



Universidade do Estado do Rio de Janeiro
Centro de Tecnologia e Ciências
Instituto Politécnico

Claudemir Mota da Cruz

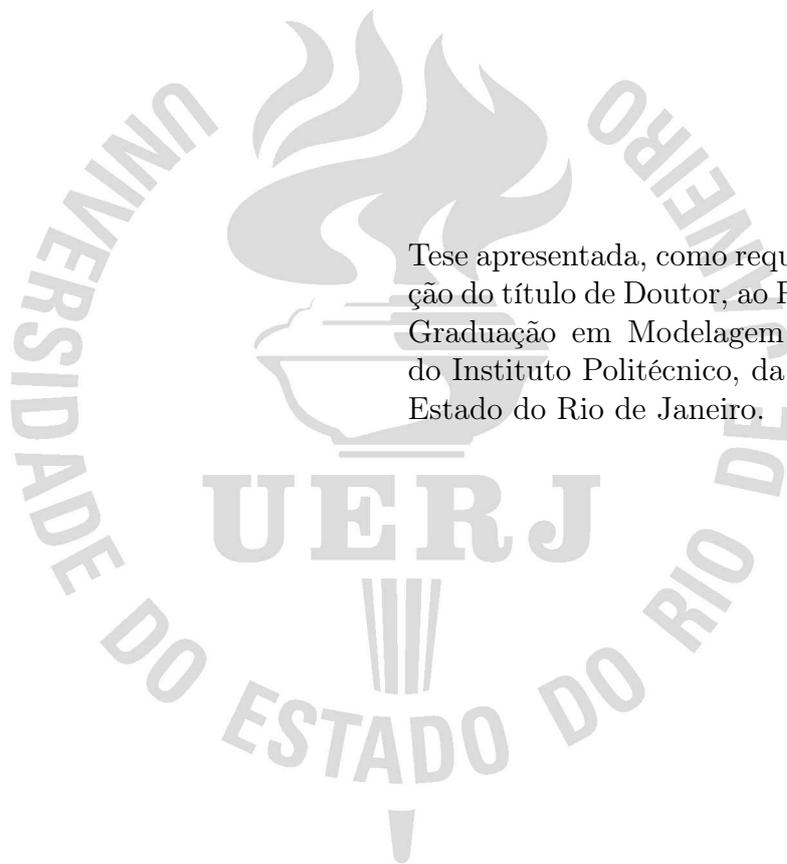
**Metamodelagem da otimização de sistemas de engenharia na
presença de incerteza**

Nova Friburgo

2024

Claudemir Mota da Cruz

**Metamodelagem da otimização de sistemas de engenharia na presença de
incerteza**



Tese apresentada, como requisito para obtenção do título de Doutor, ao Programa de Pós-Graduação em Modelagem Computacional, do Instituto Politécnico, da Universidade do Estado do Rio de Janeiro.

Orientador: Professor Dr. Gustavo Barbosa Libotte

Nova Friburgo

2024

CATALOGAÇÃO NA FONTE
UERJ / REDE SIRIUS / BIBLIOTECA CTC/E

C957 Cruz, Claudemir Mota da.
Metamodelagem da otimização de sistemas de engenharia na
presença de incerteza / Claudemir Mota da Cruz. - 2024.
122 f. : il.

Orientador: Gustavo Barbosa Libotte.

Tese (doutorado) – Universidade do Estado do Rio de Janeiro,
Instituto Politécnico.

1. Otimização matemática – Teses. 2. Processos gaussianos –
Teses. 3. Análise matemática – Teses. I. Libotte, Gustavo Barbosa.
II. Universidade do Estado do Rio de Janeiro. Instituto Politécnico.
III. Título.

CDU 519.863

Bibliotecária Fernanda Souza Cruz CRB7/7361

Autorizo, apenas para fins acadêmicos e científicos, a reprodução total ou parcial desta
tese, desde que citada a fonte.

Assinatura

Data

Claudemir Mota da Cruz

**Metamodelagem da otimização de sistemas de engenharia na presença de
incerteza**

Tese apresentada, como requisito para obtenção do título de Doutor, ao Programa de Pós-Graduação em Modelagem Computacional, do Instituto Politécnico, da Universidade do Estado do Rio de Janeiro.

Aprovada em 20 de Dezembro de 2024.

Banca Examinadora:

Professor Dr. Gustavo Barbosa Libotte (Orientador)
Instituto Politécnico - UERJ

Profa. Dra. Camila Martins Saporetti
Instituto Politécnico - UERJ

Prof. Dr. Bernardo Sotto Maior Peralva
Instituto Politécnico - UERJ

Prof. Dr. Fran Sérgio Lobato
Universidade Federal de Uberlândia

Prof. Dr. Gustavo Mendes Platt
Universidade Federal do Rio Grande

Nova Friburgo

2024

DEDICATÓRIA

Ao meu avô Manoel Carvalho (*in memoriam*), que anteviu tudo isso, mesmo sem saber.

AGRADECIMENTOS

Agradeço primeiramente a Deus e a Nossa Senhora Aparecida, por me darem forças para seguir adiante em diversos momentos de grande desafio.

Agradeço aos meus pais, cuja imensa generosidade, amor, carinho e incentivo foram fundamentais em minha trajetória acadêmica. Não pouparam esforços para me ajudar a trilhar o caminho que escolhi seguir.

Agradeço à minha amada irmã por sua presença, carinho e amor nos momentos em que minha confiança vacilou.

Agradeço aos amigos que fiz na UESC e no IPRJ, que se tornaram parte essencial desta caminhada; deixo meu sincero obrigado.

Agradeço ao Prof. Francisco Duarte Moura Neto (*in memoriam*), pela grande oportunidade de tê-lo inicialmente como professor e orientador. Embora nossa convivência tenha sido breve, ela foi fundamental para que meus olhos fossem abertos em direção a um novo caminho profissional.

Agradeço imensamente ao meu orientador, Prof. Gustavo Libotte, pela supervisão, ajuda, paciência e dedicação ao longo de todo o processo. Seu conhecimento, sugestões e cuidado foram essenciais para que este trabalho alcançasse a qualidade atual e eu pudesse aprender um pouco mais a cada dia. Minha profunda gratidão.

Agradeço aos professores e funcionários do IPRJ, que direta ou indiretamente contribuíram para a conclusão desta jornada.

Agradeço à Universidade Estadual de Santa Cruz (UESC) pelo apoio financeiro e pela oportunidade de cursar o doutorado por meio da parceria com a UERJ, através do Instituto Politécnico (IPRJ).

Agradeço também à CAPES pelo suporte institucional.

Ó Maria, concebida sem pecado,
rogai por nós que recorremos a vós. Amém.

RESUMO

CRUZ, C. M. *Metamodelagem da otimização de sistemas de engenharia na presença de incerteza*. 2024. 122 f. Tese (Doutorado em Modelagem Computacional) – Instituto Politécnico, Universidade do Estado do Rio de Janeiro, Nova Friburgo, 2024.

Esta tese explora o uso de modelos substitutos para lidar com incertezas na otimização de sistemas de engenharia, com o objetivo de aprimorar os processos de tomada de decisão em problemas complexos e reduzir os custos computacionais associados. A pesquisa considera fatores nos quais perturbações externas e incertezas podem impactar a determinação de condições ideais de operação para um sistema. De forma geral, os problemas de otimização robusta buscam minimizar a sensibilidade a tais variações, enquanto a confiabilidade tem como foco prevenir falhas nos sistemas, garantindo seu uso seguro. O objetivo principal deste trabalho é empregar a técnica de Regressão por Processos Gaussianos, em uma abordagem híbrida, com o objetivo de reduzir o número de cálculos da função objetivo na otimização com incerteza, o que, conseqüentemente, pode melhorar a eficiência do processo sem degradar a qualidade da solução. Além disso, o estudo combina um método de projeção estereográfica com um esquema de passo adaptativo para análises de confiabilidade inversa. A eficácia do método proposto é demonstrada por meio de sua aplicação em problemas de *benchmark* e em problemas de engenharia estrutural amplamente utilizados na literatura. Os resultados demonstram uma redução de pelo menos 60% no número de avaliações da função objetivo em problemas envolvendo robustez e confiabilidade, baseados em fronteiras de Pareto, quando comparado com o não emprego de metamodelos. Além disso, o método proposto apresenta robustez a variações, mantendo a qualidade das soluções com um desvio máximo de 1%.

Palavras-chave: metamodelagem; regressão com processos gaussianos; otimização robusta; otimização baseada em confiabilidade.

ABSTRACT

CRUZ, C. M. *Optimization of engineering systems metamodeling in the presence of uncertainty*. 2024. 122 f. Tese (Doutorado em Modelagem Computacional) – Instituto Politécnico, Universidade do Estado do Rio de Janeiro, Nova Friburgo, 2024.

This thesis explores the use of surrogate models to address uncertainties in the optimization of engineering systems, aiming to improve decision-making in complex problems and reduce computational costs. The research considers factors where external disturbances and uncertainties can impact the determination of optimal operating conditions. Generally, robust optimization seeks to minimize sensitivity to these variations, while reliability analysis focuses on preventing system failures, ensuring safe operation. This work employs Gaussian Process Regression, in a hybrid approach, to reduce the number of objective function evaluations in optimization under uncertainty, thereby improving efficiency without sacrificing solution quality. Furthermore, it integrates a stereographic projection method with an adaptive step size scheme for inverse reliability analysis. The effectiveness of the proposed methodology is demonstrated through its application to benchmark problems and structural engineering problems commonly used in the literature. Results demonstrate a reduction of at least 60% in the number of objective function evaluations for robustness and reliability-based optimization problems, using Pareto frontiers, compared to approaches without metamodels. Moreover, the proposed method exhibits robustness to parameter variations, maintaining solution quality within a maximum deviation of 1%.

Keywords: metamodeling; gaussian process regression; robust optimization; reliability-based optimization.

LISTA DE FIGURAS

Figura 1 - Encaixe de uma peça cilíndrica em uma base retangular	18
Figura 2 - Ilustração do conjunto e fronteira de pareto	26
Figura 3 - Método de passo adaptativo para busca linear - vizinhança convexa . .	29
Figura 4 - Método de passo adaptativo para busca linear - vizinhança côncava . .	30
Figura 5 - Indicador de hipervolume para o caso bidimensional	36
Figura 6 - Comparação entre o gráfico da função de média efetiva e o da função original	38
Figura 7 - Ilustração da região segura e de falha	41
Figura 8 - Ilustração da região segura e de falha no espaço normal padrão	43
Figura 9 - Probabilidade de falha para diferentes valores de beta	44
Figura 10 - Gráficos das funções de estado limite	45
Figura 11 - Gráficos das funções de desempenho restritas à circunferência	46
Figura 12 - Projeção estereográfica	50
Figura 13 - Fluxograma do método do tamanho de passo adaptativo com projeção estereográfica	53
Figura 14 - Projeção estereográfica no espaço normal padrão	54
Figura 15 - Nuvem de dispersão de pontos	57
Figura 16 - Distribuições <i>a priori</i> e <i>a posteriori</i> envolvendo processos gaussianos .	65
Figura 17 - Regressão com processos Gaussianos utilizando a função <i>kernel</i> RBF .	66
Figura 18 - Regressão tomando amostras segundo a função de aquisição melhoria esperada	70
Figura 19 - Regressão tomando amostras segundo a função de aquisição probabilidade de melhoria	71
Figura 20 - Regressão tomando amostras segundo a função de aquisição UCB . . .	73
Figura 21 - Fluxograma do algoritmo de avaliação da robustez utilizando metamodelagem	75
Figura 22 - Impacto de diferentes tamanhos da janela deslizante no número de avaliações da função objetivo	76
Figura 23 - Comparação entre o número de avaliações da função objetivo e o valor ótimo obtido	77
Figura 24 - Fluxograma do algoritmo de avaliação da robustez e confiabilidade utilizando metamodelagem	78
Figura 25 - Pontos ótimos obtidos em 100 execuções para 10% de Robustez	83
Figura 26 - Estrutura de treliça de 10 hastes	85
Figura 27 - Distribuição do número de avaliações e dos valores da função objetivo para 1% de robustez	86

Figura 28 - Distribuição do número de avaliações e dos valores da função objetivo para 5% de robustez	87
Figura 29 - Distribuição do número de avaliações e dos valores da função objetivo para 10% de robustez	87
Figura 30 - Evolução do valor da função objetivo em função do número de avaliações para cem execuções e com 1% de robustez	88
Figura 31 - Evolução do valor da função objetivo em função do número de avaliações para cem execuções e com 5% de robustez	89
Figura 32 - Evolução do valor da função objetivo em função do número de avaliações para cem execuções e com 10% de robustez	89
Figura 33 - Estimativas da distribuição conjunta entre o número de avaliações da função objetivo e o valor da função objetivo para diferentes níveis de robustez	90
Figura 34 - Estrutura de treliça com 25 hastes	92
Figura 35 - Distribuição do número de avaliações e dos valores da função objetivo para 1% de robustez	94
Figura 36 - Distribuição do número de avaliações e dos valores da função objetivo para diferentes níveis de robustez	94
Figura 36 - Distribuição do número de avaliações e dos valores da função objetivo para diferentes níveis de robustez	95
Figura 37 - Evolução das avaliações e da função objetivo para 1% de robustez . . .	95
Figura 38 - Evoluções do valor médio da função objetivo em função do valor médio do numero de avaliações da função objetivo por geração para 5% e 10% de robustez	96
Figura 38 - Estimativas da distribuição conjunta entre o número de avaliações da função objetivo e o valor da função objetivo para diferentes níveis de robustez	97
Figura 39 - Sequência de pontos no espaço normal padrão convergindo para o MPP	99
Figura 40 - Otimização realizada utilizando o método SORA com o ASOSL integrado à estratégia da projeção estereográfica para análise de confiabilidade inversa	101
Figura 41 - Fonteiras e conjuntos de Pareto obtidos com o uso do NSGA-II para diferentes níveis de robustez (1%, 5% e 10%)	104
Figura 42 - Evolução do número de avaliações da função objetivo e do valor do hipervolume normalizado para 1% de robustez	105
Figura 43 - Distribuição do hipervolume normalizado para 1% de Robustez usando aprendizagem ativa	106
Figura 44 - Evolução das avaliações e do valor da da distância geracional para 1% de robustez	106

Figura 45 - A estrutura de treliça composta por 10 hastes	108
Figura 46 - Distribuição do número de avaliações e resultado ótimo obtido em 30 execuções com 1% de robustez e índice de confiabilidade beta igual a 1	110
Figura 47 - Distribuição do número de avaliações e resultado ótimo obtido em 30 execuções com 5% de robustez e índice de confiabilidade beta igual a 1	111
Figura 48 - Distribuição do número de avaliações e resultado ótimo obtido em 30 execuções com 10% de robustez e índice de confiabilidade beta igual a 1	111
Figura 49 - Ciclos do SORA com as etapas de metamodelagem e refinamento para alcançar 5% de robustez e índice de confiabilidade beta igual a 1	112
Figura 50 - Fronteira de Pareto elaborada com base na otimização com robustez e confiabilidade para o problema da treliça de 10 barras	113

LISTA DE TABELAS

Tabela 1 - Avaliações da função objetivo para a convergência do algoritmo genético para diferentes níveis de robustez	82
Tabela 2 - Valores da função objetivo obtidas com o algoritmo genético para diferentes níveis de robustez	83
Tabela 3 - Avaliações da função objetivo para a convergência do algoritmo genético para diferentes níveis de robustez	85
Tabela 4 - Valores ótimos obtidos para o uso do algoritmo genético para diferentes níveis de robustez	86
Tabela 5 - Limites de tensão e compressão para a treliça de 25 barras	91
Tabela 6 - Casos de carga para treliça de 25 barras	92
Tabela 7 - Avaliações da função objetivo para a convergência do algoritmo genético para diferentes níveis de robustez	92
Tabela 8 - Valores ótimos obtidos para o uso do algoritmo genético para diferentes níveis de robustez	93
Tabela 9 - Comparação entre solução exata e os resultados obtidos com o ASOSL com a projeção estereográfica	99
Tabela 10 - Dados para a simulação do problema da treliça de 10 barras	100
Tabela 11 - Comparação dos métodos RBDO na treliça de 10 barras	102
Tabela 12 - Avaliações da função objetivo para a convergência do algoritmo NSGA-II para diferentes níveis de robustez	105
Tabela 13 - Hipervolume para diferentes níveis de robustez	107
Tabela 14 - Solução Determinística para o Problema para o problema da telça de 10 barras	108
Tabela 15 - Número de avaliações da função objetivo com o uso da metamodelagem para diferentes níveis de robustez e índices de confiabilidade	109
Tabela 16 - Valores ótimos obtidos para com o uso da metamodelagem para diferentes níveis de robustez e índices de confiabilidade	110

LISTA DE ALGORITMOS

Algoritmo 1 - Backtracking line search	28
Algoritmo 2 - Algoritmo de otimização de passo adaptativo de segunda ordem . .	31
Algoritmo 3 - Algoritmo Genético	33
Algoritmo 4 - Algoritmo NSGA-II	34
Algoritmo 5 - Algoritmo de otimização de passo adaptativo de segunda ordem com projecção estereográfica	53

LISTA DE ABREVIATURAS E SIGLAS

FORM	<i>First Order Reliability Method</i> (Método de Confiabilidade de Primeira Ordem)
GA	<i>Genetic Algorithm</i> (Algoritmo Genético)
GPR	<i>Gaussian Process Regression</i> (Regressão com Processos Gaussianos)
NSGA-II	<i>Non-dominated Sorting Genetic Algorithm II</i> (Algoritmo Genético de Ordenação Não Dominada II).
PMA	<i>Performance Measure Approach</i> (Abordagem de Medida de Desempenho)
RBDO	<i>Reliability-Based Design Optimization</i> (Otimização Baseada em Confiabilidade)
RBRDO	<i>Reliability-Based Robust Design Optimization</i> (Otimização Robusta Baseada em Confiabilidade)
RIA	<i>Reliability Index Approach</i> (Abordagem de Índice de Confiabilidade)
SORA	<i>Sequential Optimization and Reliability Assessment</i> (Otimização Sequencial e Avaliação de Confiabilidade)

LISTA DE SÍMBOLOS

Letras Latinas

$ B_\delta(\mathbf{x}) $	Hipervolume de uma vizinhança centrada em \mathbf{x} com raio δ
f	Função objetivo em problemas de otimização
f^{eff}	Função de média efetiva
f_X	Função densidade de probabilidade da variável aleatória X
g	Função de desempenho no espaço das variáveis de decisão
G	Função de desempenho no espaço normal padrão
P_f	Probabilidade de falha
P_s	Probabilidade de sucesso
P^*	Conjunto de Pareto
PF	Fronteira de Pareto
u^*	Ponto mais provável de falha (MPP)
U	Transformação de Rosenblatt para o espaço normal padrão
X	Vetor de variáveis aleatórias
Y	Espaço objetivo em problemas multiobjetivo
$\ u\ $	Norma do vetor u
\triangleq	Definido como

Letras Gregas

β_i	Índice de confiabilidade alvo
μ_X	Média da variável aleatória X
∇G	Gradiente da função G no espaço normal padrão
Φ	Função distribuição acumulada da distribuição normal padrão
σ_X	Desvio padrão da variável X
$\phi(u)$	Função densidade de probabilidade da distribuição normal padrão
Ψ	Coefficiente de penalidade
δ	Vetor de ruídos

SUMÁRIO

	INTRODUÇÃO	17
1	OTIMIZAÇÃO COM INCERTEZAS	23
1.1	Conceitos Fundamentais de Otimização	23
1.2	Método de Passo Adaptativo de Segunda Ordem	27
1.2.1	<u>Backtracking Line Search</u>	27
1.2.2	<u>Cálculo do Intervalo de Busca Linear Adaptativo</u>	28
1.3	Algoritmos Evolucionários	31
1.3.1	<u>Algoritmo Genético</u>	32
1.3.2	<u>Algoritmo de Ordenação Não Dominada II</u>	33
1.3.3	<u>Indicadores de eficiência</u>	35
1.4	Otimização Robusta	36
1.5	Otimização Baseada em Confiabilidade	40
1.6	Otimização Baseada em Robustez e Confiabilidade	46
2	MÉTODO DE PASSO ADAPTATIVO DE SEGUNDA ORDEM COM PROJEÇÃO ESTEREOGRÁFICA PARA ANÁLISE DE CONFIABILIDADE INVERSA	49
2.1	Projeção Estereográfica	49
2.2	Algoritmo do Método de Passo Adaptativo de Segunda Ordem com Projeção Estereográfica	52
3	METAMODELAGEM	56
3.1	Regressão Linear com Processos Gaussianos	56
3.2	Regressão com Processos Gaussianos	61
3.2.1	<u>Regressão com Dados Não Ruidosos</u>	63
3.2.2	<u>Regressão com Dados Ruidosos</u>	65
3.2.3	<u>Otimização dos Hiperparâmetros</u>	66
3.3	Funções de Aquisição	68
3.3.1	<u>Função de Melhoria Esperada</u>	68
3.3.2	<u>Função de Probabilidade de Melhoria</u>	70
3.3.3	<u>Função de Aquisição Confiança Superior Limitada</u>	72
4	ALGORITMO DE OTIMIZAÇÃO COM METAMODELAGEM CONSIDERANDO ROBUSTEZ E CONFIABILIDADE	74
4.1	O Algoritmo de Avaliação da Robustez Utilizando Metamodelagem	74
4.2	O Algoritmo de Avaliação da Robustez e Confiabilidade Utili- zando Metamodelagem	77
5	RESULTADOS	80
5.1	Robustez	81

5.1.1	<u>Estruturas de Treliças</u>	84
5.2	Confiabilidade	98
5.2.1	<u>Análise de confiabilidade</u>	98
5.3	Robustez e Confiabilidade	102
	CONCLUSÕES	114
	REFERÊNCIAS	117

INTRODUÇÃO

Durante a elaboração de um projeto de engenharia, a determinação de parâmetros apropriados não necessariamente deve se restringir à redução de custos e ao cumprimento dos valores mínimos exigidos para o funcionamento. Essa abordagem limitada pode negligenciar incertezas importantes, como condições climáticas adversas e imprecisões inerentes ao processo de fabricação, por exemplo (Tsutsui; Ghosh; Fujimoto, 1996).

Sendo assim, identificar o conjunto de valores capaz de otimizar o sistema em várias situações, ou seja, que proporcione a melhor solução levando em conta a segurança, dentro das restrições estabelecidas, pode ser importante para assegurar o funcionamento eficiente e seguro do sistema. Na Figura 1, pode-se observar uma situação em que a imprecisão geométrica no processo de fabricação tem o potencial de comprometer o ajuste entre componentes de uma estrutura, o que pode, por conseguinte, afetar negativamente o desempenho do sistema.

Quando um problema não considera as potenciais incertezas, as soluções otimizadas podem residir em áreas de alta sensibilidade da função objetivo a perturbações externas (Deb; Gupta, 2005). Nesses casos, é necessário que a solução seja robusta, ou seja, que ela seja capaz de manter o desempenho do sistema, mesmo quando sujeita a pequenas flutuações que não podem ser evitadas.

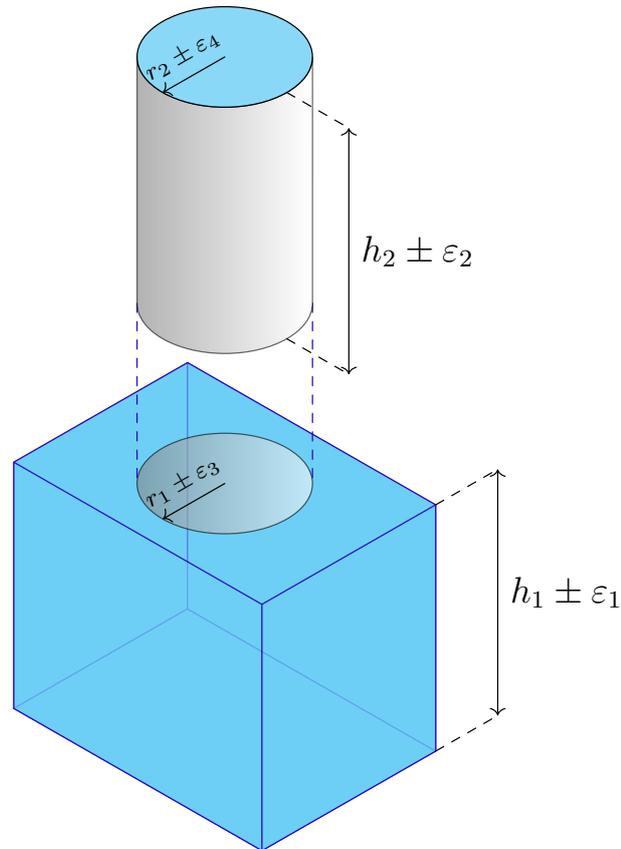
Contudo, o custo de se avaliar a robustez pode se tornar proibitivo quando o problema exige grande esforço computacional, como é o caso, por exemplo, de problemas de análise de elementos finitos (Yang et al., 2022). Esta dificuldade pode ser contornada ou suavizada por meio da utilização de metamodelos (ou modelos substitutos), que aproximam a função objetivo ou as restrições do problema por funções substitutas que, tipicamente, podem ter sua determinação e avaliação menos custosas, ou com um custo fixo já esperado, possibilitando alcançar resultados ótimos similares ao problema original (Jiang; Zhou; Shao, 2020).

Além de ser robusta, pode ser importante que a solução também seja confiável. Isso significa que as soluções devem satisfazer uma determinada probabilidade de falha, ou seja, devem atender a uma probabilidade de violação das restrições definidas, garantindo que o sistema opere sob condições seguras (Lopez; Beck, 2012).

Porém, quando se trata de elementos finitos, as simulações podem exigir milhões de avaliações a fim de obter uma solução satisfatória, que, somada à complexidade causada pela análise de robustez e confiabilidade, pode tornar o processo altamente custoso em termos de tempo, recursos computacionais e gasto energético, mesmo com o aumento contínuo do poder de processamento computacional.

Devido a este alto custo, diversos tipos de metamodelos têm sido empregados com o propósito de diminuir o custo computacional durante o processo de otimização, sendo

Figura 1 - Encaixe de uma peça cilíndrica em uma base retangular.



Legenda: As imprecisões geométricas podem ocorrer devido a imprecisão dos equipamentos envolvidos na fabricação, por exemplo.

Fonte: O autor, 2024.

comumente utilizados para predição, análise de sensibilidade, quantificação da incerteza e otimização assistida por metamodelos. Entre os tipos de metamodelos, destacam-se as superfícies de resposta, a regressão linear, as funções de base radial, a regressão do vetor de suporte, as redes neurais artificiais, a expansão do caos polinomial, a técnica de árvores impulsionadas, as florestas aleatórias e a regressão com processos gaussianos (Kudela; Matousek, 2022).

Ainda segundo Kudela e Matousek (2022), o uso de metamodelos pode ser dividido em três categorias, a saber: *i*) criação e validação de modelos substitutos especificamente voltados para predições; *ii*) exploração da análise de sensibilidade em modelos, quantificando o impacto de parâmetros incertos na dinâmica dos sistemas por meio de diversas técnicas; *iii*) otimização assistida por metamodelo, que aborda os cenários onde a função objetivo, empregada no processo de otimização, apresenta um custo de cálculo extremamente alto e uma indisponibilidade de informações referentes à sua derivada.

Os metamodelos, ao substituírem estes modelos de elevado custo computacional

por alternativas de menor custo, mostram-se como ferramentas potenciais para aumentar a eficiência energética no âmbito da simulação computacional. Isso, conseqüentemente, contribui para o Objetivo de Desenvolvimento Sustentável 7 (ODS 7) da Organização das Nações Unidas (Brasil, 2025), que visa garantir o acesso universal à energia confiável e, dentre outros objetivos, pretende também dobrar a taxa global de melhoria da eficiência energética até 2030.

Sendo assim, os metamodelos oferecem um caminho promissor para otimizar simulações e contribuir para metas de sustentabilidade. No entanto, embora a literatura já discuta a adoção de metamodelos para diminuir o custo computacional, sua aplicação em cenários de otimização que integrem robustez e confiabilidade, especialmente quando associada a metaheurísticas populacionais, ainda demanda atenção.

Nesta tese, a regressão com processos gaussianos (GPR - do inglês *Gaussian Process Regression*) é utilizada na otimização auxiliada por metamodelos, escolhida por sua capacidade de modelar funções complexas e estimar incertezas mesmo com poucos dados (Rasmussen; Williams, 2006).

Neste contexto, esta tese tem como objetivos específicos: *i*) apresentar uma metodologia híbrida baseada em metamodelos, especificamente Regressão com Processos Gaussianos, a fim de diminuir o número de avaliações da função objetivo e assim reduzir o custo computacional envolvido no processo de otimização robusta envolvendo elementos finitos, realizado com uma metaheurística populacional, enquanto o comportamento do modelo é preservado; *ii*) propor o uso de uma estratégia de projeção estereográfica integrada a um algoritmo de passo adaptativo para análise de confiabilidade inversa.

Para demonstrar a viabilidade, inicialmente são apresentados alguns conceitos essenciais para a compreensão da proposta. Em seguida, são exibidos alguns resultados obtidos para problemas de *benchmark* de engenharia, comparando como esses resultados se desviam das soluções obtidas por meio dos modelos originais.

Esta tese está estruturada da seguinte maneira: no Capítulo 1, são introduzidos os conceitos fundamentais relacionados à otimização, otimização robusta e otimização baseada em confiabilidade; no Capítulo 2 é apresentada como a projeção estereográfica pode ser utilizada para a análise de confiabilidade inversa; no Capítulo 3 são apresentadas as ideias principais envolvendo metamodelagem com Processos Gaussianos, onde são discutidos tópicos como Regressão com Processos Gaussianos e as funções de aquisição que são utilizadas para a seleção de pontos de amostragem; no Capítulo 4 é proposta uma metodologia que utiliza metamodelagem para otimização com robustez e confiabilidade, visando a redução do custo computacional; no Capítulo 5 são apresentados os resultados obtidos com a aplicação da metodologia proposta em problemas de otimização envolvendo apenas robustez e problemas de otimização envolvendo robustez e confiabilidade; finalmente, no Capítulo Conclusões, são apresentadas as conclusões que foram obtidas a partir do trabalho realizado e perspectivas para trabalhos futuros.

Trabalhos relacionados

Em Baquela e Olivera (2019), é abordada a ineficiência computacional em problemas de Otimização via Simulação (OvS) multiobjetivo, propondo-se um algoritmo que integra metamodelos Kriging ao NSGA-II. Tal abordagem visa reduzir os custos computacionais, sobretudo na aproximação de Fronteiras de Pareto, as quais demandam inúmeras avaliações da função objetivo. A adoção de metamodelos possibilita ao algoritmo acelerar a exploração do espaço de soluções, resultando em uma relação custo-qualidade superior à dos métodos tradicionais. Contudo, o referido estudo não abrange problemas de Otimização de Projeto Baseada em Robustez (RBDO, do inglês *Robustness-Based Design Optimization*) e Otimização de Projeto Baseada em Robustez e Confiabilidade (RBRDO, do inglês *Reliability and Robustness-Based Design Optimization*).

O trabalho de Shi, Chu e Braun (2019) abordou a aplicação dos modelos Kriging como substitutos no contexto da análise de incertezas em folhas de grafeno, utilizando o Método dos Elementos Finitos (FEM). O trabalho ressaltou o emprego da amostragem por hipercubo latino (LHS) para a propagação de incertezas associadas às propriedades geométricas e materiais. Além disso, a pesquisa comparou diferentes tipos de folhas de grafeno, destacando a eficácia do modelo na gestão de parâmetros incertos, assegurando alta precisão e eficiência no processo de análise de vibrações livres.

O modelo de Krigagem Cega Aprimorada (IBK, do inglês *Improved Blind Kriging*), descrito por Mai et al. (2022), aprimora a previsão e a busca por soluções viáveis em problemas de otimização de alto custo computacional. Incorporando técnicas de seleção de variáveis, o modelo é capaz de capturar tendências de alta ordem na função de regressão. Além disso, a estratégia de preenchimento, fundamentada na viabilidade, penalização e melhoria esperada restrita, permite ao IBK otimizar de forma eficiente, como demonstrado em *benchmarks* e em sua aplicação bem-sucedida em projetos de design estrutural.

No estudo de Huynh et al. (2024), é proposto o método iCE-GPR, voltado para a otimização de projetos baseada em confiabilidade (RBDO) e otimização topológica. Essa abordagem combina o método de entropia cruzada invertível (iCE) com modelagem de processo gaussiano (GPR) para estimar de maneira eficiente a probabilidade de falha estrutural sob incertezas paramétricas. O método utiliza um processo de aprendizado ativo para refinar iterativamente o modelo GPR, enquanto o iCE ajusta progressivamente o ponto mais provável, viabilizando otimizações determinísticas subsequentes.

Em Yu et al. (2024), um novo *framework* para otimização de projeto baseada em confiabilidade variante no tempo é proposto, com foco em demandas dinâmicas. Este problema aborda a complexidade de funções objetivo dinâmicas e restrições probabilísticas variantes no tempo. Para mitigar a alta carga computacional, o estudo propõe um método de substituição em dois níveis baseado em Kriging, que reduz os custos computacionais ao mapear diretamente variáveis de projeto e probabilidades.

No estudo de Shi et al. (2024), é apresentado o método ALK-SRBRD (do inglês *Active Learning Kriging-based System Reliability-based Robust Design*), desenvolvido para a otimização robusta de projetos baseados em confiabilidade (RBRDO). Esse método visa resolver os desafios associados ao aprimoramento da confiabilidade em problemas RBRDO, ao integrar perdas de qualidade e confiabilidade em uma formulação multiobjetivo. A abordagem utiliza a modelagem Kriging combinada ao aprendizado ativo para a aproximação eficiente de funções complexas, além de reduzir o custo computacional. Isso é alcançado por meio da decomposição do problema em três componentes: modelagem da fronteira de falha do sistema, resposta-alvo global e função de índice de confiabilidade, com a aplicação de critérios de amostragem otimizados.

Um método baseado em aprendizado ativo é proposto por Kim, Yi e Song (2024) para a otimização de projetos estruturais sujeitos a excitações estocásticas, como ventos e terremotos. O foco principal do método é calcular de forma eficiente a probabilidade de primeira passagem, que é uma medida da confiabilidade da estrutura. Para modelar as excitações estocásticas de alta dimensionalidade (ou seja, aquelas que variam em múltiplas direções ou fatores ao mesmo tempo), o método utiliza uma combinação de diferentes distribuições de probabilidade. Para prever a resposta da estrutura a essas excitações, é aplicado um processo gaussiano heteroscedástico, que é uma técnica estatística para lidar com dados que têm variabilidade em diferentes níveis. Além disso, um esquema de aprendizado ativo é utilizado para diminuir o custo computacional do processo, ou seja, otimizar o número de cálculos necessários para encontrar a solução.

Para aprimorar a eficiência em RBDO, Liu et al. (2024b) propõem o método MFLUM (do inglês, *Multi-Fidelity Local Update Method*), que emprega uma estratégia de atualização local de modelos substitutos baseada em uma função de aprendizado de curvatura. Essa abordagem identifica com precisão restrições ativas e concentra os esforços computacionais nas regiões críticas. O método também utiliza uma estratégia adaptativa para resolver o ponto mais provável de falha, alternando entre métodos aproximados e exatos, equilibrando precisão e eficiência computacional.

O estudo de Castro, Paula e Visnadi (2024) aplica RBDO ao projeto de rotores apoiados por mancais hidrodinâmicos. Modelos baseados no método dos elementos finitos são utilizados para representar o rotor e os mancais, incorporando incertezas em parâmetros como módulo de elasticidade, densidade e folga dos mancais. O objetivo é otimizar o diâmetro do eixo e a temperatura do óleo, garantindo confiabilidade de 99% em relação à amplitude de vibração e estabilidade. Modelos substitutos, como Kriging e expansão de caos polinomial, são empregados para lidar com a complexidade do problema.

Wang et al. (2024) apresentam um método de RBRDO para rolamentos de esferas, focado na resistência ao desgaste. O estudo desenvolve um modelo substituto baseado em Kriging, complementado por análises de confiabilidade usando o método do quarto momento. Um algoritmo genético é, então, empregado para realizar a otimização robusta,

reduzindo a sensibilidade ao desgaste frente a incertezas nos parâmetros de projeto, mantendo níveis de confiabilidade desejados.

No artigo de Xue et al. (2024), é proposta uma abordagem para otimização de tolerâncias em componentes aeroespaciais de paredes finas, visando minimizar a variabilidade na montagem sem comprometer custos e confiabilidade. Um *framework* de resposta em dois estágios combina modelos substitutos e amostragem por importância para acelerar os cálculos. A otimização utiliza o algoritmo de enxame de partículas (PSO), garantindo equilíbrio entre custo, variabilidade e confiabilidade.

Du et al. (2024) exploram a otimização robusta e baseada em confiabilidade de máquinas síncronas de ímã permanente de alta velocidade. O estudo adota um modelo substituto local baseado em Kriging para focar nas regiões de interesse da frente de Pareto, reduzindo significativamente o esforço computacional associado ao RBRDO.

Para micro-aletas em superfícies planas, Larrañaga et al. (2024) desenvolvem uma metodologia que maximiza a eficiência termo-hidráulica. Utilizando 15.694 simulações numéricas, modelos de aprendizado de máquina são treinados para prever desempenho, sendo posteriormente integrados a algoritmos genéticos para identificar geometrias ótimas que equilibram transferência de calor e queda de pressão.

Um método robusto para a otimização da produção em reservatórios é discutido por Liu et al. (2024a), empregando a arquitetura TransUNet (do inglês, *Transformer UNet*) para desenvolver modelos substitutos capazes de prever respostas de reservatórios sob distintos cenários geológicos. Esses modelos são integrados ao algoritmo EnOpt (do inglês, *Ensemble Optimization*), proporcionando a otimização das taxas de injeção de maneira eficiente e robusta em face de incertezas.

Abebe et al. (2024) abordam a otimização de canais de fluxo serpentinos em células de combustível de membrana de troca de prótons, utilizando modelos substitutos para reduzir os custos de simulação em COMSOL(plataforma usada para simular problemas baseados em física). Análises de sensibilidade de Sobol(método para analisar a sensibilidade desses modelos a vários parâmetros de entrada, auxiliando no processo de otimização) identificam os parâmetros críticos, que são otimizados com algoritmos genéticos, aumentando a confiabilidade e a eficiência operacional.

Chen et al. (2025) propõem uma estratégia baseada em quantiles aprimorados pelo método PC-Kriging(Kriging com Chaos Polinomial), abordando as limitações de precisão e viabilidade em métodos RBDO existentes. A adição de pontos estratégicos e o refinamento local garantem modelos substitutos mais precisos, otimizando eficiência e resultados.

1 OTIMIZAÇÃO COM INCERTEZAS

Atualmente, a otimização é extensivamente empregada na análise de sistemas de engenharia, em que empresas, investidores e fabricantes visam reduzir custos, evitar riscos excessivos e maximizar a eficiência. Para sua aplicação, é imprescindível identificar um objetivo, que pode ser lucro, massa de uma estrutura, tempo ou qualquer outra quantidade mensurável, bem como as variáveis das quais os objetivos dependem, além das restrições impostas a essas variáveis. Este procedimento caracteriza-se pelo que é denominado modelagem do problema.

