

RENATA BERNARDES

**PROJETO DE SISTEMAS DE ENGENHARIA UTILIZANDO
O ALGORITMO DE BUSCA FRACTAL ESTOCÁSTICA**



UNIVERSIDADE FEDERAL DE UBERLÂNDIA
FACULDADE DE ENGENHARIA MECÂNICA

2021

RENATA BERNARDES

PROJETO DE SISTEMAS DE ENGENHARIA UTILIZANDO O
ALGORITMO DE BUSCA FRACTAL ESTOCÁSTICA

Dissertação apresentada
ao Curso de Pós-graduação em
Engenharia Mecânica da
Universidade Federal de
Uberlândia, com parte dos
requisitos para a obtenção do
título de **MESTRE EM
ENGENHARIA MECÂNICA.**

Orientador: Prof. Dr. Fran
Sergio Lobato

Uberlândia – MG

2021

Dados Internacionais de Catalogação na Publicação (CIP)
Sistema de Bibliotecas da UFU, MG, Brasil.

B522p
2021 Bernardes, Renata, 1996-
 Projeto de sistemas de engenharia utilizando o algoritmo de busca
 fractal estocástica [recurso eletrônico] / Renata Bernardes. - 2021.

 Orientador: Fran Sergio Lobato.
 Dissertação (mestrado) - Universidade Federal de Uberlândia.
 Programa de Pós-Graduação em Engenharia Mecânica.
 Modo de acesso: Internet.
 Disponível em: <http://doi.org/10.14393/ufu.di.2021.5568>
 Inclui bibliografia.

 1. Engenharia mecânica. I. Lobato, Fran Sergio, 1976-, (Orient.). II.
 Universidade Federal de Uberlândia. Programa de Pós-Graduação em
 Engenharia Mecânica. III. Título.

CDU:621

Glória Aparecida
Bibliotecária - CRB-6/2047



UNIVERSIDADE FEDERAL DE UBERLÂNDIA

Coordenação do Programa de Pós-Graduação em Engenharia Mecânica
Av. João Naves de Ávila, nº 2121, Bloco 1M, Sala 212 - Bairro Santa Mônica, Uberlândia-MG, CEP 38400-902
Telefone: (34) 3239-4282 - www.posgrad.mecanica.ufu.br - secposmec@mecanica.ufu.br



ATA DE DEFESA - PÓS-GRADUAÇÃO

Programa de Pós-Graduação em:	Engenharia Mecânica				
Defesa de:	Dissertação de Mestrado Acadêmico, nº 592, COPEM				
Data:	30/07/2021	Hora de início:	09:00	Hora de encerramento:	11:40
Matrícula do Discente:	11922EMC011				
Nome do Discente:	Renata Bernardes				
Título do Trabalho:	Projeto de Sistemas de Engenharia utilizando o Algoritmo de Busca Fractal Estocástica				
Área de concentração:	Mecânica dos Sólidos e Vibrações				
Linha de pesquisa:	Projetos de Sistemas Mecânicos				
Projeto de Pesquisa de vinculação:					

Reuniu-se por meio de webconferência a Banca Examinadora, designada pelo Colegiado do Programa de Pós-graduação em Engenharia Mecânica, assim composta: Professores Doutores: Aldemir Aparecido Cavalini Júnior - FEMEC/UFU; Edson Hideki Koroishi - UTFPR; e Fran Sérgio Lobato - FEQUI/UFU, orientador do candidato. Ressalta-se que os Profs. Aldemir Aparecido Cavalini Júnior, Fran Sérgio Lobato e a Discente participaram da defesa por meio de webconferência desde a cidade de Uberlândia/MG, Prof. Edson Hideki Koroishi desde a cidade de São Cornélio Procópio/PR em atendimento a Portaria nº 36, de 19 de março de 2020, da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - CAPES.

Iniciando os trabalhos o presidente da mesa, Dr. Fran Sérgio Lobato, apresentou a Comissão Examinadora e a candidata, agradeceu a presença do público, e concedeu ao Discente a palavra para a exposição do seu trabalho. A duração da apresentação da Discente e o tempo de arguição e resposta foram conforme as normas do Programa.

A seguir o senhor(a) presidente concedeu a palavra, pela ordem sucessivamente, aos(às) examinadores(as), que passaram a arguir o(a) candidato(a). Ultimada a arguição, que se desenvolveu dentro dos termos regimentais, a Banca, em sessão secreta, atribuiu o resultado final, considerando o(a) candidato(a):

Aprovado.

Esta defesa faz parte dos requisitos necessários à obtenção do título de Mestre.

O competente diploma será expedido após cumprimento dos demais requisitos, conforme as normas do Programa, a legislação pertinente e a regulamentação interna da UFU.

Nada mais havendo a tratar foram encerrados os trabalhos. Foi lavrada a presente ata que após lida e achada conforme foi assinada pela Banca Examinadora.



Documento assinado eletronicamente por **Fran Sergio Lobato, Professor(a) do Magistério Superior**, em 30/07/2021, às 11:41, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



Documento assinado eletronicamente por **Edson Hideki Koroishi, Usuário Externo**, em 30/07/2021, às 11:42, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



Documento assinado eletronicamente por **Aldemir Aparecido Cavalini Junior, Professor(a) do Magistério Superior**, em 30/07/2021, às 11:42, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



A autenticidade deste documento pode ser conferida no site https://www.sei.ufu.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador **2924010** e o código CRC **F33CAD26**.

Agradecimentos

Primeiramente à Deus por colocar no meu caminho desafios que eu consigo cumprir e que me motivam, além de pessoas incríveis que fazem da minha jornada mais tranquila.

Aos meus pais por todo o apoio que sempre me proporcionaram, por toda a torcida, toda a luta, todo o incentivo para que eu me torne sempre a melhor versão de mim mesma.

Ao meu companheiro de tantos anos por sempre acreditar em mim e na minha capacidade e me mostrar que, mesmo no desespero, existem caminhos a serem percorridos para se chegar ao sucesso.

A todos os meus amigos e familiares que direta ou indiretamente contribuíram para minha formação com palavras de carinho e conforto.

Ao Prof. Dr. Fran Sérgio Lobato por tamanha maestria em me auxiliar a conduzir a pesquisa e me fornecer tanta segurança nessa caminhada.

E, finalmente, à Universidade Federal de Uberlândia e à Faculdade de Engenharia Mecânica pela oportunidade de participar do Programa de Pós Graduação em Engenharia Mecânica e por toda a infraestrutura oferecida.

Resumo

O projeto de sistemas de engenharia tem um papel importante no desenvolvimento no mundo como conhecemos hoje. Para essa finalidade, a otimização configura-se como um conjunto de ferramentas que auxiliam a tomada de decisões durante a etapa de projeto. Neste contexto, o estudo de Métodos Heurísticos é imprescindível para que novas e versáteis ferramentas possam ser desenvolvidas. Diante do que foi apresentado, este trabalho tem como objetivo estudar e aplicar o algoritmo de Busca Fractal Estocástica (BFE), que é baseado na teoria dos fractais, para o projeto de sistemas de engenharia. Além disso, também propõem-se operadores para aumentar a diversidade na população, prevenir a convergência prematura e reduzir o custo computacional. Para essa finalidade, a associação entre o algoritmo de BFE com Modelos Caóticos de Busca com uma estratégia para reduzir o tamanho da população ao longo das gerações foi proposta. Para avaliar a qualidade dos resultados obtidos foram realizadas a análise de sensibilidade dos parâmetros do algoritmo, a sua aplicação em estudos de caso mono e multi-objetivos, bem como a atualização de seus parâmetros. Os resultados obtidos demonstraram o potencial desta recente abordagem evolutiva, bem como dos operadores propostos no que tange o projeto de sistemas de engenharia.

Palavras-chave: Projeto de sistemas de engenharia, Busca Fractal Estocástica, Otimização, Parâmetros Auto-Adaptativos, Modelos Caóticos de Busca.

Abstract

Engineering systems design has an important role in development of world as we know it today. In this context, the optimization is configured as a set of tools that help the decision making during the design stage. In this scenario, the study of Heuristic Methods is essential for new and versatile tools be developed. In this contribution the aims is to study and apply the Stochastic Fractal Search (SFS) algorithm, which is based on fractal theory, to engineering systems design. Furthermore, operators to increase population diversity, to prevent premature convergence and to reduce computational cost are proposed. For this purpose, the association between the BFE algorithm with Search Chaotic Models and a strategy to reduce population size along generations was proposed. To evaluate the quality of obtained results, the sensibility analysis of the algorithm parameters, its application in single and multi-objective case studies, as well as the updating of its parameters were performed. The results obtained demonstrate the potential of this recent evolutionary approach, as well as the proposed operators to engineering systems design.

Keywords: Engineering systems design, Stochastic Fractal Search, Optimization, Self-Adpatative Parameters, Search Chaotic Pattern.

Lista de Figuras

1	Categorias dos Algoritmos de Otimização	4
2	Opções de Compra de um Carro: Custo Versus Conforto	12
3	Representação Esquemática da Difusão de uma Partícula	15
4	Algoritmo de Busca Fractal Estocástica	19
5	Gráfico Tridimensional do Exemplo Analisado	20
6	Atualização da População para o Exemplo Analisado	24
7	Influência do Tamanho da População no Algoritmo de BFE	29
8	Influência do Número de Gerações no Algoritmo de BFE	31
9	Influência do Número de Difusões no Algoritmo de BFE	33
10	Influência da Escolha do Passo no Algoritmo de BFE	35
11	Projeto de uma Mola Tensão/Compressão	40
12	Projeto de uma Viga Soldada	43
13	Projeto de um Vaso de Pressão	46
14	Projeto de uma Coluna Tubular Sujeita a uma Carga Compressiva	48
15	Projeto de uma Barra Cantilever	51
16	Definição dos Eixos, Velocidades, Forças, Momentos e Ângulos de Euler da Aeronave	52
17	Sistema de Controle Longitudinal	53
18	Sistema de Controle Látero-direcional	53
19	Ângulo de Arfagem e Deflexão do Profundor Considerando Diferentes Abordagens	55
20	Altura e Deflexão do Profundor da Aeronave Considerando Diferentes Abordagens	56
21	Ângulo de Rolagem e Deflexão do Aileron Considerando Diferentes Abordagens	58
22	Ângulo de Guinada Considerando Diferentes Abordagens	59
23	Ângulo de Deflexão do Aileron Considerando Diferentes Abordagens	60
24	Ângulo de Deflexão do Leme Considerando Diferentes Abordagens	61
25	Gráfico da Função de Himmelblau	62
26	Curvas de Nível para a Função de Himmelblau	62
27	Curva de Pareto para a Função de Schaffer Considerando o BFE e o MSP	64
28	Projeto de uma Viga I	65
29	Curva de Pareto para o Problema do Projeto de uma Viga I Considerando Diferentes Abordagens	66
30	Projeto de uma Viga Soldada	67
31	Curva de Pareto para o Problema do Projeto de uma Viga Soldada Considerando Diferentes Abordagens	69

Lista de Tabelas

1	Inicialização do Algoritmo de BFE	20
2	População Inicial do Algoritmo de BFE Ordenada	21
3	Representação do Processo de Difusão no Algoritmo de BFE	21
4	Processo de Ranqueamento no Algoritmo de BFE	21
5	Primeira Etapa para a Atualização de Candidatos no Algoritmo de BFE	22
6	Primeira Etapa para a Atualização de Candidatos no Algoritmo de BFE Ordenado	22
7	Segunda Etapa de atualização no Algoritmo de BFE	23
8	Resultados Obtidos com a Execução do Algoritmo de BFE	23
9	Parâmetros do Algoritmo de BEF considerados na Análise de Sensibilidade	28
10	Influência do Tamanho da População no Algoritmo de BFE	30
11	Influência do Número de Gerações no Algoritmo de BFE	32
12	Influência do Número de Difusões no Algoritmo de BFE	34
13	Influência do Passo no Algoritmo de BFE	36
14	Resultados Obtidos pelo Algoritmo de BFE para a Função de Beale	38
15	Resultados Obtidos pelo Algoritmo de BFE para a Função de Rastrigin	39
16	Resultados Obtidos pelo Algoritmo de BFE para a Função de Michalewicz	39
17	Resultados Obtidos pelo Algoritmo de BFE para a Função de Levy	41
18	Resultados Obtidos pelo Algoritmo de BFE para o Problema do Projeto de uma Mola	42
19	Resultados Obtidos para o Problema do Projeto de uma Mola Considerando Diferentes Abordagens	42
20	Resultados Obtidos pelo Algoritmo de BFE para o Problema do Projeto de uma Viga Soldada	45
21	Resultados Obtidos para o Problema do Projeto de uma Viga Considerando Diferentes Abordagens	45
22	Resultados Obtidos pelo Algoritmo de BFE para o Problema do Projeto de um Vaso de Pressão	47
23	Resultados Obtidos para o Problema do Projeto de um Vaso de Pressão Considerando Diferentes Abordagens	47
24	Resultados Obtidos pelo Algoritmo de BFE para o Problema do Projeto de uma Coluna Tubular Sujeita a uma Carga Compressiva	49
25	Resultados Obtidos para o Problema do Projeto de uma Coluna Tubular Sujeita a uma Carga Compressiva Considerando Diferentes Abordagens	50
26	Resultados Obtidos pelo Algoritmo de BFE para o Problema do Projeto de uma Barra Cantilever	51
27	Resultados Obtidos para o Problema do Projeto de uma Barra Cantilever Considerando Diferentes Abordagens	52
28	Resultados Obtidos para Problema do Controle Longitudinal com Malha Interna Considerando Diferentes Abordagens	54
29	Resultados Obtidos para o Problema do Controle Longitudinal com Malha Externa Considerando Diferentes Abordagens	55
30	Resultados Obtidos para o Problema do Controle Látero-Direcional com Malha Interna Considerando Diferentes Abordagens	57
31	Resultados Obtidos para o Problema do Controle Látero-Direcional com Malha Externa Considerando Diferentes Abordagens	59
32	Número de Soluções Encontradas pelo Algoritmo de BFE em Função do Parâmetro Δ_{par}	63

33	Resultados Obtidos pelo Algoritmo de BFE para o Problema do Projeto de Viga I .	67
34	Resultados Obtidos pelo Algoritmo de BFE para o Problema do Projeto de Viga Soldada	70
35	Resultados Obtidos pelos Algoritmos BFEA, BFEML e BFEMS para os Estudos de Caso	72
36	Resultados Obtidos pelo Algoritmo BFEVP para os Estudos de Caso	74
37	Resultados Obtidos pelo Algoritmo BFEM para os Estudos de Caso	75
38	Número de Avaliações da Função Objetivo Considerando os Algoritmos BFE, BFEA, BFEML, BFEMS, BFEVP e BFEM	76

Lista de Símbolos

CAPÍTULO 2

Letras Romanas

c_i - Constante de normalização da função objetivo.
 d - Solução codificada.
 f - Função objetivo.
 $f_k^{pior}(x)$ - Pior valor da k -ésima função objetivo.
 $f_k^*(x)$ - Melhor valor da k -ésima função objetivo.
 f° - Função objetivo otimizada individualmente.
 f^* - Função objetivo normalizada.
 g - Vetor de restrições de desigualdade.
 h - Vetor de restrições de igualdade.
 i, j, k - Contador.
 J - Número de restrições de desigualdade.
 K - Número de restrições de igualdade.
 m, M - Número de objetivos.
 n - Número de variáveis de projeto.
 P - Função de penalidade.
 p - Número real positivo.
 r_p, r'_p - Fator de penalidade.
 s - Espaço viável de busca.
 T - Traçador.
 w - Vetor de parâmetros de ponderação.
 x - Vetor de variáveis de projeto.
 x_i^{inf} - Limite inferior para a variável x_i .
 x_i^{sup} - Limite superior para a variável x_i .
 x° - Estimativa inicial de projeto.
 Z - Espaço de objetivos.

Letras Gregas

ζ - Espaço viável de busca.
 λ_j - Multiplicador de Lagrange.
 $\Phi(x, r_p)$ - Função pseudo-objetivo.
 ψ_j - Relação de atualização para o tratamento de restrições.

CAPÍTULO 3

Letras Romanas

BP - Posição do melhor ponto da população.
 D - Limite inferior para o espaço de busca.
 E - Energia potencial elétrica máxima para o problema.
 E_{antiga}^t - Energia potencial elétrica da partícula t antes da redistribuição de energia.
 E_i - Energia potencial elétrica de cada partícula.

E_{nova}^t - Energia potencial elétrica da partícula t após a redistribuição de energia.
 GW - Caminhada Gaussiana.
 i, j, t - Contador.
 i_{dif} - Índice de difusão.
 L - Distribuição de Lévy.
 LB - Limite inferior de busca para a variável de projeto.
 Lim_{inf} - Limite inferior da barreira em torno de $x_{ótimo}$.
 Lim_{sup} - Limite superior da barreira em torno de $x_{ótimo}$.
 $\min(\bar{E})$ - Energia mínima que cada partícula pode ter.
 N - Dimensão do problema.
 N_{dif} - Número de difusões.
 N_{ger} - Número de gerações.
 N_p - Número de indivíduos da população.
 N_p^{inf} - Limite inferior para o tamanho da população.
 N_p^{sup} - Limite superior para o tamanho da população.
 N_{passo} - Passo da caminhada Gaussiana.
 P - População.
 P' - População após a primeira atualização de candidatos.
 P'' - População após a segunda atualização de candidatos.
 P_a - Resultado do ranqueamento dos pontos.
 P_t, P_r, P'_t, P'_r - Pontos aleatórios da população.
 q_{dif} - Número de partículas adquirido no processo de difusão.
 r_p - Fator de Penalidade.
 U - Limite superior para o espaço de busca.
 UB - Limite superior de busca para a variável de projeto.
 x - Vetor de variáveis de projeto.
 $x_{ótimo}$ - Solução ótima.

Letras Gregas

α - Fator de escala de distribuição.
 α_i - Parâmetro de convergência.
 $\gamma, \gamma', \epsilon, \epsilon', \dot{\epsilon}$ - Números aleatórios entre 0 e 1.
 Δ_{par} - Parâmetro para a geração de barreira (número real).
 ζ - Número de partículas.
 λ - Índice de distribuição.
 μ - Número real positivo.
 μ_{BP} - Módulo da posição do melhor ponto da população.
 μ_P - Módulo do ponto P_i .
 ϕ - Quantidade de energia das partículas descartadas.
 ϑ - Número de partículas criadas no espaço de busca.

CAPÍTULO 4

Letras Romanas

G - Função objetivo.
 i - Contador.
 N_{dif} - Número de difusões.
 N_{ger} - Número de gerações.
 N_p - Número de indivíduos da população.
 N_{passo} - Passo da caminhada Gaussiana.
 n - Dimensão do problema.
 x - Vetor de variáveis de projeto.
 $x_{ótimo}$ - Solução ótima.

Letras Gregas

Não possui.

CAPÍTULO 5

Letras Romanas

b - Largura da viga soldada.
 D - Diâmetro médio da espiral da mola.
 d - Diâmetro do fio da mola, Diâmetro médio da coluna tubular.
 E - Fator de Young.
 $e(h_k)$ - Erro referente à altitude da aeronave.
 $e(\theta_k)$ - Erro referente ao ângulo de arfagem da aeronave.
 $e(\Phi_k)$ - Erro referente ao ângulo de rolagem da aeronave.
 $e(\Psi_k)$ - Erro referente ao ângulo de guinada da aeronave.
 f - Função objetivo.
 f° - Função objetivo otimizada individualmente (solução ideal).
 G - Módulo de cisalhamento.
 g - Função de restrição.
 h - Altura da solda da viga soldada, Altitude da aeronave.
 h_k - Perturbação da altitude da aeronave.
 h_{ref} - Valor de referência da altitude da aeronave.
 I - Momento de inércia.
 i, j, k - Contador.
 L - Comprimento em balanço da viga soldada, Comprimento da secção cilíndrica do vaso de pressão, Momento de rolagem da aeronave.
 l - Comprimento soldado da viga soldada, Comprimento da coluna tubular.
 M - Momento de arfagem da aeronave.
 M_Y, M_Z - Momentos máximos nas direções y e z .
 N - Momento de guinada da aeronave.
 n - Dimensão do problema.
 n_e - Número de Elementos dos Vetores
 N_{esp} - Número de espirais da mola.

N_{eval} - Número de avaliações da função objetivo.
 $N_{eval,p}$ - Número de avaliações da função objetivo referentes à p -ésima metodologia proposta.
 $N_{eval,c}$ - Número de avaliações da função objetivo referentes ao algoritmo canônico.
 P - Carga aplicada, Velocidade angular de rolagem da aeronave.
 P_c - Carga de flambagem na barra.
 P_{kr} - Carga de flambagem de Euler.
 Q - Velocidade angular de arfagem da aeronave, Carga aplicada.
 R - Raio interno do vaso de pressão, Velocidade angular de guinada da aeronave.
 r_p - Fator de penalidade.
 T_s - Espessura da casca do vaso de pressão.
 T_h - Espessura da tampa do vaso de pressão.
 t - Altura da viga soldada, Espessura da coluna tubular.
 U - Velocidade longitudinal da aeronave.
 u_{max} - Limite máximo para deslocamento da extremidade da viga soldada.
 W_Y, W_Z - Módulos resistentes da seção nas direções y e z .
 w - Fator de ponderação.
 x - Vetor de variáveis de projeto.
 x_k - Valor de x no instante k .
 $x_{ótimo}$ - Solução ótima.
 x_{ref} - Valor de referência para x .
 x_{SBRF} - Vetor de variáveis de projeto do controle de uma aeronave de asa fixa.

Letras Gregas

Γ - Taxa de redução (redução percentual que relaciona o número de avaliações entre dois algoritmos).
 Δa_k - Deflexão do aileron da aeronave.
 ΔP_k - Deflexão do profundor da aeronave.
 Δ_{par} - Parâmetro para criação de barreira (número real).
 Δr_k - Deflexão do leme da Aeronave.
 δ - Deflexão na barra.
 δ_{max} - Deflexão máxima na barra.
 Θ, θ - Ângulo de arfagem da aeronave.
 θ_k - Perturbação do ângulo de arfagem da aeronave.
 θ_{ref} - Valor de referência do ângulo de arfagem da aeronave.
 μ - Número real positivo definido de acordo com o modelo de busca caótico.
 ρ - Densidade do material.
 σ - Tensão de flexão na viga.
 σ_{max} - Tensão máxima de cisalhamento da viga soldada, Tensão de projeto da viga I.
 σ_y - Tensão de escoamento.
 τ - Tensão de cisalhamento.
 τ_{max} - Tensão máxima de cisalhamento.
 Φ - Ângulo de rolagem da aeronave.
 Φ_k - Perturbação do ângulo de rolagem da aeronave.
 Φ_{ref} - Valor de referência do ângulo de rolagem da aeronave.
 Ψ - Ângulo de guinada da aeronave.
 Ψ_k - Perturbação do ângulo de guinada da aeronave.
 Ψ_{ref} - Valor de referência do ângulo de guinada da aeronave.

Abreviaturas

AG - Algoritmo Genético.
APF - Algoritmo de Polinização de Flores.
BF - Busca Fractal.
BFE - Busca Fractal Estocástica.
BFEA - Busca Fractal Estocástica com variação de parâmetros de forma Aleatória.
BFEM - Busca Fractal Estocástica Modificada.
BFEML - Busca Fractal Estocástica com variação de parâmetros utilizando o modelo de busca caótico Mapa Logístico.
BFEMS - Busca Fractal Estocástica com variação de parâmetros utilizando o modelo de busca caótico Mapa Seno.
BFEVP - Busca Fractal Estocástica com Variação da População.
CV - Colônia de Vagalumes.
ED - Evolução Diferencial.
EP - Enxame de Partículas.
MCPE - Método do Critério Ponderado Exponencial.
MOC - Método da Otimização de Compromisso.
MSP - Método da Soma Ponderada.
RMS - *Root Mean Square*.
RS - Recozimento Simulado.
SFS - *Stochastic Fractal Search*.
SQP - *Sequential Quadratic Programming*.

Conteúdo

1	Introdução	1
2	Revisão Bibliográfica	4
2.1	O que é a Otimização	4
2.2	Classificação dos Problemas de Otimização	4
2.3	Caracterização do Problema de Otimização	5
2.3.1	Variáveis de Projeto	5
2.3.2	Função Objetivo	5
2.3.3	Restrições de Projeto	6
2.3.4	Formulação Matemática do Problema de Otimização	6
2.4	Algoritmos Heurísticos de Otimização	6
2.4.1	Algoritmo Genético	6
2.4.2	Colônia de Vagalumes	7
2.4.3	Enxame de Partículas	7
2.4.4	Evolução Diferencial	7
2.4.5	Polinização de Flores	7
2.4.6	Recozimento Simulado	8
2.5	Tratamento de Restrições	8
2.5.1	Métodos Baseados em Funções de Penalidade	8
2.5.2	Método da Pena de Morte	10
2.5.3	Métodos Baseados na Preservação das Soluções Viáveis	10
2.6	Classificação dos Métodos para a Resolução de Problemas Mono e Multi-objetivos	10
2.6.1	Transformação do Problema de Otimização Multi em um Mono-objetivo	10
2.6.2	Ótimo de Pareto	12
2.7	Atualização dos Parâmetros em Algoritmos de Otimização	13
3	Busca Fractal Estocástica	14
3.1	Concepção Conceitual	14
3.1.1	Fractais	14
3.1.2	Busca Fractal	14
3.2	O Algoritmo de BFE	17
3.2.1	Aplicação Passo a Passo do Algoritmo de BFE	18
3.3	Operadores Propostos para Aumentar a Diversidade e Reduzir o Custo Computacional no Algoritmo de BEF	24
3.3.1	Atualização Dinâmica do Número de Difusões e do Número de Passos da Caminhada Gaussiana	25
3.3.2	Atualização Dinâmica do Tamanho da População	25
3.4	Estratégia para Determinar Todas as Soluções em Problemas de Otimização	26
4	Estudo da Sensibilidade do Algoritmo BFE	27
4.1	Influência do Tamanho da População	29
4.2	Influência do Número de Gerações	31
4.3	Influência do Número de Difusões	33
4.4	Influência do Passo	34
4.5	Consolidado dos Resultados	36

5	Resultados e Discussões	38
5.1	Problemas Matemáticos Mono-objetivos	38
5.1.1	Função de Beale	38
5.1.2	Função de Rastrigin	38
5.1.3	Função de Michalewicz	39
5.1.4	Função de Levy	40
5.2	Problemas de Projeto Mono-objetivos	40
5.2.1	Projeto de uma Mola de Tensão/Compressão	40
5.2.2	Projeto de uma Viga Soldada	43
5.2.3	Projeto de um Vaso de Pressão	45
5.2.4	Projeto de uma Coluna Tubular Sujeita a uma Carga Compressiva	48
5.2.5	Projeto de uma Barra Cantilever	50
5.2.6	Controle de uma Aeronave de Asa Fixa	51
5.3	Aplicação da Metodologia Proposta para Determinar Todas as Soluções de um Problema de Otimização	61
5.4	Problemas Matemáticos e de Engenharia Multi-objetivos	63
5.4.1	Função SCH1 de Schaffer	64
5.4.2	Projeto de uma Viga I	65
5.4.3	Projeto de uma Viga Soldada	67
5.5	Atualização do Número de Difusões e do Passo da Caminhada Gaussiana	70
5.6	Atualização do Tamanho da População	72
5.7	Atualização do Número de Difusões, do Passo da Caminhada Gaussiana e do Tamanho da População	73
5.8	Consolidado dos Resultados	75
5.8.1	Algoritmo BFE	75
5.8.2	Algoritmo BFE com Atualização dos Parâmetros	76
6	Conclusões e Trabalhos Futuros	77
6.1	Conclusões	77
6.2	Trabalhos Futuros	78

1 Introdução

Durante décadas, o uso de métodos baseados em informações sobre a derivada da função objetivo e das restrições empregados para a resolução do problema de otimização têm sido exaustivamente estudados e avaliados em estudos de caso com diferentes níveis de complexidade. Com o surgimento dos métodos heurísticos, uma série de algoritmos com diferentes fundamentações teóricas têm sido propostos e desenvolvidos também para essa finalidade. Devido às características bem particulares que podem ser encontradas no projeto de sistemas de engenharia, tais como não linearidades, multimodalidade, presença de restrições e de variáveis mistas (contínuas, binárias, discretas e inteiras), a aplicação deste tipo de abordagem nestes tipos de problemas tem-se mostrado bastante promissora. É importante enfatizar que a otimização tem-se consagrado como metodologia base para o projeto de sistemas de engenharia, sendo que nos últimos anos os pesquisadores têm buscado formas de sistematizar tais processos via uso de recursos computacionais cada vez mais sofisticados. Na prática, isso implica no uso cada vez mais frequente dos reconhecidos métodos, visto que a redução do tempo de processamento de problemas realísticos favorece a aplicação deste tipo de abordagem.

A engenharia possui uma grande importância no desenvolvimento do mundo hoje. Um país só é considerado desenvolvido se possui uma indústria forte, competitiva e avançada tecnologicamente. Além disso, a engenharia é considerada indispensável para a infraestrutura, melhoria na qualidade de serviços prestados à sociedade e resolução de problemas em geral. Compreende-se aplicações que vão desde projetos de fluxo de tráfego à maquinaria pesada como os meios de transporte (automóveis, trens e metrô, navios), bem como construções de edifícios, túneis, barragens, além também do projeto de produção de produtos que ainda não estão disponíveis no mercado. Neste contexto, os projetos de sistemas de engenharia são necessários em qualquer ramo da atividade humana, visto que objetiva-se extrair o maior potencial dos processos ao mesmo tempo que minimiza-se os recursos disponíveis.

Para essa finalidade, deve-se formular e resolver o denominado problema de otimização (MICHALEWICZ *et al.*, 1999), podendo ser empregados os algoritmos de otimização. De forma geral, estes podem ser classificados em duas grandes classes, a saber, os métodos clássicos (também chamados de determinísticos) e os métodos heurísticos (também denominados como não determinísticos ou não clássicos) (SARAMAGO, 1999; DEB, 2001). Os primeiros são baseados no uso de informações sobre o gradiente da função objetivo e das restrições para atualizar uma estimativa inicial, definida pelo usuário. Assim, estratégias baseadas em derivadas de primeira e/ou segunda ordem podem ser empregadas.

Em linhas gerais, a principal vantagem no uso da abordagem clássica é a rápida taxa de convergência nas vizinhanças da solução ótima (MICHALEWICZ *et al.*, 1999). Por outro lado, estes apresentam algumas limitações, dentre as quais pode-se citar (SARAMAGO, 1999; MEDEIROS & KRIPKA, 2012): *i*) são métodos de busca local; *ii*) apresentam dificuldades em trabalhar com variáveis mistas e *iii*) não são capazes de lidar com funções não diferenciáveis. Já os métodos heurísticos são baseados em algoritmos que, em geral, buscam imitar o comportamento de espécies, bem como os fenômenos que podem ser observados na natureza. Estes apresentam como principal característica o fato de não fazerem uso de informações sobre o gradiente da função objetivo e das restrições, diferentemente dos clássicos (LOBATO, 2008). São reconhecidamente métodos de busca global, o que representa a sua principal vantagem. Todavia, como estes métodos, em sua maioria, trabalham com uma população de candidatos, diferentemente da abordagem clássica, requerem um grande número de avaliações da função objetivo e das restrições (SARAMAGO, 1999; DEB, 2001).

No contexto dos métodos clássicos, Gould & Robinson (2010) demonstraram a convergência do método SQP (*Sequential Quadratic Programming*) no contexto do projeto de sistemas de engenharia. Nesta mesma linha, Shen *et al.* (2010) também demonstraram a convergência do algoritmo SQP

aplicado em problemas de otimização com restrições. Marinho & Santos Neto (2010) aplicaram um algoritmo Simplex para problemas de rede com diferentes níveis de complexidade. Changyu *et al.* (2013) propuseram uma nova versão do algoritmo SQP para a resolução de problemas de engenharia com restrições.

Já no contexto heurístico, Moreira & Krohling (2015) desenvolveram uma abordagem utilizando como base o algoritmo de Polinização de Flores associado com a idéia de entropia populacional para controle dos modos de busca local e global. Paiva (2018) em seu trabalho propõe uma adaptação ao algoritmo de busca por Enxame de Partículas utilizando uma adaptação de serendipidade. Lautert (2019) faz uso do algoritmo de Otimização por Baleias para o projeto de amortecedores em estruturas submetidas a excitações sísmicas. Bezerra *et al.* (2018) propuseram uma melhoria no algoritmo de Busca Cuco para aumentar a velocidade de convergência do método original com aplicações em funções puramente matemáticas. Barbosa (2017) aborda oito diferentes algoritmos (algoritmos genéticos, otimização por colônia de formigas, otimização por enxame de partículas, colônia de abelhas artificiais, algoritmo do vaga-lume, algoritmo de busca do pássaro cuco, algoritmo do morcego e algoritmo de busca auto-adaptativa do pássaro cuco) para a resolução do problema da previsão da energia eólica a partir da velocidade do vento com dados reais coletados de dois parques eólicos e clusterização de padrões. Santos (2019) apresenta em seu trabalho o problema de identificação de acidentes em usinas nucleares considerando para essa finalidade o algoritmo de Busca Cuco.

Há que se destacar na literatura as abordagens híbridas, que consistem na junção de métodos heurísticos com métodos clássicos para a resolução de problemas de otimização. Zhou *et al.* (2017) apresentaram um sistema estocástico-determinístico para o projeto de um solvente. Silva (2017a) desenvolveu um algoritmo híbrido para a solução de um problema de direção automotiva. Silva (2017b) na mesma linha, apresentou um algoritmo híbrido para a solução de um problema envolvendo o equilíbrio químico em um processo de combustão. Já Coelho & Mariani (2006b) propuseram um algoritmo híbrido baseado em Evolução Diferencial e SQP para a resolução do problema de despacho econômico de energia elétrica.

O sucesso apresentado pelos métodos heurísticos tem impulsionado o desenvolvimento de novas abordagens neste mesmo cenário. Dentre um dos representantes mais recentes destaca-se o algoritmo de Busca Fractal Estocástica - BFE (*Stochastic Fractal Search*), proposto por Salimi (2015). Em linhas gerais, o BFE é uma estratégia de otimização baseada na teoria dos fractais para definir uma relação para a atualização do vetor de variáveis de projeto. Apesar dos bons resultados apresentados pelo autor desta abordagem, inúmeras são as questões envolvendo o desempenho deste algoritmo no que tange a sua aplicabilidade em sistemas mais complexos.

Diante do que foi apresentado, a presente contribuição tem por objetivo aplicar o algoritmo de BFE no projeto de sistemas de engenharia mono e multi-objetivos com diferentes níveis de complexidade. Além disso, também deseja-se estudar a influência dos parâmetros do algoritmo, bem como propor a atualização dos parâmetros deste algoritmo durante a sua execução de modo a verificar a influência desta variação nos resultados obtidos. Para essa finalidade será utilizado como linguagem de programação o *software* comercial MatLab.

Cabe enfatizar que esta proposta de dissertação de mestrado configura a continuidade de trabalhos orientados no Programa de Pós-Graduação da Faculdade de Engenharia Mecânica da Universidade Federal de Uberlândia. Neste contexto, Simões (2002) aplicou o Algoritmo Genético e o Recozimento Simulado para a identificação de falhas em eixos de máquinas rotativas. Oliveira (2005) empregou Algoritmos Genéticos para a otimização da trajetória de robôs com estrutura paralela. Para fins de comparação entre diferentes estratégias de otimização, Oliveira (2006) empregou Algoritmos Genéticos, Evolução Diferencial e Recozimento Simulado para o projeto de sistemas de engenharia. Lobato (2008) propôs a extensão do algoritmo de Evolução Diferencial para problemas multi-objetivos para a resolução de problemas matemáticos e de engenharia. Queiroz (2008) utilizou

os Algoritmos Genéticos e Evolução Diferencial para o projeto de colunas semi-enterradas. Purcina (2012) resolveu sistemas lineares formulados como problemas de otimização considerando para essa finalidade os Algoritmos Genéticos e Evolução Diferencial. Oliveira (2012) projetou robôs manipuladores 3R considerando a topologia do espaço de trabalho via aplicação dos Algoritmos Genéticos, Evolução Diferencial e Programação Quadrática Sequencial. Cavalini Júnior (2013) propôs técnicas para o monitoramento da integridade estrutural voltadas para a detecção e identificação de trincas transversais incipientes em eixos de máquinas rotativas. Para essa finalidade, o autor aplicou a Evolução Diferencial para caracterizar as assinaturas de trincas nas respostas espectrais do rotor. Brandão (2014) projetou um robô manipulador com três juntas rotacionais (3R) via implementação de uma versão melhorada do algoritmo de Evolução Diferencial. Carvalho (2020) utilizou o algoritmo de Evolução Diferencial para a determinação dos parâmetros de um controlador considerado em rotores com mancais magnéticos.

Esse trabalho será estruturado como segue. No Capítulo 2 é apresentada uma breve revisão bibliográfica sobre conceitos relacionados com otimização. A descrição do algoritmo de BFE, bem como os operadores propostos para aumentar a diversidade e reduzir o custo computacional neste algoritmo são apresentados no Capítulo 3. Já no Capítulo 4 é realizado um estudo acerca da influência dos parâmetros do algoritmo. Aplicações considerando problemas matemáticos e de engenharia para a versão canônica e para o algoritmo considerando os operadores propostos são apresentados no Capítulo 5. Finalmente, o último capítulo apresenta as conclusões deste trabalho.

2 Revisão Bibliográfica

2.1 O que é a Otimização

Otimização é o processo de se determinar, entre várias opções, aquela que é a melhor possível dentro de certos critérios de escolha e limitações, com os recursos disponíveis (BRASIL & SILVA, 2019). Analisando de forma matemática, a otimização tem como objetivo encontrar valores extremos de uma função (máximos e mínimos da chamada função objetivo), computados em função do vetor de variáveis de projeto, e sujeita às restrições. A seguir são apresentados conceitos importantes para o bom entendimento do problema de otimização.

2.2 Classificação dos Problemas de Otimização

De acordo com Haupt & Haupt (1998), os algoritmos de otimização podem ser classificados de acordo com as categorias apresentadas na Figura 1.

