

OTIMIZAÇÃO POR ALGORITMO GENÉTICO EM ASAS DE AERONAVES CARGUEIRAS RADIO CONTROLADAS PARA COMPETIÇÕES DE AERODESIGN

LAURA VITÓRIA DOS SANTOS MATOS¹, LEONARDO MARTINS PESSOA², DIEGO BANDEIRA DE MELO AKEL THOMAZ³, PEDRO TIAGO SILVA MELO⁴ e RENATA DA ENCARNAÇÃO ONETY⁵

¹Graduanda em Engenharia de Materiais, UEA, Manaus-AM, lvdsm.emt19@uea.edu.br;

²Graduando em Engenharia Mecânica, UEA, Manaus-AM, imp.eng17@uea.edu.br;

³Graduando em Engenharia Mecânica, UEA, Manaus-AM, diegoakel1@gmail.com;

⁴Graduando em Engenharia Mecânica, UEA, Manaus-AM, ptsm.eng19@uea.edu.br;

⁵Dra. em Engenharia Elétrica, Profa. Adjunta UEA, Manaus-AM, ronety@uea.edu.br

Apresentado no
Congresso Técnico Científico da Engenharia e da Agronomia – CONTECC
15 a 17 de setembro de 2021

RESUMO: Este trabalho foi desenvolvido com o intuito de propor uma solução para a escolha de um modelo de asa a ser utilizado em competições de aerodesign. Anteriormente, todo o processo era realizado ao se propor um único modelo de asa e efetuar testes sobre ele. Isso decorria da falta de experiência e tempo para análise de outros modelos por conta da competição aerodesign. Porém, ao se fazer isso, não havia garantia de se encontrar um modelo mais favorável às condições que eram exigidas segundo o regulamento da competição SAE BRASIL. O algoritmo proposto tem o objetivo de realizar a análise de diversos modelos de asas e fazer a combinação dos mesmos na busca de um modelo que mais se adeque aos requisitos desejados, de forma automatizada.

PALAVRAS-CHAVE: Aerodesign, Algoritmo Genético, Modelos de asa.

OPTIMIZATION BY GENETIC ALGORITHM IN RADIO CONTROLLED CARGO AIRCRAFT WINGS FOR AERODESIGN COMPETITIONS

ABSTRACT: This work was developed in order to propose a solution for the choice of a wing model to be used in aerodesign competitions. Previously, the entire process was performed by proposing a single wing model and testing it. This was due to the lack of experience and time to analyze other models due to the aerodesign competition. However, in doing so, there was no guarantee of finding a model more favorable to the conditions that were required under the SAE BRASIL competition regulations. The proposed algorithm aims to carry out the analysis of several wing models and to combine them in the search for a model that best suits the desired requirements, in an automated way.

KEYWORDS: Aerodesign, Genetic Algorithm, Wing models.

INTRODUÇÃO

Em meados do século XIX, os naturalistas acreditavam que cada espécie havia sido criada separadamente através de geração espontânea ou por um ser supremo. O trabalho do naturalista Carolus Linnaeus sobre a classificação biológica de organismos despertou o interesse pela semelhança entre certas espécies, levando a acreditar na existência de certa relação entre elas. Outras escritas influenciaram os naturalistas em direção à teoria da seleção natural, tais como os de Jean Baptiste Lamarck, que sugeriu uma teoria evolucionária no “uso e desuso” de órgãos; e de Thomas Robert Malthus, que até propôs que fatores ambientais tais como doenças e carência de alimentos, limitavam o crescimento de uma população (Mathias, 2015).

Depois de mais de 20 anos de observações e experimentos, Charles Darwin apresentou, em 1858, sua teoria de evolução através de seleção natural, sincronicamente com outro naturalista inglês,

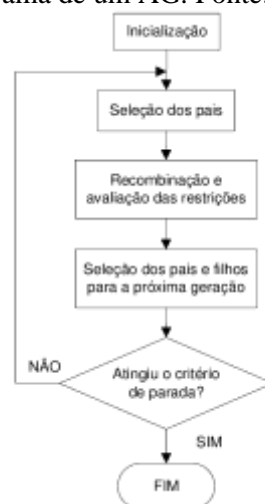
Alfred Russel Wallace. No ano seguinte, Darwin publica o seu “On the Origin of Species by Means of Natural Selection” com a sua teoria completa, sustentada por diversas evidências colhidas durante suas viagens a bordo do Beagle (Darwin, 1859).

