

**UNIVERSIDADE ESTADUAL PAULISTA “JÚLIO DE MESQUITA FILHO”
FACULDADE DE ENGENHARIA
CAMPUS DE ILHA SOLTEIRA**

DANIEL PEREIRA DOS SANTOS

**SOLUÇÃO DE PROBLEMAS COM RESTRIÇÃO VIA TÉCNICAS DE
PENALIZAÇÃO ACOPLADAS AO ALGORITMO DE COMPETIÇÃO
IMPERIALISTA**

ILHA SOLTEIRA

2019

DANIEL PEREIRA DOS SANTOS

**SOLUÇÃO DE PROBLEMAS COM RESTRIÇÃO VIA TÉCNICAS DE
PENALIZAÇÃO ACOPLADAS AO ALGORITMO DE COMPETIÇÃO
IMPERIALISTA**

Dissertação de Mestrado apresentada ao Programa de Pós-graduação em Engenharia Mecânica, Faculdade de Engenharia de Ilha Solteira – UNESP, como parte dos requisitos necessários para obtenção do título de Mestre em Engenharia Mecânica.

Orientador: Prof. Dr. Fábio Roberto Chavarette

ILHA SOLTEIRA

2019

FICHA CATALOGRÁFICA
Desenvolvido pelo Serviço Técnico de Biblioteca e Documentação

S237s Santos, Daniel Pereira dos.
Solução de problemas com restrição via técnicas de penalização acopladas
ao Algoritmo de Competição Imperialista / Daniel Pereira dos Santos. -- Ilha
Solteira: [s.n.], 2019
69 f. : il.

Dissertação (mestrado) - Universidade Estadual Paulista. Faculdade de
Engenharia de Ilha Solteira. Área de conhecimento: Mecânica dos Sólidos, 2019

Orientador: Fábio Roberto Chavarette
Inclui bibliografia

1. Otimização com restrição. 2. Algoritmo de Competição Imperialista. 3.
Métodos de penalização. 4. Tratamento de restrições.

Raiane da Silva Santos
Raiane da Silva Santos

Supervisora Técnica de Seção
Setor Técnico de Referência, Atendimento ao usuário e Documentação
Diretoria Técnica de Biblioteca e Documentação
CRB/8 - 0099

CERTIFICADO DE APROVAÇÃO

TÍTULO DA DISSERTAÇÃO: Solução de problemas com restrição de penalização acopladas ao algoritmo de competição imperialista

AUTOR: DANIEL PEREIRA DOS SANTOS

ORIENTADOR: FABIO ROBERTO CHAVARETTE

Aprovado como parte das exigências para obtenção do Título de Mestre em ENGENHARIA MECÂNICA, área: Mecânica dos Sólidos pela Comissão Examinadora:



Prof. Dr. FABIO ROBERTO CHAVARETTE
Departamento de Matemática / Faculdade de Engenharia de Ilha Solteira



Prof. Dr. FRANCISCO VILLARREAL ALVARADO
Departamento de Matemática / Faculdade de Engenharia de Ilha Solteira



Prof. Dr. ROBERTO OUTEIRO
Faculdade de Tecnologia Professor Fernando Amaral de Almeida Prado / Faculdade de Tecnologia de Araçatuba, FATEC

Ilha Solteira, 08 de março de 2019

Dedico esta dissertação à minha Esposa Stéfanni, por todo incentivo e carinho que me inspirou à concluir mais esta etapa em minha vida. Aos meus Pais, Mauro e Márcia, por proporcionarem minha base de educação e caráter que hoje retribuo com mais uma conquista. E a todos familiares e amigos, que de formas diferentes e não menos especiais, me ajudaram a obter o título de Mestre.

RESUMO

O trabalho do profissional que atua nas diferentes áreas da engenharia resume-se na incessante busca pelo aumento da produtividade e rendimento, e pela redução do impacto das restrições como, consumo, custo, peso sobre os sistemas. Contudo, a solução de problemas de engenharia na sua forma analítica e exata só é possível para sistemas muito simples, desta forma, sistemas mais complexos, que geralmente possuem várias soluções, faz-se necessário optar por procedimentos de aproximação numérica com precisão aceitável e métodos de otimização para determinar a melhor solução dentre as existentes. Nos últimos anos, as técnicas de otimização têm estudado os comportamentos de fenômenos da natureza, visando a concepção de heurísticas de otimização, no qual, são utilizados para a solução de problemas de engenharia com restrições. Neste estudo é realizada a aplicação de diferentes técnicas de penalização, para tratamento de restrições, acoplados a um Algoritmo de Competição Imperialista (ICA), uma meta-heurístico que se baseia no imperialismo colonial adotado pelos países no século XVI com o objetivo de estender seus limites territoriais através da conquista de outros territórios denominados colônias. Diversos experimentos computacionais são realizados visando avaliar o desempenho do algoritmo considerando vários problemas testes encontrados na literatura.

Palavras-chave: Otimização com restrição, Algoritmo de Competição Imperialista, Métodos de penalização, tratamento de restrições.

ABSTRACT

The work of the professional who works in the different areas of engineering boils down to the incessant search for increased productivity and efficiency, and for reducing the impact of constraints such as consumption, cost, and weight on systems. However, the solution of engineering problems in its analytical and exact form is only possible for very simple systems, in this way, more complex systems, which usually have several solutions, it becomes necessary to opt for numerical approximation procedures with acceptable precision and methods to determine the best solution among the existing ones. In recent years, optimization techniques have studied the behavior of natural phenomena, aiming at the design of optimization heuristics, in which they are used to solve constrained engineering problems. In this study the application of different penalization techniques for the treatment of constraints is coupled with an Imperialist Competitive Algorithm (ICA), a metaheuristic that is based on colonial imperialism adopted by the countries in the 16th century with the objective of extending its limits through the conquest of other territories called colonies. Several computational experiments are performed aiming to evaluate the performance of the algorithm considering several test problems found in the literature.

Palavras-chave: Otimização com restrição, Algoritmo de Competição Imperialista, Métodos de penalização, tratamento de restrições.

LISTA DE FIGURAS

Figura 1 – Mínimo e máximo de uma função.	16
Figura 2 – Classificação dos problemas de otimização.	18
Figura 3 – Ótimos locais vs ótimo global.	19
Figura 4 – Problema bidimensional sujeito a restrições de desigualdade.	20
Figura 5 – Problema bidimensional sujeito à restrição de igualdade.	21
Figura 6 – Deslocamento de uma colônia em direção ao imperialista.	30
Figura 7 – Fluxograma ICA	33
Figura 8 – Determinação de $f(x)$.	40
Figura 9 – Projeto de mola sob tração ou compressão.	45
Figura 10 – Comparativo das convergência mínima e média.	49
Figura 11 – Convergência dos métodos de penalização.	50
Figura 12 – Projeto de viga engastada.	50
Figura 13 – Comparativo das convergências mínima e média.	56
Figura 14 – Convergência dos métodos de penalização.	57
Figura 15 – Projeto de vaso de pressão.	57
Figura 16 – Comparativo das convergência mínima e média	61
Figura 17 – Convergência dos métodos de penalização.	62
Figura 18 – Perfis de desempenho utilizando os melhores resultados obtidos.	63
Figura 19 – Perfis de desempenho utilizando a média dos resultados obtidos.	64

LISTA DE TABELAS

Tabela 1 – Informações do projeto de mola.	45
Tabela 2 – Resultados numéricos para o projeto de mola.	47
Tabela 3 – Melhor solução para o projeto de mola.	48
Tabela 4 – Informações do projeto de viga engastada.	51
Tabela 5 – Resultados numéricos para o projeto de viga.	54
Tabela 6 – Melhor solução para o projeto de viga.	55
Tabela 7 – Informações do projeto de vaso de pressão.	58
Tabela 8 – Resultados numéricos para o projeto de mola.	59
Tabela 9 – Melhor solução para o projeto de mola.	60
Tabela 10 – Área normalizada dos perfis de desempenho utilizando os melhores resultados obtidos.	63
Tabela 11 – Área normalizada dos perfis de desempenho utilizando a média dos resultados obtidos.	64
Tabela 12 – Comparação dos resultados para os problema propostos.	65

SUMÁRIO

INTRODUÇÃO	13
1.1 ORGANIZAÇÃO DA DISSERTAÇÃO	14
CONCEITOS DE ALGORITMOS DE OTIMIZAÇÃO	15
2.1 OTIMIZAÇÃO.....	16
2.2 FORMULAÇÃO DO PROBLEMA DE OTIMIZAÇÃO.....	17
2.2.1 Ponto Ótimo	19
2.2.2 Ótimos locais e ótimos globais.....	19
2.2.3 Representação gráfica de um problema bidimensional	19
2.3 ALGORITMOS DE OTIMIZAÇÃO.....	21
2.3.1 Métodos Determinísticos	21
2.3.2 Métodos Probabilísticos.....	22
2.3.2.1 Método de Recozimento Simulado	23
2.3.2.2 Algoritmos Genéticos.....	24
2.3.2.3 Colônia de Formigas	25
2.3.2.4 Enxame de Partículas	25
2.3.2.5 Colônia de Abelhas Artificial	26
2.3.2.6 Algoritmo de Competição Imperialista	27
ALGORITMO DE COMPETIÇÃO IMPERIALISTA	27
3.1 O ALGORITMO ICA.....	28
3.1.1 Geração da população inicial	28

3.1.2	Deslocamento das colônias em direção aos Imperialistas	29
3.1.3	Revolução.....	30
3.1.4	Troca de posição do imperialista e uma colônia	31
3.1.5	Poder total de um império	31
3.1.6	A Competição Imperialista	31
3.1.7	Critério de Parada	32
3.2	FLUXOGRAMA ICA	33
MÉTODOS DE PENALIZAÇÃO		33
4.1	PROBLEMAS COM RESTRIÇÕES	34
4.2	TÉCNICAS DE PENALIZAÇÃO	35
4.2.1	Penalização Estática.....	36
4.2.2	Penalização Dinâmica.....	36
4.2.3	Penalização Adaptativa.....	37
4.2.3.1	Adaptive Penalty Method (APM)	38
4.2.3.2	APM Esporádico	40
4.2.3.3	APM Esporádico com Acúmulo das Violações.....	41
RESULTADOS NUMÉRICOS		41
5.1	PARÂMETROS DO ALGORITMO ICA	42
5.2	MÉTODO DE ANÁLISE DE DESEMPENHO	43
5.3	MOLA DE TRAÇÃO/COMPRESSÃO.....	43
5.4	VIGA ENGASTADA.....	50
5.5	VASO DE PRESSÃO.....	57

5.6	ANÁLISE DE DESEMPENHO.....	62
	CONSIDERAÇÕES FINAIS	66
	REFERÊNCIAS.....	67

1. INTRODUÇÃO

Mediante a natureza finita dos recursos naturais à disposição da humanidade, a necessidade de otimizar a utilização destes recursos desde a antiguidade é um dos grandes desafios para a sociedade. Nesse contexto, a otimização pode ser considerada um poderoso ferramental matemático que, para um determinado problema, é capaz de explorar um vasto número de possibilidades e definir qual, ou quais soluções, é mais conveniente para resolvê-lo. Em outras palavras, é o processo que visa encontrar o valor máximo ou mínimo de uma função matemática, chamada função objetivo, que contém em sua formulação as variáveis que definem o problema.

A tradução de problemas reais em modelos matemáticos passou a ser utilizada para o estudo e resolução de muitos problemas de otimização no âmbito da engenharia e em outras áreas da ciência. Após meados do século XX, com o impulsionamento da tecnologia computacional, uma grande variedade de métodos, como os métodos de programação matemática e os algoritmos meta-heurísticos, foram desenvolvidos para resolver uma infinidade de problemas, gerando ganhos significativos para a indústria como, maximização do uso de matéria prima, minimização de custos de produção, otimização de roteiros de transporte, entre outros.

Um dos algoritmos meta-heurísticos em ascensão na atualidade é o Algoritmo de Competição Imperialista, ICA, do inglês Imperialist Competitive Algorithm, proposto inicialmente por Atashpaz-Gargari e Lucas (2007). Sua heurística faz analogia a conceitos de relações sociopolíticas e pode ser visto como a versão sociopolítica do Algoritmo Genético desenvolvido por Holland (1973), que como a maioria dos algoritmos evolutivos, tem sua heurística baseada em fenômenos biológicos.

Contudo, é válido ressaltar que, todos esses algoritmos, em essência, foram criados para lidar com problemas sem restrição. Problemas reais de engenharia muitas vezes envolvem restrições, o que impossibilita a aplicação destes algoritmos em sua forma plena. As estratégias mais comuns em algoritmos de otimização para lidar com restrições baseiam-se em converter o problema com restrição em um problema sem restrição. Neste contexto, destacam-se as técnicas de penalização, que fazem a conversão do problema ao adicionar à função objetivo penalizações às soluções que violarem as restrições impostas ao problema.

As técnicas de penalização foram originalmente propostas por Courant (1943) e expandidas por Carroll (1961) e Fiacco e McCormick (1966); Bean e Hadj-Alouane (1992); Smith e Tate (1993); Schoenauer e Xanthakis (1993); Homaifar et.al (1994); Joines e Houck (1994); Coit et.al (1996); Kulturel-Konak et.al (2004); Barbosa e Lemonge (2004).

Vários trabalhos na literatura testam a robustez de algumas de técnicas de penalização acopladas em algoritmos meta-heurísticos de busca, através de métodos de análise de desempenho, quando aplicados em problemas de engenharia.

Em Barbosa e Lemonge (2002), foi mensurado o desempenho de uma de Técnica de Penalização Adaptativa (APM) acoplado a um algoritmo genético evolucionário, no qual, mostrou-se eficaz e robusto quando aplicado para testar problemas da literatura de computação evolutiva, bem como vários problemas de otimização da engenharia estrutural. Simões e Ebecken (2015), utiliza o APM desenvolvido por Barbosa e Lemonge (2002) acoplado ao Algoritmo de Enxame de Partículas (PSO) aplicado à otimização topológica de suporte laterais de fornos para refinarias de petróleo. Já Duarte et.al (2012) realiza um estudo comparativo de diferentes técnicas de penalização acopladas no algoritmo colônia de abelhas artificiais aplicados a problemas de otimização em engenharia.

Motivado pelo surgimento de novos algoritmos meta-heurísticos essa dissertação tem por objetivo explorar a capacidade de diferentes técnicas de penalização acopladas ao Algoritmo de Competição Imperialista (ICA), um algoritmo de busca meta-heurístico. É apresentado um estudo comparativo e detalhado com o objetivo de analisar e testar a aplicação do algoritmo meta-heurístico ICA, em problemas clássicos da engenharia mecânica, e identificar as técnicas de penalização que apresentam os melhores desempenhos. Para facilitar a visualização e interpretação dos resultados obtidos nas simulações, é adotado o método de análise de desempenho, desenvolvido por Dolan e Moré (2002).

1.1 ORGANIZAÇÃO DA DISSERTAÇÃO

Este texto é composto por 6 capítulos, conforme descrito a seguir:

Capítulo 1: Introdução

Neste capítulo é apresentada a introdução sobre o problema, os fundamentos que justificam o presente trabalho, objetivos e motivações.

Capítulo 2: Conceitos de Algoritmos de Otimização

No capítulo 2 é apresentado a revisão bibliográfica acerca dos conceitos que definem os problemas de otimização onde, é detalhado com maior ênfase, os problemas de otimização com restrição. E, por fim, são apresentados alguns algoritmos de otimização citados na literatura.

Capítulo 3: Algoritmo de Competição Imperialista

No capítulo 3 aprofunda-se o estudo acerca do método de busca utilizado nessa dissertação, o Algoritmo de Competição Imperialista.

Capítulo 4: Métodos de Penalização

No capítulo 4 detalham-se as técnicas de penalização, no qual, há um destaque maior para a técnica de penalização adaptativa APM.

Capítulo 5: Aplicação numérica dos Algoritmos de Otimização

No capítulo 5, são abordados os problemas de engenharia com restrições, encontrados na literatura no âmbito da Engenharia Mecânica. E ainda são apresentados, os resultados numéricos de convergência obtidos, assim como, os resultados de análise de desempenho dos algoritmos.

Capítulo 6: Considerações Finais

No capítulo 6, são apresentadas as conclusões e as intenções futuras de estudos.

2. CONCEITOS DE ALGORITMOS DE OTIMIZAÇÃO

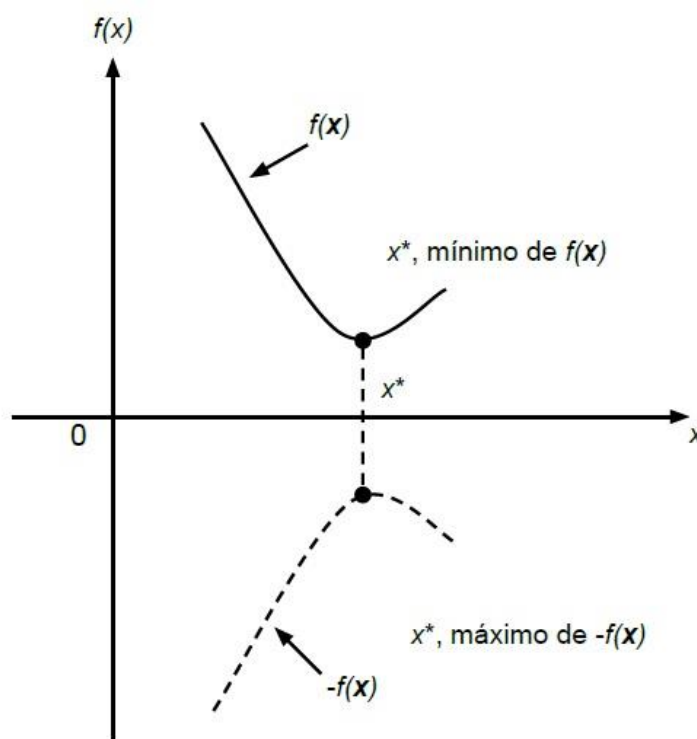
Este capítulo visa detalhar as técnicas de otimização propostas neste estudo. Nele os algoritmos serão apresentados e o seu funcionamento explicado.