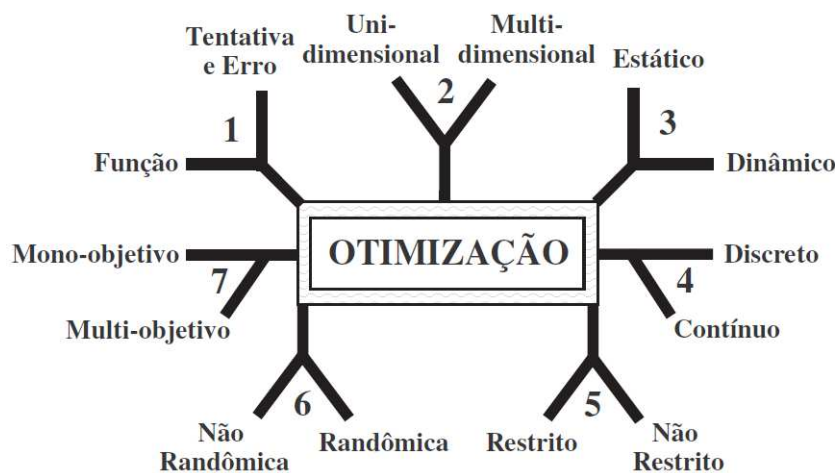


Figura 1: Categorias dos Algoritmos de Otimização. Reproduzido de Haupt & Haupt (1998).

- **Tentativa e Erro x Função:** A Otimização por Função ocorre quando o processo é conhecido, e conseqüentemente, sua formulação matemática. A Tentativa e Erro é referente a uma otimização onde se tem a definição de um ajuste de parâmetros de um processo por experiência, quando não se conhece a formulação do problema (LOBATO, 2008).
- **Uni-dimensional x Multi-dimensional:** Problemas onde existe apenas uma variável de projeto são denominados de uni-dimensionais. Por outro lado, os que apresentam mais de uma variável de projeto são chamados de multi-dimensionais (LOBATO, 2008).
- **Estático x Dinâmico:** A Otimização Estática é aquela que é independente do tempo, enquanto na Otimização Dinâmica a saída muda conforme o tempo (LOBATO, 2008). No geral, o nível de complexidade de problemas dinâmicos é maior quando comparado aos estáticos.

- **Discreto x Contínuo:** O processo de otimização pode ser discreto ou contínuo. O primeiro diz respeito a um número finito de possíveis valores para as variáveis, enquanto que no segundo infinitos valores para as variáveis de projeto são considerados (LOBATO, 2008).
- **Restrito x Não-restrito:** Os estudos de caso encontrados no mundo real são, naturalmente, restritos. Estas se devem às limitações de natureza física, ambiental, operação, entre outras. Por outro lado, os irrestritos são aqueles que não são limitados, isto é; não apresentam relações que limitam o espaço onde a solução ótima será procurada (LOBATO, 2008).
- **Randômica x Não Randômica:** Os métodos clássicos (não randômicos), também chamados de determinísticos, são baseados no uso do gradiente da função objetivo e das restrições para a busca do valor ótimo da função objetivo (SALES *et al.*, 2018). Nesse caso, esta função deve ser contínua e diferenciável no espaço de busca. Estes métodos apresentam teoremas que garantem convergência para uma solução ótima, porém, sem a garantia de solução global. Sendo assim, este tipo de método não é indicado para otimizar funções que possuem vários ótimos locais. Os métodos de otimização heurística (randômicos) são baseados em processos ou fenômenos observados na natureza ou buscam imitar comportamentos observados em espécies. De forma geral, estes trabalham com uma população de candidatos em potencial, o que implica em um número bem maior de avaliações da função objetivo em relação aos métodos baseados em derivadas. Como não faz o uso de derivadas primeiras e segundas da função objetivo e das restrições para a busca do valor ótimo, estes, naturalmente, abrangem um número maior de problemas disponíveis na vida em sociedade (funções não diferenciáveis e não contínuas). Também devido a este fator, tais métodos são reconhecidamente algoritmos de busca global.
- **Mono-objetivo x Multi-objetivo:** Os problemas mono-objetivos são caracterizados pela presença de apenas uma função, cuja solução ótima pode ser facilmente caracterizada. Já um problema multi-objetivo apresenta um vetor de objetivos, o que implica que a solução do mesmo é uma superfície formada por pontos não-dominados (DEB, 2001). Neste caso, para o problema multi-objetivo não existe uma definição como nos problemas mono-objetivos, mas sim um conceito que é utilizado para caracterizar este problema, denominado de ótimo de Pareto (DEB, 2001; LOBATO, 2008).

2.3 Caracterização do Problema de Otimização

Conforme mencionado anteriormente, o problema de otimização (mono ou multi-objetivo) é caracterizado pela função (ou vetor de funções) objetivo, pelo vetor de variáveis de projeto e pelas restrições. Cada uma destas características são detalhadas a seguir.

2.3.1 Variáveis de Projeto

As variáveis de projeto representam as incógnitas presentes no modelo que define o problema de otimização (BRASIL & SILVA, 2019). Estas possuem independência entre si, podem ter natureza mista e influenciam no valor do objetivo no problema considerado.

2.3.2 Função Objetivo

A função objetivo (ou o vetor de funções objetivo) é a métrica considerada para analisar o problema de otimização. Esta é definida em função do vetor de variáveis de projeto, isto é; a partir dos valores assumidos por cada variável de projeto, pode-se mensurar o valor do vetor de função(ões) objetivo de forma que a melhor solução possa ser encontrada (BRASIL & SILVA, 2019).

2.3.3 Restrições de Projeto

As restrições podem ser entendidas como barreiras que limitam o campo de atuação do algoritmo de otimização considerado. Estas relacionam as variáveis de projeto, podendo ser definidas como restrições dinâmicas (impostas ao longo de todo o intervalo de tempo no qual o sistema é analisado) e estáticas (relacionadas a limites geométricos da estrutura e independem do tempo) (LOBATO, 2008). Estas restrições podem ser classificadas em:

- **Restrições de desigualdade:** delimitam uma região do espaço de projeto onde seu valor deve ser maior ou igual/menor ou igual a um valor pré-estabelecido;
- **Restrições de igualdade:** delimitam uma região do espaço de projeto onde as variáveis de projeto define um valor pré-determinado para a restrição;
- **Restrições laterais:** delimita os valores de máximos e mínimos que podem ser adotados pelas variáveis de projeto.

2.3.4 Formulação Matemática do Problema de Otimização

Matematicamente, o problema de otimização multi-objetivo pode ser definido como (DEB, 2001):

$$\min f_m(x), \quad m = 1, 2, \dots, M \quad (2.1)$$

$$\begin{cases} g_j(x) \leq 0, & j = 1, 2, \dots, J \\ h_k(x) = 0, & k = 1, 2, \dots, K \\ x_i^{inf} \leq x_i \leq x_i^{sup}, & i = 1, 2, \dots, n \end{cases} \quad (2.2)$$

em que f é o vetor com M funções objetivo, x é o vetor de n variáveis de projeto em que x_i^{sup} e x_i^{inf} representam o maior e o menor valores que a i -ésima variável de projeto pode alcançar. g_j e h_k representam os vetores de restrições de desigualdade (com J posições no vetor) e igualdade (com K posições no vetor), respectivamente. É importante destacar que se m for igual a unidade tem-se um problema mono-objetivo. Além disso, também pode-se formular o problema como sendo um em que deseja-se maximizar uma ou mais funções objetivo.

2.4 Algoritmos Heurísticos de Otimização

Nesta seção são apresentados alguns dos principais algoritmos de otimização heurística encontrados na literatura, de forma a ilustrar, brevemente, o seu funcionamento e a sua inspiração da natureza.

2.4.1 Algoritmo Genético

O Algoritmo Genético (AG) foi a primeira estratégia no contexto evolutivo proposta. Esta é fundamentada na genética populacional proposta por Darwin, onde a variabilidade entre os indivíduos de uma população de organismos que se reproduzem sexualmente é descrita pela mutação e recombinação genética (MICHALEWICZ, 1998). A formulação matemática detalhada deste método e todas as suas etapas podem ser encontradas em Saramago (1999).

2.4.2 Colônia de Vagalumes

O algoritmo de Colônia de Vagalumes (CV) é um método baseado na bioluminescência dos vagalumes. Yang (2008) ao propô-lo, considerou que os vagalumes não possuem sexo, isto é; qualquer vagalume pode atrair ou ser atraído por outro. Além disso, que a atratividade de cada um é proporcional ao próprio brilho emitido e é inversamente proporcional à distância entre os vagalumes e este o brilho é determinado pela avaliação do objetivo. Neste caso, quanto melhor a avaliação em termos do valor da função objetivo, mais brilhante será o vagalume. A formulação matemática é descrita com detalhes em Yang (2008).

2.4.3 Enxame de Partículas

O algoritmo de otimização por Enxame de Partículas (EP) foi proposto por Kennedy & Eberhart (1995) e é baseado no comportamento social observado em espécies de pássaros e cardume de peixes, ou até pelo comportamento social humano. A teoria sócio-cognitiva que guia esse algoritmo é embasada na experiência de cada indivíduo de uma população e na estimativa da qualidade dessa experiência, a chamada aprendizagem individual ou cognitiva. Sendo indivíduos sociais, estes ainda possuem ou podem possuir conhecimento sobre o comportamento dos seus vizinhos, também conhecido esse conceito como transmissão cultural ou social. Portanto, de forma geral, o algoritmo de EP se fundamenta em uma população de indivíduos capazes de interagir entre si e com o meio ambiente e, com base nas propriedades que esses indivíduos possuem de auto avaliação, comparação e interação, estes conseguem lidar com as situações que o ambiente lhes apresenta. A formulação matemática detalhada deste algoritmo pode ser encontrada em Kennedy & Eberhart (1995).

2.4.4 Evolução Diferencial

A Evolução Diferencial (ED) é uma abordagem baseada no uso de operações vetoriais para a geração de um candidato em potencial à solução do problema de otimização (STORN & PRICE, 1996). Esta estratégia foi modificada a partir do algoritmo híbrido denominado de Recozimento Genético (PRICE, 1994), de forma a trabalhar com codificação de ponto flutuante e operações aritméticas. O algoritmo de ED requer poucas variáveis de controle e apresenta boa adaptabilidade à computação paralela (STORN & PRICE, 1996). A formulação matemática deste algoritmo pode ser encontrada em Storn & Price (1996).

2.4.5 Polinização de Flores

Proposto por Yang (2012), o Algoritmo de Polinização de Flores (APF) tem como inspiração biológica o processo de polinização que ocorre na natureza. Processo esse que ocorre pela transferência de pólen via contribuição de diferentes espécies, tais como insetos, pássaros, morcegos, entre outros e que compreende 90 % das plantas. De forma geral neste algoritmo tem-se as seguintes etapas de polinização: *i*) biótica cruzada (é um processo global de polinização e os agentes polinizadores se comportam conforme modelo de distribuição de Lévy); *ii*) abiótica e auto-polinização (são etapas consideradas como polinização local); *iii*) global e local são controladas por uma probabilidade de troca p ; *iv*) devido a proximidade física e fatores como o vento, a polinização local pode ter uma probabilidade de influência significativa na polinização global; *v*) cada planta possui apenas uma flor e cada flor libera apenas uma unidade de pólen. Matematicamente, as etapas deste algoritmo podem ser encontradas em Yang (2012).

2.4.6 Recozimento Simulado

Proposto por Metropolis *et al.* (1953), o método do Recozimento Simulado (RS) foi proposto tomando como base o recozimento de metais. De forma geral, o processo acontece em duas etapas. Na primeira, a temperatura do metal é aumentada. Já na segunda, o resfriamento deve ser realizado lentamente. Nesta segunda fase, executada lentamente, os átomos que compõem o material organizam-se numa estrutura uniforme com energia mínima. Isto resulta em que os átomos desse material ganhem energia para se movimentarem livremente e, ao arrefecer de forma controlada, dar-lhes uma melhor hipótese de se organizarem numa configuração com menor energia interna, para ter uma redução dos defeitos do material, como resultado prático. Baseado neste princípio, o algoritmo de RS consiste em realizar buscas de forma que a solução ótima global possa ser encontrada. A formulação matemática mais detalhada sobre o método pode ser encontrada em Metropolis *et al.* (1953).

2.5 Tratamento de Restrições

A maioria dos problemas de otimização realísticos são constituídos por restrições. Estas são advindas de limitações operacionais, físicas, ambientais, econômicas, entre outras. Para tratá-las pode ser empregadas diferentes abordagens encontradas na literatura (LOBATO, 2008). Nessa seção serão apresentados alguns dos principais métodos para o tratamento de restrições (COELHO, 2004; DEB, 2001).

2.5.1 Métodos Baseados em Funções de Penalidade

Este método configura a concepção mais simples e mais utilizada na literatura para o tratamento de restrições. Esta abordagem consiste na penalização que quaisquer violações. Para essa finalidade, o problema original restrito é reescrito como equivalente irrestrito conforme segue:

$$\min \Phi(x, r_p) = f(x) + r_p P(x) \quad (2.3)$$

onde $\Phi(x, r_p)$ é a função pseudo-objetivo, $P(x)$ é a função de penalidade e r_p um escalar associado a $P(x)$. Este escalar, indiretamente, define o nível de precisão requerido pela abordagem. Neste caso, se o valor de r_p for muito pequeno, permite-se violações. Por outro lado, se o valor de r_p tender a infinito a solução do problema formulado com a função pseudo-objetivo se aproxima da solução do problema original restrito (VANDERPLAATS, 1999; DEB, 2001).

De todas as metodologias existentes para essa transformação, pode-se destacar três principais, de acordo com Vanderplaats (1999) e com Edgar *et al.* (2001):

- **Método da Função de Penalidade Interior:**

Este método penaliza a função pseudo-objetivo ainda no interior da região viável através de uma tolerância pré-estabelecida pelo usuário (LOBATO, 2008). Este é descrito como segue:

$$P(x) = r'_p \sum_{j=1}^m \frac{-1}{g_j(x)} + \sum_{k=1}^l (h_k(x))^2 \quad (2.4)$$

A Equação (2.4) também pode ser representada por:

$$P(x) = \sum_{j=1}^m -\log(-g_j(x)) \quad (2.5)$$

esta é mais recomendada por ser melhor condicionada numericamente. $g_j(x)$ deve ser uma parcela negativa, $-1/g_j(x)$ será um número positivo que se aproximará do infinito quando $g_j(x) \rightarrow 0$ (VANDERPLAATS, 1999). Além disso, r'_p inicialmente é tomado como um valor elevado (normalmente da ordem de 10^6), sendo reduzido durante o processo iterativo por um fator de 0,3.

- **Método da Função da Penalidade Exterior:**

Este penaliza a função pseudo-objetivo quando qualquer uma das restrições é violada (LOBATO, 2008). Matematicamente tem-se:

$$P(x) = \sum_{j=1}^m (\max[0, g_j(x)])^2 + \sum_{k=1}^l (h_k(x))^2 \quad (2.6)$$

Em relação ao valor de r_p , deve-se destacar que pequenos valores (na ordem de 10) levam a uma minimização de $\Phi(x, r_p)$ sem muita dificuldade, porém com grandes violações nas restrições. Além disso, para valores elevados de r_p (na ordem de 10^6), as restrições serão atendidas com relação a uma tolerância pré-definida, mas esse valor pode levar a um mau condicionamento numérico do procedimento usado (VANDERPLAATS, 1999). De acordo com o autor, o processo iterativo é inicializado com r_p igual a 1 e seu valor é aumentado através de um fator de correção a cada iteração.

- **Método dos Multiplicadores de Lagrange Aumentado:**

Esse método possui como principal vantagem a redução de dependência de escolha de parâmetros de penalidade e da forma como são atualizados durante a otimização. Estabelece, a partir das condições de Kuhn-Tucker, uma função pseudo-objetivo que combina a função objetivo inicial com os multiplicadores de Lagrange aplicados às restrições e com funções de penalidade exterior (LOBATO, 2008). A função de penalidade é dada por:

$$P(x) = \sum_{j=1}^m (\lambda_j \psi_j + r_p \psi_j^2) + \sum_{k=1}^l (\lambda_{k+m} h_k(x) + r_p (h_k(x))^2) \quad (2.7)$$

onde:

$$\psi_j = \max \left(g_j(x), -\frac{\lambda_j}{2r_p} \right) \quad (2.8)$$

Os multiplicadores são atualizados de acordo com as seguintes fórmulas:

$$\begin{cases} \lambda_j^{p+1} = \lambda_j^p + 2r_p \left(\max \left(g_j(x), -\frac{\lambda_j^p}{2r_p} \right) \right) & j = 1, \dots, m \\ \lambda_{k+m}^{p+1} = \lambda_{k+m}^p + 2r_p h_k(x) & k = 1, \dots, l \end{cases} \quad (2.9)$$

De acordo com Vanderplaats (1999) e com Edgar *et al.* (2001), o fator de penalidade mantido constante durante todo o processo de otimização pode resultar em uma convergência prematura enquanto valores pequenos levam a uma convergência lenta. Devido a estes pontos, os autores aconselham o uso de fatores de penalidade atualizados dinamicamente a cada iteração, sendo utilizado inicialmente valores pequenos que permitem uma ampla exploração do espaço de busca e incrementado progressivamente.

2.5.2 Método da Pena de Morte

Este método consiste em simplesmente eliminar soluções não viáveis. De acordo com Van Kampen *et al.* (1996), a desvantagem dessa abordagem é a não exploração da informação sobre o domínio não viável.

2.5.3 Métodos Baseados na Preservação das Soluções Viáveis

Esse método se baseia na utilização de operadores especializados para transformar os indivíduos não-viáveis em indivíduos viáveis, assumindo apenas restrições lineares e a inicialização se dá a partir de uma configuração viável. As restrições lineares de igualdade são utilizadas para que se possa eliminar algumas variáveis, já as restrições de desigualdade são atualizadas a partir dos operadores que garantem que uma região viável seja respeitada (LOBATO, 2008).

2.6 Classificação dos Métodos para a Resolução de Problemas Mono e Multi-objetivos

Em relação à forma de tratamento dos problemas multi-objetivos, as abordagens podem ser divididos em três categorias principais (COELHO, 2004): *i*) Métodos *a Posteriori* - utilizados quando a Curva de Pareto, que é o conjunto das soluções não-dominadas do espaço de funções objetivo, é completamente determinada (LOBATO, 2008); *ii*) Progressivos - utilizados durante o processo de otimização, podendo ser encontradas abordagens na literatura que não fazem o uso de agregação de funções ou do conceito de otimalidade de Pareto; *iii*) Métodos *a Priori* - utilizados antes da inicialização do processo de otimização, onde é necessário que o usuário especifique preferências, que podem ser articuladas em termos de metas ou da importância relativa dos diferentes objetivos (LOBATO, 2008). Essas abordagens descritas no último item serão apresentadas a seguir.

2.6.1 Transformação do Problema de Otimização Multi em um Mono-objetivo

De acordo com Lobato (2008), a metodologia para a resolução de problemas de otimização multi-objetivos consiste em algumas etapas básicas:

- (a) Transformação do problema original em um problema com um único objetivo;
- (b) Utilização de uma técnica para tratamento de restrições (abordada na seção 2.5);
- (c) Após a reformulação do problema, resolvê-lo n vezes considerando uma abordagem particular de forma que a Curva de Pareto possa ser obtida.

• Método da Soma Ponderada (MSP)

Este é a abordagem mais popular entre os métodos *a Priori* existentes. Este consiste na transformação do problema original multi-objetivo em um equivalente puramente escalar, através da soma das m funções objetivo, conforme a Equação (2.10):

$$\min f(x) = \sum_{i=1}^m w_i f_i(x) \quad (2.10)$$

onde os pesos (fatores de ponderação) devem obedecer a seguinte relação:

$$\sum_{i=1}^m w_i = 1 \quad (2.11)$$

Deve-se chamar atenção ao fato de que os resultados obtidos são dependentes dos valores de w_i e que estes não refletem proporcionalmente a importância relativa dos objetivos, sendo apenas ponderações (OSYCZKA, 1984). Para problemas de otimização que não possuem funções expressas em unidades que assumam aproximadamente o mesmo valor numérico, deve-se modificar os pesos de forma que os mesmos reflitam a importância de cada objetivo (OSYCZKA, 1978). Essa modificação pode ser feita conforme a equação (2.12).

$$\min f^*(x) = \sum_{i=1}^m w_i f_i(x) c(i) \quad (2.12)$$

onde c_i são as constantes que normalizam a função objetivo e apresentam melhores resultados quando adota-se $c_i = 1/f_i^\circ$, onde f_i° é o i -ésimo valor da função objetivo otimizada individualmente com a presença de todas as restrições originais (OSYCZKA, 1984).

O método da Soma Ponderada é uma abordagem desenvolvida para o tratamento de problemas convexos e seu uso em problemas não-convexos ou côncavos, não garante que uma solução possa ser obtida. Ainda, não existe garantia de que a curva de Pareto será bem distribuída mesmo variando w_i (OSYCZKA, 1984).

• Método da Otimização de Compromisso (MOC)

Essa metodologia é fundamentada na combinação dos objetivos a serem alcançados considerando a seguinte relação (VANDERPLAATS, 1999):

$$\min f(x) = \left(\sum_{k=1}^m \left(\frac{w_k(f_k(x) - f_k^*(x))}{f_k^{prior}(x) - f_k^*(x)} \right)^2 \right)^{0,5} \quad (2.13)$$

onde w_k são os fatores de ponderação da k -ésima função objetivo, $f_k(x)$ é a k -ésima função objetivo e, $f_k^*(x)$ e $f_k^{prior}(x)$ são respectivamente o melhor valor e o pior valor da k -ésima função objetivo.

Vanderplaats (1999) atribui a dificuldade da metodologia ao fato de se ter que arbitrar os valores dos pesos e para os melhores e piores valores da função objetivo, individualmente. Os fatores de ponderação deverão ser escolhidos como unitários, porém se utiliza um fator de ponderação maior para alguns casos, quando alguns objetivos são mais importantes que outros. $f_k^{prior}(x)$ é a função objetivo associada com a estimativa inicial de projeto x° , $f_k^*(x)$ é o valor mais difícil de ser atribuído por ser o valor desejado para a função objetivo, alternativamente, esse valor pode ser o valor ótimo de cada objetivo individualmente, considerando todas as restrições do problema original. Deve-se ressaltar que uma alteração em qualquer um desses parâmetros, pode levar a valores ótimos diferentes (VANDERPLAATS, 1999).

• Método do Critério Ponderado Exponencial (MCPE)

O MCPE é uma metodologia proposta por Athan & Papalambros (1996) em resposta à inabilidade do MSP em capturar pontos em regiões não convexas da Curva de Pareto:

$$\min f^*(x) = \sum_{i=1}^k (\exp(pw_i) - 1) \exp(pf_i(x)) \quad (2.14)$$

onde o argumento do somatório é uma função utilidade individual para a i -ésima função objetivo. Além disso, grandes valores atribuídos a p podem conduzir a uma saturação do processo (ATHAN & PAPALAMBROS, 1996).

Como alternativa para os métodos de transformação do problema multi-objetivo em um mono-objetivo, o conceito de Ótimo de Pareto se configura como a estratégia mais empregada na análise de problemas com mais de um objetivo. Este conceito é apresentado a seguir.

2.6.2 Ótimo de Pareto

A noção de “ótimo” foi inicialmente proposta por Francis Ysidro Edgeworth (EDGEWORTH, 1881) e aperfeiçoado pelo economista e socialista Vilfredo Pareto (PARETO, 1896). Essa definição de ótimo segundo esses idealizadores é baseada na convicção intuitiva de que um ponto x^* é ótimo se nenhum critério utilizado puder melhorar a solução sem piorar pelo menos um outro critério. Considerando isso, o ótimo de Edgeworth-Pareto geralmente não fornece uma solução única, fornecendo um conjunto de soluções denominadas de soluções não-dominadas.

Para melhor compreender tal conceito, considere o exemplo apresentado por Ticona (2003). Seja uma situação hipotética de compra de um carro, onde os critérios para a escolha sejam preço e conforto. A Figura 2 mostra várias alternativas de escolha.

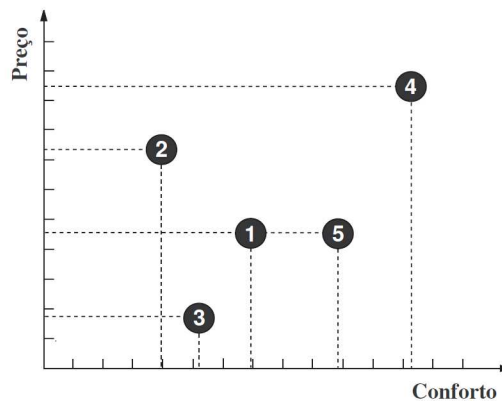


Figura 2: Opções de Compra de um Carro: Custo Versus Conforto. Reproduzido de Ticona (2003).

O objetivo desse exemplo é obter uma solução que tenha máximo conforto pelo menor preço possível, o famoso custo-benefício. O gráfico apresenta cinco opções de compra diferentes, cada número representando um carro. De forma intuitiva, a opção 1 é descartada uma vez que a 5 oferece mais conforto pelo mesmo preço. Além disso, exclui-se também a solução 2, uma vez que esta apresenta menor conforto que a 5, por um preço maior. Restam como boas opções de compra as alternativas 3, 4 e 5. Analisando essas três, nenhuma é necessariamente melhor que a outra pois apresentam diferentes níveis de custo-benefício, uma vez que uma apresenta uma opção que possui melhor conforto, mas maior preço, enquanto outra menor custo e menor conforto, etc. Nesse caso, existe o chamado *compromisso* entre os objetivos, ou seja, quanto maior o conforto, maior o preço do automóvel e vice-versa.

Em relação à dominância, diz-se que uma solução domina a outra se seus valores são melhores em todos os objetivos. Nesse exemplo, a solução 5 domina a solução 1 e não é dominada por nenhuma outra. Assim como as soluções 3 e 4 não são dominadas. Como não se conhece a importância de cada objetivo (se conforto ou preço seriam mais relevantes para a solução), então pode-se dizer que as soluções 3, 4 e 5 são igualmente boas. Esse conjunto de soluções ótimas encontrados é o conjunto não-dominado. As soluções 1 e 2 compõem o conjunto dominado.

Deb (2001) define propriedades para esses conjuntos, de forma que:

- Qualquer solução do conjunto não-dominado deve ser não-dominada em relação a outra solução do mesmo conjunto, ou seja, no exemplo, as soluções 3, 4 e 5 não devem ser dominadas por

elas mesmas de nenhuma forma;

- Qualquer solução contida no conjunto não-dominado deve ser dominado pelo menos por uma solução do conjunto não-dominado (no exemplo, as soluções 1 e 2 são dominadas pela solução 5).

2.7 Atualização dos Parâmetros em Algoritmos de Otimização

Todos os algoritmos, sejam eles clássicos ou heurísticos, possuem parâmetros a serem informados pelo usuário para serem executados (CARVALHO, 2018). Normalmente esses parâmetros são considerados constantes ao longo das gerações, porém estes não garantem a convergência ou a diversidade exploração do espaço de busca de forma a proporcionar a diversidade da população (ZAHARIE, 2002). Cabe ressaltar que, em relação ao número de parâmetros que devem ser informados durante a resolução de um problema de otimização, os algoritmos evolutivos apresentam um número superior aos clássicos.

A literatura especializada apresenta algumas propostas para melhorar o desempenho de métodos de otimização heurísticos via atualização dos seus parâmetros, bem como na redução do número de avaliações da função objetivo. Em sua maioria, as estas abordagens produzem como efeito a diminuição da dependência dos métodos em relação à definição inicial dos parâmetros, e ainda, aumenta a intensidade de exploração do espaço de busca. Neste contexto, Zaharie (2002) propôs estratégias para a atualização dos parâmetros no algoritmo de Evolução Diferencial baseando em informações estatísticas da população. Seguindo no mesmo algoritmo, Lobato *et al.* (2007) avaliaram o desempenho de diferentes Modelos Caóticos de Busca associado ao algoritmo de ED. Coelho & Mariani (2006a) utilizaram uma estratégia híbrida baseada na associação entre modelos caóticos de busca, ED e Programação Quadrática Sucessiva.

Coelho *et al.* (2011) propuseram um algoritmo caótico de busca associado ao Algoritmo de Colônia de Vagalumes para o tratamento de problemas de confiabilidade. Coelho & Mariani (2013) propuseram um algoritmo que modifica os parâmetros do ACV segundo a distribuição Gaussiana. Mahadavi *et al.* (2007) apresentaram um método para o refinamento no algoritmo de Busca Harmônica com a variação dinâmica de seus parâmetros. Omar *et al.* (2016), em seu estudo sobre o Algoritmo de Busca Harmônica, propôs uma estratégia para a redução do tamanho da memória harmônica a partir de limites pré-definidos. Lobato & Steffen Jr (2016) propuseram um algoritmo para a atualização dinâmica da taxa de perturbação, da probabilidade de cruzamento e do número de indivíduos da população no algoritmo de Evolução Diferencial. Para essa finalidade foram considerados os conceitos de diversidade da população e de taxa de convergência. Os resultados demonstraram que a utilização de informações sobre o andamento do processo evolutivo é de grande importância para uma atualização adequada dos parâmetros do referido algoritmo.

3 Busca Fractal Estocástica - BFE

Conforme descrito anteriormente, a presente contribuição tem por objetivo avaliar o desempenho do algoritmo de Busca Fractal Estocástica (BFE) em problemas matemáticos e de engenharia. Em linhas gerais, esta estratégia baseia-se no uso de fractais para imitar o fenômeno de deslocamento de partículas (candidatos a solução do problema de otimização) no processo de difusão. Neste movimento, as partículas tendem a explorar o espaço de busca considerando uma propriedade de difusão empregada em fractais aleatórios.

Para uma boa compreensão deste algoritmo de otimização, a próxima seção apresenta uma breve revisão sobre a concepção conceitual do mesmo, bem como dos seus operadores e da aplicação prática dos mesmos. Finalmente, os operadores considerados para aumentar a capacidade de exploração neste algoritmo e a sua adaptação para a determinação de todas as soluções em problemas multi-modais de forma automática são apresentados no final deste capítulo.

3.1 Concepção Conceitual

3.1.1 Fractais

De forma geral, define-se fractal como sendo a propriedade que um objeto possui para explicar sua auto-similaridade em todas as escalas (SALIMI, 2015). Como exemplo, tomando uma pequena porção de um objeto, i.e., uma célula de um corpo, podemos perceber nessa mesma célula as características presentes no corpo todo, como características da pele, cor do olho, cor e textura do cabelo, além de apresentar também características que não são manifestadas no organismo inteiro, mas que se encontra presente no DNA, sendo uma herança genética. Neste contexto, os objetos de estudo passam a ser avaliadas, além de quantitativa, também qualitativamente.

As principais características dos fractais são (SALIMI, 2015):

- **Extensão infinita dos limites:** Os limites fractais são enrugados e complexos, mantendo o padrão em várias escalas, desde as mínimas partes até o todo. Apesar dos limites serem enrugados, são medidos linearmente como retas e a extensão dos seus limites depende do tamanho da unidade-padrão de medida. Sendo assim, quanto mais se reduz o tamanho da unidade-padrão, maior é a extensão do que se mede, gerando um aumento do alcance de detalhes.
- **Permeabilidade dos limites:** é a característica do sistema que permite a troca de energia e de matéria no meio ambiente ou nos processos humanos, proporcionando um intercâmbio de dados, gerando informação, conhecimento e aumento da melhoria dos relacionamentos em todas as escalas. A mudança de estado e as alterações de um padrão fractal para outro são propiciadas por essa característica.
- **Auto similaridade das formas e características:** é a semelhança presente nas formas e nas características dos objetos e nos processos que podem ser observadas em várias escalas, do todo em relação aos seus componentes, exprimindo uma invariância de escala.

3.1.2 Busca Fractal

A Busca Fractal (BF) é o método base que serviu de inspiração para o desenvolvimento do algoritmo de Busca Fractal Estocástica (BFE). De forma geral, o algoritmo de BF utiliza as seguintes premissas (SALIMI, 2015):

- Cada partícula da população possui uma energia potencial elétrica;

- Cada partícula se difunde, criando outras partículas aleatórias e a energia potencial da partícula mãe é dividida entre as partículas geradas;
- Apenas as melhores partículas são mantida em cada geração, sendo o restante eliminadas da análise.

O algoritmo de BF consiste na geração de uma população inicial, que possui N_p partículas. Cada partícula P_i possui uma energia potencial E_i definida conforme a Equação (3.1):

$$E_i = \frac{E}{N_p} \quad (3.1)$$

onde E é a energia potencial elétrica máxima considerada para a solução do problema e P é o número de partículas que compõem a população inicial.

Em cada geração, cada partícula é difundida criando outras partículas baseando-se no vôo de Lévy (VISWANATHAN *et al.*, 2008). Este pode ser entendido como um tipo de movimento Browniano que envolve etapas aleatórias em que, de vez em quando, o objeto, aleatoriamente, dá um salto para outra região do espaço. A distribuição de Lévy é apresentada na Equação (3.2):

$$L(\lambda) = \frac{\mathcal{N}(0, \sigma_u^2)}{|\mathcal{N}(0, \sigma_v^2)|^{1/\lambda}} \quad (3.2)$$

onde $\mathcal{N}(\cdot)$ representa a distribuição do tipo normal, λ é o índice de distribuição, compreendido no intervalo de $[0, 2]$ (sendo geralmente arbitrado como sendo igual a 1,5), σ_v é um parâmetro definido como sendo igual a unidade e σ_u é uma função definida como (Yang, 2010):

$$\sigma_u = \left(\frac{\Gamma(1 + \lambda) \sin(\pi\lambda/2)}{\lambda\Gamma((1 + \lambda)/2)2^{(\lambda-1)/2}} \right)^{1/\lambda} \quad (3.3)$$

em que $\Gamma(\cdot)$ é a função gamma.

O processo de difusão cria partículas que são distribuídas aleatoriamente em torno da partícula original, conforme pode ser observado na Figura 3.

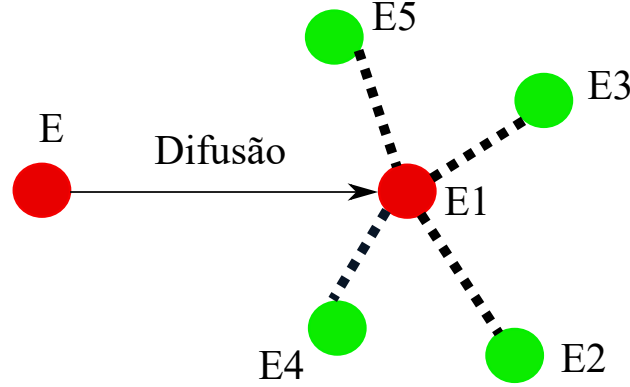


Figura 3: Representação Esquemática da Difusão de uma Partícula. Adaptado de Salimi (2015).

Para esse processo são utilizados, além da estratégia proposta por Lévy, o método de distribuição Gaussiano, conforme as Equações (3.4) e (3.6).

$$x_i^{dif} = x_i + \alpha_i^{dif} L(\lambda) \quad (3.4)$$

onde i_{dif} é um índice compreendido entre $[0, \text{número máximo de difusões}]$.

Para melhor aproveitar as vantagens das duas abordagens, o algoritmo de BF aplica ambas as estratégias, todavia de forma aleatória. Para aumentar a velocidade de convergência do processo de otimização, o parâmetro α pode-se avaliar duas estratégias para a sua atualização, a saber, as Equações (3.5a) e (3.5b):

$$\alpha_i = \frac{\log(\min(\hat{E}))(U - D)}{N_{ger} \log(E_i)} \quad (3.5a)$$

$$\alpha_i = \frac{(U - D)}{(N_{ger} \log(E_i))^{pot}} \quad (3.5b)$$

onde $\min(\hat{E})$ é a energia mínima que cada partícula apresenta. U e D são os limites superior e inferior do espaço de projeto, N_{ger} é o número de gerações, E_i é a energia da partícula P_i e pot é considerada a potência. Conforme destacado por Salimi (2015), a primeira garante uma convergência mais rápida, enquanto a segunda consegue melhor estimar, com maior precisão, o resultado final.

Para a geração de um candidato em potencial, o método de distribuição Gaussiano é empregado. Este pode ser representado como:

$$x_i^{q_{dif}} = x_i + \beta \text{Gaussiana}(P_i|BP|) - (\gamma BP - \gamma' P_i) \quad (3.6)$$

onde q_{dif} é o número de partículas adquiridas no processo de difusão, γ e γ' são números aleatórios compreendidos entre 0 e 1, $\text{Gaussiana}(P_i|BP|)$, P_i e $|BP|$ representam a média e desvio padrão da distribuição Gaussiana, respectivamente, e β é um parâmetro definido como sendo:

$$\beta = \frac{\log(N_{ger})}{N_{ger}} \quad (3.7)$$

em que N_{ger} é o número de iterações (gerações) corrente.

Após a difusão, a energia da partícula original é dividida entre as partículas geradas, considerando o melhor valor da função objetivo. Essa distribuição ocorre conforme a Equação (3.8):

$$E_i^j = \left(\frac{f_j}{f_i + \sum_{k=1}^{N_{ger}} f_k} \right) E_i \quad (3.8)$$

onde i é o índice da partícula original, j é o índice da partícula gerada, N_{ger} é a quantidade de gerações corrente. f_i é o valor da função objetivo para a partícula original e f_j o valor da função objetivo para a partícula gerada.

Para diminuir o custo de execução do algoritmo, apenas as melhores partículas geradas sobrevivem até a geração seguinte, sendo a maioria descartada. A energia das partículas descartadas é redistribuída às partículas sobreviventes e utilizada na criação de novas partículas. A equação de distribuição de energia é tal que:

$$E_{nova}^t = E_{antiga}^t + \left[\left(\frac{f_t}{\sum_{k=1}^{\zeta} f_k} f_k \right) \phi \right] \mu \quad (3.9)$$

em que ϕ é a quantidade de energia das partículas descartadas, μ é a taxa de distribuição de energia entre as partículas sobreviventes e as novas partículas, E_{antiga}^t e E_{nova}^t representam as energias da t -ésima partícula antes e depois da distribuição de energia e ζ é o número de partículas na iteração corrente.