Após a formulação do modelo referente ao problema, torna-se necessário empregar um algoritmo capaz de encontrar a solução correspondente. Devido à complexidade intrínseca ao problema, a utilização de um sistema computacional frequentemente se faz necessária para executar este processo de resolução. Cabe ressaltar que não há um algoritmo universal capaz de resolver todos os problemas de otimização; assim, o algoritmo deve ser escolhido conforme as características específicas do problema em questão (Noce-dal; Wright, 2006).

Nas seções seguintes, serão abordados os conceitos fundamentais associados à otimização, incluindo a busca linear com tamanho de passo adaptativo, os algoritmos evolucionários Algoritmo Genético (GA) e Algoritmo de Ordenação Não Dominada II (NSGA-II), bem como duas principais métricas de convergência. Finalmente, serão brevemente explorados os conceitos de otimização robusta, otimização com confiabilidade e a otimização que integra conjuntamente os aspectos de robustez e confiabilidade.

1.1 Conceitos Fundamentais de Otimização

Em um problema de otimização com um único objetivo, a finalidade central é determinar um vetor de variáveis de decisão $\mathbf{x} \in \mathbb{R}^n$ que conduza à otimização (seja ela minimização ou maximização) de uma função objetivo $f : \mathbb{R}^n \rightarrow \mathbb{R}$. Esta função pode estar sujeita a um conjunto de restrições, que podem envolver tanto condições de igualdade quanto de desigualdade, definindo, portanto, um conjunto conhecido como conjunto viável de soluções (Arora, 2012).

Assim, um problema de otimização determinística pode ser formulado como:

$$\begin{aligned}
 & \underset{\mathbf{x}}{\text{Minimizar}} && f(\mathbf{x}) \\
 & \text{Sujeito a} && g_i(\mathbf{x}) \leq 0, \quad i = 1, \dots, m, \\
 & && h_j(\mathbf{x}) = 0, \quad j = 1, \dots, p, \\
 & && a \leq \mathbf{x} \leq b,
 \end{aligned} \tag{1}$$

onde $\mathbf{x} = (x_1, \dots, x_n)^\top$, comumente referido como vetor de variáveis de decisão, restrições $g_i, h_j : \mathbb{R}^n \rightarrow \mathbb{R}$ que são funções reais, e a desigualdade $a \leq \mathbf{x} \leq b$ que deve ser entendida coordenada a coordenada.

Analogamente, se $\mathbf{f} : \mathbb{R}^n \rightarrow \mathbb{R}^m$, para $m \geq 2$ e $n \geq 1$, um problema de otimização multiobjetivo pode ser descrito como:

$$\begin{aligned} &\text{Otimizar } \mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), \dots, f_m(\mathbf{x}))^\top \\ &\text{Sujeito a } g_i(\mathbf{x}) \leq 0, i = 1, \dots, p \\ &\quad h_j(\mathbf{x}) = 0, j = 1, \dots, q \\ &\quad a \leq \mathbf{x} \leq b, \end{aligned} \tag{2}$$

onde $f_k : \mathbb{R}^n \rightarrow \mathbb{R}$, para $k = 1, \dots, m$.

Neste contexto, o termo *otimizar* refere-se ao processo de minimizar um conjunto de m_1 funções objetivo, f_1, f_2, \dots, f_{m_1} , enquanto simultaneamente se busca maximizar outro conjunto de m_2 funções objetivo, $f_{m_1+1}, f_{m_1+2}, \dots, f_{m_1+m_2}$, com a condição de que $m_1 + m_2 = m$, sendo m_1 e m_2 inteiros não negativos.

O conjunto X_f de todos os pontos que satisfazem as restrições é denominado *conjunto viável* ou *espaço de busca*. A imagem da função multiobjetivo, quando considerado o conjunto viável, é chamada de *espaço objetivo*, é definida como $\mathbf{Y} = \text{Im}(f|_{X_f}) = \{\mathbf{y} = \mathbf{f}(\mathbf{x}) \in \mathbb{R}^m, \forall \mathbf{x} \in X_f\}$ (Branke et al., 2008).

Em problemas envolvendo múltiplos objetivos, é importante reconhecer a potencial natureza conflitante entre diferentes objetivos: muitas vezes, uma solução que minimiza satisfatoriamente um dado objetivo pode não ser a ideal para outro objetivo, o que permite formular o conceito de dominância.

Definição 1. *Considere um problema de otimização multiobjetivo com função objetivo $\mathbf{f} : \mathbb{R}^n \rightarrow \mathbb{R}^m$, onde $\mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), \dots, f_m(\mathbf{x}))^\top$. Assuma que as funções f_1, \dots, f_{m_1} devem ser minimizadas e as funções f_{m_1+1}, \dots, f_m devem ser maximizadas, com $m = m_1 + m_2$. Dados dois vetores de decisão $\mathbf{x}^1, \mathbf{x}^2 \in \mathbb{R}^n$ e seus respectivos vetores de objetivo $\mathbf{y}^1 = \mathbf{f}(\mathbf{x}^1)$ e $\mathbf{y}^2 = \mathbf{f}(\mathbf{x}^2)$, diz-se que \mathbf{y}^1 domina \mathbf{y}^2 , denotado por $\mathbf{y}^1 \succ \mathbf{y}^2$, se e somente se as seguintes condições forem satisfeitas:*

- *O vetor de objetivos \mathbf{y}^1 não é inferior ao vetor de objetivos \mathbf{y}^2 em todos os objetivos, o que significa que $f_r(\mathbf{x}^1) \leq f_r(\mathbf{x}^2)$ para qualquer $r = 1, \dots, m_1$, e $f_r(\mathbf{x}^1) \geq f_r(\mathbf{x}^2)$ para todos $r = m_1 + 1, \dots, m$.*
- *O vetor de objetivos \mathbf{y}^1 é estritamente melhor que \mathbf{y}^2 em pelo menos um dos objetivos; isto é, $f_r(\mathbf{x}^1) < f_r(\mathbf{x}^2)$ para algum $r \in \{1, \dots, m_1\}$, ou $f_r(\mathbf{x}^1) > f_r(\mathbf{x}^2)$ para algum $r \in \{m_1 + 1, \dots, m\}$.*

Assim, um vetor viável $\mathbf{x}^* \in X_f$ é dito ser um *otimizador de Pareto* se não existir um outro vetor viável $\mathbf{x} \in X_f$, tal que $f_k(\mathbf{x}) \leq f_k(\mathbf{x}^*)$ para todo $k = 1, \dots, m_1$

e $f_k(\mathbf{x}) < f_k(\mathbf{x}^*)$ para algum índice $k \in \{1, \dots, m_1\}$, e $f_k(\mathbf{x}) \geq f_k(\mathbf{x}^*)$ para todo $k = m_1+1, \dots, m_1+m_2$ e $f_k(\mathbf{x}) > f_k(\mathbf{x}^*)$ para algum índice $k \in \{m_1+1, \dots, m_1+m_2\}$. Em outras palavras, \mathbf{x}^* é um minimizador de Pareto se não existir outro vetor viável que o domine em todos os objetivos, constituindo assim um conjunto de soluções não dominadas.

Analogamente, um vetor de objetivos $\mathbf{y}^* \in \mathbf{Y}$ é dito ser um *ótimo de Pareto* se não existir um outro vetor de objetivos $\mathbf{y} \in \mathbf{Y}$ tal que $y_i \leq y_i^*$ para todo $i = 1, \dots, m_1$ e $y_i < y_i^*$ para algum índice $i \in \{1, \dots, m_1\}$ e $y_i \geq y_i^*$ para todo $i = m_1+1, \dots, m_1+m_2$ e $y_i > y_i^*$ para algum índice $i \in \{m_1+1, \dots, m_1+m_2\}$. Equivalentemente, \mathbf{y}^* é um ótimo de Pareto se o vetor viável correspondente for um otimizador de Pareto (Talbi, 2009).

O conjunto de todos os otimizadores de Pareto é denominado *conjunto de Pareto* e é denotado por P^* e, analogamente, o conjunto de todos os ótimos de Pareto é denominado *frente de Pareto* ou *Frontera de Pareto* e é denotado por $PF = \{\mathbf{f}(\mathbf{x}) \mid \mathbf{x} \in P^*\}$.

De modo geral, a notação $\mathbf{x}^1 \succ \mathbf{x}^2$ é empregada para denotar que o vetor \mathbf{x}^1 domina o vetor \mathbf{x}^2 e pode-se demonstrar que a relação de dominância é caracterizada como uma relação de ordem parcial estrita.

Esta relação possui as seguintes propriedades:

1. **Anti-reflexiva:** Para cada vetor \mathbf{x} pertencente ao conjunto de busca X_f , tem-se que $\mathbf{x} \not\succeq \mathbf{x}$, indicando que nenhum elemento é capaz de dominar a si mesmo.
2. **Anti-simétrica:** Se ocorre que $\mathbf{x}^1 \succ \mathbf{x}^2$, então $\mathbf{x}^2 \not\succeq \mathbf{x}^1$. Ou seja, se o elemento \mathbf{x}^1 domina o elemento \mathbf{x}^2 , isso implica que \mathbf{x}^2 não domina \mathbf{x}^1 .
3. **Transitiva:** No caso em que $\mathbf{x}^1 \succ \mathbf{x}^2$ e simultaneamente $\mathbf{x}^2 \succ \mathbf{x}^3$, segue-se que $\mathbf{x}^1 \succ \mathbf{x}^3$.

Exemplo 1. Considere o problema de otimização multiobjetivo com dois objetivos e duas restrições, definido por Binh e Korn (1997), dado por:

$$\min_{\mathbf{x}} \mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x})) = (4x_1^2 + 4x_2^2, (x_1 - 5)^2 + (x_2 - 5)^2)$$

$$\text{Sujeito a } g_1(\mathbf{x}) = (x_1 - 5)^2 + x_2^2 \leq 25,$$

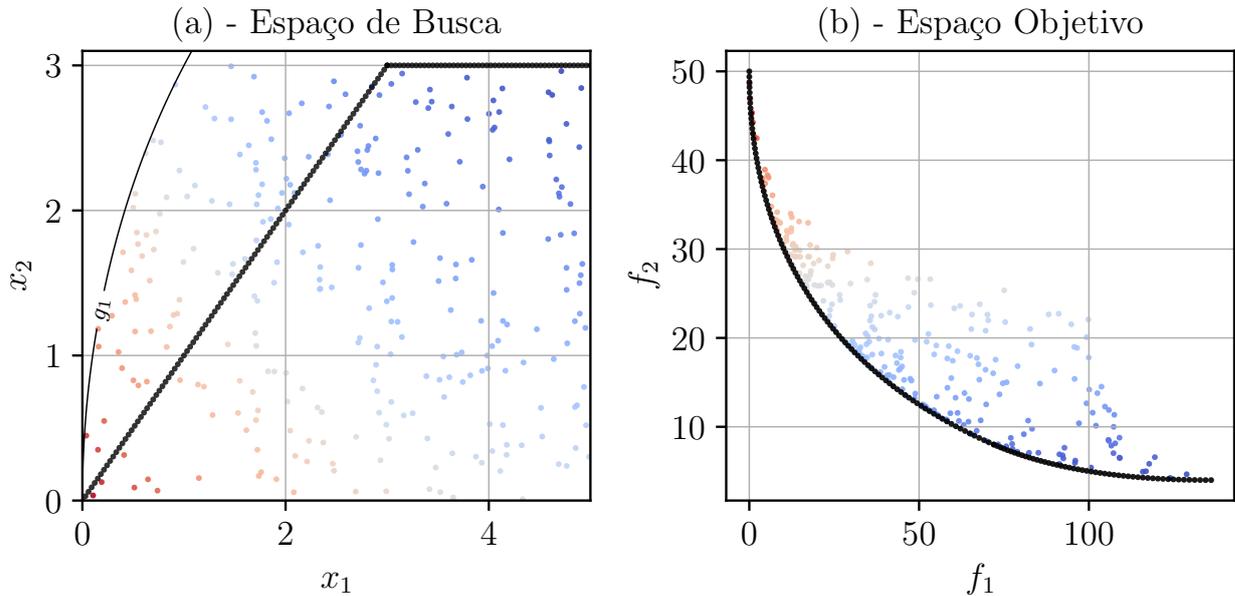
$$g_2(\mathbf{x}) = (x_1 - 8)^2 + (x_2 + 3)^2 \geq 7,7,$$

$$0 \leq x_1 \leq 5,$$

$$0 \leq x_2 \leq 3.$$

Este problema define o conjunto de Pareto P^* , composto por dois subconjuntos: o primeiro contém pares $(x_1, x_2) \in \mathbb{R}^2$ com $x_2 = x_1$ e $x_1 \in [0, 3]$. O segundo abrange pares $(x_1, x_2) \in \mathbb{R}^2$ onde $x_2 = 3$ e $x_1 \in [3, 5]$. A Figura 2(a) ilustra o conjunto de Pareto, enquanto a Fronteira de Pareto PF , contendo imagens dos pontos em P^* , aparece na Figura 2(b). Os pontos com tons variando de vermelho a azul mostram pontos dominados.

Figura 2 - Ilustração do conjunto e fronteira de Pareto.



Legenda: (a) A curva exibida em preto ilustra o conjunto de Pareto, denotado como P^* , enquanto os pontos em cores diferentes indicam o conjunto de soluções que são dominadas. (b) De maneira análoga, a curva em preto simboliza a Fronteira de Pareto, designada como PF , e os pontos em outras cores representam a imagem correspondente das soluções dominadas.

Fonte: O autor, 2024.

Observa-se na Figura 2(b) que a fronteira de Pareto corresponde ao conjunto de imagens de todas as soluções que, do ponto de vista da qualidade, não são inferiores a nenhuma solução alternativa. Considerando a perspectiva do tomador de decisão.

Em problemas monobjetivos envolvendo funções diferenciáveis, a solução pode ser obtida por meio de técnicas baseadas no gradiente descendente, como o *backtracking line search* com tamanho de passo adaptativo, o qual será detalhado na seção 1.2.

Em situações que envolvem funções não diferenciáveis ou multimodais, uma abordagem alternativa é a utilização de algoritmos evolutivos, como os algoritmos genéticos e a otimização por enxame de partículas. Para problemas de otimização multiobjetivo, a fronteira de Pareto também pode ser aproximada utilizando esses algoritmos evolutivos, como o NSGA-II (do inglês, *Non-dominated Sorting Genetic Algorithm II*) e o MOEA (do inglês *Multi-objective Evolutionary Algorithm*).

Inspirados nos princípios de seleção natural de Darwin, esses algoritmos evolutivos oferecem uma alternativa interessante para lidar com a complexidade inerente a tais problemas. Seu funcionamento se baseia em uma população de soluções candidatas que evolui através da aplicação de operadores de seleção, cruzamento e mutação, gerando novas e melhores soluções. Uma breve descrição de alguns algoritmos evolutivos será apresentada na seção 1.3.

1.2 Método de Passo Adaptativo de Segunda Ordem

Nesta seção, são apresentados os principais conceitos envolvendo o método *Backtracking Line Search* com tamanho de passo adaptativo, proposto por Andrei (2004) e apresentado por Libotte et al. (2020) no contexto de otimização com confiabilidade. Esse método, baseado no princípio do gradiente descendente, busca determinar uma sequência de pontos que convirja para o ponto ótimo da função objetivo. Sua principal vantagem reside na capacidade de ajustar dinamicamente o tamanho do intervalo do passo durante a busca, o que pode acelerar a convergência para a solução ótima.

1.2.1 Backtracking Line Search

Considere uma função $G : \mathbb{R}^n \rightarrow \mathbb{R}$ diferenciável até a segunda ordem, que aqui será denominada de *função de desempenho*, que possui como ponto de mínimo \mathbf{u}^* . Utilizando o método do gradiente descendente, espera-se poder construir uma sequência de pontos $\mathbf{u}^{(k)}$ que convirja para \mathbf{u}^* . Neste caso, a sequência de pontos pode ser dada por:

$$\mathbf{u}^{(k+1)} = \mathbf{u}^{(k)} - \alpha_k p_k, \quad (3)$$

onde α_k é o tamanho do passo na iteração k e $p_k = \nabla G(\mathbf{u}^{(k)})$ é o gradiente da função G avaliado no ponto $\mathbf{u}^{(k)}$.

Do ponto de vista ideal, o tamanho do passo α_k poderia ser determinado minimizando a função de desempenho G ao longo da direção p_k , isto é,

$$\alpha_k = \arg \min_{\alpha > 0} G(\mathbf{u}^{(k)} - \alpha p_k). \quad (4)$$

Contudo, segundo Nocedal e Wright (2006), obter o valor exato de α_k pode ser computacionalmente custoso, e utilizam-se métodos inexatos de busca: uma alternativa é adotar um método de busca linear para encontrar uma sequência de valores para α , tal que, dentre esses valores, escolha-se aquele que proporciona a melhor redução no valor de $G(\mathbf{u}^{(k)} - \alpha p_k)$, a um custo computacional viável.

Na busca linear, um procedimento comum é o *backtracking line search*. Este método consiste em reduzir progressivamente o coeficiente α até que a condição $G(\mathbf{u}^{(k)} - \alpha p_k) < G(\mathbf{u}^{(k)}) - c\alpha p_k$ seja satisfeita. Aqui, $c \in (0, 1)$ representa um parâmetro de redução. Especificamente, a diminuição de α é implementada pela operação $\alpha \leftarrow \alpha\rho$. O Algoritmo 1 detalha este procedimento.

Neste contexto, Andrei (2004) introduziu um algoritmo baseado no método do gradiente descendente para otimização sem restrições, o qual permite determinar de maneira

Algoritmo 1 - Backtracking line search.

1. **Entrada:** $\mathbf{u}^{(0)}$, p_0 , α_0 , $c \in (0, 1)$, $\rho \in (0, 1)$
2. **Saída:** α_k
3. $\alpha \leftarrow \alpha_0$
4. **enquanto** $(G(\mathbf{u}^{(k)} - \alpha p_k) > G(\mathbf{u}^{(k)}) - c\alpha p_k)$, **fazer**
5. | $\alpha \leftarrow \rho\alpha$
6. **fim enquanto**
7. **return** α

adaptativa o intervalo para busca linear, sem exigir o cálculo da matriz Hessiana de G , cujo detalhamento é feito na subsecção 1.2.2.

1.2.2 Cálculo do Intervalo de Busca Linear Adaptativo

Partindo da ideia que a função de desempenho G é duas vezes diferenciável, o Teorema de Taylor para funções de várias variáveis permite afirmar que

$$\begin{aligned}
 \mathcal{G}_k(\alpha) &\triangleq G(\mathbf{u}^{(k+1)}) \\
 &= G(\mathbf{u}^{(k)} - \alpha p_k) \\
 &= G(\mathbf{u}^{(k)}) - \alpha p_k^\top p_k + \underbrace{\frac{1}{2} \alpha^2 p_k^\top \nabla^2 G(z) p_k}_{\text{Resto de Lagrange}}.
 \end{aligned} \tag{5}$$

para algum $z \in [\mathbf{u}^{(k)}, \mathbf{u}^{(k)} - \alpha p_k]$. Daí, tomando $z = \mathbf{u}^{(k)}$, pode-se construir aproximações quadráticas para \mathcal{G}_k e $G(\mathbf{u}^{(k)})$, respectivamente a saber,

$$\Gamma_k(\alpha) \triangleq G(\mathbf{u}^{(k)}) - \alpha p_k^\top p_k + \frac{1}{2} \alpha^2 p_k^\top \nabla^2 G(\mathbf{u}^{(k)}) p_k. \tag{6}$$

e

$$G(\mathbf{u}^{(k)}) \approx G(\mathbf{u}^{(k-1)}) - \alpha^{(k-1)} p_{(k-1)}^\top p_{(k-1)} + \frac{1}{2} (\alpha^{(k-1)})^2 p_{(k-1)}^\top \nabla^2 G(\mathbf{u}^{(k)}) p_{(k-1)}. \tag{7}$$

Derivando a Equação (6) em relação a α e igualando a zero, obtém-se o ponto de mínimo:

$$\tilde{\alpha}^k = \frac{p_k^\top p_k}{p_k^\top \nabla^2 G(\mathbf{u}^{(k)}) p_k}. \tag{8}$$

Assim, se G for convexa, é aceitável tomar o intervalo $[0, \tilde{\alpha}^k]$ como intervalo da busca linear. Contudo, esta abordagem ainda requer o cálculo da matriz Hessiana

$\nabla^2 G(\mathbf{u}^{(k)})$, o que pode ser computacionalmente dispendioso. Por outro lado, a partir da Equação (7), obtém-se que

$$p_k^\top \nabla^2 G(\mathbf{u}^{(k)}) p_k \approx \frac{2(G(\mathbf{u}^{(k)}) - G(\mathbf{u}^{(k-1)}) + \alpha^{(k-1)} p_{(k-1)}^\top p_{(k-1)})}{(\alpha^{(k-1)})^2}. \quad (9)$$

Isso permite que a determinação do intervalo de busca do passo seja feita sem a necessidade de calcular a matriz Hessiana, tomando

$$\bar{\alpha}^k = \frac{(\alpha^{(k-1)})^2 p_k^\top p_k}{2(G(\mathbf{u}^{(k)}) - G(\mathbf{u}^{(k-1)}) + \alpha^{(k-1)} p_{(k-1)}^\top p_{(k-1)})}. \quad (10)$$

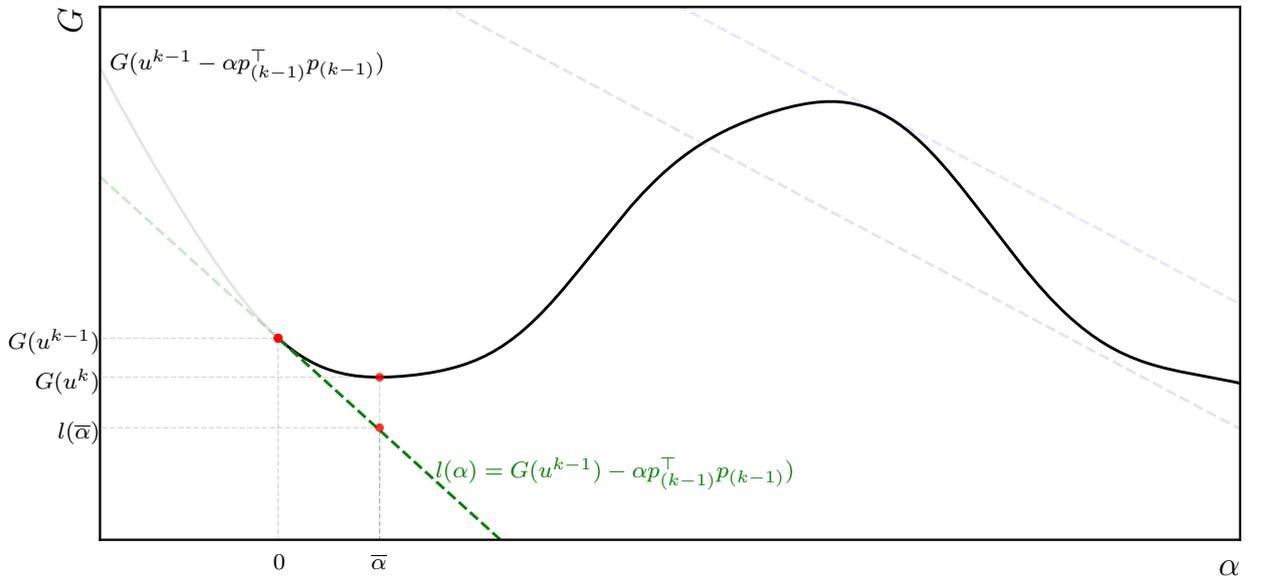
Aqui existem duas possibilidades: ou o denominador de $\bar{\alpha}^k$ é positivo e a função G é convexa em uma vizinhança do ponto u^{k-1} ou negativo e a função G é côncava em uma vizinhança do ponto u^{k-1} , conforme ilustrado nas Figuras 3 e 4.

De fato, afirmar que o denominador é positivo, equivale a dizer que

$$G(\mathbf{u}^{(k)}) > G(\mathbf{u}^{(k-1)}) - \alpha^{(k-1)} p_{(k-1)}^\top p_{(k-1)}, \quad (11)$$

ou seja, em uma vizinhança do ponto $\mathbf{u}^{(k-1)}$, a função de desempenho G está acima da reta tangente ao ponto $\mathbf{u}^{(k-1)}$, conforme é possível visualizar na Figura 3.

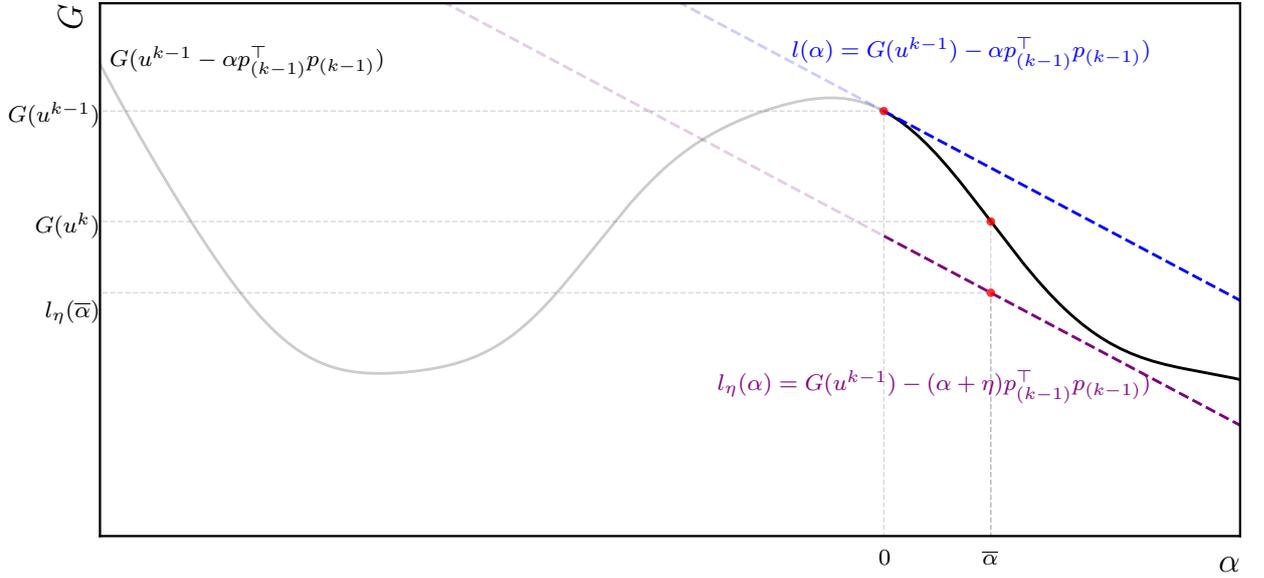
Figura 3 - Ilustração do método para o caso em que a função de desempenho G é convexa na vizinhança de um ponto u^{k-1} .



Fonte: O autor, 2024.

Analogamente, se $G(\mathbf{u}^{(k)}) < G(\mathbf{u}^{(k-1)}) - \alpha^{(k-1)} p_{(k-1)}^\top p_{(k-1)}$, então o gráfico da função de desempenho G está abaixo da reta tangente ao ponto $\mathbf{u}^{(k-1)}$, conforme a Figura

Figura 4 - Ilustração do método para o caso em que a função de desempenho G é côncava na vizinhança de um ponto u^{k-1} .



Fonte: O autor, 2024.

4. Contudo, ao deslocar verticalmente para baixo a reta tangente à curva, por um valor conveniente, o gráfico da função de desempenho passará a ficar acima da reta tangente.

De fato, considere η_{k-1} um número real tal que

$$G(\mathbf{u}^{(k)}) > G(\mathbf{u}^{(k-1)}) - (\alpha^{(k-1)} + \eta_{k-1}) p_{(k-1)}^\top p_{(k-1)}, \quad (12)$$

o que é equivalente a

$$\eta_{(k-1)} > \frac{1}{p_{(k-1)}^\top p_{(k-1)}} [G(\mathbf{u}^{(k-1)}) - G(\mathbf{u}^{(k)}) + \alpha^{(k-1)} p_{(k-1)}^\top p_{(k-1)}]. \quad (13)$$

Tomando $\delta > 0$ um número real suficientemente pequeno, pode-se definir

$$\eta_{(k-1)} = \frac{1}{p_{(k-1)}^\top p_{(k-1)}} [G(\mathbf{u}^{(k-1)}) - G(\mathbf{u}^{(k)}) + \alpha^{(k-1)} p_{(k-1)}^\top p_{(k-1)}] + \delta, \quad (14)$$

o que garante que α^k seja positivo. Neste caso, α^k passará a ser expresso como

$$\bar{\alpha}^k = \frac{(\alpha^{(k-1)} + \eta_k)^2 p_k^\top p_k}{2(G(\mathbf{u}^{(k)}) - G(\mathbf{u}^{(k-1)})) + (\alpha^{(k-1)} + \eta_k) p_{(k-1)}^\top p_{(k-1)}}. \quad (15)$$

Com isso, o intervalo de busca linear pode ser fixado como $[0, \bar{\alpha}^{(k)}]$, e o algoritmo pode ser resumido no Algoritmo 2.

Algoritmo 2 - Algoritmo de otimização de passo adaptativo de segunda ordem.

1. Selecione um ponto inicial $\mathbf{u}^{(0)} \in \mathbb{R}^n$
2. Obtenha o valor de $G(\mathbf{u}^{(0)})$
3. Determine o gradiente $p_0 = \nabla G(\mathbf{u}^{(0)})$
4. Calcule $\alpha^{(0)} = \arg \min_{0 < \alpha < 1} G(\mathbf{u}^{(0)} - \alpha p_0)$ utilizando backtracking
5. Calcule $\mathbf{u}^{(1)} = \mathbf{u}^{(0)} - \alpha^{(0)} p_0$
6. Determine o gradiente $p_1 = \nabla G(\mathbf{u}^{(1)})$
7. **enquanto** (Não atingir o critério de parada), **fazer**
8. | Calcule $\bar{\alpha}^{(k)}$ de acordo com a Equação (10)
9. | **se** ($\bar{\alpha}^{(k)}$ for negativo), **então**
10. | | Tome $\delta > 0$ um número real
11. | | Atualize o valor de $\bar{\alpha}^{(k)}$ de acordo com as equações (14) e (15)
12. | **fim se**
13. | Determine $\alpha^{(k-1)} = \arg \min_{0 < \alpha < \bar{\alpha}^{(k)}} G(\mathbf{u}^{(k-1)} - \alpha p_{k-1})$
14. | Atualize o ponto $\mathbf{u}^{(k)} = \mathbf{u}^{(k-1)} - \alpha^{(k-1)} p_{k-1}$ e o gradiente $p_k = \nabla G(\mathbf{u}^{(k)})$
15. **fim enquanto**

1.3 Algoritmos Evolucionários

Os algoritmos evolucionários constituem uma categoria de técnicas de otimização baseadas nos princípios da evolução natural. Em tais algoritmos, semelhante ao que ocorre na natureza, a competição por recursos propicia a seleção natural, ocasionando no melhoramento gradual da aptidão da população ao longo do tempo. A ideia principal envolve a geração de um conjunto de soluções candidatas, cuja qualidade é analisada por meio de uma função de aptidão. A partir dessa análise, são selecionadas as soluções mais promissoras para serem utilizadas na formação da próxima geração, através dos processos de recombinação e mutação (Eiben; Smith, 2015).

A recombinação genética envolve a combinação de genes oriundos de dois ou mais indivíduos (pais), originando assim novos indivíduos, ao passo que a mutação refere-se à alteração de um gene dentro de um indivíduo, resultando em uma nova solução. Este processo iterativo, que consiste nas etapas de seleção, recombinação e mutação, prossegue até que se encontre uma solução que satisfaça os critérios de qualidade estabelecidos ou até que algum critério de parada previamente determinado seja atingido.

Os mecanismos de variação, tais como a recombinação e a mutação, geram a diversidade dentro da população, enquanto o processo de seleção melhora a qualidade média das soluções, proporcionando a evolução, que pode ser interpretada como um processo de otimização ou adaptação, no qual a aptidão indica o grau de conformidade com as exigências do ambiente. Vários componentes deste processo são de natureza estocástica; em particular, a seleção e a recombinação são realizadas de maneira aleatória. Essa

aleatoriedade possibilita que mesmo os indivíduos menos aptos tenham a oportunidade de contribuir para a próxima geração, promovendo assim a diversidade e evitando uma convergência prematura para ótimos locais (Eiben; Smith, 2015).

Essa abordagem de evolução e adaptação é a base dos algoritmos evolutivos que formam um extenso conjunto de métodos de otimização que são inspirados na evolução natural. Entre esses algoritmos, destacam-se os Algoritmos Genéticos e o Algoritmo de Ordenação Não Dominada II, que são utilizados nesta tese.

1.3.1 Algoritmo Genético

Segundo Goldberg (1989), os Algoritmos Genéticos (GA) combinam a sobrevivência dos mais aptos com uma troca de informações estruturada e aleatória para formar um algoritmo de busca. Esses algoritmos foram desenvolvidos por John Holland e sua equipe na Universidade de Michigan, e são amplamente utilizados em problemas de otimização devido à sua capacidade de explorar grandes espaços de busca.

Os algoritmos genéticos são o tipo mais conhecido de algoritmo evolucionário, sendo que o algoritmo genético clássico, ou “simple GA” (SGA), utiliza representação binária, seleção proporcional à aptidão, baixa probabilidade de mutação e recombinação genética para gerar novas soluções. Apesar de sua simplicidade, o SGA é amplamente usado para *benchmarking* de novos algoritmos. Ao longo dos anos, melhorias como elitismo e modelos não-geracionais foram adicionadas para acelerar a convergência. Além disso, alternativas como o *crossover* uniforme e a adaptação automática das taxas de mutação foram desenvolvidas para melhorar a eficiência dos GAs (Eiben; Smith, 2015).

No seu funcionamento, inicialmente gera-se uma população de forma aleatória utilizando uma estratégia de amostragem pré-definida como o hipercubo latino, por exemplo, onde cada membro da população representa uma solução potencial para o problema. Posteriormente, cada indivíduo passa por uma avaliação por meio da função objetivo do problema, o que determina sua aptidão, ou seja, a qualidade da solução que ele representa. Esta avaliação é decisiva para o processo de seleção natural, onde indivíduos mais aptos são mais propensos a sobreviver e a gerar descendentes, refletindo a ideia de “sobrevivência do mais apto”, ou seja, indivíduos com maior aptidão são selecionados para a próxima geração, enquanto os menos aptos são descartados.

Logo depois, ocorre a fase de recombinação, ou cruzamento, que se segue à seleção. Nesta etapa, pares de indivíduos são selecionados para trocar segmentos de suas estruturas genéticas, resultando em novos descendentes. Subsequentemente, aplica-se a mutação aos descendentes com certa probabilidade, promovendo pequenas mudanças aleatórias nos genes, incentivando a diversidade genética e prevenindo que a população atinja prematuramente soluções que não sejam ideais. Este processo é descrito no Algoritmo 3.

Algoritmo 3 - Algoritmo Genético.

ALGORITMO GENÉTICO

1. Inicialize a população P_0 com N indivíduos
2. Avalie a população P_0
3. **enquanto** (Não atingir o critério de parada), **fazer**
4. | Seleccione os pais da população P_t usando torneio binário ou roleta
5. | Aplique operadores de recombinação (crossover) para gerar a população filha Q_t
6. | Aplique operadores de mutação na população filha Q_t
7. | Avalie a população Q_t
8. | Combine as populações P_t e Q_t para formar R_t
9. | Seleccione os N melhores indivíduos de R_t para formar a nova população P_{t+1}
10. **fim enquanto**

FIM ALGORITMO

Apesar de o Algoritmo Genético ter demonstrado uma eficiência na resolução de problemas de otimização com um único objetivo, ele enfrenta limitações quando aplicado a problemas que envolvem múltiplos objetivos, não sendo a escolha mais apropriada. Para superar essas limitações, uma variedade de algoritmos de otimização multiobjetivo tem sido desenvolvida. Entre esses, destaca-se o NSGA-II, apresentado por Deb et al. (2002), cujo funcionamento será abordado na subseção subsequente.

1.3.2 Algoritmo de Ordenação Não Dominada II

A primeira versão do NSGA foi proposta por Srinivas e Deb (1994) e posteriormente aprimorada por Deb et al. (2002), resultando no conhecido NSGA-II, o qual atualmente possui várias outras variantes (Ma et al., 2023). O NSGA-II é amplamente utilizado em problemas de otimização multiobjetivo devido à sua capacidade de encontrar múltiplas soluções pareto-ótimas em uma única execução do algoritmo, classificando soluções com base em níveis de não dominância e utilizando um operador de distância de aglomeração (*crowding distance*), para manter a diversidade da população.

Conforme indicado por Mirjalili e Dong (2020), a melhora introduzida no NSGA original, conhecida como NSGA-II, demonstrou vantagens em relação ao seu predecessor, particularmente em termos de eficiência computacional e implementação do elitismo.

Além do uso do elitismo, a sua superioridade é atribuída à aplicação de uma técnica de ordenação baseada em não dominância. Esta técnica emprega um modelo hierárquico para organizar os diferentes níveis de não dominância, o que é importante para diminuir o custo computacional, no sentido de eliminar a necessidade de comparar

todos os indivíduos após a primeira análise de níveis. Além disso, o seu desempenho é influenciado pelo operador que determina a distância de aglomeração, que não requer a definição de parâmetros prévios.

No algoritmo proposto por Deb et al. (2002), e apresentado brevemente no Algoritmo 4, existem dois procedimentos principais: divisão dos indivíduos da população em frentes de dominância; e um segundo procedimento que calcula a distância de aglomeração em relação aos indivíduos mais próximos.

No procedimento inicial, são identificados os indivíduos p que não são dominados, estabelecendo-se para eles um contador de dominância com valor $n_p = 0$, o que indica que p não é dominado por nenhum outro indivíduo. Para cada indivíduo q que é dominado por p , o contador de dominância n_q é diminuído em uma unidade. Caso o contador de dominância de algum indivíduo se torne zero, este indivíduo é inserido em uma lista separada, que representa a segunda frente não dominada. Este processo é repetido para todos os membros da segunda frente, com o intuito de determinar a terceira frente, e assim sucessivamente, até que todas as frentes sejam completamente identificadas.

Algoritmo 4 - Algoritmo NSGA-II.

ALGORITMO NSGA-II

1. Inicialize a população P_0 com N indivíduos
 2. Avalie a população P_0
 3. Classifique a população P_0 em frentes de não-dominância
 4. Calcule a distância de aglomeração (crowding distance) para cada indivíduo
 5. **enquanto** (Não atingir o critério de parada), **fazer**
 6. | Selecione os pais da população P_t usando torneio binário baseado na classificação de não-dominância e na distância de aglomeração
 7. | Aplique operadores de recombinação e mutação para gerar a população filha Q_t
 8. | Avalie a população Q_t
 9. | Combine as populações P_t e Q_t para formar R_t
 10. | Classifique a população R_t em frentes de não-dominância
 11. | Calcule a distância de aglomeração para cada indivíduo em R_t
 12. | Selecione os N melhores indivíduos de R_t para formar a nova população P_{t+1} , com base na classificação de não-dominância e na distância de aglomeração
 13. **fim enquanto**
- FIM ALGORITMO

No segundo procedimento, seguindo a descrição do algoritmo proposto por Deb et al. (2002), calcula-se a distância de aglomeração, do inglês *crowding distance*, utilizada para estimar a densidade de soluções. Para tanto, os valores de cada função objetivo são

organizados em ordem crescente, atribuindo-se uma distância de aglomeração infinita aos valores extremos, ou seja, aos menores e maiores valores. As soluções situadas entre o menor e o maior valor possuem uma distância de aglomeração correspondente à diferença normalizada absoluta entre os valores das funções de duas soluções adjacentes. Esse procedimento é executado para todos os objetivos, sendo que o valor total da distância de aglomeração é a soma dos valores individuais de distância relativos a cada objetivo.