2.1 OTIMIZAÇÃO

Otimização é um ramo da matemática que se preocupa com a obtenção de resultados que dão o “valor extremo” de uma função, ou conjunto de funções, sob determinadas circunstâncias.

Em outras palavras, otimizar é um processo de encontrar o valor ótimo (máximo ou mínimo) de uma função (RAO, 2009). Como pode ser observado na Figura 1, se um ponto x^* corresponde ao valor mínimo da função $f(x)$, o mesmo ponto, também corresponde ao valor máximo da função negativa $-f(x)$. No contexto desta dissertação, os termos “mínimo” e “ótimo” são referenciados como termos iguais.

Figura 1 – Mínimo e máximo de uma função.



Fonte: Adaptado de Rao (2009).

Analogamente, para uma melhor compreensão, os problemas de otimização podem ser entendidos como a subida em uma montanha. Suponha que a forma da superfície de uma

montanha represente o gráfico de uma função qualquer e o objetivo é atingir o seu pico (ponto máximo) desta montanha. Intuitivamente, antes de começar a andar, procura-se uma direção para depois decidir o quanto andar nesta direção. A certeza de estar seguindo o caminho certo dependerá da experiência adquirida durante o caminho, por exemplo, se durante o caminho havia um buraco (ponto de mínimo local) então devemos evita-lo, assim uma nova direção deve ser escolhida para seguir no passo seguinte, e assim por diante até chegar ao pico.

Segundo Rao (2009), a criação e utilização dos métodos de otimização foram possíveis devido as contribuições de Newton e Leibnitz, Bernoulli, Euler, Lagrange e Weirstrass que dedicaram esforços ao cálculo das variações, que está intimamente ligado a minimização de funções. Já a otimização para problemas restritos tornou-se conhecida por Lagrange, que introduziu os “Multiplicadores de Lagrange” utilizados em muitas técnicas de otimização. Apesar das contribuições citadas, o progresso sobre o assunto foi muito baixo até meados do século XX, quando computadores digitais de alta velocidade foram utilizados para procedimentos de otimização, entusiasmando a pesquisa de outros métodos.

2.2 FORMULAÇÃO DO PROBLEMA DE OTIMIZAÇÃO

De maneira geral, os problemas de otimização com restrição seguem um padrão composto por um objetivo, que deve decidir se deseja-se o mínimo ou o máximo valor dentre as possíveis soluções, um conjunto de restrições que devem ser satisfeitos indicando que os valores obtidos foram aceitos. Aliado a isso, segundo Kargupta e Goldberg (1995) a base para a formulação de um problema de otimização é a construção de um modelo matemático bem definido partindo das descrições físicas do problema.

Pode-se formular matematicamente um problema de otimização como demonstrado a seguir:

Encontrar um vetor x contendo as variáveis de projeto

$$x = (x_1, x_2, \dots, x_n), \quad x_i \in \mathbb{R}, \quad i = 1, \dots, n \quad (1)$$

que minimize a função objetivo $f(x)$

$$f(x) = f(x_1, x_2, \dots, x_N) \quad (2)$$

sujeito a

$$g_j(x) \leq 0, \quad j = 1, 2, \dots, n_g \quad (3)$$

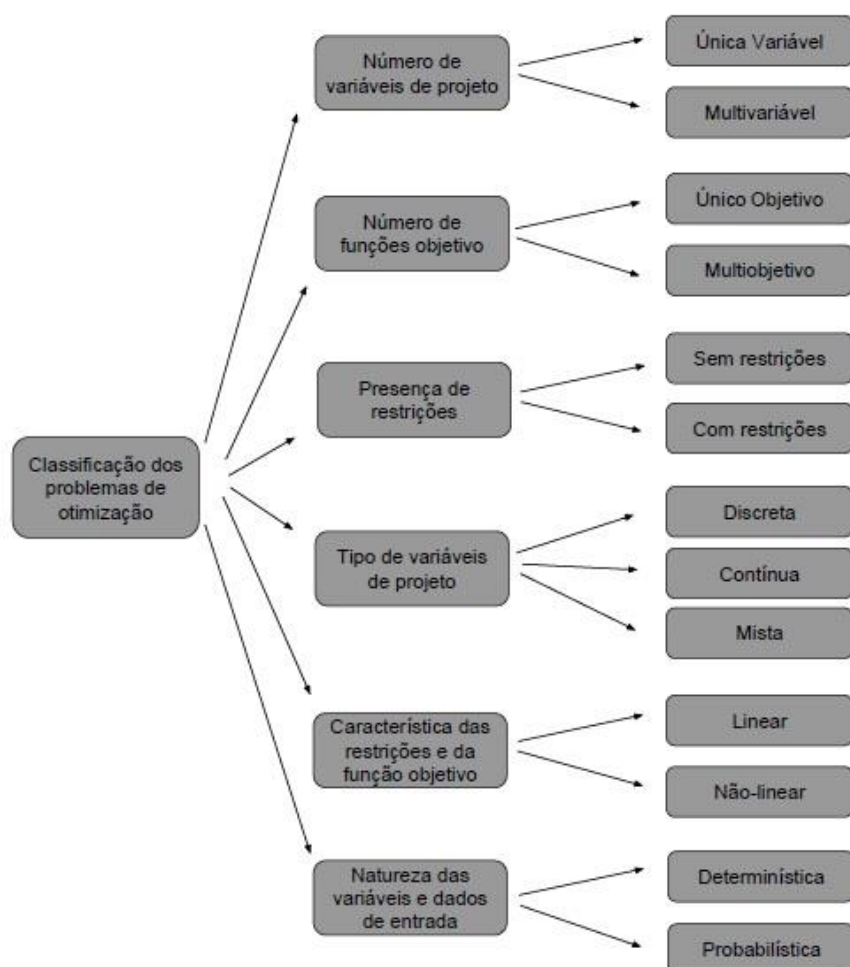
$$h_k(x) = 0, \quad k = 1, 2, \dots, n_h \quad (4)$$

$$x_i^L \leq x_i \leq x_i^U, \quad i = 1, 2, \dots, n \quad (5)$$

onde $h_k(x)$ e $g_j(x)$ são as restrições de igualdade e desigualdade; e n_h e n_g são, respectivamente, o número de restrições de igualdade e desigualdade. A equação 5 representa a restrição de fronteira, ou seja, são os limites de cada variável, onde x_i^U é o limite superior e x_i^L é o limite inferior de uma dada variável.

Vários tipos de problemas de otimização podem ser encontrados no cotidiano dependendo da escolha das variáveis de projeto, função objetivo e restrições. De acordo com Rao (2009), através da Figura 2, é mostrado uma classificação dos problemas de otimização.

Figura 2 – Classificação dos problemas de otimização.



Fonte: Adaptado de Rao (2009).

2.2.1 Ponto Ótimo

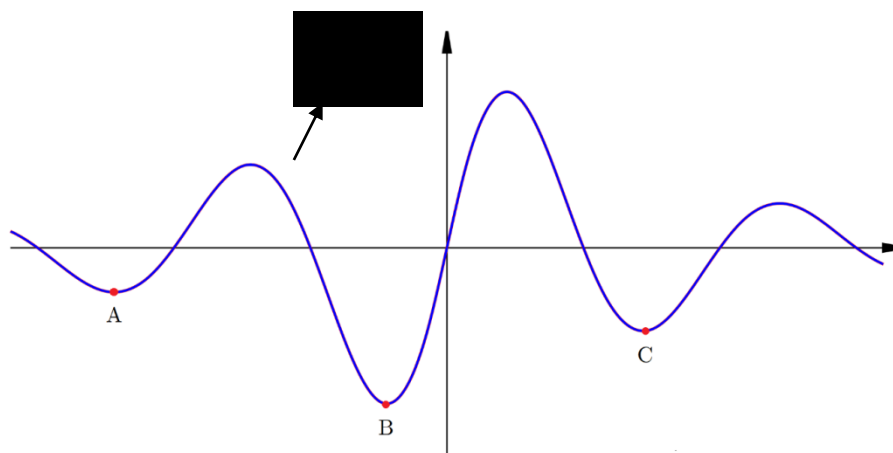
Um ponto no domínio da função objetivo pode ser entendido como um ponto ótimo quando qualquer variação infinitesimal das variáveis de projeto nos leva a um resultado com desempenho igual ou inferior. Ou seja, $f(x) \geq f(x^*)$, no qual, a igualdade entre $f(x)$ e $f(x^*)$ somente ocorrerá se a função objetivo possuir um valor constante em uma região, sendo assim, esta poderia ser designada como uma região ótima. Deste modo, denominamos como um ponto ótimo qualquer, um vetor x^* com n elementos.

$$x^* = (x_1^*, x_2^*, \dots, x_n^*) \quad (6)$$

2.2.2 Ótimos locais e ótimos globais

Depois de modelar do problema com suas variáveis e restrições de projeto, na maioria dos problemas de engenharia é necessário lidar com a formação de vários ótimos locais da função objetivo. A Figura 3 mostra que uma $f(x)$ qualquer possui três mínimos locais A, B e C; no qual, é nítido afirmar que o mínimo global desta função é o ponto B.

Figura 3 – Ótimos locais vs ótimo global.



Fonte: próprio autor.

2.2.3 Representação gráfica de um problema bidimensional

Considere uma função bidimensional hipotética $f(x_1, x_2)$ sujeito a três restrições de desigualdade e duas restrições de fronteira.

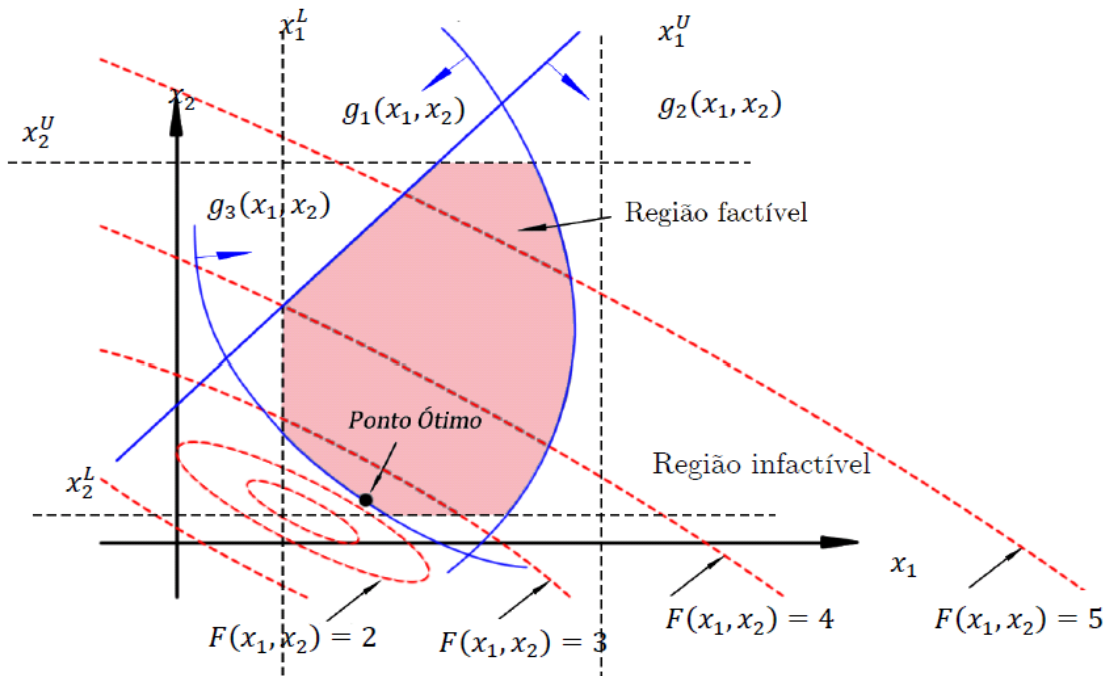
$$g_j(x_1, x_2) < 0, \quad j \in \mathbb{R}, \quad j = 1, \dots, 3 \quad (7)$$

$$\begin{aligned} x_1^L &\leq x_1 \leq x_1^U \\ x_2^L &\leq x_2 \leq x_2^U \end{aligned} \quad (8)$$

Na Figura 4, é mostrado a representação gráfica da problema hipotético onde há duas áreas, uma região factível, que representa a área onde pretende-se achar o melhor ponto ótimo, e outra infactível (espaço em branco), que representa a região de não interesse para a busca. Deste modo, a ilustração mostra como as restrições limitam a região factível de busca e quais são suas características principais.

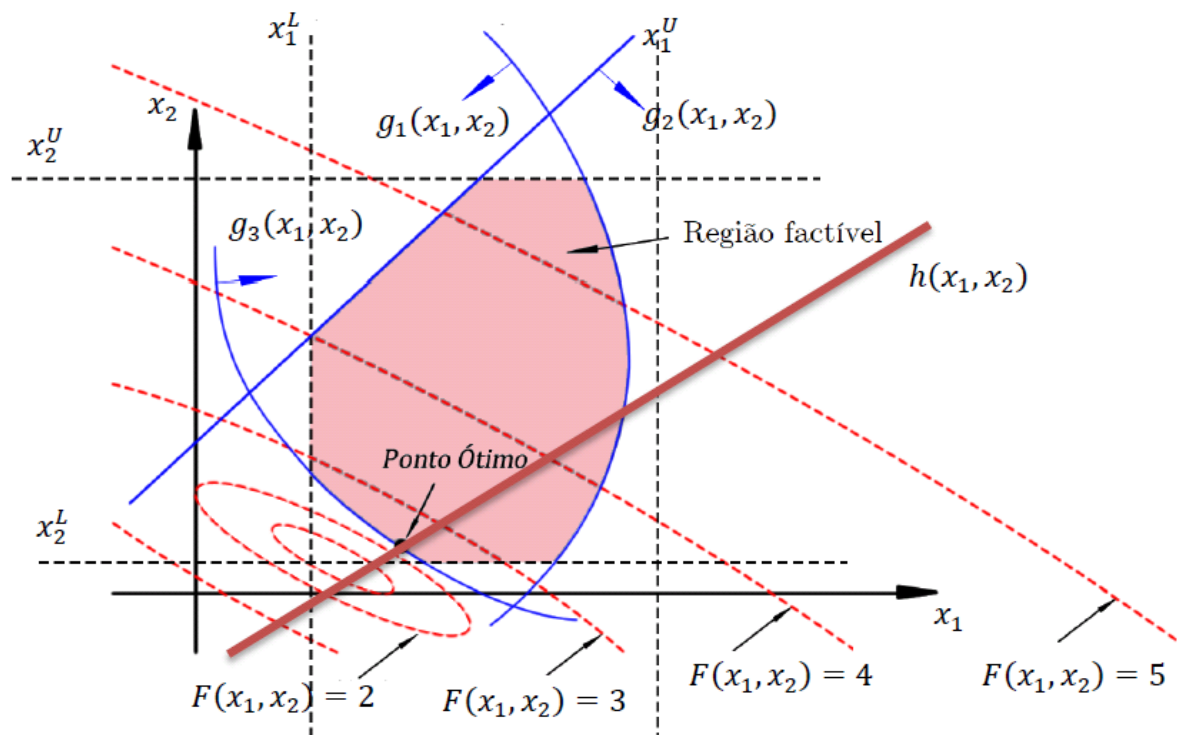
As restrições de fronteira ou laterais, representadas pelas pontilhadas que cortam os pontos x_1^L , x_1^U , x_2^L e x_2^U em seus respectivos eixos, definem os limites das variáveis do projeto, assim nenhum ponto da região factível pode extrapolar estes limites. Já as restrições $g_1(x_1, x_2)$, $g_2(x_1, x_2)$ e $g_3(x_1, x_2)$ são restrições de desigualdade que geralmente restringem consideravelmente a região factível. Na Figura 5, que representa o mesmo problema anterior, adiciona uma restrição de igualdade $h(x_1, x_2)$. Assim, a região factível deixa de ser representada por uma área e passa a ser representada pela linha que define $h(x_1, x_2)$.

Figura 4 – Problema bidimensional sujeito a restrições de desigualdade.



Fonte: Adaptado de Rao [1].

Figura 5 – Problema bidimensional sujeito à restrição de igualdade.



Fonte: Adaptado de Rao (2009)

2.3 Algoritmos de Otimização

Para a obtenção de solução de problemas de otimização deve-se levar em conta diversos fatores presentes na modelagem do problema que deseja-se resolver. Fatores como a natureza da função objetivo (linear, não linear, convexa, contínua, multimodal), a suavidade da(s) função(ões) objetivo(s) e das restrições, a quantidade de funções objetivo.