A rica teoria da evolução faz a combinação entre a genética e as ideias de Darwin e Wallace sobre a seleção natural, criando o princípio básico de Genética Populacional: a variabilidade entre indivíduos em uma população de organismos que se reproduzem sexualmente é produzida pela mutação e pela recombinação genética. Este princípio foi desenvolvido durante os anos 30 e 40 por biólogos e matemáticos.

Nas décadas seguintes, vários biólogos começaram a gerar simulações computacionais envolvendo sistemas genéticos. No entanto, foi John Holland quem começou a desenvolver as principais pesquisas no tema. No início, a ideia de Holland era tentar imitar certas etapas do processo de evolução natural das espécies incluindo-as em um algoritmo computacional. Em 1975 Holland publicou o seu livro “Adaptation in Natural and Artificial Systems”, que muitos consideram a Bíblia de Algoritmos Genéticos (Secchi, 2018).

Os Algoritmos Genéticos (AG) são parte de uma técnica de busca utilizada na ciência da computação para achar soluções aproximadas em problemas de otimização e busca. Os AG são implementados como uma simulação de computador em que uma população de representações abstratas de solução é selecionada em busca de soluções melhores. A evolução geralmente se inicia a partir de um conjunto de soluções criado aleatoriamente e é realizada por meio de gerações. A adaptação de cada solução na população é avaliada, alguns indivíduos são selecionados para a próxima geração e recombinados ou mutados para formar uma nova população. A nova população então é utilizada como entrada para a próxima iteração do algoritmo (Koza, 1992).

Figura 1. Fluxograma de um AG. Fonte: Corrêa et al., 2016.



Alguns dos componentes dos algoritmos genéticos são: o indivíduo, a seleção, a reprodução e a função-objetivo. O indivíduo é um portador do código genético, que é uma representação do espaço de busca do problema a ser resolvido. Por exemplo, para otimizações em problemas cujos valores de entrada são inteiros positivos menores que 255, podemos usar 8 bits, dígito binário, com a representação binária normal. Esses problemas com múltiplas entradas podem combinar as entradas em uma única sequência de bits, ou trabalhar com mais de um “cromossomo”, cada um representando uma das entradas. O código genético deve ser uma representação capaz de evidenciar todo o conjunto dos valores no espaço de busca, e precisa ter tamanho finito (Widmaier, 2005).

A seleção é outra parte chave de um algoritmo. No geral, usa-se o algoritmo de seleção por “roleta”, onde os indivíduos são ordenados de acordo com a função-objetivo e lhes são atribuídas probabilidades decrescentes de serem escolhidos. A escolha é feita aleatoriamente de acordo com as probabilidades. Dessa forma, é possível escolher os pais mais adaptados sem deixar de lado a diversidade dos menos adaptados (Goldberg, 1989).

A reprodução, tradicionalmente, é dividida em três etapas: acasalamento, recombinação e mutação. O acasalamento é a escolha de dois indivíduos para se reproduzirem. A recombinação é um

processo que limita o meio biológico homônimo na reprodução sexuada: os descendentes recebem em seu código genético parte do código do pai e parte do código da mãe. A mutação tem como objetivo permitir maior variabilidade genética na população, impedindo que a busca fique estagnada em um mínimo local (Mitchell, 1997).

Uma função-objetivo que é o objeto de uma determinada otimização, pode ser um problema de otimização ou um conjunto de testes para identificar os indivíduos mais aptos.

A codificação é a primeira questão a ser feita quando começamos a resolver um problema utilizando os AG. A codificação depende muito do problema e existem algumas que já foram aplicadas com bastante sucesso, como a codificação binária, codificação por permutação, codificação de valores e a codificação em árvore (Obitko, 1998).

MATERIAL E MÉTODOS

A princípio, o algoritmo desenvolvido possui três etapas: Seleção, Cruzamento e Mutação. O AG foi aplicado em Python por membros da equipe Urutau Aerodesign pertencente à Universidade do Estado do Amazonas e que participa anualmente da competição SAE BRASIL Aerodesign, classe Advanced.

Na seleção, se escolhe quais indivíduos serão utilizados para combinação. Esse algoritmo seletor consegue ter a percepção de todas as asas da população. Ele calcula o valor de pontuação e assim ranqueia as asas em ordem decrescente de pontuação (Matos, 2009).