Com a determinação destas medidas, é possível determinar a distância entre todos os membros da população e assim afirmar que uma solução com um valor menor dessa métrica é mais aglomerada do que outras soluções.

O Algoritmo 4 fornece uma visão sucinta de como o NSGA-II opera. Para um entendimento mais aprofundado sobre este algoritmo, recomenda-se a leitura de Deb et al. (2002), Mirjalili e Dong (2020), onde os conceitos e formulações são discutidos mais detalhadamente.

1.3.3 Indicadores de eficiência

Em um contexto de otimização com um único objetivo, a eficiência de um algoritmo pode ser avaliada pela proximidade da solução gerada em relação a um valor ótimo previamente conhecido. Em contraste, em cenários de otimização multiobjetivo, não há um único valor ótimo, mas sim um conjunto de soluções que muitas vezes não é conhecido, o que exige a utilização de métodos alternativos para avaliar a eficiência. Dois indicadores amplamente empregados são: a Distância Geracional (GD), que quantifica a distância entre as soluções geradas pelo algoritmo e um conjunto conhecido de soluções Pareto-ótimas; o Hipervolume (HV), que é uma métrica que mede o volume no espaço n -dimensional dos objetivos, considerando a região das soluções dominadas em relação a um ponto de referência.

Seja $P = \{p^{(1)}, p^{(2)}, \dots, p^{(|P|)}\}$ um conjunto de valores obtido no espaço objetivo a partir de um algoritmo multiobjetivo, ou seja, a fronteira de Pareto determinada pelo algoritmo e seja $Z = \{z_1, z_2, \dots, z_{|Z|}\}$ a fronteira de Pareto de referência. Então a distância geracional pode ser expressa por:

$$GD(P) = \frac{1}{|P|} \left(\sum_{i=1}^{|P|} d_i^2 \right)^{\frac{1}{2}}, \quad (16)$$

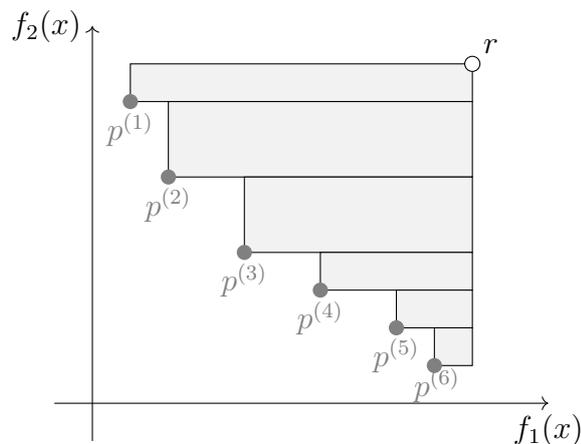
onde d_i denota a distância euclidiana de $p^{(i)}$ até seu ponto de referência mais próximo em Z .

Uma GD pequena sugere que as soluções produzidas pelo algoritmo multiobjetivo são mais próximas do conjunto de soluções Pareto-ótimas conhecidas.

Já o hipervolume, conforme detalhado em Fonseca, Paquete e Lopez-Ibanez (2006),

em um problema de minimização com d objetivos, quantifica o espaço simultaneamente dominado pelo conjunto P e delimitado por um ponto de referência $r \in \mathbb{R}^d$. Este ponto r deve satisfazer a condição $r \geq (\max p^{(1)}, \dots, \max p^{(d)})$, onde a relação de desigualdade \geq é analisada coordenada a coordenada. A área (hipervolume) resultante é um polítopo ortogonal, visualizado como a união de n hiperretângulos alinhados aos eixos, originado de um vértice comum, o ponto de referência r . Esta ideia é ilustrada para o caso bidimensional na Figura 5.

Figura 5 - Indicador de hipervolume para o caso bidimensional.



Fonte: Adaptado de Fonseca, Paquete e Lopez-Ibanez (2006).

Desta forma, conforme Coello, Lamont e Veldhuizen (2007), o hipervolume pode ser definido matematicamente por

$$\mathbf{HV} \triangleq \left\{ \bigcup_i vol_i \mid p^{(i)} \in \text{PF}_{\text{Conhecida}} \right\}, \quad (17)$$

onde $\bigcup_i vol_i$ denota a união dos volumes dominados por cada solução $p^{(i)}$ pertencente à fronteira de Pareto conhecida $\text{PF}_{\text{Conhecida}}$.

Um valor elevado de HV indica que o conjunto de soluções domina uma região maior no espaço dos objetivos, o que sugere uma maior diversidade e uma melhor qualidade.

1.4 Otimização Robusta

No âmbito da otimização, a robustez refere-se a soluções que exibem baixa sensibilidade a perturbações externas em problemas sujeitos a incertezas. Tais incertezas podem advir de diversos fatores, dentre os quais se destacam os erros de medição, imprecisões geométricas, erros de modelagem e erros de arredondamento, entre outros. Em

consequência, uma solução ótima de um problema que desconsidera a robustez (conhecido como problema nominal) pode revelar-se inviável quando aplicada na execução de um projeto de engenharia.

Essa constatação permite afirmar que pode não ser conveniente adotar uma determinada solução ótima se a função objetivo for sensível a pequenas perturbações, ou seja, quando pequenas alterações no ponto ótimo resultam em significativas variações no valor da função objetivo. Este conceito é particularmente relevante em áreas onde a segurança em relação a variações é imprescindível, como no ajuste de parâmetros de sistemas de controle aeroespacial ou de energia nuclear (Tsutsui; Ghosh; Fujimoto, 1996).

Assim, dada uma função $f : \mathbb{R}^n \rightarrow \mathbb{R}$, ao se levar em conta a robustez, o objetivo é encontrar um vetor $\mathbf{x} \in \mathbb{R}^n$ que minimiza f e com a garantia de que o desempenho do sistema é mantido mesmo com a ocorrência de pequenas variações. Com isso, seguindo os passos de Deb e Gupta (2005), pode-se elaborar a seguinte definição:

Definição 2. Para um problema de minimização de uma função $f : X_f \subset \mathbb{R}^n \rightarrow \mathbb{R}$, onde X_f é a região viável, uma solução \mathbf{x}^* é dita solução robusta do tipo I se ela é o ponto de mínimo global da função de média efetiva com respeito a uma δ -vizinhança, onde

$$f^{\text{eff}}(\mathbf{x}) \triangleq \frac{1}{|\mathcal{B}_\delta(\mathbf{x})|} \int_{\xi \in \mathcal{B}_\delta(\mathbf{x})} f(\xi) d\xi, \quad (18)$$

e $|\mathcal{B}_\delta(\mathbf{x})|$ representa o hipervolume da δ -vizinhança centrada em \mathbf{x} com raio δ .

Contudo, o processo de cálculo da integral que aparece na Equação (18) pode ser de difícil execução ou até inviável. Portanto, adota-se uma aproximação para essa função de média efetiva, representada por

$$f^{\text{eff}}(\mathbf{x}) \approx \frac{1}{N} \sum_{i=1}^N f(\xi_i), \quad (19)$$

onde $\xi_i \in \mathcal{B}_\delta(\mathbf{x})$ e N é o número de amostras dentro da δ -vizinhança de \mathbf{x} .

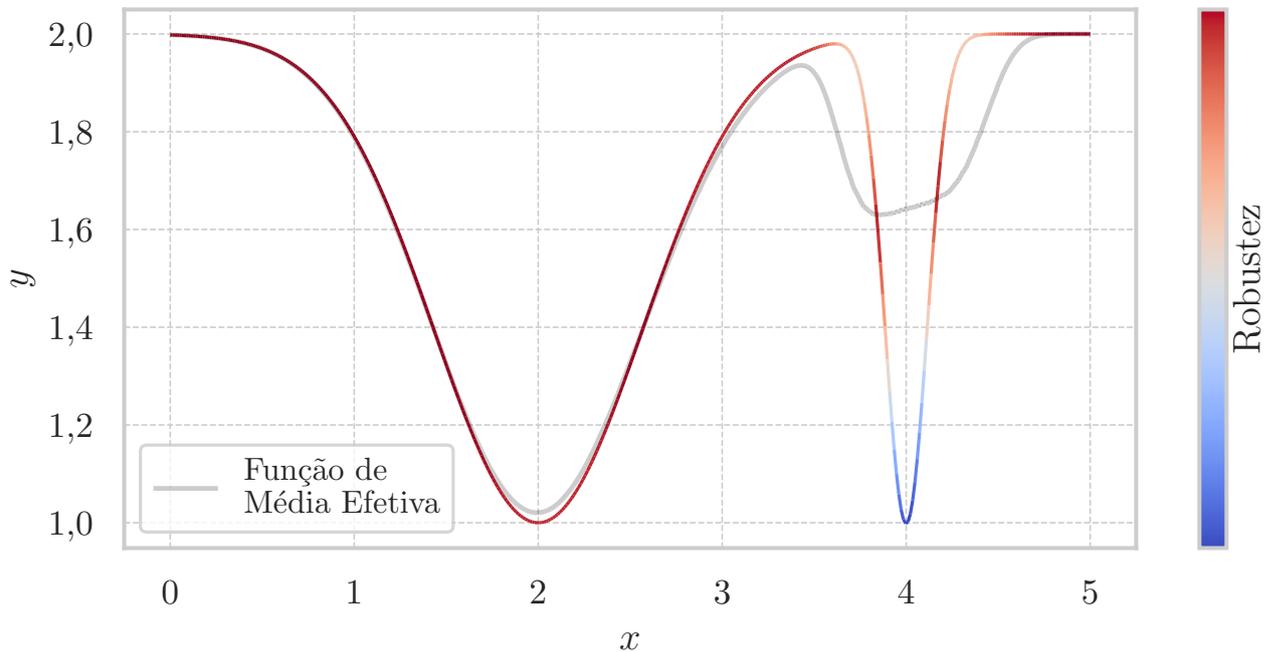
Exemplo 2. Considere a função $f : \mathbb{R} \rightarrow \mathbb{R}$ definida por:

$$f(x) = -\exp\left(-625\left(\frac{x}{4} - 1\right)^2\right) - \exp\left(-25\left(\frac{x}{4} - 0,5\right)^2\right) + 2, \quad (20)$$

que possui dois pontos de mínimo, que ocorrem em $x = 2$ e $x = 4$, onde $f(2) = f(4) = 1$.

Prosseguindo com a busca pelo ponto robusto, através da minimização da função de média efetiva, e considerando uma variação de 10%, ou seja, um índice de robustez de 10%, no intervalo $[0, 5]$, obtém-se uma solução robusta de tipo I em $\mathbf{x}^* = 2$, onde $f^{\text{eff}}(2) \approx 1,0203$. Em contrapartida, obtém-se $f^{\text{eff}}(4) \approx 1,6392$, o que sustenta a afirmação

Figura 6 - Gráficos $f(x) = -\exp(-625(\frac{x}{4} - 1)^2) - \exp(-25(\frac{x}{4} - 0,5)^2) + 2$ e da respectiva função de média efetiva.



Legenda: Nesta ilustração, o gráfico da função f é apresentado em conjunto com o da função de média efetiva, ambos posicionados lado a lado.

Fonte: O autor, 2024.

de que f apresenta maior sensibilidade a variações no ponto $x = 4$ comparativamente ao ponto $x = 2$.

A análise dos gráficos de f e f^{eff} , conforme ilustrado na Figura 6, reforça esta conclusão: observa-se que as duas funções exibem comportamentos bastante semelhantes ao longo da maior parte do intervalo $[0, 5]$. No entanto, em proximidade de $x = 4$, pode-se notar que a função de média efetiva atinge valores mais elevados.

Em outras palavras, um problema de otimização robusta do tipo I caracteriza-se pela determinação de um vetor \mathbf{x}^* que minimiza ou maximiza $f^{\text{eff}}(\mathbf{x})$, condicionado a um conjunto de restrições:

$$\begin{aligned}
 &\underset{\mathbf{x}}{\text{Minimizar}} && f^{\text{eff}}(\mathbf{x}) \\
 &\text{Sujeito a} && g_i(\mathbf{x}) \leq 0, \quad i = 1, \dots, m, \\
 & && h_j(\mathbf{x}) = 0, \quad j = 1, \dots, p. \\
 & && a_\ell \leq x_\ell \leq b_\ell, \quad \ell = 1, \dots, n.
 \end{aligned} \tag{21}$$

Definição 3. Para um problema de minimização de uma função $f : X_f \subset \mathbb{R}^n \rightarrow \mathbb{R}$, onde X_f é a região viável, uma solução \mathbf{x}^* é dita solução robusta do tipo II se ela é o mínimo global de f sujeito a $\frac{\|f^p(\mathbf{x}) - f(\mathbf{x})\|}{\|f(\mathbf{x})\|} \leq \eta$, onde f^p pode ser selecionado como a função de média efetiva f^{eff} , ou como o pior resultado dentre os N pontos na δ -vizinhança $\mathcal{B}_\delta(\mathbf{x})$

e η é o parâmetro de controle da perturbação.

Como afirmado por Deb e Gupta (2005), a caracterização de soluções ótimas robustas pode ser facilmente ampliada para incluir problemas de otimização multiobjetivo. Em tais problemas não se terá uma solução robusta que será o mínimo global, mas sim um conjunto de Pareto onde todas as soluções são menos sensíveis às perturbações. Para tanto, cada objetivo deve ser avaliado seguindo um dos critérios abordados acima, o que adiciona uma camada de complexidade ao processo de otimização. Assim pode-se definir:

Definição 4. *Uma solução \mathbf{x}^* é denominada solução robusta multiobjetivo se ela satisfizer uma das seguintes condições:*

1. **Solução Robusta Multiobjetivo do Tipo I:** \mathbf{x}^* é um otimizador de Pareto para um problema de minimização multiobjetivo, com respeito à função de média efetiva

$$\mathbf{f}^{\text{eff}}(\mathbf{x}) = \left(f_1^{\text{eff}}(\mathbf{x}), \dots, f_m^{\text{eff}}(\mathbf{x}) \right). \quad (22)$$

onde $\mathbf{x} \in \mathbf{X}_f$.

2. **Solução Robusta Multiobjetivo do Tipo II:** \mathbf{x}^* é solução para o problema de minimização:

$$\begin{aligned} \underset{\mathbf{x}}{\text{Minimizar}} \quad & \mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), \dots, f_m(\mathbf{x})) \\ \text{Sujeito a} \quad & \frac{\|\mathbf{f}^p(\mathbf{x}) - \mathbf{f}(\mathbf{x})\|}{\|\mathbf{f}(\mathbf{x})\|} \leq \eta, \end{aligned} \quad (23)$$

onde $\mathbf{x} \in \mathbf{X}_f$, $\mathbf{f}^p(\mathbf{x})$ denota o vetor objetivo distorcido, η é um parâmetro de controle para a perturbação, e $\|\cdot\|$ denota a norma euclidiana.

Nesse contexto, o vetor objetivo \mathbf{f}^p pode ser selecionado como a função de média efetiva \mathbf{f}^{eff} , ou como o pior resultado dentre os N pontos na δ -vizinhança $\mathcal{B}_\delta(x)$, isto é,

$$\mathbf{f}^p(\mathbf{x}) = \left\{ \max_{i=1, \dots, N} f_k(\mathbf{x}_i) \right\}, \quad (24)$$

para $k = 1, \dots, m$.

Embora a robustez seja de grande importância na determinação de soluções que exibem menor variação em sua imagem mesmo quando perturbadas na proximidade dos pontos do domínio correspondente, essas soluções podem encontrar-se próximas à fronteira do conjunto viável no domínio da função objetivo. Esta proximidade acentua a probabilidade de que ocorra uma violação das restrições impostas. Nesse contexto, torna-se relevante que as soluções encontradas atendam a uma probabilidade específica de violação das restrições, assunto que será explorado na seção 1.5, que é dedicada à confiabilidade.

1.5 Otimização Baseada em Confiabilidade

No desenvolvimento de um projeto de engenharia, é frequentemente importante definir parâmetros de segurança para garantir que o sistema funcione adequadamente. Um exemplo claro disso se dá no projeto de uma estrutura de treliças, que pode ter variações nas áreas de suas seções transversais, e sua performance pode ser afetada por diversos fatores ambientais e operacionais, como a ação de ventos, precipitação pluviométrica, variações de temperatura e a carga aplicada.

Dada a influência desses fatores, pode ser relevante minimizar a probabilidade de falha do sistema, aumentando assim sua confiabilidade. A confiabilidade de um sistema é quantificada pela chance de falha, que é medida pela probabilidade de que os parâmetros estabelecidos para o sistema não cumpram alguma das restrições definidas durante a elaboração do projeto (Haldar; Mahadevan, 2000).

Sendo assim, considere um problema de otimização com restrições, dados por

$$\begin{aligned}
 &\underset{\mathbf{x}}{\text{Otimizar}} && f(\mathbf{x}) \\
 &\text{Sujeito a} && g(\mathbf{x}) \geq 0, \\
 & && h(\mathbf{x}) = 0, \\
 & && a \leq \mathbf{x} \leq b,
 \end{aligned} \tag{25}$$

onde f é a função objetivo, g e h são as funções de restrição, e a e b são os limites inferior e superior, respectivamente, do vetor de variáveis de projeto \mathbf{x} .

A probabilidade de falha considerando a restrição g é definida por

$$P_f = P[g(X) \leq 0] = \int_{g(X) \leq 0} f_X(\mathbf{x}) d\mathbf{x}, \tag{26}$$

onde $f_X(\mathbf{x})$ é a função densidade de probabilidade conjunta de $X = (X_1, X_2, \dots, X_n)$, que é um vetor de n variáveis aleatórias contínuas. Conforme destacado por Lopez e Beck (2012), resolver a integral na Equação (26) é, em geral, inviável, devido à escassez de observações para definir a distribuição de probabilidade conjunta $f_X(\mathbf{x})$ e à complexidade de se lidar com uma integral multidimensional onde a função de desempenho g pode não ter uma forma explícita.

Para aproximar a integral da Equação (26), assume-se em geral que $X \sim \mathcal{N}(\mu_X, \sigma_X^2)$, com μ_X como média e σ_X^2 como variância, e que as variáveis aleatórias X são independentes e normais contínuas, representando os parâmetros do sistema, sendo que $g(X) = 0$ define a função de estado limite, a qual separa a zona segura da zona de falha.

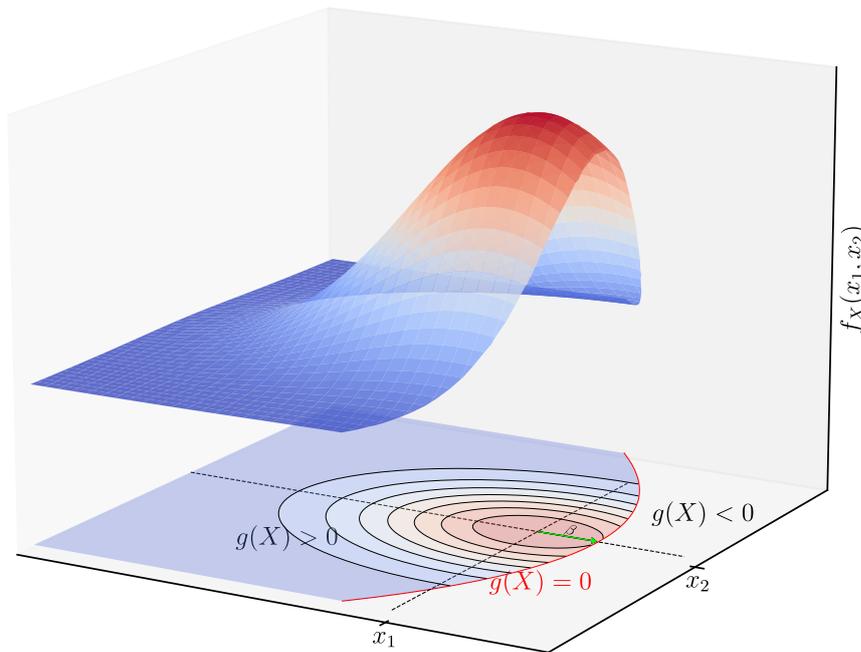
A partir destas informações, imediatamente pode-se inferir que a probabilidade de

sucesso é dada por

$$P_s = 1 - P_f = \int_{g(X) > 0} f_X(\mathbf{x}) d\mathbf{x}. \quad (27)$$

Na Figura 7, $g(X) < 0$ representa o conjunto de falha e $g(X) \geq 0$ o de sucesso, enquanto $f_X(\mathbf{x})$ é a função densidade conjunta para a distribuição normal.

Figura 7 - Ilustração da região segura e de falha.



Fonte: O autor, 2024.

A partir da suposição de que $X \sim \mathcal{N}(\mu_X, \sigma^2)$ e definindo $U = \frac{X - \mu_X}{\sigma_X}$ para mapear o espaço das variáveis aleatórias X no espaço normal padrão U , ou seja, aplicando a *Transformação de Rosenblatt* (Rosenblatt, 1952), obtém-se que $U \sim \mathcal{N}(0, 1)$. Dessa forma, a probabilidade de falha pode ser reescrita como

$$\begin{aligned} P_f &= 1 - P[g(X) \geq 0] \\ &= 1 - P[g(\sigma_X U + \mu_X) \geq 0] \\ &= 1 - P[G(U) \geq 0] \\ &= 1 - \int_{G(U) \geq 0} \phi(u) du, \end{aligned} \quad (28)$$

onde $\phi(u)$ é a função densidade de probabilidade da distribuição normal padrão e $G(U) = g(\sigma_X U + \mu_X)$.

Mas observe que pelo fato das variáveis aleatórias serem independentes, a função

densidade de probabilidade conjunta será dada por

$$\begin{aligned}\phi(u) &= \prod_{i=1}^n \phi(u_i) \\ &= \frac{1}{(2\pi)^{n/2}} \exp\left(-\frac{\|u\|^2}{2}\right)\end{aligned}\quad (29)$$

Contudo, ao se considerar as hipersuperfícies de nível $\phi(u) = c$, c constante, obtém-se hiperesferas centradas na origem, cuja aquela de maior raio dentro da região segura determinará o(s) ponto(s) da função de estado limite mais próximo(s) da origem.

De acordo com Haldar e Mahadevan (2000), após a transformação descrita acima para o espaço normal padrão, a menor distância β entre a origem e a fronteira $G(U) = 0$ entre as regiões segura e de falha é denominada *índice de confiabilidade* ou fator de segurança. O ponto sobre a curva $G(U) = 0$ que está a menor distância da origem recebe o nome de ponto mais provável de falha (MPP, de *Most Probable Point*).

Segue daí que para encontrar o ponto com maior probabilidade de falha, é necessário encontrar o ponto que minimiza a função $\beta(u) = \|u\|$, ou seja,

$$u^* = \begin{cases} \arg \min_u \|u\| \\ \text{sujeito a } G(u) = 0. \end{cases}\quad (30)$$

Este conceito é demonstrado na Figura 8a e, de acordo com Tu, Choi e Park (1999), é denominado *Reliability Index Approach* (RIA).

Apesar da simplificação tomada com a padronização das variáveis, o cálculo da integral dada na Equação (28) ainda enfrenta dificuldade devido às possíveis características da função G , que pode ser dada de forma implícita, por exemplo.

Seguindo Lopez e Beck (2012), uma abordagem que pode ser utilizada para contornar esta dificuldade com a função de desempenho G no cálculo da integral, é a sua substituição por um hiperplano tangente à curva estado limite no ponto mais provável de falha, caracterizando o método de confiabilidade de primeira ordem (FORM, de *First Order Reliability Method*).

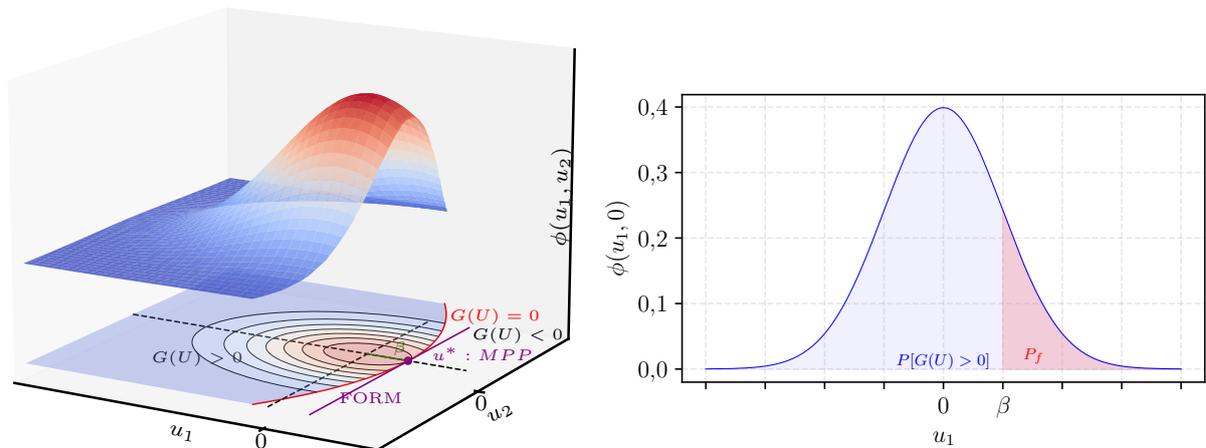
Assim,

$$\begin{aligned}P_f &= 1 - \int_{G(U) \geq 0} \phi(u) du \\ &\approx 1 - \Phi(\beta) \\ &= \Phi(-\beta),\end{aligned}\quad (31)$$

onde Φ é a distribuição acumulada da normal padrão.

A Figura 9 mostra a probabilidade de falha para diferentes valores de β na distribui-

Figura 8 - Ilustração da região segura e de falha no espaço normal padrão.



(a) Distribuição de probabilidade conjunta gaussiana restrita ao conjunto $G(U) > 0$.

(b) Corte transversal da distribuição gaussiana, ilustrando a região referente $P[G(U) \geq 0]$, com destaque para o local β , onde localiza-se o MPP.

Fonte: O autor, 2024.

ção normal padrão. Nota-se que um aumento em β leva a uma redução na probabilidade de falha e, por exemplo, $\beta = 3$ resulta em aproximadamente 0,13% de probabilidade de falha, ou seja, implicando 99,87% de sucesso.

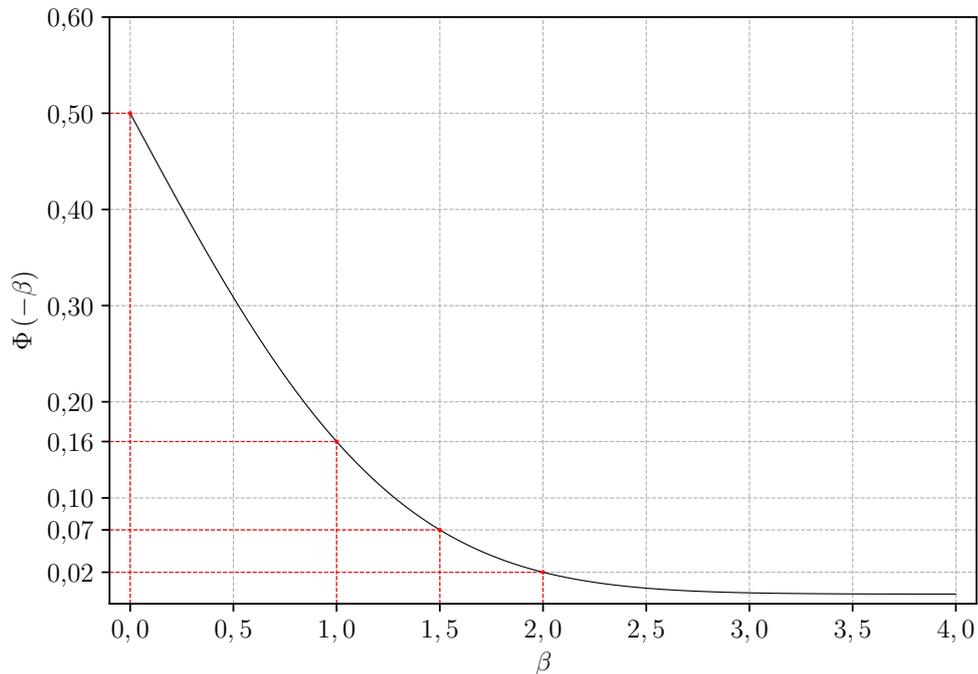
Outra abordagem para a busca do ponto mais provável de falha é aquela conhecida como Abordagem de Medida de Desempenho (PMA, de *performance measure approach*), ou análise de confiabilidade inversa e foi apresentada por Tu, Choi e Park (1999). Neste caso, fixa-se o índice de confiabilidade alvo β_t e busca-se na hipersfera de raio β_t o ponto que minimiza a função de desempenho $G(u)$, ou seja,

$$u^* = \begin{cases} \arg \min_u G(u) \\ \text{sujeito a } \|u\| = \beta_t. \end{cases} \quad (32)$$

É pertinente discutir a distinção entre as abordagens RIA e PMA: na abordagem RIA, o MPP é obtido por meio da minimização da norma \mathbf{u} sujeito à condição $G(\mathbf{u}) = 0$, determinando, assim, o ponto mais próximo da origem no espaço normal padrão. Por outro lado, na abordagem PMA, fixa-se o índice de confiabilidade alvo β_t e minimiza-se a função de desempenho G no espaço normal padrão, sujeito à condição $\|\mathbf{u}\| = \beta_t$. Quando o ponto de mínimo \mathbf{u}^* da função G não coincide com a curva de estado limite $G(\mathbf{u}) = 0$, \mathbf{u}^* é denominado ponto-alvo mínimo de desempenho (MPTP, de *Minimum Performance Target Point*).

Dessa maneira, conforme Du e Chen (2004), um problema de otimização funda-

Figura 9 - Probabilidade de falha para diferentes valores de β .



Legenda: Nesta figura é ilustrada a probabilidade de falha para diferentes valores de β , onde Φ é a função de distribuição acumulada da distribuição normal padrão.

Fonte: O autor, 2024.

mentado em confiabilidade pode ser formulado como:

$$\begin{aligned}
 & \underset{d}{\text{Minimizar}} \quad \mathbf{f}(\mathbf{d}) \\
 & \text{Sujeito a} \quad P_f = P[g(d, X) \leq 0] \leq \Phi(-\beta_t), \\
 & \quad \quad \quad d^{\min} \leq d \leq d^{\max}
 \end{aligned} \tag{33}$$

em que P_f representa a probabilidade máxima de falha aceita para a restrição g e β_t o índice de confiabilidade alvo.

No exemplo abaixo, é abordado um problema no qual as funções de desempenho são analisadas ao longo de uma circunferência de raio β_t . Isto permite verificar que, à medida que o valor da função de desempenho diminui ao longo da circunferência, mais próximo se encontra do ponto mais provável de falha.

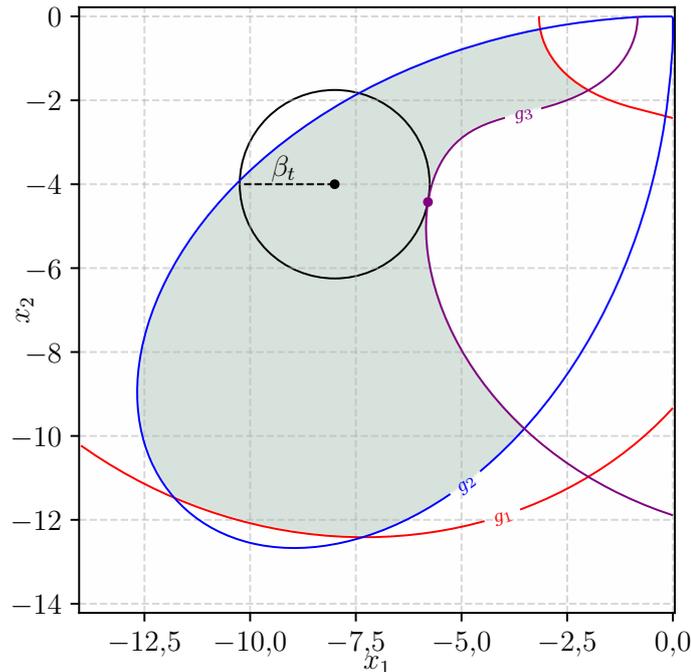
Exemplo 3. Considere três funções de desempenho $g_1, g_2, g_3 : B \subset \mathbb{R}^2 \rightarrow \mathbb{R}$, onde $B = [-13, 0] \times [-13, 0]$, definidas por:

$$\begin{aligned}
 g_1(\mathbf{x}) &= - \left[(\bar{x}_1^2 + \bar{x}_2^2 + 2r_1\bar{x}_1)^2 - 4r_1^2 (\bar{x}_1^2 + \bar{x}_2^2) + 2 \right], \\
 g_2(\mathbf{x}) &= - \left[(\bar{x}_1^2 + \bar{x}_2^2)^3 - 4r_2\bar{x}_1^2\bar{x}_2^2 \right], \\
 g_3(\mathbf{x}) &= - \left[((\bar{x}_1 + 1)^2 + \bar{x}_2^2 - 2r_3(\bar{x}_1 + 1))^2 + 4r_3^2 ((\bar{x}_1 + 1)^2 + \bar{x}_2^2) - 2 \right],
 \end{aligned}$$

com $r_1 = 1,2054$, $r_2 = 1,6936 \times 10^1$, $r_3 = 1,2054$ e $\mathbf{x} = (\bar{x}_1, \bar{x}_2) = (\frac{x_1}{4}, \frac{x_2}{4})$ com $X_1 \sim \mathcal{N}(-8, 1)$ e $X_2 \sim \mathcal{N}(-4, 1)$ e $\beta_t = 2,2464$, indicando o índice de confiabilidade alvo.

Com este índice de confiabilidade, ocorre uma violação da restrição g_2 , conforme ilustrado na Figura 11b, onde a função de desempenho restrita à circunferência de raio β_t e centro $(-8, -4)$ assume valores negativos, especificamente quando a circunferência se encontra fora da área sombreada na Figura 10.

Figura 10 - Gráficos das funções de estado limite.



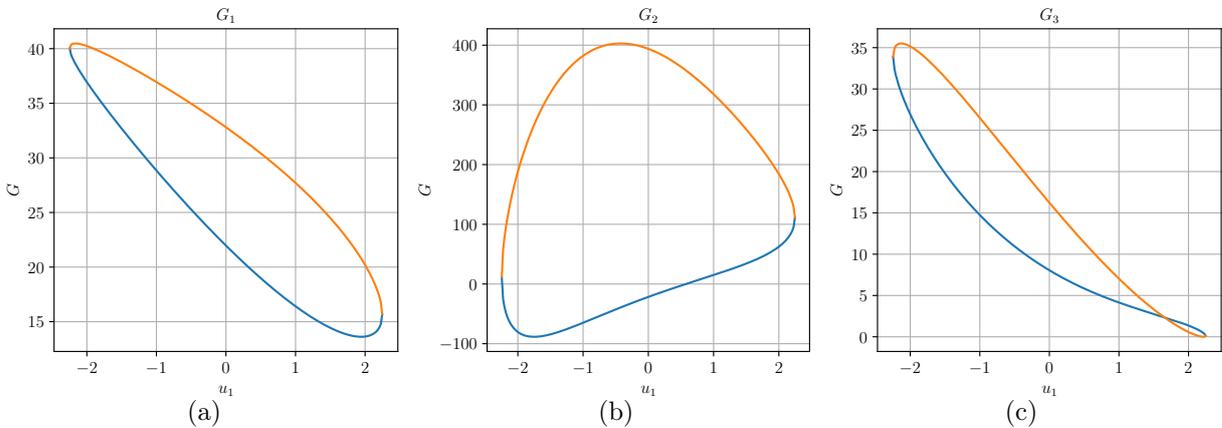
Legenda: A região sombreada denota a área viável, enquanto as curvas g_1 , g_2 e g_3 representam as funções de estado limite. A circunferência está centrada em $C = (-8, -4)$ com raio $\beta_t = 2,2464$, que representa o índice de confiabilidade alvo.

Fonte: O autor, 2024.

As restrições g_1 e g_3 são atendidas, conforme mostrado nas Figuras 11a e 11c onde as mesmas são positivas. Para g_3 , o ponto mais provável de falha é $x^* = (-5,7939; -4,4236)$, onde tem-se $g_3(x^*) = 0$.

Para identificar o ponto mais provável de falha usando a abordagem PMA, Libotte et al. (2020) propuseram um algoritmo de passo adaptativo de segunda ordem (ASOSL, de *Adaptive Second Order Step Length*), baseado na busca linear, que ajusta adaptativamente o comprimento do passo em cada iteração, realizando projeções sucessivas na hipersfera a cada iteração. Contudo, o gradiente calculado em cada iteração pode não se aproximar do vetor nulo conforme o ponto mínimo da função G na hipersfera é atingido.

Figura 11 - Gráficos das funções de desempenho restritas à circunferência $u_1^2 + u_2^2 = \beta_t^2$, onde $\beta_t = 2,2464$, no espaço normal padrão. A curva em laranja representa a respectiva função restrita ao hemisfério sul da circunferência e a em azul ao hemisfério norte.



Fonte: O autor, 2024.

Uma maneira de alcançar este objetivo é realizar cálculos num espaço equivalente aos hemisférios da hipersfera, localizar o correspondente ao MPP e, após sua localização, retornar à hipersfera. A projeção estereográfica pode ser utilizada para este fim e será analisada na Seção 3.

1.6 Otimização Baseada em Robustez e Confiabilidade

Conforme discutido na seção 1.5, a probabilidade de falha é definida por $P_f = \Phi(-\beta_t)$, onde Φ representa a função de distribuição acumulada da normal padrão, e β_t denota o índice de confiabilidade alvo. Este índice corresponde à distância entre o ponto de projeto e o ponto mais provável de falha; assim, quanto maior for o valor de β_t , menor será a probabilidade de falha. Adicionalmente, na seção 1.4, o conceito de robustez foi descrito por meio da abordagem da função de média efetiva e da diferença normalizada, sendo que, para o uso da função de média efetiva, incorpora-se um ruído δ associado a cada coordenada do vetor de projeto.

Em um contexto de otimização que integra robustez e confiabilidade, o objetivo é identificar soluções ótimas que apresentem mínima sensibilidade às perturbações externas, enquanto preservam níveis de confiabilidade pré-estabelecidos. Contudo, conforme Libotte (2020), frequentemente a otimização robusta e a maximização do índice de confiabilidade constituem metas conflitantes, possibilitando, assim, a formulação de um problema de otimização multiobjetivo.