Segundo Kargupta e Goldberg (1995) os algoritmos usados em um problema de otimização podem ser classificados em determinísticos ou probabilísticos. A escolha de uma metodologia ou outra deve considerar o problema e suas especificidades e, este processo, constitui uma etapa importante para o sucesso da busca da solução para um problema.

2.3.1 Métodos Determinísticos

Nos métodos determinísticos a solução é dependente do ponto de partida fornecido, ou seja, uso destes métodos faz com que sempre uma mesma resposta seja encontrada quando se parte de um mesmo ponto, assim para um dado valor inicial sempre será gerada uma mesma solução. De modo geral, o método pode convergir para um ótimo local ao invés do ótimo global.

Os algoritmos de natureza determinística podem ser classificados em lineares e não-lineares.

- Programação não-linear: onde a função objetivo e/ou pelo menos uma das restrições envolvidas não são funções lineares. Alguns exemplos são: Método da Programação Linear Sequencial, Método da Programação Geométrica, Método da Programação Quadrática Sequencial, Método das Direções Variáveis, Método do Gradiente Reduzido entre outros (BORGWARDT, 1987).
- Programação linear: é quando a função objetivo e as restrições são funções lineares das variáveis de projeto. Um exemplo clássico da programação linear é o Método Simplex (BORGWARDT, 1987).

Para Izmailov e Solodov (2009) embasado no teorema de Weierstruss para a existência de mínimos, pode-se até provar a convergência no uso de métodos determinísticos para a solução ótima, porém sem evidências de que ela é a global. A explicação para isso deve-se à dependência da solução encontrada para com o ponto de partida e quando da existência de múltiplos ótimos locais.

2.3.2 Métodos Probabilísticos

Embasado na publicação de Medeiros e Kripka (2012), métodos clássicos de programação matemática são em geral ineficientes quando aplicados em problemas reais de engenharia de maior complexidade, devido principalmente ao comportamento das funções que os descrevem. É nesse contexto que os métodos probabilísticos são mais eficazes (KARGUPTA; GOLDBERG, 1995).

Os métodos de otimização baseados nos algoritmos probabilísticos em geral usam somente a avaliação da função objetivo e introduzem no processo de otimização parâmetros estocásticos. Essa classe de algoritmos possui vantagens em relação aos algoritmos determinísticos: a função objetivo e as restrições não precisam estar explicitadas; não há restrição quanto ao ponto de partida dentro do espaço de busca da solução; trabalham adequadamente com variáveis contínuas e discretas ou uma combinação delas. Contudo, sua principal desvantagem é o custo computacional elevado (MEDEIROS; KRIPKA, 2012).

Alguns exemplos de métodos probabilísticos, discutidos a seguir, são: Método de Recozimento Simulado; Algoritmos Genéticos; Colônia de Formigas, Enxame de Partículas, Colônia de Abelhas, entre outros.

2.3.2.1 Método de Recozimento Simulado

O Recozimento Simulado ou Simulated Annealing (SA) é um algoritmo heurístico que se baseia na termodinâmica, mais especificamente na mecânica probabilística do processo de resfriamento dos materiais, onde substâncias físicas tais como os metais que são elevados a altas temperaturas até que o material se funda e posteriormente é gradualmente resfriado até alcançar um estado mínimo de energia com uma massa homogênea (KIRKPATRICK et.al, 1983).

O SA tem por finalidade a simulação do comportamento de um grupo de átomos (variáveis do projeto) em processo de busca de um estado de baixa energia. Uma redução bem lenta da temperatura desde o estado de fusão do material vai resultar em uma estrutura cristalina pura. Caso contrário, se a redução de temperatura for brusca os cristais resultantes serão muito frágeis e o material pode se transformar em vidro, que é uma estrutura ótima localmente.

A busca parte de uma solução inicial t e, a cada processo de iteração, um vizinho t_v é gerado randomicamente. É feito o seguinte cálculo de variação:

$$\Delta = f(t) - f(t_v) \quad (9)$$

Um mecanismo de subida de temperatura é inserido no sistema para garantir a fuga dos mínimos locais facilitando a obtenção do ótimo global. Se $\Delta < 0$, então t_v é aceito como a nova solução e a solução t é descartada. E, se for gerado vizinhos piores que uma quantidade $\Delta (\Delta > 0)$, então ele é aceito com uma probabilidade:

$$P(\Delta E) = e^{\left(\frac{-\Delta E}{K_b T}\right)} \quad (10)$$

em que K_b é a constante de Boltzmann, e T é um parâmetro que representa a temperatura. Quando o parâmetro T assume valores muito pequenos tal que não seja mais possível fazer novas atualizações, então o processo é finalizado, e dito que o sistema se estabilizou.

Assim, o progresso das iterações do SA é limitado à probabilidade de se aceitar uma nova solução por meio da redução da temperatura. Nota-se que o algoritmo permite que uma solução pior do que a solução corrente possa ser aceita. Essa permissão tem a finalidade de escapar de ótimos locais em busca do ótimo global.

É importante ressaltar que a eficiência do algoritmo está atrelada ao bom dimensionamento do parâmetro T , que caso possua um valor desnecessariamente alto, culminará na aceitação de um grande número de resultados indesejáveis. Da mesma forma, ao ajustar um valor demasiadamente pequeno para T , o algoritmo terá dificuldade de escapar dos mínimos locais (SARAMAGO, 2003).

2.3.2.2 Algoritmos Genéticos

O Algoritmo Genético (AG) foi desenvolvido por Holland (1973) durante a década de 60. O processo de convergência de um AG é inspirado nos mecanismos da genética e na teoria de seleção natural de Charles Darwin, onde uma população randômica inicial de cromossomos é gerada e cada cromossomo representa uma possível solução para o problema dentro dos limites definidos para cada variável. Cada indivíduo da população é avaliado e recebe seu valor de aptidão (*fitness*). Em geral, é possível na própria função de avaliação incorporar uma função de penalidade, que será capaz de avaliar e penalizar os indivíduos que eventualmente violem as restrições do problema. Os cromossomos são selecionados para a próxima geração e transferem suas características para seus descendentes através de dois mecanismos: *crossover* e mutação. Todo esse processo é repetido até que seja encontrada uma solução satisfatória.

Existem várias formas de representação de uma possível solução para um problema via AG. A mais utilizada, devida sua simplicidade, é a representação binária, por poder ser facilmente utilizada em uma grande quantidade de problemas. Outras formas de representação são: real, permutação de símbolos, árvores, entre outros.

Em relação à estratégia de reposição das populações nos AGs, destacam-se dois tipos: geracional, onde toda a população é substituída a cada geração, ocasionando grande variabilidade do material genético entre as gerações; e steady-state, onde somente uma parcela da população (geralmente 2 indivíduos) é substituída a cada iteração (HOLLAND, 1973).

2.3.2.3 Colônia de Formigas

O algoritmo de otimização conhecido como Ant Colony Optimization (ACO) foi introduzido por Dorigo (1992) e seus associados no começo da década de 90 e imita o comportamento cooperativo de uma colônia de formigas. As formigas utilizam o feromônio para guardar um caminho enquanto elas se movimentam. Isso acontece quando elas saem de sua colônia e caminham aleatoriamente na busca de alimento. Quando o alimento é encontrado, as formigas retornam à colônia deixando um rastro de feromônio para que as outras formigas da colônia possam encontrar o alimento (COLORNI et.al, 1991).

Cada solução candidata do algoritmo ACO é representada por uma formiga. Cada formiga escolhe uma rota e deposita uma certa quantidade de feromônio em cada rota percorrida. As melhores rotas seriam identificadas pela quantidade de feromônio.

2.3.2.4 Enxame de Partículas

O método de otimização *Particle Swarm Optimization* (PSO) foi originalmente desenvolvido por Kennedy (2001) e é inspirada no voo de pássaros, que são convenientemente denominados de partículas, baseado nos estudos do sócio-biologista Edward Osborne Wilson (1971). A princípio, os pássaros iniciam sua revoada de forma desordenada e, de acordo com o tempo, observa-se que o bando se organiza e um padrão de busca é concretizado. Quando o alimento for encontrado, espera-se que todo o bando se dirija para ele (WILSON, 1995).

De acordo com Shi e Eberhart (1998) o PSO inicializa aleatoriamente uma população de partículas (matriz $n \times m$) com posições e velocidades em um espaço de busca n -dimensional, são atribuídos os valores para $pbest$ (cada melhor posição individual) e $gbest$ (a melhor posição entre todas as partículas do enxame) e os indivíduos são avaliados através da função de avaliação (*fitness*). Após essa inicialização os resultados da função de avaliação são comparados ao $pbest$ e $gbest$, caso os valores correntes sejam melhores, então $pbest$ e $gbest$ são atualizados. Já a velocidade v_i a posição da partícula x_i , são atualizadas considerando a experiência da partícula correspondente e a experiência das outras partículas presentes na população, de acordo com as seguintes equações:

$$v_i(t + 1) = w \cdot v_i(t) + C_1 \cdot rand(PBEST - x_i(t)) + C_2 \cdot rand(GBEST - x_i(t)) \quad (11)$$

$$x_i(t + 1) = x_i(t) + v_i(t + 1) \quad (12)$$

A notação t representa o incremento temporal unitário, $v_i(t)$ representa o vetor velocidade de cada partícula i na iteração t e $x_i(t)$ representa o vetor posição de cada partícula i na iteração t . C_1 e C_2 representam respectivamente os parâmetros social e cognitivo, que influenciam cada partícula em direção a PBEST e GBEST. O *rand* representa uma função geradora de números aleatórios de distribuição uniforme entre $[0,1]$ e w é o fator de inércia, que controla o impacto da velocidade previa da partícula sobre sua velocidade atual Shi e Eberhart (1998). Por fim, o processo iterativo deve se repetir até que um critério de parada seja encontrado, usualmente um valor de erro pré-definido ou um numero máximo de iterações.

Para uma melhor compreensão da influencia dos diferentes parâmetros no algoritmo. C_1 e C_2 controlam o fluxo de informação entre o enxame atual. Se $C_2 > C_1$, então a partícula põe mais confiança no enxame, de outra forma, a partícula põe mais confiança nela mesma. No algoritmo básico inicial do PSO, proposto por Kennedy (2001), $C_1 = C_2 = 2$. Trabalhos posteriores indicam a preferência pelo uso de outros valores para os parâmetros cognitivo e social. Um fator de inércia w alto facilita uma exploração global do espaço de busca, enquanto que um valor pequeno deste parâmetro possibilita uma busca local (ajuste fino), portanto, a apropriada seleção do fator de amortecimento fornece um balanço entre a capacidade de busca local e de busca global do algoritmo e permitirá um número menor de iterações (em media) para identificar o ponto ótimo global. A literatura propõe usar $0 < \omega < 1.4$, $C_1 = C_2 = 2$ com $C_1 + C_2 \leq 4$, (CARLISLE; DOZIER, 2001).

2.3.2.5 Colônia de Abelhas Artificial

Baseia-se no procedimento das abelhas ao saírem em busca de alimentos e retornar à colmeia com informações precisas da direção e distância do local visitado e uma amostra do alimento encontrado. O algoritmo inspirado nesse comportamento foi proposto por Karaboga (2005).

Segundo Akay e Karaboga (2012) as abelhas se organizam e desenvolvem uma inteligência coletiva baseado em três características: auto organização, adaptação e divisão do trabalho.

O algoritmo ABC inicia com uma população de abelhas gerada aleatoriamente. As abelhas são classificadas em três grupos: abelhas operárias campeiras (scout), abelhas operárias (employed) e abelhas seguidoras (recruited). As campeiras são responsáveis por procurar fontes de alimentos. A partir do momento que uma abelha campeira explora uma fonte de alimento, ela se torna uma abelha operária. Quando a fonte de alimento se esgota, a abelha operária volta a ser uma abelha campeira. E as seguidoras são as abelhas que ajudam as operárias a explorar a fonte de alimento.

2.3.2.6 Algoritmo de Competição Imperialista

O Algoritmo de Competição Imperialista ou *Imperialist Competitive Algorithm* (ICA) é um dos algoritmos meta-heurísticos mais recentes, proposto por Atashpaz-Gargari e Lucas (MOUSSAVI RAD et.al, 2012). Ele é inspirado pelo processo sócio-político da competição imperialista. Comparado com outros algoritmos evolucionários convencionais, o ICA tem demonstrado desempenho superior, ou seja, convergência mais rápida e maior capacidade para alcançar a solução ótima global. Essa meta-heurística é a técnica utilizada nessa dissertação e é apresentada com mais detalhes no Capítulo 3.

3. ALGORITMO DE COMPETIÇÃO IMPERIALISTA

Este capítulo visa detalhar a técnicas de otimização proposta neste estudo. Nele, o Algoritmo ICA será apresentado e o seu funcionamento explicado.

3.1 O ALGORITMO ICA

A Competição Imperialista, ICA, do inglês, proposto inicialmente por Atashpaz-Gargari e Lucas (2007), é baseado no imperialismo, política adotado por um governo que tem por objetivo estender seus limites territoriais através da conquista de outros territórios denominados colônias.

Assim como a maioria dos algoritmos evolutivos o ICA é particularmente eficiente para a solução de problemas de otimização global. A Heurística ICA faz analogia a conceitos de relações sociopolíticas e pode ser visto como a versão sociopolítica do Algoritmo Genético que, como a maioria dos algoritmos evolutivos, tem sua heurística baseada em fenômenos biológicos.

3.1.1 Geração da população inicial

De acordo com Moradi et al. 2014, cada indivíduo da população é convenientemente chamado de País e uma possível solução para o problema. O número de países é o parâmetro que define a quantidade de soluções que serão analisadas a cada iteração, é diretamente proporcional ao custo computacional.

Os países são representados por uma string de dimensão $1 \times N$, onde N é o número de variáveis do problema

$$país = (p_1, p_2, \dots, p_N), \quad i = 1, \dots, N \quad (13)$$

Assim o método tem início com a geração aleatória de uma população inicial, em um espaço N -dimensional, e com a definição do número de países Np e a quantidade de imperialistas $Nimp$. Um número maior de imperialistas resulta em uma maior capacidade de diversificação das soluções, contudo, custa uma necessidade de mais iterações para convergência do algoritmo.

A partir do primeiro conjunto de países, é tomado o custo de cada país através do cálculo da Função Objetivo:

$$custo = f(país) = f(p_1, p_2, \dots, p_N) \quad (14)$$

Os N_{imp} países com os melhores valores (menor custo) são selecionados e passam a ser denominados Imperialistas e todos os demais N_{col} países são designados Colônias. O valor da função objetivo em cada Imperialista define o Poder deste Imperialista. Em seguida, todas as colônias são distribuídas entre os imperialistas de maneira proporcional ao poder de cada Imperialista, esta distribuição ocorre conforme:

$$P_n = \frac{|C_n|}{\sum_{i=1}^{N_{imp}} C_i}, \quad n \in \mathbb{R}, \quad n = 1, \dots, N_{imp} \quad (15)$$

sendo,

$$C_n = c_n - \max_i(c_i) \quad (16)$$

onde, P_n é o poder normalizado de um imperialista, C_n é o custo normalizado e c_n é o custo do n -ésimo imperialista. Assim, fazendo com que o Imperialista com maior poder fique com o maior número de Colônias e o conjunto (Imperialista + Colônias) passa a ser designado Império.

A quantidade de Colônias que o Imperialista inicialmente irá possuir é determinada por:

$$NC_n = \text{round}(P_n \cdot N_{col}), \quad n = 1, \dots, N_{imp} \quad (17)$$

onde NC_n é o número de colônias iniciais que o i -ésimo imperialista irá possuir e N_{col} é quantidade total de colônias. A operação "*round*" atribui o valor inteiro mais próximo.

3.1.2 Deslocamento das colônias em direção aos Imperialistas

A cada iteração todas as colônias se movimentam em direção aos seus respectivos Imperialistas. Esse movimento é regido por dois parâmetros do algoritmo:

(i) Coeficiente de assimilação (β): define o “tamanho do passo” dado por cada colônia a cada década. Seguindo o roteiro do algoritmo, o tamanho x do passo dado por cada colônia é um valor randômico com distribuição uniforme escolhido entre:

$$x \sim U(0, \beta \times d) \quad \beta > 1 \quad (18)$$

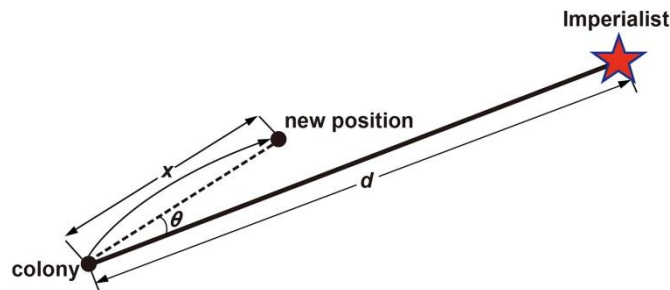
no qual, d é o vetor distância entre o imperialista e a colônia, e β é um valor maior que 1 para garantir que as colônias se aproximem do imperialista por ambos os lados.