O número de partículas criadas e alocadas no espaço de busca é definido conforme a Equação (3.10):

$$\vartheta = \frac{\log(\text{número de partículas descartadas})}{\log(\text{número máximo de difusões})} \quad (3.10)$$

A energia distribuída para cada nova partícula criada é igual para todas e dada conforme a Equação (3.11):

$$E'_c = \frac{\phi(1 - \mu)}{\vartheta} \quad c = 1, 2, \dots, \vartheta \quad (3.11)$$

3.2 O Algoritmo de BFE

Conforme enfatizado por Salimi (2015), embora o algoritmo de BF tenha um bom desempenho, o mesmo possui algumas desvantagens. Segundo o autor, uma das mais preocupantes é o fato de não existir troca de informações entre as partículas numa tentativa de acelerar o processo de convergência. Além disso, a possibilidade do aumento no número de partículas durante o processo pode inviabilizar o método do ponto de vista computacional, bem como na precisão do mesmo. Com o objetivo de minimizar tais obstáculos, Salimi (2015) incorporou ao algoritmo original a denominada fases de atualização, resultando em um novo método chamado de BFE. Neste novo algoritmo, duas fases distintas podem ser encontradas. A primeira, denominada de difusão, ocorre de forma semelhante à BF, isto é; permite que cada partícula se difunda em torno da sua posição e aumente a exploração da busca, o que incrementa as chances de encontrar os mínimos globais da função, bem como pode evitar a estagnação em um ponto de mínimo local. Já a segunda fase é denominada de difusão estática. Esta consiste, em vez de aumentar o número de partículas na geração, em apenas considerar o melhor candidato gerado na difusão descartando os restantes. Assim, para criar novas partículas são utilizados dois tipos de relações, definidas conforme as Equações (3.12a) e (3.12b):

$$GW_1 = \text{Gaussiana}(\mu_{BP}, \sigma) + (\epsilon \times BP - \epsilon' \times P_i) \quad (3.12a)$$

$$GW_2 = \text{Gaussiana}(\mu_P, \sigma) \quad (3.12b)$$

onde ϵ e ϵ' são números aleatórios distribuídos uniformemente no intervalo $[0,1]$, BP é a posição do melhor ponto, P_i é a i -ésima posição no grupo. Na Equação (3.12a), μ_{BP} corresponde a $|BP|$ e na Equação (3.12b), μ_P é igual a $|P_i|$. O parâmetro σ é definido como:

$$\sigma = \left| \frac{\log(N_{ger})}{N_{ger}} (P_i - BP) \right| \quad (3.13)$$

A inicialização da população P no algoritmo de BFE ocorre conforme outras estratégias com a mesma natureza. A população inicial é gerada segundo a Equação (3.14):

$$P_j = LB + \epsilon(UB - LB) \quad (3.14)$$

onde LB e UB são os limites de busca das variáveis de projeto, ϵ é um número aleatório distribuído uniformemente no intervalo $[0,1]$ e $j = 1, 2, \dots, N$, sendo N a dimensão do problema. Após a geração da população inicial, avalia-se cada candidato de acordo com a função objetivo de modo a encontrar o melhor entre eles.

Conforme mencionado anteriormente, o objetivo do processo de difusão é a exploração do espaço de busca por cada partícula da população. Para melhorar essa propriedade, são aplicados dois

procedimentos estatísticos. O primeiro atua em cada candidato da população, onde é realizado o ranqueamento desses candidatos no valor da função objetivo, conforme a Equação (3.15):

$$Pa_i = \frac{\text{rank}(P_i)}{N_p} \quad (3.15)$$

em que $\text{rank}(P_i)$ é o ranqueamento de P_i e N_p é o número total de pontos.

Para cada ponto que satisfaz a condição $Pa_i < \epsilon$, sendo ϵ um número aleatório distribuído uniformemente no intervalo $[0,1]$, é aplicada a Equação (3.16) para atualizar cada componente de P_i , garantindo assim uma maior diversidade da amostra.

$$P'_i(j) = P_r(j) - \epsilon(P_t(j) - P_i(j)) \quad (3.16)$$

onde P_r e P_t são pontos escolhidos aleatoriamente na população e ϵ um número aleatório distribuído uniformemente no intervalo $[0,1]$.

Já o segundo procedimento modifica a posição de cada vetor através de um novo ranqueamento dos pontos obtidos no primeiro procedimento estatístico. Se $Pa_i < \epsilon$ é satisfeito para cada ponto P'_i , a posição de P'_i é modificada segundo as equações (3.17a) e (3.17b):

$$P''_i = P'_i - \epsilon'(P'_t - BP), \quad \text{se } \epsilon' \leq 0,5 \quad (3.17a)$$

$$P''_i = P'_i + \epsilon'(P'_t - P'_r), \quad \text{se } \epsilon' > 0,5 \quad (3.17b)$$

em que P'_t e P'_r são posições aleatórias selecionadas no primeiro procedimento e ϵ' são números aleatórios gerados pela distribuição Normal Gaussiana. O novo ponto P''_i substitui P'_i se tiver um custo melhor que P'_i .

A Figura 4 apresenta os fluxogramas do algoritmo de BFE, da fase de difusão e da fases de atualização, respectivamente.

Como o aprimoramento do algoritmo original (BF), trabalhos em diferentes campos da ciência empregando a nova versão (BFE) podem ser encontradas na literatura especializada. Khalilpourazari *et al.* (2019) utilizaram o algoritmo de BFE para resolver problemas multi-objetivos complexos. Já Khalilpourazari & Khalilpourazary (2018) aplicaram o algoritmo de BFE em um problema de engenharia para a retificação de superfície. Hinojosa *et al.* (2018) utilizaram o referido algoritmo para a segmentação de imagens em procedimento de histologia em mamas. Chen *et al.* (2019) propuseram uma variante desse método a partir da perturbação de soluções em potencial para a determinação de parâmetros em sistemas fotovoltaicos.

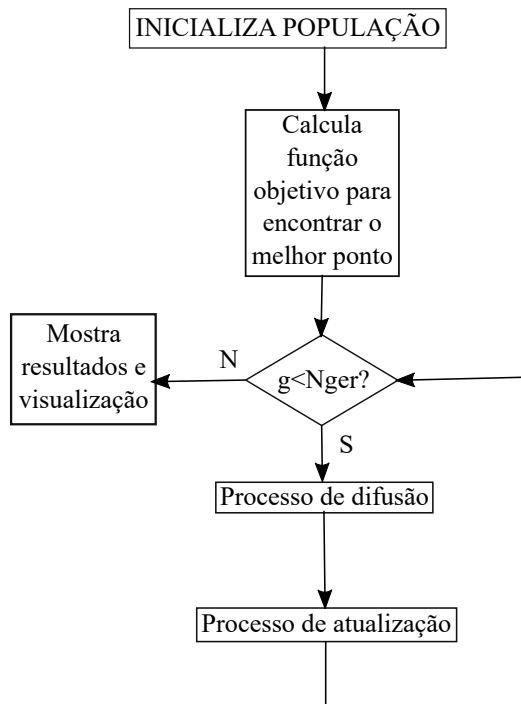
3.2.1 Aplicação Passo a Passo do Algoritmo de BFE

Essa seção tem o objetivo de demonstrar todas as operações envolvidas nas etapas do algoritmo de BFE, que vai desde a geração inicial de indivíduos até a atualização da população. Para esta finalidade considere o seguinte problema de otimização (HAUPT & HAUPT, 1998):

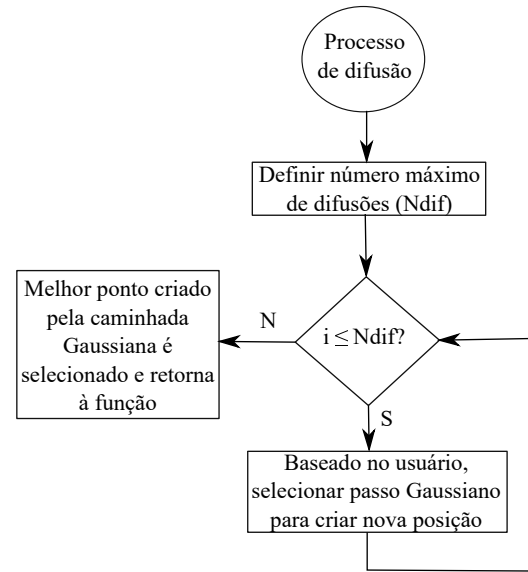
$$\min f(x) = x_1 \sin(4x_1) + 1, 1x_2 \sin(2x_2) \quad (3.18)$$

em que x_i ($i=1,2$) representa o vetor de variáveis de projeto, ambos no intervalo $[0,10]$. A Figura 5 apresenta o gráfico 3D da função considerada.

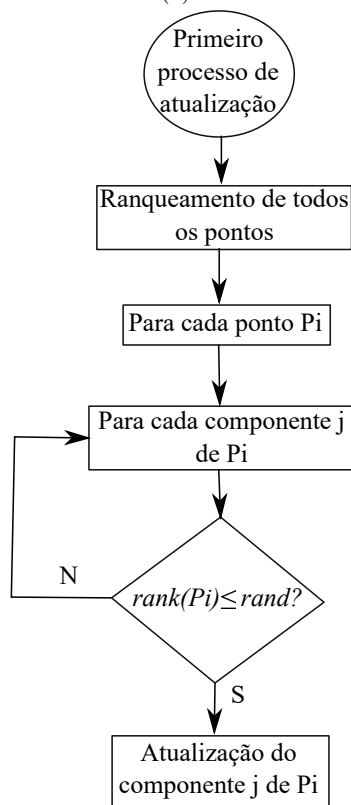
Os parâmetros do algoritmo de BFE utilizados nesta aplicação são (Salimi, 2015): número de indivíduos da população ($N_p = 5$), número de difusões ($N_{dif} = 2$), passo do caminho Gaussiano (passo igual a 1 e passo igual a 0 - N_{passo}). Cabe ressaltar que os parâmetros utilizados nesta seção justificam-se pelo caráter didático da mesma.



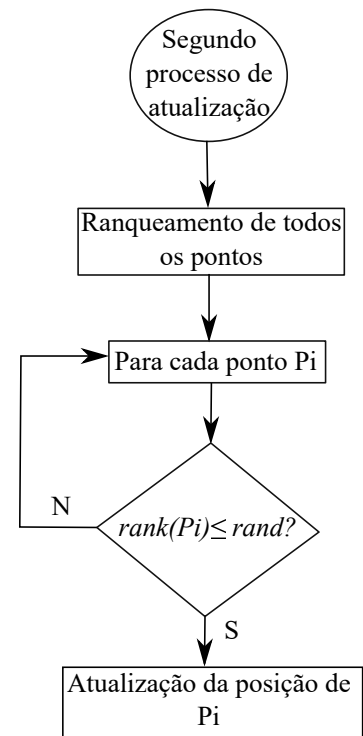
(a) Estrutura Geral.



(b) Processo de Difusão.



(c) Primeira Atualização.



(d) Segunda Atualização.

Figura 4: Algoritmo de Busca Fractal Estocástica.

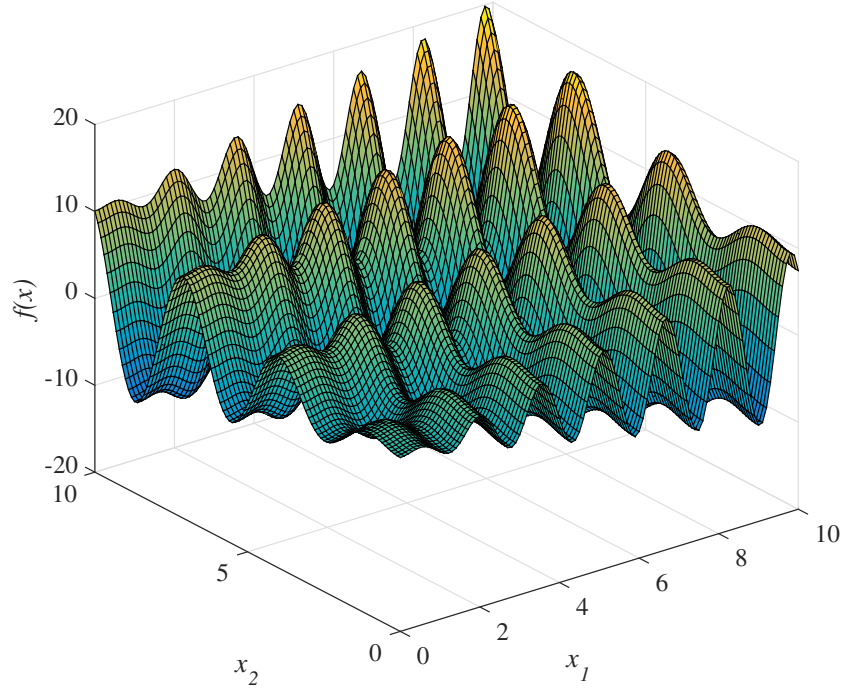


Figura 5: Gráfico Tridimensional do Exemplo Analisado.

Inicialmente, gera-se a população inicial considerando os parâmetros definidos, bem como os limites laterais impostos. Para essa finalidade considera-se a Equação (3.19):

$$P_i = LB + (UB - LB) \times \text{rand} \quad (3.19)$$

onde P_i é o vetor posição do i -ésimo indivíduo na população, LB é o vetor limite inferior, UB é o vetor limite superior do problema e rand é um gerador de números aleatórios definidos no intervalo $[0,1]$. Neste caso, para o i -ésimo contador, um número aleatório diferente é considerado.

Dessa forma, a população inicial gerada utilizando uma semente igual a 0 no gerador de números aleatórios, bem como a avaliação da mesma conforme a Equação (3.18) é apresentada na Tabela 1. Enfatiza-se que essa população inicial é utilizada para os dois estudos ($N_{passo} = 1$ e $N_{passo} = 0$).

Tabela 1: Inicialização do Algoritmo de BFE.

P_i	rand_1	rand_2	x_1	x_2	$f(x)$
1	0,815	0,098	8,147	0,975	8,508
2	0,906	0,278	9,058	2,785	-11,014
3	0,127	0,547	1,270	5,469	-7,191
4	0,913	0,958	9,134	9,575	-5,270
5	0,632	0,965	6,324	9,649	5,617

Os candidatos apresentados na Tabela 1 são ordenados de acordo com o melhor valor da função objetivo, conforme a Tabela 2.

Tabela 2: População Inicial do Algoritmo de BFE Ordenada.

P_i	x_1	x_2	$f(x)$
1	9,058	2,785	-11,014
2	1,270	5,469	-7,191
3	9,134	9,575	-5,270
4	6,324	9,649	5,617
5	8,147	0,975	8,508

O processo de difusão consiste na geração de novos pontos a partir da caminhada Gaussiana, conforme as Equações (3.12a) e (3.12b), sendo que, quando N_{passo} é igual a 1, a primeira equação é utilizada e quando é igual a 0, a segunda. Os candidatos obtidos, para cada valor de N_{passo} durante o processo, são apresentados na Tabela 3.

Tabela 3: Representação do Processo de Difusão no algoritmo de BFE.

P_i	$N_{passo} = 1$			$N_{passo} = 0$		
	x_1	x_2	$f(x)$	x_1	x_2	$f(x)$
1	9,058	2,785	-11,014	9,058	2,785	-11,014
2	1,270	5,469	-7,191	1,270	5,469	-7,191
3	9,134	9,575	-5,270	9,134	9,575	-5,270
4	7,830	4,984	-3,572	6,324	9,649	5,617
5	1,386	6,959	6,539	8,147	0,975	8,508

A primeira e segunda atualizações ocorrem conforme as Equações (3.15) a (3.17b). Os resultados obtidos durante esse processo são apresentados nas Tabelas 4 a 7, especificando quando o candidato foi atualizado de acordo com as condições do referido algoritmo.

Tabela 4: Processo de Ranqueamento no Algoritmo de BFE.

P_i	Pa_i
1	1,000
2	0,800
3	0,600
4	0,400
5	0,200

A Tabela 5 apresenta os candidatos obtidos a partir da primeira atualização de candidatos do algoritmo. É importante relembrar que, nessa etapa, são atualizados os elementos de cada candidato, de acordo com a parcela aleatória do algoritmo, comparada ao vetor Pa_i , correspondente ao processo de ranqueamento. Sendo assim, os elementos das partículas que apresentam pior valor de função objetivo possuem mais chances de serem atualizadas, conforme observado nas partículas 3 e 5 (com

N_{passo} igual a 1) e partículas 2, 4 e 5 (com N_{passo} igual a 0). P_r e P_t representam posições aleatórias selecionadas no primeiro procedimento.

Tabela 5: Primeira Etapa para a Atualização de Candidatos no Algoritmo de BFE.

	P_i	P_r	P_t	$rand > Pa_i$	x_1	$rand > Pa_i$	x_2	$f(x)$
$N_{passo} = 1$	1	5	3	Não	9,058	Não	2,785	-11,014
	2	3	4	Sim	1,270	Não	5,469	-7,191
	3	1	2	Sim	9,340	Sim	5,880	-7,772
	4	4	5	Não	7,830	Sim	4,984	-3,572
	5	2	1	Não	1,386	Sim	8,721	-10,398
$N_{passo} = 0$	1	4	4	Não	9,058	Não	2,785	-11,014
	2	3	5	Sim	4,111	Não	5,469	-8,766
	3	5	1	Não	9,134	Não	9,575	-5,270
	4	1	2	Sim	3,955	Sim	4,963	-3,071
	5	2	3	Sim	0,788	Sim	3,674	3,530

Ao final da etapa, o algoritmo ordena os candidatos de acordo com o melhor valor da função objetivo obtido pelos mesmos, conforme Tabela 6.

Tabela 6: Primeira Etapa para a Atualização de Candidatos no Algoritmo de BFE Ordenado.

	$N_{passo} = 1$			$N_{passo} = 0$		
P'_i	x_1	x_2	$f(x)$	x_1	x_2	$f(x)$
1	9,058	2,785	-11,014	9,058	2,785	-11,014
2	1,386	8,721	-10,398	4,111	5,469	-8,766
3	9,340	5,880	-7,772	9,134	9,575	-5,270
4	1,270	5,469	-7,191	3,955	4,963	-3,071
5	7,830	4,984	-3,572	0,788	3,674	3,530

A Tabela 7 apresenta os candidatos obtidos através do segundo processo de atualização do algoritmo. Este consiste na modificação da posição dos pontos candidatos. Além de contar com a parcela aleatória para realizar a atualização das posições, o valor da função objetivo obtidos pela atualização também é comparado com o valor obtido por P'_i (na primeira atualização), sendo que, a nova posição só será incorporada na população se possuir melhor valor de $f(x)$. No caso apresentado, passa pela atualização os candidatos 5 (com N_{passo} igual a 1), 3, 4 e 5 (com N_{passo} igual a 0), porém, somente o último candidato citado se mostra com menor $f(x)$. P'_r e P'_t representam posições aleatórias selecionadas no segundo procedimento.

Tabela 7: Segunda Etapa de atualização no Algoritmo de BFE.

	P'_i	$rand > Pa_i$	P'_r	P'_t	$rand < 0,5$	$f(P'') < f(P')$	x_1	x_2	$f(x)$
$N_{passo} = 1$	1	Não	-	-	-	-	9,058	2,785	-11,014
	2	Não	-	-	-	-	1,386	8,721	-10,398
	3	Não	-	-	-	-	9,340	5,880	-7,772
	4	Não	-	-	-	-	1,270	5,469	-7,191
	5	Sim	1	4	Sim	Não	7,830	4,984	-3,572
$N_{passo} = 0$	1	Não	-	-	-	-	9,058	2,785	-11,014
	2	Não	-	-	-	-	4,111	5,469	-8,766
	3	Sim	5	4	Sim	Não	9,134	9,575	-5,270
	4	Sim	1	4	Sim	Não	3,955	4,963	-3,071
	5	Sim	4	5	Não	Sim	6,987	3,244	2,951

A Tabela 8 apresenta os resultados obtidos, após o fim de uma geração, pela execução do algoritmo de BFE considerando os dois valores para o parâmetro N_{passo} .

Tabela 8: Resultados Obtidos com a Execução do Algoritmo de BFE.

	$N_{passo} = 1$			$N_{passo} = 0$		
P''_i	x_1	x_2	$f(x)$	x_1	x_2	$f(x)$
1	9,058	2,785	-11,014	9,058	2,785	-11,014
2	1,386	8,721	-10,398	4,111	5,469	-8,766
3	9,340	5,880	-7,772	9,134	9,575	-5,270
4	1,270	5,469	-7,191	3,955	4,963	-3,071
5	7,830	4,984	-3,572	6,987	3,244	2,951

Ainda que os melhores valores obtidos utilizando $N_{passo} = 1$ e $N_{passo} = 0$ tenham sido os mesmos (isto se deve ao fato de apenas uma geração foi realizada, o que, para a referida aplicação não implicou na melhor do(s) melhor(es) indivíduo(s) da população corrente), pode-se perceber uma diferença na evolução de cada população utilizando cada um dos valores considerados para esse parâmetro. Neste caso, observa-se que os candidatos encontrados pelo algoritmo utilizando $N_{passo} = 1$ se aproximaram, na média, do valor ótimo para o problema em questão ($f = -18,5546$ (HAUPT & HAUPT, 1998)).

A Figura 6 apresenta um gráfico do tipo curva de nível, bem como a população inicial e as atualizadas segundo os dois passos considerados em uma geração.

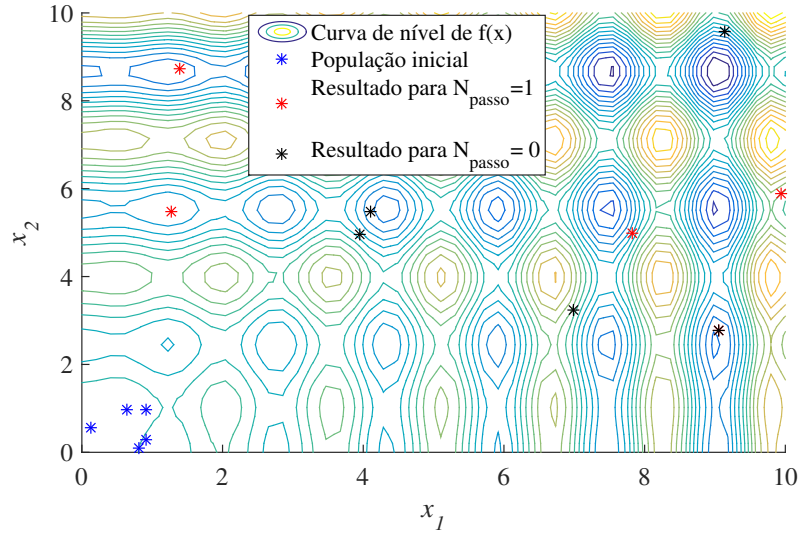


Figura 6: Atualização da População para o Exemplo Analisado.

Nesta figura, conforme o gráfico tridimensional (ver a Figura 5), observa-se que a função analisada possui ótimos locais (máximo e mínimo) e um ótimo global, sendo este localizado em $x_1=9,0389$ e $x_2=8,6681$, o que corresponde ao valor já mencionado de $f=-18,5546$. Os pontos representados pela cor azul são referentes à população inicial, os de cor vermelha e preta são referentes às populações obtidas pelo algoritmo de BFE ao final de uma geração utilizando os valores de N_{passo} iguais a 1 e 0, respectivamente. Nesta mesma figura observa-se que, após uma geração, ambas as populações atualizadas encontram-se mais *próximas* à solução global, o que implica que o algoritmo de BFE, na média, evoluiu em relação à população original, puramente aleatória. Cabe ressaltar que os candidatos representados pelos pontos vermelhos estão *mais próximos* do ótimo global quando comparados aos candidatos representados pelos pontos pretos. Sendo assim, até a presente geração, conclui-se que a caminhada Gaussiana mais adequada para o problema apresentado, que é simples, é aquela considerando $N_{passo} = 1$. Finalmente, espera-se que, aumentando-se o número de gerações, o algoritmo convirja em direção da solução global.

3.3 Operadores Propostos para Aumentar a Diversidade e Reduzir o Custo Computacional no Algoritmo de BEF

Em quaisquer métodos baseados em população empregados para a resolução de problemas de otimização, faz-se necessário a definição dos seus parâmetros. O uso de um conjunto de parâmetros *default* sugerido pela literatura especializada não implica na obtenção da solução global ou em um número otimizado de avaliações da função objetivo (LOBATO & STEFFEN JR, 2016). Assim, conforme apresentado no capítulo anterior, inúmeras estratégias podem ser combinadas para melhorar o desempenho deste tipo de algoritmo. Neste contexto, com o objetivo de aumentar a capacidade de exploração e reduzir o número de avaliações da função objetivo no algoritmo de BFE, a presente seção apresenta operadores propostos para essa finalidade.

3.3.1 Atualização Dinâmica do Número de Difusões (N_{dif}) e do Número de Passos da Caminhada Gaussiana (N_{passo})

Nesta contribuição, propõem-se a variação dos parâmetros N_{dif} e N_{passo} considerando três procedimentos diferentes, a saber, um puramente aleatório e os outros dois usando modelos caóticos de busca (Mapa Logístico e Mapa Seno) associados ao algoritmo de BFE (LOBATO *et al.*, 2007).

- Variação Aleatória

Para esta finalidade, os referidos parâmetros são modificados via uso de um gerador de números aleatórios distribuídos uniformemente entre $[0,1]$, cada qual adequado à magnitude correspondente. Neste caso, o algoritmo de Busca Fractal Estocástico Aleatória (BFEA) deve sempre garantir que N_{dif} seja maior que 1 e que N_{passo} esteja contido no intervalo de $[0 \ 1]$.

- Mapa Logístico

De acordo com Ott (2002), dentro do contexto dos modelos caóticos de busca, os mapas de busca uni-dimensionais são sistemas simples que tem a capacidade de geração de movimento caótico (LOBATO *et al.*, 2007). Para essa finalidade é proposto o algoritmo de Busca Fractal Estocástico baseado em Mapa Logístico (BFEML), cuja atualização se dá segundo o mapa logístico definido como (OTT, 2002):

$$x_{n+1} = \mu x_n (1 - x_n) \quad (3.20)$$

em que μ é um parâmetro definido no intervalo $[3,57 \ 4]$ de forma que a expressão acima resulte em um modelo caótico.

- Mapa Seno

Para esta atualização, propõe-se o algoritmo de Busca Fractal Estocástico baseado no Mapa do tipo Seno (BFEMS), cuja expressão é dada como (DEVANEI, 1987):

$$x_{n+1} = \frac{\mu}{4} \sin(\pi x_n) \quad (3.21)$$

em que μ é um parâmetro definido no intervalo $[0,001 \ 4]$ de forma que a expressão acima resulte em um modelo caótico.

Cabe enfatizar que ambos os parâmetros N_{dif} e N_{passo} podem ser atualizados considerando os mapas apresentados. Assim, definindo-se um valor inicial para cada um destes parâmetros (neste caso o valor *default*) e o valor do parâmetro μ , estes modelos podem ser empregados para variar N_{dif} e N_{passo} ao longo das gerações. Além disso, apesar do número de difusões, a priori, ser um definido como um valor inteiro, Salimi (2015) ressalta que este não precisa ser inteiro, o que permite o uso das estratégias apresentadas a este parâmetro.

3.3.2 Atualização Dinâmica do Tamanho da População

Tradicionalmente, o tamanho da população nos algoritmo heurísticos é considerado fixo durante todo o processo de otimização. Do ponto de vista de implementação, tal característica simplifica os algoritmos, mas por outro lado deixa de considerar um aspecto muito interessante da evolução biológica, a saber, a variação do número de indivíduos da população, que é inerente aos sistemas biológicos reais. Em termos práticos, se o número de gerações em um algoritmo heurístico tender ao infinito, implica que a população caminha para se tornar homogênea, isto é, todos os candidatos da população terão, aproximadamente, o mesmo valor de função objetivo. Neste caso, se essa condição

é satisfeita, não existe a necessidade de avaliar todos os candidatos, o que implica na redução do número de avaliações da função objetivo (LOBATO & STEFFEN JR, 2016).

Para esta finalidade, o objetivo desta seção é apresentar uma estratégia para a redução do tamanho da população ao longo das gerações. Tal abordagem é baseada no trabalho de Omar *et al.* (2016), em que um operador para reduzir o tamanho da população no algoritmo de Busca Harmônica foi proposto. De forma geral, esta consiste na diminuição do tamanho da memória harmônica (população) de forma linear, a partir de 50% de iterações decorridas no algoritmo, começando a partir do limite superior definido para o tamanho da memória até o limite inferior.

Assim como idealizado por Omar *et al.* (2016), no presente estudo é proposta uma abordagem semelhante e denominada por Busca Fractal Estocástica com Variação da População (BFEVP). Neste caso, a variação do tamanho da população se dá de acordo com a equação:

$$N_{p,j} = N_{p,j-1} - 2 \quad (3.22)$$

onde $j = (0, 5N_{ger}), (0, 5N_{ger} + 1), \dots, N_{ger}$, sendo N_{ger} o número de gerações definido pelo usuário. Nesta abordagem, o tamanho da população é reduzida de forma suave ao longo das gerações, isto é; sem a eliminação de muitos candidatos de uma única vez, o que, na prática, pode prejudicar a diversidade do processo. Cabe ressaltar que, quando esse operador está ativo (a partir da metade do número de gerações consideradas), sempre são eliminados os dois piores indivíduos da população. Além disso, o tamanho da população é reduzido até um valor mínimo limite (N_p^{inf}), de forma também a preservar a diversidade do processo evolutivo.

3.4 Estratégia para Determinar Todas as Soluções em Problemas de Otimização

O algoritmo de BFE, assim como todas as outras estratégias de otimização foram idealizadas para encontrar, a cada execução, uma única solução. Para problemas que apresentam uma combinação de variáveis de projeto que levam a um mesmo valor de função objetivo, tanto o algoritmo de BFE quanto outras estratégias não são capazes de obter todas as soluções em uma única execução. Na prática, esta tarefa pode ser realizada executando-se o algoritmo em análise uma série de vezes de forma que, a cada execução, a(s) solução(ões) previamente encontrada(s) seja(m) penalizada(s).

Neste contexto, a presente seção tem por objetivo apresentar uma proposta para automatizar a busca por todas as soluções neste tipo de problema usando o algoritmo de BFE. Para essa finalidade, após cada execução do referido algoritmo, a solução encontrada é penalizada considerando a seguinte relação:

$$\text{Lim}_{inf}(j) = x_{ótimo}(j) - \Delta_{par} x_{ótimo}(j) \quad (3.23a)$$

$$\text{Lim}_{sup}(j) = x_{ótimo}(j) + \Delta_{par} x_{ótimo}(j) \quad (3.23b)$$

em que $j = 1, 2, \dots, N$, sendo N a dimensão do problema, $x_{ótimo}$ é a solução já encontrada e que está sendo restringida e Δ_{par} é um parâmetro definido pelo usuário para a criação de uma barreira em torno de $x_{ótimo}$.

Em resumo, após a obtenção da k -ésima solução do problema ser encontrada, são geradas k -ésimas barreiras de modo a impedir que o algoritmo encontre as k -ésimas soluções já obtidas. Ressalta-se que, na primeira execução do algoritmo de BFE, não existe nenhuma penalização deste tipo, visto que ainda não foi encontrada uma das soluções ótimas. Com relação ao número de execuções, pode-se, a priori, considerar o número de soluções para esse parâmetro. Caso não seja conhecido, pode-se, facilmente, realizar uma análise de sensibilidade para determinar o número de soluções.

O próximo capítulo apresenta a análise de sensibilidade dos parâmetros do algoritmo de BFE em problemas puramente matemáticos com e sem restrição.

4 Estudo da Sensibilidade do Algoritmo BFE

Com o intuito de avaliar a influência de cada um dos parâmetros do algoritmo da BFE, este capítulo apresenta o estudo da sensibilidade. Para esta finalidade são consideradas cinco funções matemáticas com diferentes níveis de complexidade. Estas são apresentadas a seguir.

- Problema $G_1(x)$ (COUTO *et al.*, 2015):

$$\begin{aligned} \min G_1(x) = & -\exp(-(x_1 - 4)^2 - (x_2 - 4)^2) - \exp(-(x_1 + 4)^2 - (x_2 - 4)^2) + \\ & -2\exp(-x_1^2 - (x_2 + 4)^2) - 2\exp(-x_1^2 - x_2^2) \end{aligned} \quad (4.1)$$

Este estudo de caso possui duas variáveis de busca, com dois mínimos locais em $x = (-4, 4)$ e $x = (4, 4)$ em que $G_1(x) = -1$ e dois mínimos globais em $x_{\text{ótimo}} = (0, -4)$ e $x_{\text{ótimo}} = (0, 0)$ com $G_1(x_{\text{ótimo}}) = -2$. Como domínio considera-se $0 \leq x_1, x_2 \leq 10$.

- Problema $G_2(x)$ (COUTO *et al.*, 2015):

$$\min G_2(x) = \sum_{i=1}^{10} h(x)^2 \quad (4.2)$$

onde:

$$h(x) = \exp(-0,1(i+1)x_1) - \exp(-0,1(i+1)x_2) - (\exp(-0,1(i+1)) - \exp(-(i+1))x_3) \quad (4.3)$$

O problema possui três variáveis de busca e apresenta as restrições laterais: $0,9 \leq x_1, x_3 \leq 1,2$ e $9 \leq x_2 \leq 11,2$. Este estudo de caso possui solução dada como $x_{\text{ótimo}} = (1, 10, 1)$ e que corresponde à $G_2(x_{\text{ótimo}}) = 0$.

- Problema $G_3(x)$ (COUTO *et al.*, 2015):

$$\min G_3(x) = \sum_{i=1}^n |x_i \sin(x_i) + 0,1x_i| \quad (4.4)$$

Este estudo de caso apresenta n variáveis de busca. Para fins de aplicação, considerou-se $n = 10$, em que a solução ótima é dada como $x_{\text{ótimo}} = (0, 0, \dots, 0)$ e que corresponde à $G_3(x_{\text{ótimo}}) = 0$. Os limites laterais para as variáveis de busca foram definidos como $-10 \leq x_i \leq 10$, $i = 1, 2, \dots, 10$.

- Problema $G_4(x)$ (GARCIA *et al.*, 2018):

$$\min G_4(x) = (x_1 - 10)^3 + (x_2 - 20)^3 \quad (4.5)$$

sujeito às seguintes restrições:

$$g_1(x) \equiv -(x_1 - 5)^2 - (x_2 - 5)^2 + 100 \leq 0 \quad (4.6a)$$

$$g_2(x) \equiv (x_1 - 6)^2 + (x_2 - 5)^2 - 82,81 \leq 0 \quad (4.6b)$$

O problema apresenta duas variáveis de busca e solução ótima dada como $x_{\text{ótimo}} = (14, 09494; 0, 84284)$ e que corresponde à $G_4(x_{\text{ótimo}}) = -6961, 88176$. Os limites laterais para as variáveis foram definidos como $13 \leq x_1 \leq 100$ e $0 \leq x_2 \leq 100$.

- Problema $G_5(x)$ (GARCIA *et al.*, 2018):

$$\min G_5(x) = 5x_1 + 5x_2 + 5x_3 + 5x_4 - 5 \sum_{i=1}^4 x_i^2 - \sum_{i=5}^{13} x_i \quad (4.7)$$

sujeito às seguintes restrições:

$$g_1(x) \equiv 2x_1 + 2x_2 + x_{10} + x_{11} - 10 \leq 0 \quad (4.8a)$$

$$g_2(x) \equiv -8x_1 + x_{10} \leq 0 \quad (4.8b)$$

$$g_3(x) \equiv -2x_4 - x_5 + x_{10} \leq 0 \quad (4.8c)$$

$$g_4(x) \equiv 2x_1 + 2x_3 + x_{10} + x_{12} - 10 \leq 0 \quad (4.8d)$$

$$g_5(x) \equiv -8x_2 + x_{11} \leq 0 \quad (4.8e)$$

$$g_6(x) \equiv -2x_6 - 2x_7 + x_{11} \leq 0 \quad (4.8f)$$

$$g_7(x) \equiv 2x_2 + 2x_3 + x_{11} + x_{12} - 10 \leq 0 \quad (4.8g)$$

$$g_8(x) \equiv -8x_3 + x_{12} \leq 0 \quad (4.8h)$$

$$g_9(x) \equiv -2x_8 - x_9 + x_{12} \leq 0 \quad (4.8i)$$

$$0 \leq x_i \leq 1, \quad i = 1, \dots, 9, 13 \quad (4.8j)$$

$$0 \leq x_i \leq 100, \quad i = 10, 11, 12 \quad (4.8k)$$

Este problema apresenta treze variáveis de busca. O mínimo global para este estudo é dado como $x_{\text{ótimo}} = (1, 1, 1, 1, 1, 1, 1, 1, 1, 3, 3, 1)$ e que corresponde à $G_5(x_{\text{ótimo}}) = -15$.

Para realizar a análise de sensibilidade, foram fixados valores para todos os parâmetros e, para cada análise particular, o parâmetro de interesse foi variado. Os valores fixos considerados são apresentados na Tabela 9.

Tabela 9: Parâmetros do Algoritmo de BEF Considerados nas Funções Matemáticas Analisadas.

Parâmetro	Valor
N_p	10
N_{ger}	100
N_{dif}	1
N_{passo}	1

Cabe ressaltar que, em cada análise, o algoritmo de BFE será executado 20 vezes para fins de obtenção da média e do desvio padrão em relação ao valor da função objetivo. Além disso, enfatiza-se que para inicializar o algoritmo de BFE sempre foram consideradas sementes aleatórias para alimentar o gerador de números aleatórios.

4.1 Influência do Tamanho da População

A Figura 7 apresenta o gráfico do tipo *boxplot* referente à influência do tamanho da população considerando cada uma das funções matemáticas apresentadas anteriormente. Para essa finalidade considera-se populações com 10, 25, 50 e 100 indivíduos.

