do vetor entre o ponto em 's' e o campo de pontos 'x,y'; θ é o ângulo do vetor, e $\mathrm{u}_i nfty = \mathrm{q}_\infty \cos\alpha$, $\mathrm{v}_i nfty = \mathrm{q}_\infty \sin\alpha$ são as componentes de velocidade do fluxo livre.

Drela (1989) ainda afirma que os contornos e a trajetória da esteira são discretizadas em painéis planos, com 'N' nós na superfície do aerofólio, e 'N_w nós na esteira como na Fig. (4.25). Cada geometria possui uma distribuição linear de vorticidade definidos por valores de nós γ_i $(1 \leq i \leq N$. Cada painél e esteira possui forças σ_i $(1 \leq i \leq N + N_w - 1)$ associadas, e que são relacionadas com as camadas viscosas.

Para os aerofólios com painéis planos, a Eq. (4.25) avalia as expressões para as funções em qualquer ponto 'x,y'.

$$\Psi(x,y) = \mathrm{u}_\infty \mathrm{y} - \mathrm{v}_\infty \mathrm{x} + \frac{1}{4\pi}\sum_{j=1}^{N+N_w-1} \quad \Psi_j^\sigma(x,y)2\sigma_j \tag{4.26}$$

$$+\tfrac{1}{4\pi}\textstyle\sum_{j=1}^{N-1} \quad \Psi_j^{\gamma+}(x,y)(\gamma_{j+1}+\gamma_j) + \Psi_j^\sigma(x,y)2\sigma_j$$

$$+\tfrac{1}{4\pi}\Psi_j^\sigma(x,y)|\hat{s}\cdot\hat{t}|(\gamma_1+\gamma_N)$$

As equações apresentadas são descritas nas coordenadas locais do painél como visto na Fig. (25).

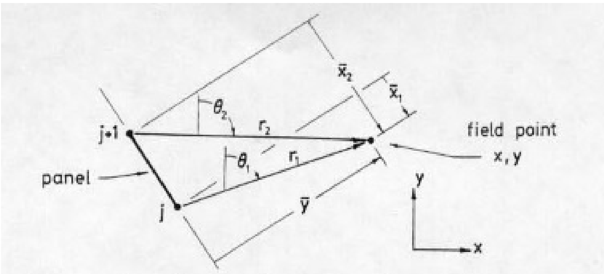


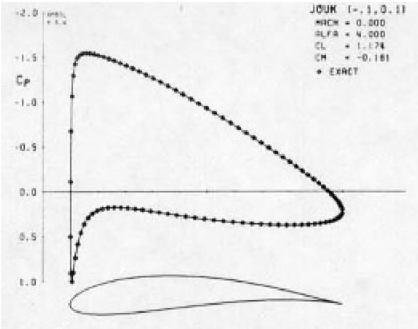
Figura 25: Coordenadas do painél locail.

Fonte: (DRELA, 1989)

O procedimento de análise invíscida é regida pelas condições de Kutta $\gamma_1 + \gamma_N = 0$. E resolvendo o sistema, retorna a solução para a vorticidade no aerofólio:

$$\gamma_i = \gamma_{0i}\cos\alpha + \gamma_{90i}\sin\alpha + \sum j = 1^{N+N_w-1} b`_{ij}\sigma_j \qquad ; 1 \leq i \leq N \tag{4.27}$$

Onde γ_0 e γ_{90} são as distribuições de vorticidade correspondentes as correntes para α de 0° e 90° e $b`_{ij} = -a_{ij}^{-1}b_{ij}$ é característica base da matriz. Configurando $\sigma_i = 0$ na expressão da Eq. (4.27), e especificando o ângulo de ataque, a solução é imediantamente encontrada (DRELA, 1989). Para fluxos viscosos, as forças σ_i não são prioridade, e então a Eq. (4.27) deve ser solucionada com equações de contorno.



CPU Requirements on MicroVAX II

N	CPU (s)	C_L Error
40	4.0	0.766 %
60	7.5	0.340 %
100	20.0	0.175 %
160	54.0	0.085 %

Figura 26: Caso teste para o aerofólio Joukowsky. N= 120 e distribuição.

Fonte: (DRELA, 1989)

Figura (26) mostra a comparação entre os valores calculados e a distribuição de pressão em um aerofólio Joukowsky. Como o fluxo dentro do aerofólio está estagnado, a velocidade na superfície é igual ao da superfície da vorticidade, levando o coeficiente de pressão para a forma $C_p = 1 - (\gamma/\mathrm{q}_\infty)^2$.

4.5 Verificação e Validação

O atesto do método utilizado é necessário para garantir a confiança dos resultados computacionais comparados aos resultados experimentais, e por isso, é necessário a validação dos códigos. A distribuição de pressão, bem como as características aerodinâmicas são verificadas comparando então, com seu respectivo modelo experimental descrito na literatura. Para tal, os estudos experimentais feitos pelo Laboratório Nacional de Energias Renováveis dos Estados Unidos (NREL) foi tomado como base. E como modelo de estudo, o aerofólio clássico para aeronaves (NACA 4415) é comparado em relação as características aerodinâmicas simulada pelo algoritmo em questão trabalhado.

Apesar da aplicabilidade de conceitos computacionais ainda é necessário testes específicos para comprovar o correto desenvolvimento dos algoritmos. Assim, o casamento dos resultados ajudam no desenvolvimento de novos produtos e novas variáveis, algo importante para esse trabalho, já que se procura a otimização de uma dada geometria.

O trabalho de (HOFFMAN; RAMSAY; GREGOREK, 1996) utilizou o aerofólio NACA 4415 para examinar as características do perfil no laboratório da universidade do estado Ohio em um túnel de vento (3x5m) sob fluxo laminar em modelo estacionário. Os testes compreenderam as faixas de Reynolds (0,75; 1; 1,25 e 1,5 milhões) para ângulos de ataque de $-10° a +40°$.

A Figura (27) representa os testes realizados em túnel de vento pelo NREL.

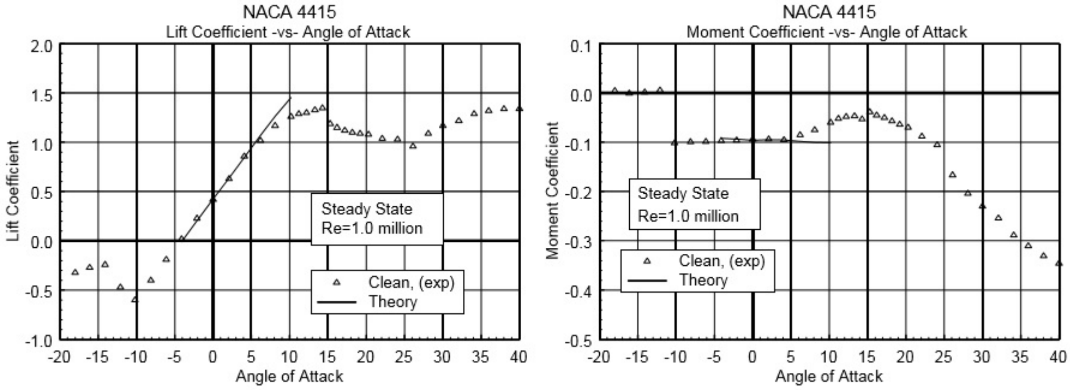


Figura 27: Testes experimentais em túnel de vento em comparação com a teoria.

Fonte: (HOFFMAN; RAMSAY; GREGOREK, 1996)

A Figura (28) representa os testes realizados com os algoritmos presentes nesse trabalho em comparação com os dados fornecidos pelo NREL.

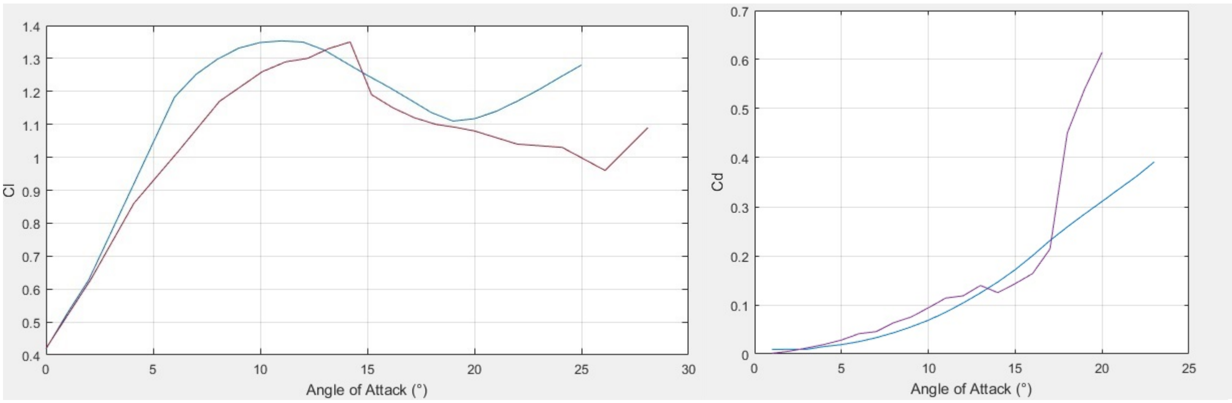


Figura 28: Caso teste NACA4415 com os algoritmos Xfoil/Matlab (AZUL). Testes experimentais NREL (ROXO)

Fonte: Autor

Essa comparação mostra que a análise pelos algoritmos usados representa, com certa acurácia, valores moderados a certos ângulos de ataque com certas relações. As características do ângulo de "stall" foram similares para ambos os casos com decaimento nas curvas de sustentação de mesmo grau. Contudo é visível uma certa valorização dos resultados computacionais em relação aos experimentais, principalmente em maiores ângulos de ataque.

Para o presente trabalho, optou-se então por trabalhar com um modelo de otimização para faixas menos acentuadas ($0°a10°$) para que não haja tanta propagação de erros como mostrados nas curvas de validação.

5 Resultados e discussões

5.1 Algoritmos de otimização e parametrização

O algoritmo em Matlab foi estabelecido de forma a seguir determinado ciclo de iterações, sendo ele compreendido pela parametrização, otimização e atesto dos resultados seguidamente. Para a inicialização do código, o perfil do aerofólio a ser otimizado deve ser inscrito em formato .dat com as coordenadas cartesianas.

O processo do levantamento das coordenadas PARSEC seguem a rotina apresentada no fluxograma mostrado na Fig. (29). Nesse processo, há o cálculo dos parâmetros polinomiais pela implementação das equações apresentadas nos capítulos passados. Três funções no ambiente matlab foram essênciais para a varredura das coordenadas do aerofólio, sendo elas, a compreensão e análise das coordenadas pelo software XFOIL, a aquisição dos 11 parâmetros requeridos para adentrar a próxima etapa da otimização e a função reversa de transformação dos parâmetros em coordenadas cartesianas. Estes passos são fundamentais para a parametrização inicial do aerofólio já que se pode presentar uma grande gama de aerofólios com poucos parâmetros, e assim, trabalhar na otimização geométrica dos mesmos com maior facilidade e agilidade.

Após a primeira etapa, adentra-se ao algoritmo de otimização descrito na representação do fluxograma na Fig. (30). Compreendido por 16 funções, nesses processos há um grande número de cálculos e processamento. A otimização durou cerca de 4 horas e 13 minutos, sendo que as primeiras iterações sempre levam mais tempo para se completarem pela grande quantidade de design ruins "randomizadas"pelos processos BSPLINE. No processo de verificação dos aerofólios pelo Xfoil, também deve-se configurar as faixas de otimização, ou seja, para o presente trabalho variou-se os aerofólios em uma faixa de 0 a 10° para o ângulo de ataque. E para um melhor aproveitamento da faixa configurada, um segundo algoritmo que chama o Xfoil foi criado, mas que inicializa a faixa a partir de 1°, fazendo com que cada processo inicialize a análise do aerofólio de dois pontos diferentes para aumento da confiabilidade dos resultados.

Para o algoritmo de otimização PSO, a função objetivo foi configurado para 10 pontos de verificação, ou seja,o Xfoil avaliará a mesma faixa de otimização de antes, a um passo de 1 grau.