Definição 5. *Seja $\mathbf{f}(\mathbf{d}, \mathbf{X}, \delta) = (f_1(\mathbf{d}, \mathbf{X}, \delta), \dots, f_m(\mathbf{d}, \mathbf{X}, \delta))$, onde $\mathbf{d} \in \mathbb{R}^{n_d}$ é o vetor de projeto com variáveis determinísticas, $\mathbf{X} \in \mathbb{R}^{n_s}$ é o vetor de variáveis aleatórias, e $\delta \in \mathbb{R}^{n_s}$ é o vetor de ruídos associados às variáveis aleatórias. Então, o problema de*

otimização multiobjetivo baseado em robustez e confiabilidade é definido como:

$$\begin{aligned}
& \text{Otimizar } (\mathbf{f}(\mathbf{d}, \mathbf{X}, \boldsymbol{\delta}), \beta_t) \\
& \text{Sujeito a } P_f = \mathbb{P} [g_i(\mathbf{d}, \mathbf{X}, \boldsymbol{\delta}) \leq 0] \leq \Phi(-\beta_t), i = 1, \dots, p, \\
& h_j(\mathbf{d}, \mathbf{X}) = 0, j = 1, \dots, q, \\
& \mathbf{d}^{inf} \leq \mathbf{d} \leq \mathbf{d}^{sup}, \\
& \beta_t^{inf} \leq \beta_t \leq \beta_t^{sup},
\end{aligned} \tag{34}$$

onde p e q representam, respectivamente, o número de restrições de desigualdade e de igualdade.

Nestas circunstâncias, utilizando a função de média efetiva para a avaliação da robustez, o objetivo é maximizar o índice de confiabilidade β_t , minimizar $f_1^{\text{eff}}, f_2^{\text{eff}}, \dots, f_{m_1}^{\text{eff}}$ e maximizar $f_{m_1+1}^{\text{eff}}, f_{m_1+2}^{\text{eff}}, \dots, f_{m_1+m_2}^{\text{eff}}$, onde m_1 e m_2 são números inteiros positivos, com $m = m_1 + m_2$.

Para tanto, considere uma população de v indivíduos, em uma meta-heurística populacional, que estão organizados em uma matriz $\mathbf{D} \in \mathbb{R}^{v \times (n+1)}$. Nesta matriz, cada linha representa um vetor de projeto \mathbf{d} , ao qual é acrescida a variável de confiabilidade β_t na última coluna. Assim, o indivíduo da linha i é dado por $\mathbf{D}_i = [\mathbf{d}_{i1}, \mathbf{d}_{i2}, \dots, \mathbf{d}_{in}, \beta_{ti}]$, com $i = 1, 2, \dots, v$.

Além disso, tem-se um vetor de ruídos $\boldsymbol{\delta} = [\delta_1, \delta_2, \dots, \delta_n, 0]$, onde $\delta_j \in \mathbb{R}_+$, com $j = 1, \dots, n$. Este vetor representa o nível de ruído associado à variável de projeto correspondente, onde a última coordenada de $\boldsymbol{\delta}$ é zero, pois o ruído associado ao índice de confiabilidade é nulo.

Desta forma, pode-se definir $\mathcal{R} \in \mathbb{R}^{M \times (n+1)}$ como uma matriz com M pontos gerados aleatoriamente na vizinhança de um indivíduo \mathbf{d} da população. A última coluna de \mathcal{R} é composta apenas por zeros. Estes M pontos são gerados de forma que $\mathbf{D}_{ij}(1 - \delta_j) \leq \mathcal{R}_{sj} \leq \mathbf{D}_{ij}(1 + \delta_j)$, onde $j = 1, 2, \dots, n$ e $s = 1, 2, \dots, M$, seguindo alguma distribuição de probabilidade simétrica. Daí, a função de média efetiva em torno de um indivíduo \mathbf{d} é dada por

$$f_r^{\text{eff}}(\mathbf{d}, \mathbf{X}, \boldsymbol{\delta}) = \frac{1}{M} \sum_{j=1}^M f_r(\mathcal{R}_j, \mathbf{X}, \boldsymbol{\delta}), \tag{35}$$

onde $r = 1, 2, \dots, m$ e \mathcal{R}_j devem ser entendidos como a j -ésima linha da matriz \mathcal{R} .

Além do cálculo da função de média efetiva no ponto de projeto \mathbf{d} , tomando o valor médio da função objetivo em M amostras na vizinhança de um indivíduo, é necessário verificar se estes mesmos M pontos atendem às restrições de confiabilidade. Para tanto, ao adotar a análise de confiabilidade inversa em cada uma das restrições probabilísticas, é necessário calcular o valor g_i^* , $i = 1, \dots, p$, que é o menor valor da função de desempenho

g_i dentre todos os M prováveis pontos de falha encontrados. Se esta condição for satisfeita, ou seja, $g_i^* > 0$, o ponto de projeto \mathbf{d} é considerado viável.

Com isso, pode-se definir a nova função objetivo \mathcal{F}_r que considera as restrições probabilísticas, penalizando a função de média efetiva nos pontos onde a restrição não é atendida. Ou seja,

$$\mathcal{F}_r(\mathbf{d}, \mathbf{X}, \boldsymbol{\delta}) = F_r^{\text{eff}}(\mathbf{d}, \mathbf{X}, \boldsymbol{\delta}) + \Psi \sum_{i=1}^p \max(-g_i^*, 0), \quad (36)$$

onde Ψ é o coeficiente de penalidade. Geralmente, este coeficiente é adotado como um valor suficientemente grande para os casos em que \mathcal{F}_r é minimizado. Quando se trata de um problema de maximização, o coeficiente de penalidade é tomado como $-\Psi$.

2 MÉTODO DE PASSO ADAPTATIVO DE SEGUNDA ORDEM COM PROJEÇÃO ESTEREOGRÁFICA PARA ANÁLISE DE CONFIABILIDADE INVERSA

A Abordagem PMA, vista no capítulo anterior, busca o ponto mais provável de falha u^* que minimiza a função de desempenho G na hipersfera $\mathbb{S}_U \subset U$ de raio β_t , onde U é o espaço normal padrão de dimensão n . No entanto, durante a execução da busca linear com passo adaptativo com a metodologia descrita por meio dos Algoritmos 1 e 2, a soma $u^{k-1} - \alpha^{k-1}p_{k-1}$ pode não pertencer à hipersfera.

Para solucionar esse problema, uma alternativa utilizada por Libotte et al. (2020) foi projetar este novo ponto em \mathbb{S}_U utilizando $u_{\mathbb{S}}^k = \beta_t \frac{u_{\mathbb{S}}^{k-1} - \alpha^{k-1}p_{k-1}}{\|u_{\mathbb{S}}^{k-1} - \alpha^{k-1}p_{k-1}\|}$. Desta forma, garante-se que o ponto $u_{\mathbb{S}}^k$ pertence à hipersfera, permitindo que a busca se mantenha dentro da região de interesse a fim de obter ponto de mínimo da função de desempenho G em \mathbb{S}_U .

A abordagem proposta por Libotte et al. (2020), embora eficaz, não explora o fato de o gradiente da função de desempenho tender a zero à medida que se aproxima do ponto mínimo.

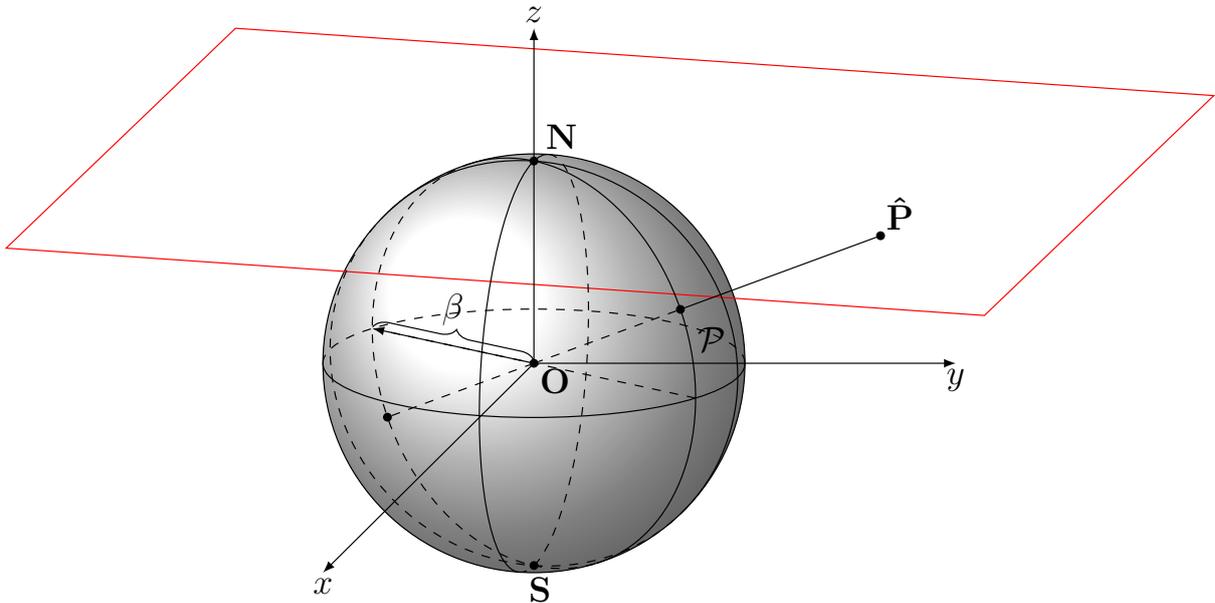
Visando aprimorar esta metodologia e eliminar a necessidade de projeções sucessivas, este trabalho propõe a utilização da projeção estereográfica para a obtenção do ponto mais provável de falha. Esta estratégia baseia-se no homeomorfismo entre os hemisférios de \mathbb{S}^n e o espaço \mathbb{R}^n , o que permite buscar o mínimo da função G em \mathbb{S}^n diretamente em \mathbb{R}^n . Os detalhes sobre o funcionamento da projeção estereográfica e sua aplicação estão descritos nas seções 2.1 e 2.2 subsequentes.

2.1 Projeção Estereográfica

Considere o espaço \mathbb{R}^n como um hiperplano de dimensão n , que está imerso em um espaço de dimensão superior, especificamente \mathbb{R}^{n+1} . Este hiperplano pode ser representado pelo conjunto $\zeta = \{\mathbf{x} \in \mathbb{R}^{n+1} | x_{n+1} = 1\}$. Suponha ainda que este hiperplano é tangente à esfera \mathbb{S}^n no polo norte. O conjunto $H^+ = \{\mathbf{x} \in \mathbb{S}^n | x_{n+1} > 0\}$ delimita o hemisfério norte da esfera e, de maneira análoga, o polo sul pode ser representado por $H^- = \{\mathbf{x} \in \mathbb{S}^n | x_{n+1} < 0\}$ e o equador $\mathbb{S}^{n-1} = \{\mathbf{x} \in \mathbb{S}^n | x_{n+1} = 0\}$. Na Figura 12 é possível observar esta representação para o caso $n = 2$, quando se toma $\beta = 1$ como raio da esfera.

Na Figura 12, tem-se um ponto, denominado $\hat{\mathbf{P}}$, que é projetado para um outro ponto, \mathcal{P} , localizado no hemisfério norte da esfera \mathbb{S}^2 . Essa projeção ocorre através da interseção de uma reta, que passa pelo ponto $\hat{\mathbf{P}}$ e pelo centro da esfera. Ao estabelecer essa

Figura 12 - Projeção Estereográfica.



Fonte: Adaptado de Trzeciak (2008).

correspondência, se está efetivamente criando uma bijeção, ou seja, uma relação de um para um, entre o espaço \mathbb{R}^2 e o hemisfério norte da esfera \mathbb{S}^2 . Esse processo é conhecido como *Projeção Estereográfica*.

Segundo Braun (2007), este procedimento pode ser generalizado para o espaço \mathbb{R}^n , onde a projeção estereográfica ($\beta = 1$), é dada por

$$\pi^+ : \mathbb{R}^n \longrightarrow H^+$$

$$x \longmapsto \pi^+(x) = (f_1(x), f_2(x), \dots, f_n(x), f_{n+1}(x)) = \frac{1}{\Delta(x)}(x_1, x_2, \dots, x_n, 1) \quad (37)$$

onde $\Delta(x) = (1 + \sum_{i=1}^n x_i^2)^{\frac{1}{2}}$.

A aplicação π^+ é de classe C^∞ , pois suas funções coordenadas o são. Além disso, sua inversa é dada por

$$(\pi^+)^{-1}(w_1, w_2, \dots, w_n, w_{n+1}) = \left(\frac{w_1}{w_{n+1}}, \frac{w_2}{w_{n+1}}, \dots, \frac{w_{n-1}}{w_{n+1}}, \frac{w_n}{w_{n+1}} \right). \quad (38)$$

Analogamente, é possível definir a aplicação

$$\pi^- : \mathbb{R}^n \longrightarrow H^-$$

$$x \longmapsto \pi^-(x) = -\frac{1}{\Delta(x)}(x_1, x_2, \dots, x_n, 1), \quad (39)$$

a qual possui inversa semelhante a acima exposta, que mapeia todo o espaço \mathbb{R}^n no hemisfério sul da esfera \mathbb{S}^n .

Ao projetar tanto no hemisfério norte quanto no sul, se está, na verdade, excluindo o equador da hiperesfera. Mas, quando $\hat{\mathbf{P}} \rightarrow \infty$ tem-se que o ponto $\mathcal{P} \in \mathbb{S}^n$ converge para o equador \mathbb{S}^{n-1} , onde a última coordenada é zero. Portanto, os pontos que estão localizados no equador da hiperesfera são identificados como pontos no infinito em \mathbb{R}^n .

Uma vez que \mathbb{S}^n é um conjunto compacto, toda função contínua definida em \mathbb{S}^n possui máximo e mínimo em \mathbb{S}^n . Portanto, tomando $g : \mathbb{S}^n \rightarrow \mathbb{R}$ como uma função diferenciável até segunda ordem e, portanto, contínua, existem pontos de máximo e mínimo de g em \mathbb{S}^n .

Mas a função g pode ser estendida para todo o espaço \mathbb{R}^n por meio da aplicação da projeção estereográfica, isto é, $g^+(x) = g(\pi^+(x))$ e $g^-(x) = g(\pi^-(x))$, que serão diferenciáveis até segunda ordem em \mathbb{R}^n , uma vez que a composta de funções diferenciáveis é diferenciável. Daí, três possibilidades devem ser consideradas: o ponto de máximo (ou mínimo) de g está localizado no hemisfério norte, no hemisfério sul ou no equador da hiperesfera.

Se o ponto de máximo (ou mínimo) \mathcal{P}^* está no hemisfério norte ou sul, pode-se usar a busca linear para encontrar o ponto de máximo (ou mínimo) $\hat{\mathbf{P}}^* \in \mathbb{R}^n$ que minimiza (ou maximiza) a função g^+ (ou g^-).

Por outro lado, se o ponto \mathcal{P}^* está localizado no equador \mathbb{S}^{n-1} , a busca linear possivelmente fará com que $\hat{\mathbf{P}} \rightarrow \infty$, permitindo obter um ponto tão próximo quanto se queira do ponto de mínimo.

Esta técnica pode ser utilizada para a análise de confiabilidade inversa, onde a função de desempenho $g|_{\mathbb{S}_U}$, com $\mathbb{S}_U \subset \mathbb{R}^n$, é estendida para todo o espaço \mathbb{R}^{n-1} e, a partir disso, é possível encontrar o ponto que a minimiza, que é o ponto mais provável de falha. Neste caso, as aplicações π^+ e π^- devem projetar o ponto $\hat{\mathbf{P}}$ em uma esfera de raio β_t , onde β_t é o índice de confiabilidade alvo.

Portanto, ao considerar uma hiperesfera com raio β_t , as fórmulas relativas às projeções π^+ e π^- podem ser reformuladas como

$$\begin{aligned} \pi^+ : \mathbb{R}^{n-1} &\longrightarrow H^+ \\ x &\longmapsto \pi^+(x) = \beta_t \frac{1}{\Delta_{\beta_t}(x)}(x_1, x_2, \dots, x_{n-1}, \beta_t) \end{aligned} \quad (40)$$

e

$$\begin{aligned} \pi^- : \mathbb{R}^{n-1} &\longrightarrow H^- \\ x &\longmapsto \pi^-(x) = -\beta_t \frac{1}{\Delta_{\beta_t}(x)}(x_1, x_2, \dots, x_{n-1}, \beta_t), \end{aligned} \quad (41)$$

onde $\Delta_{\beta_t}(x) = (\beta_t^2 + \sum_{i=1}^n x_i^2)^{\frac{1}{2}}$.

2.2 Algoritmo do Método de Passo Adaptativo de Segunda Ordem com Projeção Estereográfica

Em seu estudo, Libotte et al. (2020) propuseram a utilização do Algoritmo 2 para determinar o ponto mais provável de falha utilizando a abordagem PMA, caracterizando o método de comprimento de passo adaptativo de segunda ordem (ASOSL, de *Adaptive Second Order Step Length*). Nesta abordagem, conforme foi discutido na seção 1.5, busca-se o ponto mais provável de falha u^* , minimizando a função de desempenho G na hipersfera $\mathbb{S}_U \subset U$ de raio β_t , onde U é o espaço normal padrão de dimensão n .

Para isso, o ASOSL leva em consideração que a soma $u^{k-1} - \alpha^{k-1}p_{k-1}$ pode não pertencer à hipersfera, projetando este novo ponto em \mathbb{S}_U por meio de

$$u_{\mathbb{S}}^k = \beta_t \frac{u_{\mathbb{S}}^{k-1} - \alpha^{k-1}p_{k-1}}{\|u_{\mathbb{S}}^{k-1} - \alpha^{k-1}p_{k-1}\|} \quad (42)$$

a cada iteração. Com a adoção desta abordagem, é possível que $p_{k-1} \not\rightarrow 0$ quando o ponto de mínimo é alcançado na hipersfera, dado que os pontos de mínimo de G no espaço U não estão necessariamente situados na hipersfera.

Para superar essa limitação, está sendo proposta a integração da projeção estereográfica com o ASOSL, realizando a composição da função de desempenho G no espaço normal padrão com as funções de projeção descritas nas equações (40) e (41), permitindo a busca em um espaço de mesma dimensão da hipersfera.

Para tanto, considere uma função de desempenho $G : \mathbb{R}^n \rightarrow \mathbb{R}$, com as variáveis padronizadas e uma hipersfera em \mathbb{R}^n centrada na origem e de raio β_t . Considere também o hiperplano $\zeta = \{\mathbf{u} \in \mathbb{R}^n | u_n = \beta_t\}$ e as funções de projeção estereográfica $\pi^+(u) = \beta_t \frac{1}{\Delta_{\beta_t}(u)}(u_1, u_2, \dots, u_{n-1}, \beta_t)$ e $\pi^-(u) = -\beta_t \frac{1}{\Delta_{\beta_t}(u)}(u_1, u_2, \dots, u_{n-1}, \beta_t)$. Desta forma, pode-se construir as funções $G \circ \pi^+ : \zeta \rightarrow \mathbb{R}$ que, por simplicidade, será aqui denotada por G^+ e $G \circ \pi^- : \zeta \rightarrow \mathbb{R}$, por G^- . O funcionamento do ASOSL integrado à projeção estereográfica pode ser visualizado pelo algoritmo 5.

No fluxograma da Figura 13, é possível observar o funcionamento do ASOSL utilizando a projeção estereográfica. Aqui, a principal diferença em relação ao ASOSL proposto por Libotte et al. (2020) está na ausência da necessidade de projeção sobre a hipersfera a cada novo ponto obtido $u^k = u^{k-1} - \alpha^{k-1}p_{k-1}$ e a garantia de que o gradiente se aproxima do vetor nulo quando o ponto de mínimo é atingido.

Neste mesmo fluxograma é importante compreender que o termo G^\pm indica que os cálculos correspondentes devem ser aplicados tanto para a projeção estereográfica do hemisfério norte quanto para a do sul. Esta necessidade decorre do fato de que o ponto mais provável de falha pode estar localizado em qualquer um dos hemisférios. Portanto, o procedimento é executado duas vezes, sendo uma para cada projeção: a primeira para o hemisfério norte e a segunda para o hemisfério sul. Ao final, o ponto mais provável de

Algoritmo 5 - Algoritmo de otimização de passo adaptativo de segunda ordem com projeção estereográfica.

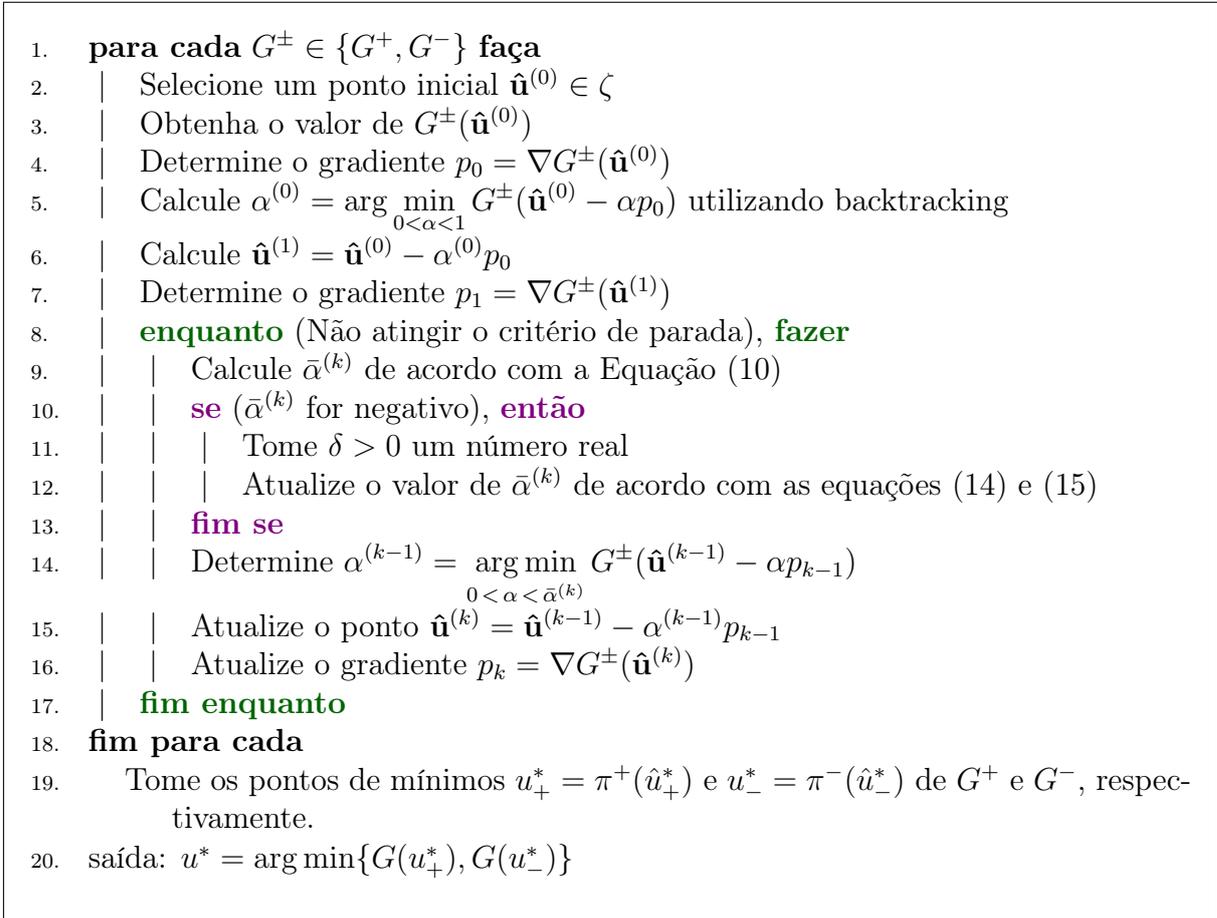
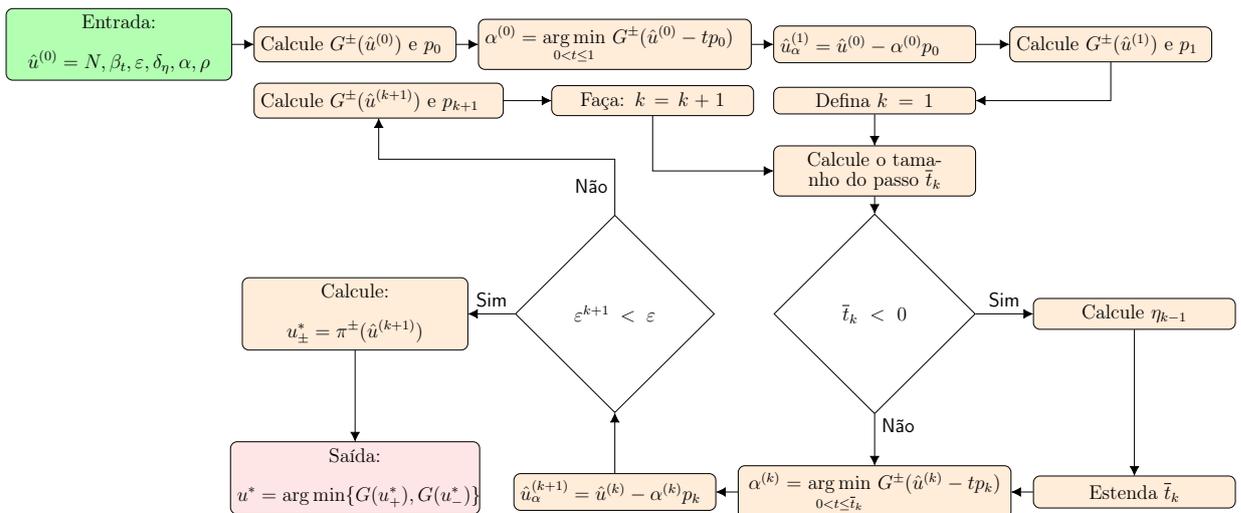


Figura 13 - Fluxograma do método do tamanho de passo adaptativo com projeção estereográfica.



Fonte: O autor, 2024.

falha é aquele que minimiza a função de desempenho G dentre os dois obtidos.

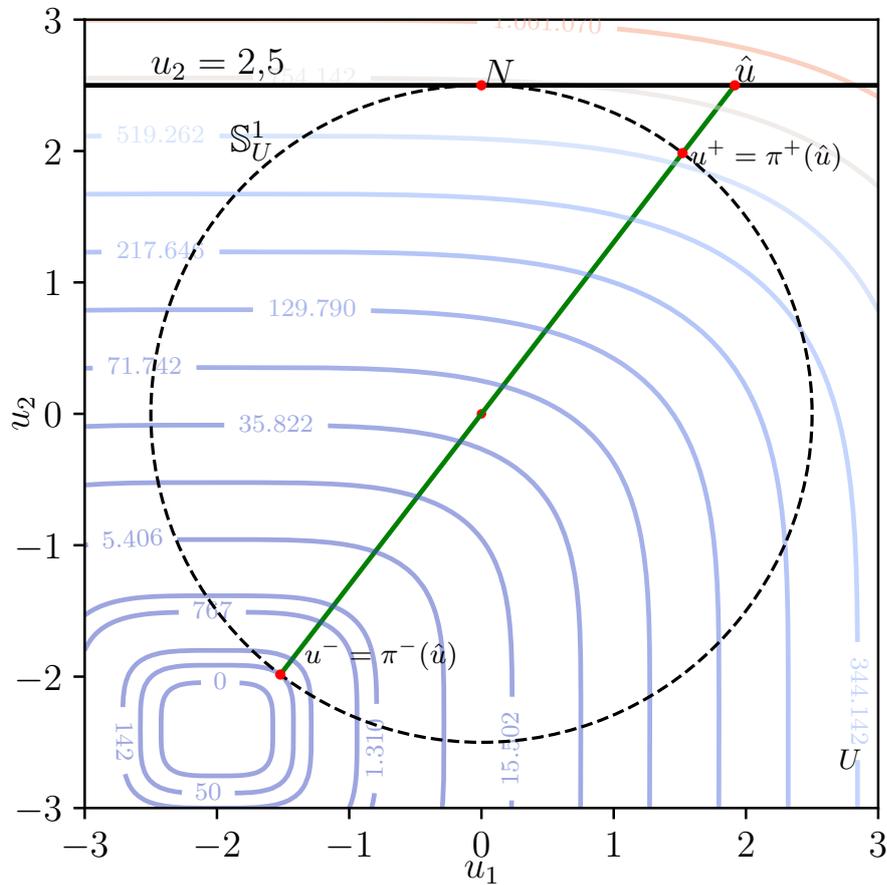
Exemplo 4. Como ilustração da projeção estereográfica em conjunto com a transformação de Rosemblatt, considere a função de desempenho $g : \mathbb{R}^2 \rightarrow \mathbb{R}$ dada por:

$$g(X_1, X_2) = X_1^4 + 2X_2^4 - 20, \quad (43)$$

com $X_1 \sim \mathcal{N}(10, 5^2)$ e $X_2 \sim \mathcal{N}(12, 5^2)$.

Considere também que o índice de confiabilidade alvo é $\beta_t = 2,5$. Aplicando a transformação de Rosemblatt, tomando a circunferência de raio $\beta_t = 2,5$ centrada na origem do espaço normal padrão U , e as curvas de nível de G , obtém-se a Figura 14.

Figura 14 - Projeção estereográfica no espaço normal padrão.



Legenda: As curvas em diversos tons de azul representam as curvas de nível da função de desempenho G no espaço normal padrão U .

Fonte: O autor, 2024.

Para localizar o ponto mais provável de falha, é necessário minimizar a função de desempenho G na circunferência de raio $\beta_t = 2,5$. Como cada ponto nos hemisférios da circunferência \mathbb{S}_U^1 corresponde a um ponto na reta $u_2 = 2,5$ no espaço normal padrão U ,

a minimização de G na circunferência \mathbb{S}_U^1 é equivalente a minimizar as funções $G \circ \pi^+$ e $G \circ \pi^-$, na reta $u_2 = 2,5$. Após encontrar o ponto que minimiza G^\pm , o ponto mais provável de falha sobre \mathbb{S}_U^1 pode ser obtido por $\pi^\pm(\hat{u}^*) = u^*$. Especificamente, tem-se que $\hat{u}^* = 1,9159$ no espaço equivalente à hiperesfera e $u^* = (-1,5207, -1,9843)$, onde $G(u^*) = 5,0310 \times 10^1$.

3 METAMODELAGEM

O processo de busca por soluções robustas, utilizando meta-heurísticas populacionais, requer a avaliação da função objetivo e das restrições em uma ampla gama de pontos. Dependendo da complexidade das funções envolvidas, isso pode ser computacionalmente desafiador e, para atenuar este custo, é possível empregar metamodelos. Estes são modelos matemáticos que podem ser usados para aproximar a função objetivo ou as restrições, diminuindo assim o número de avaliações necessárias durante o processo de otimização.

Existem várias técnicas de metamodelagem, mas neste trabalho é utilizada a Regressão com Processos Gaussianos, devido à sua flexibilidade em lidar com funções complexas e à possibilidade de se computar as incertezas associadas ao processo de predição.

Em certos contextos, a técnica de regressão que utiliza processos gaussianos é conhecida como Kriging. Isso é particularmente comum no campo da geoestatística, onde é amplamente aplicada para a interpolação espacial no setor de mineração. O termo Kriging deve seu nome ao geólogo sul-africano Danie Krige, que teve uma influência significativa nesta área (Wackernagel, 2003).

Dentro do contexto do aprendizado de máquina e estatística, a terminologia preferida tende a ser GPR, ou *Gaussian Process Regression*. Uma outra importante distinção reside nas propriedades das funções que são usadas para gerar as matrizes de covariância. No Kriging, essas funções são conhecidas como variogramas e devem atender a critérios específicos para serem aplicáveis. Em contraste, na GPR há uma maior flexibilidade através do uso de funções *kernel*. É importante observar, além disso, que o Kriging abrange várias modalidades, sendo as principais:

1. Kriging ordinário, que parte do pressuposto de uma média de dados desconhecida, mas considerada constante;
2. Kriging simples, que opera sob a suposição de que a média dos dados é tanto conhecida quanto constante.

Para uma explicação mais detalhada sobre Kriging, consulte Wackernagel (2003).

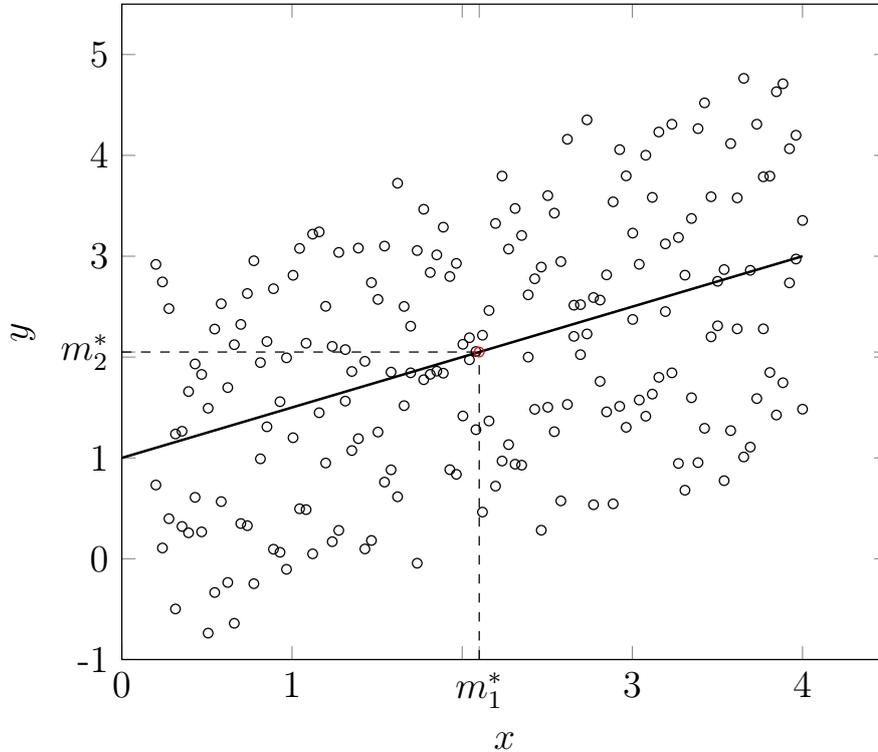
Nas próximas seções serão abordados os fundamentos da regressão com processos gaussianos e também alguns breves exemplos serão expostos, a fim de exibir como a metodologia funciona. Uma abordagem aprofundada sobre o tema pode ser consultada em Rasmussen e Williams (2006).

3.1 Regressão Linear com Processos Gaussianos

Considere um conjunto de pontos $\mathcal{D} = \{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$, com $(x_i, y_i) \in \mathbb{R}^2$, conforme exibido na Figura 15, cujas coordenadas dos pontos médios são $m_1^* =$

$\frac{1}{n} \sum_{i=1}^n x_i$, $m_2^* = \frac{1}{n} \sum_{i=1}^n y_i$, a covariância $\sigma_{xy} = \frac{1}{n} \sum_{i=1}^n (x_i - m_1^*)(y_i - m_2^*)$, e variâncias $\sigma_x^2 = \frac{1}{n} \sum_{i=1}^n (x_i - m_1^*)^2$, $\sigma_y^2 = \frac{1}{n} \sum_{i=1}^n (y_i - m_2^*)^2$.

Figura 15 - Nuvem de dispersão de pontos.



Fonte: Adaptado de Wackernagel (2003).

Uma função que poderia ser utilizada para aproximar os pontos da Figura 15 é a função linear $f(x) = ax + b$. Neste caso, deve-se encontrar os valores apropriados de a e b , que melhor se aproximam dos dados. Estes valores são os que minimizam a soma dos quadrados dos resíduos, isto é, que minimizam a função $F(a, b)$, dada por:

$$\begin{aligned}
 F(a, b) &= \frac{1}{n} \sum_{i=1}^n (y_i - f(x_i))^2 \\
 &= \frac{1}{n} \sum_{i=1}^n (y_i - ax_i - b)^2 \\
 &= \frac{1}{n} \sum_{i=1}^n (a^2 x_i^2 + 2abx_i - 2ax_i y_i + b^2 - 2by_i + y_i^2).
 \end{aligned} \tag{44}$$

A fim de se obter tais valores de a e b , pode-se transladar o sistema de coordenadas para o ponto (m_1^*, m_2^*) e substituir as expressões das variâncias e covariância. Desta forma, a expressão da reta passa a ser $f(x) = ax - \underbrace{am_1 + m_2 + b}_{b'}$ e a função $F(a, b)$:

$$\begin{aligned}
F(a, b') &= \frac{1}{n} \sum_{i=1}^n \left(a^2(x_i - m_1^*)^2 + 2ab'(x_i - m_1^*) - 2a(x_i - m_1^*)(y_i - m_2^*) \right. \\
&\quad \left. + b'^2 - 2b'(y_i - m_2^*) + (y_i - m_2^*)^2 \right) \\
&= a^2\sigma_x^2 - 2a\sigma_{xy} + b'^2 + \sigma_y^2.
\end{aligned} \tag{45}$$

Assim, calculando as derivadas parciais de $F(a, b')$ em relação a a e b' , igualando a zero, e retornando ao sistema de coordenadas original, obtém-se o ponto de mínimo da função $F(a, b)$, que é dado por:

$$a = \frac{\sigma_{xy}}{\sigma_x^2}, \tag{46}$$

$$b = m_2^* - am_1^*, \tag{47}$$

onde a é o coeficiente angular da reta e b é o coeficiente linear. Isso caracteriza o que é conhecido como regressão linear.

Uma outra forma de escrever a expressão para f é

$$f(x) = \mathbf{x}^\top \mathbf{w}, \tag{48}$$

onde $\mathbf{w} = [a, b]^\top$ e $\mathbf{x} = [x, 1]^\top$. Neste caso, a função $F(a, b)$ passa a ser escrita como $F(\mathbf{w}) = \frac{1}{n} \sum_{i=1}^n (y_i - \mathbf{x}_i^\top \mathbf{w})^2$, onde $\mathbf{x}_i = [x_i, 1]^\top$. Assim, ao calcular as derivadas parciais de F em relação a a e a b , e igualá-las a zero, obtém-se o vetor \mathbf{w} que minimiza F , que é dado por $\mathbf{w} = (\mathbf{X}\mathbf{X}^\top)^{-1}\mathbf{X}\mathbf{y}$, onde \mathbf{X} é a matriz de entrada ($2 \times n$) e \mathbf{y} é o vetor de saída ($n \times 1$).

A regressão linear, sob a perspectiva da inferência Bayesiana, é interpretada como um caso especial de um Processo Gaussiano, o qual é um modelo probabilístico que assume que qualquer conjunto finito de variáveis aleatórias de uma coleção segue uma distribuição Gaussiana conjunta. Isso permite que, ao invés de simplesmente ajustar uma linha reta aos dados, considerar uma distribuição sobre todas as possíveis retas que poderiam se ajustar aos dados (Rasmussen; Williams, 2006).