As regras definidas que o algoritmo irá executar, são as seguintes:

1. Se temos um ranking com 50 asas, a 1° irá combinar com a 50°, ou seja, a melhor asa combinaria com a pior. Isso é importante por adicionar variabilidade genética aos novos indivíduos. Se acontecer o cruzamento somente entre as melhores asas, o resultado ficaria preso à um só tipo de asa, o que não pode ser um ótimo global e sim um ótimo local.
2. Se a população de pais for um número ímpar (49), o algoritmo iria gerar mais um pai aleatoriamente para fechar em um número de pares. Com 50 pais, é possível obter 25 pares.
3. Cada par irá gerar 3 filhos. Foi determinado dessa forma com a preocupação da população entrar em extinção. Através das eliminações pelas asas inválidas e pelas restrições estruturais, poderia ocorrer perda de muitos indivíduos. Mas como não há certeza do número exato de perdas, é necessário ser testado mais vezes e com verificações mais específicas.
4. A cada geração, será salvo 90% dos melhores. Por exemplo: temos 50 novas asas. A análise dessas asas será feita em um software de análise aerodinâmica e dinâmica de voo de aeronaves rígidas e de configuração arbitrária, no AVL (Athena Vortex Lattice), e será utilizado a função de pontuação da competição SAE BRASIL Aerodesign. Com isso, ocorrerá o ranqueamento de 1° a 50°, dessa quantidade só será salva a 1° colocação até a 45° e o que sobrar será descartado. Para se salvar todas as asas, depende-se do número de indivíduos que será gerado, do tamanho e da qualidade do banco de dados de asas. A função de pontuação da competição é informada no regulamento que fica disponível no site, anualmente. A função de 2020 é a seguinte:

$$Pcp = 8.3 \times \exp (CP/6), \quad (1)$$

Onde:

Pcp: pontos obtidos;

CP: carga paga total;

exp (x): função que retorna o número de Euler elevado a x

5. Esses 90% melhores salvos já irão entrar para a próxima rodada de cruzamento. Eles serão usados para gerar novos filhos.
6. Os ciclos de cruzamento serão testados indefinitivamente até que seja alcançado o parâmetro de parada com o número de asas desejadas. Por exemplo: se o usuário adicionar que o parâmetro de parada são 15 asas com pontuação >50, então o programa será executado continuamente até que existam 15 asas com pontuação igual ou maior que 50. Estas asas serão mostradas para o usuário no final, já ranqueadas e em forma de uma lista.

Já na mutação, o objetivo é mudar um pequeno valor no indivíduo, podendo modificar até mesmo um único bit. Essa pequena alteração pode ser capaz de evitar os mínimos locais do algoritmo (Bortolote, 2017) (Kurcewicz, 2017).

A partir da seleção dos pares para realizar as combinações, cada indivíduo passa por uma função que transforma seus respectivos valores em números binários. Finalizada essa operação, os números resultantes são divididos ao meio. A primeira parte de uma numeração será unida à parte final da outra. Assim, são gerados dois novos filhos (representados ainda por números binários) que posteriormente são convertidos em valores reais (Araújo e Costa, 2017) (Freitas, 2012).

RESULTADOS E DISCUSSÃO

Para obter resultados rápidos, foram utilizados valores de envergadura (comprimento da asa) e corda aleatórios para os testes, a figura 3 representa essas variáveis geométricas. O resultado desses primeiros indivíduos aleatórios e dos seus filhos estão representados na Tabela 1. O Algoritmo foi todo desenvolvido em linguagem Python e trabalhado no terminal VSCode (Visual Studio Code).

Figura 3. Ilustração da Asa e suas Variáveis. Fonte: Autor.

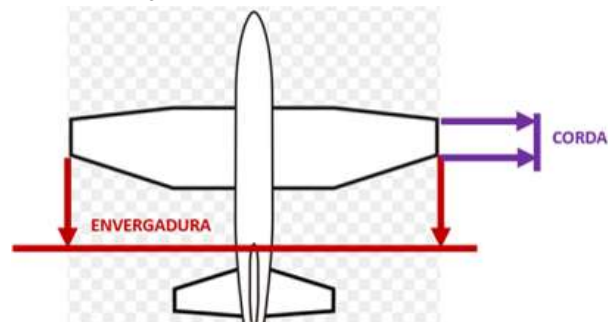


Tabela 1. Valores de asas gerados pelo AG. Fonte: Autor.