Dentro do fluxograma na Fig. (30), na parte da função objetivo, o código permite identificar o "alvo"que o operador quer acertar. Para tal, 4 objetivos foram configurados, sendo 1 deles o principal e 3 secundários. Basicamente, a função tenta maximizar o objetivo principal, no caso a relação CL/CD, em detrimento dos secundários, penalizando os menores valores para CL e os aerofólios que fogem do valor minímo e máximo de espessura já ajustados.

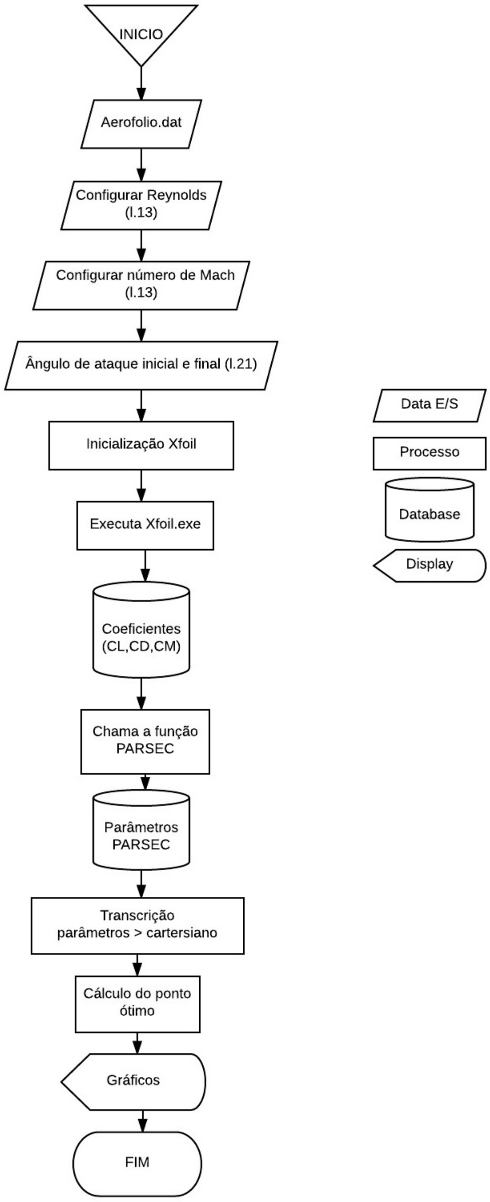


Figura 29: Rotina computacional de aquisição dos parâmetros PARSEC.

Fonte: Autor

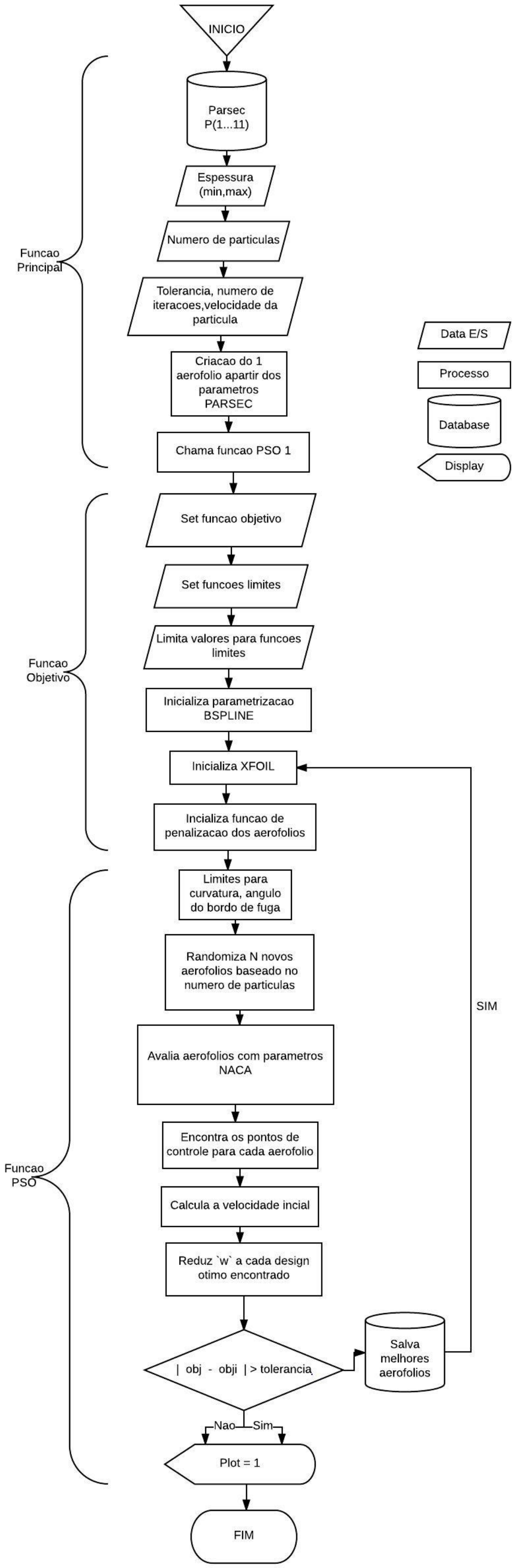


Figura 30: Rotina computacional de otimização pelo processo BSpline / PSO.

Fonte: Autor

5.2 Processo Computacional

A metodologia e a proposta utilizadas nesse trabalho buscam a otimização de um aerofólio específico mediante ao uso de algoritmos genéticos juntamente à parametrização Parsec e Bspline, como já demonstrado. A otimização pode ser configurada para vários parâmetros como coeficiente de sustentação, coeficiente de arrasto, espessura do aerofólio, dentre outros. Para esse estudo, o objetivo principal se estabelece na otimização da relação entre sustentação e arrasto. E para isso, uma rotina computacional completa foi estabelecida compreendendo os seguintes passos e exemplificado na Fig. (31):

1. Recriação do aerofólio em ambiente computacional utilizando a ferramenta CATIA.
    - Retirada de pontos cartesianos do perfil da turbina G2.
2. Obtenção dos coeficientes Parsec pelo método RMRB.
3. Integração dos parâmetros na parametrização Bspline para geração randômica de perfis.
4. Utilização do método de otimização iterativa SPO.
5. Utilização de algoritmo baseado no método de painéis com acoplamento de camada limite (Xfoil 6.9) para aquisição de coeficientes aerodinâmicos.
    - Esse estudo é utilizado junto ao método de iteração de forma a comparar o número de parâmetros na convergência dos coeficientes aerodinâmicos, bem como a verificação comparativa da otimização.

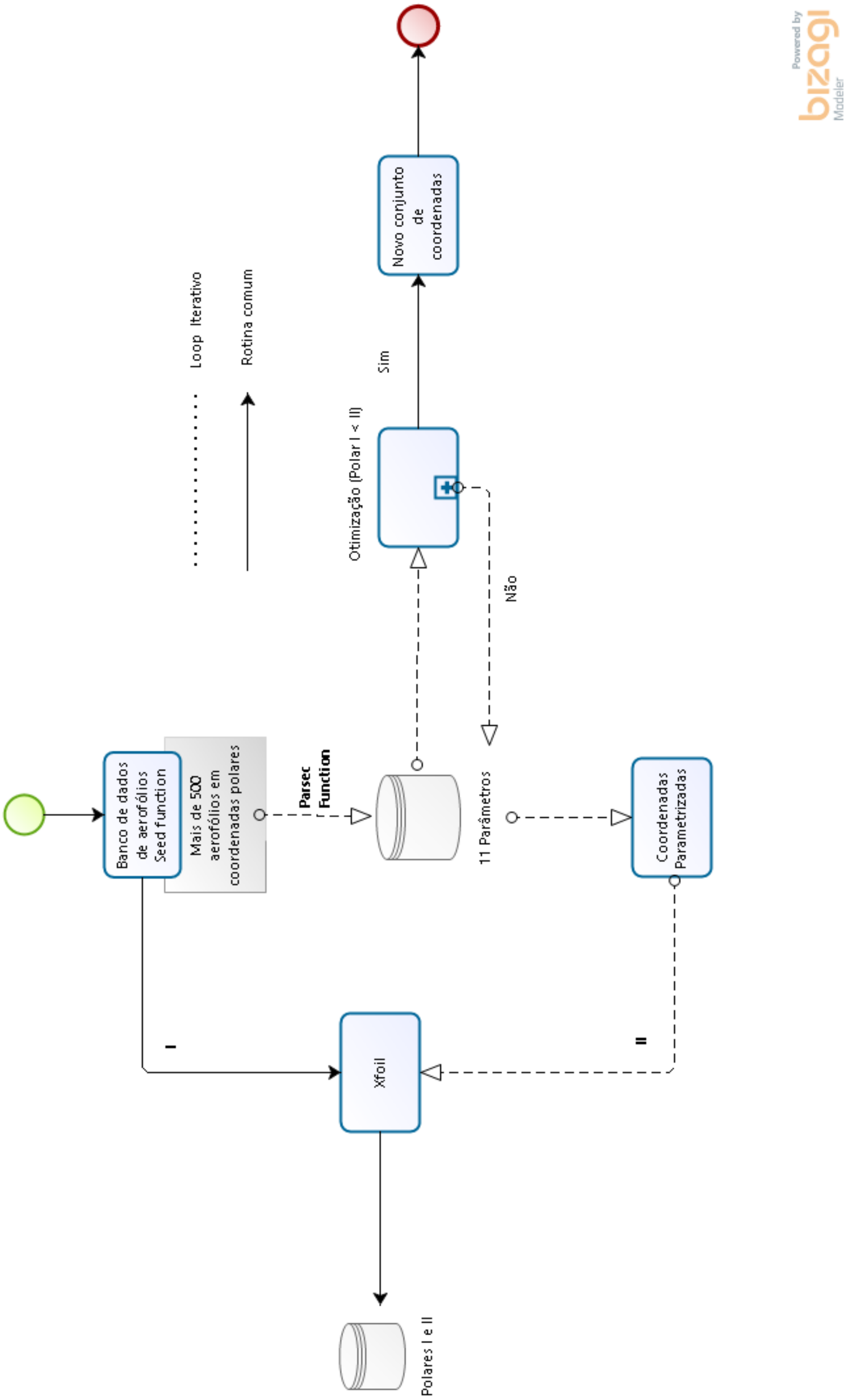


Figura 31: Rotina computacional.

Fonte: Autor

O primeiro passo é mostrada nas Figs. (32) e (33). O aerofólio original foi construído por um projeto mecânico particular, ou seja, fora feito especificamente para as características requisitadas no projeto Tucunaré.

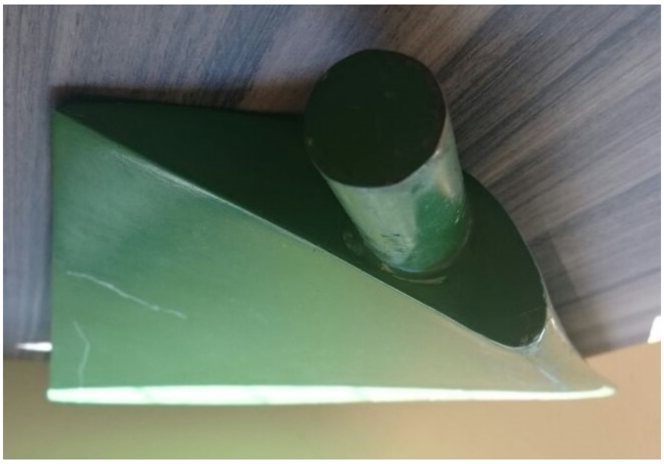


Figura 32: Aerofólio original Turbina G2 Lea.