Considere agora que a saída y possui um ruído gaussiano com média zero e variância σ_n^2 , ou seja, $y = f(x) + \epsilon$, onde $\epsilon \sim \mathcal{N}(0, \sigma_n^2)$. A suposição de ruído em conjunto com o modelo $f(x) = \mathbf{x}^\top \mathbf{w}$ leva diretamente à verossimilhança, que é a densidade de probabilidade das observações, considerando os parâmetros. Assim

$$\begin{aligned}
p(y|X, w) &= \prod_{i=1}^n p(y_i|x_i, w) \\
&= \prod_{i=1}^n \frac{1}{\sqrt{2\pi\sigma_n^2}} \exp\left(-\frac{1}{2\sigma_n^2}(y_i - \mathbf{x}_i^\top \mathbf{w})^2\right) \\
&= \frac{1}{(2\pi\sigma_n^2)^{n/2}} \exp\left(-\frac{1}{2\sigma_n^2} \sum_{i=1}^n (y_i - \mathbf{x}_i^\top \mathbf{w})^2\right) \\
&= \frac{1}{(2\pi\sigma_n^2)^{n/2}} \exp\left(-\frac{1}{2\sigma_n^2} \|\mathbf{y} - \mathbf{X}^\top \mathbf{w}\|^2\right) \\
&= \mathcal{N}(\mathbf{X}^\top \mathbf{w}, \sigma_n^2 \mathbf{I}).
\end{aligned} \tag{49}$$

Dentro do formalismo Bayesiano, é necessário levar em conta a distribuição *a priori* dos parâmetros, que será aqui fixado por

$$p(\mathbf{w}) = \mathcal{N}(\mathbf{0}, \Sigma_p), \tag{50}$$

onde Σ_p é a matriz de covariância da distribuição a priori. A partir disso, a distribuição *a posteriori* dos parâmetros é expressa como

$$p(\mathbf{w}|\mathbf{X}, \mathbf{y}) = \frac{p(\mathbf{y}|\mathbf{X}, \mathbf{w})p(\mathbf{w})}{p(\mathbf{y}|\mathbf{X})}. \tag{51}$$

Mas,

$$\begin{aligned}
p(\mathbf{y}|\mathbf{X}, \mathbf{w})p(\mathbf{w}) &= \mathcal{N}(\mathbf{X}^\top \mathbf{w}, \sigma_n^2 \mathbf{I})\mathcal{N}(\mathbf{0}, \Sigma_p) \\
&= \frac{1}{(2\pi\sigma_n^2)^{n/2}} \exp\left(-\frac{1}{2\sigma_n^2} \|\mathbf{y} - \mathbf{X}^\top \mathbf{w}\|^2\right) \frac{1}{(2\pi)^{p/2} |\Sigma_p|^{1/2}} \exp\left(-\frac{1}{2} \mathbf{w}^\top \Sigma_p^{-1} \mathbf{w}\right) \\
&= \frac{1}{(2\pi\sigma_n^2)^{n/2} (2\pi)^{p/2} |\Sigma_p|^{1/2}} \exp\left(-\frac{1}{2\sigma_n^2} \|\mathbf{y} - \mathbf{X}^\top \mathbf{w}\|^2 - \frac{1}{2\sigma_n^2} \mathbf{w}^\top \Sigma_p^{-1} \mathbf{w}\right).
\end{aligned} \tag{52}$$

Tomando $A = \mathbf{X}\mathbf{X}^\top + \sigma_n^2 \Sigma_p^{-1}$, $Q = (2\pi\sigma_n^2)^{n/2} (2\pi)^{p/2} |\Sigma_p|^{1/2}$ e $\bar{\mathbf{w}} = A^{-1} \mathbf{X}\mathbf{y}$, tem-se:

$$\begin{aligned}
p(\mathbf{y}|\mathbf{X}, \mathbf{w})p(\mathbf{w}) &= \frac{1}{Q} \exp\left(-\frac{1}{2\sigma_n^2} (\mathbf{y} - \mathbf{X}^\top \mathbf{w})^\top (\mathbf{y} - \mathbf{X}^\top \mathbf{w}) - \frac{1}{2\sigma_n^2} \mathbf{w}^\top \Sigma_p^{-1} \mathbf{w}\right) \\
&= \frac{1}{Q} \exp\left(-\frac{1}{2\sigma_n^2} (\mathbf{y}^\top \mathbf{y} - \mathbf{w}^\top A \bar{\mathbf{w}} + (\mathbf{w} - \bar{\mathbf{w}})^\top A \mathbf{w})\right) \\
&= \frac{1}{Q} \exp\left(-\frac{1}{2\sigma_n^2} (\mathbf{y}^\top \mathbf{y} - \bar{\mathbf{w}}^\top A \bar{\mathbf{w}} + (\mathbf{w} - \bar{\mathbf{w}})^\top A (\mathbf{w} - \bar{\mathbf{w}}))\right) \\
&\propto \exp\left(-\frac{1}{2\sigma_n^2} (\mathbf{w} - \bar{\mathbf{w}})^\top A (\mathbf{w} - \bar{\mathbf{w}})\right).
\end{aligned} \tag{53}$$

Assim, a distribuição posterior pode ser vista como gaussiana com média $\bar{\mathbf{w}}$ e matriz de covariância $\sigma_n^2 \mathbf{A}^{-1}$, isto é,

$$p(\mathbf{w}|\mathbf{X}, \mathbf{y}) \sim \mathcal{N}(\bar{\mathbf{w}}, \sigma_n^2 \mathbf{A}^{-1}). \quad (54)$$

A partir da expressão para f dada em (48), a distribuição preditiva para um novo ponto x_* é dada por

$$\begin{aligned} p(y_*|x_*, \mathbf{X}, \mathbf{y}) &= \int p(y_*|x_*, \mathbf{w})p(\mathbf{w}|\mathbf{X}, \mathbf{y})d\mathbf{w} \\ &= \mathcal{N}(\mathbf{x}_*^\top \bar{\mathbf{w}}, \mathbf{x}_*^\top \sigma_n^2 \mathbf{A}^{-1} \mathbf{x}_*). \end{aligned} \quad (55)$$

onde $\bar{\mathbf{w}} = \mathbf{A}^{-1} \mathbf{X} \mathbf{y}$ e $\mathbf{A} = \mathbf{X} \mathbf{X}^\top + \sigma_n^2 \Sigma_p^{-1}$.

De acordo com Rasmussen e Williams (2006), embora a regressão linear seja um instrumento valioso, ela apresenta limitações naturais, especialmente quando se trata de modelar relações não lineares entre variáveis. Por exemplo, se a relação entre as variáveis for de natureza quadrática ou exponencial, a regressão linear não conseguirá representar adequadamente os dados. Nesses casos, torna-se imprescindível a utilização de um modelo mais flexível, que se adapte mais precisamente à natureza dos dados.

Uma alternativa, ainda de acordo com Rasmussen e Williams (2006), visando superar estas dificuldades, é primeiro projetar os pontos em um espaço de alta dimensão, usando um conjunto de funções base, para em seguida utilizar o modelo linear neste espaço. Por exemplo, considere a função de base $\phi(x) = [1, x, x^2, x^3, \dots, x^N]$, utilizada na regressão polinomial. Neste caso, a função poderia ser expressa como $f(x) = \phi(x)^\top \mathbf{w}$, onde \mathbf{w} é o vetor de pesos, e a função f passa a ser linear em relação a \mathbf{w} , mas não em relação a x .

Utilizando novamente a inferência Bayesiana, a distribuição preditiva para um novo ponto x_* pode ser dada por

$$f_*|x_*, \mathbf{X}, \mathbf{y} \sim \mathcal{N}(\phi(x_*)^\top \bar{\mathbf{w}}, \phi(x_*)^\top \sigma_n^2 \mathbf{A}^{-1} \phi(x_*)), \quad (56)$$

onde $\bar{\mathbf{w}} = \mathbf{A}^{-1} \Phi(\mathbf{X}) \mathbf{y}$ e $\mathbf{A} = \Phi(\mathbf{X}) \Phi(\mathbf{X})^\top + \sigma_n^2 \Sigma_p^{-1}$ com $\Phi(\mathbf{X})$ sendo a matriz de funções de base aplicadas aos n pontos de treinamento.

Contudo, \mathbf{A} é uma matriz de ordem $N \times N$, o que pode tornar o cálculo de sua inversa computacionalmente caro, caso o espaço de funções base tenha uma dimensão muito elevada.

Para contornar este possível problema, observe que

$$\begin{aligned}
A\Sigma_p\Phi &= \Phi\Phi^\top\Sigma_p\Phi + \sigma_n^2 I\Phi \\
&= \Phi(\Phi^\top\Sigma_p\Phi + \sigma_n^2 I) \\
&= \Phi(K + \sigma_n^2 I),
\end{aligned} \tag{57}$$

onde $\Phi = \Phi(X)_{N \times n}$ e $K = (\Phi^\top\Sigma_p\Phi)_{n \times n}$. Daí, multiplicando ambos os lados desta igualdade por \mathbf{A}^{-1} e por $(K + \sigma_n^2 I)^{-1}$, obtém-se

$$\Sigma_p\Phi(K + \sigma_n^2 I)^{-1} = A^{-1}\Phi. \tag{58}$$

Isto mostra que pode-se substituir $A^{-1}\Phi$ por $\Sigma_p\Phi(K + \sigma_n^2 I)^{-1}$, na expressão de $\bar{\mathbf{w}}$, e assim reduzir o custo para cálculo da inversa nos casos em que $n < N$.

Para determinar a expressão para a variância em função de \mathbf{K} , pode-se utilizar a fórmula de Woodbury (ver Rasmussen e Williams (2006)), que é dada por

$$(Z + UWV^\top)^{-1} = Z^{-1} - Z^{-1}U(W^{-1} + V^\top Z^{-1}U)^{-1}V^\top Z^{-1}, \tag{59}$$

e assim, obter que

$$\begin{aligned}
A^{-1} &= (\sigma_n^2 \Sigma_p^{-1} + \Phi\Phi^\top)^{-1} \\
&= \sigma_n^{-2} (\Sigma_p^{-1} + \Phi\sigma_n^{-2} \mathbf{I}\Phi^\top)^{-1} \\
&= \sigma_n^{-2} (\Sigma_p - \Sigma_p\Phi(\sigma_n^2 \mathbf{I} + \Phi^\top\Sigma_p\Phi)^{-1}\Phi^\top\Sigma_p) \\
&= \sigma_n^{-2} (\Sigma_p - \Sigma_p\Phi(K + \sigma_n^2 I)^{-1}\Phi^\top\Sigma_p).
\end{aligned} \tag{60}$$

Daí, a distribuição preditiva para um novo ponto x_* pode ser reescrita como

$$f_*|x_*, X, y \sim \mathcal{N}(\phi(x_*)^\top \bar{\mathbf{w}}, \phi(x_*)^\top \Sigma_p \phi(x_*) - \phi(x_*)^\top \Sigma_p \Phi (K + \sigma_n^2 I)^{-1} \Phi \Sigma_p \phi(x_*)), \tag{61}$$

onde $\bar{\mathbf{w}} = \Sigma_p \Phi (K + \sigma_n^2 I)^{-1} \mathbf{y}$.

Nesta seção, foi apresentada a regressão linear com Processos Gaussianos, onde a distribuição ocorria sobre os pesos. No entanto, a distribuição pode ser feita diretamente no espaço das funções, o que será discutido na próxima seção.

3.2 Regressão com Processos Gaussianos

Considere uma função $f : \mathcal{X} \rightarrow \mathcal{Y}$, que mapeia um espaço de entrada \mathcal{X} em um espaço de saída \mathcal{Y} . Suponha que esta função seja desconhecida—só se conhece alguns pares de entrada e saída, mas não a regra que os relaciona—ou é muito custosa de ser

avaliada em todos os pontos de \mathcal{X} , por isso só se dispõe de um conjunto limitado de dados $\mathcal{D} = \{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$, onde $x_i \in \mathcal{X}$ e $y_i \in \mathcal{Y}$. O objetivo é estimar f em um conjunto de pontos $\mathcal{X}^* = \{x_1^*, x_2^*, \dots, x_m^*\}$, onde $x_i^* \in \mathcal{X}$ usando Regressão com Processos Gaussianos.

Aqui, pode-se considerar duas situações: os dados de saída y são ruidosos ou não. No primeiro caso, a saída y é dada por $y = f(x) + \epsilon$, onde $\epsilon \sim \mathcal{N}(0, \sigma_n^2)$, e no segundo caso, $y = f(x)$ (Rasmussen; Williams, 2006).

Um Processo Gaussiano, denotado como $f(x) \sim \mathcal{GP}(m(x), k(x, x'))$, é integralmente determinado por sua média e covariância. A média e a covariância são respectivamente definidas como $m(x) = \mathbb{E}[f(x)]$ e $k(x, x') = \mathbb{E}[(f(x) - m(x))(f(x') - m(x'))]$, onde \mathbb{E} é o operador de esperança, k é a função *kernel* ou função de covariância, e m é a média da função.

A função de covariância estabelece a relação entre os pontos do conjunto de entrada, sendo expressa por:

$$\begin{aligned} \text{cov}(f(x), f(x')) &= \mathbb{E}[(f(x) - m(x))(f(x') - m(x')))] \\ &= k(x, x'). \end{aligned}$$

Note que a função de covariância estabelece a relação entre os valores de $f(x)$ e $f(x')$ com base na relação entre x e x' .

A função *kernel* é um componente crucial para o Processo Gaussiano e existem vários tipos, cada uma com suas características e propriedades. No entanto, todas elas têm em comum o fato de serem positivas definidas, ou seja, a matriz com as covariâncias gerada por elas é sempre simétrica positiva definida.

O fato da matriz de covariância \mathbf{K} , onde cada elemento K_{ij} é dado por $K_{ij} = k(x_i, x_j)$ ser simétrica e positiva definida, garante que ela possa ser decomposta pela matriz de Cholesky, isto é, $\mathbf{K} = \mathbf{L}\mathbf{L}^\top$, onde \mathbf{L} é uma matriz triangular inferior, permitindo assim amostrar valores de f que seguem uma distribuição normal multivariada com média zero e covariância \mathbf{K} .

De fato, seja $\mathbf{f}^* = \mathbf{m} + \mathbf{L}\mathbf{z}$, onde \mathbf{m} é o vetor de médias, \mathbf{L} é a matriz de Cholesky de \mathbf{K} e \mathbf{z} é um vetor de variáveis aleatórias com distribuição normal padrão. Assim, a covariância entre os pontos de f^* é dada por

$$\begin{aligned} \text{cov}(\mathbf{f}^*) &= \mathbb{E}[(\mathbf{f}^* - \mathbf{m})(\mathbf{f}^* - \mathbf{m})^\top] \\ &= \mathbb{E}[(\mathbf{L}\mathbf{z})(\mathbf{L}\mathbf{z})^\top] \\ &= \mathbb{E}[\mathbf{L}\mathbf{z}\mathbf{z}^\top\mathbf{L}^\top] \\ &= \mathbf{L}\mathbb{E}[\mathbf{z}\mathbf{z}^\top]\mathbf{L}^\top \\ &= \mathbf{L}\mathbf{I}\mathbf{L}^\top \\ &= \mathbf{K}. \end{aligned} \tag{62}$$

Observação 1. Na decomposição de Cholesky, a matriz de covariância K é, comumente, substituída por $K + \varepsilon I$, onde ε é um valor muito pequeno que não afeta significativamente o resultado da decomposição. Essa substituição é feita para garantir a estabilidade numérica durante o processo de decomposição.

Uma das opções mais comuns para função *kernel* é a função RBF (*Radial Basis Function*), que também é chamada de função gaussiana, e que é expressa como

$$\text{cov}(y_p, y_q) = k(x_p, x_q) = \exp\left(-\frac{1}{2l^2}\|x_p - x_q\|^2\right), \quad (63)$$

onde l representa o comprimento de escala. A função de covariância pode capturar variações nos dados em maior ou menor grau, dependendo do valor l que é denominado hiperparâmetro. Além disso, quanto mais próximos os pontos x_p e x_q estão, maior será o valor da função de covariância, e quanto mais distantes, menor será o valor.

Na seção 3.2.3 será discutida a importância de determinar o valor ótimo para os hiperparâmetros presentes na função de covariância.

3.2.1 Regressão com Dados Não Ruidosos

Suponha que os dados de saída y não são ruidosos, isto é, $y = f(x)$. Neste caso, a distribuição conjunta *a priori* entre os pontos de treinamento x e os pontos não observados x_* pode ser dada por

$$\begin{bmatrix} \mathbf{y} \\ \mathbf{f}^* \end{bmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} \mathbf{K} & \mathbf{K}_* \\ \mathbf{K}_*^\top & \mathbf{K}_{**} \end{bmatrix}\right), \quad (64)$$

onde $\mathbf{K}(\mathbf{X}, \mathbf{X}) \triangleq \mathbf{K}$ é a matriz de covariância entre os pontos de treinamento, $\mathbf{K}(\mathbf{X}, \mathbf{X}_*) \triangleq \mathbf{K}_*$ é a matriz de covariância entre os pontos de treinamento e os pontos não observados, $\mathbf{K}(\mathbf{X}_*, \mathbf{X}_*) \triangleq \mathbf{K}_{**}$ é a matriz de covariância entre os pontos não observados e \mathbf{y} é o vetor de saída dos dados observados.

Com a distribuição *a priori*, é possível obter infinitas funções que satisfazem a matriz de covariância nos pontos não observados. Uma alternativa, não muito sábia em termos de eficiência computacional, é escolher, dentre todas as funções obtidas, aquelas que respeitam os dados observados.

Por outro lado, condicionando os dados não observados aos dados observados (von Mises, 1964, p. 427, Seção 9.3), é possível obter as funções que melhor se ajustam aos dados observados. Daí, a distribuição *a posteriori* é dada por

$$p(\mathbf{f}^* | \mathbf{X}, \mathbf{y}, \mathbf{X}_*) = \mathcal{N}(\mathbf{K}_* \mathbf{K}^{-1} \mathbf{y}, \mathbf{K}_{**} - \mathbf{K}_* \mathbf{K}^{-1} \mathbf{K}_*). \quad (65)$$

Explicitamente, a predição em um ponto x_* e sua covariância podem ser dados, respectivamente, por

$$m(x_*) = \mathbf{K}_* \mathbf{K}^{-1} \mathbf{y} \quad (66)$$

$$k(x_*) = \mathbf{K}_{**} - \mathbf{K}_* \mathbf{K}^{-1} \mathbf{K}_*. \quad (67)$$

No exemplo abaixo, é apresentado um exemplo que ilustra o processo de predição utilizando a distribuição priori e *a posteriori* para dados não ruidosos.

Exemplo 5. *Considere um conjunto com 5 pontos no intervalo $[-4, 4]$, a saber, $X^* = \{-4, -3, -1, 0, 2\}$ e a correspondente matriz de covariância K gerada pela função RBF com $l = 1$.*

$$\mathbf{K} = \begin{bmatrix} 1,0000 & 6,0653 \times 10^{-1} & 1,1109 \times 10^{-2} & 3,3546 \times 10^{-4} & 1,5230 \times 10^{-8} \\ 6,0653 \times 10^{-1} & 1,0000 & 1,3534 \times 10^{-1} & 1,1109 \times 10^{-2} & 3,7267 \times 10^{-6} \\ 1,1109 \times 10^{-2} & 1,3534 \times 10^{-1} & 1,0000 & 6,0653 \times 10^{-1} & 1,1109 \times 10^{-2} \\ 3,3546 \times 10^{-4} & 1,1109 \times 10^{-2} & 6,0653 \times 10^{-1} & 1,0000 & 1,3534 \times 10^{-1} \\ 1,5230 \times 10^{-8} & 3,7267 \times 10^{-6} & 1,1109 \times 10^{-2} & 1,3534 \times 10^{-1} & 1,0000 \end{bmatrix}$$

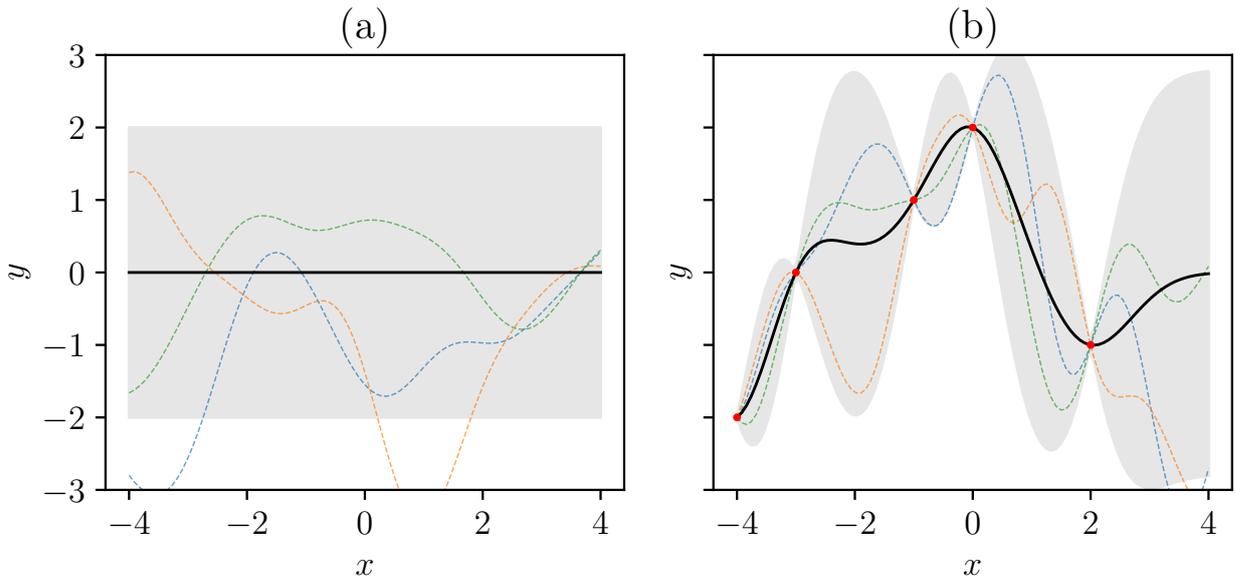
Utilizando a decomposição de Cholesky para \mathbf{K} , obtém-se a matriz \mathbf{L} tal que $\mathbf{K} = \mathbf{L}\mathbf{L}^\top$:

$$\mathbf{L} = \begin{bmatrix} 1,0000 & 0,0000 & 0,0000 & 0,0000 & 0,0000 \\ 6,0653 \times 10^{-1} & 7,9506 \times 10^{-1} & 0,0000 & 0,0000 & 0,0000 \\ 1,1109 \times 10^{-2} & 1,6175 \times 10^{-1} & 9,8677 \times 10^{-1} & 0,0000 & 0,0000 \\ 3,3546 \times 10^{-4} & 1,3717 \times 10^{-2} & 6,1241 \times 10^{-1} & 7,9042 \times 10^{-1} & 0,0000 \\ 1,5230 \times 10^{-8} & 4,6756 \times 10^{-6} & 1,1257 \times 10^{-2} & 1,6250 \times 10^{-1} & 9,8665 \times 10^{-1} \end{bmatrix}$$

Considerando um vetor de variáveis aleatórias com distribuição normal padrão $\mathbf{z} = [0,3870; 1,4394; 0,1550; -0,1211; 0,7096]$, e utilizando $\mathbf{f}^* = \mathbf{m} + \mathbf{L}\mathbf{z}$, onde \mathbf{m} é o vetor de médias que neste momento será tomado como zero, obtém-se o vetor $\mathbf{f}^* = [0,3870; 1,3791; 0,3901; 0,0190; 0,6822]$.

Para cada vetor de variáveis aleatórias \mathbf{z} obtém-se um vetor \mathbf{f}^* diferente, permitindo assim diferentes funções que satisfazem a matriz de covariância. Na Figura 16(a) são apresentadas 3 funções que satisfazem uma matriz de covariância para 100 pontos no intervalo $[-4, 4]$, tomando a distribuição a priori. Já na Figura 16(b) são apresentadas 3 funções que satisfazem a matriz de covariância considerando a distribuição a posteriori a partir dos pontos $(-4, -2)$, $(-3, 0)$, $(-1, 1)$, $(0, 2)$, $(2, -1)$.

Figura 16 - (a) Distribuição *a priori*. (b) Distribuição *a posteriori*.



Legenda: Diferentes funções a partir da distribuição *a priori* e posteriori, respectivamente. Em ambas as figuras a curva em preto representa a média e a área sombreada um intervalo de confiança de 95%. A função de covariância utilizada foi a RBF com $l = 0,699$.

Fonte: O autor, 2024.

3.2.2 Regressão com Dados Ruidosos

Suponha agora que os dados de saída y são ruidosos, isto é, $y = f(x) + \epsilon$, onde $\epsilon \sim \mathcal{N}(0, \sigma_n^2)$. Neste caso, usando a função RBF, a função de covariância poderia ser dada por

$$\text{cov}(y_p, y_q) = k(x_p, x_q) = \sigma_f^2 \exp\left(-\frac{1}{2l^2} \|x_p - x_q\|^2\right) + \sigma_n^2 \delta_{pq}, \quad (68)$$

A distribuição conjunta entre os pontos de treinamento (pontos observados), e os pontos não observados é dada por

$$\begin{bmatrix} \mathbf{y} \\ \mathbf{f}^* \end{bmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} \mathbf{K} + \sigma_n^2 \mathbf{I} & \mathbf{K}_* \\ \mathbf{K}_*^T & \mathbf{K}_{**} \end{bmatrix}\right), \quad (69)$$

onde \mathbf{I} é a matriz identidade. Assim, analogamente ao que se tem na expressão dada na Equação (65), o valor predito para os pontos não observados segue uma distribuição gaussiana, dada por

$$\mathbf{f}^* \sim \mathcal{N}(\mathbf{K}_*^T (\mathbf{K} + \sigma_n^2 \mathbf{I})^{-1} \mathbf{y}, \mathbf{K}_{**} - \mathbf{K}_*^T (\mathbf{K} + \sigma_n^2 \mathbf{I})^{-1} \mathbf{K}_*). \quad (70)$$

Novamente, de forma explícita, a predição em um ponto x_* e sua covariância podem

ser dados, respectivamente, por

$$m(x_*) = \mathbf{K}_*^T(\mathbf{K} + \sigma_n^2\mathbf{I})^{-1}\mathbf{y} \quad \text{e} \quad k(x_*) = \mathbf{K}_{**} - \mathbf{K}_*^T(\mathbf{K} + \sigma_n^2\mathbf{I})^{-1}\mathbf{K}_*. \quad (71)$$

3.2.3 Otimização dos Hiperparâmetros

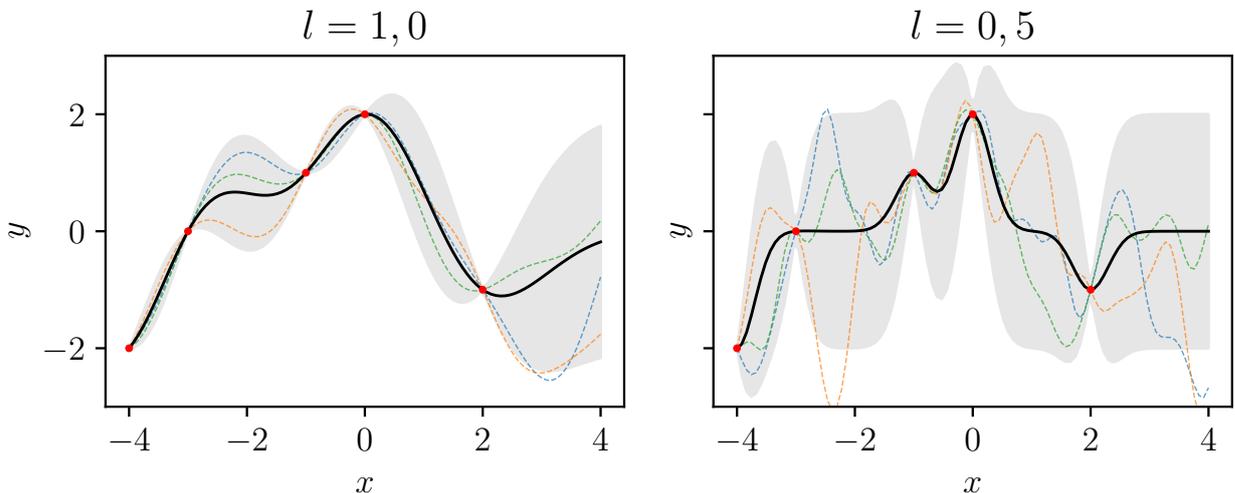
A escolha dos hiperparâmetros pode influenciar significativamente a capacidade da função de covariância em representar adequadamente a relação entre os dados. Por exemplo, considere o *kernel* RBF

$$k(x_p, x_q) = \sigma_f^2 \exp\left(-\frac{1}{2l^2}\|x_p - x_q\|^2\right), \quad (72)$$

onde σ_n^2 é a variância do ruído e σ_f^2 e l são a variância de f e o comprimento de escala, respectivamente.

Ao se realizar a regressão com Processos Gaussianos utilizando esta função de covariância e fixando os valores em $l = 1$ ou $l = 0,5$, e utilizando como pontos observados os mesmos presentes na Figura 16(b), pode-se obter diferentes estimativas para f , conforme pode ser visualizado Figura 17.

Figura 17 - Regressão com processos Gaussianos utilizando a função *kernel* RBF para diferentes valores de l e $\sigma_f^2 = 1$.



Fonte: O autor, 2024.

Observe que a função de covariância RBF com $l = 1$ resulta em uma curva mais suave. Em contrapartida, ao utilizar $l = 0,5$, a curva torna-se mais oscilatória, evidenciando a importância da escolha adequada dos hiperparâmetros ao se utilizar a regressão com Processos Gaussianos.

Na Figura 16(b), os hiperparâmetros l e σ_f^2 foram estimados a partir dos dados, onde obteve-se $l = 0,699$ e $\sigma_f^2 = 1,4^2$. Em geral, o *kernel* possui como hiperparâmetros a variância σ_f^2 , o comprimento de escala l e a variância do ruído σ_n^2 .

Apesar da presença destes parâmetros, a regressão com Processos Gaussianos é dita não paramétrica, pois estes não são fixados a priori, mas são estimados a partir dos dados: daí advém a denominação de *hiperparâmetros* para os parâmetros livres na função de covariância.

A fim de estimar estes hiperparâmetros, considere o conjunto de dados observados $\mathcal{D} = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \dots, (\mathbf{x}_n, y_n)\}$, onde \mathbf{x}_i representa o vetor de entrada e y_i é a saída correspondente. Aqui a saída y é considerada como ruidosa, ou seja, $y = f(\mathbf{x}) + \epsilon$, onde $\epsilon \sim \mathcal{N}(0, \sigma_n^2)$.

O objetivo é encontrar os hiperparâmetros que maximizam a verossimilhança desses dados, ou seja, encontrar os hiperparâmetros que maximizam a probabilidade de se obter os dados observados, onde a verossimilhança é definida como:

$$p(\mathbf{y}|\mathbf{X}, \theta) = \int p(\mathbf{y}|\mathbf{f}, \mathbf{X})p(\mathbf{f}|\mathbf{X}, \theta)d\mathbf{f}. \quad (73)$$

Aqui, $\theta = \{\sigma_f^2, l, \sigma_n^2\}$ representa o vetor de hiperparâmetros, \mathbf{f} é o vetor de saída dos dados não observados e \mathbf{X} é a matriz de entrada.

Porém, ao se considerar que \mathbf{y} segue uma distribuição normal multivariada com média zero e covariância $\mathbf{K} + \sigma_n^2\mathbf{I}$, a verossimilhança pode ser expressa como:

$$p(\mathbf{y}|\mathbf{X}, \theta) = \frac{1}{(2\pi)^{n/2}|\mathbf{K} + \sigma_n^2\mathbf{I}|^{1/2}} \exp\left(-\frac{1}{2}\mathbf{y}^\top(\mathbf{K} + \sigma_n^2\mathbf{I})^{-1}\mathbf{y}\right). \quad (74)$$

Daí, log-verossimilhança associada é:

$$\ln p(\mathbf{y}|\mathbf{X}, \theta) = -\frac{1}{2}\mathbf{y}^\top(\mathbf{K} + \sigma_n^2\mathbf{I})^{-1}\mathbf{y} - \frac{1}{2} \ln |\mathbf{K} + \sigma_n^2\mathbf{I}| - \frac{n}{2} \ln(2\pi). \quad (75)$$

Assim, pode-se estimar os hiperparâmetros maximizando a log-verossimilhança utilizando o gradiente descendente, por exemplo.

Uma das vantagens do uso de Processos Gaussianos está na possibilidade de estimar a incerteza associada à predição do modelo. Isso pode ser útil na busca pelo ponto mais promissor no espaço de busca para a avaliação da função objetivo e a atualização do metamodelo. Na seção 3.3, são apresentados métodos que utilizam a incerteza associada à predição do modelo para escolher o próximo ponto a ser avaliado. Essa abordagem permite associar Processos Gaussianos a uma meta-heurística populacional, na qual os indivíduos mais promissores da população são selecionados para a avaliação da função objetivo e a atualização do metamodelo.

3.3 Funções de Aquisição

No processo de otimização, a seleção das amostras para atualizar o metamodelo durante a execução de um algoritmo de busca pode, inicialmente, ser feita de maneira aleatória. No entanto, uma estratégia frequentemente empregada é o uso de uma função de aquisição para determinar quais amostras devem ser avaliadas a seguir para a atualização do metamodelo. De acordo com Schulz, Speekenbrink e Krause (2018), três funções de aquisição são comumente utilizadas: Melhoria Esperada (Expected Improvement, EI), Probabilidade de Melhoria (Probability of Improvement, PI) e Confiança Superior Limitada (Upper Confidence Bound, UCB).

Para entender como essas funções de aquisição podem ser utilizadas, considere a função $f : [0, 4] \rightarrow \mathbb{R}$, definida por

$$f(x) = \exp(-(x-2)^2) + \exp\left(\frac{-(x-6)^2}{10}\right) + \frac{1}{x^2+1}. \quad (76)$$

Suponha que o objetivo seja utilizar a Regressão com Processos Gaussianos para maximizar a função f . Em geral, a lei que define a função f é desconhecida, mas para fins de ilustração e comparação com o valor real, será assumido que a função f é conhecida.

A questão então é: como selecionar os pontos a serem avaliados pela função f de modo a encontrar o ponto ótimo? Nas subseções a seguir, serão apresentadas as funções de aquisição EI, PI e UCB, que podem ser empregadas para escolher os pontos a serem avaliados pela função f .

3.3.1 Função de Melhoria Esperada (EI)

Considere a função indicadora de melhoria dada por

$$I(x) = \begin{cases} f(x^*) - Y(x) & \text{se } Y(x) < f(x^*) \\ 0 & \text{se } Y(x) \geq f(x^*), \end{cases} \quad (77)$$

onde $f(x^*)$ é o melhor valor encontrado e $Y(x)$ é o valor predito pelo processo gaussiano para a amostra x , tal que $Y(x) \sim N(\mu(x), \sigma(x))$.

Tomando $z^*(x) = \frac{f(x^*) - \mu(x)}{\sigma(x)}$ e $z(x) = \frac{Y(x) - \mu(x)}{\sigma(x)}$, tem-se que a função de melhoria esperada pode ser dada por

$$\begin{aligned}
E[I(x)] &= \int_{-\infty}^{z^*} I(x)\phi(z)dz \\
&= \int_{-\infty}^{z^*} (f(x^*) - Y(x))\phi(z)dz \\
&= \int_{-\infty}^{z^*} \sigma(x)(z^* - z)\phi(z)dz \\
&= \sigma(x) \int_{-\infty}^{z^*} (z^* - z)\phi(z)dz \\
&= \sigma(x) \left[z^* \Phi(z^*) - \int_{-\infty}^{z^*} z\phi(z)dz \right] \\
&= \sigma(x) [z^* \Phi(z^*) - \phi(z^*)].
\end{aligned} \tag{78}$$

E assim finalmente é possível escrever,

$$EI(x) = (f(x^*) - \mu(x))\Phi\left(\frac{f(x^*) - \mu(x)}{\sigma(x)}\right) + \sigma(x)\phi\left(\frac{f(x^*) - \mu(x)}{\sigma(x)}\right). \tag{79}$$

Analogamente, se o problema for de maximização, a função de melhoria esperada pode ser dada por

$$EI(x) = (\mu(x) - f(x^*))\Phi\left(\frac{\mu(x) - f(x^*)}{\sigma(x)}\right) + \sigma(x)\phi\left(\frac{\mu(x) - f(x^*)}{\sigma(x)}\right).$$

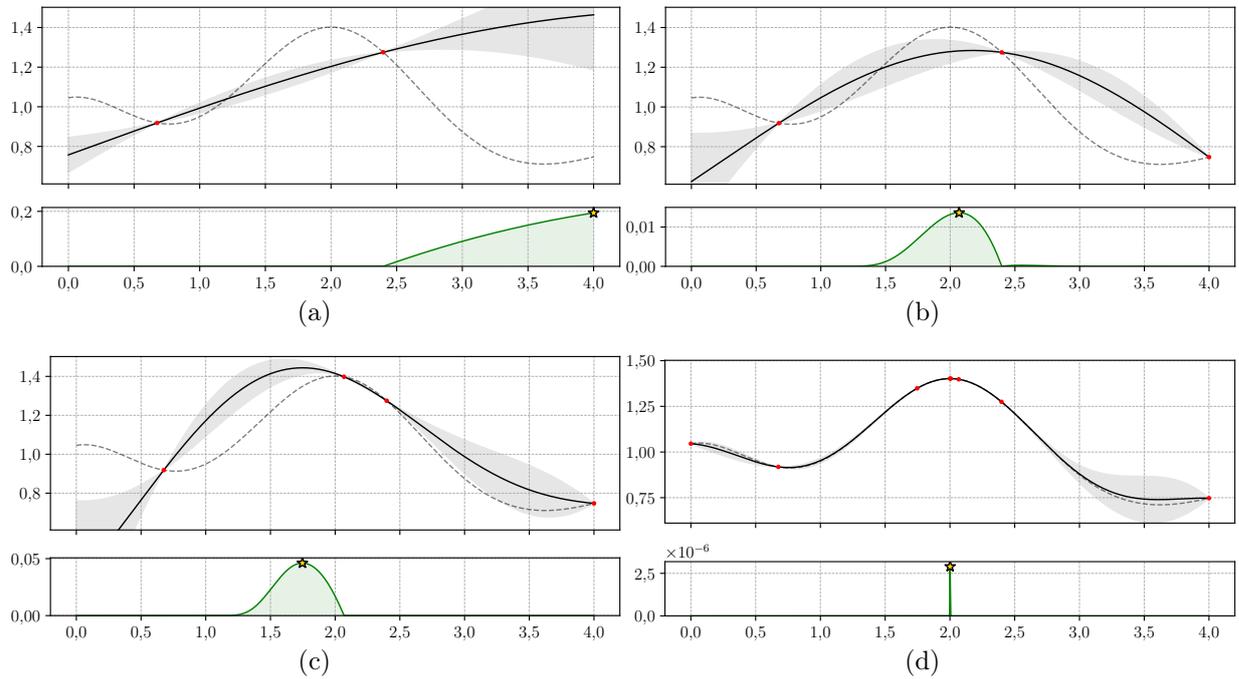
Retomando a função f que foi definida anteriormente na Equação (76), considere dois pontos aleatórios no intervalo $[0, 4]$ e os valores de f nesses pontos. Isto é ilustrado na Figura 18a, onde o gráfico de f é representado por uma linha tracejada.

Com base nessa escolha inicial, é possível construir o metamodelo e obter a função de melhoria esperada. Na Figura 18a, o gráfico resultante do metamodelo e da função de melhoria esperada são representados em preto e verde, respectivamente.