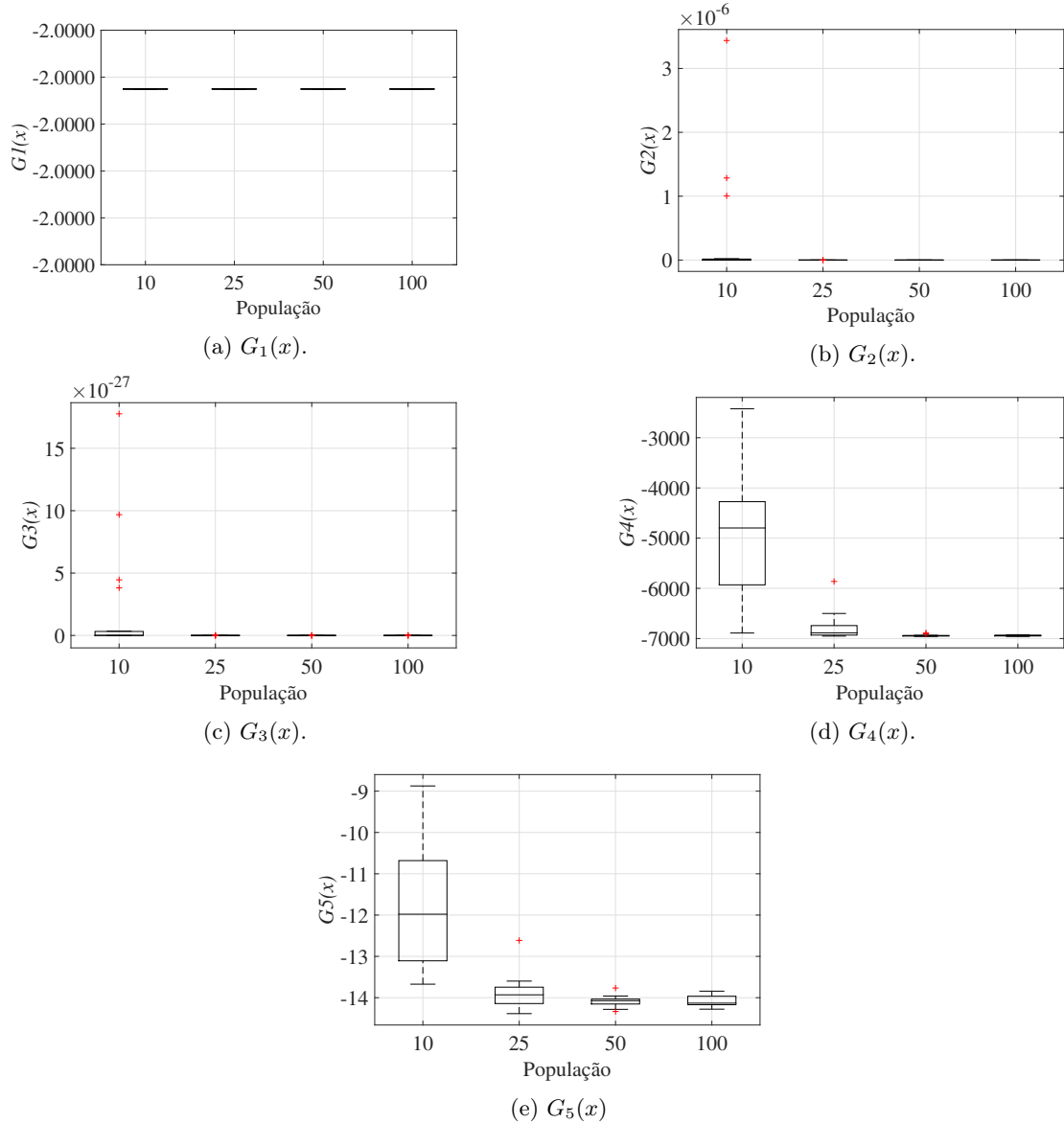


Figura 7: Influência do Tamanho da População no Algoritmo de BFE.

Nesta figura observa-se que, para as três primeiras funções matemáticas, os valores da média e do desvio padrão são muito próximos ao valor de ótimo esperado. Além disso, que para os tamanhos considerados, não existem diferenças significativas. Isto implica que, para as funções $G_1(x)$, $G_2(x)$ e

$G_3(x)$, este parâmetro tem pouca influência no resultado. Já para as funções com restrição ($G_4(x)$ e $G_5(x)$) observa-se que boas estimativas para as soluções ótimas foram obtidas para os maiores valores do parâmetro N_p . Este resultado já é esperado, visto que o aumento do número de indivíduos na população implica naturalmente no incremento da diversidade, o que aumenta a chance da solução ótima ser encontrada.

Já a Tabela 10 apresenta, para cada tamanho de população considerado, os valores médios, os melhores valores e o desvios padrão obtidos para cada uma das cinco funções matemáticas avaliadas.

Tabela 10: Influência do Tamanho da População no Algoritmo de BFE.

	N_p	Média	Desvio	Melhor valor
$G_1(x)$	10	-2,000000	$9,112518 \times 10^{-16}$	-2,000000
	25	-2,000000	$9,112518 \times 10^{-16}$	-2,000000
	50	-2,000000	$9,112518 \times 10^{-16}$	-2,000000
	100	-2,000000	$9,112518 \times 10^{-16}$	-2,000000
$G_2(x)$	10	$8,567296 \times 10^{-8}$	$3,652477 \times 10^{-7}$	$4,884091 \times 10^{-17}$
	25	$1,242535 \times 10^{-13}$	$2,001682 \times 10^{-13}$	$2,705687 \times 10^{-15}$
	50	$4,454540 \times 10^{-14}$	$4,547252 \times 10^{-14}$	$1,546382 \times 10^{-15}$
	100	$2,862747 \times 10^{-14}$	$3,129681 \times 10^{-14}$	$8,980543 \times 10^{-16}$
$G_3(x)$	10	$4,336599 \times 10^{-25}$	$1,922160 \times 10^{-24}$	$3,855199 \times 10^{-35}$
	25	$2,871844 \times 10^{-29}$	$8,269822 \times 10^{-29}$	$3,228627 \times 10^{-36}$
	50	$9,804604 \times 10^{-32}$	$2,344162 \times 10^{-31}$	$1,908492 \times 10^{-34}$
	100	$1,223399 \times 10^{-33}$	$2,705542 \times 10^{-33}$	$7,873217 \times 10^{-38}$
$G_4(x)$	10	-5334,282242	912,475124	-6750,621667
	25	-6702,35736	437,491615	-6954,244607
	50	-6912,670429	41,257946	-6960,495432
	100	-6941,938311	12,094552	-6957,148142
$G_5(x)$	10	-11,896645	1,462086	-14,236565
	25	-13,984435	0,308882	-14,334738
	50	-14,087688	0,149132	-14,275877
	100	-14,074850	0,158129	-14,273816

Como pode ser observado nesta tabela, para as funções irrestritas, independentemente do valor do parâmetro N_p , o algoritmo de BFE sempre convergiu para a solução ótima global. Por outro lado, para os problema restritos, apenas quando parâmetro N_p foi maior ou igual a 50 que o algoritmo de BFE obteve boas aproximações para as soluções ótimas globais. Conforme destacado anteriormente, este resultado já era esperado, visto que, quanto maior a complexidade do estudo de caso, maior a dificuldade encontrada pelo algoritmo de otimização. Todavia, ressalta-se que, quanto maior o valor do parâmetro N_p , maior o custo computacional, computado em termos do número de avaliações da função objetivo que, para os estudos em questão foram iguais à 1010, 2525, 5050 e 10100, respectivamente.

Em resumo, para os estudos de caso avaliados, considerar N_p igual a 50 é uma boa escolha para este parâmetro em termos de convergência e redução de custo computacional.

4.2 Influência do Número de Gerações

A Figura 8 apresenta o gráfico do tipo *boxplot* referente à influência do número de gerações considerando cada uma das funções matemáticas apresentadas anteriormente.

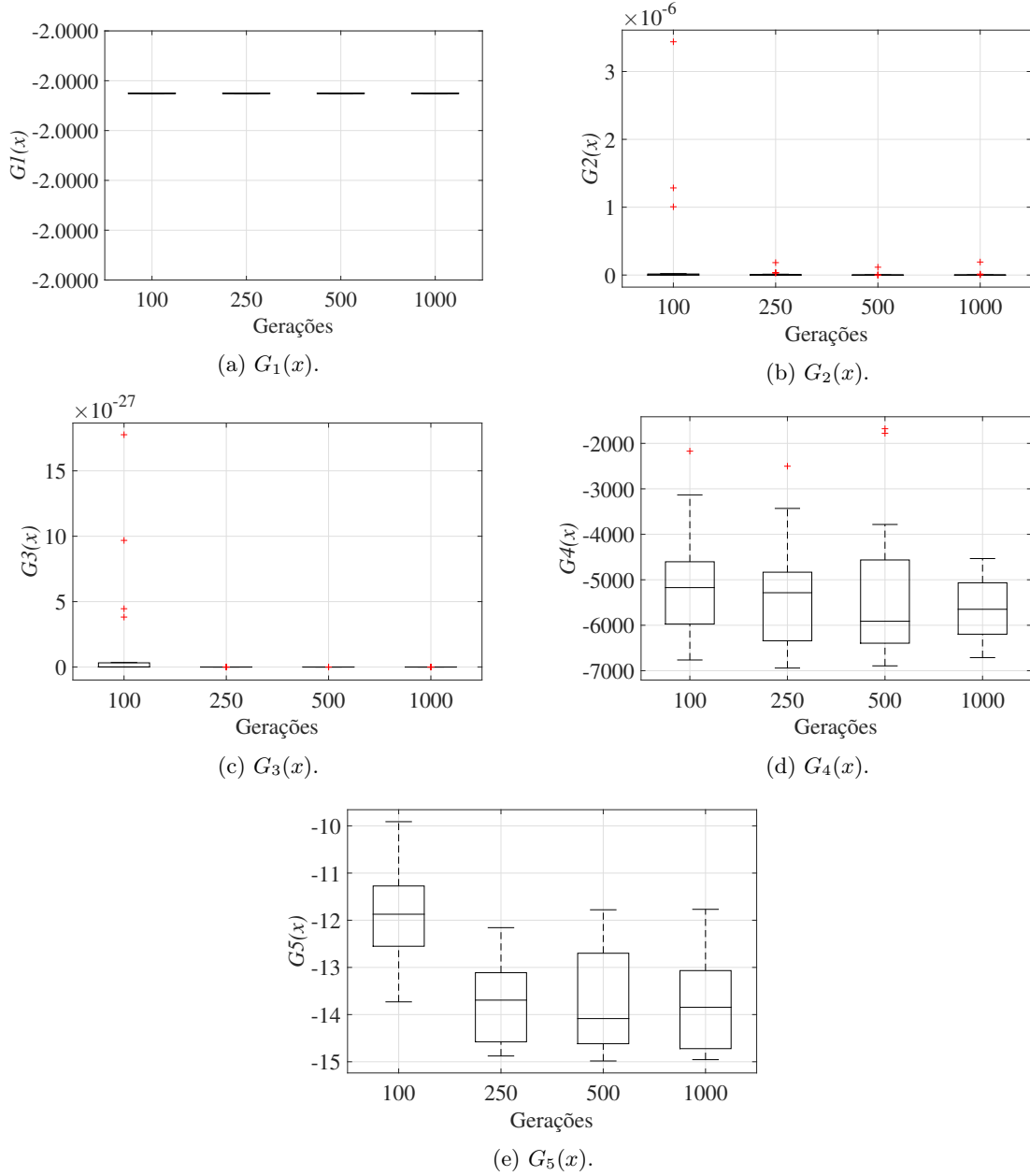


Figura 8: Influência do Número de Gerações no Algoritmo de BFE.

Nesta figura observa-se, para um tamanho fixo de indivíduos na população, que ao aumentar a

complexidade do estudo de caso, exige-se mais gerações para que uma boa aproximação da solução ótima seja encontrada. Para esta análise, como o número máximo de gerações são iguais a 100, 250, 500 e 1000 gerações e o número de indivíduos é igual a 10, o número de avaliações da função objetivo para cada uma das funções é igual a 1010, 2510, 5010 e 10010, respectivamente. O resultado apresentado na Figura 8 está em concordância com o esperado e com o reportado ao se analisar a influência do tamanho da população (ver a Figura 7). Assim, ao se incrementar o valor do parâmetro N_{ger} aumenta-se o número de vezes que os operadores do algoritmo de BFE são utilizados para evoluir a população de candidatos à solução do problema de otimização.

Já na Tabela 11 são apresentados, para cada número máximo de gerações, os valores médios, os melhores valores e o desvio padrão obtidos para cada uma das cinco funções matemáticas avaliadas. Ao se analisar os resultados apresentados nesta tabela conclui-se que o algoritmo de BFE sempre foi robusto para os problemas irrestritos e apresentou uma variação significativa para os problemas restritos, como pode ser constatado pelo valor dos desvios padrão obtidos. Neste contexto, é importante destacar que o algoritmo de BFE não foi capaz de encontrar, precisamente, as soluções globais reportadas para os problemas restritos. Isto se deve, provavelmente, ao tamanho de população associado às execuções realizadas. Assim, espera-se que, ao aumentar o tamanho da população, que o algoritmo seja capaz de encontrar boas estimativas para as soluções globais nestes estudos de caso. Neste cenário, considera-se que os valores de N_{ger} iguais a 500 e 1000 são boas estimativas para se inicializar um novo estudo de caso.

Tabela 11: Influência do Número de Gerações no Algoritmo de BFE.

	N_{ger}	Média	Desvio	Melhor valor
$G_1(x)$	100	-2,000000	$9,112518 \times 10^{-16}$	-2,000000
	250	-2,000000	$9,112518 \times 10^{-16}$	-2,000000
	500	-2,000000	$9,112518 \times 10^{-16}$	-2,000000
	1000	-2,000000	$9,112518 \times 10^{-16}$	-2,000000
$G_2(x)$	100	$8,567296 \times 10^{-8}$	$3,652477 \times 10^{-7}$	$4,884091 \times 10^{-17}$
	250	$1,364402 \times 10^{-9}$	$5,221565 \times 10^{-9}$	$2,210659 \times 10^{-29}$
	500	$8,726431 \times 10^{-9}$	$3,144960 \times 10^{-8}$	0
	1000	$5,446968 \times 10^{-9}$	$2,170583 \times 10^{-8}$	0
$G_3(x)$	100	$4,336599 \times 10^{-25}$	$1,922160 \times 10^{-24}$	$3,855199 \times 10^{-35}$
	250	$7,113477 \times 10^{-72}$	$2,314962 \times 10^{-71}$	$1,642825 \times 10^{-83}$
	500	$1,684770 \times 10^{-140}$	$7,534508 \times 10^{-140}$	$5,092509 \times 10^{-162}$
	1000	$3,758424 \times 10^{-297}$	0	$2,675157 \times 10^{-318}$
$G_4(x)$	100	-5150,429159	1198,286430	-6763,656521
	250	-5350,522306	1186,900688	-6941,516280
	500	-5299,666705	1517,644316	-6893,320340
	1000	-5633,989531	650,185498	-6711,829326
$G_5(x)$	100	-11,899394	0,983349	-13,730246
	250	-13,704259	0,858383	-14,877186
	500	-13,663078	1,099553	-14,983769
	1000	-13,772936	1,050651	-14,955467

4.3 Influência do Número de Difusões

Na Figura 9 são apresentados os gráficos *boxplot* referentes à influência do número de difusões para as cinco funções matemáticas consideradas. Para fins de análise considera-se esse parâmetro assumindo os seguintes valores 1, 5, 10 e 20.

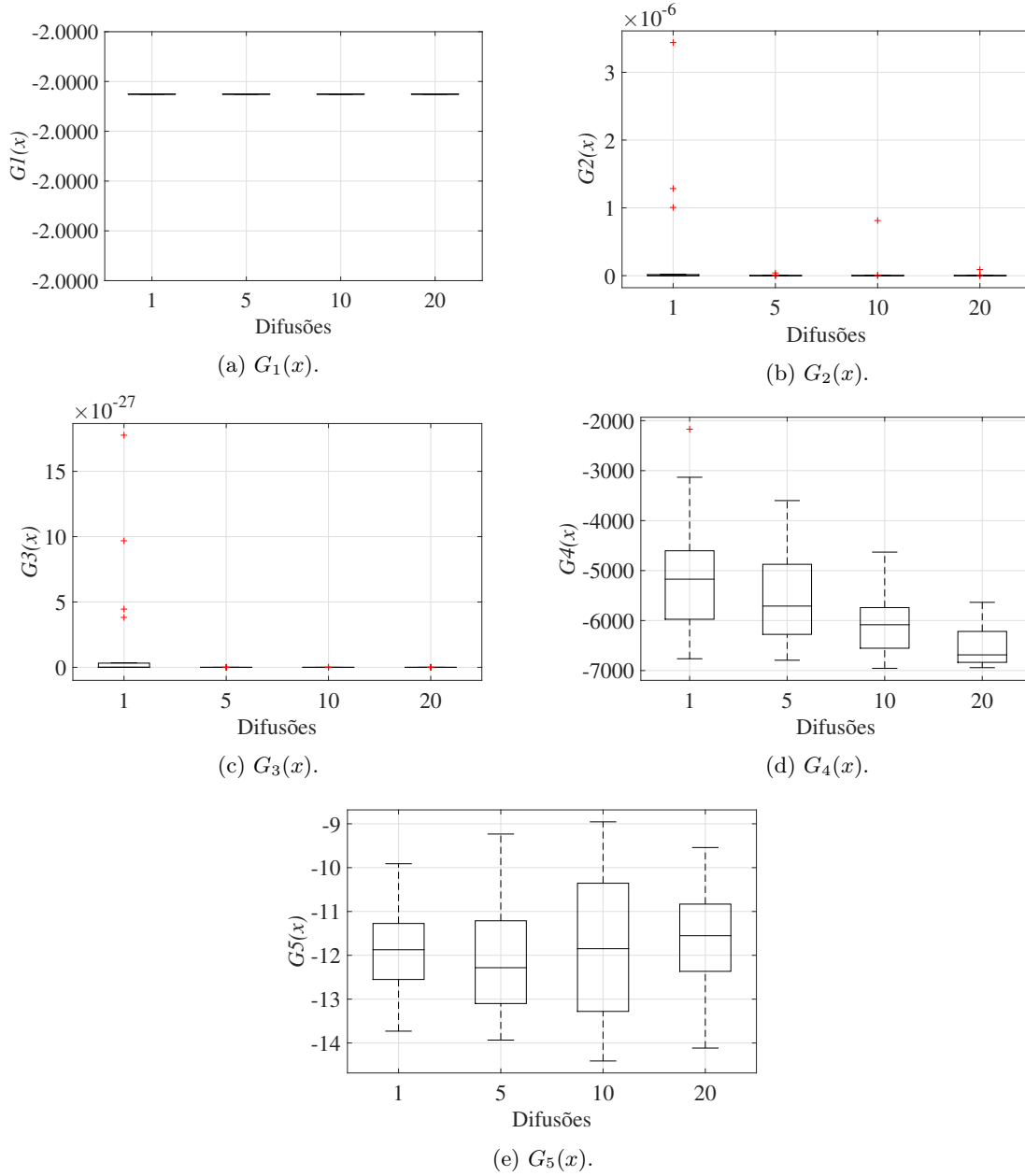


Figura 9: Influência do Número de Difusões no Algoritmo de BFE.

Nesta figura, assim como constatado para os dois primeiros parâmetros analisados, que o algoritmo de BFE sempre converge para a solução ótima para cada uma destas funções, diferentemente dos estudos de caso restritos. Para a função $G_4(x)$ observa-se uma boa aproximação da solução ótima para 20 difusões. Todavia, a função $G_5(x)$ não apresenta esta mesma tendência de redução do valor da função objetivo com o aumento do parâmetro N_{dif} , isto é; não se pode concluir, para este caso, que o aumento no número de difusões resulta na melhora do resultado obtido. Em termos do custo computacional, como ambos o número de indivíduos da população e o número de gerações são fixos (ver a Tabela 9), o número de avaliações da função objetivo em todas as avaliações é igual a 1010.

Na Tabela 12 são apresentados os valores médios, os melhores valores e os desvios padrão obtidos para cada uma das cinco funções matemáticas avaliadas considerando a variação do N_{dif} . Ao se analisar estes resultados observa-se a boa aproximação obtida pelo algoritmo de BFE para os problemas de otimização irrestritos. Isto demonstra a pouca sensibilidade dos resultados com a variação deste parâmetro, pelo menos para os problemas irrestritos analisados. Por outro lado, para os problemas restritos, constata-se que apenas para os maiores valores de N_{dif} (10 e 20) foi possível obter uma boa aproximação para a função $G_4(x)$ e uma razoável aproximação para a função $G_5(x)$.

Tabela 12: Influência do Número de Difusões no Algoritmo de BFE.

	N_{dif}	Média	Desvio	Melhor valor
$G_1(x)$	1	-2,000000	$9,112518 \times 10^{-16}$	-2,000000
	5	-2,000000	$9,112518 \times 10^{-16}$	-2,000000
	10	-2,000000	$9,112518 \times 10^{-16}$	-2,000000
	20	-2,000000	$9,112518 \times 10^{-16}$	-2,000000
$G_2(x)$	1	$8,567296 \times 10^{-8}$	$3,652477 \times 10^{-7}$	$4,884091 \times 10^{-17}$
	5	$1,402461 \times 10^{-7}$	$4,492830 \times 10^{-7}$	$6,616763 \times 10^{-16}$
	10	$3,857652 \times 10^{-9}$	$1,709517 \times 10^{-8}$	$2,617087 \times 10^{-16}$
	20	$5,316260 \times 10^{-9}$	$2,352962 \times 10^{-8}$	$6,943982 \times 10^{-16}$
$G_3(x)$	1	$4,336599 \times 10^{-25}$	$1,922160 \times 10^{-24}$	$3,855199 \times 10^{-35}$
	5	$4,466230 \times 10^{-65}$	$1,880563 \times 10^{-64}$	$3,740121 \times 10^{-76}$
	10	$3,995874 \times 10^{-86}$	$1,702908 \times 10^{-85}$	$1,583497 \times 10^{-96}$
	20	$4,745395 \times 10^{-111}$	$2,122097 \times 10^{-110}$	$3,414962 \times 10^{-130}$
$G_4(x)$	1	-5150,429159	1198,286430	-6763,656521
	5	-5514,287119	968,789670	-6793,605891
	10	-5986,325995	704,138557	-6956,453532
	20	-6531,425102	368,009812	-6942,072681
$G_5(x)$	1	-11,899394	0,983349	-13,730246
	5	-12,127062	1,255714	-13,934198
	10	-11,704604	1,727591	-14,410961
	20	-11,665878	1,174286	-14,115807

4.4 Influência do Passo

Na Figura 10 são apresentados os gráficos *boxplot* referentes à influência do passo para as cinco funções matemáticas consideradas. Para realizar esta análise de sensibilidade considera-se esse parâmetro assumindo os seguintes valores 0,25; 0,5; 0,75 e 1.

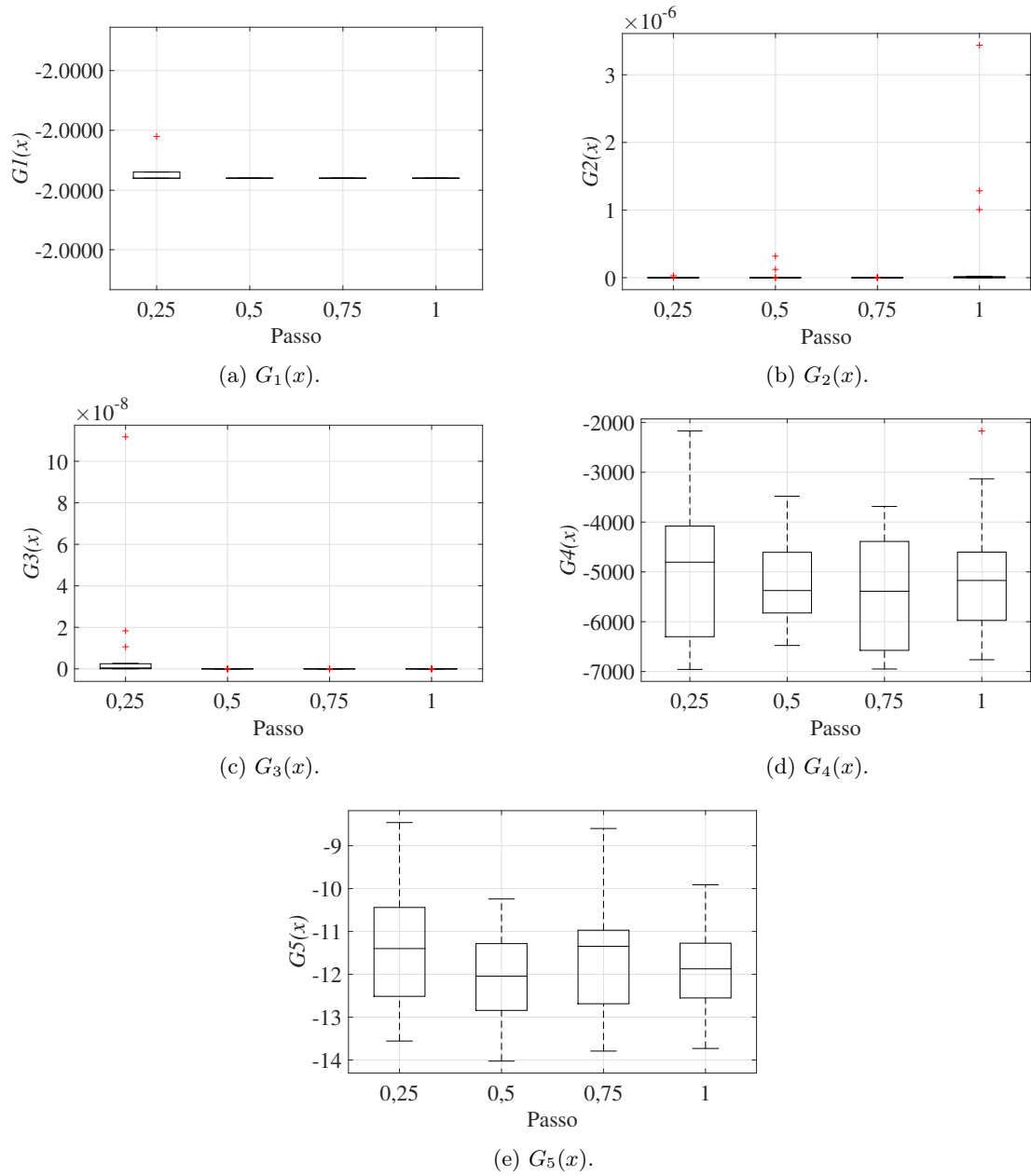


Figura 10: Influência da Escolha do Passo no Algoritmo de BFE.

Assim como observado nas análises de sensibilidade anteriores, percebe-se na Figura 10 que o algoritmo de BEF sempre foi capaz de convergir para as soluções ótimas em cada um dos problemas irrestritos, independentemente do valor considerado para o parâmetro N_{passo} (ver as Figuras 10a, 10b e 10c). Por outro lado, para o mesmo conjunto de parâmetros considerados, o algoritmo apresentou mais dificuldades para encontrar boas aproximações para as soluções ótimas globais nos problemas restritos, isto é; nem sempre a referida estratégia de otimização conseguiu convergir para a melhor

solução, o que pode ser constatado pelos valores dos desvios padrão apresentados na Tabela 13. Neste caso, para as funções $G_4(x)$ e $G_5(x)$, bons valores para o parâmetro N_{passo} são 0,75 e 0,5, respectivamente. Finalmente, em termos do custo computacional, como ambos os parâmetros (N_p e N_{ger}) são fixos, conforme a Tabela 9, o número de avaliações da função objetivo em todas as avaliações é igual a 1010.

Tabela 13: Influência do Passo no Algoritmo de BFE.

	N_{passo}	Média	Desvio	Melhor valor
$G_1(x)$	0,25	-2,000000	$2,090720 \times 10^{-13}$	-2,000000
	0,5	-2,000000	$9,112518 \times 10^{-16}$	-2,000000
	0,75	-2,000000	$9,112518 \times 10^{-16}$	-2,000000
	1	-2,000000	$9,112518 \times 10^{-16}$	-2,000000
$G_2(x)$	0,25	$2,398248 \times 10^{-7}$	$8,303714 \times 10^{-7}$	$2,426719 \times 10^{-19}$
	0,5	$8,301338 \times 10^{-8}$	$3,712292 \times 10^{-7}$	$9,419930 \times 10^{-17}$
	0,75	$5,108160 \times 10^{-9}$	$1,692110 \times 10^{-8}$	$5,049869 \times 10^{-17}$
	1	$8,567296 \times 10^{-8}$	$3,652477 \times 10^{-7}$	$4,884091 \times 10^{-17}$
$G_3(x)$	0,25	$2,039162 \times 10^{-7}$	$8,908791 \times 10^{-7}$	$1,892285 \times 10^{-13}$
	0,5	$8,240270 \times 10^{-15}$	$2,298407 \times 10^{-14}$	$6,249754 \times 10^{-20}$
	0,75	$3,180086 \times 10^{-19}$	$1,342248 \times 10^{-18}$	$4,971993 \times 10^{-26}$
	1	$4,336599 \times 10^{-25}$	$1,922160 \times 10^{-24}$	$3,855199 \times 10^{-35}$
$G_4(x)$	0,25	-5065,128207	1316,207880	-6958,758038
	0,5	-5229,979667	767,291731	-6477,334030
	0,75	-5389,905548	1106,351292	-6950,171140
	1	-5150,429159	1198,286430	-6763,656521
$G_5(x)$	0,25	-11,430686	1,403593	-13,556628
	0,5	-12,011910	1,044439	-14,025902
	0,75	-11,630985	1,317593	-13,790939
	1	-11,899394	0,983349	-13,730246

4.5 Consolidado dos Resultados

De forma geral, observa-se que a influência do aumento da população na melhora dos resultados é expressiva, uma vez que proporciona uma maior variabilidade de indivíduos e assim, uma maior chance de obtenção do resultado ótimo. O mesmo pode ser constatado para o aumento no número de gerações, isto é; quanto maior o valor deste parâmetro, maior é a chance da solução ótima ser encontrada, visto que mais vezes os operadores do algoritmo de BEF são aplicados aos indivíduos da população. Todavia, ressalta-se que o aumento em qualquer um destes dois parâmetros resulta no aumento do número de avaliações da função objetivo. Assim, um bom balanceamento entre estes dois parâmetros é de fundamental importância para o atendimento do binômio custo computacional versus convergência. O aumento no número de difusões provoca uma melhora nos resultados, mesmo que para alguns casos não se tenha mantido uma alta variabilidade dos resultados. Já para a análise do passo, pode-se concluir que, para problemas mais simples, quanto maior o passo melhor os resultados. Todavia, para os problemas mais complexos avaliados neste capítulo, não se encontrou um valor *default* mais interessante para este parâmetro.

Em resumo, o algoritmo de BFE foi sempre capaz de convergir para a solução ótima global em problemas irrestritos. Já para os problemas restritos, como esperado, o algoritmo apresentou

dificuldades de obter uma boa aproximação para a solução ótima, pelo menos para o conjunto de parâmetros considerados neste capítulo. Assim, baseando-se nos resultados obtidos para os estudos de caso abordados pode-se sugerir os seguintes parâmetros: população com 50 indivíduos, 500 gerações, número de difusões iguais a 10 e, apesar da análise do passo não ser conclusiva, um passo da ordem de 0,75 é uma boa escolha inicial. Cabe enfatizar que, embora tenham sido sugeridos valores *default* para uma nova aplicação, que estes podem ser diferentes para cada novo estudo de caso. Assim, na prática, é importante, para cada aplicação, realizar a análise de sensibilidade de forma a encontrar a melhor configuração para um dado estudo de caso.

No próximo capítulo são apresentados os resultados obtidos com a avaliação do algoritmo de BFE, bem com dos operadores propostos em estudos de caso matemáticos e de engenharia com diferentes níveis de complexidade.

5 Resultados e Discussões

No capítulo anterior o algoritmo de BFE foi aplicado a uma série de funções matemáticas com diferentes níveis de complexidade para avaliar a sensibilidade dos parâmetros da referida estratégia de otimização. Neste capítulo são apresentados os resultados obtidos com a aplicação do algoritmo de BFE em problemas distintos da ciência. Para essa finalidade, os resultados obtidos são comparados com a solução analítica conhecida e com resultados obtidos por outras estratégias, bem como com o número de avaliações da função objetivo para os estudos de caso de projeto de sistemas de engenharia.

5.1 Problemas Matemáticos Mono-objetivos

Nesta seção são reportados os resultados obtidos pelo algoritmo de BFE à problemas de otimização mono-objetivos puramente matemáticos irrestritos e com diferentes números de variáveis de busca. Para essa finalidade, em todas estas aplicações os seguintes parâmetros foram considerados: número de gerações (100), tamanho da população (20), número de difusões (3) e passo (1). Com estes parâmetros, o número de avaliações da função objetivo é igual a 2020. Finalmente, ressalta-se que, para cada aplicação desta seção, foram consideradas 20 execuções do algoritmo de BFE (em que não foram definidas sementes iniciais específicas para a inicialização do gerador de números aleatórios requeridos pelo algoritmo).

5.1.1 Função de Beale

A função de Beale apresenta duas variáveis de busca definidas no intervalo $-4,5 \leq x_1, x_2 \leq 4,5$. O valor ótimo para este estudo é dado como $x_{\text{ótimo}} = (3; 0,5)$ o que corresponde à $f(x_{\text{ótimo}}) = 0$. Matematicamente, esta função é dada pela relação (SURJANOVIC & BINGHAM, 2013):

$$\min f(x) = (1,5 - x_1 + x_1x_2)^2 + (2,25 - x_1 + x_1x_2^2)^2 + (2,625 - x_1 + x_1x_2^3)^2 \quad (5.1)$$

A Tabela 14 apresenta os resultados obtidos (média, desvio padrão e melhor solução) com a aplicação do algoritmo de BFE. Nesta tabela observa-se que os resultados obtidos para o vetor de variáveis de busca e para a função objetivos foram sempre iguais à solução ótima reportada pela literatura, independentemente da execução do algoritmo. O desvio padrão igual a zero demonstra que a estratégia utilizada sempre foi robusta, isto é; sempre foi capaz de obter a solução ótima.

Tabela 14: Resultados Obtidos pelo Algoritmo de BFE para a Função de Beale.

	Média	Desvio Padrão	Melhor Solução
x_1	3,000000	0	3,000000
x_2	0,500000	0	0,500000
$f(x_{\text{ótimo}})$	0	0	0

5.1.2 Função de Rastrigin

A função de Rastrigin é definida pela seguinte relação (SURJANOVIC & BINGHAM, 2013):

$$\min f(x) = 10n + \sum_{i=1}^n [x_i^2 - 10 \cos(2\pi x_i)] \quad (5.2)$$

onde n é o número de variáveis de busca e cujos limites laterais são definidos como $-5,12 \leq x_i \leq 5,12$, $i = 1, \dots, n$.

Para fins de aplicação considerou-se n igual a 2, o que implica que a solução ótima é dada como $x_{\text{ótimo}} = (0; 0)$ o que corresponde à $f(x_{\text{ótimo}}) = 0$. A Tabela 15 apresenta os resultados obtidos (média, desvio padrão e melhor solução) com a aplicação do algoritmo de BFE. Nesta tabela observa-se, assim como para o primeiro estudo de caso, que os resultados obtidos para o vetor de variáveis de busca e para a função objetivos sempre convergiram para a solução ótima, demonstrando que a estratégia utilizada foi robusta, conforme pode ser observado pelos valores de média e desvio padrão.

Tabela 15: Resultados Obtidos pelo Algoritmo de BFE para a Função de Rastrigin.

	Média	Desvio Padrão	Melhor Solução
x_1	$0,174841 \times 10^{-9}$	$0,154942 \times 10^{-8}$	$0,474540 \times 10^{-9}$
x_2	$-0,091041 \times 10^{-9}$	$0,181622 \times 10^{-8}$	$-0,028555 \times 10^{-9}$
$f(x_{\text{ótimo}})$	0	0	0

5.1.3 Função de Michalewicz

Seja a função de Michalewicz representada por (SURJANOVIC & BINGHAM, 2013):

$$\min f(x) = - \sum_{i=1}^n \sin(x_i) \left(\sin \left(\frac{ix_i^2}{\pi} \right) \right)^{2n} \quad (5.3)$$

Para esse estudo de caso considerou-se o número de variáveis de busca (n) igual a 10 e limites laterais definidos como $0 \leq x_i \leq \pi$, $i = 1, 2, \dots, 10$. O valor ótimo da referida função é $f(x_{\text{ótimo}}) = -9,66015$ com o vetor sendo dado como $x_{\text{ótimo}} = (2,2029; 1,5708; 1,2850; 1,9231; 1,7205; 1,5708; 1,4544; 1,7561; 1,6557; 1,5708)$.

A Tabela 16 apresenta os resultados obtidos (média, desvio padrão e melhor solução) com a aplicação do algoritmo de BFE.

Tabela 16: Resultados Obtidos pelo Algoritmo de BFE para a Função de Michalewicz.

	Média	Desvio Padrão	Melhor Solução
x_1	2,202909	$1,614301 \times 10^{-5}$	2,202906
x_2	1,570796	$0,2870126 \times 10^{-5}$	1,570796
x_3	1,284995	$1,471258 \times 10^{-5}$	1,284992
x_4	1,801668	0,296477	1,923058
x_5	1,720470	$0,101688 \times 10^{-5}$	1,720470
x_6	1,570796	$0,080718 \times 10^{-5}$	1,570796
x_7	1,454415	$0,358221 \times 10^{-5}$	1,454414
x_8	1,696758	0,144896	1,756086
x_9	1,655718	$0,135550 \times 10^{-5}$	1,655717
x_{10}	1,585180	0,064329	1,570796
$f(x_{\text{ótimo}})$	-9,651095	0,017418	-9,660152

Nesta tabela é possível observar que os valores referentes a média para cada uma das variáveis de busca encontram-se, em sua maioria, próximos aos resultados de ótimo reportado por Surjanovic

& Bingham (2013), exceto para as variáveis x_4 , x_8 e x_{10} , que são as que possuem maiores valores de desvio padrão. Apesar disso, conclui-se que o algoritmo de BFE foi capaz de encontrar uma boa aproximação para o problema apresentado, visto o melhor valor descrito nesta tabela.

5.1.4 Função de Levy

A função de Levy é definida como (SURJANOVIC & BINGHAM, 2013):

$$\min f(x) = \sin^2(\pi z_1) + \sum_{i=1}^{n-1} [(z_i - 1)^2(1 + 10 \sin^2(\pi z_i + 1))] + (z_n - 1)^2(1 + \sin^2(2\pi z_n)) \quad (5.4)$$

onde $z_i = 1 + \frac{x_i - 1}{4}$ e $-10 \leq x_i \leq 10$, $i = 1, \dots, n$ (n é o número de variáveis de projeto, que para esta aplicação foi igual a 30). A solução ótima é dada como $x_{\text{ótimo}} = (1, \dots, 1)$ o que corresponde à $f(x_{\text{ótimo}}) = 0$.

A Tabela 17 apresenta os resultados obtidos (média, desvio padrão e melhor solução) com a aplicação do algoritmo de BFE. Nesta tabela considera-se que, apesar do alto número de variáveis de busca, o algoritmo de BFE foi capaz de obter boas estimativas para o vetor de variáveis de busca, conforme observado para os valores de média e de desvio padrão.