Fonte: Autor

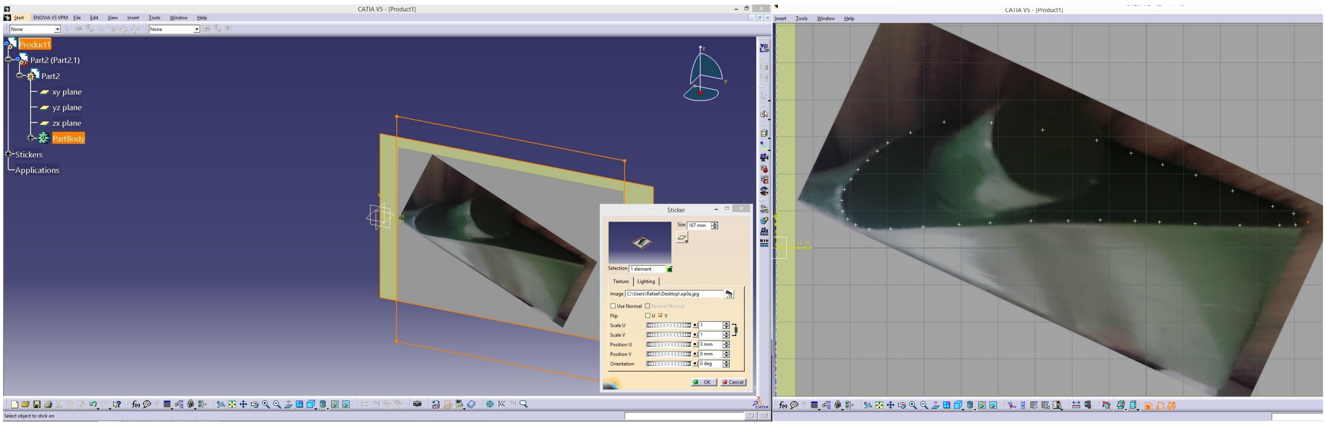


Figura 33: Retirada de pontos cartesianos pelo CATIA V5.

Fonte: Autor

O método dos painéis, utilizado no software Xfoil, possuem entradas bem definidas que ditam a execução do projeto. uma dessas entradas se baseia nas coordenadas polares do aerofólio a ser estudado. Para tal, as coordenadas cartesianas foram recriadas pelo uso da ferramenta *Shape Design* da plataforma CATIA V5, como referenciada na figura acima. Com isso, foram criados 60 pontos a fim de recriar o perfil do aerofólio sem grandes perdas geométricas. Apesar, do número elevado do controle, principalmente na curvatura do bordo de ataque, local mais suscetível a perdas estruturais e computacionais, houveram pontos de saturação que necessitaram ser suavizados. Para tal, o uso de softwares livre para estudo aerodinâmico são recomendáveis como o Java Foil, que possui funções *"smooth"* que em conjuntura ao CATIA, rearranjaram os pontos do perfil.

Fator de influência crítica neste trabalho é o estudo numérico e formulação matemática dos pontos que constroem o aerofólio. O conjunto de todas essas variáveis é traduzido nos parâmetros Parsec. A Fig. (34) mostra o registro desses parâmetros, bem como o ângulo ótimo para o ponto máximo da relação coeficiente de sustentação e coeficiente de arrasto registrado pelo Xfoil para o aerofólio original.

```
>> Main
Rodando Xfoil. Por favor, aguarde...
Analise Xfoil terminado
      Parametros Parsec:0.069561     0.31016     0.21649     -2.3293     0.063785    -0.057093      6.8274 -0.00091305    0.0016357    -0.081485

      alpha opt.:10.5
      CL    opt.:1.5638
      CD    opt.:0.0258
```

Figura 34: Registro de parâmetros Parsec.

A análise aerodinâmica é feita por meio do método dos painéis Xfoil com acoplamento da camada limite. O método consiste em um código *free software* com boa convergência em relação a resultados experimentais.

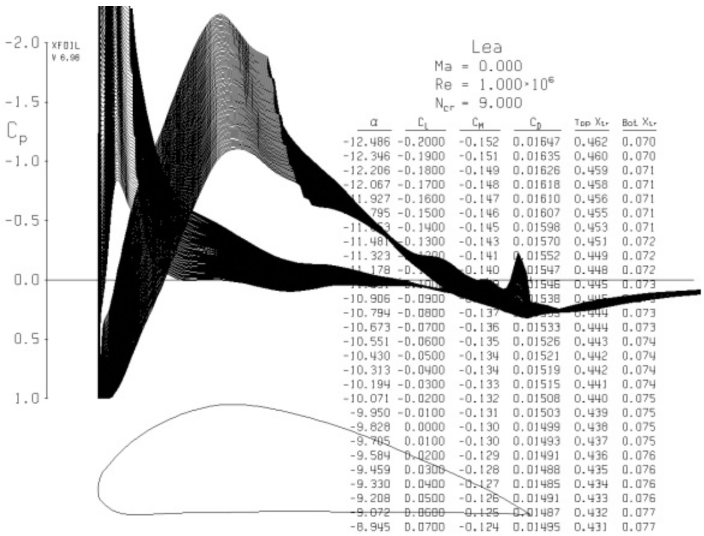


Figura 35: Método dos painéis Xfoil 6.96 sendo implementado.

Fonte: Autor

As iterações de convergência determinam o quão aerodinâmico o aerofólio atende a determinados requisitos. Pode-se, pelo software, variar diversos parâmetros de acordo com a vontade do operador sobre determinado aspecto de avaliação. Pela Fig. (35), nota-se o processo de iterações em processo, sob as derivações do software adotado, demonstrando a distribuição de pressão ao longo do perfil. Nesse aspecto, fora variados casos singulares de coeficientes de sustentação em uma faixa de $(-0.2; 1.0)$ (Segunda coluna da Fig. (35) , variando 0.01, em um ambiente de Reynolds a 10^6. Essa variação, devolve uma base grande de quais aspectos e quais faixas são necessárias para o melhor processo de otimização.

O coeficiente de arrasto também é parte integrante das forças de atrito entre a superfície do hidrofólio e o fluido. Como a água tem uma viscosidade maior que o ar, o aumento da velocidade, de zero para o valor do escoamento, ocorre em mais lentamente na região da camada limite. Na Figura (39), é possível notar o deslocamento da camada limite quando atinge determinado ângulo de ataque, precisamente 16° representado pelo brusco decaimento da curva de sustentação e no deslocamento da curva amarela na Fig. (36). Essa separação, chamada pelo termo *stall* é nítida em corpos sólidos principalmente em grandes velocidades e/ou viscosidades baixas do fluido, o que geralmente acarreta a um número de Reynolds superior. O *stall* pode ser retardado pelo atrito da parede e deslocado pelo escoamente externo devido ao efeito da viscosidade, mas principalmente retardado pelo gradiente adverso de pressão.

Para o número de Reynolds, por exemplo do escoamento em um rio, as partículas do fluido nas paredes do hidrofólio em questão possui velocidade inferior pelo efeito da viscosidade. Portanto, a energia e a quantidade de movimento das partículas não resistem ao gradiente de pressão adverso devido a um ângulo maior de ataque. E então, as partículas próximo a superfície são forçadas a seguir no sentido inverso ao fluido devido a baixa pressão local, fazendo com que o "fluido se desprenda"da parede do hidrofólio gerando um coeficiente de pressão positivo naquele ponto, acarretando perda de sustentação e turbulência na esteira.

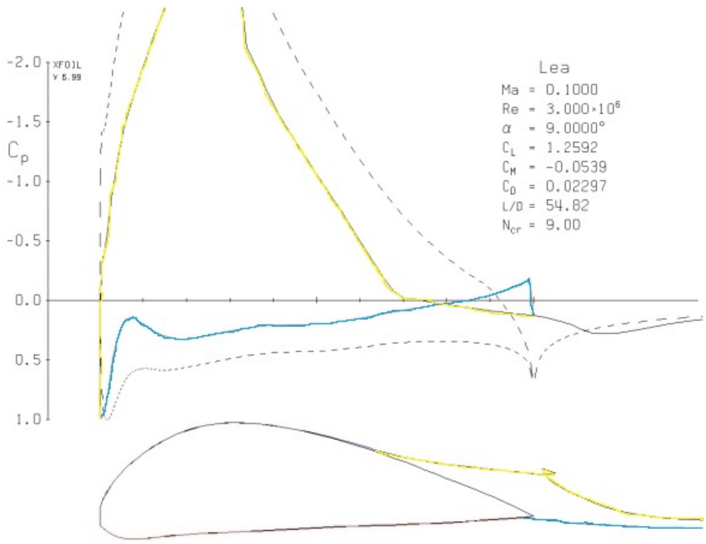


Figura 36: Processo do método dos painéis por acoplamento da camada limite.

Fonte: Autor

Da mesma forma, utilizando o método dos painéis pelo acoplamento da camada limite como ferramenta de prospecção de parâmetros aerodinâmicos, tem-se a Fig. (36). Neste caso, a configuração do software se assemelha ao processo adotado anteriormente, porém com a diferença de se trabalhar com o aspecto ângulo de ataque e o comportamento do deslocamento da camada limite que o fluido desenvolve no perfil (notado pelas cores amarela - extradorso e azul - intradorso). Neste caso, fora variados o ângulo do perfil ao eixo do fluxo em uma faixa de $(0; 10°)$, variando 1°, em um ambiente de Reynolds 3.10^6

e Mach 0.1. Pode-se notar o desenvolvimento do coeficiente de pressão ao longo do perfil do aerofólio a medida que o ângulo de ataque é variado na representação da Fig. (37) abaixo. Percebe-se quanto maior o ângulo, melhor é o coeficiente de sustentação, porém em 14°, a taxa de crescimento passa a ser negativo para o aerofólio da turbina G2 pelo deslocamento da camada limite, algo que deve ser avaliado no projeto final da turbina. Em contra passo, o coeficiente de arrasto tem um crescimento constante em todo o processo. O que termina em uma maior turbulência após o rotor das pás, principalmente com o aumento do ângulo de ataque, e consequentemente, maior pertubação na energia de saída. Assim, o hidrofólio perde a capacidade de maior aproveitamento energético, acarretando também aos problemas dos efeitos de cavitação.

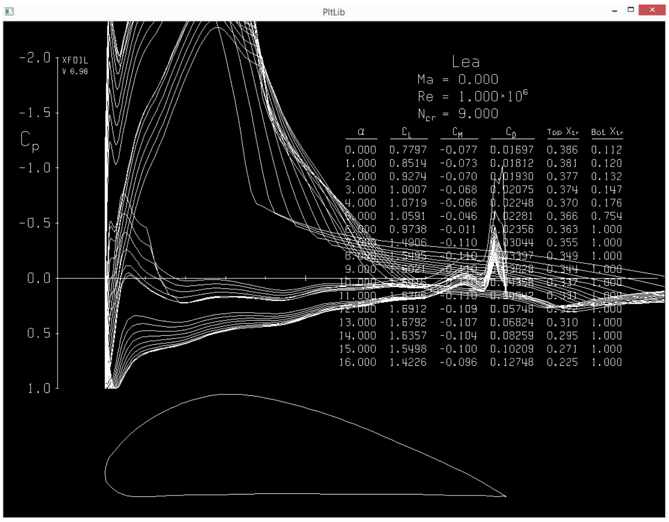


Figura 37: Processo do método dos paineis por acoplamento da camada limite para variação angular.

Fonte: Autor

Os resultados das distribuições polares da turbina G2 do laboratório de engenharia e ambiente, bem como sua representação gráfica, é então, o primeiro passo. A avaliação das coordenadas cartesianas pelo algoritmo Matlab é passo integrante da parametrização e levantamento dos parâmetros Parsec, bem como do processo de otimização. Pode-se então avaliar três disposições fundamentais na Fig. (38):

1. A primeira trata-se da representação gráfica do coeficiente de sustentação e arrasto. Em projetos de turbinas, a busca pela otimização de CL é o mais requirido, principalmente na capacidade do aerofólio em manter níveis de sustentação constantes, sem grandes variações. Nota-se que a turbina em questão é configurada para trabalhos específicos pela variação de CL em função de CD.

2. O segundo gráfico mostra a variação do coeficiente de sustentação em função do ângulo de ataque, bem como o coeficiente de momento "*pitching moment*". É fácil notar que a característica ótima para o coeficiente de sustentação é em torno de

18°, contudo representa o mínimo para o momento. Conhecer todas as variáveis denotam na melhor otimização para o projeto;

3. A terceira representação mostra em que local do perfil do aerofólio o coeficiente de sustentação atinge seu ponto máximo e ponto mínimo. É uma relação entre posição na corda "xtr"e a corda do aerofólio.