(ii) Coeficiente angular de assimilação (γ): define o limite máximo do ângulo aleatório existente entre o movimento da colônia e a direção do imperialista em cada iteração. Assim, a direção do movimento feito por cada colônia é, a direção da colônia até o imperialista adicionado de um desvio randômico uniforme (θ):

$$\theta \sim U(-\gamma, \gamma) \quad (19)$$

Desta forma β e θ forçam o algoritmo a procurar melhores soluções na região em torno do imperialista. A Figura 6 demonstra o deslocamento de uma colônia.

Figura 6 – Deslocamento de uma colônia em direção ao imperialista.



Fonte: Adaptado de Atashpaz-Gargari e Lucas [18].

3.1.3 Revolução

Assim mostrado por Kaveh e Talatahari (2010) é ainda possível que uma colônia mude para uma posição aleatória, ou seja, é possível que ocorra revoluções em colônias que estão subjugadas, resultando em uma melhora dessa colônia. Este mecanismo de revolução, de acordo com Moussab Rad et al. (2012), aumenta o fator de exploração do algoritmo, e evita a convergência prematura de países para um ótimo local. A probabilidade de isso ocorrer se deve a um parâmetro ajustável denominado Taxa de Revolução T_{rev} . Assim, um

determinado número de colônias do império deverá ser substituído pelo mesmo número de novos países, de acordo com a seguinte equação:

$$NRC = \text{round}(T_{rev} \times NC_n), \quad n = 1, \dots, N_{imp} \quad (20)$$

3.1.4 Troca de posição do imperialista e uma colônia

Semelhante a outras técnicas de otimização, a melhor solução deve ser selecionado para orientar o algoritmo a convergir para o mínimo global. No final das movimentações, caso uma colônia se torne mais poderosa que seu imperialista (menor valor para a Função Objetivo) a colônia e o imperialista trocarão sua posição.

3.1.5 Poder total de um império

Após as movimentações, é calculado o poder total de um império, ou seja, a soma entre o custo do imperialista mais uma porcentagem do custo de suas colônias, como segue:

$$TC_n = f(\text{Imperialista}_n) + \varepsilon \times \text{média } f(\text{colônias do Imperialista}_n) \quad (21)$$

onde, TC_n é o custo total do império. ε é o coeficiente de posição, uma constante entre [0,1]; que define a proporção dada ao poder de todas as colônias de um império em relação ao poder do Imperialista na definição do poder dos impérios. Já o operador *média* calcula a média dentre todas as colônias do n -ésimo imperialista.

3.1.6 A Competição Imperialista

O próximo passo é promover a competição imperialista. O algoritmo ICA modela essa regra selecionando algumas das colônias mais fracas dos impérios mais fracos e dividindo entre outros impérios. A colônia com pior desempenho do império mais fraco é selecionada para ser conquistada pelo império com maiores chances de adquiri-la. Para isso, é definido o custo normalizado dos impérios por:

$$NTC_n = TC_n - \text{máx}_i(TC_n) \quad (22)$$

onde NTC_n é o custo normalizado do n -ésimo império e $\text{máx}_i(TC_n)$ é o custo total do império com o maior custo total de todos. Com isto, a probabilidade de posse P_{P_n} para cada império é dada por:

$$P_{pn} = \frac{|NTC_n|}{\left| \sum_{i=1}^{N_{imp}} NTC_i \right|} \quad (23)$$

São definidos também o vetor $\mathbf{P}$, com as probabilidades de posse de cada império, e o vetor $\mathbf{R}$, do tamanho de $\mathbf{P}$, com números aleatórios distribuídos uniformemente entre [0.1], tal que:

$$\mathbf{P} = [P_{p1}, P_{p2}, \dots, P_{pN_{imp}}] \quad (24)$$

$$\mathbf{R} = [r_1, r_2, \dots, r_{N_{imp}}] \quad (25)$$

Feito isto, cria-se outro vetor denominado $\mathbf{D}$, de mesmo tamanho de $\mathbf{P}$ e $\mathbf{R}$, contendo em cada índice a subtração dos valores $\mathbf{P}$ com os valores $\mathbf{R}$:

$$\mathbf{D} = \mathbf{P} - \mathbf{R} = [D_1, D_2, \dots, D_{N_{imp}}] \quad (26)$$

O império escolhido a possuir a colônia será aquele com o índice do maior valor de $\mathbf{D}$. Em seguida, o algoritmo analisa se há algum império sem colônias, se houver, o império é eliminado.

E então através de uma comparação de poderes entre os Impérios, de maneira concomitante ao movimento das colônias, a competição imperialista retira as colônias dos impérios mais fracos e as anexa ao Império com maior Poder.

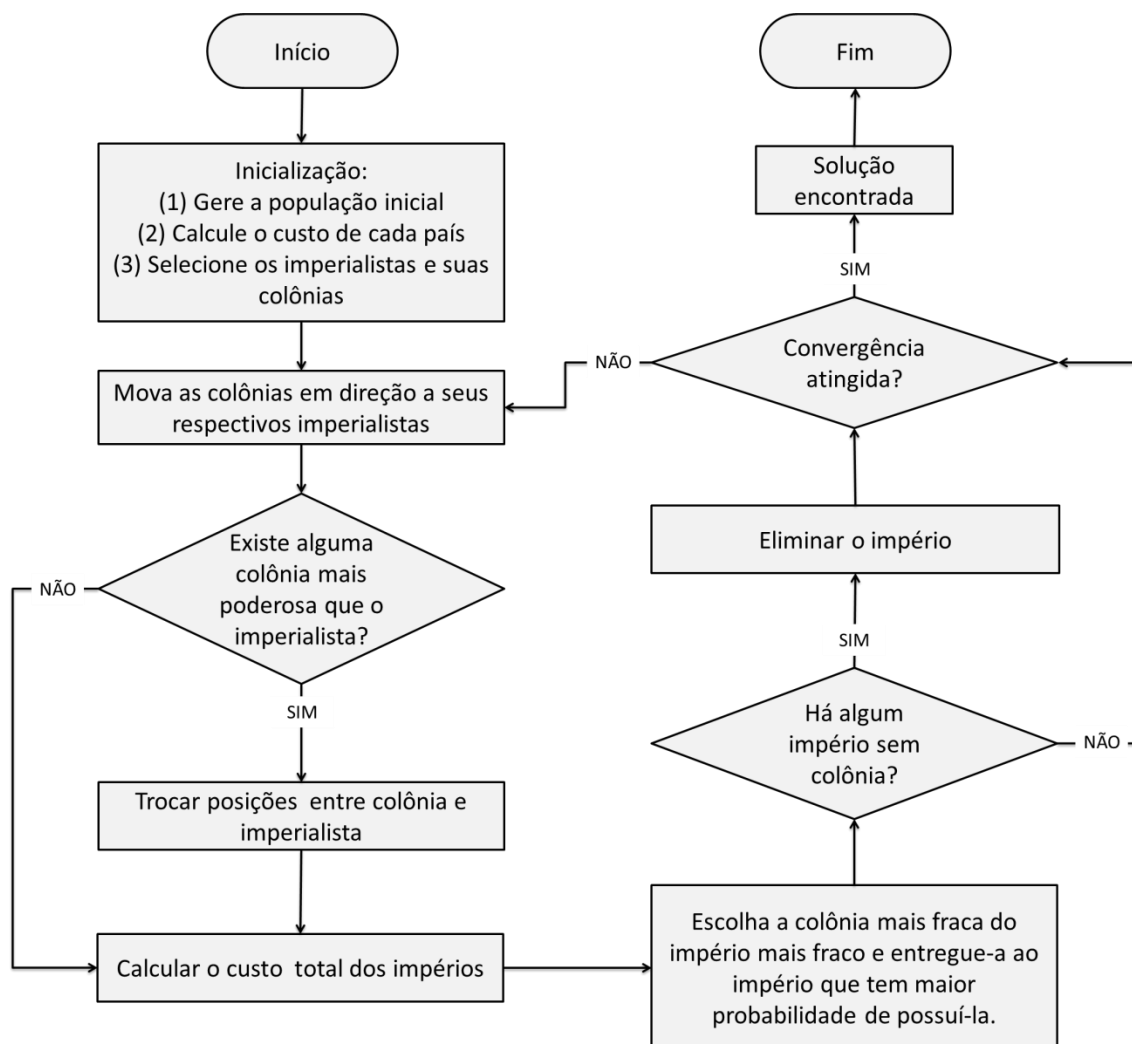
3.1.7 Critério de Parada

Atashpaz-Gargari e Lucas (2007) apontam que ao final do processo de imperialismo haverá apenas um império dominando todas as colônias, porém, todas as colônias serão tão desenvolvidas quanto o império, culminando na idéia de uma sociedade perfeita. No ICA, essa ideia é representada quando há apenas um império localizado no ótimo global possuindo todas as colônias presentes no espaço mundo que possuam o mesmo custo do império ou que o processo de busca atinja o limite de décadas pré-estabelecido (KAVEH; TALATAHARI, 2010).

3.2 FLUXOGRAMA ICA

A Figura 7 a seguir mostra o fluxograma de funcionamento do Algoritmo de Competição Imperialista.

Figura 7 – Fluxograma ICA



Fonte: Elaborado pelo autor.

4. MÉTODOS DE PENALIZAÇÃO

A modelagem matemática tem sido utilizada para o estudo e resolução de muitos problemas no âmbito da engenharia e em outras áreas da ciência. Os problemas de otimização com restrições são abundantes e a abordagem mais comum em algoritmos evolutivos para lidar com restrições é o uso de penalidades, Correia et al. (2008). Neste capítulo são apresentadas diferentes abordagens de penalização aplicadas a métodos de otimização meta-heurísticos, a dar destaque ao Método de Penalização Adaptativa e suas variantes.

4.1 PROBLEMAS COM RESTRIÇÕES

Meta-heurísticas bio-inspiradas são normalmente relacionadas a problemas de otimização sem restrições, em que se procura, para um conjunto de variáveis a minimização (ou maximização) de uma determinada função $f(x)$. Contudo, problemas reais de engenharia muitas vezes envolvem restrições, o que impossibilita a aplicação destes algoritmos em sua forma plena.

As estratégias utilizadas para trabalhar com restrições em algoritmos meta-heurísticos podem ser classificadas em estratégias diretas (interiores), quando apenas elementos factíveis pertencentes ao espaço de busca são considerados, ou indiretas (exteriores), quando ambos os elementos factíveis e infactíveis são utilizados durante o processo de busca.

No primeiro caso, as estratégias diretas incluem: Técnicas de reparo, onde se deseja mover os indivíduos infactíveis para dentro de um domínio factível conhecido; Operadores especiais, um exemplo clássico, são os operadores genéticos criados, para que, no cruzamento de pais factíveis, sempre sejam gerados filhos factíveis; Pena de morte, quando qualquer elemento inviável é descartado do conjunto de soluções candidatas.

Já as estratégias indiretas fazem o uso de: Multiplicadores de Lagrange; Técnicas de Seleção, onde se podem privilegiar os indivíduos factíveis; Técnicas de Penalização, utilizado para transformar um problema de otimização com restrições em um problema irrestrito atribuindo à função objetivo dos indivíduos infactíveis uma penalização; e Métodos Híbridos que podem combinar várias técnicas.

Como as Técnicas de Penalização é objeto de estudo desta dissertação, estas serão detalhadas a fimco nas próximas sessões.

4.2 TÉCNICAS DE PENALIZAÇÃO

As técnicas de penalização foram originalmente propostas por Courant (1943) em meados do século XX e posteriormente expandida por Carroll (1961) e Fiacco (1966). Atualmente, há um grande acervo de publicações sobre o assunto e, algumas delas podem ser consultadas nas seguintes referências: Homaifar et.al (1994); Joines e Houck (1994); Bean e Hadj-Alouane (1992); Smith e Tate (1993); Coit et.al (1996); Kulturel-Konak et al. (2004); Schoenauer e Xanthakis (1993); Barbosa e Lemonge (2008). Deste modo, através do levantamento da literatura é notório identificar que, para o tratamento de problemas restritos, é comum o uso de técnicas de penalização devido a sua fácil implementação.

A ideia principal é transformar problemas de otimização com restrições em problemas irrestritos acoplando à função objetivo uma função de penalização baseado na quantidade de violações das restrições presentes em uma determinada solução.

A função de penalização pode ser formulada de acordo com a equação 27:

$$F(x) = f(x) + p(x) \quad (27)$$

onde, $F(x)$ é a função objetivo expandida a ser minimizada, $f(x)$ é a função objetivo original, e $p(x)$ é a função de penalização descrita conforme equação 28:

$$p(x) = \begin{cases} 0, & \text{se } x \text{ for factível} \\ k \times \sum_{j=1}^m v_j(x), & \text{caso contrário} \end{cases} \quad (28)$$

Sendo que, m é o número total de restrições de igualdade e desigualdade e k uma constante positiva chamada de fator de penalização. Ainda que a função de penalização seja considerada de fácil implementação, essas funções exigem um ajuste cuidadoso do fator de penalização k , que em algumas técnicas é definido em um processo árduo de tentativa e erro, pois este é altamente dependente do problema (RUNARSSON; YAO, 2000). A violação das restrições $v_j(x)$ é definida esta expressa na equação 29:

$$v_j(x) = \begin{cases} |h_j(x)|, & \text{para restrição de igualdade} \\ \max(0, g_j(x)), & \text{caso contrário} \end{cases} \quad (29)$$

onde, $g_j(x)$ são as violações das restrições de desigualdade e $h_j(x)$ são as violações das restrições de igualdade.

Existe na literatura uma diversidade de maneiras para a aplicação das técnicas de penalização, segundo Mezura-Montes e Coello (2011), destacam-se: o método de penalização estática, penalização dinâmica e adaptativa.

4.2.1 Penalização Estática

Na penalização estática a penalidade atribuída aos indivíduos infactíveis permanece inalterada durante todo o processo iterativo. A primeira abordagem ao método estático foi feita por Homaifar (1994), no qual cria-se grupos de indivíduos que violam as restrições dentro de uma mesma faixa de valores e, em cada grupo, define-se um coeficiente de penalização proporcional à violação estabelecida. A equação 30 mostra essa aplicação.

$$F(x) = f(x) + \sum_{i=1}^m R_{k,i} \times \max(0, g_i(x)^2) \quad (30)$$

onde, $F(x)$ é a função objetivo com penalidade, $f(x)$ é a função objetivo sem penalidade, $R_{k,i}$ é o i -ésimo fator de penalização definido de acordo com o nível de violação $k = 1, \dots, l$, sendo l o número de níveis de violação das restrições definidas.

O grande revés na aplicação desta técnica baseia-se no fato do usuário ter que definir inúmeros fatores de penalização de forma empírica.

4.2.2 Penalização Dinâmica

As penalizações dinâmicas vieram como alternativa de suprir a dificuldade do usuário em determinar os coeficientes de penalização, uma desvantagem das penalizações estáticas. Nas penalizações dinâmicas, o período ou instante do processo evolutivo está diretamente envolvido com os coeficientes de penalização.

Um método pioneiro utilizando esse tipo de abordagem foi proposto por Joines e Houck [38] em 1994. Os indivíduos são avaliados conforme a equação 31.

$$F(x) = f(x) + (C \cdot t)^\alpha \times \Omega(\beta, x) \quad (31)$$

onde, as constantes C , α e β são definidas pelo usuário e $\Omega(\beta, x)$ é uma função dependente do nível de violação de um indivíduo x nas restrições do problema e é definida como:

$$\Omega(\beta, x) = \sum_{i=1}^{n_g} g_i(x) + \sum_{j=1}^{n_h} h_j(x) \quad (32)$$

Mesmo tendo evidências de que a penalizações dinâmicas são mais eficientes do que penalizações estáticas, elas exigem a definição de parâmetros, o que, na maioria dos casos, é uma tarefa árdua. Estes parâmetros, quando definidos de forma aleatória e sem análise prévia, podem afetar negativamente o processo evolutivo com convergências prematuras a ótimos locais ou, até mesmo, a não obtenção de soluções factíveis.

4.2.3 Penalização Adaptativa

Nas funções com penalização adaptativa, de acordo com informações coletadas da população, os parâmetros de penalização são atualizados a cada evolução. Bean e Hadje-Alouane (1992) propuseram um método adaptativo em que o fator de penalização tem seu valor alterado a cada evolução, a função objetivo expandida pode ser expressa pela equação 33.

$$F(x) = f(x) + \lambda(t) \left[\sum_{i=1}^{n_g} g_i(x) + \sum_{j=1}^{n_h} |h_j(x)| \right] \quad (33)$$

onde, λ é atualizado a cada geração t da seguinte maneira:

$$\lambda(i+1) = \begin{cases} \frac{1}{\beta_1} \lambda(i), & \text{se } b^i \text{ for sempre factível} \\ \beta_2 \lambda(i), & \text{se } b^i \text{ nunca for factível} \\ \lambda(i), & \text{caso contrário} \end{cases} \quad (34)$$

onde, b^i é atualizado a cada geração i , $\beta_1 \neq \beta_2$ e $\beta_1, \beta_2 > 1$. A adaptabilidade do fator de penalização $\lambda(i+1)$ deste método funciona da seguinte forma, diminui quando os melhores indivíduos das ultimas gerações são factíveis, e aumenta se esses indivíduos são infactíveis, ou então não sofrem alteração.