O passo seguinte envolve a seleção do próximo ponto a ser avaliado. Este ponto é escolhido como aquele que maximiza a função de melhoria esperada no domínio de f , fato que é ilustrado an Figura 18a. Este ponto é então utilizado para avaliar a função f e atualizar o metamodelo. A Figura 18b mostra o metamodelo atualizado com o ponto obtido em 18a e a função de melhoria esperada correspondente, com o novo ponto de maximização destacado.

Este procedimento é repetido até que o ponto de máximo da função de melhoria sofra poucas variações, como ilustrado nas Figuras 18c e 18d. Para a obtenção da Figura 18d foram necessários 7 pontos serem avaliados.

Figura 18 - Estimativa do gráfico de f a partir de pontos amostrados segundo a função de melhoria esperada.



Legenda: Nas imagens, a linha pontilhada ilustra o gráfico de f , a linha preta mostra a estimativa de f por processos gaussianos, a área cinza indica o intervalo de confiança de 95%, e a linha verde exibe a função EI, com destaque para o ponto de valor máximo.

Fonte: O autor, 2024.

3.3.2 Função de Probabilidade de Melhoria (PI)

Novamente, suponha que se deseje minimizar uma função objetivo $f : \mathbb{R}^n \rightarrow \mathbb{R}$, e que x^* é o melhor valor encontrado até o momento. Semelhante ao que foi feito na seção 3.3.1, pode-se definir a função indicadora de melhoria como

$$I(x) = \begin{cases} f(x^*) - Y(x) & \text{se } Y(x) < f(x^*) \\ 0 & \text{se } Y(x) \geq f(x^*), \end{cases} \quad (80)$$

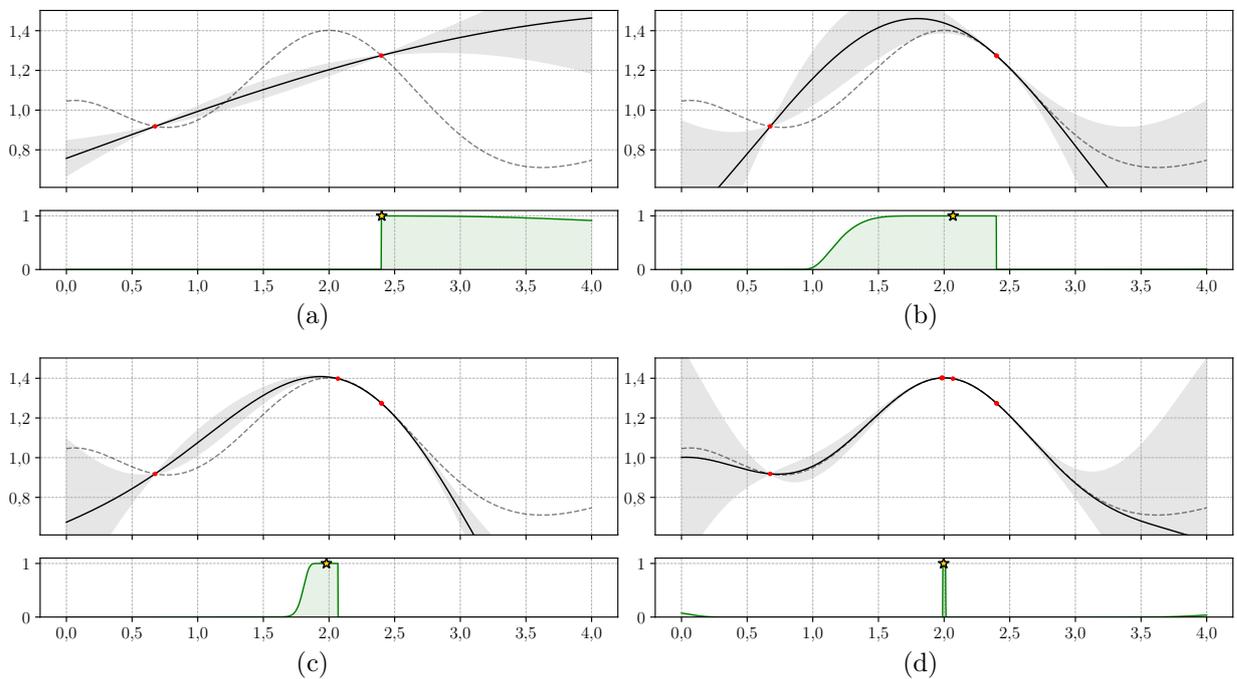
onde $Y(x)$ é o valor predito pelo processo gaussiano para a amostra x , tal que $Y(x) \sim N(\mu(x), \sigma(x))$.

A distribuição de probabilidade acumulada da função indicadora de melhoria pode ser dada por $\Phi_{I(x)}(i) = P[I(x) < i]$.

Tomando $i > 0$, e $Z = \frac{Y(x) - \mu(x)}{\sigma(x)}$, tem-se que

$$\begin{aligned}
\Phi_{I(x)}(i) &= P[f(x^*) - Y(x) < i] \\
&= P[f(x^*) - \mu(x) - \sigma(x)Z < i] \\
&= P\left[-Z < \frac{-f(x^*) + \mu(x) + i}{\sigma(x)}\right] \\
&= 1 - \Phi\left(\frac{f(x^*) - \mu(x) - i}{\sigma(x)}\right).
\end{aligned} \tag{81}$$

Figura 19 - Estimativa do gráfico de f a partir de pontos amostrados segundo a função probabilidade de melhoria.



Legenda: As imagens apresentam a curva pontilhada que representa f , a curva preta é a estimativa de f por processos gaussianos, a área cinza indica o intervalo de confiança de 95%, e a curva verde é a função PI, destacando o ponto de valor máximo.

Fonte: O autor, 2024.

Assim a probabilidade de melhoria pode ser dada, segundo Passos (2020), por

$$\begin{aligned}
PI(x) &= 1 - P[I(x) \leq 0] \\
&= 1 - P[I(x) < 0] - P[I(x) = 0] \\
&= 1 - 0 - \Phi_{I(x)}(0) \\
&= 1 - 1 + \Phi\left(\frac{f(x^*) - \mu(x)}{\sigma(x)}\right) \\
&= \Phi\left(\frac{f(x^*) - \mu(x)}{\sigma(x)}\right).
\end{aligned} \tag{82}$$

Assim, pode-se definir a função de probabilidade de melhoria como:

$$PI(x) = \Phi \left(\frac{f(x^*) - \mu(x)}{\sigma(x)} \right). \quad (83)$$

Analogamente, se o problema for de maximização, a função de probabilidade de melhoria pode ser dada por:

$$PI(x) = \Phi \left(\frac{\mu(x) - f(x^*)}{\sigma(x)} \right). \quad (84)$$

Novamente retomando a função f definida na Equação (76), e seguindo os mesmos procedimentos da subseção anterior, mas agora utilizando a função de probabilidade de melhoria, obtém-se as Figura 19, onde a imagem 19d é obtida com 7 pontos avaliados utilizando a função de probabilidade de melhoria.

3.3.3 Função de Aquisição Confiança Superior Limitada (UCB)

Uma ideia intuitiva que pode ser utilizada para a seleção das amostras dentro de um determinado conjunto é explorar a incerteza estimada pelo processo gaussiano. Quando o processo é mais incerto, pode-se escolher amostras para avaliação e, assim, atualizar o metamodelo. Uma das funções de aquisição mais simples e comumente utilizadas é a função de aquisição de Confiança Superior Limitada (UCB), que utiliza o valor predito pelo metamodelo e a incerteza associada à predição. A função de aquisição UCB é definida por:

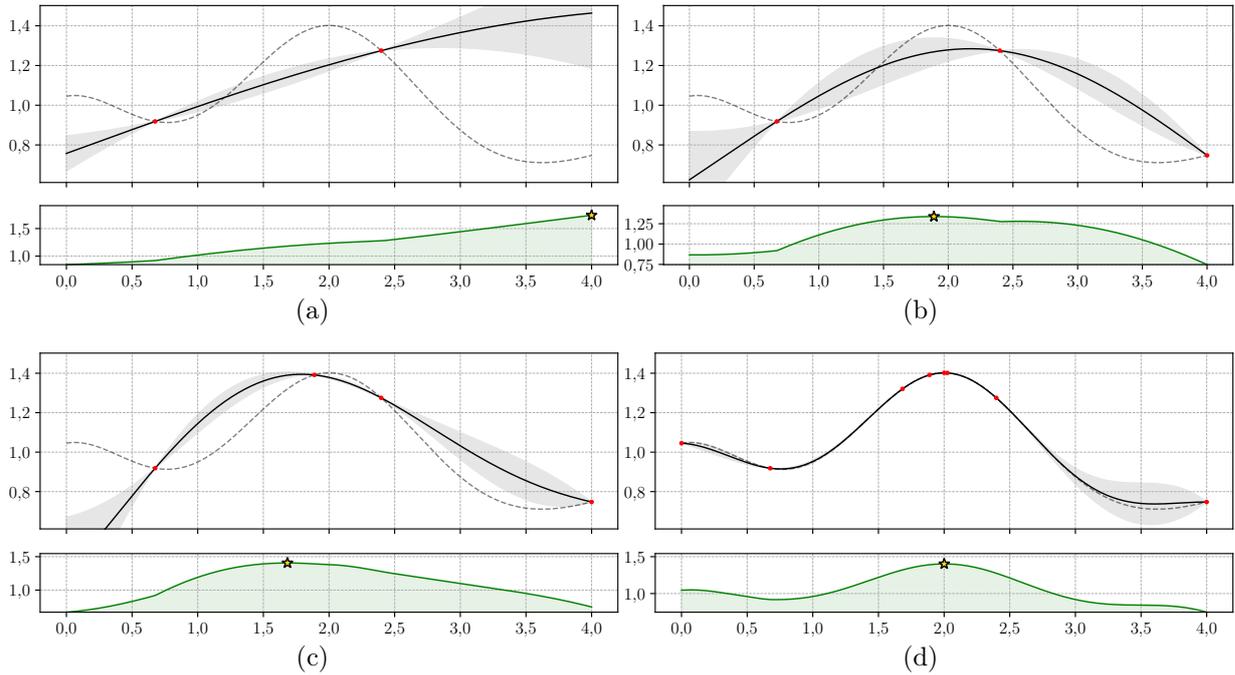
$$UCB(x) = \mu(x) + \kappa\sigma(x), \quad (85)$$

onde κ é um parâmetro que controla o impacto da incerteza (exploração e exploração), $\mu(x)$ e $\sigma(x)$ são o valor esperado e o desvio padrão, respectivamente, tal que $Y(x) \sim N(\mu(x), \sigma(x))$.

Conforme Shahriari et al. (2016), a função de aquisição UCB possibilita um equilíbrio entre exploração e exploração. Isso significa que ela permite a seleção de amostras que otimizam tanto o valor médio previsto pelo metamodelo quanto a incerteza associada a essa previsão. O equilíbrio entre esses dois aspectos é alcançado através do ajuste do parâmetro κ , que determina a importância da incerteza na seleção das amostras: valores maiores de κ favorecem uma maior exploração, enquanto valores menores de κ conduzem a uma maior exploração.

Nas imagens presentes na Figura 20, a função de aquisição UCB é utilizada para selecionar os pontos a serem avaliados pela função f definida na Equação (76), de forma

Figura 20 - Estimativa do gráfico de f a partir de pontos amostrados segundo a função UCB.



Legenda: Nas imagens, a curva pontilhada é o gráfico de f , a curva preta é a regressão usando processos gaussianos, a área cinza representa o intervalo de confiança de 95%, e a curva verde é a função UCB com $\kappa = 2$, destacando o ponto de valor máximo.

Fonte: O autor, 2024.

análoga ao que foi visto nas duas seções anteriores. A imagem 20d foi obtida com 7 pontos avaliados utilizando a função de aquisição UCB.

De acordo com Jones, Schonlau e Welch (1998), maximizar a função de aquisição permite selecionar a próxima amostra a ser avaliada com maior precisão, o que, por sua vez, possibilita atualizar o metamodelo nas regiões de maior incerteza, tornando o processo de otimização mais eficiente.

4 ALGORITMO DE OTIMIZAÇÃO COM METAMODELAGEM CONSIDERANDO ROBUSTEZ E CONFIABILIDADE

Um dos principais desafios de se analisar a robustez está na necessidade de se avaliar a função objetivo para cada uma das M amostras ξ_i da δ -vizinhança $\mathcal{B}_\delta(\mathbf{x})$, o que pode ser computacionalmente inviável, dependendo da complexidade da função objetivo envolvida.

Visando reduzir o custo computacional ocasionado pela análise de robustez, é proposta a utilização de metamodelos, especificamente regressão com processos gaussianos, para o cálculo dos valores da função de média efetiva e, assim, reduzir o custo computacional na perspectiva de quantidade de avaliações da função objetivo durante o processo de otimização, mas alcançando uma solução tão próxima quanto possível àquela do problema correspondente em que metamodelos não são empregados.

Nas seções seguintes será descrito como a metamodelagem pode ser empregada em conjunto com uma meta-heurística populacional para o processo de otimização envolvendo robustez e confiabilidade.

4.1 O Algoritmo de Avaliação da Robustez Utilizando Metamodelagem

Para a realização do processo de otimização, será utilizado um algoritmo evolutivo que emprega metamodelos para avaliar a robustez em substituição à função objetivo original. O algoritmo proposto é composto por quatro estágios, que são descritos a seguir.

O algoritmo evolutivo é iniciado com uma população de soluções candidatas gerada por meio da estratégia do hipercubo latino (HCL), e cada indivíduo dessa população é avaliado segundo a função objetivo. Com os dados de entrada e saída obtidos (rotulados), o modelo de regressão baseado em Processos Gaussianos é construído (treinado).

Durante o segundo estágio do algoritmo, a população gerada na etapa anterior serve como a população inicial. Além disso, o modelo de regressão baseado em Processos Gaussianos, previamente treinado, atua como a função objetivo na avaliação da função de média efetiva. Aqui, a função de média efetiva é a função que se busca otimizar.

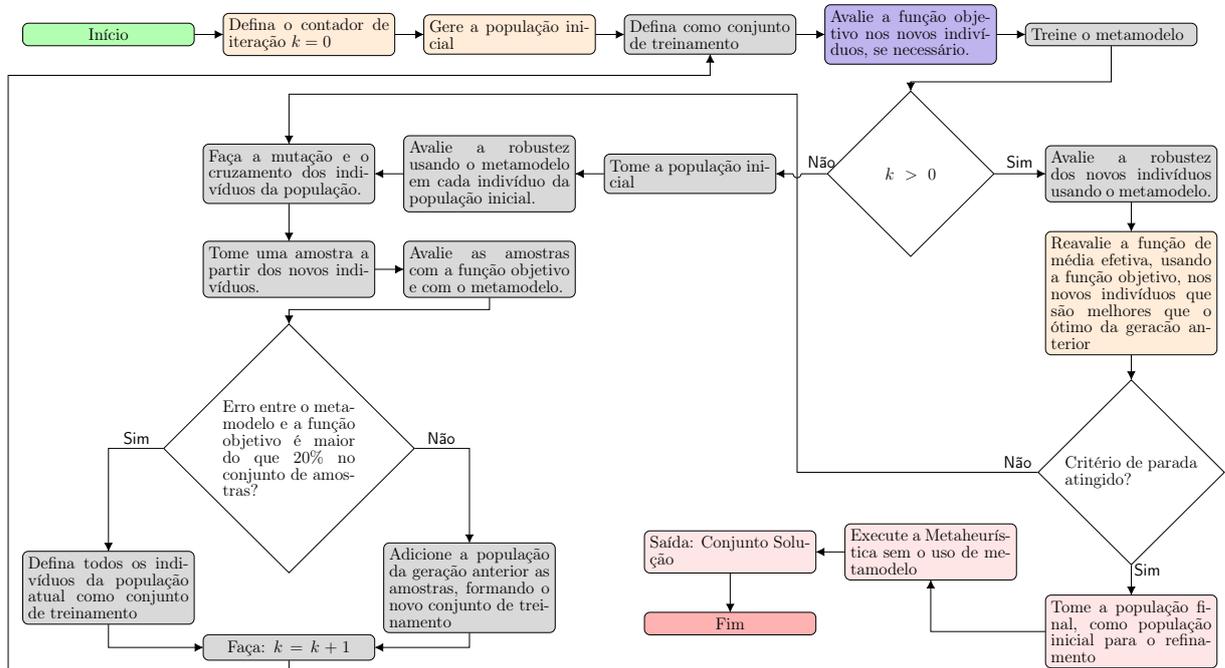
No terceiro estágio, um conjunto de teste é tomado aleatoriamente, ou utilizando uma das funções de aquisição, a partir dos novos indivíduos criados na etapa de mutação e *crossover* da nova geração. Se o erro entre o metamodelo e a função original nestes indivíduos for arbitrariamente maior do que 20%, a função objetivo é avaliada em todos os indivíduos da população atual, e o metamodelo é atualizado; se o erro for menor ou igual a 20%, os indivíduos do conjunto de teste são adicionados ao conjunto de treinamento e o metamodelo é atualizado. Em seguida, usa-se o modelo de regressão baseado em Processos

Gaussianos (atualizado), como a função objetivo para avaliar a robustez dos candidatos a solução por meio da abordagem da função de média efetiva. Em cada geração, se um novo ponto ótimo robusto é encontrado, este é avaliado pela função original a fim de determinar o valor da função de média efetiva. Esse processo se repete a cada geração, até que uma condição de parada na utilização do modelo de regressão seja satisfeita.

No momento em que um critério de término para o uso do metamodelo é alcançado, indicando o encerramento da fase de metamodelagem, o processo avança para o quarto estágio, aqui denominado como a fase de refinamento. Nesse estágio, o algoritmo deixa de utilizar o metamodelo e passa a avaliar os indivíduos da população por meio da função objetivo original para o cálculo da média efetiva. Isto pode ser visualizado no fluxograma da Figura 21, onde se tem o fluxo de funcionamento do algoritmo proposto.

A fase de refinamento é adotada para garantir que o metamodelo não seja utilizado para determinar a solução robusta final, pois o metamodelo é uma aproximação da função objetivo e pode obter soluções imprecisas se comparadas ao modelo original. Com isso, pode-se dizer que o metamodelo é utilizado na fase exploratória do algoritmo, e a função objetivo original é utilizada na fase de refinamento, adotando uma abordagem semelhante à realizada por Baquela e Olivera (2019).

Figura 21 - Fluxograma do algoritmo de avaliação da robustez utilizando metamodelagem.



Fonte: O autor, 2024.

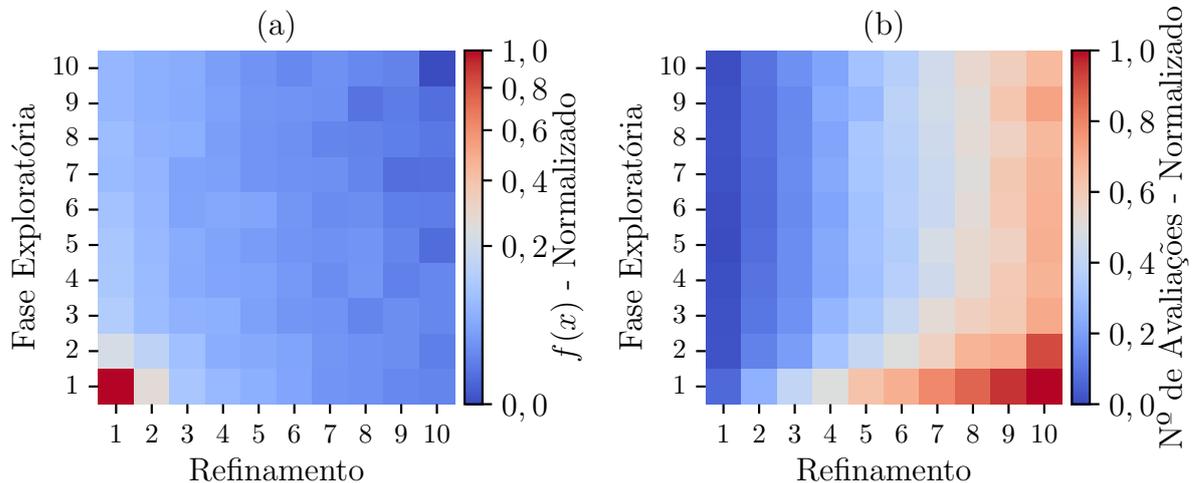
Nesta abordagem híbrida (metamodelo e modelo original), um dos parâmetros essenciais para o funcionamento do algoritmo é a escolha do critério de parada para a utilização do metamodelo e o início da fase de refinamento.

Neste trabalho, foi adotado como critério principal de parada o uso de uma janela

deslizante: se não houver melhoria no valor da função de média efetiva ou no conjunto de soluções encontradas dentro de uma determinada tolerância e número de gerações, o uso do metamodelo é interrompido e dá-se início à fase de refinamento. Assim, nesta fase final, a função objetivo original é utilizada para avaliar os indivíduos da população no cálculo da função de média efetiva, adotando novamente a janela deslizante como critério de parada.

Na Figura 22 é apresentada uma comparação entre diferentes valores para a quantidade de gerações fixada na janela deslizante na utilização do metamodelo e na fase de refinamento. Em cada combinação de parâmetros, o experimento foi executado 100 vezes e, em seguida, calculada a média e normalizados os resultados a partir do problema de minimização que será visto na seção 1.

Figura 22 - Impacto de diferentes tamanhos da janela deslizante no número de avaliações da função.



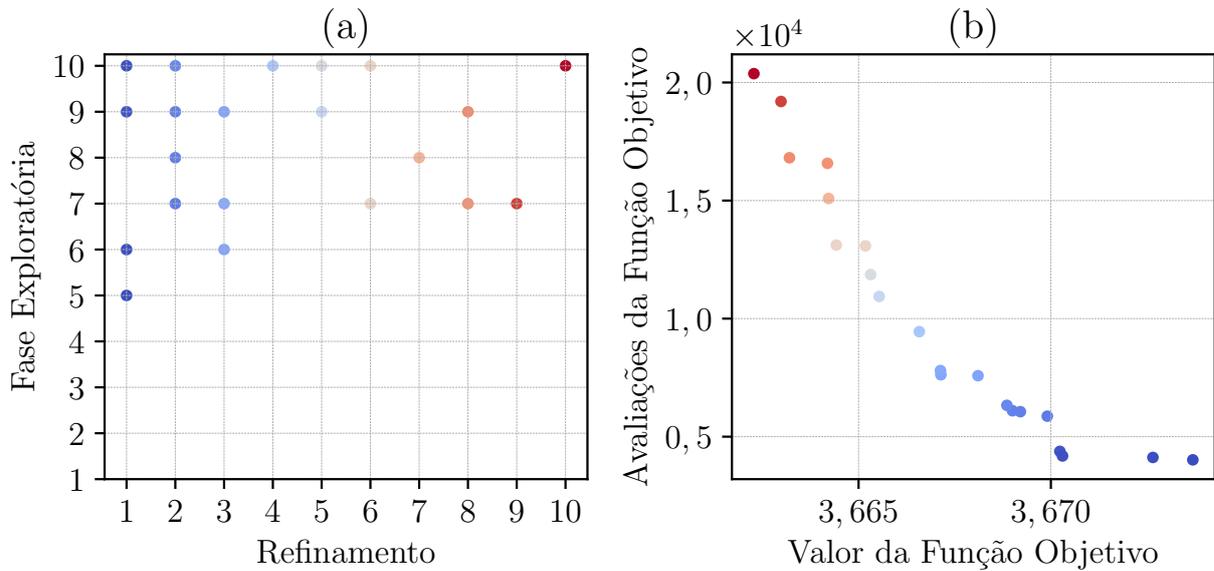
Legenda: Na Figura (a), é apresentado o impacto do tamanho da janela deslizante de cada fase no valor da função objetivo. Já na Figura (b), é apresentado o impacto do tamanho da janela deslizante em cada fase no número de avaliações da função objetivo.

Fonte: O autor, 2024.

Numa comparação entre as Figuras 22a e 22b, observa-se que o aumento excessivo da quantidade de gerações estáveis na fase de refinamento pode não proporcionar ganhos significativos no valor da função objetivo em termos de valores ótimos melhores, mas aumentar o número de avaliações, o que não é desejável.

Para obter um ponto ótimo robusto e reduzir o número de avaliações da função objetivo durante o processo de otimização, a escolha do tamanho da janela deslizante em cada fase pode ser conflitante. Uma janela deslizante pequena na fase de metamodelagem (exploratória) e grande na fase de refinamento, pode não reduzir o número de avaliações da função objetivo como esperado. Por outro lado, uma janela deslizante pequena na fase de refinamento e grande na de metamodelagem pode resultar em valores subótimos.

Figura 23 - Comparação entre o número de avaliações da função objetivo e o valor ótimo obtido.



Legenda: Na Figura (a), é apresentado o conjunto de pareto considerando os diferentes valores das janela deslizante. Já na Figura (b), é apresentada a curva de pareto relacionando os valores da função objetivo com o número de avaliações necessários para obtê-lo .

Fonte: O autor, 2024.

Assim, pode-se encontrar a curva que descreve o compromisso entre o valor da função objetivo e o número de avaliações da função objetivo de acordo com o tamanho de cada janela deslizante. Essa curva pode ser visualizada na Figura 23.

O comparativo revela a importância da seleção do tamanho da janela deslizante para as duas diferentes fases de execução do algoritmo. Destaca-se que o dimensionamento da janela na fase exploratória pode proporcionar benefícios significativos, reduzindo o custo computacional e otimizando o valor alcançado. Observando a Figura 22b, também é possível notar que o aumento do tamanho da janela deslizante na fase exploratória tem o potencial de diminuir o número de avaliações da função objetivo. Essa calibração deve ser ajustada à complexidade do problema em questão, sendo que uma escolha imprópria pode resultar em um incremento no número de avaliações necessárias da função objetivo, o que poderia até mesmo exceder os custos computacionais de métodos que não utilizam metamodelos.

4.2 O Algoritmo de Avaliação da Robustez e Confiabilidade Utilizando Metamodelagem

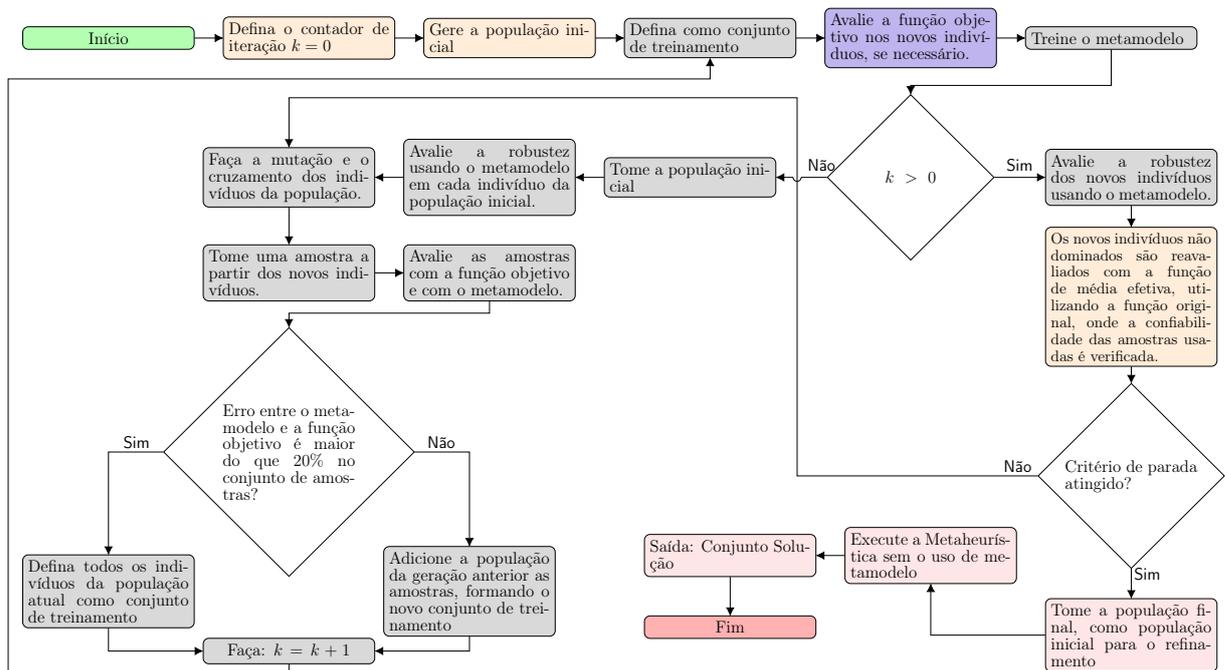
De forma similar a metodologia apresentada na seção anterior, o algoritmo de otimização baseado em robustez e confiabilidade com metamodelagem também é estruturado em quatro etapas, utilizando uma meta-heurística populacional. No entanto, neste caso,

lida-se com um problema multiobjetivo, conforme o descrito na seção 1.6, sendo que a função que representa o índice de confiabilidade, um objetivo que se busca maximizar, não é metamodelado. Ou seja, um metamodelo é desenvolvido para cada função objetivo, com exceção do índice de confiabilidade.

Seguindo a mesma metodologia da metamodelagem focada apenas na robustez, o metamodelo de cada função objetivo é atualizado sempre que o erro entre o metamodelo e a função objetivo original exceder 20% em um conjunto de amostras nos novos indivíduos que surgem a cada geração.

Para garantir a eficácia do metamodelo, sempre que ele estimar um valor em que a solução não é dominada em relação aos indivíduos da geração anterior, as funções originais são utilizadas para avaliar a robustez dessa solução. Se a solução permanecer não dominada após a avaliação da função objetivo, verifica-se a confiabilidade da solução e das M amostras utilizadas para o cálculo da robustez, juntamente com o valor de β da solução. Na Figura 24, é possível observar o fluxograma do algoritmo proposto para o caso de otimização baseada em robustez e confiabilidade com metamodelagem.

Figura 24 - Fluxograma do algoritmo de avaliação da robustez e confiabilidade utilizando metamodelagem.



Fonte: O autor, 2024.

Uma abordagem alternativa que pode ser utilizada para a otimização baseada em confiabilidade é o método SORA (do inglês *Sequential Optimization and Reliability Assessment*), proposto por Du e Chen (2004). Nesse método, a análise de confiabilidade é separada (desacoplada) da otimização determinística e realizada em ciclos iterativos. No primeiro ciclo, a otimização determinística é conduzida considerando as variáveis aleató-

rias em seus valores médios. Após essa etapa, a confiabilidade é avaliada para cada uma das restrições, buscando-se o MPP correspondente. Se for verificado que uma restrição probabilística é violada, essa restrição é ajustada por meio de um vetor de deslocamento aplicado à restrição original. Esse processo iterativo continua até que todas as restrições probabilísticas sejam satisfeitas (Kaveh; Zaerreza, 2022).

Essa abordagem pode ser utilizada para problemas que envolvem robustez e confiabilidade (RBRDO), integradas à metamodelagem: Cada ciclo do SORA pode ser modificado, dividindo a otimização robusta em duas fases: a fase de metamodelagem e a fase de refinamento. Ambas as fases focam na obtenção de soluções robustas usando a função de média efetiva. Após essas fases, realiza-se a análise de confiabilidade para cada uma das restrições, com um índice de confiabilidade pré-definido. Esse processo se repete até que todas as restrições sejam satisfeitas. É importante destacar que a análise de confiabilidade é feita em um conjunto de M amostras ao redor da solução obtida.

Para mais detalhes sobre o funcionamento do SORA consulte Du e Chen (2004), Kaveh e Zaerreza (2022).

5 RESULTADOS

A aplicação da Regressão com Processos Gaussianos na metamodelagem da otimização baseada em robustez e confiabilidade visa diminuir o esforço computacional envolvido na análise de incertezas. Neste capítulo, são apresentados os resultados dessa aplicação, demonstrando como essa técnica contribui para a otimização com incertezas no tocante à redução do custo computacional.

Ao se analisar o custo computacional em problemas de otimização, é importante utilizar uma métrica que seja minimamente influenciada por fatores externos ao problema, como o hardware, por exemplo. A velocidade do processador, a quantidade de memória RAM e a presença de GPUs influenciam significativamente o tempo de execução, tornando essa métrica variável e dependente do ambiente de execução.

Neste contexto, o número de avaliações da função objetivo se mostra como uma métrica mais apropriada, permitindo uma comparação consistente do desempenho. Sendo assim, esta métrica é adotada ao longo das seções seguintes deste trabalho, para comparar o impacto do uso e do não uso de metamodelos na otimização sob incertezas.

Para o cálculo da função de média efetiva, a quantidade de amostras foi modificada de acordo com a complexidade do problema. Em cenários de menor complexidade, foram utilizadas 20 amostras, enquanto em contextos mais desafiadores, foram empregadas 50 amostras.

Já no tocante às metaheurísticas populacionais, estas foram empregadas utilizando a biblioteca Python *Pymoo*, elaborada por Blank e Deb (2020). Os parâmetros, incluindo a probabilidade de mutação, de *crossover*, o tamanho da população e o número de descendentes, foram selecionados com base no número de variáveis envolvidas em cada problema, buscando-se balancear a exploração e a exploração do espaço de busca.

Para a construção e atualização dos metamodelos baseados em Regressão com Processos Gaussianos, utilizou-se a função kernel RBF da biblioteca Python *Scikit-learn*, baseada em Rasmussen e Williams (2006). Os hiperparâmetros $\theta = \{\sigma_f^2, l, \sigma_n^2\}$ do kernel RBF foram otimizados pelo método L-BFGS-B (do inglês *Limited-memory Broyden-Fletcher-Goldfarb-Shanno Bound*), visando maximizar a função de log-verossimilhança marginal, buscando os valores ótimos que melhor se ajustam aos dados de treinamento, conforme discutido na seção 3.2.3.

Para a análise estrutural dos problemas de engenharia abordados neste capítulo, foi implementado um método de elementos finitos focado na determinação dos deslocamentos nodais e tensões resultantes. A base para esta implementação foi o trabalho detalhado de Ferreira e Fantuzzi (2020), que fornece uma orientação para a construção.

Inicialmente, na Seção 5.1, são apresentados estes resultados relacionados apenas à robustez, e uma comparação com a otimização sem o uso de metamodelos é realizada,

visando observar a eficiência da metodologia proposta na redução do custo computacional.

Na seção 5.2, são abordados problemas exclusivamente relacionados à confiabilidade (excluindo robustez), com o objetivo de verificar se a utilização da projeção estereográfica possibilita a obtenção de soluções comparáveis às registradas na literatura.

Na Seção 5.3 são apresentados os resultados da aplicação da metamodelagem da otimização baseada em robustez integrada com a otimização baseada em confiabilidade, onde se avalia se o emprego de metamodelos mantém a redução do custo computacional e a eficácia da metodologia proposta.

5.1 Robustez

Nesta seção, as soluções obtidas com e sem a utilização de metamodelos são comparadas. Inicialmente, um problema numérico de baixa dimensão, composto por duas variáveis de projeto, é resolvido com o intuito de ilustrar o comportamento da metodologia proposta nos resultados alcançados. Tal problema inicial serve como um exemplo que demonstra a tendência de eficácia da abordagem, além de facilitar a compreensão dos conceitos envolvidos.

Em seguida, dois problemas de engenharia similares são analisados, visando verificar a eficiência da abordagem proposta quando a dimensão é um pouco mais elevada. Em todos os casos citados, Algoritmos Genéticos (GA) são utilizados para a resolução dos problemas, e a robustez é avaliada por meio da função de média efetiva.

Problema 1 Considere a função $f : \mathbb{R}^2 \rightarrow \mathbb{R}$, e o respectivo problema de *benchmark* dado por

$$\begin{aligned} \underset{\mathbf{x}}{\text{Minimize}} \quad & f(\mathbf{x}) = \exp\left(\frac{1}{100}(\bar{x}_1 + \bar{x}_2)^2\right) - 0,8 \exp(-6,25((\bar{x}_1 + 2)^2 + (\bar{x}_2 + 1)^2)) \\ & - \exp(-25((\bar{x}_1 + 2,5)^2 + (\bar{x}_2 + 2,5)^2)) \\ \text{Sujeito a} \quad & g_1(\mathbf{x}) = (\bar{x}_1^2 + \bar{x}_2^2 + 2r_1\bar{x}_1)^2 - 4r_1^2(\bar{x}_1^2 + \bar{x}_2^2) + 2 \geq 0 \\ & g_2(\mathbf{x}) = (\bar{x}_1^2 + \bar{x}_2^2)^3 - 4r_2\bar{x}_1^2\bar{x}_2^2 \geq 0 \\ & g_3(\mathbf{x}) = ((\bar{x}_1 + 1)^2 + \bar{x}_2^2 - 2r_3(\bar{x}_1 + 1))^2 + 4r_3^2((\bar{x}_1 + 1)^2 + \bar{x}_2^2) - 2 \geq 0 \\ & -14 \leq x_1, x_2 \leq 0, \end{aligned} \tag{86}$$

onde $r_1 = 1,2054$, $r_2 = 1,6936 \times 10^1$, $r_3 = 1,2054$, e $\mathbf{x} = (\bar{x}_1, \bar{x}_2) = (\frac{x_1}{4}, \frac{x_2}{4})$.

Este problema é multimodal, e o mínimo global está localizado em $\mathbf{x}^g = (-10; -10)$, com $f(\mathbf{x}^g) \approx 0,284025$. Além disso, há um mínimo local em $\mathbf{x}^l = (-8; -4)$, com $f(\mathbf{x}^l) \approx 0,294174$ (ver Figura 25). Se considerarmos um nível de robustez de 10%, avaliando-se a função de média efetiva com 20 amostras nas vizinhanças dos pontos \mathbf{x}^g e \mathbf{x}^l , sem usar metamodelagem, obtém-se $f^{\text{eff}}(x^g) \approx 0,859956$ e $f^{\text{eff}}(x^l) \approx 0,371370$. Isso ilustra que o

ponto \mathbf{x}^l é menos sensível a variações do que o ponto \mathbf{x}^g , sendo assim uma solução mais adequada considerando incertezas, quando comparadas entre si. Desta forma, é esperado que ao se empregar metamodelagem ou não, a convergência ocorra para coordenadas próximas ao ponto $\mathbf{x}^l = (-8; -4)$, e assim se obtenha valores próximos $f^{\text{eff}}(\mathbf{x}^l)$.

Tabela 1 - Avaliações da função objetivo para a convergência do algoritmo genético para diferentes níveis de robustez.

Nível de Robustez	Usando Metamodelagem			Variação percentual entre as médias
	Sem o uso de Metamodelagem			
	Intervalo	Média	Desvio Padrão (C.V.)	
1%	(7210, 31530)	13705,98	5312,05 (38,75%)	71,45%
	(25200, 74700)	48015,00	10936,29 (22,77%)	
5%	(7110, 34134)	11905,00	4493,85 (37,74%)	67,43%
	(16200, 66000)	36558,00	9904,17 (27,09%)	
10%	(6950, 26823)	11704,33	4386,48 (37,47%)	64,85%
	(18000, 57900)	33300,00	9490,95 (28,50%)	

Legenda: Para cada nível de robustez a tabela apresenta o intervalo (mínimo e máximo), média, desvio padrão e a diferença percentual entre médias com e sem metamodelagem para o número de avaliações da função objetivo, obtido em execuções independentes.

Fonte: O autor, 2024.

Para a execução da metodologia proposta com este problema, os seguintes parâmetros foram utilizados nas 100 execuções efetuadas: tamanho da população fixa em 30 indivíduos, quantidade de amostras utilizadas para o cálculo da robustez $N = 20$, quantidade máxima de gerações $G_{max} = 11000$, probabilidades de mutação $p_m = 0,4$ e cruzamento $p_c = 0,8$ e número de descendentes 15.