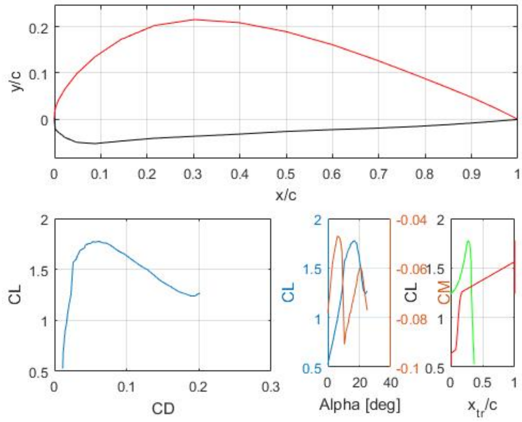


Figura 38: Avaliação das curvas polares em paralelo com algoritmos Matlab.

Fonte: Autor

O programa Xfoil retorna, além das polares do perfil, os parâmetros característicos do desempenho em matrizes de coeficientes. Pelo algoritmo, faz-se uma varredura da matriz e determina o ponto em que a relação Cl sobre CD é a maior dentre a faixa especificada, para esse processo de verificação, varia-se o ângulo de ataque do aerofólio de 0 a 25°. Para a turbina original, o ponto ótimo se encontra em um alfa (ângulo de ataque) de 10.5° representando um coeficiente de sustentação de 1.6, como pode ser visualizado na Fig. (39).

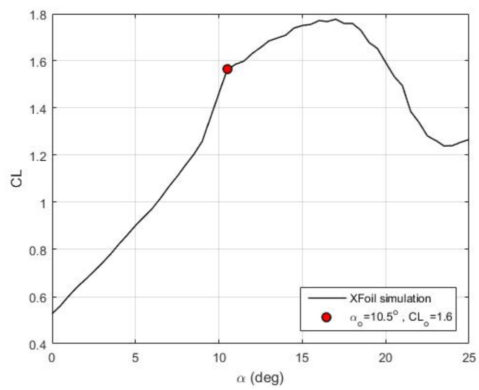


Figura 39: Representação do ponto para CL da turbina G2 com relação a relação ao ótimo Cl/Cd.

Fonte: Autor

A Figura (40) representa comparação entre a parametrização PARSEC e o aerofólio original. A representação pelos onze parâmetros desta parametrização é bastante efetiva à aerofólios da família NACA, com diferenças entre as geometrias de apenas 3 funções de base. Contudo, o aerofólio em estudo possui um perfil bastante peculiar, aproximando da família alemã Goettigen, com uma espessura bastante acentuada perto do centro aerodinâmico da pá, o que dificulta a construção exata da parametrização por esse método. Por isso, é possível notar uma variação no bordo de ataque do perfil, mas que, apesar de algumas variações, a tolerância ao erro de espessura em aerofólios em um túnel de vento chega a 0.1% é satisfeita (KULFAN; BUSSOLETTI et al., 2006). Essa característica oscilante é comum em funções minimizadas por mínimos quadrados.

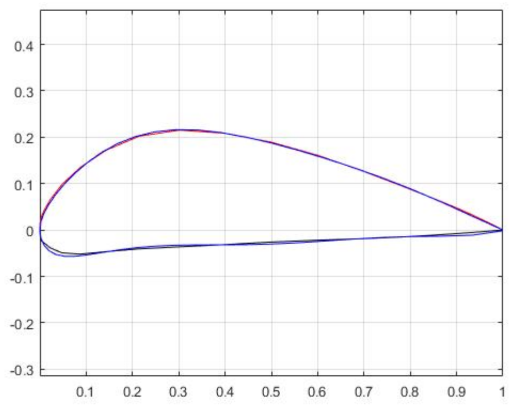


Figura 40: Comparação geométrica entre o aerofólio original e o aerofólio parametrizado.

Fonte: Autor

Por fim da primeira etapa, as 4 representações principais das características do aerofólio original podem ser visualizadas na Fig. (41).

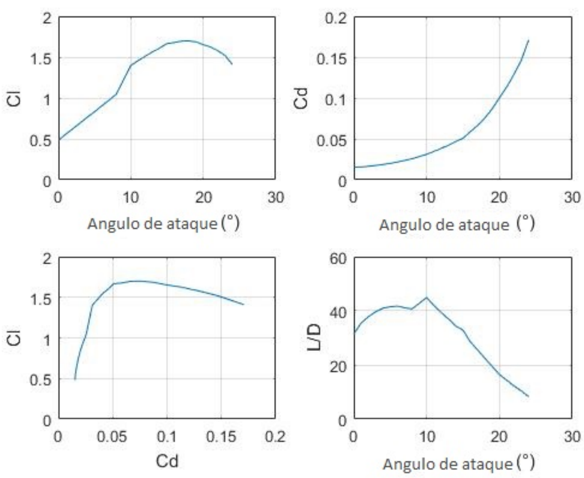


Figura 41: Características principais da Turbina G2 LEA.

Fonte: Autor

Após a obtenção dos parâmetros iniciais, a parametrização Parsec é então integrada a parametrização Bspline para então ser gerida pelo algoritmo genético PSO, de forma que o ponto de partida da otimização e da segunda parametrização fosse os 11 parâmetros do aerofólio original.

O método iterativo retorna então um mapa de amostragem para a visualização da convergência do método de otimização. A função objetivo é peculiar ao objetivo do algoritmo, portanto varia de acordo com o desejo do operador. Neste estudo, a função objetivo foi definida para maximizar a relação CL/CD, como já discutido na sessão de otimização. A otimização termina após 30 iterações atingirem sem que haja nenhuma melhora na relação. Para a otimização do aerofólio em questão, 157 iterações, como mostrado na Fig. (42), com um espaço amostral de 25 pontos foram necessários. Com isso, 4710 funções são avaliadas o quê denota 9420 formas diferentes de aerofólios avaliados pelo algoritmo já que o Xfoil roda 2 vezes para melhor acurácia.

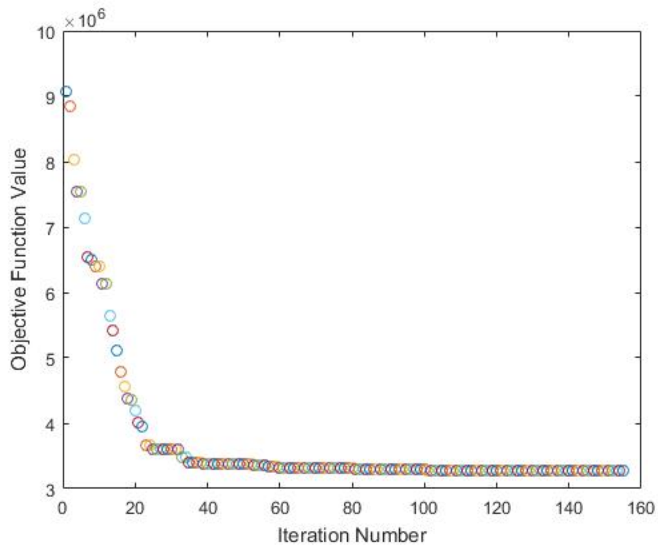


Figura 42: Mapa de iterações obtida a partir do método de otimização.

Fonte: Autor

O processo iterativo é então representado na Fig. (43). Cada uma das imagens representa um modelo diferente de aerofólio. Basicamente, o perfil base construído pela parametrização PARSEC é modificada em seus parâmetros para a geração de novos perfis. Esses perfis, então passam pelo processo de otimização, sendo que a base de funções que descrevem cada um se configura de forma a convergir para o aerofólio mais adequado ao objetivo. Nota-se que há uma certa regularidade na construção de perfis após a 40° iteração, variando pouco o perfil. Se o operador, configurar a otimização para o coeficiente de arrasto, as iterações convergem para uma espessura mínima. Neste estudo, o algoritmo e a geração de aerofólios tentam achar um equilíbrio entre CD e CL.

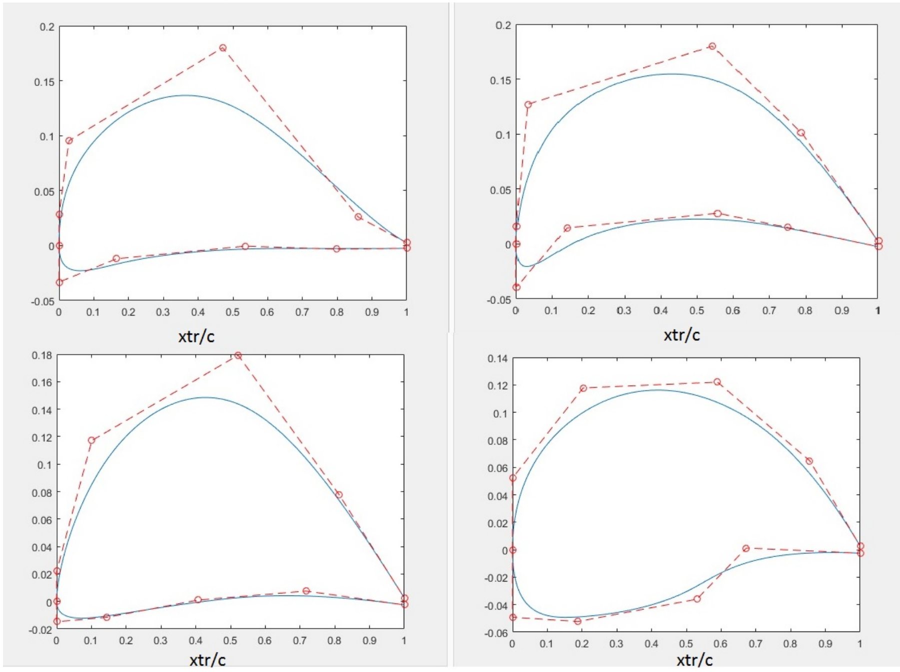


Figura 43: Processo de iteração e construção de aerofólios a partir do aerofólio base (*seed function*).

Fonte: Autor

Após as interações, novas coordenadas cartesianas são geradas com o aerofólio ótimo. Essas coordenadas passam então por um tratamento para suavização de sua curva e geração de novos parâmetros Parsec. A construção dos 11 parâmetros polares se mostrou efetiva e necessária no processo de otimização e construção de aerofólios para a metodologia apresentada. Coordenadas cartesianas detém de uma infinidade de variáveis e pontos que devem ser modificados para que se chegue a um objetivo geométrico. “Traduzir"a descrição cartesiana do perfil para um menor número de parâmetros traz uma grande vantagem computacional, bem como capacidade de manuseio do objeto em estudo.

A primeira série de testes retornou o aerofólio da Fig. (44). Nota-se pela figura, principalmente pela Fig. (46), que os resultados foram satisfatórios tanto para o objetivo especificado, quanto para a faixa de otimização apresentada. Em comparação com o aerofólio original, houve um ganho de mais de 80 unidades no quociente Cl/Cd, chegando a 126 u. contra 43 u. do perfil da turbina G2. Contudo, o ângulo de ataque ótimo para este se manteve em 10.5°, enquanto o perfil otimizado abaixou para 5°. Se compararmos os dois em pontos comuns temos a Tab. (3).

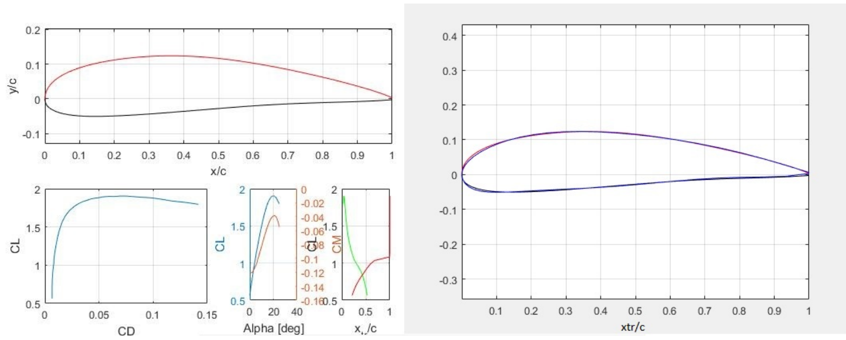


Figura 44: Características do primeiro teste do aerofólio otimizado em relação a CD, juntamente com a parametrização.