Hamida e Schoenauer (2000) apresentaram um mecanismo de adaptação ao nível da população. Inicialmente, uma função adaptativa de penalização define os coeficientes de penalização de acordo com a proporção de indivíduos factíveis na população corrente. Logo após, uma estratégia de seleção combina os indivíduos factíveis com os infactíveis, para com isso, explorar a região ao redor da fronteira do domínio factível. Ao final, a seleção é ajustada para privilegiar um determinado número de indivíduos factíveis.

Já em Smith e Tate (1993), a função objetivo expandida $F(x)$ é calculada de acordo com um intervalo, denominado magnitude de penalização, que tem seu valor modificado dinamicamente, levando-se em consideração o valor da aptidão do melhor indivíduo encontrado em uma dada geração. Segue conforme a equação 35.

$$F(x) = f(x) + (b_{factível} - b_{geral}) \times \left[\sum_{i=1}^m \frac{g_i(x)}{NTF(t)} \right]^k \quad (35)$$

onde, $b_{factível}$ é o melhor valor de aptidão da geração t , b_{geral} é o melhor valor geral da função objetivo não penalizada da geração t , $g_i(x)$ é o valor acumulado da violação da restrição i , k é uma constante que determina a severidade da penalização. A função $NTF(t)$ é definido como o limiar da distância da região factível e infactível do espaço de busca.

Mezura-Montes (2009), em seu livro “Constraint Handling in Evolutionary Optimization”, apresenta várias propostas de algoritmos e estratégias de penalização para problemas sujeitos a restrições em várias áreas da engenharia. Dentre os destaques apresentados pelo autor estão as tendências futuras na área de tratamento de restrições em algoritmos evolucionistas. Alguns destes temas são: tratamento de restrições na otimização multi-objetivo; controle de parâmetros ótimos para informar e auxiliar o usuário a adequar os mesmos; sugestões de algoritmos híbridos; entre outros.

Uma técnica de penalização adaptativa que é abordada nesta dissertação, conhecida como APM (*Adaptive Penalty Method*), foi proposta por Barbosa e Lemonge (2002) e será apresentado em detalhes na próxima seção.

4.2.3.1 Método de Penalização Adaptativa (APM)

O Método APM, originalmente acoplado a um algoritmo genético, foi proposto por Barbosa e Lemonge (2002) para tratamento de restrições em problemas de engenharia, os

autores introduziram um método sem nenhuma necessidade do fator de penalização ser definido pelo usuário. Foi desenvolvido um esquema adaptativo que utiliza informações da população, como a média da função objetivo e o nível de violação de cada restrição durante a evolução.

A função de aptidão proposta é escrita como:

$$F(x) = \begin{cases} f(x), & \text{se } x \text{ for factível} \\ \bar{f}(x) + \sum_{j=1}^m k_j \times v_j(x), & \text{caso contrário} \end{cases} \quad (36)$$

onde,

$$\bar{f}(x) = \begin{cases} f(x), & \text{se } f(x) > \langle f(x) \rangle \\ \langle f(x) \rangle, & \text{caso contrário} \end{cases} \quad (37)$$

A equação 37 representa um mecanismo onde, em dado momento o algoritmo apresenta soluções factíveis e infactíveis, caso o valor da função objetivo de uma solução infactível estiver abaixo de $\langle f(x) \rangle$, que é a média das funções objetivo da população corrente, o algoritmo APM altera este valor para o valor médio da função objetivo.

A Figura 8 exemplifica este funcionamento. Dentre o conjunto das seis soluções infactíveis (1, 2,..., 6), os indivíduos 1, 2, 3 e 4 (representados por círculos abertos), têm seus valores de função objetivo menores que $\langle f(x) \rangle$, logo, o valor de $\bar{f}(x) = \langle f(x) \rangle$ para estas soluções. Os indivíduos 5 e 6 têm o valor da função objetivo pior do que a média da população e, assim, $\bar{f}(x) = f(x)$.

A variável $v_j(x)$ armazena o valor da violação do indivíduo x na restrição j e é definido por,

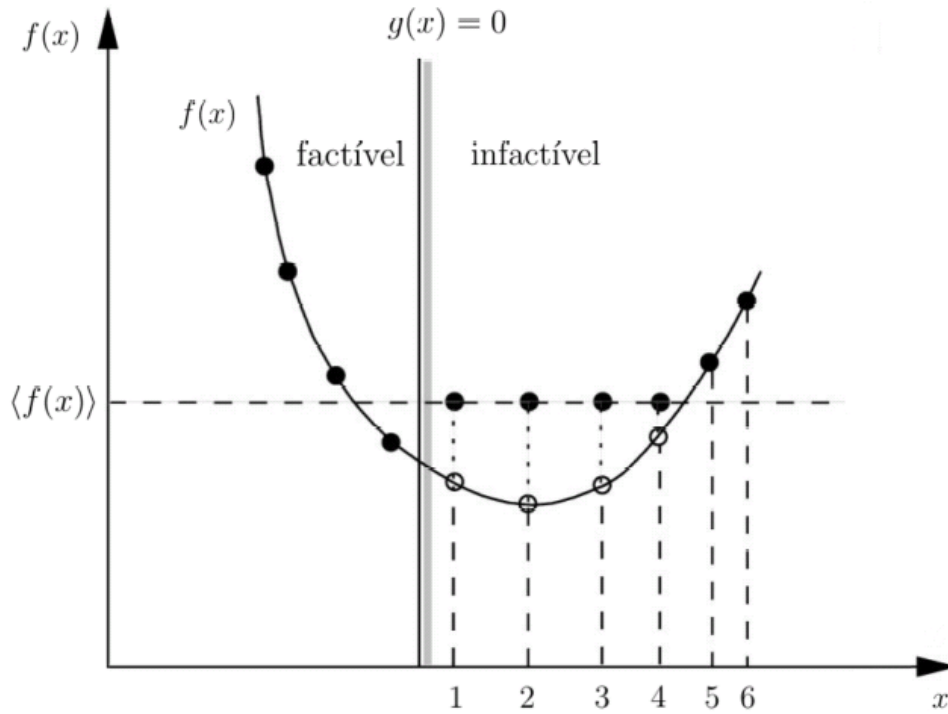
$$v_j(x) = \begin{cases} |h_j(x)|, & \text{para restrição de igualdade} \\ \max(0, g_j(x)), & \text{para restrição de desigualdade} \end{cases} \quad (38)$$

e o fator de penalização da j -ésima restrição k_j foi desenvolvido, de modo que, restrições mais difíceis de serem satisfeitas tenham seu fator de penalização atribuídos a valores mais altos, assim penaliza-se mais fortemente indivíduos infactíveis naquela região.

$$k_j = \langle f(x) \rangle \times \frac{\langle v_j(x) \rangle}{\sum_{l=1}^m [\langle v_l(x) \rangle]^2} \quad (39)$$

onde, $\langle v_j(x) \rangle$ é a média dos valores da l -ésima restrição na população corrente.

Figura 8 – Determinação de $\bar{f}(x)$.



Fonte: Adaptado de Barbosa e Lemonge [34].

Com o intuito de redução de custos computacionais, Barbosa e Lemonge (2002) propuseram algumas variantes ao Método APM, sendo todas testadas pelos autores. Dentre as variantes propostas, duas mostram-se mais eficientes, o Método APM Esporádico e o Método APM Esporádico com Acúmulo das Violações. Estes são detalhados a seguir.

4.2.3.2 APM Esporádico

Na técnica de penalização (APM_Spor), os fatores de penalização k_j são mantidos constantes por uma quantidade fixa de gerações, a ser definido pelo usuário. Desta maneira, o cálculo dos fatores de penalização não é necessário a cada geração reduzindo assim o custo computacional. O método segue as seguintes etapas:

- (i) calcula-se as violações das restrições v_j na população corrente;

(ii) atualiza-se os fatores de penalidade k_j ;

(iii) Mantém os parâmetros de penalização k_j inalterados por uma quantidade fixa de gerações.

4.2.3.3 APM Esporádico com Acúmulo das Violações

Nesta técnica (APM_Spor_Acum), como o próprio nome diz, acumula-se os valores das violações v_j durante um valor fixo de gerações, também a ser definido pelo usuário; e mantém-se o fator de penalização k_j fixo durante o mesmo número de gerações. O método segue as seguintes etapas:

(i) acumula-se os valores das violações por uma quantidade fixa de gerações;

(ii) atualiza-se os fatores de penalidade k_j ;

(iii) Mantém os parâmetros de penalização k_j inalterados por uma quantidade fixa de gerações

5. RESULTADOS NUMÉRICOS

Neste capítulo são detalhados alguns problemas clássicos da Engenharia mecânica e avalia-se o perfil de desempenho, através do método de análise de desempenho proposto por Dolan e More (2002), do Algoritmo de Competição Imperialista acoplado a diferentes técnicas de penalização descritas no capítulo 4.

5.1 PARÂMETROS DO ALGORITMO ICA

O Algoritmo de Competição Imperialista ICA foi adotado como algoritmo de busca no presente estudo. As programações foram desenvolvidas em rotinas Matlab® 2015 e implementados num computador com sistema operacional Microsoft® Windows 7 Professional, processador Intel® Core™ i3, frequência do CPU de 2.13 GHz e 3 GB de RAM.

O algoritmo foi ajustado com os seguintes parâmetros:

- (i) Coeficiente de assimilação $\beta = 1.05$;
- (ii) Coeficiente angular de assimilação $\gamma = 10$;
- (iii) Coeficiente de posição $\varepsilon = 0,003$;
- (iv) Taxa de revolução $T_{rev} = 0,1$;
- (v) Número de décadas $N_{décadas} = 500$.

Foram acoplados ao algoritmo ICA as técnicas de penalização: Penalização Estática (SPM); Penalização Dinâmica (DPM); Penalização Adaptiva (APM); APM Esporádico (APM_Spor) e APM Esporádico com Acúmulo de Violações (APM_Spor_Acum). Para cada técnica foram realizados um total de 100 execuções.

Devido à singularidade de cada problema proposto para análise numérica, utilizaram-se diferentes valores de número de países (Np) e número de imperialistas (N_{imp}).

O critério de parada do algoritmo foi ajustado para ser satisfeito, se somente se, os países convergirem para um único império ou quando o processo de busca atingir o limite de décadas pré-estabelecido.

5.2 MÉTODO DE ANÁLISE DE DESEMPENHO

O método adotado para analisar e comparar o desempenho dos algoritmos, desenvolvidos neste trabalho, foi desenvolvido por Dolan e Moré (2002). Os autores o criaram para com o objetivo de facilitar a visualização e interpretação dos resultados obtidos em experimentos, comparando-se um conjunto de algoritmos de modo a identificar o com melhor desempenho aplicado ao um conjunto de problemas.

O método considera um conjunto P de problemas teste p_j , com $j = 1, 2, \dots, n_p$, um conjunto de algoritmos a_i , com $i = 1, 2, \dots, n_a$ e uma métrica de desempenho $t_{p,a}$ (tempo computacional, média dos valores da função objetivo, etc.).

A razão de desempenho (sempre maior ou igual a 1), é definida como:

$$r_{p,a} = \frac{t_{p,a}}{\min(t_{p,a} : a \in A)} \quad (40)$$

O perfil de desempenho do algoritmo é dado por:

$$\rho_a(\tau) = \frac{1}{n_p} |(p \in P : r_{p,a} \leq \tau)| \quad (41)$$

onde, $\rho_a(\tau)$ é a fração de problemas resolvidos pelo algoritmo com desempenho dentro de um fator τ do melhor desempenho obtido, considerando todos os algoritmos.

Barbosa et al. (1988) destaca algumas relações a serem observados de acordo com os resultados de razão e perfil de desempenho. O algoritmo com menor valor de τ , tal que $\rho_a(\tau) = 1$, é o que consegue resolver todos os problemas e também é o que obtém os melhores resultados na maioria dos problemas. Já o algoritmo que atinge o maior valor para $\rho_a(1)$ obtém melhores resultados em um número maior de problemas. Por fim, a área sob a curva ρ_a é um indicador global de desempenho do algoritmo, onde, quanto maior sua área, maior é o desempenho.

5.3 MOLA DE TRAÇÃO/COMPRESSÃO

Para formular o problema de projetar uma mola helicoidal submetida a uma carga axial, veja a notação dos dados de projeto na Tabela 1.

Tabela 1 – Informações do projeto de mola.

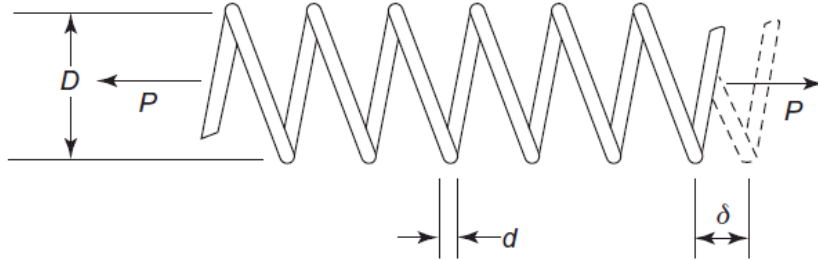
Notação	Dado
Deflexão da mola	• δ, in
Diâmetro médio da mola	• D, in
Diâmetro do fio da mola	• d, in
Número de espiras ativas da mola	• N
Aceleração da gravidade	• $g = 386 in/s^2$
Frequência natural	• ω, Hz
Densidade de peso do material	• $\gamma = 0.285 lb/in^3$
Módulo de cisalhamento	• $G = (1.15 \times 10^7) lb/in^2$
Densidade do material $\rho = \gamma/g$	• $\rho = (7.38342 \times 10^{-4}) lb.s^2/in^4$
Tensão admissível de cisalhamento	• $\tau_a = 80 lb/in^2$
Número de espirais inativas	• $Q = 2$
Carga axial	• $P = 10 lb$
Deflexão mínima	• $\Delta = 0.5 in$
Limite inferior da frequência natural	• $\omega_o = 100 Hz$
Limite superior diâmetro médio da mola	• $D_o = 1.5 in$

Fonte: Adaptado de Arora [46]

O propósito deste problema é projetar uma mola de volume mínimo, mostrada na Figura 9, para suportar uma determinada carga axial (chamada de mola de compressão ou tensão) sem falha de material e satisfazendo dois requisitos de desempenho: A mola deve ter

uma deflexão mínima (Δ) e a frequência de pico não deve ser menor que a frequência natural (ω_o) [46].

Figura 9 – Projeto de mola sob tração ou compressão.



Fonte: Adaptado de Arora (2012)

A mola deforma quando é submetida a uma carga de tração ou compressão. Portanto, a tensão de cisalhamento precisa ser calculada para que uma restrição sobre ela possa ser incluída na formulação. Além disso, a frequência natural precisa ser calculada. Estas e outras equações de projeto para a mola são dadas como:

Equação do carregamento:

$$P = K\delta \quad (42)$$

Constante elástica:

$$K = \frac{d^4 G}{8D^3 N} \quad (43)$$

Tensão de cisalhamento:

$$\tau = \frac{8kPD}{\pi d^3} \quad (44)$$

Fator de concentração de tensão:

$$k = \frac{(4D-d)}{4(D-d)} + \frac{0.615d}{D} \quad (45)$$

Frequência de pico:

$$\omega = \frac{d}{2\pi ND^2} \sqrt{\frac{G}{2\rho}} \quad (46)$$

Para definição da função objetivo e as restrições de projeto são consideradas as seguintes variáveis de projeto, a espessura da mola [d], o diâmetro helicoidal [D] e o número de espirais ativas [N]. A massa da mola pode ser obtida pela expressão:

$$Mass = \frac{1}{4} (N + Q)\pi^2 D d^2 \rho \quad (47)$$

Como as variáveis de projeto foram definidas apenas em razão de d, D e N podemos ignorar a constante $(\pi^2 \rho / 4)$, assim o problema de otimização resume-se em minimizar apenas o volume da mola:

$$f(d, D, N) = (N + 2)Dd^2 \quad (48)$$

Sujeito às seguintes restrições:

Restrição de deflexão: padrão para projeto de molas.

$$g_1 = \frac{P}{K} \geq \Delta \quad (49)$$

$$g_1 = 1 - \frac{D^3 N}{(71875 d^4)} \leq 0 \quad (50)$$

Restrição de tensão de cisalhamento: para evitar a sobrecarga do material, a tensão de cisalhamento no material não deve ser maior que τ_a , que é expressa na forma matemática como,

$$g_2 = \tau \leq \tau_a \quad (51)$$

$$g_2 = \frac{D(4D-d)}{12566d^3(D-d)} + \frac{2.46}{12566d^2} - 1 \leq 0 \quad (52)$$

Restrição de vibração: para evitar a ressonância em aplicações dinâmicas deseja-se que a frequência natural (ao longo da mola) seja o maior possível. Para o problema atual exige-se a frequência natural da mola seja pelo menos ω_0 (Hz). A restrição é expressa na forma matemática como,

$$g_3 = \omega \geq \omega_0 \quad (53)$$

$$g_3 = 1 - \frac{140.45d}{D^2 N} \leq 0 \quad (54)$$

Restrição de diâmetro médio da mola: pode representar, por exemplo, uma restrição de fabricação.

$$g_4 = D + d \leq D_o \quad (55)$$

$$g_4 = \frac{D+d}{1,5} - 1 \leq 0 \quad (56)$$

Restrições de fronteira para cada variável de projeto:

$$0.05 \leq d \leq 0.2$$

$$0.25 \leq D \leq 1.3 \quad (57)$$

$$2 \leq N \leq 15$$

Na aplicação numérica foram realizadas 200 execuções de cada versão do algoritmo de busca, totalizando em 50000 o número de avaliações da função objetivo. A Tabela 3 apresenta o estudo comparativo entre as diferentes técnicas de penalização acopladas ao ICA, para o projeto de mola de tração/compressão. Desta forma, mostram-se os melhores valores obtidos por cada método, a média das execuções, o desvio padrão e o pior resultado encontrado dentre todas as execuções.