5.2 Problemas de Projeto Mono-objetivos

A presente seção tem por objetivo apresentar a aplicação do algoritmo de BFE em problemas de engenharia. Estes apresentam como característica básica a presença de restrições advindas de limitações físicas, ambientais e de projeto. Para o tratamento das restrições, o Método da Penalidade Estática considerando um fator de penalidade r_p da ordem de 10^{10} foi utilizado.

5.2.1 Projeto de uma Mola de Tensão/Compressão

Qualquer elemento mecânico que possui uma elasticidade e que responde elasticamente a uma solicitação pode ser considerado uma mola. As molas podem ter diferentes formatos, sendo mais comuns na engenharia as helicoidais, planas, prato e as de tensão constante (SANTOS JUNIOR, 2001).

O problema em análise foi estudado por Coello & Montes (2002), He & Wang (2007) e Mirjalili & Lewis (2016) e consiste na minimização do volume de uma mola de tensão/compressão, helicoidal, sujeita às restrições de deflexão mínima, tensão de cisalhamento, frequência de surto e limites no diâmetro externo, conforme ilustrado na Figura 11.

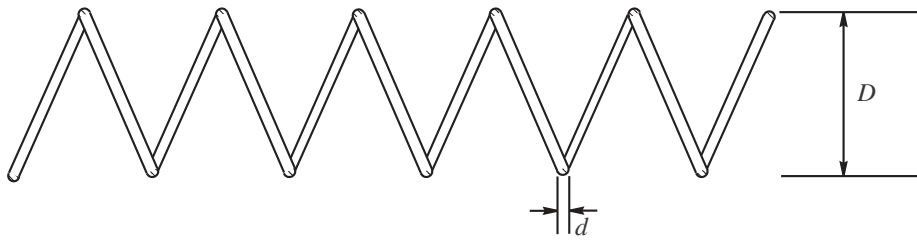


Figura 11: Projeto de uma Mola Tensão/Compressão.

Tabela 17: Resultados Obtidos pelo Algoritmo de BFE para a Função de Levy.

	Média	Desvio Padrão	Melhor Solução
x_1	0,999999	$0,499755 \times 10^{-5}$	0,999999
x_2	0,999999	$0,625755 \times 10^{-5}$	0,999999
x_3	0,999999	$0,560915 \times 10^{-5}$	1,000002
x_4	0,999999	$0,689152 \times 10^{-5}$	1,000001
x_5	1,000000	$0,811188 \times 10^{-5}$	0,999993
x_6	1,000001	$0,729748 \times 10^{-5}$	1,000001
x_7	1,000000	$0,726307 \times 10^{-5}$	0,999996
x_8	1,000000	$0,920245 \times 10^{-5}$	1,000000
x_9	1,000000	$0,705201 \times 10^{-5}$	0,999998
x_{10}	1,000000	$0,658969 \times 10^{-5}$	1,000006
x_{11}	0,999999	$0,641823 \times 10^{-5}$	1,000003
x_{12}	1,000000	$0,662819 \times 10^{-5}$	0,999999
x_{13}	1,000000	$0,723150 \times 10^{-5}$	1,000002
x_{14}	1,000001	$0,411797 \times 10^{-5}$	0,999999
x_{15}	0,999999	$0,703932 \times 10^{-5}$	1,000003
x_{16}	1,000000	$0,500332 \times 10^{-5}$	1,000000
x_{17}	1,000002	$0,734440 \times 10^{-5}$	0,999998
x_{18}	0,999999	$0,548983 \times 10^{-5}$	0,999999
x_{19}	0,999998	$0,877372 \times 10^{-5}$	0,999999
x_{20}	0,999999	$0,727973 \times 10^{-5}$	0,999997
x_{21}	1,000002	$0,564256 \times 10^{-5}$	0,999999
x_{22}	1,000002	$0,433933 \times 10^{-5}$	0,999998
x_{23}	1,000001	$0,476705 \times 10^{-5}$	0,999998
x_{24}	1,000000	$0,702723 \times 10^{-5}$	0,999999
x_{25}	1,000001	$0,579377 \times 10^{-5}$	1,000002
x_{26}	0,999996	$0,654154 \times 10^{-5}$	0,999999
x_{27}	0,999999	$0,707942 \times 10^{-5}$	0,999999
x_{28}	0,999998	$0,749572 \times 10^{-5}$	1,000001
x_{29}	0,999999	$0,656913 \times 10^{-5}$	0,999998
x_{30}	1,000000	$0,234147 \times 10^{-5}$	1,000001
$f(x_{\text{ótimo}})$	$7,133238 \times 10^{-10}$	$6,686959 \times 10^{-10}$	$8,933112 \times 10^{-11}$

As variáveis de projeto são para esta aplicação são: diâmetro do fio (d), diâmetro médio da espiral (D) e número de espirais (N_{esp}). D e d são representados em polegadas (in), N_{esp} possui um valor adimensional (ARORA, 1989).

Considerando a seguinte notação ($\vec{x} = [x_1 \ x_2 \ x_3] = [d \ D \ N_{esp}]$), matematicamente o problema pode ser definido como (MIRJALILI & LEWIS, 2016):

$$\min f(x) = (x_3 + 2)x_2x_1^2 \quad (5.5)$$

sujeito às seguintes restrições:

$$g_1(x) \equiv 1 - \frac{x_2^3x_3}{71785x_1^4} \leq 0 \quad (5.6a)$$

$$g_2(x) \equiv \frac{4x_2^2 - x_1x_2}{12566(x_2x_1^3 - x_1^4)} + \frac{1}{5108x_1^2} - 1 \leq 0 \quad (5.6b)$$

$$g_3(x) \equiv 1 - \frac{140,45x_1}{x_2^2x_3} \leq 0 \quad (5.6c)$$

$$g_4(x) \equiv \frac{x_2 + x_1}{1,5} - 1 \leq 0 \quad (5.6d)$$

$$0,05 \leq x_1 \leq 2,00; \quad 0,25 \leq x_2 \leq 1,30; \quad 2,00 \leq x_3 \leq 15,00 \quad (5.6e)$$

Na Tabela 18 são apresentados os resultados obtidos (média, desvio padrão e melhor solução) aplicando o algoritmo de BFE considerando os seguintes parâmetros: número de gerações (45), tamanho da população (20), número de difusões (3) e passo (1), bem como 20 execuções independentes da referida estratégia de otimização. Nesta tabela observa-se que o algoritmo de BFE foi capaz de convergir para um mesmo valor de função objetivo, conforme pode ser visto pelos valores de média e desvio padrão. Cabe ressaltar que, apesar da variável x_3 representar o número de espirais, a mesma foi tratada como sendo uma variável real ao invés de uma inteira, conforme tem sido considerado na literatura especializada (COELLO & MONTES, 2002; HE & WANG, 2007; MIRJALILI & LEWIS, 2016).

Tabela 18: Resultados Obtidos pelo Algoritmo de BFE para o Problema do Projeto de uma Mola.

	Média	Desvio Padrão	Melhor Solução
x_1 (in)	0,051686	0,000018	0,051694
x_2 (in)	0,356657	0,000431	0,356838
x_3 (-)	11,292556	0,025260	11,281889
f (in ³)	0,012665	$1,914767 \times 10^{-8}$	0,012665

Na Tabela 19 são apresentados os resultados obtidos por diferentes autores considerando outros algoritmos heurísticos. Coello & Montes (2002) resolveram este mesmo problema usando um método derivado dos Algoritmos Genéticos e que apresenta uma estratégia de nichos baseado em Pareto. Já He & Wang (2007) e Mirjalili & Lewis (2016) empregaram o Método de Enxame de Partículas e o Algoritmo de Otimização de Baleias para a resolução do problema de projeto de uma mola. Nesta tabela observa-se que os resultados obtidos pelo algoritmo de BFE são coerentes com os reportados considerando outras abordagens heurísticas. Todavia, há que se destacar que a metodologia proposta obteve a solução com um menor custo computacional (menor número de avaliações da função objetivo) quando comparado com as outras abordagens.

Tabela 19: Resultados Obtidos para o Problema do Projeto de uma Mola Considerando Diferentes Abordagens (N_{eval} é o Número de Avaliações da Função Objetivo).

	BFE	Coello & Montes (2002)	Mirjalili & Lewis (2016)	He & Wang (2007)
x_1 (in)	0,051694	0,051989	0,051207	0,051728
x_2 (in)	0,356838	0,363965	0,345215	0,357644
x_3 (-)	11,281889	10,890522	12,004032	11,244543
f (in ³)	0,012665	0,012681	0,012676	0,012675
N_{eval}	920	80000	4410	5460

Assim, observa-se uma redução de, aproximadamente, 98,85%, 77,78% e 83,16% avaliações da função objetivo com relação aos trabalhos de Coello & Montes (2002), Mirjalili & Lewis (2016) e He & Wang (2007), respectivamente.

5.2.2 Projeto de uma Viga Soldada

De forma geral, as vigas podem ser descritas como elementos lineares em que a flexão é preponderante (NBR 6118: 2003, item 14.4.1.1), sendo estruturas utilizadas de forma ampla na engenharia. Como aplicação prática pode-se citar o seu uso em edifícios, nos quais as vigas são empregadas como apoio para lajes e paredes, de forma a conduzir os esforços para os pilares. As vigas podem ser engastadas/em balanço ou apoiadas, e ainda, apresentarem modelos híbridos, como as vigas apoiadas com balanço (PINHEIRO, 2007).

Para fins de aplicação, considere uma viga soldada em balanço e que deve ser projetada a partir da determinação das dimensões (h, l, t, b) via minimização do seu custo $(f(x))$, conforme a Figura 12 (COELLO & MONTES, 2002).

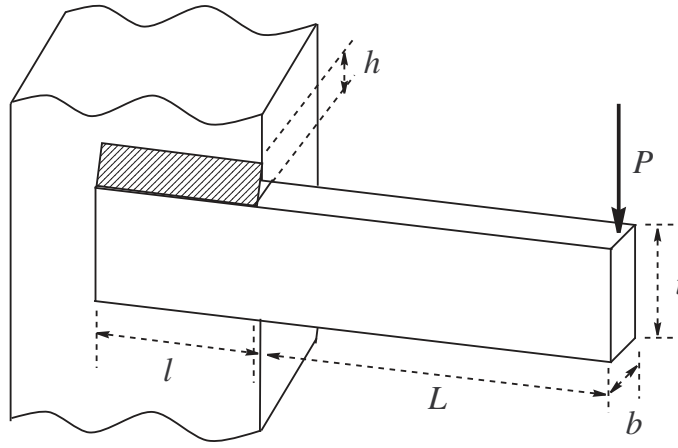


Figura 12: Projeto de uma Viga Soldada.

Matematicamente, considerando a seguinte notação $(\vec{x} = [x_1 \ x_2 \ x_3 \ x_4] = [h \ l \ t \ b])$ (in), o problema pode ser definido como (COELLO & MONTES, 2002):

$$\min f(x) = 1,10471x_1^2x_2 + 0,04811x_3x_4(14,0 + x_2) \quad (5.7)$$

Fisicamente, este sistema está sujeito à limitações de tensão de cisalhamento (τ) , tensão de flexão na viga (σ) , carga de flambagem na barra (P_c) , deflexão na barra (δ) e restrições laterais, que são descritas nas equações (5.8a) à (5.8g).

$$g_1(x) \equiv \tau(x) - \tau_{max} \leq 0 \quad (5.8a)$$

$$g_2(x) \equiv \sigma(x) - \sigma_{max} \leq 0 \quad (5.8b)$$

$$g_3(x) \equiv x_1 - x_4 \leq 0 \quad (5.8c)$$

$$g_4(x) \equiv 0,10471x_1^2 + 0,04811x_3x_4(14,0 + x_2) - 5 \leq 0 \quad (5.8d)$$

$$g_5(x) \equiv 0, 125 - x_1 \leq 0 \quad (5.8e)$$

$$g_6(x) \equiv \delta(x) - \delta_{max} \leq 0 \quad (5.8f)$$

$$g_7(x) \equiv P - P_c(x) \leq 0 \quad (5.8g)$$

$$0, 1 \leq x_1 \leq 2, 0; \quad 0, 1 \leq x_2 \leq 10, 0; \quad 0, 1 \leq x_3 \leq 10, 0; \quad 0, 1 \leq x_4 \leq 2, 0 \quad (5.8h)$$

em que:

$$\tau(x) = \sqrt{\tau'^2 + 2\tau'\tau''\frac{x_2}{2R} + \tau''^2} \quad (5.9a)$$

$$\tau' = \frac{P}{\sqrt{2}x_1x_2} \quad (5.9b)$$

$$\tau'' = \frac{MR}{J} \quad (5.9c)$$

$$M = P \left(L + \frac{x_2}{2} \right) \quad (5.9d)$$

$$R = \sqrt{\frac{x_2^2}{4} + \left(\frac{x_1 + x_3}{2} \right)^2} \quad (5.9e)$$

$$J = 2 \left[\sqrt{2}x_1x_2 \left(\frac{x_2^2}{12} + \left(\frac{x_1 + x_3}{2} \right)^2 \right) \right] \quad (5.9f)$$

$$\sigma(x) = \frac{6PL}{x_4x_3^2} \quad (5.9g)$$

$$\delta(x) = \frac{4PL^3}{Ex_3^3x_4} \quad (5.9h)$$

$$P_c(x) = \frac{4,013E\sqrt{\frac{x_3^2x_4^6}{36}}}{L^2} \left(1 - \frac{x_3}{2L}\sqrt{\frac{E}{4G}} \right) \quad (5.9i)$$

Os parâmetros considerados nesta análise são (COELLO & MONTES, 2002): $P = 6000$ lb (força aplicada), $L = 14$ in (comprimento em balanço da viga), $E = 30 \times 10^6$ psi (módulo de elasticidade), $G = 12 \times 10^6$ psi (módulo de cisalhamento), $\tau_{max} = 13600$ psi (tensão máxima de cisalhamento), $\sigma_{max} = 30000$ psi (tensão máxima de flexão) e $\delta_{max} = 0,25$ in (deflexão máxima).

A Tabela 20 são apresentados os resultados obtidos (média, desvio padrão e melhor solução) aplicando o algoritmo de BFE considerando os seguintes parâmetros: número de gerações (140), tamanho da população (35), número de difusões (4) e passo (1), bem como 20 execuções independentes da referida estratégia de otimização. De forma geral, na referida tabela observa-se que o algoritmo de BFE sempre convergiu para a mesma solução, conforme pode ser visto pelos valores de média e desvio padrão, o que demonstra robustez para o problema em questão.

Tabela 20: Resultados Obtidos pelo Algoritmo de BFE para o Problema do Projeto de uma Viga Soldada (N_{eval} é o Número de Avaliações da Função Objetivo).

	Média	Desvio Padrão	Melhor Solução
x_1 (in)	0,205730	$0,011860 \times 10^{-9}$	0,205730
x_2 (in)	1,964959	$0,160614 \times 10^{-9}$	1,964959
x_3 (in)	9,036624	$0,048379 \times 10^{-9}$	9,036624
x_4 (in)	0,205730	$0,001335 \times 10^{-9}$	0,205730
f (\$)	1,519802	$9,234190 \times 10^{-12}$	1,519802

Já na Tabela 21 são apresentados os resultados ótimos obtidos por outros métodos de otimização em comparação com aqueles obtidos pelo algoritmo de BFE. O problema foi resolvido por Coello & Montes (2002) usando um método derivado dos tradicionais Algoritmos Genéticos considerando uma estratégia de nichos baseado em Pareto, por Rashedi *et al.* (2009) usando o Algoritmo de Busca Gravitacional e por Mirjalili & Lewis (2016) usando o Algoritmo de Otimização por Baleias.

Tabela 21: Resultados Obtidos para o Problema do Projeto de uma Viga Considerando Diferentes Abordagens.

	BFE	Coello & Montes (2002)	Mirjalili & Lewis (2016)	Rashedi <i>et al.</i> (2009)
x_1 (in)	0,205730	0,205986	0,205396	0,182129
x_2 (in)	1,964959	3,471328	3,484293	3,856979
x_3 (in)	9,036624	9,020224	9,037426	10,000000
x_4 (in)	0,205730	0,206480	0,206276	0,202376
f (\$)	1,519802	1,728226	1,730499	1,879952
N_{eval}	4935	80000	9900	10750

Nesta tabela observa-se que o valor da função objetivo obtido pelo algoritmo de BFE foi inferior aos apresentados por outros autores. Este menor valor se deve à diferença entre a estimativa obtida pelo algoritmo de BFE e as outras abordagens consideradas para a variável de projeto x_2 . Cabe ressaltar que a solução encontrada, na média, requereu um número menor de avaliações em comparação com a literatura, resultando em uma redução de, aproximadamente, 93,84%, 50,16% e 54,09% avaliações da função objetivo com relação aos trabalhos de Coello & Montes (2002), Mirjalili & Lewis (2016) e Rashedi *et al.* (2009), respectivamente.

5.2.3 Projeto de um Vaso de Pressão

Os vasos de pressão são reservatórios que necessitam resistir à pressões internas diferentes da ambiente de forma a garantir a segurança, a qualidade do conteúdo no seu interior, sendo este equipamento bastante empregado em processos industriais que fazem uso de fluidos ou gases (WAGNER, 2015).

Considere o projeto de um vaso de pressão em que deseja-se minimizar o seu custo total (COELLO & MONTES, 2002), conforme ilustrado na Figura 13.

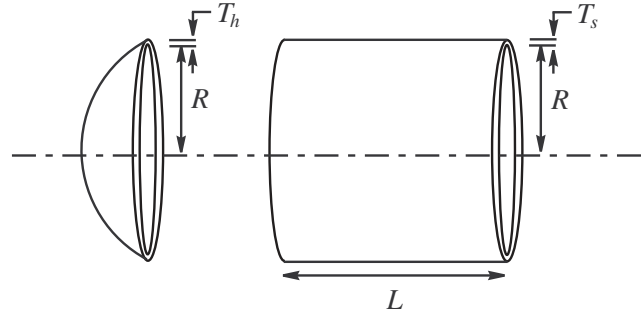


Figura 13: Projeto de um Vaso de Pressão.

Nesta aplicação as variáveis de projeto são: a espessura da casca (T_s), a espessura da tampa (T_h), o raio interno (R) e o comprimento da seção cilíndrica do vaso (L), representadas por $\vec{x} = [x_1 \ x_2 \ x_3 \ x_4]$, respectivamente. Todas as variáveis são apresentadas em polegadas (in), T_s e T_h são inteiros múltiplos de 0,0625 in.

Matematicamente, o problema de projeto pode ser descrito como (COELLO & MONTES, 2002):

$$\min f(x) = 0,6224x_1x_3x_4 + 1,7781x_2x_3^2 + 3,1661x_1^2x_4 + 19,84x_1^2x_3 \quad (5.10)$$

Sujeito às seguintes restrições (entre variáveis de projeto e de volume):

$$g_1(x) \equiv -x_1 + 0,0193x_3 \leq 0 \quad (5.11a)$$

$$g_2(x) \equiv -x_2 + 0,00954x_3 \leq 0 \quad (5.11b)$$

$$g_3(x) \equiv -\pi x_3^2x_4 - \frac{4}{3}\pi x_3^3 + 1296000 \leq 0 \quad (5.11c)$$

$$g_4(x) \equiv x_4 - 240 \leq 0 \quad (5.11d)$$

$$0 \leq x_1 \leq 10; \quad 0 \leq x_2 \leq 10; \quad 10 \leq x_3 \leq 100; \quad 100 \leq x_4 \leq 240 \quad (5.11e)$$

Para resolver este estudo de caso em que as variáveis são múltiplos inteiros de 0,0625; é necessária uma pequena adaptação no momento em que o candidato é avaliado segundo a função objetivo. A proposta de adaptação considerada para essa finalidade foi proposta por Lobato (2008). Em linhas gerais, esta abordagem consiste em gerar o candidato à solução do problema de otimização no domínio contínuo, conforme descrito para o algoritmo de BFE. A partir daí, efetua-se possíveis correções em seu valor de acordo com o exemplo apresentado a seguir:

- Seja um problema com uma única variável de projeto x_{disc} que assume apenas valores discretos ($x_{disc} \in \{1,24 \ 2,45 \ 3,09 \ 4,67 \ 5,87\}$). Inicialmente define-se como domínio da variável de projeto um intervalo qualquer, por exemplo $0 \leq x_{cont} \leq 1$. Dessa maneira, como se tem 5 valores discretos, por exemplo, pode-se escrever o seguinte critério de escolha:

$$x_{disc} = \begin{cases} 1,24 & \text{se } 0,0 \leq x_{cont} < 0,2 \\ 2,45 & \text{se } 0,2 \leq x_{cont} < 0,4 \\ 3,09 & \text{se } 0,4 \leq x_{cont} < 0,6 \\ 4,67 & \text{se } 0,6 \leq x_{cont} < 0,8 \\ 5,87 & \text{se } 0,8 \leq x_{cont} \leq 1,0 \end{cases} \quad (5.12)$$

Assim, se um candidato tiver valor contínuo arbitrado como sendo 0,564, esse valor é corrigido para 3,09 (que seria o domínio discreto correspondente para essa variável contínua). Com base nesta lógica, pode-se tratar outros tipos de variáveis, como inteiras e binárias.

Neste contexto, realizando-se esta transformação antes de se avaliar a função objetivo, permite-se resolver problemas considerando diferentes tipos de variáveis sem ter que alterar a estrutura original do algoritmo de BFE.

A Tabela 22 apresenta os resultados obtidos (média, desvio padrão e melhor solução) com a aplicação do algoritmo de BFE considerando os seguintes parâmetros: número de gerações (150), tamanho da população (50), número de difusões (3) e passo (1), bem como 20 execuções independentes da referida estratégia de otimização. Os resultados apresentados nesta tabela demonstram que a metodologia proposta foi capaz de encontrar boas estimativas para o vetor de variáveis de projeto, bem como do valor de função objetivo, conforme pode ser visto pelos valores de média e desvio padrão. Cabe ressaltar que, para as x_3 , x_4 e $f(x)$ foram observados os maiores valores desvio padrão. Apesar disto, entende-se que o algoritmo do BFE se mostrou, devido a ordem de grande da função objetivo, uma abordagem robusta.

Tabela 22: Resultados Obtidos pelo Algoritmo de BFE para o Problema do Projeto de um Vaso de Pressão.

	Média	Desvio Padrão	Melhor Solução
x_1 (in)	0,81250	0	0,812500
x_2 (in)	0,43750	0	0,437500
x_3 (in)	42,09717	0,002075	42,098456
x_4 (in)	176,6608	0,041142	176,636488
f (\$)	6060,0932	0,674126	6059,714085

A Tabela 23 apresenta os resultados ótimos obtidos por outros métodos de otimização, bem como pelo algoritmo de BFE. Para essa finalidade Coello & Montes (2002) resolveu este estudo de caso considerando um método derivado dos Algoritmos Genéticos associado com uma estratégia de nichos baseado em Pareto. Já Rashedi *et al.* (2009) e Mirjalili & Lewis (2016) resolveram este mesmo problema usando o Algoritmo de Busca Gravitacional e o Algoritmo de Otimização por Baleias, respectivamente.

Tabela 23: Resultados Obtidos para o Problema do Projeto de um Vaso de Pressão Considerando Diferentes Abordagens (N_{eval} é o Número de Avaliações da Função Objetivo).

	BFE	Coello & Montes (2002)	Mirjalili & Lewis (2016)	Rashedi <i>et al.</i> (2009)
x_1 (in)	0,8125	0,81250	0,81250	1,12500
x_2 (in)	0,4375	0,43750	0,43750	0,62500
x_3 (in)	42,0984	42,09739	42,09827	55,98866
x_4 (in)	176,6364	176,65404	176,63899	84,45420
f (\$)	6059,714	6059,94634	6059,74100	8538,83590
N_{eval}	7550	80000	6300	7110

Nesta tabela observa-se que o valor do custo obtido pelo algoritmo de BFE ($f(x)$) é coerente com aqueles reportados por outros autores, demonstrando a qualidade da abordagem proposta. Em

termos do custo computacional, observa-se uma redução de, aproximadamente, 90,57% do número de avaliações da função objetivo do algoritmo de BFE em relação ao trabalho de Coello & Montes (2002). Todavia, em relação aos trabalhos de Mirjalili & Lewis (2016) e Rashedi *et al.* (2009), foi observado um aumento de, aproximadamente, 19,84% e 6,19%, respectivamente, em relação a metodologia proposta. Apesar deste aumento, considera-se que o algoritmo de BFE foi competitivo com relação a estas duas abordagens.

5.2.4 Projeto de uma Coluna Tubular Sujeita a uma Carga Compressiva

Colunas tubulares podem estar ligadas à construções onde a arquitetura justifica seu uso, porém, pelo ponto de vista estrutural, a mesma pode oferecer algumas vantagens, sendo a principal delas a boa resistência à compressão axial com baixa tendência ao desenvolvimento de instabilidades localizadas (FREITAS, 2009).

Para avaliar a aplicabilidade do algoritmo de BFE considere o estudo de caso proposto por Nigdeli *et al.* (2016). Este consiste de uma coluna tubular submetido a uma carga P axialmente aplicada em que suas extremidades são apoiadas por rolamentos articulados e está sujeita à restrições de compressão e flambagem, conforme a Figura 14. Do ponto de vista da otimização deseja-se minimizar o custo material e de construção da coluna ($f(x)$) via determinação das suas dimensões (d e t).

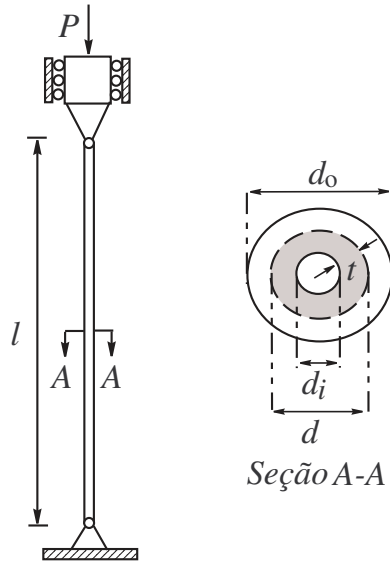


Figura 14: Projeto de uma Coluna Tubular Sujeita a uma Carga Compressiva.

Matematicamente, considerando a notação ($\vec{x} = [x_1 \ x_2] = [d \ t]$) (cm), o problema de otimização é definido como (NIGDELI *et al.*, 2016):

$$\min f(x) = 9,8x_1x_2 + 2x_1 \quad (5.13)$$

Sujeito às seguintes restrições (limites laterais, de compressão e de flambagem):

$$g_1(x) \equiv \frac{P}{\pi x_1 x_2 \sigma_y} - 1 \leq 0 \quad (5.14a)$$

$$g_2(x) \equiv \frac{8Pl^2}{\pi^3 E x_1 x_2 (x_1^2 + x_2^2)} - 1 \leq 0 \quad (5.14b)$$

$$g_3(x) \equiv \frac{2,0}{x_1} - 1 \leq 0 \quad (5.14c)$$

$$g_4(x) \equiv \frac{x_1}{14} - 1 \leq 0 \quad (5.14d)$$

$$g_5(x) \equiv \frac{0,2}{x_2} - 1 \leq 0 \quad (5.14e)$$

$$g_6(x) \equiv \frac{x_2}{0,8} - 1 \leq 0 \quad (5.14f)$$

$$2 \leq x_1 \leq 14; \quad 0,2 \leq x_2 \leq 0,8 \quad (5.14g)$$

Para resolver este estudo de caso são considerados os seguintes parâmetros (NIGDELI *et al.*, 2016): $P = 2500$ kgf (força axial), $\sigma_y = 500$ kgf/cm² (tensão de escoamento), $E = 0,85 \times 10^6$ kgf/cm² (módulo de elasticidade), $\rho = 0,0025$ kgf/cm³ (densidade do material) e $l = 250$ cm (comprimento da coluna). É importante ressaltar que P deve ser menor que a carga de flambagem da coluna, definida como a carga de flambagem de Euler:

$$P_{kr} = \frac{\pi^2 EI}{l^2} \quad (5.15)$$

onde I é o momento de inércia da seção da coluna tubular.

Na Tabela 24 são apresentados os resultados obtidos (média, desvio padrão e melhor solução) aplicando o algoritmo de BFE considerando os seguintes parâmetros: número de gerações (100), tamanho da população (30), número de difusões (4) e passo (1), bem como 20 execuções independentes da referida estratégia de otimização. Como pode ser observado nesta tabela, os valores médios e de desvios padrão demonstram que o algoritmo de BFE foi capaz de sempre convergir para um mesmo valor no que tange o vetor de variáveis de projeto e de função objetivo.

Tabela 24: Resultados Obtidos pelo Algoritmo de BFE para o Problema do Projeto de uma Coluna Tubular Sujeita a uma Carga Compressiva.

	Média	Desvio Padrão	Melhor Solução
x_1 (cm)	5,451157	$2,724431 \times 10^{-6}$	5,451156
x_2 (cm)	0,291965	$1,720006 \times 10^{-7}$	0,291965
f (\$)	26,499502	$5,814367 \times 10^{-6}$	26,499498

Para avaliar o algoritmo de BFE em termos de convergência e custo computacional, na Tabela 25 são apresentados os melhores resultados obtidos considerando diferentes métodos de otimização e o algoritmo de BFE. Para essa finalidade considera-se o algoritmo proposto por Hsu & Liu (2007) e que considera o desenvolvimento de um método baseado em um controlador do tipo *fuzzy* proporcional-derivativo, o algoritmo de Busca Cuco proposto por Gandomi *et al.* (2013) e por o algoritmo de Polinização de Flores proposto por Nigdeli *et al.* (2016).

A partir da análise das informações apresentados na Tabela 25 é possível concluir que os resultados obtidos pelo algoritmo de BFE estão em concordância com aqueles reportados por outros trabalhos. Além disso, enfatiza-se que todas as restrições do estudo de caso foram perfeitamente atendidas durante o processo de otimização, sendo que as restrições representadas pelas Equações

(5.14a) e (5.14b) são ativas (as restrições de desigualdade se tornam restrições de igualdade) na solução ótima encontrada. Finalmente, em termos do número de avaliações da função objetivo observa-se uma redução de, aproximadamente, 79,80% e 39,40% em relação aos trabalhos de Gandomi *et al.* (2013) e Nigdeli *et al.* (2016), respectivamente.

Tabela 25: Resultados Obtidos para o Problema do Projeto de uma Coluna Tubular Sujeita a uma Carga Compressiva Considerando Diferentes Abordagens (N_{eval} é o Número de Avaliações da Função Objetivo).

	BFE	Hsu & Liu (2007)	Gandomi <i>et al.</i> (2013)	Nigdeli <i>et al.</i> (2016)
x_1 (cm)	5,451156	4,450700	5,451390	5,451160
x_2 (cm)	0,291965	0,292000	0,291960	0,291965
g_1	$1,677223 \times 10^{-6}$	-0,00003	0,024100	$9,434300 \times 10^{-7}$
g_2	$1,772274 \times 10^{-6}$	0,000132	-0,109500	$-4,249000 \times 10^{-7}$
g_3	-0,633105	-0,633100	-0,633100	-0,633100
g_4	-0,610632	-0,610700	-0,610600	-0,610600
g_5	-0,314986	-0,315100	-0,315000	-0,315000
g_6	-0,635044	-0,635000	-0,635100	-0,635000
f (\$)	26,499498	26,499100	26,532170	26,499480
N_{eval}	3030	NI	15000	5000

NI - Não informado pelos autores.

5.2.5 Projeto de uma Barra Cantilever

A barra Cantilever é um elemento estrutural muito utilizado na engenharia, se destacando em sistemas que apresentam dentes (como as engrenagens), na estrutura de telhados, nas asas de um avião, nos eixos dianteiros de um automóvel, entre outras aplicações (UFPA, 2011).

O problema de otimização abordado aqui consiste na minimização do custo de uma barra cantilever, conforme estudado por (NIGDELI *et al.*, 2016). A barra se encontra engastada em uma das extremidades e é aplicada uma força vertical P na extremidade livre, sendo que a espessura (t) é fixada em $2/3$ para cada seção. A mesma é dividida em 5 diferentes seções transversais, conforme Figura 15.

Matematicamente, o problema pode ser representado como (NIGDELI *et al.*, 2016):

$$\min f(x) = 0,0624(x_1 + x_2 + x_3 + x_4 + x_5) \quad (5.16)$$

em que x_i ($i = 1, \dots, 5$) representam as alturas de cada seção transversal. Este problema está sujeito às seguintes restrições:

$$g \equiv \frac{61}{x_1^3} + \frac{37}{x_2^3} + \frac{19}{x_3^3} + \frac{7}{x_4^3} + \frac{1}{x_5^3} - 1 \leq 0 \quad (5.17a)$$

$$0,01 \leq x_j \leq 100 \quad j = 1, 2, \dots, 5. \quad (5.17b)$$

Na Tabela 26 são apresentados os resultados obtidos (média, desvio padrão e melhor solução) aplicando o algoritmo de BFE considerando os seguintes parâmetros: número de gerações (110), tamanho da população (50), número de difusões (4) e passo (1), bem como 20 execuções independentes da referida estratégia de otimização. Nesta tabela percebe-se que o algoritmo de BFE sempre foi capaz de convergir para um mesmo valor de variáveis de projeto e, consequentemente, de função objetivo, conforme pode ser visto pelos valores de média e desvio padrão.

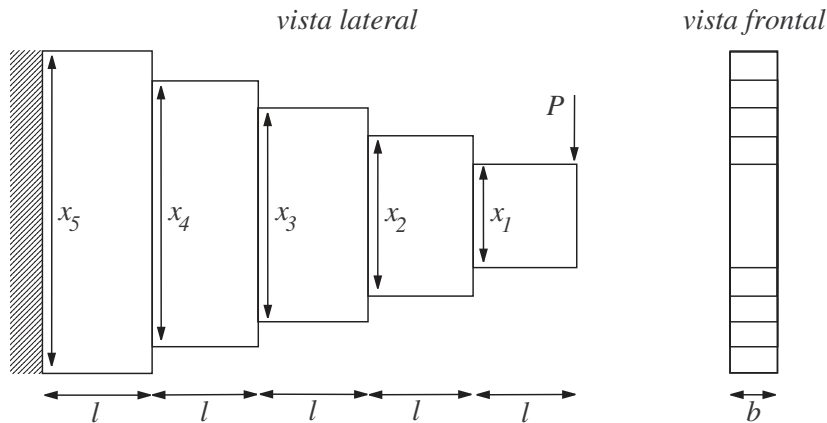


Figura 15: Projeto de uma Barra Cantilever.

Tabela 26: Resultados Obtidos pelo Algoritmo de BFE para o Problema do Projeto de uma Barra Cantilever.

	Média	Desvio Padrão	Melhor Solução
x_1 (cm)	6,016466	0,029978	6,021266
x_2 (cm)	5,315296	0,024496	5,313862
x_3 (cm)	4,495288	0,026771	4,480014
x_4 (cm)	3,498183	0,023045	3,504302
x_5 (cm)	2,150881	0,019059	2,154423
f (cm)	1,340110	$8,811669 \times 10^{-5}$	1,339969

Já na Tabela 27 são apresentadas as melhores soluções reportadas pela literatura especializada. Este problema foi resolvido por Chickermane & Gea (1996) utilizando o denominado Algoritmo Convexo Generalizado, por Gandomi *et al.* (2013) usando um algoritmo de Busca Cuco e por Nigdeli *et al.* (2016) utilizando o algoritmo de Polinização de Flores. Nesta tabela observa-se que os resultados obtidos pelo algoritmo de BFE estão em concordância com aqueles reportados por outros trabalhos. Todavia, enfatiza-se que o resultado foi obtido às custas de uma redução de, aproximadamente, 63% e 26,25% no que tange o número de avaliações da função objetivo em relação aos trabalhos de Gandomi *et al.* (2013) e Nigdeli *et al.* (2016), respectivamente.

5.2.6 Controle de uma Aeronave de Asa Fixa

A perda do controle de aeronaves durante o voo é uma das principais causas de morte em acidentes aéreos. Para minimizar essa possibilidade é importante o desenvolvimento de controladores capazes de atuar em diferentes situações que podem acontecer. Neste contexto, propõe-se como aplicação nesta dissertação o uso do algoritmo de BFE para a determinação de parâmetros para auxiliar a calibração de um controlador em uma aeronave de asa fixa do modelo Cessna 172, conforme proposto e estudado por Bernardes (2018).

Tabela 27: Resultados Obtidos para o Problema do Projeto de uma Barra Cantilever Considerando Diferentes Abordagens (N_{eval} é o Número de Avaliações da Função Objetivo).

Variáveis	BFE	Chickermane & Gea (1996)	Gandomi <i>et al.</i> (2013)	Nigdeli <i>et al.</i> (2016)
x_1 (cm)	6,0212	6,010000	6,008900	6,020200
x_2 (cm)	5,3138	5,300000	5,304900	5,308200
x_3 (cm)	4,4800	4,490000	4,502300	4,504200
x_4 (cm)	3,5043	3,490000	3,507700	3,485600
x_5 (cm)	2,1544	2,150000	2,150400	2,155700
f (cm)	1,3399	1,340000	1,339990	1,339970
N_{eval}	5550	NI	15000	7525

NI - Não informado pelos autores.

Neste problema, a atitude de uma aeronave é composta pelos ângulos de Euler, definidos por: ângulo de arfagem (Θ), ângulo de rolagem (Φ) e ângulo de guinada (Ψ), conforme a Figura 16.

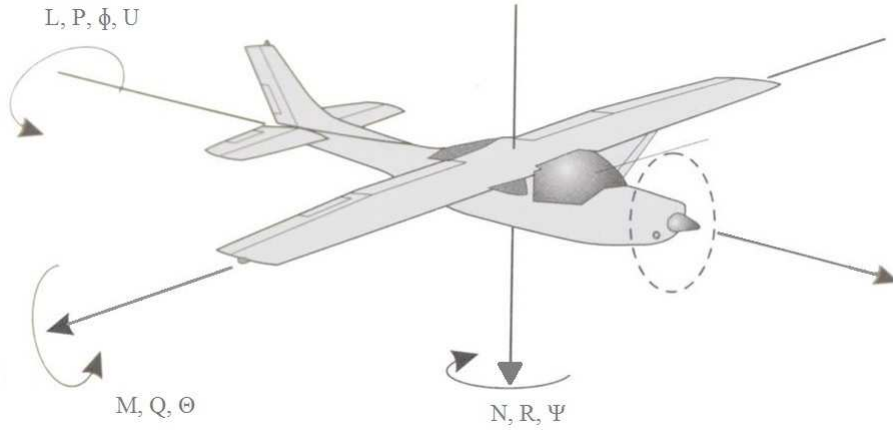


Figura 16: Definição dos Eixos, Velocidades, Forças, Momentos e Ângulos de Euler da Aeronave (BERNARDES, 2018).