Fonte: Autor

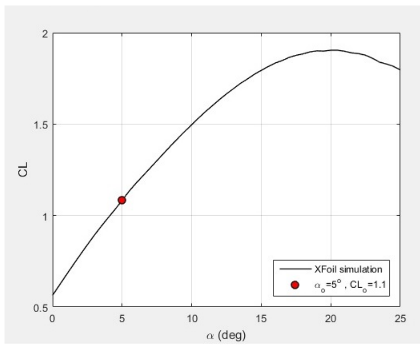


Figura 45: Ponto ótimo Cl/Cd em curva Cl x Alpha para o primeiro aerofólio ótimo.

Fonte: Autor

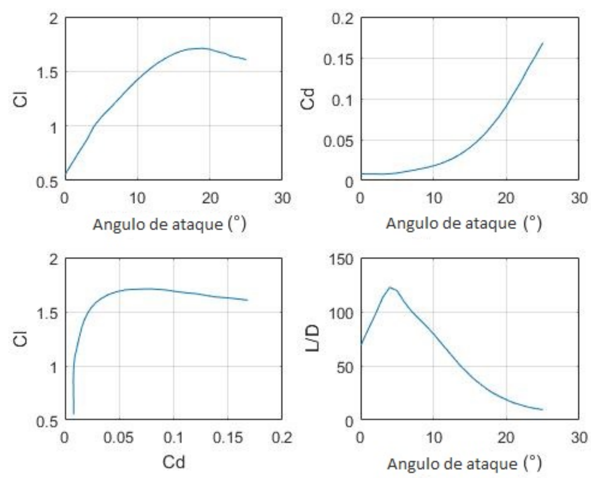


Figura 46: Características fundamentais do primeiro teste de otimização.

Fonte: Autor

Tabela 3: Comparação de Cl/Cd entre aerofólio original e otimizado para os pontos ótimos de cada.

Ângulo de ataque (°)	**Aerofólio original** (u)	**Aerofólio otimizado** (u)
10.5	*43*	*77*
5	*41*	*126*

Em contra partida, o coeficiente de arrasto teve uma taxa de crescimento bem inferior ao atual perfil, o que contribui com o aumento da relação a ser otimizado. Isso se deve ao fato de o perfil ter diminuído consideravelmente a espessura local, além de ter sido suavizado tanto o bordo de ataque, quanto o de fuga, o quê confere maior suavidade para o escoamento e otimiza o ponto de *stall*, principal questão na geração de forças adversas a sustentação. Entretanto, perfis com baixa espessura, além de deterem uma limitação de construção, apresentam boa sustentação e consequentemente maior rotação de suas pás. E essa maior rotação, leva ao aumento das velocidades tangenciais e também ao aumento das forças tangenciais, conferindo maior turbulência.

Pela Fig. (47) é possível notar as informações apresentadas do ponto ótimo, além dos parâmetros Parsec para tal aerofólio.

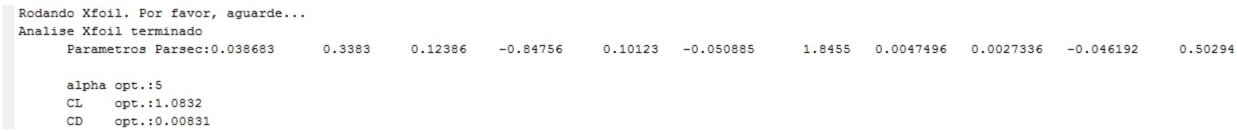


Figura 47: Parâmetros Parsec e ponto ótimo.

Fonte: Autor

Pelo primeiro teste, percebe-se pelos resultados, que o algoritmo tende a otimizar o aerofólio diminuindo a espessura do seu perfil. A diferença é bastante acentuada caso não se estabeleça condições de contorno de acordo com o perfil original (22 u de espessura - Fig. (40) contra 15 u de espessura - Fig. (48). Isso se deve ao fato, de que um dos grandes parâmetros de avaliação do algoritmo é o coeficiente de arrasto. E, como já demonstrado, as funções objetivos que estabelecem o curso da otimização. Elas ditam as penalidades que serão impostas as funções principais de otimização, e por tal, quanto mais "fino"o perfil, maior é a resistência ao deslocamento da camada limite, e consequentemente, menor o arrasto.

A Figura (48) representa o segundo teste, considerando os limites de espessura do aerofólio original. Nota-se que a espessura do bordo de fuga foi bastante alterado conferindo um arrasto baixo para ângulos de ataque pequeno. Porém, é fato que com o aumento do mesmo ângulo o arrasto tem um crescimento exponencial.

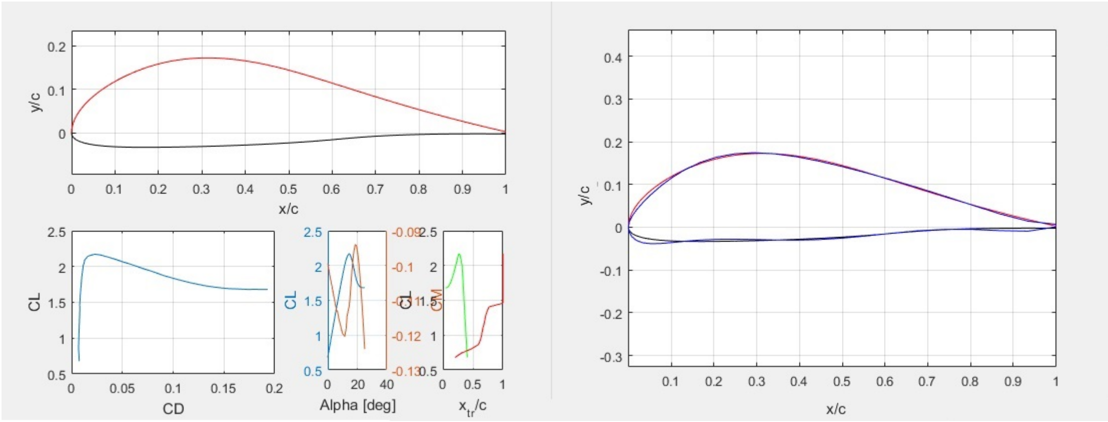


Figura 48: Características do segundo teste do aerofólio otimizado, juntamente com a parametrização.

Fonte: Autor

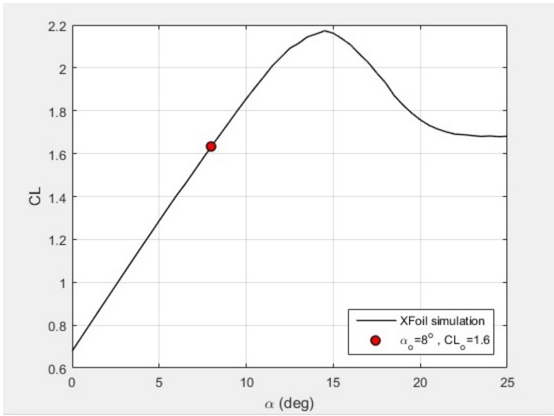


Figura 49: Ponto ótimo para a geometria do segundo perfil otimizado para a relação Cl/Cd em gráfico de Cl x Alpha.

Fonte: Autor

A Figura (51) traduz as características principais do segundo aerofólio otimizado. No algoritmo, pode-se estabelecer quais parâmetros, bem qual, quais as faixas de otimização o aerofólio será submetido. Para o presente trabalho, definiu-se o ângulo baixo $(0°, 10°)$, ou seja as iterações focarão na otimização dos parâmetros para tal faixa de valores. Percebe-se que após o limite de 10° os parâmetros começam a decrescer, principalmente o coeficiente L/D, o que mostra a eficiência e fidelidade do algoritmo para a faixa que o operador configura a otimização. Note que o coeficiente de arrasto cresce exponencialmente após a faixa, ratificando a afirmação da otimização dentro dos valores configurados.

Para a geração dos parâmetros ótimos designados, foram dispendidos 4 horas e 13 minutos. Esse valor muda dependendo da complexidade do aerofólio bem como a função objetivo apresentada pelo operador. Como já afirmado, o objetivo principal desta otimização fora a maximação do quociente CL/CD pela otimização PSO, portanto, a varredura bem como a geração do espaço amostral - ”SWARM” tendenciaram a diminuir a espessura local do aerofólio em detrimento de outros parâmetros. Todo o processo se deu em um sistema operacional regido pela plataforma Windows 8, com software Matlab 2015 e Xfoil 6.9. O hardware é regido por um processador intel i7, com 4 Gb de memória RAM.

Com isso, a comparação do aerofólio original e otimizado é descrito pelas curvas na Fig. (50), no qual é possível notar as singularidade de cada parâmetro aerodinâmico. Para a razão do coeficiente de sustentação pelo coeficiente de arrasto notou-se um crescimento abrupto e uma leve estabilização no seu pico.

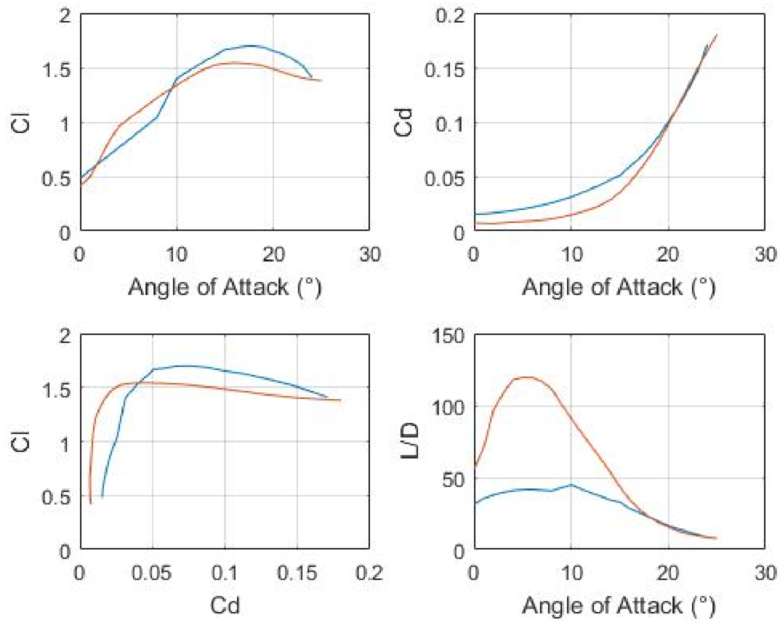


Figura 50: Sobreposição aerofólio original (Azul) e aerofólio otimizado (Vermelho).

Fonte: Autor

Sendo assim, retornando as equações que regem as forças de sustentação e arrasto, é possível saber a magnitude das forças no elemento de pá da turbina e assim notar a grande diferença, até mesmo na capacidade de gerar torques maiores.

$$L = \frac{C_L \rho \mathbf{A} \vec{W}^2}{2} \tag{5.1}$$

$$D = \frac{C_D \rho \mathbf{A} \vec{W}^2}{2} \tag{5.2}$$

Sendo ’W’ a velocidade relativa do escoamento nas pás. Adota-se a velocidade medida de 2 m/s e área determinada no projeto de 0,4825 metros quadrados e a densidade da água a 25° sendo 997,0479 kg/m^3, tem-se a Tab. (4) das forças exercidas para os pontos ótimos de cada perfil. Para o ponto ótimo, a relação Cl/Cd foi de aproximadamente 130 para o perfil otimizado contra 40 unidades do aerofólio original. Sabendo que ambos perfis mantiveram o coeficiente de sustentação em 1,6 u., o coeficiente de arrasto se configura para 0,0123 u (otimizado) e 0,04 (original).

Tabela 4: Distribuição de forças de arrasto e sustentação para perfil original e otimizado.

Forças (N)	**Aerofólio original** (u)	**Aerofólio otimizado** (u)
Sustentação	*1539,44*	*1539,44*
Arrasto	*38,486*	*11,834*

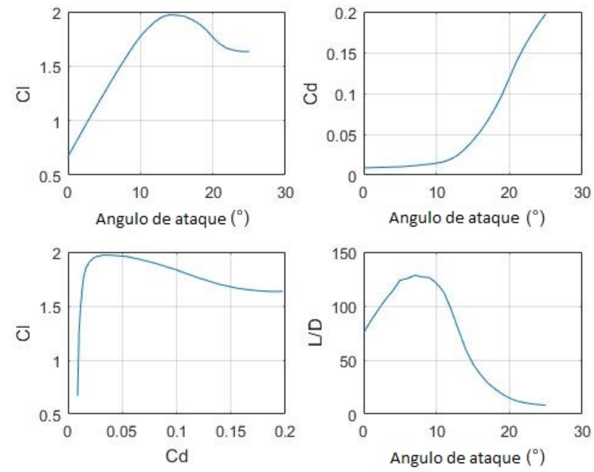


Figura 51: Características fundamentais do segundo teste de otimização.