Tabela 2 – Resultados numéricos para o projeto de mola.

Métodos	Melhor	Média	Desvio	Pior
SPM	0.012667548911	0.012775226073	6.701e-05	0.013167346826
DPM	0.012665245582	0.013122384465	9.348e-04	0.017773158097
APM	0.012665249757	0.012691813643	3.282e-05	0.012795436098
APM_Spor	0.012665280378	0.012748847005	1.708e-04	0.013618638471
APM_Acum	0.012665244703	0.012835376137	4.062e-04	0.015736028527

Fonte: Elaborado pelo autor.

Observa-se que todas as técnicas de penalização acopladas ao ICA obtiveram bons resultados, permanecendo iguais até a quarta casa decimal. O APM_Acum obteve o melhor resultado; o APM obteve a melhor média e valor de desvio padrão; e o DPM obteve o pior valor entre as 200 execuções.

Na Tabela 3 são apresentadas as variáveis de projeto para o melhor resultado para o problema de mola de tração/compressão

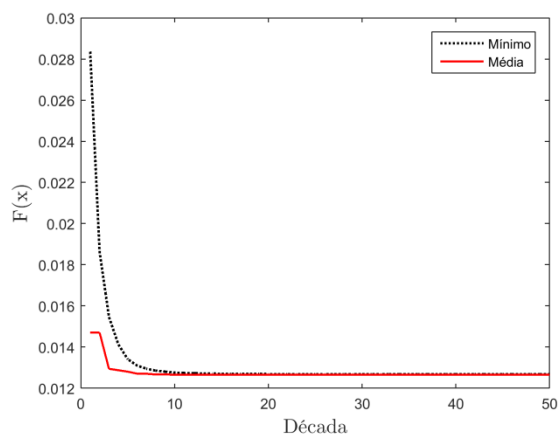
Tabela 3 – Melhor solução para o projeto de mola.

Variáveis	APM_Acum
d	0.051672553196
N	0.356320609731
D	11.312290930660
$F(d, N, D)$	0.012665244703

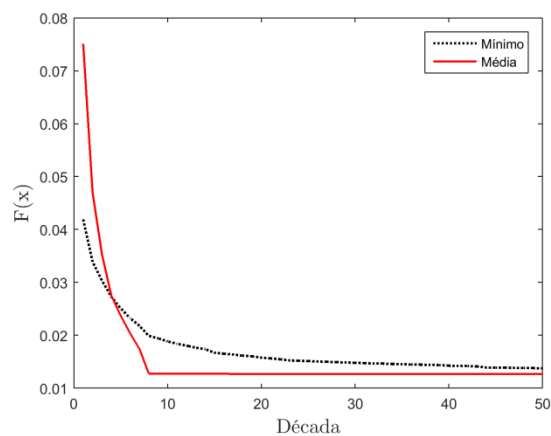
Fonte: Elaborado pelo autor.

A seguir é mostrado a convergência deste problema em função do número de décadas obtida pelo ICA com as diferentes técnicas de penalização adotadas. Em vermelho está representado o valor mínimo encontrado a cada iteração, a linha pontilhada representa a média dos valores avaliados a cada iteração. A Figura 10(a) representa o Método de Penalização Estático, a Figura 10(b) o Método de Penalização Dinâmico, a Figura 10(c) o Método de Penalização Adaptativo, a Figura 10(d) o Método de Penalização Adaptativo Esporádico e a Figura 10(e) o Método de Penalização Adaptativo Esporádico.

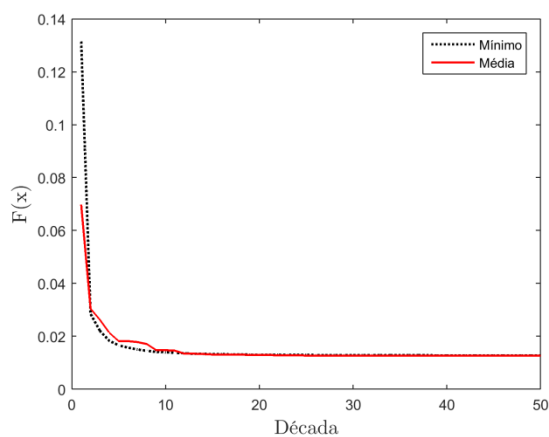
Figura 10 – Comparativo das convergência mínima e média.



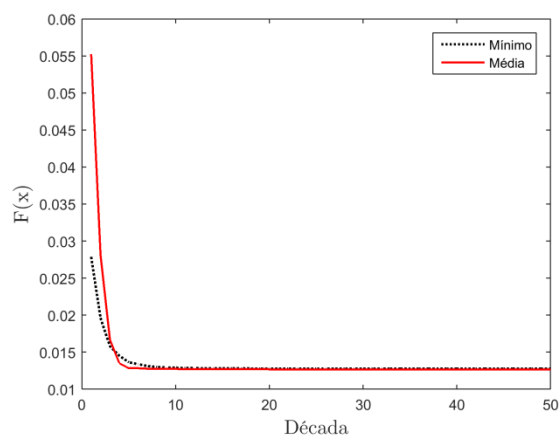
(a) Método SPM



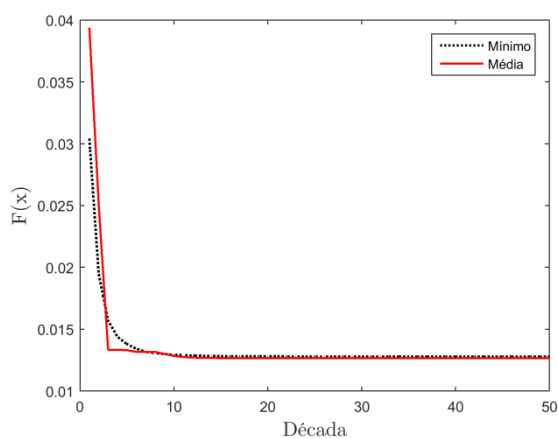
(b) Método DPM



(c) Método APM



(d) Método APM_Spor

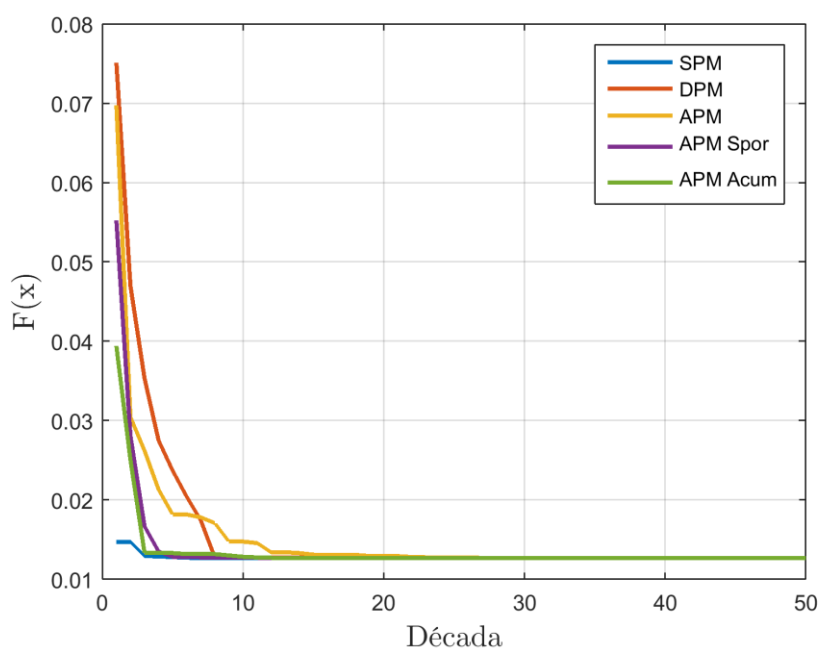


(e) Método APM_Acum

Fonte: Elaborado pelo autor.

Também a Figura 11 mostra em um mesmo gráfico a convergência de todas as técnicas de penalização acopladas ao Algoritmo de Competição Imperialista.

Figura 11 – Convergência dos métodos de penalização.

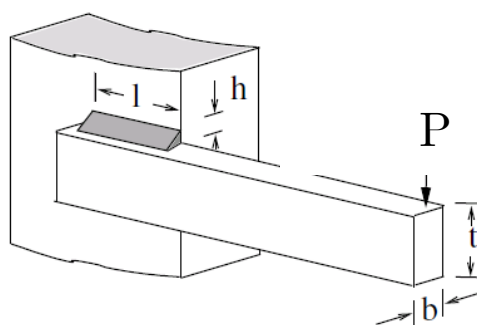


Fonte: Elaborado pelo autor.

5.4 Viga Engastada

O propósito deste problema é projetar uma viga com mínimo custo considerando restrições de tensão de cisalhamento, tensão normal da viga, esforço por carga na extremidade livre, deflexão máxima e restrições de fronteira, conforme mostrado na Figura 12.

Figura 12 – Projeto de viga engastada.



Fonte: Adaptado de Barbosa e Lemonge (2002).

Para formular o problema de projetar uma viga engastada submetida a uma carga na extremidade livre, segue a notação dos dados de projeto na tabela 4.

Tabela 4 – Informações do projeto de viga engastada.

Notação	Dado
Espessura da junta soldada	• h, in
Comprimento da junta de soldada	• l, in
Largura da viga	• t, in
Espessura da viga	• b, in
Módulo de Young	• $E = (30.10^6) psi$
Módulo de cisalhamento	• $G = (12 \times 10^6) psi$
Comprimento livre da viga	• $L = 14 in$
Tensão de cisalhamento máxima da solda	• $\tau_{m\acute{a}x} = 13600 psi$
Tensão de flexão máxima da viga	• $\sigma_{m\acute{a}x} = 30000 psi$
Deflexão máxima da viga	• $\delta_{m\acute{a}x} = 0.25 in$
Carregamento	• $P = 6000 lb$
Custo do material de solda por unidade de volume	• $C_1 = 0.10471 \$/in^3$
Custo do material da viga por unidade de volume	• $C_2 = 0.04811 \$/in^3$
Custo de mão de obra por unidade de volume de solda	• $C_3 = 1 \$/in^3$

Fonte: Elaborado pelo autor.

As funções de volume de material de solda e de viga utilizados no projeto correspondem às seguintes equações

$$V_{solda} = h^2 l \quad (58)$$

$$V_{viga} = tb(L + l) \quad (59)$$

Assim a função objetivo de custo do projeto pode ser expressa da seguinte forma

$$f(h, l, t, b) = (C_1 + C_3) \cdot V_{solda} + C_2 V_{viga} \quad (60)$$

Substituindo os valores das constantes do projeto obtemos:

$$f(h, l, t, b) = 1.1047h^2 l + 0.04811tb(14 + l) \quad (61)$$

Sujeito às seguintes restrições:

Restrição de tensão de cisalhamento: a tensão de cisalhamento no local de suporte do feixe não pode exceder o máximo permitido para o material.

$$g_1 = \tau - \tau_{m\acute{a}x} \leq 0 \quad (62)$$

Restrição de tensão de flexão: a tensão de flexão normal na posição de suporte do feixe não pode exceder a força máxima de escoamento para o material.

$$g_2 = \sigma - \sigma_{m\acute{a}x} \leq 0 \quad (63)$$

Restrição de projeto: a espessura do membro deve ser maior que a espessura da solda e a espessura da solda deve ser maior que o mínimo definido.

$$g_3 = h - b \leq 0 \quad (64)$$

$$g_4 = C_1 h^2 + C_2 tb(L + l) - 5 \leq 0 \quad (65)$$

$$g_5 = 0.125 - h \leq 0 \quad (66)$$

Restrição de deflexão da viga: a deflexão não pode exceder a deflexão máxima.

$$g_6 = \delta - \delta_{m\acute{a}x} \leq 0 \quad (67)$$

Restrição de carregamento: a carga aplicada não deve exceder o carregamento máximo.

$$g_7 = P - P_c \leq 0 \quad (68)$$

Restrições de fronteira:

$$\begin{aligned} 0.1 &\leq h \leq 2 \\ 0.1 &\leq l \leq 10 \\ 0.1 &\leq t \leq 10 \\ 0.1 &\leq b \leq 2 \end{aligned} \quad (69)$$

As expressões τ , σ , δ e P_c são dadas por

$$\tau = \sqrt{(\tau')^2 + \frac{2\tau'\tau''l}{2R} + (\tau'')^2} \quad (70)$$

$$\tau' = \frac{P}{\sqrt{2}hl} \quad (71)$$

$$\tau'' = \frac{MR}{J} \quad (72)$$

$$M = P \left(L + \frac{l}{2} \right) \quad (73)$$

$$R = \sqrt{\frac{l^2}{4} + \left(\frac{h+t}{2} \right)^2} \quad (74)$$

$$J = 2 \left\{ \sqrt{2}hl \left[\frac{l^2}{12} + \left(\frac{h+t}{2} \right)^2 \right] \right\} \quad (75)$$

$$\sigma = \frac{6PL}{bt^2} \quad (76)$$

$$\delta = \frac{4PL^3}{Ebt^3} \quad (77)$$

$$P_c = \frac{4.013E\sqrt{\frac{t^3b^6}{36}}}{L^2}\left(1 - \frac{t}{2L}\sqrt{\frac{E}{4G}}\right)$$

(78)

Na aplicação numérica foram realizadas 200 execuções de cada versão do algoritmo de busca, totalizando em 50000 o número de avaliações da função objetivo. A Tabela 5 apresenta o estudo comparativo entre as diferentes técnicas de penalização acopladas ao ICA, para o projeto de mola de tração/compressão. Desta forma, mostra-se os melhores valores obtidos por cada método, a média das execuções, o desvio padrão e o pior resultado encontrado dentre todas as execuções.

Tabela 5 – Resultados numéricos para o projeto de viga.

Métodos	Melhor	Média	Desvio	Pior
SPM	2.534481803802	2.539166402840	0.0061	2.565647679489
DPM	2.534505831086	2.538551089300	0.0047	2.554714408739
APM	2.534506955563	2.546433143249	0.0134	2.584509105145
APM_Spor	2.534480986160	2.536907827552	0.0031	2.549424290484
APM_Acum	2.534482460598	2.538479586204	0.0063	2.566047523053

Fonte: Elaborado pelo autor.

Observa-se que todas as técnicas de penalização acopladas ao ICA obtiveram bons resultados, e assim como no projeto de viga, os resultados permaneceram em média iguais até a quarta casa decimal. O APM_Spor foi a técnica que sobressaiu, quando aplicado ao problema de viga engastada obteve o melhor resultado, a melhor média e desvio padrão. O APM obteve o pior valor entre as 200 execuções.

Na Tabela 6 são apresentadas as variáveis de projeto para do melhor resultado para o problema de mola de tração/compressão

Tabela 6 – Melhor solução para o projeto de viga.

Variáveis	APM_Spor
h	0.201671569308
l	8.178462551292
t	8.291471454191
b	0.244368972046
$F(h, l, t, b)$	2.534480986160

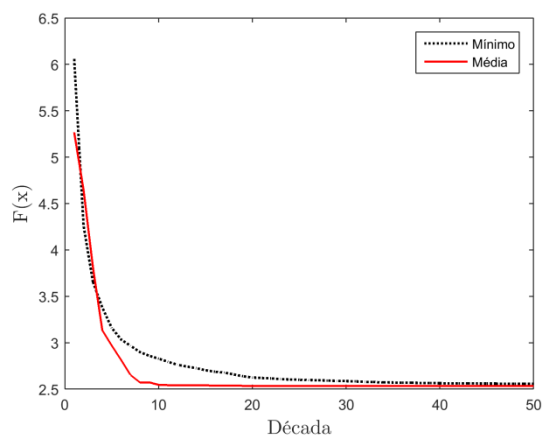
Fonte: Elaborado pelo autor.

A seguir é mostrado a convergência deste problema em função do número de décadas obtida pelo ICA com as diferentes técnicas de penalização adotadas.

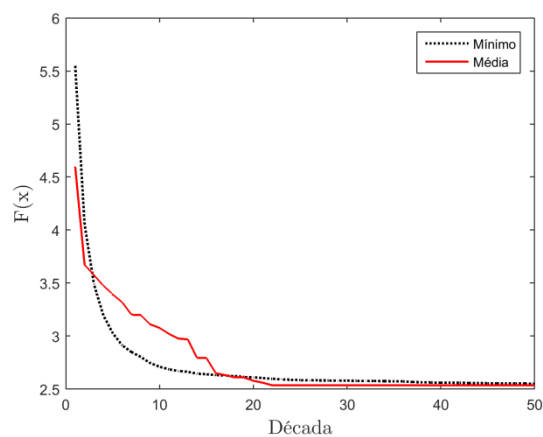
Em vermelho está representado o valor mínimo encontrado a cada iteração, a linha pontilhada representa a média dos valores avaliados a cada iteração.