Como critério de parada, utilizou-se uma tolerância de 10^{-6} para a diferença entre o melhor indivíduo da população atual e o melhor da geração anterior, durante uma janela de 40 gerações. Adicionalmente, foi aplicada uma tolerância de 10^{-8} para a diferença entre o melhor valor da função objetivo da população atual e o melhor valor da geração anterior, também por 40 gerações consecutivas. Estes parâmetros foram os utilizados para a execução sem o uso da metamodelagem (otimização robusta convencional com algoritmos genéticos) e para a fase que antecede o refinamento com a metodologia proposta. Já na fase de refinamento foi adotado como padrão a execução por uma janela de 20 gerações, mas com os mesmos demais parâmetros citados anteriormente.

Seguindo a metodologia proposta e observando os resultados presentes na Tabela 1, é possível perceber que houve uma economia de pelo menos 64,85% no número de avaliações da função objetivo para os diferentes níveis de robustez. É importante frisar que, nesta contagem de avaliações, estão computadas tanto as avaliações da função objetivo para o treinamento e atualização do metamodelo quanto para a fase de refinamento.

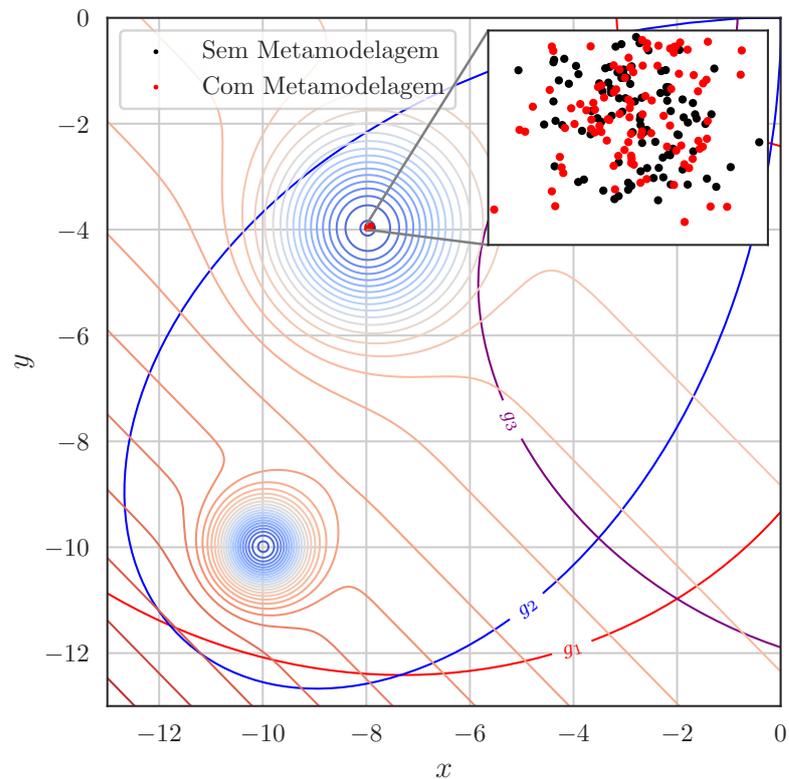
Tabela 2 - Valores da função objetivo obtidas com o algoritmo genético para diferentes níveis de robustez em 100 execuções.

Nível de Robustez	Usando Metamodelagem			Variação percentual entre as médias
	Sem o uso de Metamodelagem			
	Intervalo	Média	Desvio Padrão (C.V.)	
1%	(0,2945; 0,2946)	0,2945	$1,6000 \times 10^{-5}$ (0,005%)	0,002161%
	(0,2945; 0,2946)	0,2945	$1,2000 \times 10^{-5}$ (0,004%)	
5%	(0,3129; 0,3137)	0,3133	$1,6100 \times 10^{-4}$ (0,051%)	0,019489%
	(0,3128; 0,3141)	0,3132	$2,0100 \times 10^{-4}$ (0,064%)	
10%	(0,3656; 0,3684)	0,3672	$5,3900 \times 10^{-4}$ (0,14%)	0,086021%
	(0,3644; 0,36831)	0,3668	$6,5000 \times 10^{-4}$ (0,17%)	

Legenda: Para cada nível de robustez a tabela apresenta o intervalo (mínimo e máximo), média, desvio padrão e a diferença percentual entre médias com e sem metamodelagem para o valor ótimo da função objetivo, obtido em execuções independentes.

Fonte: O autor, 2024.

Figura 25 - Pontos ótimos obtidos em 100 execuções para 10% de Robustez.



Legenda: As curvas de nível representam a função objetivo. A região delimitada pelas restrições g_1 , g_2 e g_3 representam a região factível do problema.

Fonte: O autor, 2024.

Na Tabela 2, é possível notar que os valores ótimos obtidos para a função de média efetiva utilizando a metodologia proposta diferem no máximo 0,086021% da abordagem tradicional. Isso indica que a metodologia proposta pode ser eficaz em encontrar soluções robustas para o problema, economizando no número de avaliações da função objetivo.

Na Figura 25, é possível observar a região factível do problema, as curvas de nível da função objetivo e os pontos ótimos robustos encontrados para as 100 execuções do algoritmo genético com e sem o uso de metamodelos, revelando que a metamodelagem não provocou alterações significativas nas coordenadas dos pontos ótimos encontrados.

5.1.1 Estruturas de Treliças

Os dois problemas que são apresentados nesta subseção envolvem a otimização das áreas das seções transversais de estruturas de treliças, visando reduzir a massa total da estrutura, respeitando as restrições de tensão e deslocamento. A implementação da análise de elementos finitos foi construída a partir do trabalho de Ferreira e Fantuzzi (2020).

Ambos os problemas podem ser caracterizados como:

$$\begin{aligned}
 \text{Minimize}_{\mathbf{A}} \quad & f(\mathbf{A}) = \sum_{i=1}^N \rho A_i l_i \\
 \text{Sujeito a} \quad & \\
 & \sigma_{i \min} \leq \sigma_i \leq \sigma_{i \max}, i = 1, \dots, N \\
 & d_{j \max} \leq d_j \leq d_{j \max}, j = 1, \dots, M \\
 & A_{i \min} \leq A_i \leq A_{i \max}, i = 1, \dots, N.
 \end{aligned} \tag{87}$$

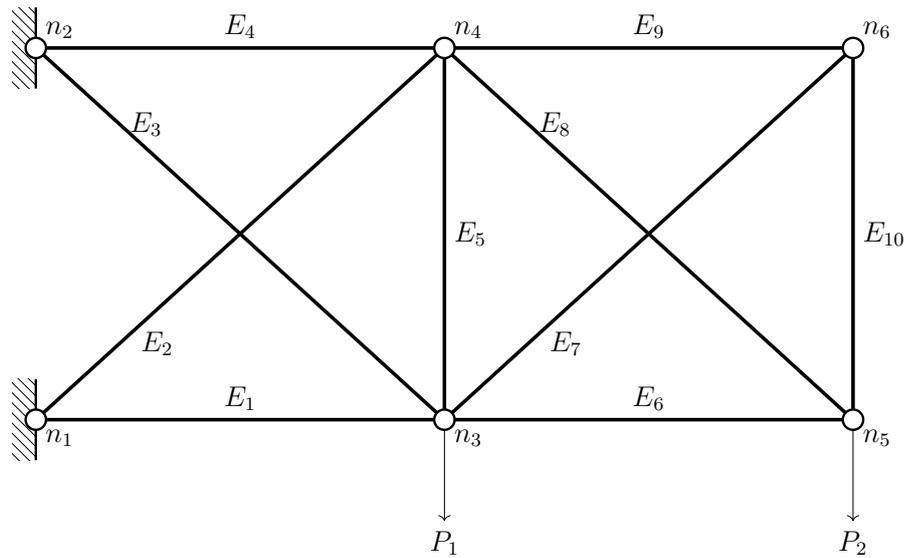
onde A_i é a área da seção transversal da haste i , l_i é o comprimento da haste i , ρ é a densidade do material da haste, σ_i é a tensão na haste i , d_j é o deslocamento do nó j .

Os dois problemas apresentam diferenças tanto na quantidade de variáveis quanto no formato da estrutura: enquanto o primeiro considera uma estrutura 2D composta por 10 hastes, o segundo aborda uma estrutura 3D mais complexa, com 25 hastes.

Problema 2 (Estruturas de treliças de 10 hastes) O primeiro problema envolve uma estrutura de treliça, composta por 10 hastes e 6 nós, conforme ilustrado na Figura 26, caracterizando um problema de 10 variáveis de projeto.

Os parâmetros de entrada incluem o módulo de elasticidade, estabelecido em $6,8948 \times 10^{10}$ Pa, e a densidade do material, fixada em $\rho = 2,7680 \times 10^3$ kg/m³ para todos os elementos estruturais. A tensão máxima permitida, tanto em compressão quanto em tração, é de $1,7237 \times 10^8$ Pa. O deslocamento máximo aceitável é de 50,8 mm, e as hastes, tanto horizontais quanto verticais, possuem um comprimento de 9,1440 m. A estrutura enfrenta uma carga aplicada de $4,4482 \times 10^5$ N nos nós n_3 e n_5 , conforme a Figura 26. As

Figura 26 - Estrutura de treliça de 10 hastes.



Fonte: Adaptado de Renkavieski e Parpinelli (2021).

áreas mínima e máxima da seção transversal para cada elemento são de $6,4516 \times 10^1 \text{ mm}^2$ e $2,2581 \times 10^4 \text{ mm}^2$, respectivamente.

Tabela 3 - Avaliações da função objetivo para a convergência do algoritmo genético para diferentes níveis de robustez.

Nível de Robustez	Usando Metamodelagem			Variação percentual entre as médias
	Sem o uso de Metamodelagem			
	Intervalo	Média	Desvio Padrão (C.V.)	
1%	$(2,6462 \times 10^5, 2,4902 \times 10^6)$	$6,6270 \times 10^5$	$4,3609 \times 10^5$ (65,80%)	73,12%
	$(1,2875 \times 10^6, 6,2500 \times 10^6)$	$2,4651 \times 10^6$	$8,6066 \times 10^5$ (34,91%)	
5%	$(2,6157 \times 10^5, 1,6928 \times 10^6)$	$6,0196 \times 10^5$	$2,6804 \times 10^5$ (44,52%)	73,14%
	$(1,0775 \times 10^6, 8,1175 \times 10^6)$	$2,2412 \times 10^6$	$1,0362 \times 10^6$ (46,23%)	
10%	$(2,5958 \times 10^5, 1,4064 \times 10^6)$	$6,3823 \times 10^5$	$2,5336 \times 10^5$ (39,69%)	65,95%
	$(9,3250 \times 10^5, 4,4875 \times 10^6)$	$1,8747 \times 10^6$	$7,5103 \times 10^5$ (40,06%)	

Fonte: O autor, 2024.

Na análise de robustez, sem a utilização de metamodelagem, foram utilizados 50 pontos de amostragem para o cálculo da função de média efetiva, onde foram utilizados Algoritmos Genéticos (GA), com o tamanho da população constante também de 50 indivíduos e 50 descendentes. Os parâmetros definidos foram: $G_{max} = 11000$, $p_m = 0,8$, $p_c = 0,8$. Foi estabelecido um critério de parada baseado em uma tolerância de 10^{-6} para a diferença entre os melhores indivíduos e de 10^{-6} para a função objetivo ao longo de 200 gerações.

Na metodologia proposta, que envolve uma fase exploratória (metamodelagem), e uma fase de refinamento, foram utilizados também Algoritmos Genéticos com os mesmos parâmetros da abordagem convencional, mas foram realizadas algumas alterações:

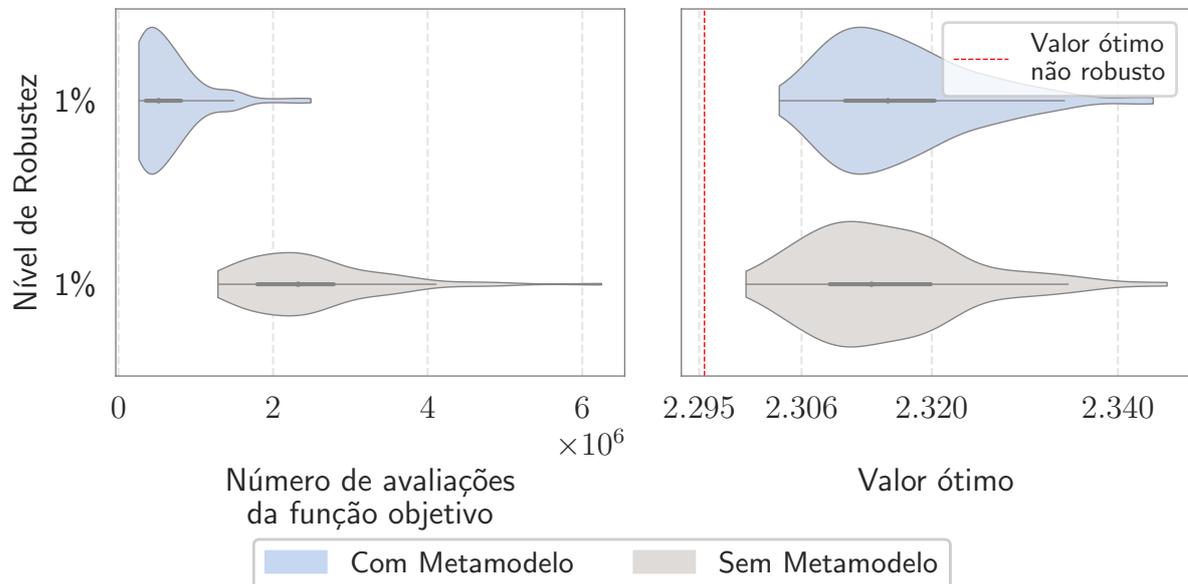
Tabela 4 - Valores ótimos obtidos para o uso do algoritmo genético para diferentes níveis de robustez. O valor ótimo sem considerar robustez é, aproximadamente, 2295,56.

Nível de Robustez	Usando Metamodelagem			Variação percentual entre as médias
	Sem o uso de Metamodelagem			
	Intervalo	Média	Desvio Padrão (C.V.)	
1%	(2303,6055; 2343,7936)	2316,6378	7,8888 (0,32%)	0,066%
	(2300,0300; 2345,2965)	2315,1014	8,6251 (0,37%)	
5%	(2306,3008; 2378,5386)	2331,1228	13,4803 (0,58%)	0,40%
	(2308,0805; 2416,3559)	2340,4788	22,8971 (0,97%)	
10%	(2323,2066; 2452,7243)	2358,6904	26,9444 (1,14%)	0,85%
	(2321,8706; 2483,8184)	2379,0819	30,6521 (1,28%)	

Fonte: O autor, 2024.

estipulou-se um período de estabilidade para o valor mínimo da função objetivo de 100 gerações para cada fase (exploratória e de refinamento). Importante ressaltar que, além dessa modificação, todos os demais parâmetros permaneceram idênticos aos do algoritmo genético convencional. Ambas as metodologias foram submetidas a 100 execuções independentes.

Figura 27 - Distribuição do número de avaliações e dos valores da função objetivo para 1% de robustez.

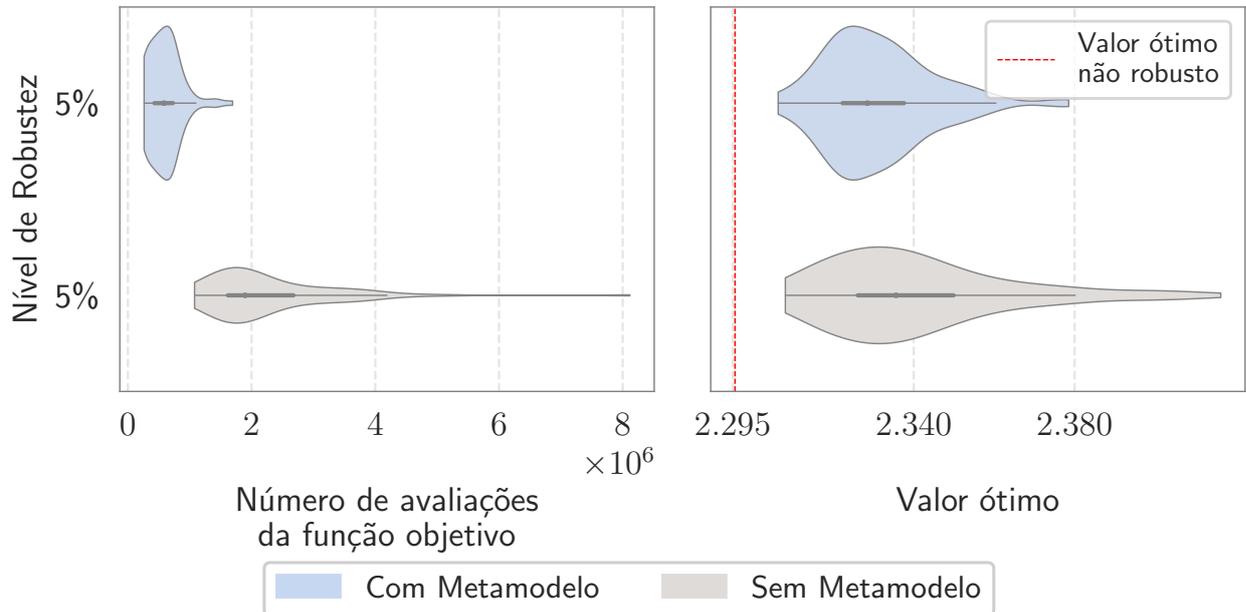


Fonte: O autor, 2024.

Na Tabela 3, são apresentados os resultados em número de avaliações da função objetivo para diferentes níveis de robustez. A economia de avaliações da função objetivo foi de pelo menos 65,95% para os diferentes níveis de robustez, o que configura um indicativo da possível efetividade da metodologia proposta.

Apesar da redução do número de avaliações da função objetivo, não houve di-

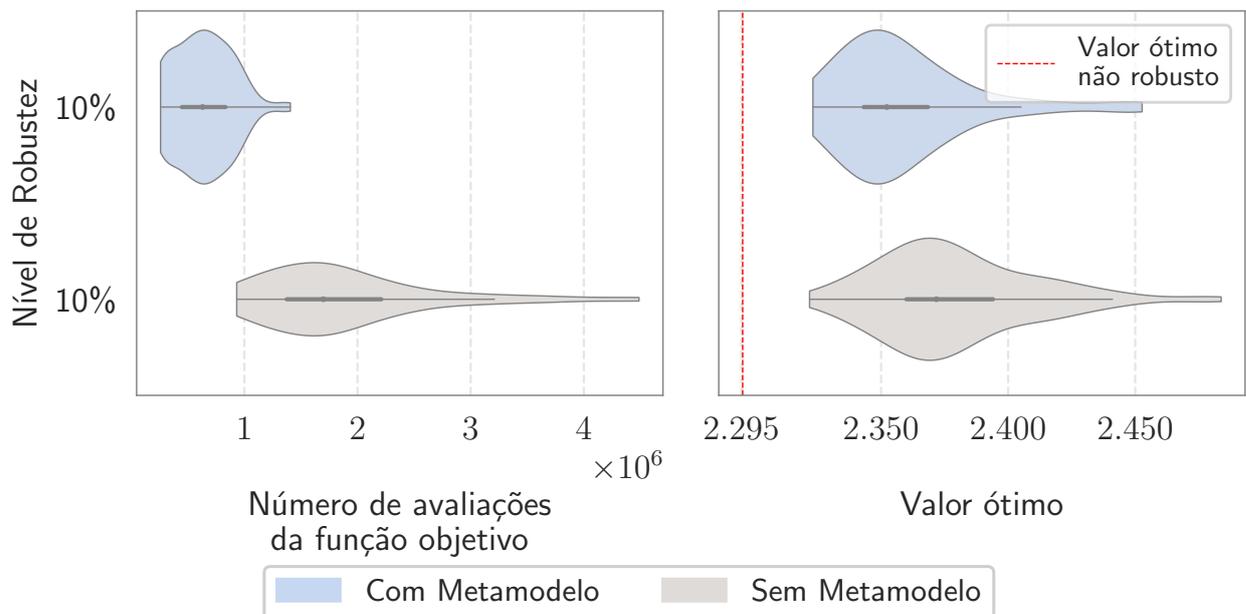
Figura 28 - Distribuição do número de avaliações e dos valores da função objetivo para 5% de robustez.



Fonte: O autor, 2024.

ferenças significativas entre as médias dos valores ótimos obtidos com e sem o uso de metamodelagem, conforme é possível notar na Tabela 4. A diferença entre as médias dos valores obtidos foi de, no máximo, 0,85%, o que indica que a metodologia é capaz de encontrar soluções robustas para o problema, economizando no número de avaliações da função objetivo.

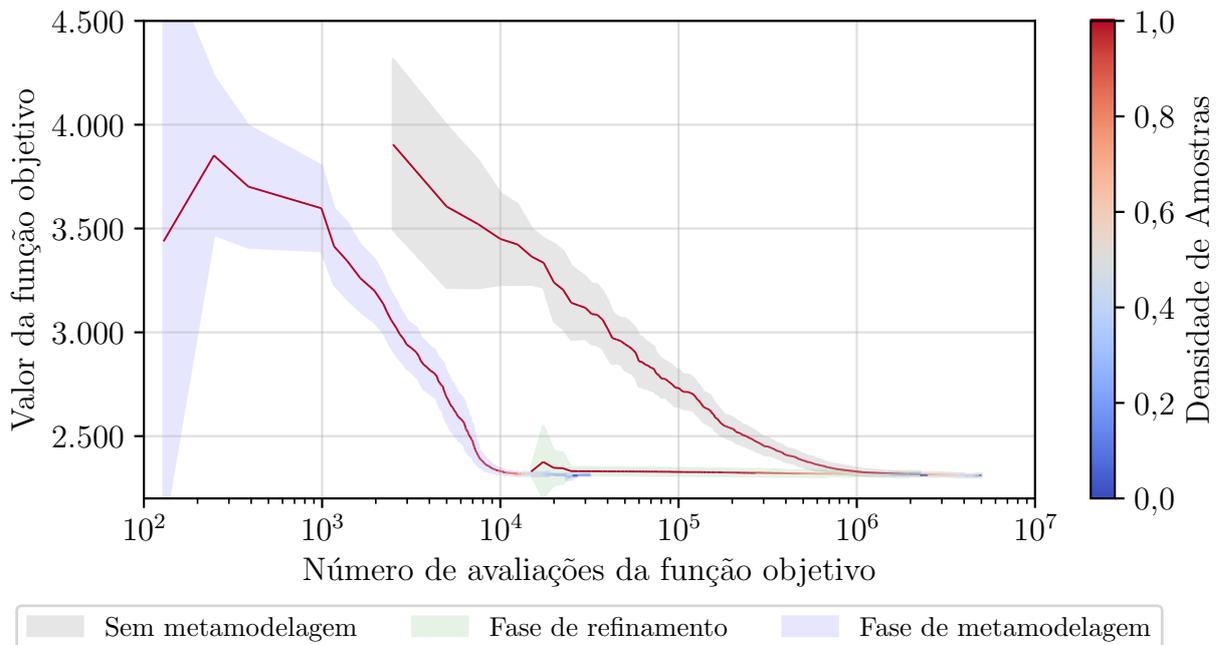
Figura 29 - Distribuição do número de avaliações e dos valores da função objetivo para 10% de robustez.



Fonte: O autor, 2024.

A análise das Figuras 27–29, revela que a distribuição dos valores da função objetivo e do número de avaliações para diferentes níveis de robustez (10%, 5% e 1%), baseada em 100 execuções, possui uma tendência de redução na dispersão desses valores ao aplicar a metamodelagem, mantendo-se a qualidade dos resultados.

Figura 30 - Evolução do valor da função objetivo em função do número de avaliações para cem execuções e com 1% de robustez.



Legenda: A área sombreada representa o desvio padrão e a linha central o valor médio da função objetivo para 100 execuções.

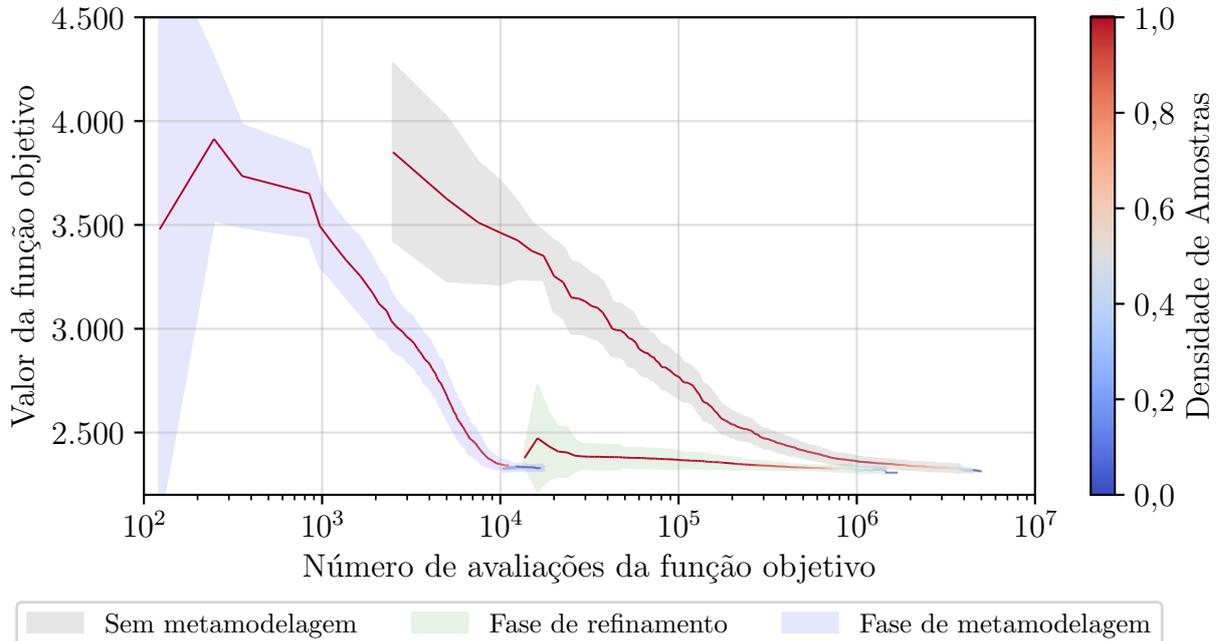
Fonte: O autor, 2024.

Nas Figuras 30–32, são apresentadas as evoluções médias do valor ótimo da função objetivo em relação ao número de avaliações a cada geração. Esses dados novamente são baseados em 100 execuções e consideram diferentes níveis de robustez (10%, 5% e 1%). Os gráficos foram construídos utilizando um histograma, baseado no número médio de avaliações da função objetivo por geração (classes), e no valor médio da função objetivo para cada classe.

É importante notar que o critério de parada pode ser atingido em diferentes gerações e quantidades de avaliações. Isso resulta em uma variação na quantidade de amostras utilizadas para calcular o valor médio do número de avaliações da função objetivo. Essa variação é representada pela barra indicadora de mapa de calor nas Figuras 30–32, indicando a densidade de amostras utilizadas para o cálculo do valor médio.

Ainda acerca das Figuras 30–32, nota-se uma descontinuidade nos gráficos na transição da fase de metamodelagem para a fase de refinamento. Isso ocorre porque, durante a metamodelagem, a estimativa do valor da função objetivo pode ser imprecisa devido à natureza aproximada do metamodelo.

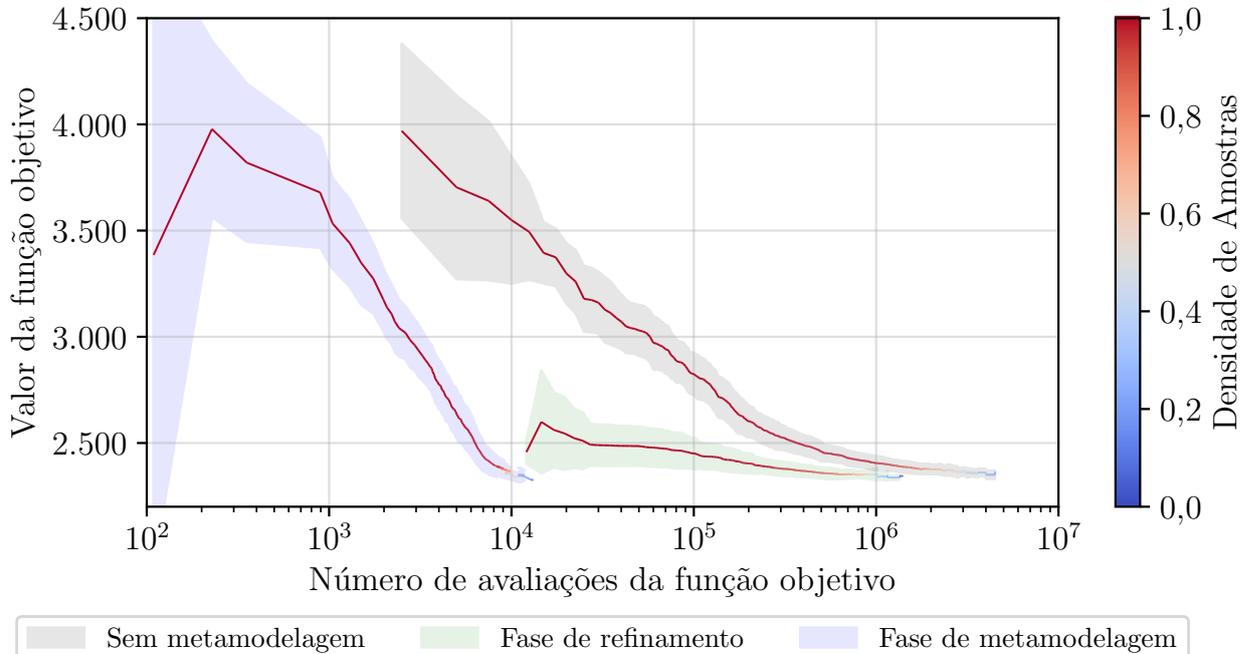
Figura 31 - Evolução do valor da função objetivo em função do número de avaliações para cem execuções e com 5% de robustez.



Legenda: A área sombreada representa o desvio padrão e a linha central o valor médio da função objetivo para 100 execuções.

Fonte: O autor, 2024.

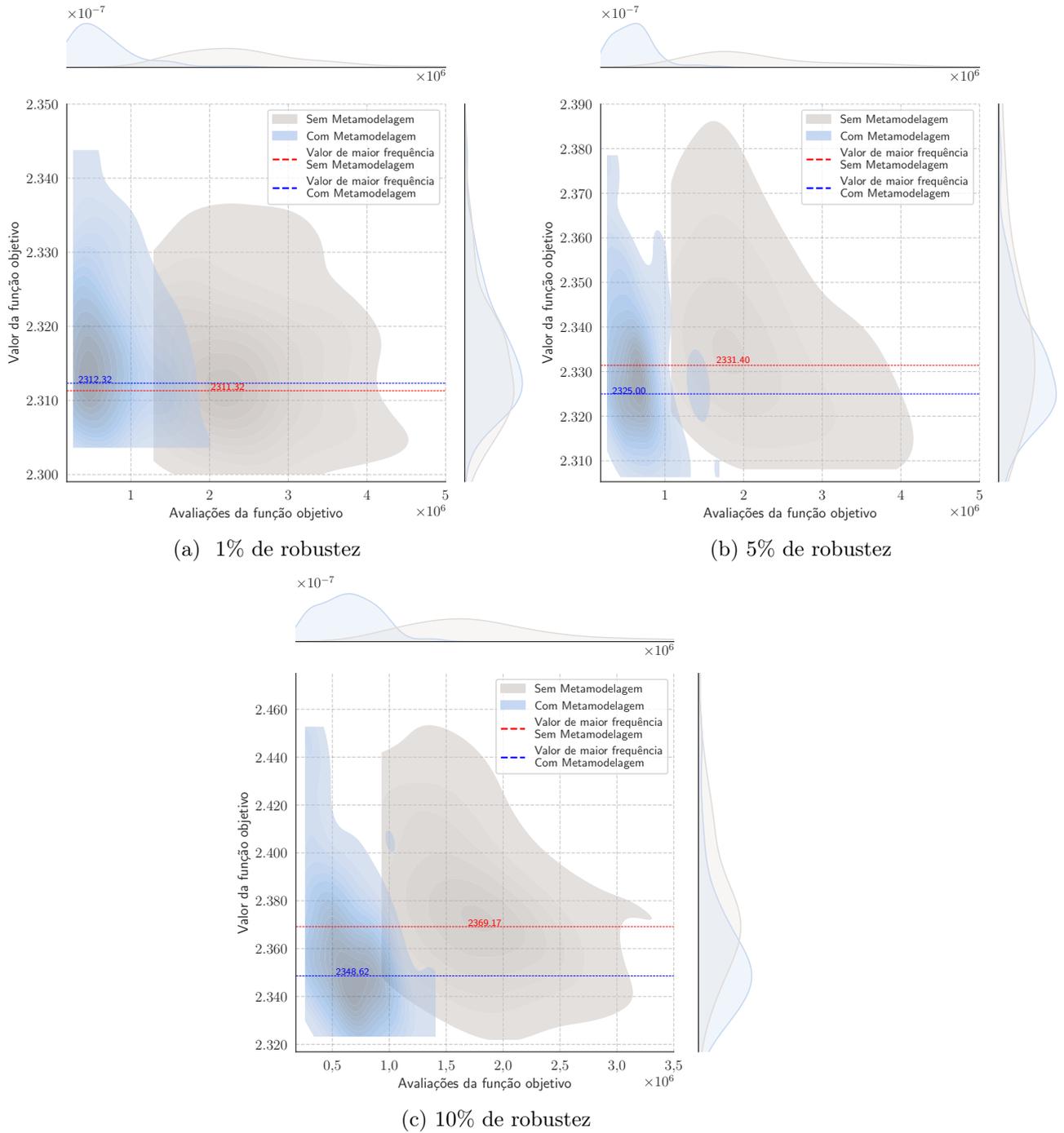
Figura 32 - Evolução do valor da função objetivo em função do número de avaliações para cem execuções e com 10% de robustez.



Legenda: A área sombreada representa o desvio padrão e a linha central o valor médio da função objetivo para 100 execuções.

Fonte: O autor, 2024.

Figura 33 - Estimativas da distribuição conjunta entre o número de avaliações da função objetivo e o valor da função objetivo para diferentes níveis de robustez.



Fonte: O autor, 2024.

No entanto, na fase de refinamento, essa imprecisão é corrigida, pois nesta etapa abandona-se a metamodelagem e passa-se a avaliar a função objetivo diretamente, permitindo a obtenção dos valores mais precisos para a função de média efetiva. Isso evidencia a importância da fase de refinamento e justifica o motivo pelo qual o metamodelo não é usado para a obtenção do valor ótimo final. Além disso, o comportamento dos gráficos

não se altera, mesmo com diferentes níveis de robustez.

Uma análise mais aprofundada do comportamento dos valores da função objetivo em relação ao número de avaliações nas Figuras 30–32, revela que a metodologia que utiliza metamodelagem inicia o processo de otimização com um número consideravelmente menor de avaliações da função objetivo. Essa diferença inicial ocorre porque, sem o uso da metamodelagem, são necessárias 2000 avaliações iniciais (uma população de 100 indivíduos com 20 amostras ao redor de cada um) para calcular a função de média efetiva. Em contraste, a abordagem com metamodelagem requer inicialmente apenas 100 avaliações para a construção do metamodelo.

As Figuras 33a–33c ilustram como a metamodelagem, como uma abordagem de redução de complexidade, permite uma exploração mais eficiente do espaço de soluções, evidenciado pela tendência de se obter soluções robustas com um número menor de avaliações da função objetivo, o que sugere uma economia significativa de recursos e tempo, a depender da complexidade da função objetivo.

Problema 3 (Estrutura de treliças de 25 hastes) Considere uma estrutura composta por 25 elementos de treliça, ilustrada na Figura 34. O objetivo principal novamente é minimizar a massa total da estrutura, de acordo com as inequações dadas em (87), levando em consideração as restrições máximas de tensão e compressão dos elementos, conforme especificado na Tabela 5. A estrutura deve ser projetada para suportar, de maneira independente, duas diferentes situações de cargas que podem atuar nos nós da estrutura. Essas situações estão descritas na Tabela 6.

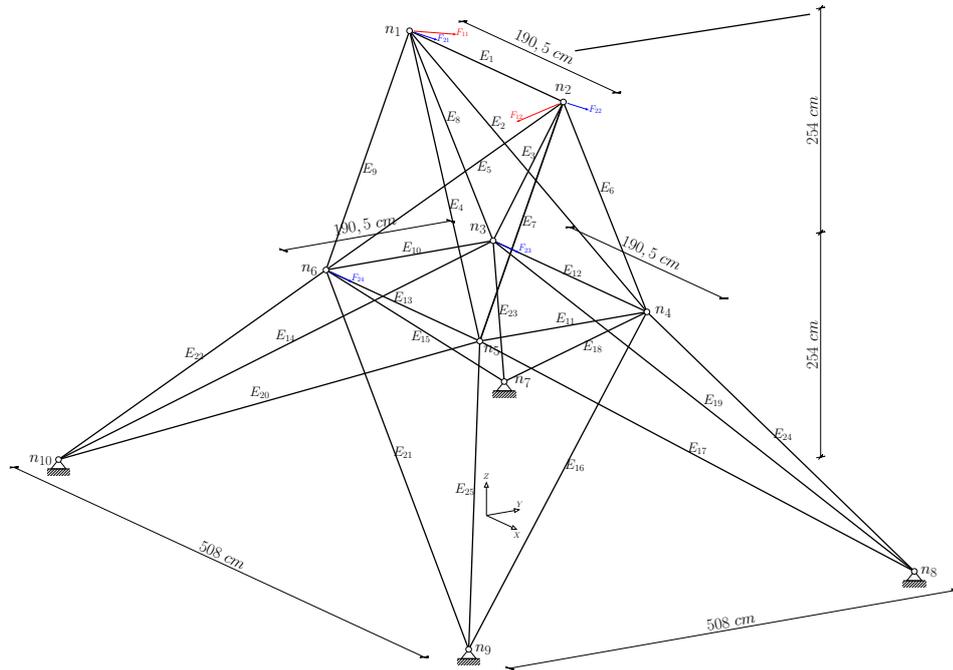
Tabela 5 - Limites de tensão e compressão para a treliça de 25 barras.

Variável	Elemento	Limite de tensão de compressão (MPa)	Limite de tensão (MPa)
1	E_1	$2,4195 \times 10^2$	275,79
2	$E_2 - E_5$	$7,9910 \times 10^1$	275,79
3	$E_6 - E_9$	$1,1933 \times 10^2$	275,79
4	$E_{10} - E_{11}$	$2,4195 \times 10^2$	275,79
5	$E_{12} - E_{13}$	$2,4195 \times 10^2$	275,79
6	$E_{14} - E_{17}$	$4,6602 \times 10^1$	275,79
7	$E_{18} - E_{21}$	$4,6602 \times 10^1$	275,79
8	$E_{22} - E_{25}$	$8,1372 \times 10^1$	275,79

Fonte: Adil e Cengiz (2019)

Para esta estrutura, o módulo de Young foi considerado de $6,8948 \times 10^4$ MPa e a densidade do material é de 2768 kg/m^3 . As áreas mínimas e máximas das seções transversais das hastes foram fixadas em $6,4516 \text{ mm}^2$ e $2,1935 \times 10^3 \text{ mm}^2$, respectivamente. A Tabela 5 também apresenta a relação entre as variáveis de projeto e os elementos da

Figura 34 - Estrutura de treliça com 25 hastes.