Fonte: Autor

Por fim, pela Fig. (52) então é possível notar as informações apresentadas do ponto ótimo (ângulo ótimo para Cl/Cd e ponto ótimo para ClxCd), além dos parâmetros Parsec para tal aerofólio, sendo cada valor uma representação polar do aerofólio pelos polinômios PARSEC. Tal qual, a Fig. (53) demonstra a construção 3D do resultado final.

```
>> Main
Rodando Xfoil. Por favor, aguarde...
Analise Xfoil terminado
      Parametros Parsec:0.035315    0.29312    0.17412    -2.3054    0.38061   -0.030269    0.55926   0.0036164   0.0062773    0.14816    0.28609

      alpha opt.:8
      CL    opt.:1.6335
      CD    opt.:0.00875
```

Figura 52: Parâmetros Parsec e ponto ótimo para o segundo teste.

Fonte: Autor

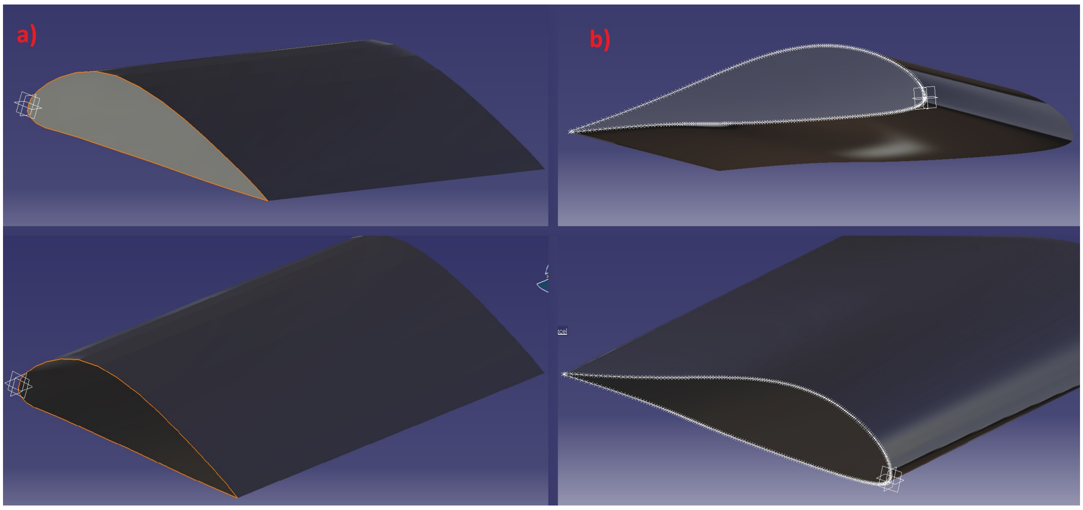


Figura 53: Parâmetros Parsec e ponto ótimo para o segundo teste. a) Aerofólio original, b)Aerofólio otimizado

Fonte: Autor

6 Conclusão

A utilização de ferramentas computacionais como uma ferramenta de importância na compreensão dos fenômenos físicos já é algo comprovado. E não distante, conclui-se que a presente metodologia de otimização e análise de aerofólios foram satisfatórias pela qualidade dos resultados gerados, como pela razoabilidade de carga computacional. O algoritmo mostrou ser poderoso no atesto de problemas inerente a construção e avaliação de projetos aerodinâmicos.

A implantação do código, bem como a interação entre seus diversos fatores tem suas dificuldades, pois lidar com programação computacional com muitas rotinas e iterações requer cuidados, principalmente por propagações de erros. Porém, verificou-se que uma divisão por funções dos algoritmos no MATLAB minimizava esse problema, pois cada qual, validava seus objetivos antes de solucionar o problema como um todo.

Em relação a parametrização, os polinômios PARSEC se mostraram bastante eficientes na descrição do aerofólio com baixos parâmetros, apesar que, diminuindo-os, alguns erros da primeira derivada para construção das curvas ainda serem vistos. Os 11 parâmetros que descrevem o aerofólio cobrem uma grande gama de famílias, apesar de que em algumas, o erro se acentua mais do que em outras. Além do mais, a representação possui uma formulação simples de ser implementada, principalmente pela busca dos coeficientes pelo método do mínimos quadrados.

A parametrização Bspline é bastante concisa em se tratando de soluções bem específicas pois utiliza de pontos de controle para a designação das curvas. E com isso, eles podem ser randomizados a partir da base Parsec para a geração de outros aerofólios com uma base estrutural parecida ao original. O fato de se utilizar essas duas parametrizações foi exatamente essa, pois a capacidade de geração de aerofólios por meio de um processo iterativo é mais flexível com o Bspline, pois este não define a geometria do perfil por características geométricas específicas, facilitando a mudança das formas no processo de otimização. Além de ter maior flexibilidade do controle sobre a superfície modificando o grau dos polinômios e o número dos pontos de controle, não existentes na parametrização Parsec.

A integração da parametrização PARSEC e BSPLINE ajudou no processo de otimização, pois o algoritmo PSO utiliza a base de aerofólios criados aleatoriamente para designar o processo de busca pelo aerofólio ótimo. Contudo, verificou-se que há um certo descompasso no processo iterativo, pois os aerofólios ruins não são descartados como nos AG's, eles ficam em uma base, e algumas vezes o algoritmo trazia eles novamente para serem testados caso não houvesse uma expressiva "punição" dos mesmos. Com isso, as primeiras iterações demoravam a ser concluídas. Contudo, esse problema pode ser parcialmente resolvido pelo emprego de um controle de viabilidade geométrica (apêndice) dos aerofólios, evitando os perfis inviáveis, e diminuindo o desperdício computacional.

Os resultados, por fim, foram satisfatórios para o objetivo do presente trabalho. Para aerofólio original, houve um ganho de mais de 80 unidades no quociente Cl/Cd, chegando a 126 u. contra 43 u. do perfil da turbina G2. Além de que o coeficiente de sustentação se manteve relativamente constante de 1.56 u para 1.63 u. Por outro lado, o ângulo de ataque ótimo diminui, de 10.5° para 8°, fazendo com que a faixa construtiva do sistema seja mais diminuta, mas que condiz com a faixa utilizada para a otimização, de 0° a 10°.

6.1 Sugestões para trabalhos futuros

Os trabalhos futuros se baseiam principalmente no atesto do método por princípios práticos, como a montagem do perfil em escala para experimento em túnel de vento. Assim, ambos aerofólios podem ser construídos e avaliados segundo suas características aerodinâmicas abordadas nesse presente trabalho, para a comparação dos métodos numéricos e princípios reais.

Porém, antes de definir os aspectos práticos, pode-se sugerir a construção dos aerofólios em outros programas computacionais de dinâmica dos fluidos como o Ansys CFX para maior robustez nos resultados. Com isso, além da análise dos coeficientes aerodinâmicas, pode-se desenvolver uma metodologia de testes estruturais tanto para o perfil de baixa espessura, quanto para aquele com restrições. Esses testes podem ser definidos para vários cenários e ajustes de variáveis, bem como esforços nas estruturas para definição da viabilidade de cada hidrofólio. Além de design 3D das pás, pode-se desenvolver a implementação da análise dos 6 sigmas para uma análise de fluxo 2D.

Estimula-se também o desenvolvimento de função de busca para a randomização de novos parâmetros PARSEC, a fim de excluir do contexto, a segunda parametrização de base Bspline. Com isso, poderá caracterizar uma maior acurácia por reduzir erros propagados na inclusão de vários polinômios para descrever um único perfil.

Referências

BERTIN, J. J.; SMITH, M. L. Aerodynamics for engineers. Prentice-Hall, 1998. Citado 2 vezes nas páginas 28 e 34.

BURTON, T. et al. *Wind energy handbook*. [S.l.]: John Wiley & Sons, 2001. Citado 4 vezes nas páginas 28, 30, 31 e 33.

CONSENTINO GARY B HOLST, T. L. Numerical optimization design of advanced transonic wing configuration. *Jornal of Aircraft*, v. 23, n. 3, p. 192–199, 1986. Citado na página 41.

CPFL. História da energia elétrica. 2015. Disponível em: <http://www.cpfl.com.br/energias-sustentaveis/eficiencia-energetica/uso-consciente/historia-da-energia/Paginas/default.aspx>. Citado na página 17.

DANTZIG, G. B. *Linear programming and extensions*. [S.l.]: Princeton university press, 1998. Citado 2 vezes nas páginas 54 e 55.

DINIZ, R. 100 anos de historia e energia. 2012. Disponível em: <http://www.100anosdehistoriaeenergia.com.br/capitulo-1/>. Citado na página 17.

DRELA, M. Xfoil: An analysis and design system for low reynolds number airfoils. In: *Low Reynolds number aerodynamics*. [S.l.]: Springer, 1989. p. 1–12. Citado 4 vezes nas páginas 62, 63, 64 e 65.

EGGLESTON, D. M.; STODDARD, F. Wind turbine engineering design. Van Nostrand Reinhold Co. Inc., New York, NY, 1987. Citado na página 35.

ELS, R. H. V. et al. Hydrokinetic propeller type turbine for the electrification of isolated householders or community and social end-users. In: *17 Congresso de Engenharia Mecânica*. [S.l.: s.n.], 2003. Citado na página 18.

GONZALEZ, F. E. Estudo das forças atuantes em mecanismos de regulação de Ângulo de passo e desenvolvimento de um sistema emulador de cargas. In: *Dissertação, Florianópolis,SC*. [S.l.: s.n.], 2012. p. 168p. Citado na página 38.

GRANT, I. Basic concepts in turbomachinery. In: *Ventus Publishing ApS*. [S.l.: s.n.], 2009. v. 1, p. 1–144. ISBN 978-87-7681-435-9. Citado na página 28.

HAU, E. *Wind Turbines: Fundamentals, technologies,application, economics*. [S.l.]: Springer Science & Business Media, 2005. 552 p. Citado 2 vezes nas páginas 33 e 38.

HOFFMAN, M.; RAMSAY, R. R.; GREGOREK, G. *Effects of grit rougness and pitch oscillations on the NACA 4415 airfoil*. [S.l.]: National Renewable Energy Laboratory, 1996. Citado 2 vezes nas páginas 65 e 66.

HU, X. *PSO Tutorial*. China, 2006. Citado na página 59.

JUNIOR, A. C. B. et al. Turbina hidrocinética geração 3. 2007. Citado 2 vezes nas páginas 18 e 20.

JUNIOR, A. P. R. A. C. B. On a mathematical modelling for hydrokinetic turbines hydrodynamics. Elsevier, New York, NY, 2010. Citado na página 18.

KIDWIND. *Advanced Blade Design*. Minessota, USA, 2013. 12 p. Citado na página 34.

KULFAN, B. M.; BUSSOLETTI, J. E. et al. Fundamental parametric geometry representations for aircraft component shapes. In: *Machine Learning: ECML-93*. [S.l.: s.n.], 2006. Citado 2 vezes nas páginas 41 e 77.

LEA. *Turbina Hidrocinética para pequenas comunidades-Aperfeiçoamento do Projeto HidrodinÂmico e Atualização do Protótipo, relatório técnico de pesquisa do projeto*. [S.l.]: ELETRONEORTE/UnB -FUB 55/2003, 2007. Citado 2 vezes nas páginas 19 e 20.

MCCOSKER, J. Design and optimization of a small turbine. In: *Master in Mechanical Engineering. An Engineering Project, Faculty of Rensselaer Polytechnic Institute*. [S.l.: s.n.], 2012. p. 56. Citado na página 38.

MEHER-HOMJI, C. B. 'the historical evolution of turbomachinery. In: *Proceedings of the 29th Turbomachinery Symposium, Texas A&M University, Houston, TX*. [S.l.: s.n.], 2000. Citado na página 17.