A Figura 13(a) representa o Método de Penalização Estático, a Figura 13(b) o Método de Penalização Dinâmico, a Figura 13(c) o Método de Penalização Adaptativo, a Figura 13(d) o Método de Penalização Adaptativo Esporádico e a Figura 13(e) o Método de Penalização Adaptativo Esporádico.

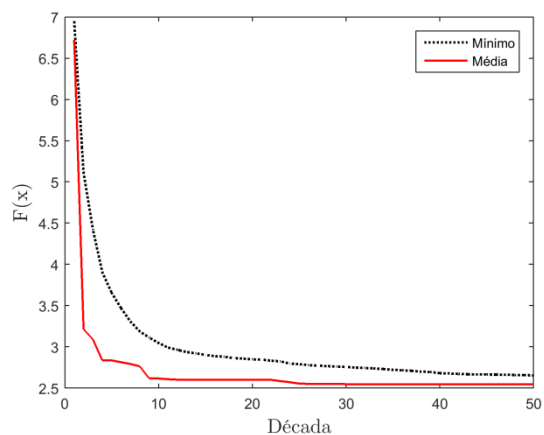
Figura 13 – Comparativo das convergências mínima e média.



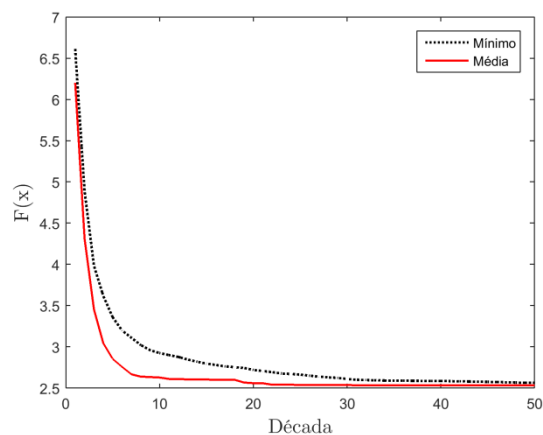
(a) Método SPM



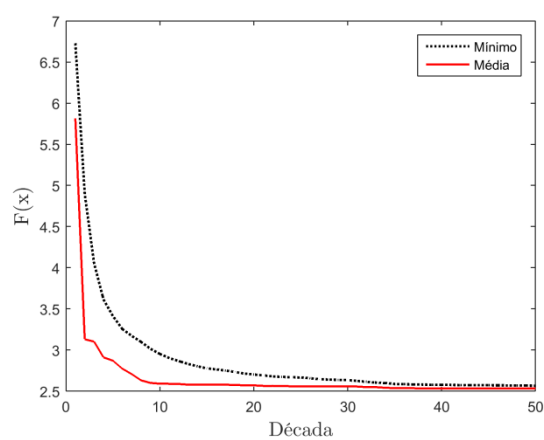
(b) Método DPM



(c) Método APM



(d) Método APM_Spor

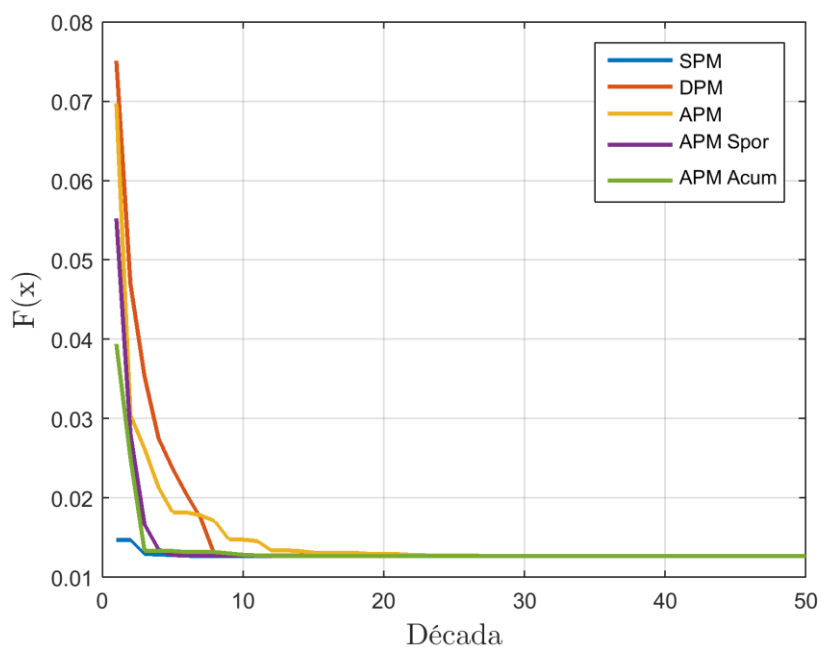


(e) Método APM_Acum

Fonte: Elaborado pelo autor.

Também a Figura 14 mostra a em um mesmo gráfico a convergência de todas as técnicas de penalização acopladas ao Algoritmo de Competição Imperialista.

Figura 14 – Convergência dos métodos de penalização.

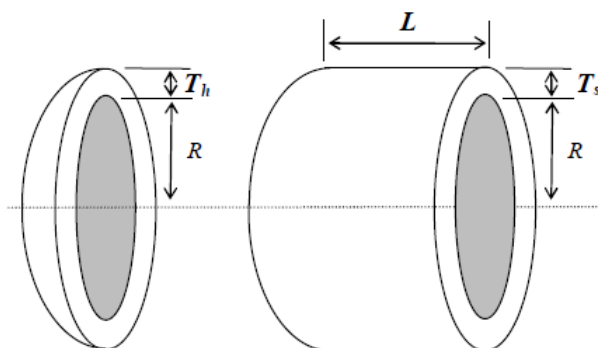


Fonte: Elaborado pelo autor.

5.5 Vaso de Pressão

O propósito deste problema é projetar um vaso de pressão de tampas esféricas onde o custo deve ser minimizado, conforme mostrado na Figura 15.

Figura 15 – Projeto de vaso de pressão.



Fonte: Elaborado pelo autor.

Este problema foi originalmente proposto por Sandgren [48] para o projeto de um vaso de pressão com custo total mínimo (material, montagem e soldagem). Para formular o

problema de projetar uma viga engastada submetida a uma carga na extremidade livre, segue a notação dos dados de projeto na Tabela 7.

Tabela 7 – Informações do projeto de vaso de pressão.

Notação	Dado
Espessura da casca	• T_s, in
Espessura da tampa	• T_h, in
Raio interno	• R, in
Comprimento da seção cilíndrica	• L, in

Fonte: Elaborado pelo autor.

O custo pode ser minimizado seguindo a seguinte função objetivo de custo:

$$f(T_s, T_h, R, L) = 0,6224T_sT_hR + 1,7881T_hR^2 + 3,1661T_s^2L + 19,84T_s^2R \quad (79)$$

Sujeito às seguintes restrições

Restrição de tensão normal: a tensão normal no arco não pode ser maior que a tensão admissível normal.

$$g_1 = -T_s + 0,0193R \leq 0 \quad (80)$$

Restrição de tensão de cisalhamento: a tensão longitudinal não deve ser maior que a tensão admissível de cisalhamento.

$$g_2 = -T_h + 0,00954R \leq 0 \quad (81)$$

Restrição de volume e comprimento:

$$g_3 = 1296000 - \pi R^2L - \frac{4}{3}\pi R^3 \leq 0 \quad (82)$$

$$g_4 = L - 240 \leq 0 \quad (83)$$

Restrições de fronteira:

$$0.0625 \leq T_s \leq 5$$
$$0.0625 \leq T_h \leq 5$$
$$10 \leq R \leq 200$$
$$10 \leq L \leq 200$$

(84)

Na aplicação numérica foram realizadas 200 execuções de cada versão do algoritmo de busca, totalizando em 50000 o número de avaliações da função objetivo.

A Tabela 8 apresenta o estudo comparativo entre as diferentes técnicas de penalização acopladas ao ICA, para o projeto de mola de tração/compressão. Desta forma, mostra-se os melhores valores obtidos por cada método, a média das execuções, o desvio padrão e o pior resultado encontrado dentre todas as execuções.

Tabela 8 – Resultados numéricos para o projeto de vaso de pressão.

Métodos	Melhor	Média	Desvio	Pior
SPM	6059.715265100	6168.380541283	1.196e+02	6793.80541283
DPM	6060.754651450	6066.011678928	10.482	6105.446041631
APM	6059.715231565	6606.300145358	1.087e+02	7338.953242247
APM_Spor	6059.715256684	6581.110464615	1.204e+02	7335.545279648
APM_Acum	6059.715235645	6584.382463019	2.080e+02	7333.945183298

Fonte: Elaborado pelo autor.

Observa-se que todas as técnicas de penalização acopladas ao ICA obtiveram bons resultados, permanecendo iguais até a terceira casa decimal. O APM obteve o melhor e o pior resultado; o DPM obteve a melhor média e valor de desvio padrão.

Na Tabela 9 são apresentadas as variáveis de projeto para do melhor resultado para o problema de mola de tração/compressão

Tabela 9 – Melhor solução para o projeto de vaso de pressão.

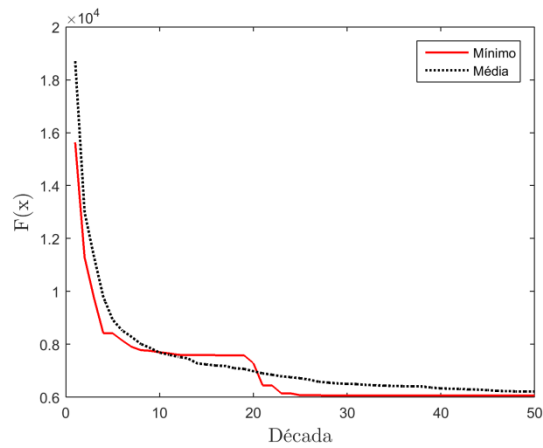
Variáveis	APM_Spor
h	0.847082513585
l	0.443873545144
t	42.09843815564
b	176.6366871566
$F(h, l, t, b)$	6059.715231565

Fonte: Elaborado pelo autor.

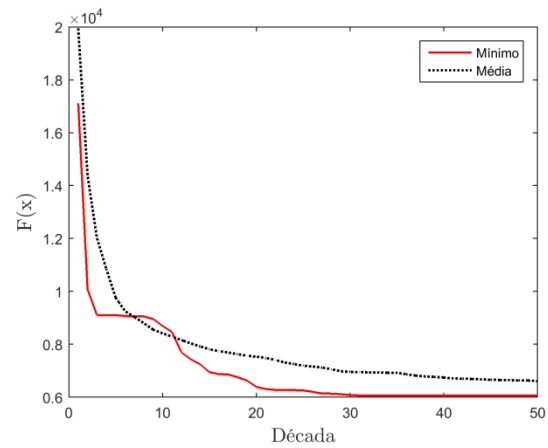
A seguir é mostrado a convergência deste problema em função do número de décadas obtida pelo ICA com as diferentes técnicas de penalização adotadas. Em vermelho está representado o valor mínimo encontrado a cada iteração, a linha pontilhada representa a média dos valores avaliados a cada iteração.

A Figura 16(a) representa o Método de Penalização Estático, a Figura 16(b) o Método de Penalização Dinâmico, a Figura 16(c) o Método de Penalização Adaptativo, a Figura 16(d) o Método de Penalização Adaptativo Esporádico e a Figura 16(e) o Método de Penalização Adaptativo Esporádico.

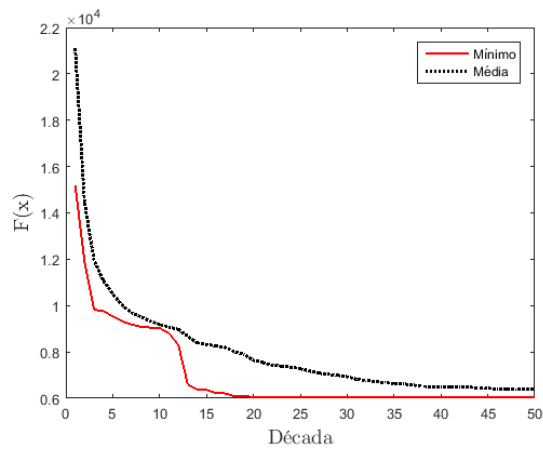
Figura 16 – Comparativo das convergência mínima e média



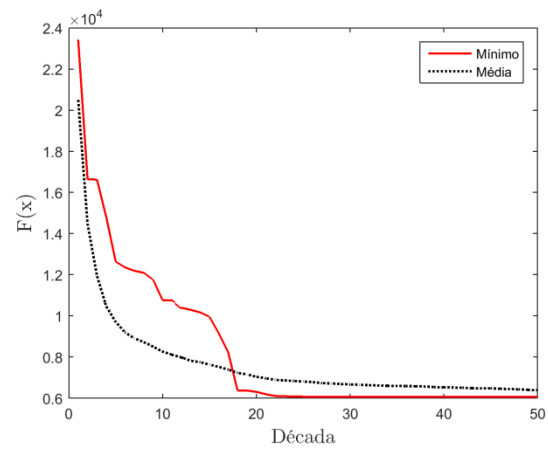
(a) Método SPM



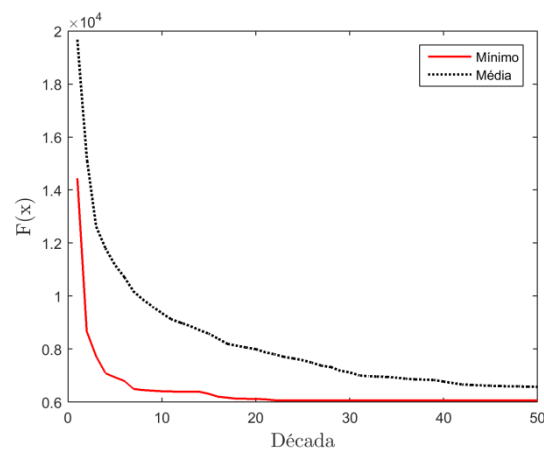
(b) Método DPM



(c) Método APM



(d) Método APM_Spor

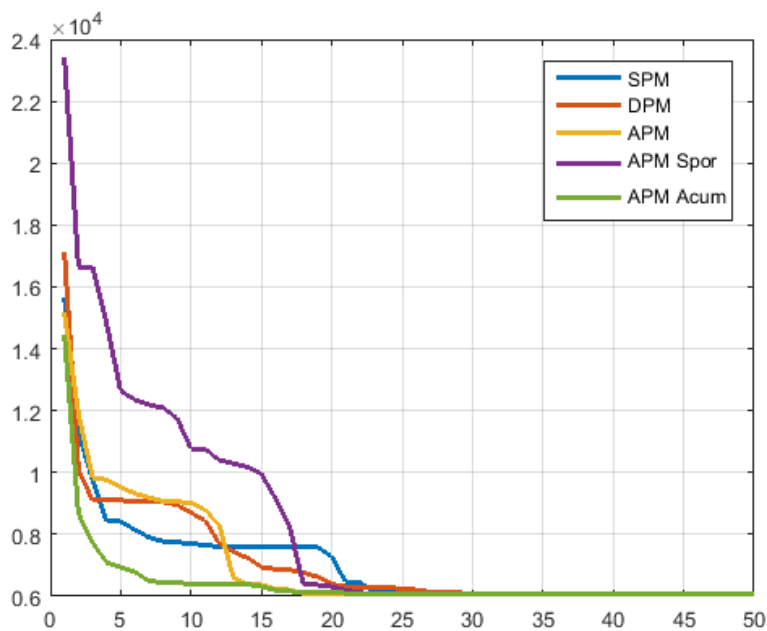


(e) Método APM_Acum

Fonte: Elaborado pelo autor.

Também a Figura 17 mostra a em um mesmo gráfico a convergência de todas as técnicas de penalização acopladas ao Algoritmo de Competição Imperialista.

Figura 17 – Convergência dos métodos de penalização.



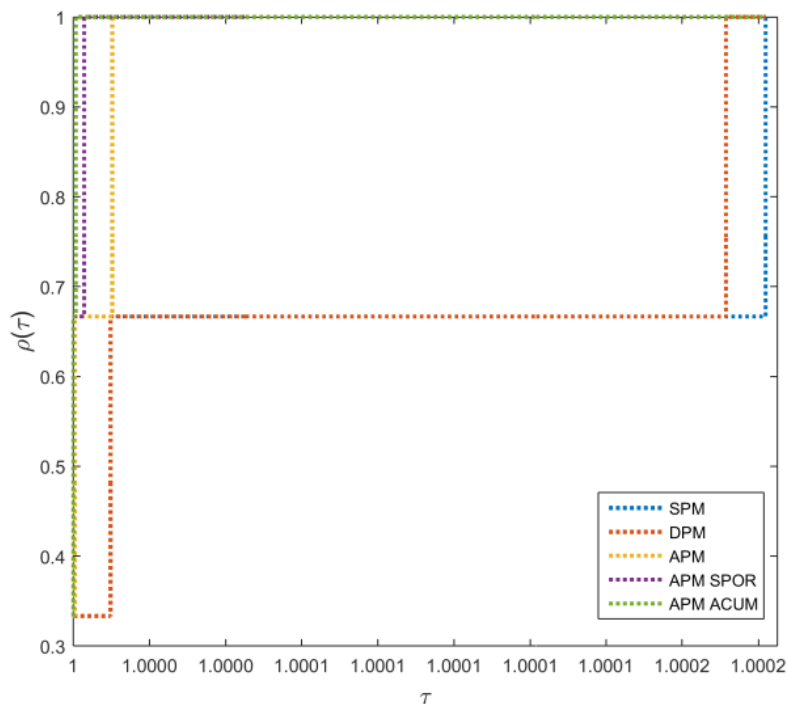
Fonte: Elaborado pelo autor.