No presente estudo as variáveis de projeto são os parâmetros utilizados por um controlador Neuro-fuzzy (x_{SBRF}) que tem por função definir a deflexão das superfícies de controle da aeronave, a saber, o profundor ΔP_k (que controla o ângulo de arfagem θ e a altitude da aeronave h), o aileron Δa_k , e o leme Δr_k (responsáveis por controlar o ângulo de guinada Ψ e o ângulo de rolagem Φ). As Figuras 17 e 18 representam os sistemas de controle, que foram divididos em dois modelos distintos, o controle longitudinal e o látero-direcional.

Para encontrar os parâmetros utilizados no referido controlador, emprega-se do método *off-line*, onde a planta é submetida a um ambiente de perturbações pré-determinadas e sua execução é em cascata, isto é; os parâmetros da malha interna são encontrados, fixados e assim busca-se encontrar os parâmetros da malha externa.

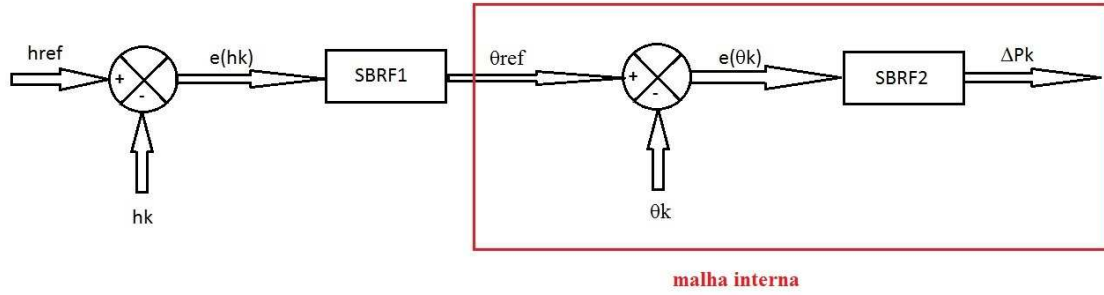


Figura 17: Sistema de Controle Longitudinal (BERNARDES, 2018).

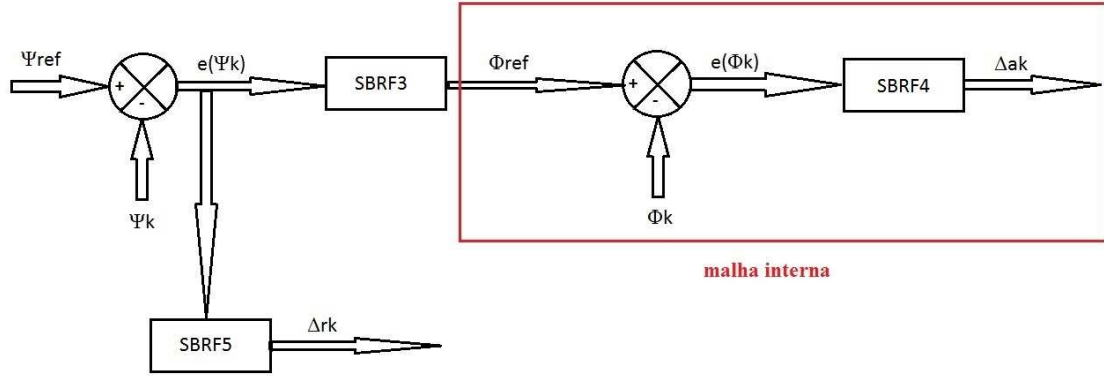


Figura 18: Sistema de Controle Látero-direcional (BERNARDES, 2018).

Neste caso, genericamente, a função objetivo considerada em cada uma das aplicações busca a minimização do erro RMS (*Root Mean Square*) - E_{RMS} , definido como:

$$\min E_{RMS} = \frac{\sum \sqrt{(x_{ref,k} - x_k)^2}}{n} \quad (5.18)$$

onde $x_{ref,k}$ é o valor de referência da variável x_k na k -ésima iteração e n é o número de pontos disponíveis para o treinamento do sistema neuro-fuzzy.

• Controle Longitudinal com Malha Interna

Como aplicação prática considere a determinação do parâmetro (x_{SBRF2}) utilizado pelo controlador inteligente. Este é responsável por determinar a deflexão do profundor (ΔP_k) ao longo da simulação. O ângulo de arfagem (θ), utilizado na formulação da função objetivo, tem seu valor de referência ($\theta_{ref,k}$) definido como sendo igual a zero em toda a simulação (existem perturbações em θ_k nos instantes $t = 0$ s e $t = 30$ s). Para formular o problema de otimização considera-se o seguinte domínio: $0 \leq x_{SBRF2} \leq 5$ (°). Para resolver este estudo de caso será empregado o algoritmo de BFE com os seguintes parâmetros: tamanho da população (20), número de gerações (10), número

de difusões (10) e passo (1). Os resultados obtidos serão comparados com aqueles obtidos por Bernardes (2018) usando os algoritmos de Evolução Diferencial (ED) (STORN & PRICE, 1996) e de Recozimento Simulado (RS) (METROPOLIS *et al.*, 1953). Nestas estratégias foram considerados os seguintes parâmetros: ED (tamanho da população (20), número de gerações (15), probabilidade de cruzamento (0,95) e fator de perturbação (0,4)) e RS (temperatura inicial (15), número de ciclos de resfriamento (200), número de iterações para cada temperatura (5), desvio padrão da distribuição normal (0,06), decaimento da temperatura (0,9)). Com estes parâmetros, o número de avaliações da função objetivo para as estratégias ED, RS e BFE foram iguais a 320, 1207 e 220, respectivamente. Ressalta-se que cada algoritmo foi executado 5 vezes para a obtenção dos resultados que serão apresentados.

Os resultados (média e desvio padrão) obtidos para a variável de projeto são apresentados na Tabela 28. De forma geral percebe-se que os algoritmos de ED e BFE convergiram para a mesma solução em termos de variável de projeto e de função objetivo. Já o algoritmo de RS convergiu para uma solução próxima em termos de função objetivo, mas distinta em termos da variável de projeto. Finalmente, cabe ressaltar que, em termos do número de avaliações da função objetivo, o algoritmo de BFE foi o que apresentou o menor custo computacional.

Tabela 28: Resultados Obtidos para o Problema do Controle Longitudinal com Malha Interna Considerando Diferentes Abordagens.

Método	$x_{\text{SBRF2}} (^{\circ})$		$f (^{\circ})$	
	Média	Desvio Padrão	Média	Desvio Padrão
ED	3,965140	$8,944272 \times 10^{-5}$	$1,237200 \times 10^{-4}$	0,000000
RS	2,648214	$3,266641 \times 10^{-2}$	$1,660639 \times 10^{-4}$	$8,079272 \times 10^{-9}$
BFE	3,965127	$2,576338 \times 10^{-4}$	$1,237248 \times 10^{-4}$	0,000000

Na Figura 19 são apresentados os perfis simulados com a solução ótima obtida por cada um dos algoritmos considerados. É possível perceber que o perfil obtido pelo algoritmo de BFE tem comportamento semelhante aos obtidos pelo algoritmo de ED. No instante $t = 0$ s existe um pico no ângulo de arfagem proveniente da perturbação de 30° . Fisicamente, verifica-se que a ação de controle minimiza o efeito da perturbação sobre o sistema, que acomoda-se em torno do valor de referência em $t = 40$ s. Já para o perfil obtido usando o algoritmo de RS percebe-se que a resposta do sistema ao controle é mais rápida que a obtida nos algoritmos de ED e de BFE, uma vez que o ângulo de arfagem responde mais rapidamente à ação de deflexão do profundor. Em $t = 40$ s o sistema atinge a acomodação e em $t = 50$ s o mesmo encontra-se efetivamente em regime permanente.

• Controle Longitudinal com Malha Externa

Nesta segunda aplicação considera-se a determinação do parâmetro (x_{SBRF1}) utilizado pelo controlador neuro-fuzzy, responsável por determinar o ângulo de arfagem de referência ($\theta_{ref,k}$), parâmetro de entrada da malha interna deste controlador, conforme definido anteriormente. A altitude da aeronave (h) tem seu valor de referência ($h_{ref,k}$) definido de forma que, para o intervalo de tempo $0 \leq t \leq 75$ (s), a altitude seja de 50 metros e, para $75 < t \leq 150$ (s), a altitude seja igual a 0 metros. Para este caso não são definidas perturbações no sistema, sendo considerado apenas a variação do parâmetro $h_{ref,k}$. Como função objetivo será considerada a minimização do erro RMS referente à altitude da aeronave ($e(h_k)$). O domínio para a variável de projeto é definida como $0 \leq x_{\text{SBRF1}} \leq 30 (^{\circ})$. Em relação a cada um dos algoritmos utilizados, serão considerados os seguintes parâmetros: BFE (tamanho da população (20), número de gerações (10), número de difusões (10) e passo (1));

ED (tamanho da população (20), número de gerações (20), probabilidade de cruzamento (0,95) e fator de perturbação (0,4)) e RS (temperatura inicial (12), número de ciclos de resfriamento (250), número de iterações para cada temperatura (5), desvio padrão da distribuição normal (0,06), decaimento da temperatura (0,9)). Com estes parâmetros, o número de avaliações da função objetivo para as estratégias ED, RS e BFE foram iguais a 420, 1207 e 220, respectivamente. Cada algoritmo foi executado 5 vezes para a obtenção dos resultados que serão apresentados.

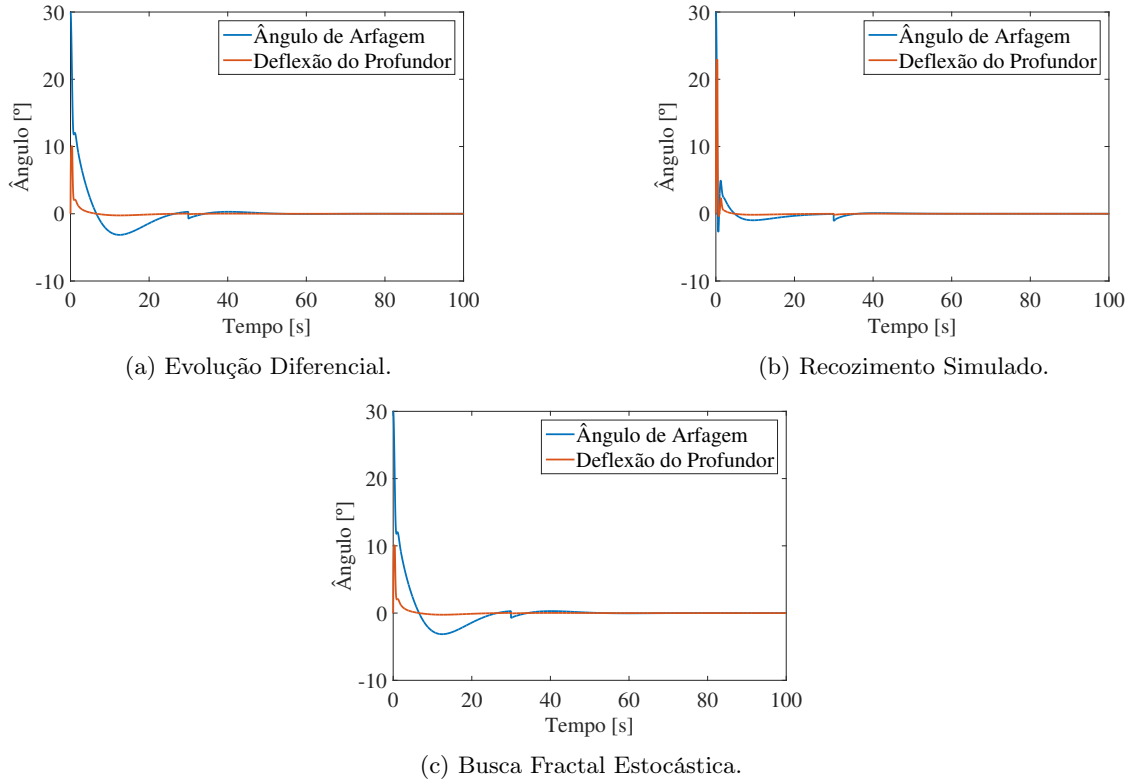


Figura 19: Ângulo de Arfagem e Deflexão do Profundor Considerando Diferentes Abordagens.

Na Tabela 29 são apresentados os resultados (média e desvio padrão) obtidos para a variável de projeto e para a função objetivo considerando diferentes estratégias de otimização.

Tabela 29: Resultados Obtidos para o Problema do Controle Longitudinal com Malha Externa Considerando Diferentes Abordagens.

Método	x_{SBRF1} (°)		$f(x_{ótimo})$ (m)	
	Média	Desvio Padrão	Média	Desvio Padrão
ED	27,027381	$3,543747 \times 10^{-3}$	0,021134	0,000000
RS	25,392747	$4,871896 \times 10^{-3}$	0,024555	$3,73321 \times 10^{-7}$
BFE	27,027212	$8,426204 \times 10^{-4}$	0,021134	0,000000

Nesta tabela observa-se que os algoritmos ED e BFE sempre convergiram para a mesma solução (variável de projeto e função objetivo), visto o valor médio e do desvio padrão. Já o algoritmo de RS convergiu para uma solução próxima em termos de função objetivo, mas distinta em termos da variável de projeto. Em relação ao número de avaliações da função objetivo, o algoritmo de BFE foi o que apresentou o menor custo computacional.

Na Figura 20 são apresentados os perfis ótimos de altura e deflexão do profundor da aeronave considerando cada um dos algoritmos apresentados.

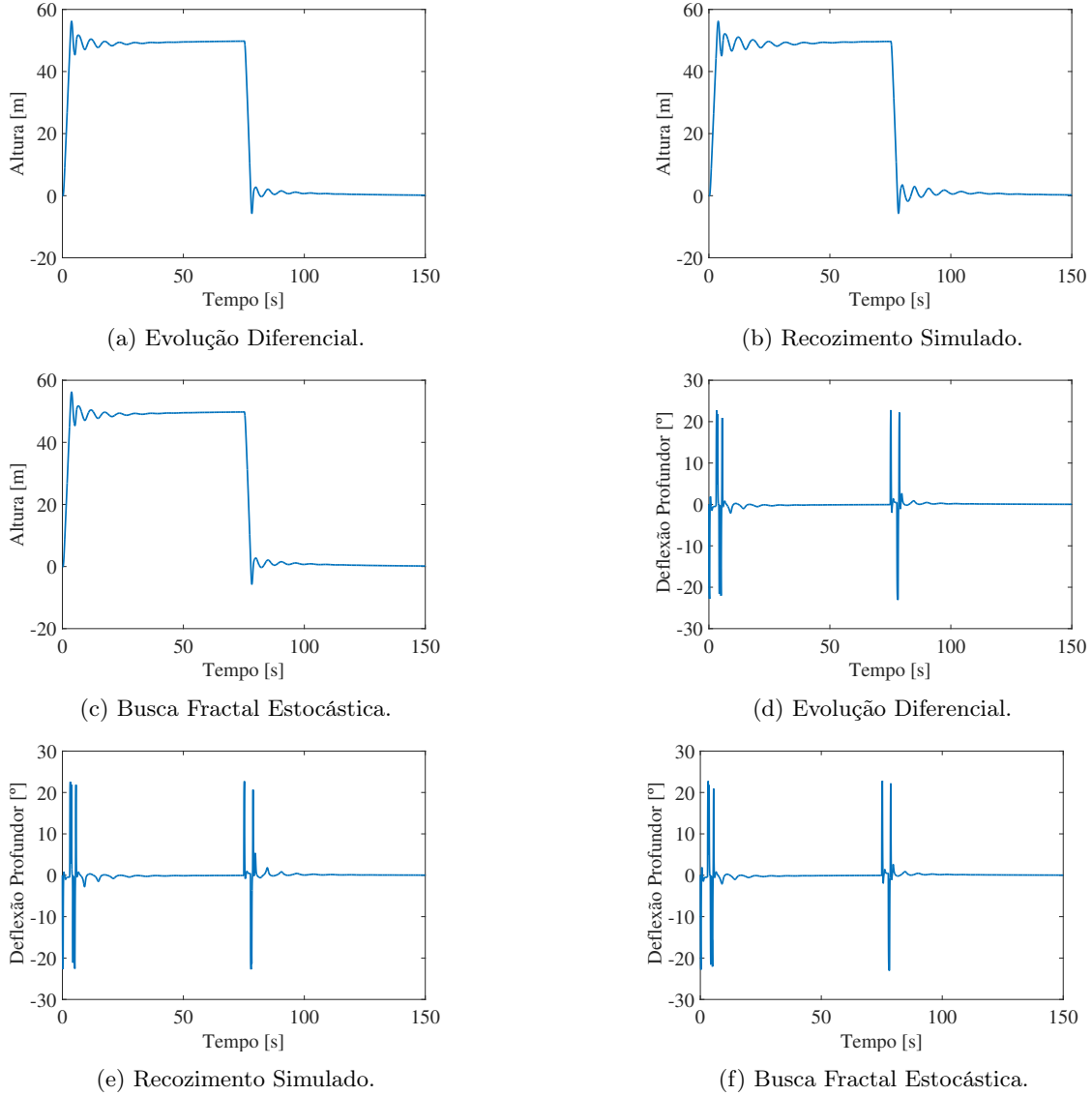


Figura 20: Altura e Deflexão do Profundor da Aeronave Considerando Diferentes Abordagens.

Nestas figuras observa-se perfis semelhantes obtidos pelos algoritmos de BFE e de ED, sendo os

mesmos coerentes com os resultados reportados por Bernardes (2018). No intervalo de $0 \leq t \leq 25$ (s) a altura se encontra em regime transitório, oscilando em torno do ponto de referência de 50 metros, acomodando-se em $t = 8$ s. Nos intervalos compreendidos entre $25 < t \leq 75$ (s) e $t > 100$ (s) o sistema se encontra estável. O sistema atinge o regime permanente em, aproximadamente, $t = 100$ s. Em relação ao comportamento da superfície de controle (ΔP_k) durante a simulação, o controlador apresenta um ganho elevado na presença de grandes perturbações, diminuindo sua magnitude de ação até atingir o regime permanente, quando permanece próximo de zero.

• Controle Látero-Direcional com Malha Interna

Para a malha interna do controle látero-direcional, considere a determinação do parâmetro (x_{SBRF4}) do controlador, responsável por controlar a deflexão do aileron (Δa_k) durante a simulação. O ângulo de rolagem (Φ), possui valor de referência ($\Phi_{ref,k}$) como sendo igual a zero em toda a simulação, existindo perturbações (Φ_k) nos instantes $t=0$ s e $t=30$ s, respectivamente. A função objetivo considerada é a minimização do erro RMS referente ao ângulo de rolagem ($e(\Phi_k)$). O domínio da variável de projeto é definido como $0 \leq x_{\text{SBRF4}} \leq 5$ (°). Para resolver este estudo de caso será empregado o algoritmo de BFE com os seguintes parâmetros: tamanho da população (20), número de gerações (10), número de difusões (10) e passo (1). Para fins de comparação foram considerado as estratégias ED e RS considerando os seguintes parâmetros: ED (tamanho da população (30), número de gerações (30), probabilidade de cruzamento (0,95) e fator de perturbação (0,4)) e RS (temperatura inicial (10), número de ciclos de resfriamento (200), número de iterações para cada temperatura (5), desvio padrão da distribuição normal (0,1), decaimento da temperatura (0,9)). Para os parâmetros apresentados, o número de avaliações da função objetivo considerando as estratégias ED, RS e BFE foram iguais a 930, 607 e 220, respectivamente. Cada algoritmo foi executado 5 vezes para a obtenção dos resultados que serão apresentados. A Tabela 30 apresenta os resultados de média e desvio padrão obtidos considerando as estratégias de otimização propostas. Os resultados obtidos por todos os algoritmos foram muito similares em termos da variável de projeto e da função objetivo. Todavia, ressalta-se o menor valor para o número de avaliações da função objetivo requerido pelo algoritmo de BFE.

Tabela 30: Resultados Obtidos para o Problema do Controle Látero-Direcional com Malha Interna Considerando Diferentes Abordagens.

Método	x_{SBRF4} (°)		$f(x_{\text{ótimo}})$ (°)	
	Média	Desvio Padrão	Média	Desvio Padrão
ED	0,215375	$5,319326 \times 10^{-4}$	$1,371712 \times 10^{-4}$	$9,137883 \times 10^{-10}$
RS	0,209172	$4,568565 \times 10^{-3}$	$1,372312 \times 10^{-4}$	$4,271234 \times 10^{-8}$
BFE	0,215234	$7,643167 \times 10^{-5}$	$1,371701 \times 10^{-4}$	$4,472136 \times 10^{-11}$

A Figura 21 apresenta os perfis simulados considerando a melhor solução obtida em cada algoritmo. Para esta aplicação, todos os perfis apresentam comportamentos semelhantes, independentemente da estratégia utilizada. Fisicamente observa-se que o ângulo de rolagem possui uma curva de decaimento suave, sem ocorrência de *overshoot*. Além disso, constata-se que o sistema se acomoda no instante de tempo t igual a 7 s e atinge o regime permanente, para todos os algoritmos, em aproximadamente 60 s.

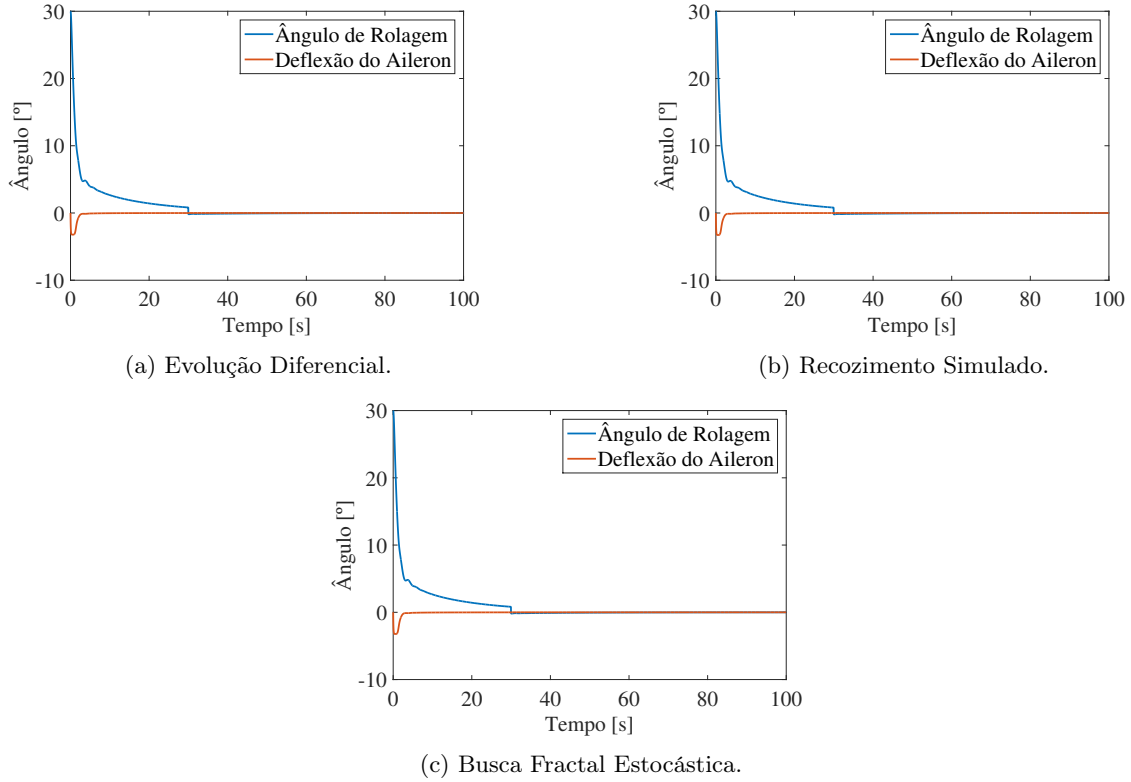


Figura 21: Ângulo de Rolagem e Deflexão do Aileron Considerando Diferentes Abordagens.

• Controle Látero-Direcional com Malha Externa

Nesta última aplicação considera-se a presença de duas variáveis de projeto, a saber, x_{SBRF3} (que determina o ângulo de rolagem de referência ($\Phi_{ref,k}$), parâmetro de entrada da malha interna deste controlador) e x_{SBRF5} (responsável por determinar a deflexão do leme (Δr_k)). Como entrada para esses dois controladores, tem-se o ângulo de guinada (Ψ), cujo valor de referência ($\Psi_{ref,k}$) é definido como 90° no intervalo $0 \leq t \leq 50$ (s) e 0° em $50 < t \leq 100$ (s). Para este estudo de caso não são aplicadas perturbações. A função objetivo considera a minimização do erro RMS referente ao ângulo de guinada ($e(\Psi_k)$). Para formular o problema de otimização, os seguintes domínios são considerado: $0 \leq x_{\text{SBRF3}}, x_{\text{SBRF5}} \leq 5$ ($^\circ$). Para resolver o problema de otimização consideram-se os algoritmos de BFE, DE e RS. Para essa finalidade tem-se os seguintes parâmetros: BFE (tamanho da população (20), número de gerações (10), número de difusões (10) e passo (1)); ED (tamanho da população (20), número de gerações (20), probabilidade de cruzamento (0,95) e fator de perturbação (0,4)) e RS (temperatura inicial (12), número de ciclos de resfriamento (100), número de iterações para cada temperatura (5), desvio padrão da distribuição normal (0,1), decaimento da temperatura (0,9)). Com estes parâmetros, o número de avaliações da função objetivo para as estratégias ED, RS e BFE foram iguais a 420, 1207 e 220, respectivamente. Cada algoritmo foi executado 5 vezes para a obtenção dos resultados apresentados.

A Tabela 31 apresenta os resultados (média e desvio padrão) das variáveis de projeto e função objetivo considerando os algoritmos de BFE, DE e RS. De forma geral observa-se que os resultados obtidos pelo algoritmo de BFE foram muito similares aos obtidos pelos algoritmos de ED e de RS,

pelo menos em termos do valor de função objetivo. Todavia, observa-se uma discrepância entre os valores de ambas as variáveis de projeto no que tange os resultados obtidos usando BFE e ED em relação ao RS. Em termos de desvio padrão, observa-se que o mais robusto foi o algoritmo de BFE. Finalmente, este resultado indica uma baixa sensibilidade no valor da função objetivo em relação à flutuação das variáveis de projeto obtidas, visto que valores próximos de objetivos tem valores muito discrepantes de variáveis de projeto.

Tabela 31: Resultados Obtidos para o Problema do Controle Látero-Direcional com Malha Externa Considerando Diferentes Abordagens.

Método	$x_{\text{SBRF3}} (^{\circ})$		$x_{\text{SBRF5}} (^{\circ})$		$f(x_{\text{ótimo}}) (^{\circ})$	
	Média	Desvio Padrão	Média	Desvio Padrão	Média	Desvio Padrão
ED	1,645775	0,123202	1,055109	0,035009	0,001202	$5,477226 \times 10^{-7}$
RS	4,745630	0,568788	3,666233	1,462398	0,001229	$1,034950 \times 10^{-5}$
BFE	1,547946	0,003895	1,070240	0,002849	0,001201	0,000000

A Figura 22 apresenta os perfis referentes ao ângulo de guinada considerando a melhor solução reportada por cada algoritmo.

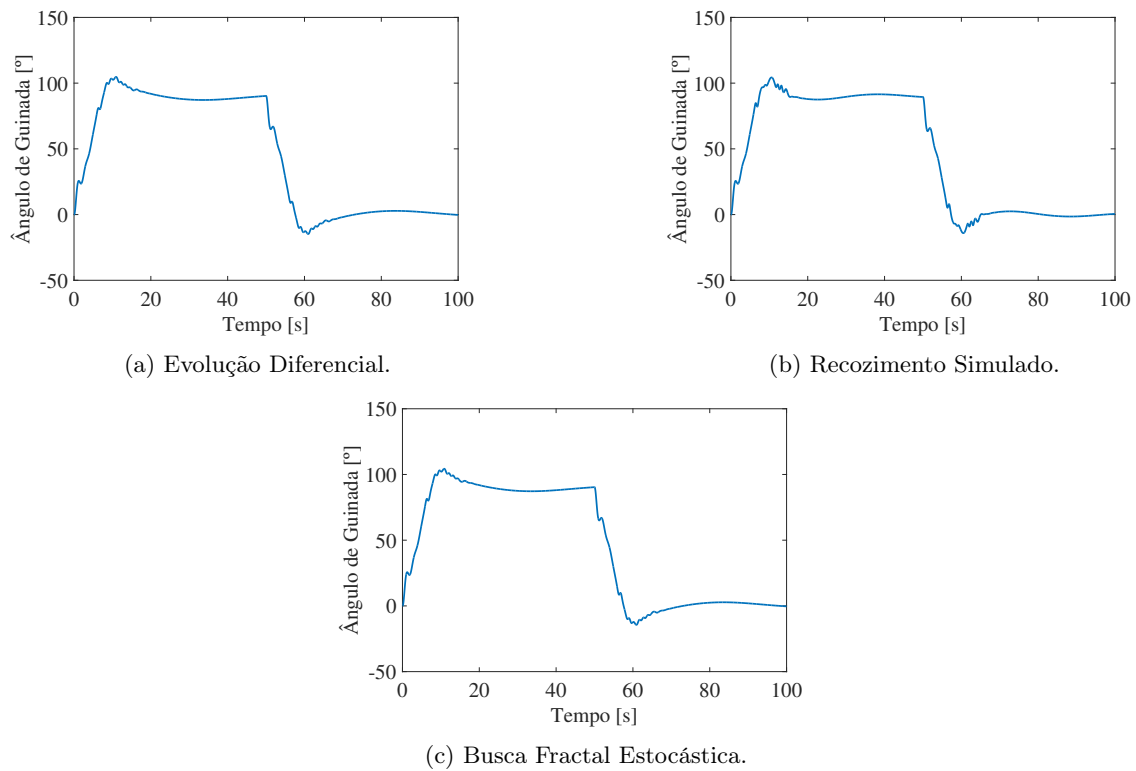


Figura 22: Ângulo de Guinada Considerando Diferentes Abordagens.

Conforme os parâmetros obtidos pelos algoritmos de BFE e ED, os perfis para o ângulo de guinada

são semelhantes. Já o perfil obtido pelo algoritmo de RS apresenta pequenas variações com relação aos outros dois. Para estes algoritmos, existe um *overshoot* em $t = 15$ s e no intervalo compreendido entre $20 \leq t \leq 50$ (s) o valor do ângulo oscila em torno do ponto de referência, atingindo o mesmo em, aproximadamente, $t = 50$ s. Já no intervalo de tempo $50 < t \leq 100$ (s), os perfis se comportam de forma semelhante, porém com concavidade oposta próximo à 100 s. O algoritmo de RS produz um perfil com características de uma curva com oscilações mais intensas que os outros métodos para se estabilizar no ponto de referência desejado. O sistema atinge o regime permanente próximo ao instante $t = 30$ s, permanecendo estável no intervalo $30 \leq t \leq 50$ (s) em torno do ponto de referência de 90° . Já para o intervalo $50 < t \leq 100$ (s), a resposta do ângulo de guinada possui o mesmo comportamento, atingindo a estabilidade em torno de 0° .

Já as Figuras 23 e 24 apresentam as curvas de comportamento das superfícies de controle durante a simulação. De forma geral, as ações de controle apresentam ganho elevado nos instantes de mudança do ponto de referência e são quase nulas próximos dos instantes de estabilidade do sistema, em $t = 30$ s e $t = 70$ s. Além disso, para os três métodos, a ação de controle por meio da deflexão do aileron ocorre de forma semelhante. As curvas de deflexão do leme (Δr_k) são semelhantes para a ED e a BFE, apresentando oscilações mais intensas no perfil obtido pelo RS.

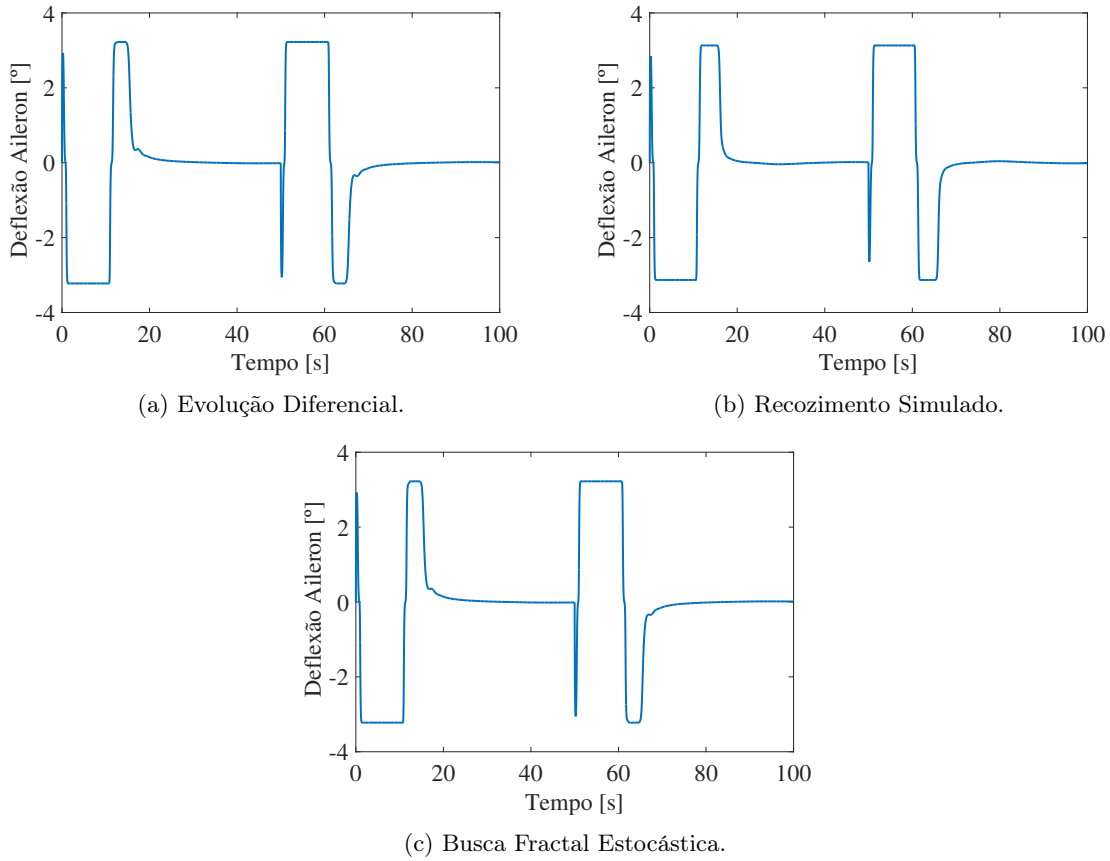


Figura 23: Ângulo de Deflexão do Aileron Considerando Diferentes Abordagens.

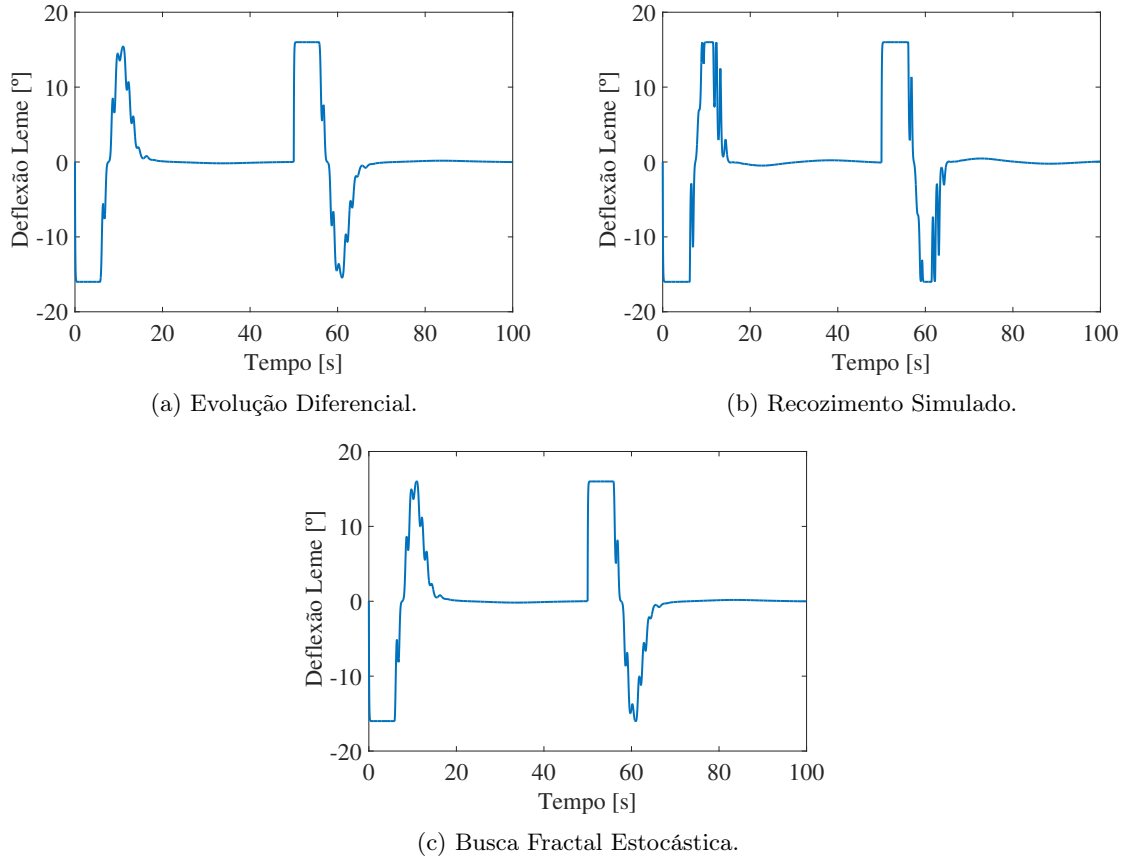


Figura 24: Ângulo de Deflexão do Leme Considerando Diferentes Abordagens.

Para mais detalhes sobre a formulação matemática deste estudo de caso podem ser encontrados em Bernardes (2018).