MIT, E. engineer department. B-spline curve. 2009. Accessed: 2016-05-16. Disponível em: <web.mit.edu/hyperbook/Patrikalakis-Maekawa-Cho/node15.html>. Citado 3 vezes nas páginas 44, 45 e 46.

NOCEDAL, J.; WRIGHT, S. *Numerical optimization*. [S.l.]: Springer Science & Business Media, 2006. Citado 2 vezes nas páginas 6 e 55.

PEREIRA, L. *Ajuste de curvas B-spline fechada com peso*. Tese (Doutorado) — UNIVERSIDADE FEDERAL DE UBERLÂNDIA, 2014. Citado 2 vezes nas páginas 48 e 49.

PIEGL, L.; TILLER, W. Curve and surface constructions using rational b-splines. 1987. Citado 3 vezes nas páginas 46, 47 e 48.

RAYMER, D. P. Aircraft design: A conceptual approach, american institute of aeronautics and astronautics. *Inc., Reston, VA*, 1999. Citado na página 21.

RODRIGUES, A. P. d. S. P. Parametrização e simulação numérica da turbina hidrocinética: otimização via algoritmos genéticos. 2007. Citado 2 vezes nas páginas 57 e 58.

ROMÁN, C. M. R. Minimum squares parsec parameters. In: SPRINGER. *Aeronautical Engineer*. [S.l.], 1993. p. 442–459. Citado na página 40.

SHI, Y.; EBERHART, R. C. Empirical study of particle swarm optimization. In: IEEE. *Evolutionary Computation, 1999. CEC 99. Proceedings of the 1999 Congress on*. [S.l.], 1999. v. 3. Citado na página 61.

SHOENBERG, I. J. Contributions to the problem of approximation of equidistant data by analytic functions. In: *IJ Schoenberg Selected Papers*. [S.l.]: Springer, 1988. p. 3–57. Citado na página 44.

SILVA, A. Briggs da. *Projeto Aerodinâmico de Turbinas Eólicas*. Tese (Doutorado) — UNIVERSIDADE FEDERAL DO RIO DE JANEIRO, 2013. Citado na página 30.

SILVA, A. J. M. *Implementaçao de um Algoritmo Genético utilizando o modelo de ilhas*. Tese (Doutorado) — UNIVERSIDADE FEDERAL DO RIO DE JANEIRO, 2005. Citado 2 vezes nas páginas 56 e 57.

SOBIECZY, H. Parametricairfoils and wings. In: *Recent Development of Aerodynamic Design Methodologies*. [S.l.]: Springer, 1998. Citado na página 42.

SOMOZA, M. H.; MACQUART, T.; MAHERI, A. Reduction of tidal turbines hydrodynamic loads employing bend-twist adaptive blades. In: IEEE. *Environmental Friendly Energies and Applications (EFEA), 2014 3rd International Symposium on*. [S.l.], 2014. p. 1–5. Citado na página 21.

SPEARS, W. M. et al. An overview of evolutionary computation. In: SPRINGER. *Machine Learning: ECML-93*. [S.l.], 1993. p. 442–459. Citado na página 56.

WHITE, F. M. *Mecânica dos Fluidos-6*. [S.l.]: McGraw Hill Brasil, 2010. Citado na página 36.

YAMAGUCHI, F. *Curves and surfaces in computer aided geometric design*. [S.l.]: Springer Science, 2012. Citado na página 48.

Apêndices

APÊNDICE A – Primeiro Apêndice

A.1 Função principal

Además das mais de 15 funções utilizadas, essas são as mais importantes no processo de otimização. Elas representam tanto as parametrizações quanto a utilização da função PSO. Abaixo seguem 3 das principais funções, sendo a primeira, a função objetivo.

```
%----------Universidade de Brasilia, campus Gama -------------------
%----------Rafael Alves da Silva Neto ------------------------------
%-----------Dept. Engenharia de Energia-----------------------------

%A funcao dessas linhas de codigos e chamar as funcoes
%secundarias, alem de localizar e executar as funcoes do Xfoil. Muitos das
%linhas que aqui estao foram modificadas de trabalhos e solucoes
%encontradas em foruns e comunidades online. Com isso, a manutencao dos
%regimes de codigos nao e algo novo, mas a integracao dos mesmo sim.

hello = XFOIL; %Integra-se o nome a variavel hello, como sendo XFOIl para eventuais
%"chamamentos" ao longo do codigo
hello.KeepFiles =false; %No processo, ha a solucao de um conjunto de iteracoes que
%retorna valores. Defina-se true ou false caso queira manter esses arquivos ou nao,
%respectivamente.
hello.Visible = true;   %Durante o processo de analise pelo Xfoil, tem-se a opcao
%de visualizar graficamente colocando true, ou false caso nao queira.

%-------------------------------------------------------------------
hello.Airfoil =  Airfoil('optfoil2.dat'); %Nessa linha, ha a busca pelo aerofolio
%desejado que deve estar dentro da pasta dos arquivos matlab em formato .dat. Entao,
% esse aerofolio eh encaminhado para a funcao Airfoil para que seja verificado.
%-------------------------------------------------------------------


hello.addFiltering(5); %Ajusta o grau smoothness das iteracoes

hello.addOperation(3E6, 0.1); %Aqui que e confugurado a parte aerodinamica
%do ambiente em que o aerofolio sera inicializado. O primeiro termo representa
%o numero de Reynolds e o segundo, o numero de Mach (mach number).

hello.addIter(100) %Pode-se tambem configurar o numero de iteracoes as
%quais o algoritmo sera 'setado'. No caso, 100 iteracoes se mostrou viavel
%para a total avaliacao do aerofolio
```

```
hello.addAlpha(0,true);

hello.addPolarFile('Polar.pol'); %O xfoil retornara um arquivo com os parametros
%polares do aerofolio. O nome do arquivo e dito nessa linha.

hello.addAlpha(0:0.5:25); %Nessa linha, ha a variacao do angulo de ataque para que
%o xfoil avalie em diferentes pontos. O inicial de 0 a 25 em um passo de 0.5.

hello.addClosePolarFile;

hello.addQuit;

hello.run %Inicializa XFOIl
disp('Rodando Xfoil. Por favor, aguarde...')

finished = hello.wait(100); %Caso XFOIl de algum problema ou dificuldade de rodar.
%O Matlab esperara 100 segundos ate fechar o programa caso nao houver resultado


if finished
    disp('Analise Xfoil terminado')

     P= parsec(hello.Airfoil.Extra,hello.Airfoil.Intra);        %As coordenadas
     % do aerofolio sao salvas em memoria RAM nos enderecos especificados.
     %Assim, essa linha chama essas coordenadas para serem integradas na
     %funcao PARSEC. A funcao entao codifica as mesmas em 11 parametros
     % polares mostradas na janela de comando do Matlab.

      strdata=num2str(P);

      st=strcat('      Parametros Parsec: ',strdata); disp(st);
      disp(' ');




     hello.readPolars;
    figure
    hello.plotPolar(1); % O numero 1 ou 0 viabiliza ou nao a representacao
    %grafica do aerofolio

    %--------------------------------
 coordenadas=parsectocoor(25,P);  % Apos gerados os parametros, ha a necessidade
 %de decodificacao das mesmas para coordenadas cartesianas a fim de serem
 % representadas graficamente e comparadas com o aerofolio original. O primeiro
 % numero representa quantos pontos cartesianos serao criados, e o P, a tomada
 % dos parametros Parsec.
```

```
 figure %Plota aerofolio original e parametrizado em um mesmo grafico

            plot(hello.Airfoil.UpperX,hello.Airfoil.UpperY,'-r');
    hold on
            plot(hello.Airfoil.LowerX,hello.Airfoil.LowerY,'-k');

     hold on
             plot(coordenadas(:,1),coordenadas(:,2),'-b');
            grid on

            axis equal
   %--------------------------------

   out=[coordenadas(:,1);coordenadas(:,2)]; %Imprime as cordenadas parametrizadas
   %e salva em memoria RAM
fid=fopen([pwd '\@XFOIL\optfoil.dat'],'w');
fprintf(fid, 'Optfoil\r\n');
fprintf(fid,'% 1.5f % 1.5f\r\n',out);
fclose(fid);
else
    hello.kill;
end
```

A.2 Função PARSEC

A primeira dita função trata da configuração PARSEC. Ela está ligada com os cálculos de matrizes representadas no corpo do trabalho e sua utilização permite codificar as coordenadas cartesianas nas 11 coordenadas polares buscadas.

```
function p=parsec(perfe,perfi)

%A funcao parsec nada mais do que utiliza do metodo dos minimos quadrados
%para codificar as coordenadas cartesianas do aerofolio em 11 parametros do
%mesmo nome. Para isso, dentro da memoria RAM, nomeou-se os termos perfe
%(para perfil extradorso) e perfi(para perfil intradorso) para armazenar as
%coordenadas cartesianas do perfil a ser codificado.

function y=pparsec(x,a)
y=a(1)*x.^(1/2)+a(2)*x.^(3/2)+a(3)*x.^(5/2)+...
a(4)*x.^(7/2)+a(5)*x.^(9/2)+a(6)*x.^(11/2);
end
function y=dparsec(x,a)
y=1/2*a(1)*x.^(-1/2)+3/2*a(2)*x.^(1/2)+5/2*a(3)*x.^(3/2)+...
```

```
7/2*a(4)*x.^(5/2)+9/2*a(5)*x.^(7/2)+11/2*a(6)*x.^(9/2);
end
function y=ddparsec(x,a)
y=-1/4*a(1)*x.^(-3/2)+3/4*a(2)*x.^(-1/2)+15/4*a(3)*x.^(1/2)+...
35/4*a(4)*x.^(3/2)+63/4*a(5)*x.^(5/2)+99/4*a(6)*x.^(7/2);
end

% Os calculos utilizados nessas linhas seguem a metodologia descrita na
% secao 4.2.3 do TCC, pois a necessidade de se encontrar os parametros para que as
%resolucoes das equacoes polinomiais sejam efetivadas. Ha a necessidade de se
% resolver a matriz que minimize o erro.
%--------------Para entao o extradorso----------------
Ae=[[mean(perfe(:,1).^3),mean(perfe(:,1).^4),mean(perfe(:,1).^5),...
mean(perfe(:,1).^6),mean(perfe(:,1).^7)];...
[mean(perfe(:,1).^4),mean(perfe(:,1).^5),mean(perfe(:,1).^6),...
mean(perfe(:,1).^7),mean(perfe(:,1).^8)];...
[mean(perfe(:,1).^5),mean(perfe(:,1).^6),mean(perfe(:,1).^7),...
mean(perfe(:,1).^8),mean(perfe(:,1).^9)];...
[mean(perfe(:,1).^6),mean(perfe(:,1).^7),mean(perfe(:,1).^8),...
mean(perfe(:,1).^9),mean(perfe(:,1).^10)];...
[mean(perfe(:,1).^7),mean(perfe(:,1).^8),mean(perfe(:,1).^9),...
mean(perfe(:,1).^10),mean(perfe(:,1).^11)]];

be=[mean(perfe(:,1).^2);mean(perfe(:,1).^3);mean(perfe(:,1).^4);...
mean(perfe(:,1).^5);mean(perfe(:,1).^6)];

ce=[mean(perfe(:,2).*perfe(:,1).^(3/2));...
mean(perfe(:,2).*perfe(:,1).^(5/2));...
mean(perfe(:,2).*perfe(:,1).^(7/2));...
mean(perfe(:,2).*perfe(:,1).^(9/2));...
mean(perfe(:,2).*perfe(:,1).^(11/2))];

%%--------------Para entao o Intradorso----------------
Ai=[[mean(perfi(:,1).^3),mean(perfi(:,1).^4),mean(perfi(:,1).^5),...
mean(perfi(:,1).^6),mean(perfi(:,1).^7)];...
[mean(perfi(:,1).^4),mean(perfi(:,1).^5),mean(perfi(:,1).^6),...
mean(perfi(:,1).^7),mean(perfi(:,1).^8)];...
[mean(perfi(:,1).^5),mean(perfi(:,1).^6),mean(perfi(:,1).^7),...
mean(perfi(:,1).^8),mean(perfi(:,1).^9)];...
[mean(perfi(:,1).^6),mean(perfi(:,1).^7),mean(perfi(:,1).^8),...
mean(perfi(:,1).^9),mean(perfi(:,1).^10)];...
[mean(perfi(:,1).^7),mean(perfi(:,1).^8),mean(perfi(:,1).^9),...
mean(perfi(:,1).^10),mean(perfi(:,1).^11)]];

bi=[mean(perfi(:,1).^2);mean(perfi(:,1).^3);mean(perfi(:,1).^4);...
mean(perfi(:,1).^5);mean(perfi(:,1).^6)];
```

```
ci=[mean(perfi(:,2).*perfi(:,1).^(3/2));...
mean(perfi(:,2).*perfi(:,1).^(5/2));...
mean(perfi(:,2).*perfi(:,1).^(7/2));...
mean(perfi(:,2).*perfi(:,1).^(9/2));...
mean(perfi(:,2).*perfi(:,1).^(11/2))];
%---------------------------------------------------------------
Ne=length(perfe(:,1));
Ni=length(perfi(:,1));

invAe=inv(Ae);
invAi=inv(Ai);

ae1=(Ne*(mean(perfe(:,2).*perfe(:,1).^(1/2))-be'*invAe*ce)-...
Ni*(mean(perfi(:,2).*perfi(:,1).^(1/2))-bi'*invAi*ci))/...
(Ne*(mean(perfe(:,1))-be'*invAe*be)+Ni*(mean(perfi(:,1))-bi'*invAi*bi));

%Com os principais parametros resolvidos pelo calculo de matrizes as equacoes 4.16
% para retorno dos coeficientes do intra e extra dorso

ae=[ae1;invAe*(ce-ae1*be)];
ai=[-ae1;invAi*(ci+ae1*bi)];

pars(1)=1/2*ae1^2;
pars(2)=fzero(@(x) dparsec(x,ae),0.5);
pars(3)=pparsec(pars(2),ae);
pars(4)=ddparsec(pars(2),ae);
pars(5)=fzero(@(x) dparsec(x,ai),0.5);
pars(6)=pparsec(pars(5),ai);
pars(7)=ddparsec(pars(5),ai);
pars(8)=sum(ae)-sum(ai);
pars(9)=1/2*(sum(ae)+sum(ai));
pars(10)=1/2*(atan(dparsec(1,ae))+atan(dparsec(1,ai)));
pars(11)=atan(dparsec(1,ai))-atan(dparsec(1,ae));

p=pars; %Salva parametros em uma string de nome 'p'.

end
```