5.6 Análise de Desempenho

De acordo com os resultados numéricos obtidos foi feita a análise dos perfis de desempenho considerando os valores de média e melhor resultado obtido por cada algoritmo em cada problema proposto.

Ao analisar os perfis de desempenho quando considerado o melhor resultado dado pela Figura 18, percebe-se que a técnica de penalização APM_Acum foi a que obteve os resultados satisfatórios para todos os problemas, pois obteve o maior de $\rho(\tau)$ para $\tau = 1$. Neste caso entende-se satisfatório como um valor muito próximo do melhor resultado encontrado.

Figura 18 – Perfis de desempenho utilizando os melhores resultados obtidos.



Fonte: Elaborado pelo autor.

Observa-se também pela Tabela 10 ao analisar as áreas dos gráficos de perfil de desempenho que o algoritmos de maior confiabilidade foram o APM_Acum, APM_Spor e APM.

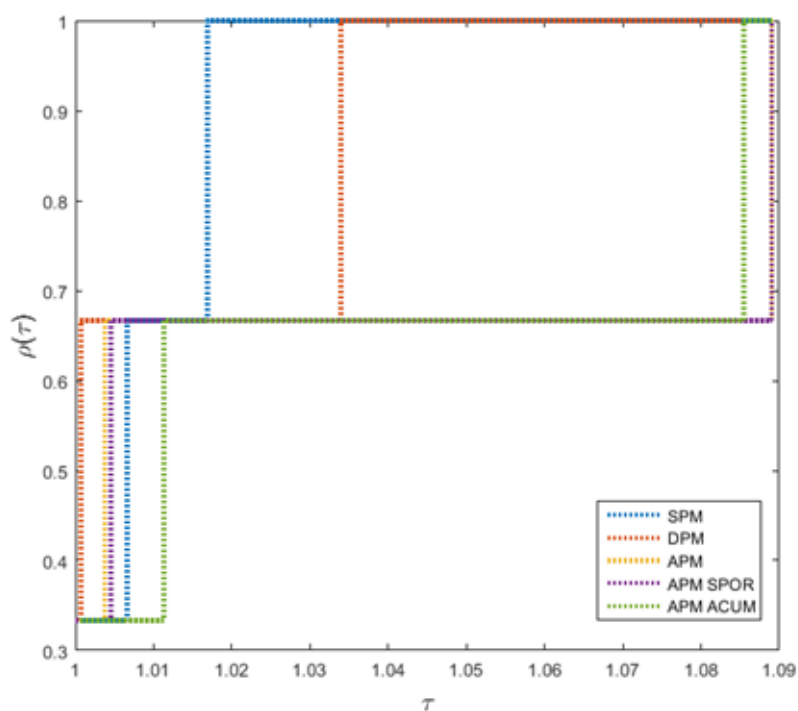
Tabela 10 – Área normalizada dos perfis de desempenho utilizando os melhores resultados obtidos.

Técnica	SPM	DPM	APM	APM_Spor	APM_Acum
Área	0.667	0.668	0.982	0.996	1.000

Fonte: Elaborado pelo autor.

Agora ao analisar os perfis de desempenho quando considerado a média resultado dado pela Figura 19, percebe-se que a técnica de penalização DPM foi a que obteve os resultados satisfatórios para todos os problemas, pois obteve o maior de $\rho(\tau)$ para $\tau = 1$. Neste caso entende-se satisfatório como um valor muito próximo do melhor resultado encontrado.

Figura 19 – Perfis de desempenho utilizando a média dos resultados obtidos.



Fonte: Elaborado pelo autor.

Observa-se também pela Tabela 11 ao analisar as áreas dos gráficos de perfil de desempenho que o algoritmos de maior confiabilidade foram o APM_Acum, APM_Spor e APM.

Tabela 11 – Área normalizada dos perfis de desempenho utilizando a média dos resultados obtidos.

Técnica	SPM	DPM	APM	APM_Spor	APM_Acum
Área	1.000	0.958	0.718	0.715	0.699

Fonte: Elaborado pelo autor.

Por fim, a Tabela 12 faz um resumo comparativo entre os resultados obtidos em cada problema de engenharia com os resultados encontrados na literatura.

Tabela 12 – Comparação dos resultados para os problema propostos.

Métodos	Mola	Viga soldada	Vaso de pressão
SPM	0.012667	2.53448	6059.71526
DPM	0.012665	2.53450	6060.75465
APM	0.012665	2.53450	6059.71523
APM_Spor	0.012665	2.53448	6059.71525
APM_Acum	0.012665	2.53448	6059.71523
Lemonge et.al [33]	0.012667	2.38159	6060.18793
Duarte [45]	0.012665	2.38200	6388.02500
Bernardino et al. [47]	0.012666	2.38100	6059.85500

Fonte: Elaborado pelo autor.

Em Barbosa e Lemonge [34] [33], utilizam o APM acoplado a um algoritmo genético evolucionário, no qual, mostrou-se eficaz e robusto quando aplicado para testar problemas da literatura. Bernardino et.al [47] utilizam um algoritmo genético hibridizado com um sistema imunológico artificial como uma alternativa para resolver problemas de otimização restrita em engenharia. Já Duarte [45] Aplica diferentes técnicas de penalização acopladas no algoritmo colônia de abelhas artificiais.

6. CONSIDERAÇÕES FINAIS

Na presente dissertação foram testadas diferentes técnicas de penalização de restrições acoplados a um algoritmo de competição imperialista ICA, de modo a comparar e analisar os perfis de desempenho dessas técnicas na aplicação em problemas clássicos da engenharia mecânica.

Portanto, foram abordados conceitos sobre a heurística ICA, que se baseia no imperialismo colonial adotado pelos países no século XVI com o objetivo de estender seus limites territoriais através da conquista de outros territórios denominados colônias. Ao aplicar o ICA nos problemas propostos, foi observado que o algoritmo independente da técnica de penalização acoplada, possui uma rápida convergência para uma solução, e ainda mostrou-se um excelente algoritmo de busca global devido a dois parâmetros de controle: (i) a Taxa de Revolução, que funciona como um mecanismo de stress, que tem por objetivo manter um certo nível de diversidade das soluções; (ii) e o Coeficiente angular de assimilação, no qual permite que, enquanto as colônia se deslocam em direção ao imperialista, explorem o espaço de busca entorno do imperialista, assim podem encontrar alguma região mais atraente.

Também foram detalhados as diferentes estratégias utilizadas para o tratamento das restrições dos problemas propostos. As técnicas de penalização estática SPM e dinâmica DPM mostraram-se ser de implementação simples e apresentarem excelentes resultados, contudo, a tarefa de definição dos fatores de penalização foi uma tarefa árdua, pois cada problema precisou de ajuste nesses parâmetros que foram ajustados por tentativa e erro. Já as técnicas APM, APM_Spor e APM_Acum não necessitam de definição dos fatores de penalização por parte do usuário, o que tornou a implementação dos algoritmos muito mais rápida e seus resultados foram os que obtiveram os melhores resultados.

Quanto aos resultados obtidos nessa dissertação, vale destacar que, os resultados do ICA acoplado a todas as técnicas de penalização testadas mostraram-se extremamente competitivos quando comparado aos resultados já conhecidos da literatura, onde, para o problema de projeto de mola de tração/compressão o APM_Acum foi o que obteve o melhor resultado; para o problema de vida soldada o APM_Spor foi o algoritmo que se destacou; e a para o problema de vaso de pressão o APM mostrou-se o melhor entre todos.

REFERÊNCIAS

- AKAY, B., KARABOGA, D., “A modified artificial bee ant colony algorithm for real-parameter optimization”, *Information Sciences*, v. 192, pp. 120-142, 2012.
- ARORA, J. *Introduction to Optimum Design*, 3ª Edição. s.l. : Academic Press, 2012.
- ATASHPAZ-GARGARI, E., LUCAS, C. "Imperialist competitive algorithm: An algorithm for optimizatio no inspired by imperialistic competition", *IEEE Congress on Evolutionary Computation*, pp. pp. 4661-4667, 2007
- BARBOSA, H. J., BERNARDINO, H. S., BARRETO, A. D. M. S., “Using performance profiles to analyze the results of the 2006 CEC constrained optimization competition". In: *Evolutionary Computation (CEC)*, 2010 IEEE Congress on, pp. 1-8, 2010
- BARBOSA, H. J., LEMONGE, A. C., “An Adaptative Penalty Scheme In Genetic Algorithms For Constrained Optimization Problems”, In: *GECCO*, v. 2, pp. 287-294, 2002.
- BEAN, J. C., BEN HADJ-ALOUANE, A., “A dual genetic algorithm for bounded integer programs”, *Ann Arbor*, v. 1001, pp. 48109-2117, 1992.
- BERNARDINO, H. S., BARBOSA, H. J. C., LEMONGE, A. C. C., FONSECA, L. G., “A new hybrid AIS-GA for constrained optimization problems in mechanical engineering”, *IEEE Congress on Evolutionary Computation (IEEE World Congress on Computational Intelligence)*, pp. 1455-1462, Hong Kong, 2008.
- BORGWARDT, K. H., *The Simplex Method. A Probabilistic Analysis, Algorithms and Combinatorics 1*. Springer-Verlag, Berlin, 1987.
- CARLISLE, A., DOZIER, G. An off-The-Shelf PSO. In *Proceedings of the Particle Swarm*. Indianapolis, Purdue School ofEngineering and Technology : IUPUI (in press), 2001.
- CARROLL, C. W., “The created response surface technique for optimizing nonlinear, restrained systems", *Operations Research*, v. 9, n. 2, pp. 169-184, 1961.
- COIT, D. W., SMITH, A. E., TATE, D. M., “Adaptative penalty methods for genetic optimization of constrained combinatorial problems”, *INFORMS Journal on Computing*, v. 8, n. 2, pp. 173-182, 1996.

COLORNI, A., DORIGO, M., MANIEZZO, V., OTHERS, "Distributed optimization by ant colonies". In: Proceedings of the first European conference on artificial life, v. 142, pp. 134-142, 1991.

CORREIA, A., MATIAS, J., SERÔDIO, C., "Métodos de penalidade exacta para resolução de problemas de optimização não linear", *Investigação Operacional*, v. 28, n. 1, pp. 17-30, 2008.

COURANT, R., "Variational methods for the solution of problems of equilibrium and vibrations", *Bull. Amer. Math. Soc.*, v. 49, n. 1, pp. 23, 1943.

DOLAN, E. D., MORE, J. J., "Benchmarking optimization software with performance profiles", *Mathematical Programming*, v. 91, pp. 201-213, 2002.

DORIGO, M., *Optimization, learning and natural algorithms*, Tese (Doutorado), Politecnico di Milano, Italy, 1992.

DUARTE, G. R., ANDRADE, J. J., LEMONGE, A. C., FONSECA, L., "Estudo comparativo de técnicas de penalização no algoritmo colônia de abelhas artificiais aplicado a problemas de otimização em engenharia", *SIMME/EMMCOMP*, 2014.

FIACCO, A. V., MCCORMICK, G. P., "Extensions of SUMT for nonlinear programming: equality constraints and extrapolation", *Management Science*, v. 12, n. 11, pp. 816-828, 1966.

HAMIDA, S. B., SCHOENAUER, M., "An adaptive algorithm for constrained optimization problems". In: *Parallel Problem Solving from Nature PPSN VI*, pp. 529-538, 2000.

HOLLAND, J. H. "Genetic algorithms and the optimal allocation of trials", *SIAM Journal on Computing*, pp. v. 2, n. 2, pp. 88-105. 1973

HOMAI FAR, A., QI, C. X., LAI, S. H., "Constrained optimization via genetic algorithms", *Simulation*, v. 62, n. 4, pp. 242-253, 1994.

IZMAILOV, A., SOLODOV, M., "Otimização-volume 1 Condições de Otimalidade", *Elementos de Análise Convexa e de Dualidade*. 2ª edição, Rio de Janeiro: IMPA, 2009.

JOINES, J. A., HOUCK, C. R., "On the use of non-stationary penalty functions to solve nonlinear constrained optimization problems with GA's". In: *Evolutionary Computation*,

1994. IEEE World Conference on Computational Intelligence., Proceedings of the First IEEE Conference on, pp. 579-584, 1994.

LINDEN, R. Algoritmos Genéticos. s.l. : Editora Braspot, 2008.

KARABOGA, D., “An idea based on honey bee swarm for numerical optimization”, Techn. Rep, TR06, Erciyes Univ. Press, Erniyies, 2005.

KARGUPTA, H., GOLDBERG, D. E. "Black box Optimization: Implications of SEARCH". University of Illinois : Urbana-Champaign, 1995.

KAVEH, A.; TALATAHARI, S., “Imperialist Competitive Algorithm for Engineering design problems”, Asian Journal of Civil Engineering (Building and Housing), Volume II, number 6., pages 675-697. 2010.

KENNEDY, J., EBERHART, R.C. Swarm Intelligence. s.l. : Morgan Kaufmann Publishers, 2001.

KIRKPATRICK, S., JR., D. G., VECCHI, M. P., “Optimization by simulated Annealing”, science, v. 220, n. 4598, pp. 671-680, 1983.

KULTEREL-KONAK, S. NORMAN, B. A., COIT, D. W., SMITH, A. E., “Exploiting tabu search memory in constrained problems”, INFORMS Journal on Computing, v. 16, n. 3, pp. 241-254, 2004.

LEMONGE, A. C., BARBOSA, H. J., “An adaptive penalty scheme for genetic algorithms in structural optimization”, International Journal for Numerical Methods in Engineering, v. 59, n. 5, pp. 703-736, 2004.

MARTINEZ, J. M., SANTOS, S. A., “Métodos computacionais de otimização", Colóquio Brasileiro de Matemática, Apostilas, v. 20, 1995.

MEDEIROS, G. F. F., KRIPKA, M., “Algumas aplicações de métodos heurísticos na otimização de estruturas”, Revista CIAREC-UPF, v. 4, n. 1, pp. 19-32, 2012.

MEZURA-MONTES, E., “Constraint-handling in evolutionary optimization”, v. 198. Springer, 2009.

MEZURA-MONTES, E., COELLO, C. A. "Constraint-handling in nature-inspired numerical optimization: past, present and future", *Swarm and Evolutionary Computation*, pp. v. 1, n. 4, pp. 173-194, 2011.

MORADI, M. H.; ZEINALZADEH, A.; MOHAMMADI, Y.; ABEDINI, M. "An efficient hybrid method for solving the optimal sitting and sizing problem of dg and shunt capacitor banks simultaneously based on imperialist competitive algorithm and genetic algorithm", *Electrical Power and Energy Systems*, v. 54, p. 101–111, 2014.

MOUSSAVI RAD, S. J.; AKHLAGHIAN TAB, F.; MOLLAZADE, K., "Application of Imperialist Competitive Algorithm for Feature Selection: A case Study on Bulk Rice Classification", *Int. Journal of Computer Applications* (0975- 8887), Vol.40 – No. 16, 2012.

RAO, S. S. *Engineering optimization: theory and practice*. s.l. : John Wiley & Sons, 2009.

RUNARSSON, T. P., YAO, X., "Stochastic ranking for constrained evolutionary optimization", *Evolutionary Computation, IEEE Transactions on*, v. 4, n. 3, pp. 284-294, 2000.

SANGREN, E., "Nonlinear integer and discrete programming in mechanical design", *Proceedings of the ASME Design Technology Conference*: pp. 95-105, 1988.

SARAMAGO, S. F. P. "Métodos de otimização randômica: algoritmos genéticos e simulated annealing". São José do Rio Preto : Minicurso do XXVI CNMAC, 2003.

SCHOENAUER, M.M XANTHAKIS, S., "Constrained GA Optimization.", In: *ICGA*, pp. 573-580, 1993.

SHI, Y. e EBERHART, R. C. *A modified particle swarm optimizer*. Anchorage, AK, USA : *IEEE International Conference on Evolutionary*, 1998. pp. 69-73.

SIMÕES, G.J., EBECKEN, N., "Algoritmo genético e enxame de partículas para a otimização de suportes laterais de fornos", *Revista Internacional de Métodos Numéricos para Cálculo y Diseño en Ingeniería*, v. 32, pp. 7-12, 2015.

SMITH, A. E., TATE, D. M., "Genetic optimization using a penalty function". In: *Proceedings of the 5th internacional conference on genetic algorithms*, pp. 499-505, 1993.

PARSOPOULOS, K. e VRAHATIS, M. Recent Approaches to Global Optimization Problems Through Particle Swarm, In Natural Computing1:. s.l. : Kluwer AcademicPublishers, 2002. pp. 232–306.

WILSON, E. O. Naturalist. New York, USA : Warner Books, 1995.

WILSON, E. O. The insect societies. Cambridge : Belknap Press of Harvard University Press, 1971.