5.3 Aplicação da Metodologia Proposta para Determinar Todas as Soluções de um Problema de Otimização

Esta seção tem por objetivo apresentar os resultados obtidos a partir da aplicação da metodologia apresentada na seção 3.4 para a determinação de todos as soluções de um problema de otimização em uma única execução.

Para fins de aplicação considere a Função de Himmelblau (HIMMELBLAU, 1972):

$$\min f(x) = (x_1^2 + x_2 - 11)^2 + (x_1 + x_2^2 - 7)^2 \quad (5.19)$$

Esse problema estudo possui a solução ótima global igual a $f(x_{\text{ótimo}}) = 0$, mas quatro combinações diferentes que levem a esta solução, a saber $x_{\text{ótimo},1} = (3,000; 2,000)$, $x_{\text{ótimo},2} = (-2,805; 3,131)$, $x_{\text{ótimo},3} = (-3,779; -3,283)$, $x_{\text{ótimo},4} = (3,584; -1,848)$. Assim, os quatro pontos apresentados tem o mesmo valor no que tange a função objetivo, conforme ilustrado nos gráficos tri-dimensional (Figura 25) e curva de nível (Figura 26), respectivamente.

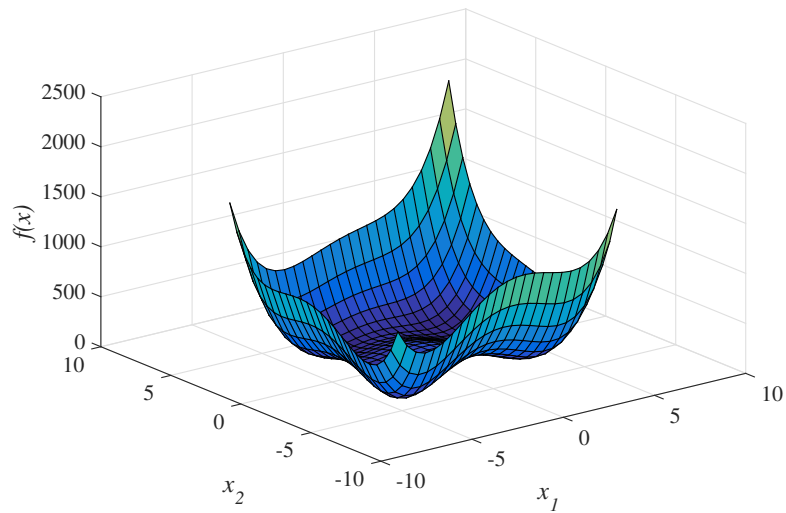


Figura 25: Gráfico da Função de Himmelblau.

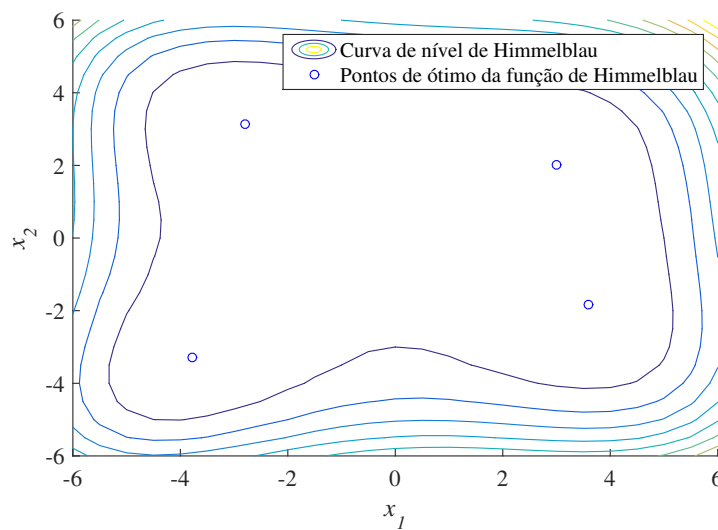


Figura 26: Curvas de Nível para a Função de Himmelblau.

Para aplicar a abordagem proposta anteriormente, considera-se os seguintes parâmetros no algoritmo de BFE: número de gerações (100), tamanho da população (50), número de difusões (2) e passo (1) (o que implica em 5050 avaliações da função objetivo). Além disso, para o tratamento da penalidade, considera-se um fator (r_p) da ordem de 10^8 . Para avaliar o desempenho da referida metodologia, considera-se diferentes valores para o parâmetro Δ_{par} ([0,002 0,005 0,020 0,050 0,100 0,150 0,200 0,350 0,500 0,650 0,800 0,950]). Além disso, foram definidas, previamente, que o algoritmo de BFE seria executado quatro vezes (o mesmo número de soluções do problema em

questão).

Na Tabela 32 são apresentados os resultados obtidos considerando a influência do parâmetro Δ_{par} no número de soluções encontradas automaticamente. Para essa finalidade, são consideradas 5 execuções independentes do algoritmo de BFE.

Tabela 32: Número de Soluções Encontradas pelo Algoritmo de BFE em Função do Parâmetro Δ_{par} .

Δ_{par}	Número de Soluções
0,002	3
0,005	3
0,020	3
0,050	4
0,100	4
0,150	4
0,200	3
0,350	3
0,500	3
0,650	2
0,800	2
0,950	2

Nesta tabela observa-se claramente que o parâmetro Δ_{par} tem papel fundamental no número de soluções encontradas de forma automática pela metodologia proposta. Assim, para valores desse parâmetro iguais a 0,05; 0,10 e 0,15, todas as quatro soluções foram encontradas automaticamente. Para outros valores deste parâmetro foi possível encontrar duas e três das quatro soluções possíveis. Para estes casos, mesmo executando o algoritmo de BFE quatro vezes, somente duas ou três soluções foram encontradas. Na prática isto significa que, para alguns valores do parâmetro Δ_{par} ([0,002 0,005 0,020 0,200 0,350 0,500 0,650 0,800 0,950]) a metodologia proposta encontrou a mesma solução uma ou duas vezes. Esta diferença no número de soluções encontradas se deve à amplitude alcançada pela técnica de penalização proposta. Isto implica que uma escolha apropriada para este parâmetro pode fazer com que a metodologia proposta tenha sucesso em sua busca. Neste caso, para cada nova aplicação, o valor mais interessante deve ser determinado a partir de execuções preliminares.

5.4 Problemas Matemáticos e de Engenharia Multi-objetivos

Nesta seção será avaliada a capacidade do algoritmo de BFE em problemas multi-objetivos. Para esta finalidade será empregado o Método da Soma Ponderada (MSP) para reescrever o problema original (multi-objetivo) em um equivalente (mono-objetivo). Neste caso, para o tratamento de problemas de engenharia, faz-se necessário reescrever o MSP considerando o conceito de solução ideal (VANDERPLAATS, 1999; DEB, 2001). Neste contexto, para problemas bi-objetivos o MSP considerando a ponderação via solução ideal é descrito como:

$$\min f(x) = w \frac{f_1(x)}{f_1^\circ(x)} + (1 - w) \frac{f_2(x)}{f_2^\circ(x)} \quad (5.20)$$

onde w é definido como peso para a soma ponderada, $f_1^\circ(x)$ e $f_2^\circ(x)$ correspondem ao valor da função objetivo otimizada individualmente, mas considerando a presença de todas as restrições originais.

Neste caso, conforme destacado por Vanderplaats (1999), o uso do conceito de solução ideal permite tratar problemas com objetivos e dimensões, bem como com ordens de grandezas distintos.

Para o tratamento de problemas restritos foi utilizado o Método da Penalidade e adotado um fator de penalidade da ordem de 10^{10} .

5.4.1 Função SCH1 de Schaffer

A função SCH1, proposta por Schaffer (1984), é um problema multi-objetivo bastante simples, mas que é muito utilizado para validar algoritmos de otimização devido a sua configuração de espaço com relação aos objetivos (LOBATO, 2008). Este é definido como (SCHAFFER, 1984):

$$\text{SCH1} = \begin{cases} \min f_1(x) = x^2 \\ \min f_2(x) = (x - 2)^2 \end{cases} \quad (5.21)$$

onde x é a variável de projeto definida no intervalo $[-10 \ 10]$. A solução ótima de Pareto para esse problema é obtida quando o espaço de projeto da variável x pertence ao intervalo $[0 \ 2]$.

Para a resolução desse problema, os parâmetros utilizados pelo algoritmo de BFE foram: número de gerações (100), tamanho da população (50), número de difusões (2) e passo (1) (que implica em 5050 avaliações da função objetivo para cada valor de w). Além disso, foi definido um vetor w com 20 posições igualmente espaçadas, no intervalo de $[0 \ 1]$.

A Figura 27 apresenta os resultados obtidos pelo algoritmo de BFE, bem como a curva de Pareto e o espaço de objetivos para esse problema.

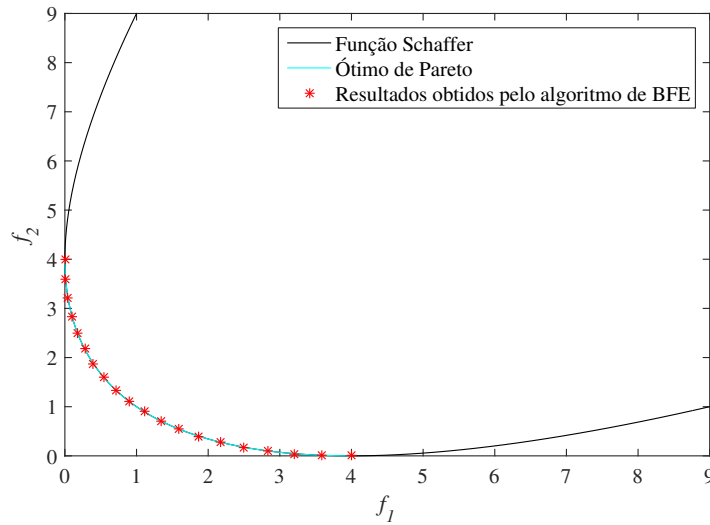


Figura 27: Curva de Pareto para a Função de Schaffer Considerando o BFE e o MSP.

Nesta figura observa-se que a metodologia proposta foi capaz de obter a curva de Pareto para a função SCH1. Além disso, também, pode ser observado uma boa diversidade entre os pontos pertencentes à essa solução.

Cabe enfatizar que, para esta aplicação, devido a mesma ser uma combinação de duas funções puramente matemáticas sem unidades e com a mesma ordem de grandeza, não é necessário usar a definição de solução ideal.

5.4.2 Projeto de uma Viga I

Considere o problema de engenharia apresentado por Castro (2001). Este consiste na determinação do conjunto ótimo de Pareto decorrente da otimização multi-objetivos da viga em I apresentada na Figura 28.

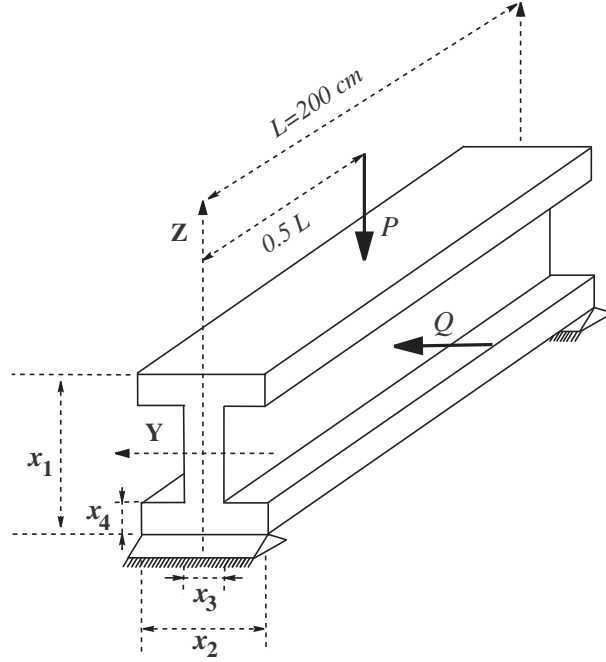


Figura 28: Projeto de uma Viga I.

Sabendo que as variáveis de projeto representam as dimensões da viga ($\vec{x} = [x_1 \ x_2 \ x_3 \ x_4]$), os objetivos a serem minimizados são a área da seção transversal (f_1) e o deslocamento estático máximo (f_2), definidos como (CASTRO, 2001):

$$\begin{cases} \min f_1(x) = 2x_2x_4 + x_3(x_1 - 2x_4) \\ \min f_2(x) = \frac{PL^3}{48EI} \end{cases} \quad (5.22)$$

onde E é o Módulo de Young, P é a carga vertical, L é o comprimento da viga e I é o momento de inércia I , definido como:

$$I = \frac{x_3(x_1 - 2x_4)^3 + 2x_2x_4(4x_4^2 + 3x_1(x_1 - 2x_4))}{12} \quad (5.23)$$

Esse problema está sujeito às seguintes restrições:

$$g_1(x) \equiv \frac{M_Y}{W_Y} + \frac{M_Z}{W_Z} - \sigma_{max} \leq 0 \quad (5.24a)$$

$$10 \text{ cm} \leq x_1 \leq 80 \text{ cm}; \quad 10 \text{ cm} \leq x_2 \leq 50 \text{ cm}; \quad 0,9 \text{ cm} \leq x_3, x_4 \leq 5 \text{ cm} \quad (5.24b)$$

onde σ_{max} é a tensão de projeto da viga, M_Y (30000 kN.cm) e M_Z (2500 kN.cm) são os momentos máximos nas direções Y e Z. W_Y e W_Z são os módulos resistentes da seção nas direções Y e Z, calculados por:

$$W_Y = \frac{x_3(x_1 - 2x_4)^3 + 2x_2x_4(4x_4^2 + 3x_1(x_1 - 2x_4))}{6x_1} \quad (5.25a)$$

$$W_Z = \frac{(x_1 - 2x_4)x_3^3 + 2x_4x_2^3}{6x_2} \quad (5.25b)$$

O problema possui as seguintes propriedades (CASTRO, 2001): $E = 2 \times 10^4$ kN/cm², $\sigma_{max} = 16$ kN/cm², $P = 600$ kN, $Q = 50$ kN e $L = 200$ cm.

Os parâmetros do algoritmo de BFE utilizados na resolução deste problema são: número de gerações (150), tamanho da população (100), número de difusões (2) e passo (1) (que implica em 15100 avaliações da função objetivo para cada valor de w). Para esta aplicação foram definidos 50 valores igualmente espaçados para o parâmetro w no intervalo de [0,1]. Nesta aplicação, devido às diferenças de unidades e de ordem de grandeza, utilizou-se o conceito de solução ideal. Assim, antes da resolução do problema multi-objetivo, foram resolvidos dois problemas mono-objetivos, a saber, um considerando a minimização de f_1 e o outro considerando a minimização de f_2 , cada um com o conjunto original de restrições. Assim, as soluções ideais são calculadas como: $f_1^*(x) = 299,742$ cm² e $f_2^*(x) = 0,0059$ cm, respectivamente. Para fins de comparação, consideram-se os resultados obtidos pelo algoritmo MODE (*Multi-objective Optimization Differential Evolution*) proposto por Lobato (2008) (número de gerações (250), tamanho da população (30), probabilidade de cruzamento (0,85), taxa de perturbação (0,50), número de pseudo-curvas (10) e taxa de redução (0,90), o que implica em 15030 avaliações da função objetivo).

A Figura 29 apresenta os resultados obtidos considerando o algoritmo de BFE associado com o MSP (alguns pontos pré-selecionados desta solução são conforme a Tabela 33) e aqueles obtidos por Lobato (2008) usando o MODE.

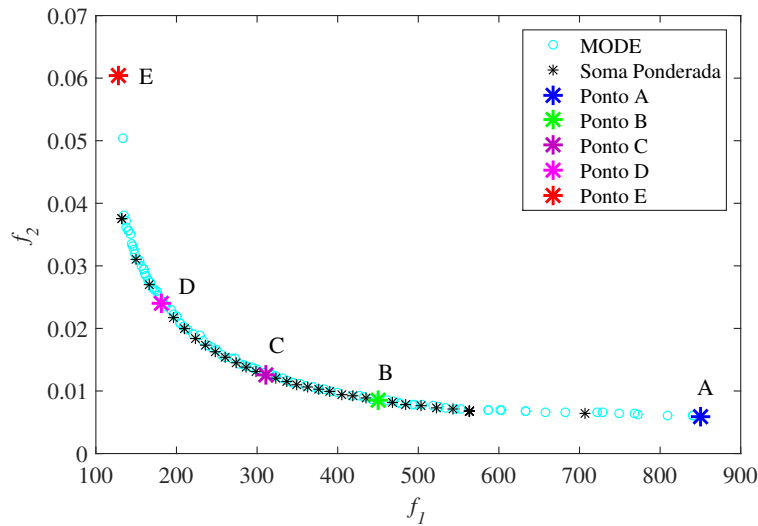


Figura 29: Curva de Pareto para o Problema do Projeto de uma Viga I Considerando Diferentes Abordagens.

Nesta figura observa-se que a metodologia proposta foi capaz de obter uma boa aproximação para a curva de Pareto em comparação com o algoritmo MODE. Todavia, ressalta-se que, devido à natureza de ambos os algoritmos (o MODE faz uso do critério de dominância de Pareto e a proposta neste trabalho não), o número de avaliações requeridas pelo algoritmo de BFE é muito superior (50 vezes 15100) ao do algoritmo MODE (15030). Todavia, é importante ressaltar que este elevado custo computacional não é apenas do algoritmo de BFE, mas sim de toda e qualquer abordagem heurística baseada em população que não faz uso do critério de dominância de Pareto para a resolução de problemas multi-objetivos. Finalmente, do ponto de vista matemático, ambos os pontos apresentados nesta figura tem a mesma importância, isto é, todos constituem a solução do problema de otimização multi-objetivo.

Na Tabela 33 observa-se que, após a obtenção da curva de Pareto, o usuário pode definir quais pontos devem ser implementados na prática. Esta análise de pós-processamento pode ser definida de acordo com um critério adicional adotado, como por exemplo, o valor médio e que represente um bom compromisso entre ambos os objetivos simultaneamente.

Tabela 33: Resultados Obtidos pelo Algoritmo de BFE para o Problema do Projeto de Viga I.

Parâmetro	Ponto A	Ponto B	Ponto C	Ponto D	Ponto E
w (-)	0	0,489795	0,693878	0,897959	1
x_1 (cm)	80,000000	80,000000	80,000000	80,000000	60,647007
x_2 (cm)	49,999999	50,000000	49,999997	49,999977	41,360180
x_3 (cm)	5,000000	0,900000	0,900000	0,900000	0,900013
x_4 (cm)	5,000000	3,857583	2,431667	1,116001	0,900069
g_1 (kN/cm ²)	-13,987545	-13,181295	-11,749099	-7,567983	-0,000492
f_1 (cm ²)	849,999975	450,814677	310,789709	181,591284	127,416999
f_2 (cm)	0,005903	0,008504	0,012572	0,023935	0,061174

5.4.3 Projeto de uma Viga Soldada

Considere a viga soldada conforme ilustrado na Figura 30.

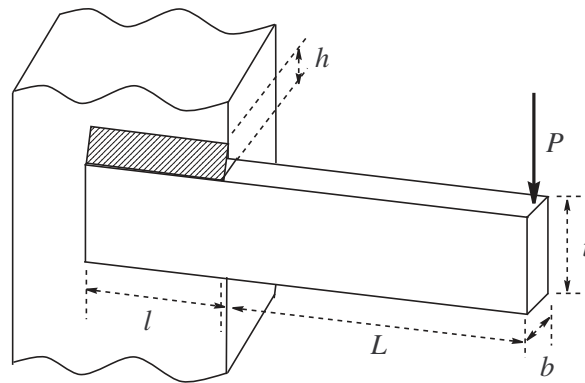


Figura 30: Projeto de uma Viga Soldada.

Esta é submetida a uma força P em sua extremidade, necessitando satisfazer às condições de

estabilidade e limitações de projeto. Nesta aplicação tem-se como objetivos a minimização do custo da viga (f_1) e do deslocamento da extremidade livre da viga (f_2) (CASTRO, 2001).

Considerando que as variáveis de projeto são as dimensões da viga e que as mesmas apresentam a seguinte notação ($\vec{x} = [x_1 \ x_2 \ x_3 \ x_4] = [h \ l \ t \ b]$) (in), matematicamente este estudo de caso pode ser definido como (CASTRO, 2001):

$$\begin{cases} \min f_1(x) = 1,10471x_1^2x_2 + 0,04811x_3x_4(L + x_2) \\ \min f_2(x) = \frac{4PL^3}{x_3^3x_4E} \end{cases} \quad (5.26)$$

sujeito às seguintes restrições:

$$g_1(x) \equiv \tau - \tau_{max} \leq 0 \quad (5.27a)$$

$$g_2(x) \equiv \sigma - \sigma_{max} \leq 0 \quad (5.27b)$$

$$g_3(x) \equiv P - P_c \leq 0 \quad (5.27c)$$

$$g_4(x) \equiv \frac{4PL^3}{x_3^3x_4E} - u_{max} \leq 0 \quad (5.27d)$$

$$g_5(x) \equiv x_1 - x_4 \leq 0 \quad (5.27e)$$

$$0,125 \leq x_1, x_4 \leq 5,0; \quad 0,1 \leq x_2, x_3 \leq 10,0 \quad (5.27f)$$

onde:

$$\tau = \sqrt{\tau'^2 + \tau''^2 + \frac{x_2\tau'\tau''}{\sqrt{0,25(x_2^2 + (x_1 + x_3)^2)}}} \quad (5.28a)$$

$$\tau' = \frac{6000}{\sqrt{2}x_1x_2} \quad (5.28b)$$

$$\tau'' = \frac{6000(14 + 0,5x_2)\sqrt{0,25(x_2^2 + (x_1 + x_3)^2)}}{2\left(0,707x_1x_2\left(\frac{x_2^2}{12} + 0,25(x_1 + x_3)^2\right)\right)} \quad (5.28c)$$

$$\sigma = \frac{504000}{x_3^2x_4} \quad (5.28d)$$

$$P_c = 64746,022(1 - 0,0282346x_3)x_3x_4^3 \quad (5.28e)$$

em que $P = 6000$ lb (força aplicada), $L = 14$ in (comprimento em balanço da viga), $E = 30 \times 10^6$ psi (módulo de elasticidade), $\tau_{max} = 13600$ psi (tensão máxima de cisalhamento), $\sigma_{max} = 30000$ psi (tensão máxima de flexão) e $u_{max} = 0,25$ in (limite máximo para o deslocamento da extremidade da viga).

Os parâmetros do algoritmo de BFE utilizados na resolução deste problema são: número de gerações (150), tamanho da população (100), número de difusões (2) e passo (1) (que implica em 15100 avaliações da função objetivo para cada valor de w). Para esta aplicação foram definidos 20 valores igualmente espaçados para o parâmetro w no intervalo de $[0,1]$. Nesta aplicação, as soluções ideais foram calculadas como: $f_1^*(x) = 1,397$ \$ e $f_2^*(x) = 0,000439$ in, respectivamente. Também nesta aplicação foi considerado os resultados obtidos pelo algoritmo MODE (LOBATO, 2008) (número de gerações (250), tamanho da população (50), probabilidade de cruzamento (0,85),

taxa de perturbação (0,50), número de pseudo-curvas (10) e taxa de redução (0,90), o que implica em 25050 avaliações da função objetivo).

A Figura 31 apresenta os resultados obtidos pelo algoritmo de BFE associado com o MSP (alguns pontos pré-selecionados desta solução são conforme a Tabela 34) e aqueles obtidos por Lobato (2008) usando o MODE. Assim como no estudo de caso anterior, nesta figura observa-se que a metodologia proposta foi capaz de obter uma boa aproximação para a curva de Pareto em comparação com o algoritmo MODE. Todavia, devido à natureza do algoritmo de BFE (não faz uso do critério de dominância de Pareto), o número de avaliações requeridas pela metodologia proposta é muito superior (20 vezes 15100) ao do algoritmo MODE (25050). Conforme já explicado, este elevado custo computacional já era esperado e não é uma característica específica do algoritmo de BFE, mas sim de toda e qualquer abordagem heurística baseada em população que não faz uso do critério de dominância de Pareto para a resolução de problemas multi-objetivos.

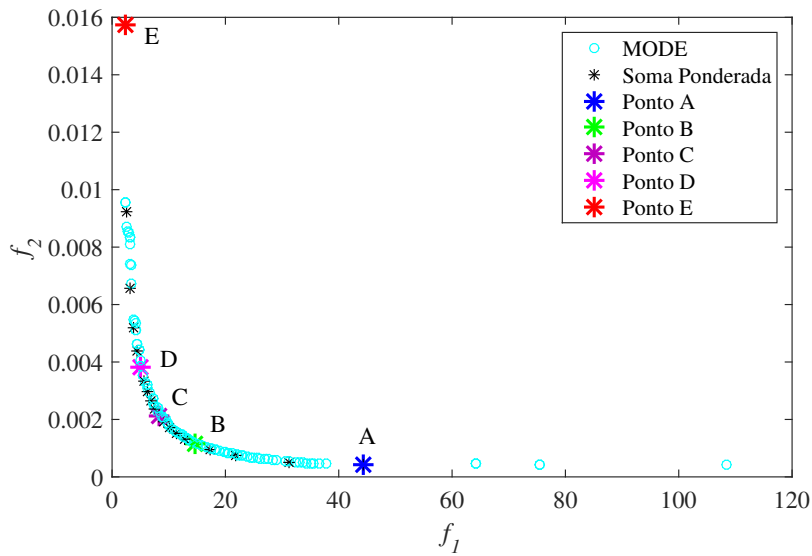


Figura 31: Curva de Pareto para o Problema do Projeto de uma Viga Soldada Considerando Diferentes Abordagens.

Na Tabela 34 observa-se alguns pontos pertencentes à curva de Pareto. Qualquer um destes pode ser escolhido pelo usuário de modo a ser implementado na prática.

Até o presente momento foram apresentados os resultados obtidos com a aplicação do algoritmo de BFE original (canônico) em problemas com diferentes níveis de complexidade. De forma geral foi observado que os resultados obtidos pelo referido algoritmo considerando os parâmetros constantes são promissores. A partir deste ponto são apresentadas aplicações considerando a associação entre o algoritmo canônico e os operadores descritos na seção 3.3. Conforme descrito anteriormente, estes operadores tem como objetivo aumentar a diversidade da população de candidatos à solução do problema de otimização ao mesmo tempo que deseja-se também a redução do custo computacional.

Tabela 34: Resultados Obtidos pelo Algoritmo de BFE para o Problema do Projeto de Viga Soldada.

Parâmetro	Ponto A	Ponto B	Ponto C	Ponto D	Ponto E
w (-)	0	0,21052	0,47368	0,73684	1
x_1 (in)	3,83139	1,04452	0,77952	0,57627	0,24438
x_2 (in)	6,53145	0,85773	1,19859	1,69889	6,21467
x_3 (in)	10,00000	9,99999	9,99999	9,99993	8,29588
x_4 (in)	5,00000	1,91294	1,03163	0,57669	0,24439
g_1 (psi)	$-1,30692 \times 10^4$	-0,19166	-0,85339	-0,57546	-1,22644
g_2 (psi)	$-2,89920 \times 10^4$	$-2,73653 \times 10^4$	$-2,51145 \times 10^4$	$-2,12604 \times 10^4$	-34,56649
g_3 (lb)	$-5,80755 \times 10^7$	$-3,24663 \times 10^6$	$-5,04155 \times 10^5$	$-8,31174 \times 10^4$	-3,82536
g_4 (in)	-0,24956	-0,24885	-0,24787	-0,24619	-0,23426
g_5 (in)	-1,16861	-0,86841	-0,25211	$-4,15258 \times 10^{-4}$	$-7,39515 \times 10^{-6}$
f_1 (\$)	155,30600	14,70763	8,34793	4,97886	2,38176
f_2 (in)	0,00043	0,00114	0,00212	0,00380	0,01573

Inicialmente, cada um dos operadores propostos serão estudados separadamente de forma a avaliar a sua influência na qualidade da solução encontrada, bem como no número de avaliações da função objetivo. Para essa finalidade, conforme destacado anteriormente, o tratamento de problemas com restrição será realizado considerando o Método da Penalização Estática com fator de penalidade da ordem de 10^{10} . Ressalta-se que a formulação matemática dos problemas analisados não serão repetidos aqui, visto que os mesmos já foram apresentados anteriormente. Finalmente, enfatiza-se que os resultados obtidos serão comparados somente com aqueles reportados pela versão canônica, visto que um comparativo com outras estratégias de otimização já foi realizado e discutido anteriormente.

5.5 Atualização do Número de Difusões e do Passo da Caminhada Gaussiana

Nesta seção o operador proposto para a atualização dinâmica do número de difusões e do passo da caminhada Gaussiana no algoritmo de BFE será avaliado. Assim, são consideradas as versões denominadas de BFEA - Busca Fractal Estocástica Aleatória (a variação dos parâmetros é realizada de forma aleatória); BFEML - Busca Fractal Estocástico baseado em Mapa Logístico (a variação dos parâmetros é realizada conforme o modelo caótico logístico dado pela Equação (3.20) e BFEMS - Busca Fractal Estocástico baseado no Mapa Seno (a variação dos parâmetros é realizada o modelo caótico de busca do tipo seno dado pela Equação (3.21). Para todos os problemas abordados nesta seção foram realizadas 10 execuções independentes de cada estratégia. Para avaliar os números de gerações (N_{ger}) e de avaliações da função objetivo (N_{eval}), definiu-se como critério de parada em cada um dos algoritmos o módulo do erro absoluto entre o valor ótimo reportado pela literatura e o valor obtido por cada algoritmo. Neste caso, se esse valor for menor do que 10^{-6} , o processo de otimização é finalizado, caso contrário o processo continua até que um número máximo de gerações seja alcançado. Para avaliar a redução percentual (%) de cada estratégia proposta em relação ao algoritmo canônico em cada aplicação, definiu-se o parâmetro Γ como sendo:

$$\Gamma = 100 \left(1 - \frac{N_{eval,p}}{N_{eval,c}} \right) \quad (5.29)$$

em que $N_{eval,p}$ e $N_{eval,c}$ representam os números de avaliações da função objetivo referentes a p -ésima estratégia proposta e ao algoritmo canônico, respectivamente.

Os parâmetros constantes ou utilizados para a inicialização em cada uma das estratégias em cada uma das aplicações são definidas como segue:

- Função de Beale (Seção 5.1.1): Solução ótima ($f(x_{ótimo}) = 0$). Parâmetros: tamanho da população (30), número de difusões (2 para o BFE e 2 para inicializar o BFEA, BFEML e o BFEMS) e passo (1 para o BFE e 0,1 para inicializar o BFEA, BFEML e o BFEMS). O parâmetro μ nos algoritmos BFEML e BFEMS foram definidos como sendo iguais a 3,6 e 0,1, respectivamente.
- Função de Rastrigin (Seção 5.1.2): Solução ótima ($f(x_{ótimo}) = 0$). Parâmetros: tamanho da população (30), número de difusões (2 para o BFE e 2 para inicializar o BFEA, BFEML e o BFEMS) e passo (1 para o BFE e 0,1 para inicializar o BFEA, BFEML e o BFEMS). O parâmetro μ nos algoritmos BFEML e BFEMS foi definido como sendo igual a 4.
- Projeto de uma Mola de Tensão/Compressão (Seção 5.2.1): Solução ótima ($f(x) = 0,012665$). Parâmetros: tamanho da população (30), número de difusões (3 para o BFE e 3 para inicializar o BFEA, BFEML e o BFEMS) e passo (1 para o BFE e 0,1 para inicializar o BFEA, BFEML e o BFEMS). O parâmetro μ nos algoritmos BFEML e BFEMS foram definidos como sendo iguais a 3,6 e 4, respectivamente.
- Projeto de uma Viga Soldada (Seção 5.2.2): Solução ótima ($f(x) = 1,519802$). Parâmetros: tamanho da população (35), número de difusões (4 para o BFE e 4 para inicializar o BFEA, BFEML e o BFEMS) e passo (1 para o BFE e 0,1 para inicializar o BFEA, BFEML e o BFEMS). O parâmetro μ nos algoritmos BFEML e BFEMS foram definidos como sendo iguais a 3,6 e 2, respectivamente.
- Projeto de uma Coluna Tubular Sujeita a uma Carga Compressiva (Seção 5.2.4): Solução ótima ($f(x) = 26,499498$). Parâmetros: tamanho da população (30), número de difusões (4 para o BFE e 4 para inicializar o BFEA, BFEML e o BFEMS) e passo (1 para o BFE e 0,1 para inicializar o BFEA, BFEML e o BFEMS). O parâmetro μ nos algoritmos BFEML e BFEMS foram definidos como sendo iguais a 4 e 1, respectivamente.
- Projeto de uma Barra Cantilever (Seção 5.2.5): Solução ótima ($f(x) = 1,339969$). Parâmetros: tamanho da população (50), número de difusões (4 para o BFE e 4 para inicializar o BFEA, BFEML e o BFEMS) e passo (1 para o BFE e 0,1 para inicializar o BFEA, BFEML e o BFEMS). O parâmetro μ nos algoritmos BFEML e BFEMS foram definidos como sendo iguais a 3,7 e 1, respectivamente.

A Tabela 35 apresenta os resultados obtidos considerando as estratégias propostas, bem como o algoritmo canônico. Em primeiro lugar, nesta tabela observa-se que todas as abordagens foram capazes de encontrar boas aproximações para a solução ótima em cada um dos problemas analisados. Além disso, quase todas as abordagens que fazem uso de algum tipo de atualização para os parâmetros N_{dif} e N_{passo} apresentaram melhor desempenho (redução do número de avaliações da função objetivo). A única exceção foi a estratégia BFEMS para a Função de Rastrigin, que resultou em um aumento de 11,8% para o parâmetro N_{eval} em relação ao algoritmo canônico. Com relação ao número de gerações para as abordagens propostas, o mesmo tende a alcançar um valor menor do que os observados para a versão canônica. Em relação às estratégias propostas, constata-se que o algoritmo BFEML se destaca por ter o menor valor do N_{eval} . Proporcionalmente, observa-se, para

cada estudo de caso, uma redução (Γ) de 54,6%, 47,1%, 66,1%, 22,3%, 32,3% e 55,3% em relação ao BFE, isto é; esta abordagem foi a que apresentou o melhor custo benefício em termos de convergência e número de avaliações da função objetivo.

Tabela 35: Resultados Obtidos pelos Algoritmos BFEA, BFEML e BFEMS para os Estudos de Caso.

		BFE	BFEA	BFEML	BFEMS
Bale	N_{ger} (Média)	75	50	42	60
	N_{ger} (Desvio)	6	3	5	6
	N_{ger} (Ótimo)	65	47	29	48
	N_{eval} (Ótimo)/ Γ (%)	1980/-	1440/27,3	900/54,6	1470/25,6
	$f(x)$	0			
Rastrigin	N_{ger} (Média)	19	17	12	23
	N_{ger} (Desvio)	2	3	2	4
	N_{ger} (Ótimo)	16	12	8	18
	N_{eval} (Ótimo)/ Γ (%)	510/-	390/23,5	270/47,1	570/11,8
	$f(x)$	0			
Mola	N_{ger} (Média)	245	645	219	258
	N_{ger} (Desvio)	35	690	142	121
	N_{ger} (Ótimo)	232	137	78	131
	N_{eval} (Ótimo)/ Γ (%)	6990/-	4140/40,8	2370/66,1	3960/43,4
	$f(x)$	0,012665			
Viga	N_{ger} (Média)	244	222	202	244
	N_{ger} (Desvio)	15	10	17	16
	N_{ger} (Ótimo)	224	206	174	216
	N_{eval} (Ótimo)/ Γ (%)	7875/-	7245/8	6125/22,3	7595/3,6
	$f(x)$	1,519802			
Coluna	N_{ger} (Média)	105	84	73	92
	N_{ger} (Desvio)	8	5	7	5
	N_{ger} (Ótimo)	89	72	60	84
	N_{eval} (Ótimo)/ Γ (%)	2700/-	2190/18,9	1830/32,3	2550/5,6
	$f(x)$	26,499498			
Cantilever	N_{ger} (Média)	188	117	102	147
	N_{ger} (Desvio)	21	16	23	11
	N_{ger} (Ótimo)	151	77	67	129
	N_{eval} (Ótimo)/ Γ (%)	7600/-	3900/48,7	3400/55,3	6500/14,5
	$f(x)$	1,339969			

5.6 Atualização do Tamanho da População

Nessa seção são apresentados os resultados obtidos para o algoritmo de BFE canônico e para a sua versão com o tamanho da população sendo atualizado dinamicamente, denominada de BFEVP (Busca Fractal Estocástica com Variação da População), conforme apresentado na seção 3.3.2. Neste caso, o número de difusões e passo são considerados fixos durante todo o processo de otimização. Para este estudo também foi utilizado o mesmo método para o tratamento de restrições e o mesmo critério para finalizar os algoritmos considerados. A redução percentual do número de avaliações da

função objetivo também é considerada, conforme a Equação (5.29).

Os parâmetros constantes ou utilizados para a inicialização em cada uma das estratégias em cada uma das aplicações são definidas como segue:

- Função de Beale (Seção 5.1.1): Solução ótima ($f(x_{\text{ótimo}}) = 0$). Parâmetros: tamanho da população (30 para o BFE e variando entre 10 e 20 para o BFEVP), número de difusões (2) e passo (1).
- Função de Rastrigin (Seção 5.1.2): Solução ótima ($f(x_{\text{ótimo}}) = 0$). Parâmetros: tamanho da população (30 para o BFE e variando entre 10 e 20 para o BFEVP), número de difusões (2) e passo (1).
- Projeto de uma Mola de Tensão/Compressão (Seção 5.2.1): Solução ótima ($f(x) = 0,012665$). Parâmetros: tamanho da população (30 para o BFE e variando entre 10 e 20 para o BFEVP), número de difusões (3) e passo (1).
- Projeto de uma Viga Soldada (Seção 5.2.2): Solução ótima ($f(x) = 1,519802$). Parâmetros: tamanho da população (35 para o BFE e variando entre 20 e 30 para o BFEVP), número de difusões (4) e passo (1).
- Projeto de uma Coluna Tubular Sujeita a uma Carga Compressiva (Seção 5.2.4): Solução ótima ($f(x) = 26,499498$). Parâmetros: tamanho da população (30 para o BFE e variando entre 10 e 20 para o BFEVP), número de difusões (4) e passo (1).
- Projeto de uma Barra Cantilever (Seção 5.2.5): Solução ótima ($f(x) = 1,339969$). Parâmetros: tamanho da população (50 para o BFE e variando entre 20 e 40 para o BFEVP), número de difusões (4) e passo (1).