A.3 Função PSO

A terceira função inicializa o processo de otimização PSO. Nela são descritos os passo os quais a rotina de iterações se configura. O processo de construção do algoritmo se baseia todo nessas linhas, pois é aqui que os operadores são definidos pelo usuário.

```
function [pg,minval,k,fevals]=particleswarm_naca_init(f,xmin,xmax,tbnd,varargin)
```

```
%Particle swarm modificado e avaliado

%[pg,minval,designs, bestdesigns]=particleswarm(f,xmin,xmax,...) retorna
%   os melhores design de variaveis, a partir das funcoes objetivos ja descritas, a
%   matriz salva todos os designs para cada iteracao, e de fato contem o melhor
%   design para cada iteracao.
%[pg,...]=particleswarm(f,xmin,xmax) usa os valores iniciais para cada
%   restricao configuradas na funcao principal, plotando opcoes, o tamanho
%    da populacao, a tolerancia de convergencia,  e o numero de iteracoes
%   realizadas para validar o criterio de parada. f, no caso, e definido
%   para o usario (nesse caso a funcao objetivo). xmin e xmax sao vetores
%   do valores minimos e maximos considerados para o design de variaveis.
%   Os minimos e maximos dos designs nao sao restricoes da solucao, mas
%   podem sofrer mudancas com o argumento 'const'.
%[pg,...]=particleswarm(f,xmin,xmax,const) permite o usuario restringir a
%   busca para serem entre xmin e xmax, Se const=1, a restricao e seguida. Se
%   const e configurado para qualquer numero diferente de 1, a otimizacao
%   buscara fora dos limites configurados.
%[pg,...]=particleswarm(f,xmin,xmax,tbnd,const,plottingopt) permite que o
%   usuario eleja ou nao a amostragem do processo grafico. Configurando
%   plottingopt igual a 1 mostra o progresso. Caso contrario =0, subtrai.
%[pg,...]=particleswarm(f,xmin,xmax,tbnd,const,plottingopt,pop) permite o
%   usuario definir o tamanho da populacao. Para o caso, 25 designs.
%[pg,...]=particleswarm(f,xmin,xmax,const,tbnd,plottingopt,pop,tol) tambem
%   permite que o usuario defina a tolerancia no criterio de convergencia. O
%   valor inicial e 1e-5.
%[pg,...]=particleswarm(f,xmin,xmax,tbnd,const,plottingopt,pop,tol,S)
%   tambem permite ao usuario dinfir quantas iteracoes o programa deve seguir
%   fora da tolerancia ate que pare. O inicial para este caso e 20.
%[pg,...]=particleswarm(f,xmin,xmax,tbnd,const,plottingopt,pop,tol,S,vmax)
%   impoe a velocidade maxima que a particula dado por vmax (escalar) - deve ser
% entre 1%-2% da velocidade para se atravessar todo o dominio amostral.

% Avalia entradas opcionais no comando 'varargin'

global P

switch size(varargin,2)
    case 0
        const=1; plottingopt=0; pop=25; tol=1e-5; S=20; vmax=0;
    case 1
        const=varargin{1,1};
        plottingopt=0; pop=25; tol=1e-5; S=20; vmax=0;
    case 2
        const=varargin{1,1};
        plottingopt=varargin{1,2};
        pop=25; tol=1e-5; S=20; vmax=0;
```

```
    case 3
        const=varargin{1,1};
        plottingopt=varargin{1,2};
        pop=varargin{1,3};
        tol=1e-5; S=20; vmax=0;
    case 4
        const=varargin{1,1};
        plottingopt=varargin{1,2};
        pop=varargin{1,3};
        tol=varargin{1,4};
        S=20; vmax=0;
    case 5
        const=varargin{1,1};
        plottingopt=varargin{1,2};
        pop=varargin{1,3};
        tol=varargin{1,4};
        S=varargin{1,5};
        vmax=0;
    case 6
        const=varargin{1,1};
        plottingopt=varargin{1,2};
        pop=varargin{1,3};
        tol=varargin{1,4};
        S=varargin{1,5};
        vmax=varargin{1,6};
end

%Troca a coluna de vetores e avalia por eventuais erros nas restricoes
if size(xmin,2)>1&&size(xmin,1)==1
    xmin=xmin';
elseif size(xmin,2)>1&&size(xmin,1)>1
    error('xmin must be a vector.')
end
if size(xmax,2)>1&&size(xmax,1)==1
    xmax=xmax';
elseif size(xmax,2)>1&&size(xmax,1)>1
    error('xmin must be a vector.')
end

% Avalia o numero de de designs
n=size(xmin,1);

%Define as configuracoes do PSO
c1=1.2;             %O fator de confianca para a particula
c2=1.2;             %Fator de confianca para a populacao
whigh=1.4;          %Parametro de inercia inicial
wlow=0.6;           %Parametro de inercia final
```

```
convrate=0.050;      %Taxa de reducao para parametro de inercia
maxit=1000;          %Iteracoes maximas
minatr=0;

%Inicializacao das iteracoes e o string de melhores valores
k=0;
minvalstore=zeros(1,S+1);
errstore=ones(1,S);

%Limites para os designs baseados nos Parametros NACA
camb=[-0.02 0.06];   %min e max envergadura
cambloc=[0.1 0.7];   %min e max local da max envergadura
teang=[-5 10];       %min e max para os angulos do bordo de fuga
nacamin=[camb(1);cambloc(1);tbnd(1);teang(1)];
nacamax=[camb(2);cambloc(2);tbnd(2);teang(2)];

%Cria a populacao inicial de designs e velocidades comecando pelo aerofolio
%original parametrizado
DV=zeros(n,pop);
vel=zeros(n,pop);
objval=zeros(1,pop);
for i=1:pop
    %Cria designs de aerofolios NACA aleatoriamente
    nacaparams=nacamin+rand(4,1).*(nacamax-nacamin);
    %Encontra os pontos de controle lacias para integrar aos designs
    DV(:,i)=matchfoils_f(nacaparams);

    %Configura a velocidade
    vel(:,i)=(rand(n,1)-0.5).*(xmax-xmin);  %A velocidade inicial pode ser positiva
    %ou negativa

    %Avalia os designs
    objval(i)=f(DV(:,i));
end

%Cria a matriz dos melhores designs e vetores para os melhores resultados
p=DV;
minvals=objval;

%O melhor global ate entao
[minval,loc]=min(objval);
pg=DV(:,loc);

%Inicializa otimizacao
converged=0;
w=whigh;
while converged==0
```

```
%Os valores minimos para as ultimas S iteracoes sao salvas
%Joga fora os resultados das  ultimas Sth
%faz os mesmos para os erros
for j=1:S
    minvalstore(j)=minvalstore(j+1);
    if j<S     %(errstore tem apenas S entradas em vez de S+1)
        errstore(j)=errstore(j+1);
    end
end

%Aumenta a contagem de iteracoes
k=k+1;

%Atualiza cada posicao das particulas
%Avalia a funcao objetivo para todos os designs
%Atualiza os melhores designs encontrados para cada particula se apropriado
for i=1:pop
    if vmax>0       %impoe limites na velocidade
        if norm(vel(:,i))>vmax
            vel(:,i)=vmax*vel(:,i)/norm(vel(:,i));
        end
    end
    DV(:,i)=DV(:,i)+vel(:,i);
    if const==1;    %Traz DVs de volta para as restricoes e configura as
    % velocidades para direcoes contrarias
        for j=1:n
            if DV(j,i)<xmin(j)
                DV(j,i)=xmin(j);
                vel(j,i)=-rand(1)*vel(j,i);
            elseif DV(j,i)>xmax(j)
                DV(j,i)=xmax(j);
                vel(j,i)=-rand(1)*vel(j,i);
            end
        end
    end
    objval(i)=f(DV(:,i));
    if objval(i)<minvals(i)
        minvals(i)=objval(i);
        p(:,i)=DV(:,i);
    end
end

%Atualiza o melhor design global se apropriado
[it_min,loc]=min(objval);
if it_min<minval
    pg=DV(:,loc);
    minval=it_min;
```

```
end
minvalstore(S+1)=minval;

%Avalia as funcoes erro para esta iteracao ate a ultima
errstore(S)=abs((minvalstore(S+1)-minvalstore(S))/minvalstore(S));

%Avalia a convergencia
if max(errstore)>tol&&k<maxit % Permite maxit iteracoes antes de forcar a saida
    converged=0;
else
    converged=1;
    if k==maxit
        disp('The optimizer was forced to exit due to the max number of
        iterations (maxit) being met.')
    end
end
if minatr~=0&&minval>=minatr&&k<maxit    %Nao permite a saida antes que um
%design factivel seja encontrado, ao menos que maxit iteracoes seja alcancado
    converged=0;
end

%Plota o progresso de minimizacao
if plottingopt==1
    figure(1),plot(k,minval,'o'),xlabel('Iteration Number'),
    ylabel('Objective Function Value')
    if minatr==0
        hold on
    elseif minatr~=0&&minval<minatr
        hold on
    end
end

%Atualiza a velocidade de cada particula
for i=1:pop
    if minatr~=0&&minval>=minatr        %Se o design factivel e encontrado,
    %utiliza menores valores de confianca
vel(:,i)=w*vel(:,i)+c1/2*rand(n,1).*(p(:,i)-DV(:,i))+c2/2*rand(n,1).*(pg-DV(:,i));

else

vel(:,i)=w*vel(:,i)+c1*rand(n,1).*(p(:,i)-DV(:,i))+c2*rand(n,1).*(pg-DV(:,i));
    end
end

%Reduz w se o design e encontrado
if minatr==0||minval<minatr
    w=w-convrate*(w-wlow);
```

```
    end

end
%Numero de funcoes avaliadas
fevals=(k+1)*pop;

if plottingopt==1
    hold off
end
```