ASAS PAIS			
ASA PAI A		ASA PAI B	
Envergadura	1,2658	Envergadura	1,3532
Corda	0,8574	Corda	0,9763

VALORES CONVERTIDOS PARA NÚMEROS BINÁRIOS			
ASA PAI A		ASA PAI B	
Envergadura	11000101110010	Envergadura	11010011011100
Corda	10000101111110	Corda	10011000100011

FILHOS GERADOS POR COMBINAÇÃO DE NÚMEROS BINÁRIOS			
ASA FILHO 1		ASA FILHO 2	
Envergadura	11000011011100	Envergadura	11010101110010
Corda	10001000100011	Corda	10010101111110

FILHOS GERADOS CONVERTIDOS PARA VALORES REAIS			
ASA FILHO 1		ASA FILHO 2	
Envergadura	1,2508	Envergadura	1,3682
Corda	0,8739	Corda	0,9598

CONCLUSÃO

O Algoritmo proposto gera potenciais resultados positivos que serão utilizados para o desenvolvimento de asas da aeronave. Ainda ocorrerão outros testes com diferentes métodos de otimização para aprimorar o processo de obtenção de novas asas.

O desempenho do AG possibilita valores mais exatos para a construção digital das asas que serão utilizadas na competição, além de proporcionar uma grande quantidade de modelos de asa próximos ao ideal para os requisitos desejados.

Quando mais desenvolvido, a intenção é liberar o algoritmo para público e formar novos métodos de otimização para mais facilidade de uso.

REFERÊNCIAS

- Araújo, L. G.; Costa, A. B. Otimização de Projeto Aerodinâmico de Pá de Microgerador Eólico Empregando o Algoritmo Genético. 2017. 85f. Trabalho de Conclusão de Curso (Graduação) – Universidade Tecnológica Federal do Paraná, Curitiba, 2017.
- Bortolete, F. A. Algoritmo de Otimização Aerodinâmica de Asas Voltadas a uma Aeronave de Baixo Número de Reynolds. 2017. 63f. Trabalho de Conclusão de Curso (Graduação) – Escola Superior de Tecnologia, Universidade do Estado do Amazonas, Manaus, 2017.
- Corrêa, R., Obregon, L.C., Morais, A.P., Mariotto, L., Cardoso, G., Guarda, F.K. Coordenação de Relés de Sobrecorrente em Sistemas de Potência Industriais: Uma Abordagem Considerando Algoritmos Genéticos. UFSM, 2016.
- Darwin, C. M. A. The origin of species. Londres, Inglaterra: John Murray, 1859.
- Freitas, J. G. Computação Evolutiva Aplicada ao Problema da Geração de Grade Horária: o Caso do Curso de Análise e Desenvolvimento de Sistemas do IFTM. 2012. 75f. Trabalho de Conclusão de Curso (Mestrado) – Universidade Federal de Uberlândia, Uberlândia, 2012.
- Goldberg, D. E. (1989). Genetic Algorithms in Search, Optimization, and Machine Learning. EUA: Addison-Wesley. 0-201-15767-5, 1989
- Koza, J.R. Genetic Programming. On the Programming of Computers by Means of Natural Selection. [S.l.]: MIT Press, 1992.
- Kurcewicz, Felipe Caumo. Otimização de Perfil Aerodinâmico para Aeronaves de Competição SAE Aerodesign por Algoritmo Evolutivo. 2017. 25f. Trabalho de Conclusão de Curso (Graduação) – Universidade Federal do Rio Grande do Sul, Porto Alegre, 2017.
- Mathias, M. S. Projeto Aerodinâmico de Aerofólios e Asa para um Avião de Classe Commuter. 2015. 92f. Trabalho de Conclusão de Curso (Graduação) – Escola de Engenharia Aeronáutica de São Carlos, Universidade de São Paulo, São Carlos, 2015.
- Matos, A. R. F. Otimização de Asas Laminares para Aeronaves “Pro-green”. 2009. 111f. Trabalho de Conclusão de Curso (Mestrado) – Instituto Superior Técnico, Universidade Técnica de Lisboa, Lisboa, 2009.
- Mitchel, M. An introduction to genetic algorithms. 3. ed. Cambridge, USA: MIT Press, 1997.
- Obitko, M. Algoritmos Genéticos: Introdução aos Algoritmos Genéticos, 1998. Disponível em: <https://www.obitko.com/tutorials/genetic-algorithms/portuguese/index.php>. Acesso em: 25 mar. 2020.
- Secchi, A. Algoritmos Genéticos: Introdução aos Algoritmos Genéticos, 2018. Disponível em: http://www2.peq.coppe.ufrj.br/Pessoal/Professores/Arge/COQ897/Naturais/aulas_piloto/aula3.pdf. Acesso em: 26 mar. 2020.
- Widmaier, K. Algoritmo Genético Aplicado à Otimização de Asas de Material Compósito de Veículos Aéreos Não Tripulados. 2005. 203f. Trabalho de Conclusão de Curso (Mestrado) – Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2005.