A Tabela 36 apresenta os resultados obtidos considerando os algoritmos BFE e BFEVP. Nesta tabela observa-se que todas as abordagens foram capazes de encontrar boas aproximações para a solução ótima para cada um dos problemas analisados. Todavia, a abordagem BFEVP sempre apresentou melhor desempenho, em termos do número de avaliações da função objetivo, em relação ao BFE. Neste caso tem-se uma redução (Γ) de 12,2%, 5,9%, 54,6%, 32,9%, 22,5% e 25,6% em relação ao BFE, demonstrando o melhor custo benefício em termos de convergência e número de avaliações da função objetivo. Este resultado já era esperado visto que o algoritmo original não apresenta nenhuma estratégia para a redução dinâmica do tamanho da população. Assim, mesmo que a estratégia proposta para a redução do tamanho da população não leve em conta o andamento do processo evolutivo, considera-se que a mesma foi eficiente em comparação com o algoritmo canônico.

5.7 Atualização do Número de Difusões, do Passo da Caminhada Gaussiana e do Tamanho da População

Conforme observado anteriormente, a aplicação individual de cada um dos operações propostos resultou, em praticamente todos os estudos de caso, na redução do custo computacional, além de todos terem convergido para a melhor solução reportada na literatura. Neste contexto, parece justo combinar os mesmos para avaliar, efetivamente, a qualidade dos resultados. Esta nova proposta, denominada de algoritmo BFEM (Busca Fractal Estocástica Melhorada), combina os operadores apresentados para a atualização dinâmica do número de difusões, do passo da caminhada Gaussiana e do tamanho da população, conforme apresentado e discutido na seção 3.3. Para esta finalidade também foi utilizado o método da penalidade para o tratamento de restrições e o mesmo critério

para finalizar os algoritmos considerados. A redução percentual do número de avaliações da função objetivo também é considerada, conforme a Equação (5.29).

A seguir são apresentadas as informações consideradas em cada um dos estudos de caso analisados.

Tabela 36: Resultados Obtidos pelo Algoritmo BFEVP para os Estudos de Caso.

		BFE	BFEVP
Bale	N_{eval} (Média)	2244	1752
	N_{eval} (Desvio)	216	69
	N_{eval} (Ótimo)/ Γ (%)	1890/-	1660/12,2
	$f(x)$	0	
Rastrigin	N_{eval} (Média)	570	555
	N_{eval} (Desvio)	57	49
	N_{eval} (Ótimo)/ Γ (%)	510/-	480/5,9
	$f(x)$	0	
Mola	N_{eval} (Média)	8124	5366
	N_{eval} (Desvio)	1624	3566
	N_{eval} (Ótimo)/ Γ (%)	6330/-	2880/54,6
	$f(x)$	0,012665	
Viga	N_{eval} (Média)	8351	5519
	N_{eval} (Desvio)	491	313
	N_{eval} (Ótimo)/ Γ (%)	7525/-	5055/32,9
	$f(x)$	1,519802	
Coluna	N_{eval} (Média)	3085	2167
	N_{eval} (Desvio)	339	96
	N_{eval} (Ótimo)/ Γ (%)	2580/-	2000/22,5
	$f(x)$	26,499498	
Cantilever	N_{eval} (Média)	5810	4088
	N_{eval} (Desvio)	525	363
	N_{eval} (Ótimo)/ Γ (%)	4650/-	3460/25,6
	$f(x)$	1,339969	

- Projeto de uma Viga Soldada (Seção 5.2.2): Solução ótima ($f(x)=1,519802$). Parâmetros: tamanho da população (35 para o BFE e variando entre 20 e 30 para o BFEM), número de difusões (4 para o BFE e 0,1 para inicializar o BFEM) e passo (1 para o BFE e 0,1 para inicializar o BFEM). O parâmetro μ no algoritmo BFEM - usando o mapa logístico - foi definido como sendo igual a 3,6.
- Projeto de uma Coluna Tubular Sujeita a uma Carga Compressiva (Seção 5.2.4): Solução ótima ($f(x)=26,499498$). Parâmetros: tamanho da população (30 para o BFE e variando entre 10 e 20 para o BFEM), número de difusões (4 para o BFE e 0,1 para inicializar o BFEM) e passo (1 para o BFE e 0,1 para inicializar o BFEM). O parâmetro μ no algoritmo BFEM - usando o mapa logístico - foi definido como sendo igual a 4.
- Projeto de uma Barra Cantilever (Seção 5.2.5): Solução ótima ($f(x)=1,339969$). Parâmetros: tamanho da população (50 para o BFE e variando entre 20 e 40 para o BFEM), número de difusões (4 para o BFE e 0,1 para inicializar o BFEM) e passo (1 para o BFE e 0,1 para

inicializar o BFEM). O parâmetro μ no algoritmo BFEM - usando o mapa logístico - foi definido como sendo igual a 3,7.

A Tabela 37 apresenta os resultados obtidos para os estudos de caso considerando os algoritmos BFE e BFEM. Nesta tabela observa-se que, assim como para as estratégias considerando os operadores separadamente, o BFEM sempre foi capaz de convergir para a melhor solução, bem como o BFE canônico. Todavia, para cada uma das aplicações, observa-se uma redução da ordem de, aproximadamente, 41,4%, 33,4% e 48,4% do BFEM com relação à sua versão canônica em termos do número de avaliações da função objetivo. Estes menores valores também estão relacionados com os números de gerações apresentados nesta tabela, isto é; quanto menor o número de gerações, menor é o número de avaliações. Tal resultado está em concordância com o esperado e reportado por Lobato & Steffen Jr (2016), visto que a atualização nos parâmetros faz com que, a priori, a diversidade da população seja aumentada, o que implica no incremento da probabilidade de se encontrar a solução ótima. Além disso, quando se reduz o tamanho da população ao longo das gerações, o custo computacional também é reduzido.

Tabela 37: Resultados Obtidos pelo Algoritmo BFEM para os Estudos de Caso.

		BFE	BFEM
Viga	N_{eval} (Média)	8351	4943
	N_{eval} (Desvio)	491	301
	N_{eval} (Ótimo)/ Γ (%)	7525/-	4415/41,4
	$f(x)$	1,519802	
Coluna	N_{eval} (Média)	3085	1803
	N_{eval} (Desvio)	339	82
	N_{eval} (Ótimo)/ Γ (%)	2580/-	1720/33,4
	$f(x)$	26,499498	
Cantilever	N_{eval} (Média)	5810	2884
	N_{eval} (Desvio)	525	288
	N_{eval} (Ótimo)/ Γ (%)	4650/-	2400/48,4
	$f(x)$	1,339969	

5.8 Consolidado dos Resultados

5.8.1 Algoritmo BFE

A partir dos resultados apresentados neste capítulo pode-se concluir que o algoritmo de BFE foi capaz de encontrar boas aproximações para cada problema de otimização (mono e multi-objetivo) analisado. No caso dos problemas mono-objetivos, os resultados foram encontrados com boa precisão visto o valor médio e do desvio padrão reportados para o vetor de variáveis de projeto e para a função objetivo. Cabe ressaltar que, para os problemas mono-objetivos, o algoritmo de BFE também foi interessante do ponto de vista do custo computacional, já que foi possível obter a mesma qualidade de solução com um número menor de avaliações da função objetivo com relação a outras estratégias heurísticas.

Por outro lado, devido à natureza da metodologia considerada para a resolução de problemas multi-objetivos (não foi considerado o critério de dominância de Pareto), o custo computacional ao se usar o Método da Soma Ponderada resulta em um número bem superior de avaliações da função objetivo. Enfatiza-se que esta aparente desvantagem pode ser ultrapassada considerando-se o

desenvolvimento de uma estratégia baseada no acoplamento entre o algoritmo de BFE e o critério de dominância de Pareto. Cabe ressaltar que tal desvantagem é inerente a qualquer método heurístico baseado em população associado ao Método da Soma Ponderada.

Ressalta-se que, para cada aplicação apresentada neste capítulo, um conjunto específico de parâmetros foi utilizado. Cada um foi definido a partir de execuções preliminares do algoritmo de BFE de forma que o binômio convergência versus custo computacional fosse contemplado.

5.8.2 Algoritmo BFE com Atualização dos Parâmetros

No final deste capítulo foram avaliados os operadores propostos para aumentar a diversidade da população e redução do custo computacional. Neste contexto, a Tabela 38 apresenta um consolidado dos resultados obtidos considerando a associação entre o algoritmo de BFE com os operadores propostos para a atualização de seus parâmetros. De forma geral observa-se que, na média, as versões do algoritmo de BFE propostos foram capazes de reduzir o número de avaliações da função objetivo ao mesmo tempo que a solução ótima foi encontrada. Particularmente para os operadores para a atualização dinâmica do número de difusões e do tamanho do passo da caminhada Gaussiana, os resultados foram bastante interessantes, pois para cada aplicação, pelo menos uma destas versões foi mais eficiente no que tange a redução do custo computacional com relação à versão canônica. Neste contexto, destaca-se o desempenho do algoritmo de BFEML, pois este sempre foi capaz de reduzir o custo computacional com relação à versão canônica.

Com relação à variação do tamanho da população observa-se que esta versão sempre foi, na média, superior ao número requerido pelo algoritmo de BFE. Este resultado já era esperado visto que o algoritmo original não apresenta nenhuma estratégia similar a que foi apresentada e utilizada aqui, isto é; o tamanho da população é reduzido linearmente a partir da metade do número de gerações.

Ao se considerar a associação de ambos os operadores ao algoritmo canônico (estratégia denominada de BFEM) observa-se, para os estudos de caso analisados, que o seu desempenho foi, no mínimo, equivalente aos reportados pelo BFEML e superiores às outras abordagens. Tal resultado já era esperado visto que ao mesmo tempo que esse algoritmo aumenta a capacidade de exploração do espaço de busca, também prioriza-se a redução do tamanho da população, permitindo desta forma que o binômio convergência versus custo computacional seja atendido. Assim, considera-se que a versão BFEM tem potencial para ser utilizada como ferramenta de otimização em problemas com maior complexidade.

Tabela 38: Número de Avaliações da Função Objetivo Considerando os Algoritmos BFE, BFEA, BFEML, BFEMS, BFEVP e BFEM.

	BFE	BFEA	BFEML	BFEMS	BFEVP	BFEM
Beale	1890	1440	900	1470	1660	-
Rastrigin	510	390	270	570	480	-
Mola	6330	4140	2370	3960	2880	-
Viga	7525	7245	6125	7595	5055	4415
Coluna	2580	2190	1830	2550	2000	1850
Cantilever	4650	3900	3400	6500	3460	2400

6 Conclusões e Trabalhos Futuros

6.1 Conclusões

Esta dissertação teve como objetivo propor avaliar a capacidade do algoritmo de BFE, bem como a sua associação com novos operadores para a resolução de problemas de otimização com diferentes níveis de complexidade. Para esta finalidade, discutiu-se aspectos relacionados aos operadores, bem como sobre as métricas de comparação utilizadas. A partir dos estudos analisados, as seguintes conclusões são apresentadas:

- No Capítulo 4 foi analisada a influência dos parâmetros do algoritmo de BFE empregando para essa finalidade funções matemáticas multi-modais com e sem restrições. Os resultados obtidos evidenciaram que o aumento no tamanho da população provoca uma melhora significativa no desempenho do algoritmo, conforme observado para outras estratégias heurísticas. Este mesmo princípio também pode ser observado no que tange o número de gerações, isto é; quanto maior o seu valor, maior é a chance que o algoritmo encontre uma região em que, provavelmente, a solução ótima está localizada. Todavia, o aumento do tamanho da população e do número de gerações implica no incremento do custo computacional. Já o número de difusões e o tamanho do passo da caminhada Gaussiana são parâmetros cuja faixa de aplicabilidade, pelo menos para os casos analisados, se mostraram mais flexíveis. Neste contexto, pode-se sugerir os seguintes parâmetros para uma nova aplicação: população com 50 indivíduos, 500 gerações, número de difusões iguais a 10 e, apesar da análise do passo não ser conclusiva, um passo da ordem de 0,75. Cabe enfatizar que, embora tenham sido sugeridos valores *default* para uma nova aplicação, que estes podem ser diferentes para cada novo estudo de caso.
- No Capítulo 5 o algoritmo de BFE foi investigado considerando uma série de problemas de otimização mono e multi-objetivos. De forma geral, conclui-se que o referido algoritmo obteve um bom desempenho em termos de convergência quando comparado com outras abordagens. Todavia, para os problemas multi-objetivos foi verificado um elevado custo computacional. Isto se deve ao fato da associação entre o algoritmo de BFE e o Método da Soma Ponderada, o que, naturalmente, implica no aumento do número de avaliações da função objetivo devido à natureza deste tipo de abordagem. Ainda neste capítulo destaca-se a aplicação do algoritmo de BFE na resolução de um problema de otimização que apresenta quatro conjuntos de variáveis de projeto que conduzem ao mesmo valor de função objetivo. Para essa finalidade foi proposta uma abordagem baseada em penalização. Neste caso, de forma automática, as quatro soluções foram encontradas a partir da definição de um parâmetro auxiliar. É importante ressaltar que, para a metodologia proposta, o número de soluções encontradas é função da escolha deste parâmetro que pode ser encontrado facilmente após testes preliminares.
- Ainda no Capítulo 5, os resultados obtidos com a associação do algoritmo de BFE com novos operadores foram apresentados e discutidos. Como esperado, todas as estratégias empregadas para aumentar a capacidade de exploração do algoritmo original, bem como a redução do custo computacional foram, na média, melhores que a versão canônica. Particularmente para os operadores empregados para a atualização dinâmica do número de difusões e do tamanho do passo da caminhada Gaussiana, observa-se que estes foram mais eficientes, em pelo menos uma destas novas versões, em termos da redução do custo computacional com relação à versão canônica. Cabe destacar o desempenho do algoritmo de BFE combinado com o modelo de busca caótica Mapa Logístico (BFEML), pois este teve o melhor desempenho. Em relação à variação do tamanho da população, mesmo esta abordagem não levando em conta informações sobre o andamento do processo evolutivo para atualizar esse parâmetro, observa-se que o uso

deste operador sempre resultou na redução do número de avaliações da função objetivo em relação a versão canônica. Ao se considerar a associação do algoritmo de BFE com ambos os operadores(BFEM) observa-se, para os estudos de caso analisados, que o seu desempenho foi, no mínimo, equivalente às outras versões consideradas e superior a sua versão canônica.

- Em resumo, conclui-se que a metodologia proposta configura-se como uma alternativa interessante em relação a outras estratégias heurísticas já difundidas na literatura especializada, visto os resultados apresentados. É importante ressaltar que, embora não tenha sido computado o tempo total de processamento em cada aplicação, indiretamente, através do número de avaliações da função objetivo, o seu valor médio pode ser estimado e um comparativo entre os estudos de caso analisados pode ser realizado. Todavia, enfatiza-se que a literatura especializada não apresenta, comumente, o valor deste parâmetro, pois o mesmo depende de vários fatores, tais como a linguagem de programação e o microcomputador considerados, o que dificulta um comparativo no que tange o tempo total de processamento obtidos por diferentes autores.

6.2 Trabalhos Futuros

Como propostas de trabalhos futuros pretende-se:

- Aplicar o algoritmo de BFE e suas novas versões em problemas com maior complexidade, tais como em controle ótimo e em problemas inversos.
- Associar o algoritmo de BFE com métodos clássicos e outras heurísticas para avaliar a sua capacidade no contexto híbrido.
- Propor a extensão do algoritmo de BFE para o contexto multi-objetivo considerando a sua associação com o critério de dominância de Pareto para fins da redução do custo computacional em relação ao uso do Método da Soma Ponderada.
- Propor novos operadores para a atualização dinâmica do tamanho da população baseados no andamento do processo evolutivo, assim como considerado por Lobato & Steffen Jr (2016).

Referências

- ARORA, J. S. "Introduction to optimum design". New York: McGraw-Hill, 1989.
- ATHAN, T. W.; PAPALAMBROS, P. Y. "A note on weighted criteria methods for compromise solutions in multi-objective optimization". *Eng. Optimization*, v. 27, n. 3, p. 155-176, 1996.
<https://doi.org/10.1080/03052159608941404>
- BARBOSA, C. E. M. "Algoritmos bio-inspirados para solução de problemas de otimização". Dissertação, Universidade Federal de Pernambuco, Recife, PE, 2017.
- BERNARDES, R. "Sistemas Neuro-fuzzy Aplicados no Controle de uma Aeronave de Asa Fixa". 2019. 64 f. Trabalho de Conclusão de Curso (Graduação em Engenharia Mecatrônica), Universidade Federal de Uberlândia, Uberlândia, 2018.
- BEZERRA, C. L. N. A.; COSTA, F. G. B. C.; NASCIMENTO, G. M.; CARVALHO, P. V. M.; PAIVA, F. A. P. "Uma Variante Melhorada do Algoritmo Busca Cuco usando uma Estratégia de Quasi Opposition-Based Learning". In: *Proceedings of IX Computer on the Beach*, p. 60-69, 2018.
- BRANDÃO, M. A. L. "Improved differential evolution implemented in parallel processing". 2014. 173 f. Tese (Doutorado em Engenharias), Universidade Federal de Uberlândia, Uberlândia, 2014.
- BRASIL, R. M. L. R. F.; SILVA, M. A. "Otimização de Projetos de Engenharia". 1. Ed. São Paulo: Edgard Blücher Ltda, 2019. p. 7-17.
- BRITO, G. P. "Computational numerical optimization applied to the dynamic behavior of a cursor crank mechanism - Case study for the implementation of a beta configuration stirling Engine". 2013. 110 f. Dissertação (Mestrado em Engenharias), Universidade Federal de Uberlândia, Uberlândia, 2013.
- CARVALHO, C. C. "Desenvolvimento de um algoritmo de otimização evolutivo auto-adaptativo para a resolução de problemas de otimização com variáveis mistas". 2018. 105 f. Dissertação (Mestrado em Modelagem e Otimização), Universidade Federal de Goiás, Catalão, 2018.
- CARVALHO, F. C. "Projeto e otimização de controladores para rotores com mancais magnéticos: Projeto Robusto". 2020. 151 f. Tese (Doutorado em Engenharia Mecânica), Universidade Federal de Uberlândia, Uberlândia, 2021.
- CASTRO, R.E. "Otimização de Estruturas com Multi-Objetivos via Algoritmos Genéticos". Tese (Doutorado) - Departamento de Engenharia Civil - COPPE/UFRJ, 2001.
- CAVALINI JUNIOR, A. A. "Detection and identification of incipient transversal cracks in flexible and horizontal shafts of rotating machines". 2013. 270 f. Tese (Doutorado em Engenharias), Universidade Federal de Uberlândia, Uberlândia, 2013.
- CHANGYU, W.; LIU, Q.; CHENG, M. "Smoothing SQP algorithm for semismooth equations with box constraints". *Computational Optimization & Applications*, v. 55, n. 2, p. 399-425, 2013.
<https://doi.org/10.1007/s10589-012-9524-5>
- CHEN, X.; YUE, H.; YU, K. "Perturbed Stochastic fractal search for solar PV parameter estimation". *Energy, China*, 189, 1-12, 2019.
<https://doi.org/10.1016/j.energy.2019.116247>

- CHICKERMANE, H.; GEA, H.C. "Structural optimization using a new local approximation method". *Int. J. Numer. Meth. Eng.*, 39, 829–846, 1996.
[https://doi.org/10.1002/\(SICI\)1097-0207\(19960315\)39:5<829::AID-NME884>3.0.CO;2-U](https://doi.org/10.1002/(SICI)1097-0207(19960315)39:5<829::AID-NME884>3.0.CO;2-U)
- COELHO, R.F. "Multi-Criteria Optimization with Expert Rules for Mechanical Design". Tese (Doutorado) - Université Libre de Bruxelles, Faculté des Sciences Appliquées, 2004.
- COELHO, L. S.; MARIANI, V. C. "Combining of chaotic differential evolution and quadratic programming for economic dispatch optimization with valve-point effect". *IEEE Transactions on Power Systems*, v. 21, n. 2, p. 989–996, 2006a.
<https://doi.org/10.1109/TPWRS.2006.873410>
- COELHO, L. S.; MARIANI, V. C. "Evolução diferencial híbrida com programação quadrática aplicada ao problema de despacho econômico de energia elétrica". *Sba Controle & Automação*, v. 17, n. 4, 2006b.
<https://doi.org/10.1590/S0103-17592006000400003>
- COELHO, L. dos S.; BERNERT, D. L. de A.; MARIANI, V. C. "A chaotic firefly algorithm applied to reliability-redundancy optimization". *2011 IEEE Congress of Evolutionary Computation (CEC)*, p. 517–521, June 2011. ISSN 1089-778X.
- COELHO, L. S.; MARIANI, V. C. "Improved firefly algorithm approach applied to chiller loading for energy conservation". *Energy and Buildings*, v. 59, p. 273–278, 2013.
<https://doi.org/10.1016/j.enbuild.2012.11.030>
- COELLO, C. A. C.; MONTES, E. M. "Constraint-handling in genetic algorithms through the use of dominance-based tournament selection". *Advanced Engineering informatics*, Mexico, n. 16, p. 193–203, 2002.
[https://doi.org/10.1016/S1474-0346\(02\)00011-3](https://doi.org/10.1016/S1474-0346(02)00011-3)
- COUTO, D. C. F.; SILVA, C. A.; BARSANTE, L. S. "Otimização de Funções Multimodais via Técnica de Inteligência Computacional Baseada em Colônia de Vaga-Lumes". In: *CILAMCE*, 36., 2015, Rio de Janeiro. Proceedings of the XXXVI Iberian Latin-American Congress on Computational Methods in Engineering, 2015.
<https://doi.org/10.20906/CPS/CILAMCE2015-0506>
- DEB, K. "Multi-Objective Optimization using Evolutionary Algorithms". First edition. New York: John Wiley & Sons, 2001.
- DEVANEI, R. L. "An Introduction to Chaotic Dynamical Systems". Addison-Wesley, 1987.
- EDGAR, T.F.; HIMMELBLAU, D. M.; LASDON, L.S. "Optimization of Chemical Process". Second edition. New York: McGraw-Hill Chemical Engineering Series, 2001.
- EDGEWORTH, F. Y. "Mathematical Physics". First edition. London, England: P. Keagan, 1881.
- FREITAS, P. C. B. "Análise Numérica de Ligações Metálicas Viga-Coluna com Coluna Tubular Circular". Dissertação, Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, SP, Brasil, 2009.
- GANDOMI, A.H.; YANG, X.S.; ALAVI, A.H. "Cuckoo search algorithm: a metaheuristic approach to solve structural optimization problems". *Eng. Comput.*, 29, 17–35, 2013.
<https://doi.org/10.1007/s00366-011-0241-y>

- GARCIA, A. M.; POSSAGNOLO, L. H. F. M.; ROMERO, R. “Algoritmo de Busca Dispersa para Otimização de Funções Contínuas Multimodais com Restrições”. In: *SBPO*, 50., 2018, Rio de Janeiro. Anais do 50º Simpósio Brasileiro de Pesquisa Operacional, 2018.
- GOULD, N. I. M.; ROBINSON, D. P. “A Second Derivate SQP Method: Global Convergence”. *SIAM Journal on Optimization*, v. 20, n. 40, p. 2023-2048, 2010.
<https://doi.org/10.1137/080744542>
- HAUPT, R.; HAUPT, S.E. “Practical Genetic Algorithms”. First edition. INC.: John Wiley & Sons, 1998.
- HE, Q.; WANG, L. “An effective co-evolutionary particle swarm optimization for constrained engineering design problems”. *Eng Appl Artif Intell*, 20, 89-99, 2007.
<https://doi.org/10.1016/j.engappai.2006.03.003>
- HIMMELBLAU, D. M. “Applied nonlinear programming”. McGraw-Hill, New York, 1972.
- HINOJOSA, S.; DHAL, K. G. G.; ELAZIZ, M. A.; OLIVA, D.; CUEVAS, E. “Entropy-based imagery segmentation for breast histology using the Stochastic Fractal Search”. *Neurocomputing*, 321, 1, 201-215, 2018.
<https://doi.org/10.1016/j.neucom.2018.09.034>
- HSU, Y.L.; LIU, T.C. “Developing a fuzzy proportional derivative controller optimization engine for engineering design optimization problems”. *Eng. Optm.*, 39(6), 679–700, 2007.
<https://doi.org/10.1080/03052150701252664>
- KENNEDY, J.; EBERHART, R. “Particle swarm optimization”. In: *Proceedings of IEEE International Conference on Neural Networks*, pp. 1942-1948, 1995.
- KHALILPOURAZARI, S.; KHALILPOURAZARY, S. “A Robust Stochastic Fractal Search approach for optimization of the surface grinding process”. *Swarm and Evolutionary Computation*, 38, 173-186, 2018.
<https://doi.org/10.1016/j.swevo.2017.07.008>
- KHALILPOURAZARI, S.; NADERI, B.; KHALILPOURAZARY, S. “Multi-objective Stochastic Fractal Search: a powerful algorithm for solving complex multi-objective optimization problems”. *Soft Computing*, 24, 3037-3066, 2019.
<https://doi.org/10.1007/s00500-019-04080-6>
- LAUTERT, E. L. “Otimização de Amortecedores por atrito em estruturas submetidas a excitações sísmicas”. 2019. 22f. Trabalho de Conclusão de Curso (Graduação em Engenharia Mecânica), Universidade Federal do Rio Grande do Sul, Porto Alegre, 2019.
- LECCA, E. R.; LIZAMA, E. R. “La geometría del método Simplex y sus aplicaciones utilizando Matlab”. *Industrial Data*, v. 10, n. 2, 2007.
- LOBATO, F. S.; STEFFEN JR, V. “Desenvolvimento de um Algoritmo de Otimização Auto-Adaptativo para a Determinação de um Protocolo Otimizado para a Administração de Drogas no Tratamento de Tumores”. *Tendencias em Matemática Aplicada e Computacional*, v. 17, n. 2, 225-245, 2016.
<https://doi.org/10.5540/tema.2016.017.02.0225>
- LOBATO, F. S.; STEFFEN JR, V.; LOPES, L. C. O. “Uma Abordagem Evolutiva Baseada em Modelos Caóticos de Busca Associado ao Algoritmo de Evolução Diferencial”. In: *PosMec*, 17., 2007, Uberlândia. Anais eletrônicos. Uberlândia, 2007.

- LOBATO, F. S. "Otimização Multi-Objetivo para o Projeto de Sistemas de Engenharia". Tese, Universidade Federal de Uberlândia, Uberlândia, MG, 2008.
- MAHADAVI, M.; FESANGHARY, M.; DAMANGIR, E. "An Improved Harmony Search Algorithm for Solving Optimization Problems". *Applied Mathematics and Computation*, vol. 188, n. 2, pp. 1567-1579, 2007.
<https://doi.org/10.1016/j.amc.2006.11.033>
- MARINHO, C. E. V.; SANTOS NETO, A. J. "A eficiência polinomial do simplex para redes: Aplicação em um problema de caminho mais curto". *Vértices*, v. 5, n. 2, pp. 123-138, 2010.
<https://doi.org/10.5935/1809-2667.20030015>
- MEDEIROS, G. F. D.; KRIPKA, M. "Algumas Aplicações de Métodos Heurísticos na Otimização de Estruturas". In: *CIATEC - UPF*, Passo Fundo, 4, 19-32, 2012.
<https://doi.org/10.5335/ciatec.v4i1.2264>
- MEDEIROS, G. F.; KRIPKA, M. "Modified Improved Harmony Search: Proposta de Melhorias para o Algoritmo de Otimização". In: *CILAMCE*, 38., 2017, Florianópolis. Proceedings of the XXXVIII Iberian Latin-American Congress on Computational Methods in Engineering, 2017.
- METROPOLIS, N.; ROSENBLUTH, A.W.; ROSENBLUTH, M.N.; TELLER, A.H. "Equations of state calculations by fast computing machines". *Journal of Chemical Physics*, 21, 1087-1092, 1953.
<https://doi.org/10.1063/1.1699114>
- MICHALEWICZ, Z. "Evolutionary Computation for NonLinear Programming Problems", 1998.
- MICHALEWICZ, Z.; DEB, K.; SCHMIDT, M.; STIDEN, T. "Evolutionary Algorithms for Engineering Applications". First edition. *Evolutionary Algorithms in Engineering and Computer Science*, Edited by Miettinen, K; Neittaanmaki, P.; Makela, M. M. e Periaux, J.: John Wiley & Sons, 1999.
- MIRJALILI, S.; LEWIS, A. "The Whale Optimization Algorithm". *Advances in engineering software*, Australia, 95, 51-67, 2016.
<https://doi.org/10.1016/j.advengsoft.2016.01.008>
- MOREIRA, D. D.; KROHLING, R. A. "Um algoritmo bioinspirado em polinização de plantas usando entropia populacional." In: *SBPO*, 47., 2015, Porto de Galinhas. XLVII Simpósio Brasileiro de Pesquisa Operacional, 2015.
- NIGDELI S.M.; BEKDAS G.; YANG X. S. "Application of the Flower Pollination Algorithm in Structural Engineering". In: Yang XS., Bekdas G., Nigdeli S. (eds) *Metaheuristics and Optimization in Civil Engineering. Modeling and Optimization in Science and Technologies*, vol 7. Springer, Cham, 2016.
https://doi.org/10.1007/978-3-319-26245-1_2
- OLIVEIRA, G. T. S. "A study and applications of differential evolution". 2006. 126 f. Dissertação (Mestrado em Engenharias), Universidade Federal de Uberlândia, Uberlândia, 2006.
- OLIVEIRA, P. J. "Optimal trajectory planning of robots with parallel structure". 2005. 145 f. Tese (Doutorado em Engenharias), Universidade Federal de Uberlândia, Uberlândia, 2005.

- OLIVEIRA, G. T. S. "Optimum design of 3R robots manipulators considering its topology of the workspace". 2012. 244 f. Tese (Doutorado em Engenharias), Universidade Federal de Uberlândia, Uberlândia, 2012.
- OMAR M.; EBRAHIM, M.A.; ABDEL GHSNY, A.M.; BENDARY, F. "Reduced Size Harmony Search Algorithm for Optimization". *Journal of Electrical Engineering*, v.1, 1-8, 2016.
<https://doi.org/10.11591/ijeecs.v1.i2.pp255-263>
- OSY CZKA, A. "An approach to multicriterion optimization problems for engineering design". *Comp. Methods in Applied Mech. and Eng.*, v. 15, p. 309-333, 1978.
[https://doi.org/10.1016/0045-7825\(78\)90046-4](https://doi.org/10.1016/0045-7825(78)90046-4)
- OSY CZKA, A. "Multicriterion Optimization in Engineering with Fortran Programs". First Edition. England: Ellis Horwood Limited, 1984.
- OTT, E. "Chaos in Dynamical Systems". Cambridge University Press, Cambridge, UK, 2002.
<https://doi.org/10.1017/CBO9780511803260>
- PAIVA, F. A. P. "Otimização por enxame de partículas: usando uma adaptação de serendipidade". 1. Ed. Natal: Editora IFRN, 2018.
- PARETO, V. "Cours D'Economie Politique". First edition. França: Vol I and II, F. Rouge, Lausanne, 1896.
- PINHEIRO, L.M. "Fundamentos do concreto e projeto de edifícios". Cap.5. São Carlos, 2007.(Apostila).
- PIRES, V. C. "Utilização de Ferramentas Computacionais para a Resolução de Problemas de Programação Linear". Trabalho de Conclusão de Curso (Licenciatura em Matemática), Universidade Estadual do Sudoeste da Bahia, Vitória da Conquista, BA, 2018.
- PURCINA, L. A. "Evolutionary Optimization Techniques Applied to Solution of Large Linear Systems". 2010. 162 f. Dissertação (Mestrado em Engenharias), Universidade Federal de Uberlândia, Uberlândia, 2010.
- PRICE, K. "Genetic annealing". *Dr Bobb's Journal*, 19(10):127-132, 1994.
<https://doi.org/10.1080/07488009408409150>
- QUEIROZ, T. A. "Optimal project of semi-embedded columns". 2008. 129 f. Dissertação (Mestrado em Engenharias), Universidade Federal de Uberlândia, Uberlândia, 2008.
- RASHEDI, E.; NEZAMABADI-POUR, H.; SARYAZDI, S. "GSA: a gravitational search algorithm". *Inf Sci.*, 179, 2232-2248, 2009.
<https://doi.org/10.1016/j.ins.2009.03.004>
- SALES, G.C.S; SIMONETTI, H.L; ALMEIDA, V.D.D. "Métodos Clássicos de Otimização". In: *VI Seminário de Iniciação Científica - IFMG*, 7., 2018, Sabará. (Resumo).
- SALIMI, H. "Stochastic Fractal Search: A powerful metaheuristic algorithm". *Knowledge-Based Systems*, 75, 1-18, 2015.
<https://doi.org/10.1016/j.knosys.2014.07.025>
- SANTOS, D. "O problema de identificação de acidentes em usinas nucleares otimizado pelo algoritmo cuco". 2019. 68f. Trabalho de Conclusão de Curso (Graduação em Engenharia Nuclear), Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2019.

- SANTOS JUNIOR, A. A. “Elementos de máquinas I: molas helicoidais e planas”. Campinas, 2001.(Apostila).
- SARAMAGO, S. F. P. “Métodos de Otimização Randômica: Algoritmos Genéticos e Simulated Annealing”. FAMAT/UFU, Notas em Matemática Aplicada, Volume 6, SBMAC, São Carlos - SP, 1999.
- SCHAFFER, J.D. “Multiple Objective Optimization with Vector Evaluated Genetic Algorithms”. Tese (Doutorado) - Vanderbilt University, 1984.
- SHEN, C.; XUE, W.; CHEN, X. “Global convergence of a robust filter SQP algorithm”. *European Journal of Operational Research*, v. 206, issua 1, p. 34-45, 2010.
<https://doi.org/10.1016/j.ejor.2010.02.031>
- SILVA, M. R. “Aplicação de um algoritmo híbrido estocástico-determinístico na solução de um problema de direção automotiva”. In: *ENEGEP*, 37., 2017, Joinville. XXXVII Encontro Nacional de Engenharia de Produção, 2017a.
- SILVA, M. R. “Algoritmo Híbrido Estocástico-Determinístico aplicado na solução do problema de equilíbrio químico de um processo de combustão de hidrocarbonetos”. In: *EMEP-PRO*, 13., 2017, Juiz de Fora. XIII Encontro Mineiro de Engenharia de Produção, 2017b.
- SIMÕES, R. C. “Identificação de falhas em eixos de rotores flexíveis utilizando técnicas de otimização.” 2002. 80 f. Dissertação (Mestrado em Engenharia Mecânica), Universidade Federal de Uberlândia, Uberlândia, 2002.
- STORN, R.; PRICE, K. “Differential Evolution – A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces”. *Journal of Global Optimization*, Netherlands, v. 11, p. 341-359, 1996.
<https://doi.org/10.1023/A:1008202821328>
- SURJANOVIC, S.; BINGHAM, D. “Virtual Library of Simulation Experiments: Test Functions and Datasets”, 2013. Disponível em: <http://www.sfu.ca/ssurjano>. Acesso em: 04 abr. 2021.
- TICONA, W. G. C. “Aplicação de Algoritmos Genéticos Multi-Objetivo para Alinhamento de Sequências Biológicas”. Dissertação (Mestrado) - Instituto de Ciências Matemáticas e de Computação - ICMC-USP, São Carlos - SP, 2003.
- UFPA. “Barra Cantilever, Notas de Estudo de Engenharia Mecânica”. Belém, 2011. (Apostila).
- VANDERPLAATS, G.N. “Numerical Optimization Techniques for Engineering Design”. Third edition. USA: VR D INC. Colorado Springs, 1999.
- VAN KAMPEN, A. H. C.; STROM, C. S.; BUYDENS, L. M. C. “Lethalization, penalty and repair functions for constrained handling in the genetic algorithm methodology”. In: *Chemometrics and Intelligent Laboratory Systems*, pp. 55-68. USA: Gen96, 1996.
[https://doi.org/10.1016/0169-7439\(96\)00010-X](https://doi.org/10.1016/0169-7439(96)00010-X)
- VISWANATHAN, G. M.; RAPOSO, E. P.; LUZ, M. G. E. “Lévy flights and superdiffusion in the context of biological encounters and random searches”. *Physics of Life Reviews*, 5, 133-150, 2008.
<https://doi.org/10.1016/j.plrev.2008.03.002>
- WAGNER, F. “O que são vasos de pressão.Rw Engenharia”, 2015. Disponível em: <https://www.rwengenharia.eng.br/o-que-sao-vasos-de-pressao/>. Acesso em: 15 jan. 2021.

- YANG, X. S. “Nature-Inspired Metaheuristic Algorithms”. Luniver Press, Cambridge, USA, first edition, 2008.
- YANG, X. S. “Engineering Optimisation: An Introduction with Metaheuristic Applications”. John Wiley and Sons, 2010.
<https://doi.org/10.1002/9780470640425>
- YANG, X. S. “Flower Pollination Algorithm for Global Optimization”. *Unconventional Computation and Natural Computation*, edited by J. Durand-Lose and N. Jonoska, Vol. 7445 of Lecture Notes in Computer Science, 240-249. Berlin: Springer, 2012.
https://doi.org/10.1007/978-3-642-32894-7_27
- YANG, D.; LI, G.; CHENG, G. “On the efficiency of chaos optimization algorithms for global optimization”. *Chaos, Solitons & Fractals*, v. 34, n. 4, p. 1366 – 1375, 2007.
<https://doi.org/10.1016/j.chaos.2006.04.057>
- ZAHARIE, D. “Critical values for the control parameters of differential evolution algorithms”. In: *Proceedings of the 8th International Conference on Soft Computing*. [S.l.: s.n.], p. 62–67, 2002.
- ZHOU, T.; ZHOU, Y.; SUNDMACHER, K. “A hybrid stochastic-deterministic optimization approach for integrated solvent and process design”. *Chemical Engineering Science*, v. 159, p. 207-216, 2017.
<https://doi.org/10.1016/j.ces.2016.03.011>
- ZITZLER, E.; LAUMANN, M.; THIELE, L. “SPEA II: Improving the Strength Pareto Evolutionary Algorithm”. *Computer Engineering and Networks Laboratory (TIK)*, Swiss Federal Institute of Technology (ETH) Zurich, Gloriastrasse 35, CH-8092. Zurich - Switzerland, 2001.