Fonte: Adaptado de Adil e Cengiz (2019).

Tabela 6 - Casos de carga para treliça de 25 barras.

Nó	Caso 1 (KN)			Caso 2 (KN)		
	F_x	F_y	F_z	F_x	F_y	F_z
n_1	$F_{11} = 0,0$	88,9644	-22,2411	$F_{21} = 4,4482$	44,4822	-22,2411
n_2	$F_{12} = 0,0$	-88,9644	-22,2411	$F_{22} = 0,0$	44,4822	-22,2411
n_3	0,0	0,0	0,0	$F_{23} = 2,2241$	0,0	0,0
n_6	0,0	0,0	0,0	$F_{24} = 2,2241$	0,0	0,0

Fonte: Adil e Cengiz (2019)

Tabela 7 - Avaliações da função objetivo para a convergência do algoritmo genético para diferentes níveis de robustez.

Nível de Robustez	Usando Metamodelagem			Variação percentual entre as médias
	Sem o uso de Metamodelagem			
	Intervalo	Média	Desvio Padrão (C.V.)	
1%	$(7,2620 \times 10^4, 5,9452 \times 10^5)$	$2,7855 \times 10^5$	$1,3151 \times 10^5$ (47,21%)	59,94%
	$(3,1500 \times 10^5, 1,2960 \times 10^6)$	$6,9543 \times 10^5$	$2,1636 \times 10^5$ (31,11%)	
5%	$(7,6566 \times 10^4, 5,8368 \times 10^5)$	$2,0912 \times 10^5$	$9,7938 \times 10^4$ (46,83%)	64,10%
	$(2,5500 \times 10^5, 1,3395 \times 10^6)$	$5,8263 \times 10^5$	$2,1076 \times 10^5$ (36,17%)	
10%	$(9,0093 \times 10^4, 6,2696 \times 10^5)$	$2,9382 \times 10^5$	$1,2482 \times 10^5$ (42,48%)	56,97%
	$(2,6850 \times 10^5, 1,4910 \times 10^6)$	$6,8283 \times 10^5$	$2,3123 \times 10^5$ (33,86%)	

Fonte: O autor, 2024.

Tabela 8 - Valores ótimos obtidos para o uso do algoritmo genético para diferentes níveis de robustez. O valor ótimo sem considerar robustez é, aproximadamente, 247,56.

Nível de Robustez	Usando Metamodelagem			Variação percentual entre as médias
	Sem o uso de Metamodelagem			
	Intervalo	Média	Desvio Padrão (C.V.)	
1%	(248,8941; 258,2202)	250,9232	1,5518 (0,61%)	0,11%
	(248,8963; 259,4327)	251,2019	1,8447 (0,73%)	
5%	(255,1895; 264,8243)	258,4860	2,0231 (0,78%)	0,22%
	(255,2502; 268,1126)	257,8986	1,8922 (0,73%)	
10%	(263,7268; 282,6440)	267,4826	3,1492 (1,17%)	0,03%
	(263,1172; 276,2690)	267,4003	2,8620 (1,07%)	

Fonte: O autor, 2024.

estrutura.

No trabalho apresentado por Adil e Cengiz (2019), a solução com estes parâmetros, desconsiderando a robustez, determina uma massa total para a estrutura de aproximadamente $2,4728 \times 10^2$ kg ($5,4516 \times 10^2$ lb).

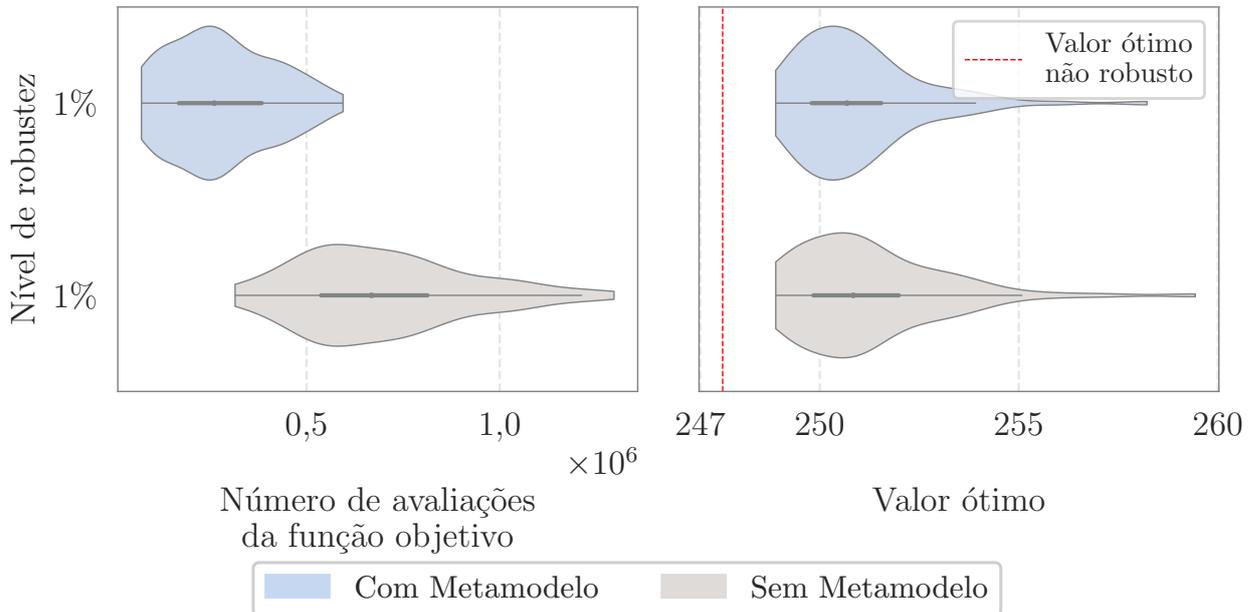
A metodologia adotada para a análise de robustez é a mesma que foi aplicada no problema anterior, com o uso de Algoritmos Genéticos e 100 execuções independentes. No entanto, neste caso, foi estabelecida uma população fixa de 30 indivíduos, 30 descendentes e um total de 11000 gerações, com uma probabilidade de mutação e cruzamento de 0,8 e 50 amostras para o cálculo da robustez. O critério de parada é baseado na estabilidade da melhor solução, com uma tolerância de 10^{-6} tanto para a diferença entre os melhores indivíduos quanto para a função objetivo ao longo de períodos pré-definidos. Para o algoritmo genético sem metamodelagem, foi definido um período de estabilidade de 70 gerações. Por outro lado, para o algoritmo com o acréscimo de metamodelagem, foi estabelecida 35 gerações de estabilidade para cada fase, tanto exploratória quanto de refinamento.

A Tabela 7 destaca a economia nas avaliações da função objetivo, que foi de no mínimo 56,97% para os diferentes níveis de robustez. Já a Tabela 8 mostra os valores ótimos alcançados tanto com quanto sem o uso de metamodelagem. A diferença máxima entre as médias dos valores obtidos foi de apenas 0,85%, sugerindo que a metodologia pode ser eficaz em encontrar soluções robustas, ao mesmo tempo que economiza no número de avaliações da função objetivo.

Embora o valor médio da função objetivo apresente um ligeiro incremento em certos resultados, tal diferença torna-se pouco significativa ao se considerar a economia no número médio de avaliações da função objetivo, fator que justifica a aplicação da metamodelagem.

As Figuras 35–36b ilustram as distribuições tanto do número de avaliações quanto

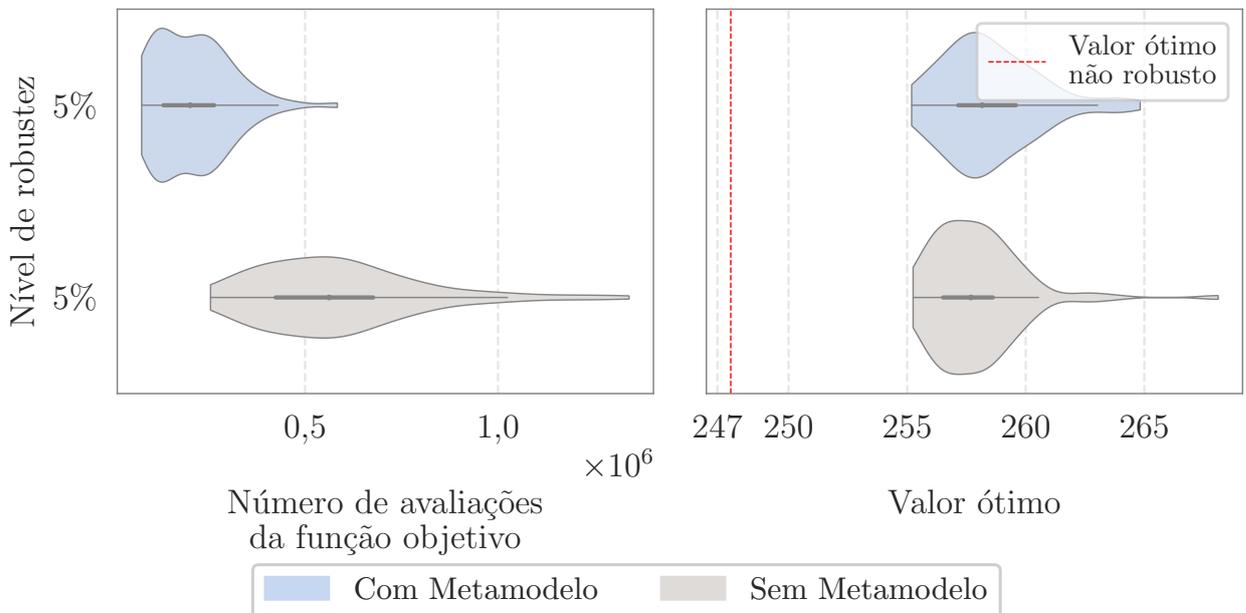
Figura 35 - Distribuição do número de avaliações e dos valores da função objetivo para 1% de robustez.



Fonte: O autor, 2024.

dos valores da função objetivo para vários graus de robustez. É interessante observar que a diminuição no número de avaliações da função objetivo não comprometeu a qualidade dos resultados alcançados. Na verdade, os resultados foram extremamente semelhantes aos obtidos sem a aplicação de metamodelagem.

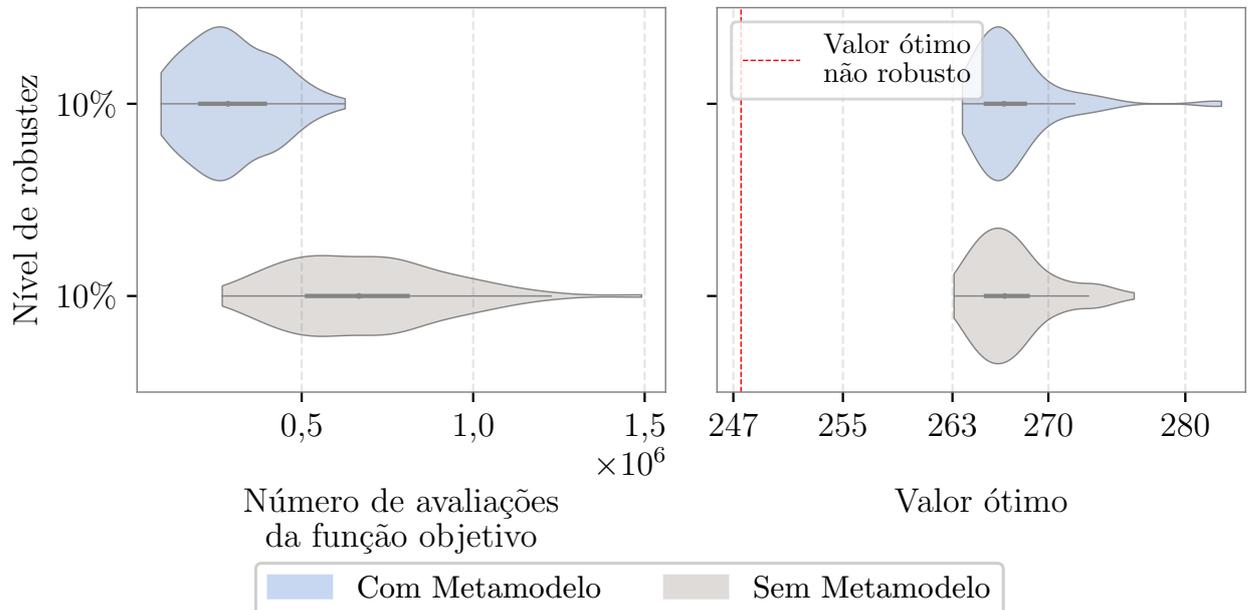
Figura 36 - Distribuição do número de avaliações e dos valores da função objetivo para diferentes níveis de robustez.(continua)



(a) Distribuição do número de avaliações e dos valores da função objetivo para 5% de robustez.

Como esperado, à medida que o nível de robustez aumenta, o valor mínimo da

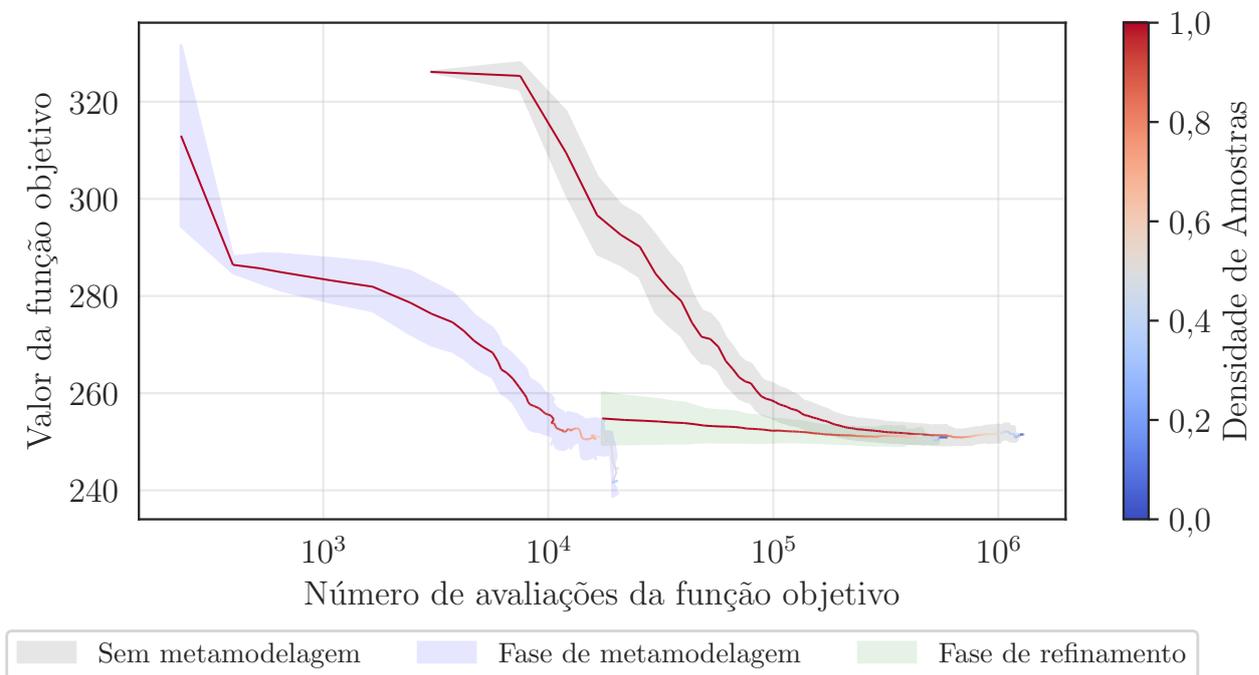
Figura 36 - Distribuição do número de avaliações e dos valores da função objetivo para diferentes níveis de robustez (continuação).



(b) Distribuição do número de avaliações e dos valores da função objetivo para 10% de robustez.

Fonte: O autor, 2024.

Figura 37 - Evolução das avaliações e da função objetivo para 1% de robustez.

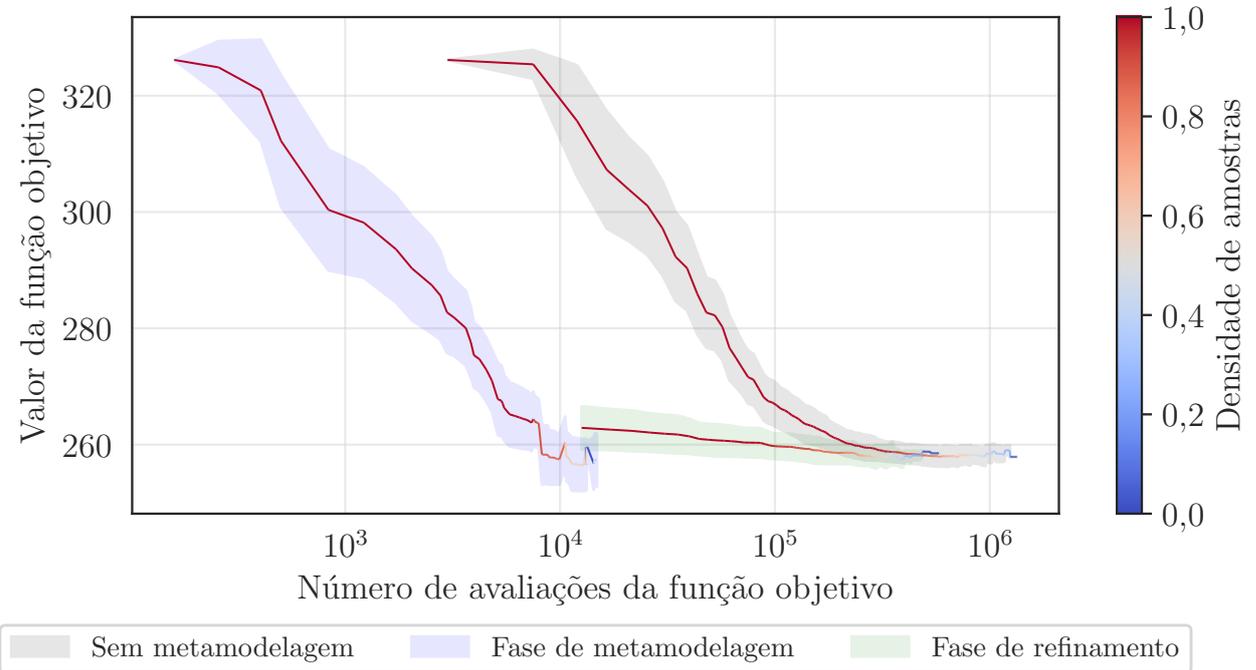


Legenda: A área sombreada representa o desvio padrão e a linha central o valor médio da função objetivo para 100 execuções.

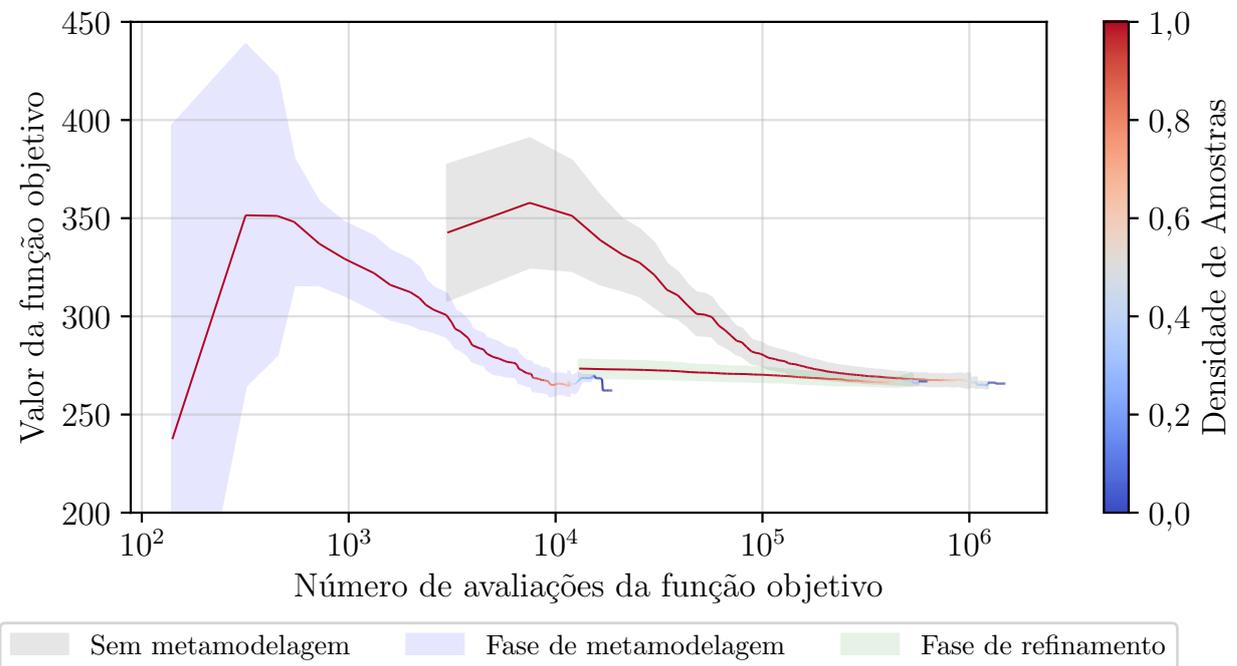
Fonte: O autor, 2024.

função objetivo também aumenta. Isso está em linha com as expectativas, pois um maior nível de robustez geralmente implica em uma solução mais conservadora, o que pode resultar em um valor de função objetivo ligeiramente maior. No entanto, essa abordagem

Figura 38 - Evoluções do valor médio da função objetivo em função do valor médio do número de avaliações da função objetivo por geração para 5% e 10% de robustez.



(a) Evolução das avaliações e da função objetivo para 5% de robustez



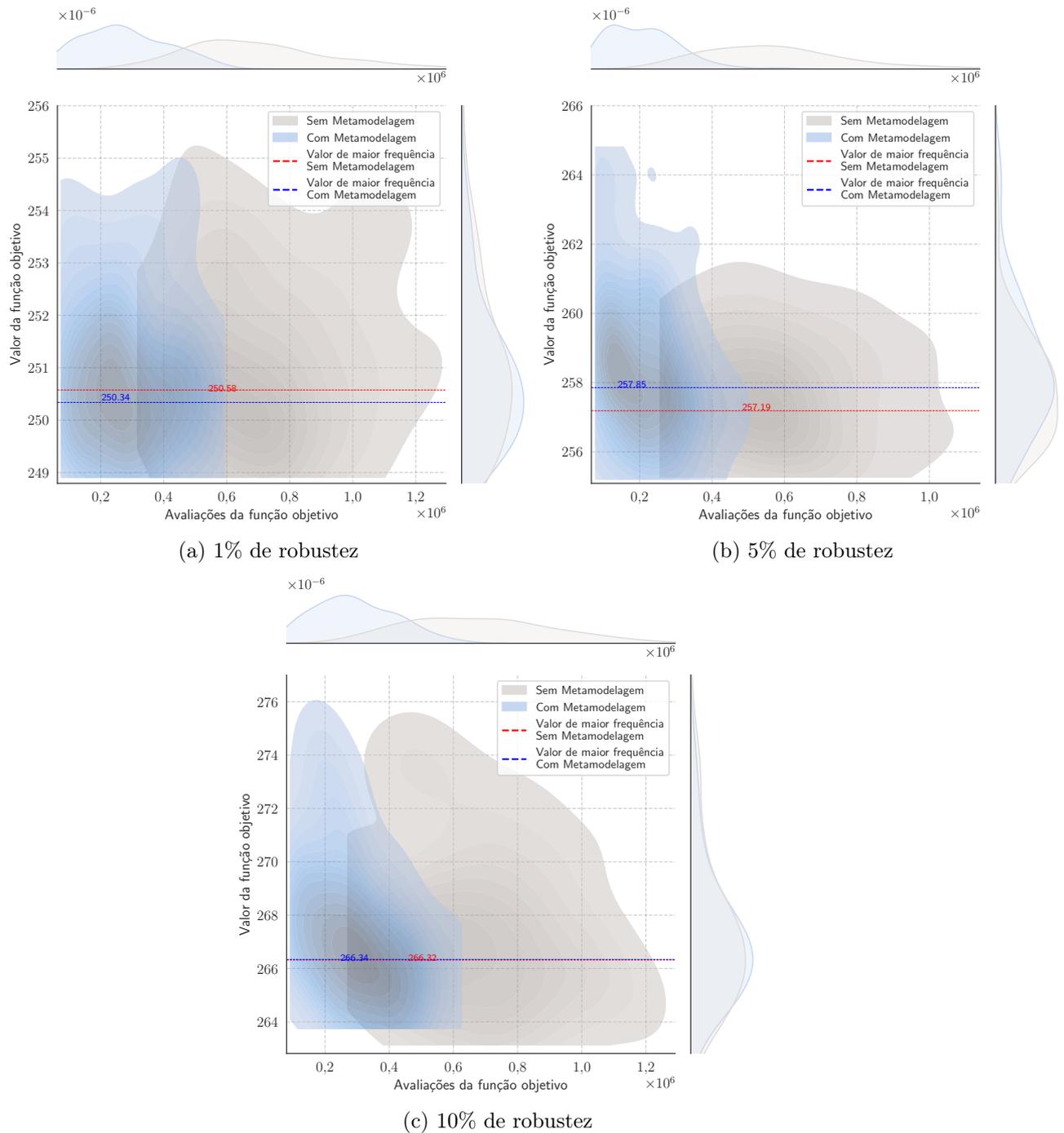
(b) Evolução das avaliações e da função objetivo para 10% de robustez

Fonte: O autor, 2024.

mais cautelosa pode proporcionar que a solução seja menos suscetível a grandes variações no valor da função objetivo.

As Figuras 37, 38a e 37b ilustram as mudanças no número de avaliações e nos

Figura 38 - Estimativas da distribuição conjunta entre o número de avaliações da função objetivo e o valor da função objetivo para diferentes níveis de robustez.



Fonte: O autor, 2024.

valores da função objetivo para vários graus de robustez. Nota-se que conforme o número de avaliações cresce, o valor da função objetivo começa a se estabilizar, indicando a convergência para uma solução ótima robusta.

Além disso, pode-se inferir que a aplicação da metamodelagem acelera esse processo de estabilização, permitindo assim alcançar a convergência com um número menor de

avaliações da função objetivo e mantendo a qualidade da solução.

Já nas Figuras 38a–38c, são apresentadas as estimativas das distribuições conjuntas do número de avaliações e dos valores da função objetivo para diferentes níveis de robustez. É possível visualizar que o uso da metamodelagem supera em eficiência a abordagem tradicional, o que indica maior probabilidade de economia no número de avaliações da função objetivo, podendo inclusive obter soluções mais refinadas.

5.2 Confiabilidade

Nesta subseção, são examinados três problemas com o objetivo de verificar a eficácia da projeção estereográfica integrada ao ASOSL na determinação do MPP, em comparação com os resultados disponíveis na literatura.

Inicialmente, dois problemas numéricos são avaliados, nos quais se busca apenas localizar os respectivos MPP. Em seguida, o primeiro problema de otimização estrutural previamente discutido na seção 5.1 é reanalisado com a modificação nas restrições para uma restrição probabilística, com o objetivo de alcançar uma solução ótima que considere exclusivamente a confiabilidade, desconsiderando a análise de robustez. Este exame é realizado através do método SORA, com a utilização do algoritmo genético como metaheurística populacional.

5.2.1 Análise de confiabilidade

Considere dois problemas utilizados na literatura (Pericaro et al., 2015; Borri; Speranzini, 1997) para avaliar o ponto mais provável de falha:

$$h_1(\mathbf{x}) = 0,1(x_1 - x_2)^2 - \frac{x_1 + x_2}{\sqrt{2}} + 2,5 \quad (88)$$

$$h_2(\mathbf{x}) = x_1^3 + x_2^3 - 18, \quad (89)$$

As variáveis na Equação (88) são assumidas como seguindo uma distribuição normal padrão, e o índice de confiabilidade alvo fixado é $\beta_t = 2,5$. Para a Equação (89), assume-se uma distribuição normal com $\mu_1 = \mu_2 = 10$ e $\sigma_1 = \sigma_2 = 5$ e índice de confiabilidade alvo $\beta_t \approx 2,2409$.

Após realizar a transformação de Rosenblatt, as curvas de nível de ambas as funções podem ser visualizadas nas Figuras 39a e 39b, respectivamente. O objetivo é localizar sobre a hipersfera o ponto mais provável de falha para cada uma das funções, fixando os valores de β_t .

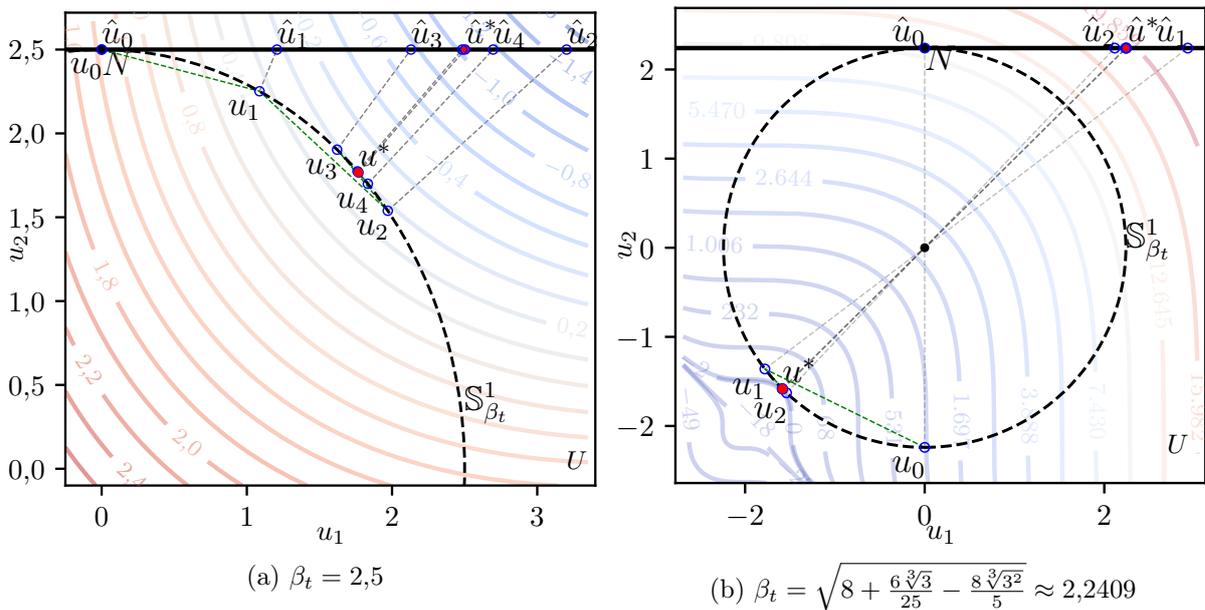
Na Tabela 9, são apresentadas as coordenadas encontradas para o ponto mais provável de falha u^* no espaço normal padrão, o valor de $G(u^*)$ e o número de iterações

Tabela 9 - Comparação entre solução exata e os resultados obtidos com o ASOSL com a projeção estereográfica.

Função de desempenho	Solução Exata		Método proposto		iter. n
	β_t	Ponto de Projeto (u_i)	Ponto de projeto (u^*)	$G(u^*)$	
Eq. (88)	2,500	(1,7677; 1,7677)	(1,7677; 1,7677)	$5,0274 \times 10^{-11}$	11
Eq. (89)	2,2409	(-1,5839; -1,5839)	(-1,5839; -1,5839)	$1,4147 \times 10^{-9}$	6

Fonte: O autor, 2024.

Figura 39 - Sequência de pontos no espaço normal padrão convergindo para o MPP.



Legenda: Nas Figuras 39a e 39b são exibidas as seqüências de pontos relativas à convergência para o ponto mais provável de falha relativa às equações (88) e (89), respectivamente. Fonte: O autor, 2024.

para a convergência dos problemas dados pelas equações (88) e (89). Os parâmetros fixados para o ASOSL em ambos os problemas foram: $\alpha = 0,1$, $\rho = 0,5$ e $\delta_\eta = 10^{-6}$. Como critério de parada, foi adotada a norma do vetor gradiente $\|p_k\| < 10^{-5}$. Nas Figuras 39a e 39b, são ilustrados os resultados obtidos, no tocante à seqüência de pontos resultantes de cada iteração até a convergência.

Ao analisar os dados da Tabela 9, verifica-se que a integração do ASOSL com a projeção estereográfica possibilitou a determinação do ponto mais provável de falha em comparação com a solução exata.

Problema 4 (Estrutura de treliças de 10 hastes, com Confiabilidade) Ao retomarmos o problema de engenharia estrutural apresentado na subseção 5.1.1, a intenção é ajustar as restrições de maneira a incorporar garantias de confiabilidade. Assim, transformamos o problema original em um problema de Otimização Baseada em Confiabilidade (RBDO, do inglês *Reliability-Based Design Optimization*). Essa abordagem

é importante para garantir que a estrutura não apenas atenda aos requisitos de desempenho, mas também que atenda a parâmetros de segurança, mesmo sob incertezas nos fatores operacionais.

Com base no problema definido por Kaveh e Zaerreza (2022) considere o seguinte problema de otimização com confiabilidade:

$$\begin{aligned}
 &\underset{\mu_A}{\text{Encontrar}} && \mu_A = \{\mu_{A_1}, \mu_{A_2}, \dots, \mu_{A_{10}}\} \\
 &\underset{\mu_A}{\text{Minimizar}} && f(\mu_A) = \rho \sum_{i=1}^{10} A_i L_i \\
 &\text{Sujeito a} && Pf = P[g(\mu_A, P, E) \leq 0] \leq \Phi(-\beta^t) \\
 &\text{Onde} && g(\mu_A, P, E) = 0,0508 - |u_2^y(\mu_A, P, E)| \\
 &&& 6,4516 \times 10^{-5} \text{ m}^2 \leq \mu_A \leq 2,2581 \times 10^{-2} \text{ m}^2 \\
 &&& \beta^t = 3.0
 \end{aligned} \tag{90}$$

com os dados sobre as variáveis de projeto e demais parâmetros descritos na Tabela 10.

Tabela 10 - Dados para a simulação do problema da treliça de 10 barras presente na Figura 26.

Variável	Descrição	Média ou valor de-terminístico	Distribuição	Coefficiente de variação
A_i	Área da seção transversal	A calcular	Normal	0,05
E	Módulo da elasticidade	$6,8948 \times 10^{10}$ Pa	Normal	0,05
P	Força aplicada	$4,5359 \times 10^4$ kg	Normal	0,05
ρ	Densidade do material	$2,7680 \times 10^3$ kg/m ³	-	-

Fonte: Adaptado de Kaveh e Zaerreza (2022)

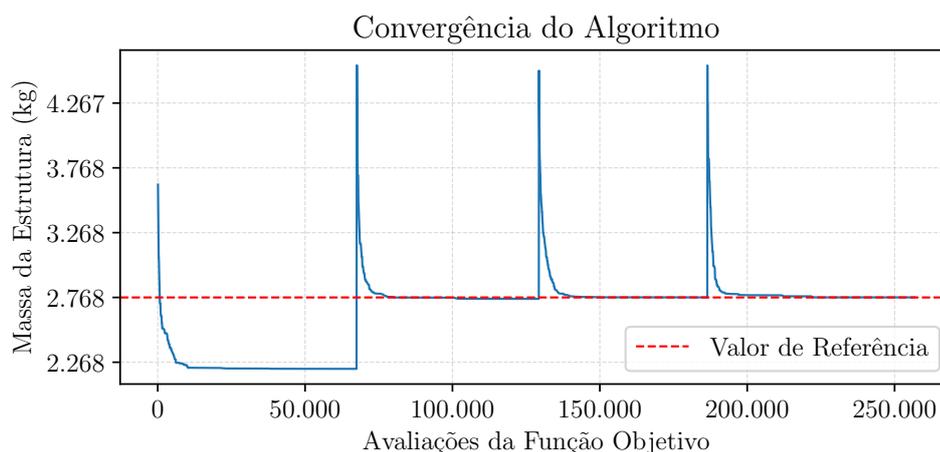
A Figura 26 ilustra a configuração da treliça, onde as forças são aplicadas nos nós n_3 e n_5 na direção y . As áreas das seções transversais dos elementos são consideradas variáveis de projeto aleatórias, enquanto as forças aplicadas e o módulo de elasticidade são tratados como parâmetros aleatórios. Os valores desses parâmetros estão detalhados na Tabela 10, sendo que o objetivo da otimização é minimizar a massa total da treliça, garantindo que a deflexão máxima no nó 5 seja inferior a $5,0800 \times 10^{-2}$ m (2 polegadas). Os demais parâmetros também podem ser visualizados na mesma tabela.

Para resolver este problema de otimização com confiabilidade, foi utilizado o método SORA, empregando um Algoritmo Genético padrão através da biblioteca python Pymoo (Blank; Deb, 2020). Inicialmente, realiza-se a otimização determinística com as restrições e parâmetros originais. Após obter a solução da primeira execução, verifica-se

a restrição probabilística, buscando-se o MPP utilizando o ASOSL com a estratégia da projeção estereográfica. Se a restrição for satisfeita, ou seja, se o ASOSL encontrar um valor positivo ou nulo para a função de desempenho G no ponto de mínimo, o algoritmo é encerrado. Caso contrário, a restrição é ajustada por meio de um vetor de deslocamento com as coordenadas do ponto de mínimo obtido na execução anterior. Este ciclo se repete até que a função de desempenho seja positiva ou nula no MPP. Para mais detalhes sobre o método SORA, consulte Du e Chen (2004), Kaveh e Zaerreza (2022).

Na Figura 40, visualizam-se os ciclos do SORA com a análise de confiabilidade realizada pelo ASOSL. Observa-se que a projeção estereográfica permitiu obter uma massa estrutural extremamente próxima ao valor de referência de outros trabalhos, conforme é possível visualizar na Tabela 11.

Figura 40 - Otimização realizada utilizando o método SORA com o ASOSL integrado à estratégia da projeção estereográfica para análise de confiabilidade inversa.



Fonte: O autor, 2024.

Na Tabela 11 é comparada a melhor solução de 30 execuções do método proposto com as melhores soluções de outros estudos. Embora o SORA-DM apresente uma solução potencialmente superior, ao se avaliar a solução indicada usando o ASOSL com a estratégia da projeção estereográfica, essa solução possui um valor negativo na função de desempenho no mais provável ponto de falha u^* , indicando possível tolerância para a restrição probabilística ou falta de precisão devido à quantidade de casas decimais disponibilizadas na referência. Em contraste, nas demais soluções, a função de desempenho é estritamente positiva, mas próxima de zero, no MPP.

Tabela 11 - Comparação dos métodos RBDO na treliça de 10 barras. O valor de $G(u^*)$ foi calculado utilizando o ASOSL para todas as soluções indicadas, sendo o critério de parada a norma do gradiente ser menor do que 10^{-6}

Variável de Projeto (m ²)	NDL-GA (Shayanfar; Abbasnia; Khodam, 2014)	NDL-IDE (Khodam et al., 2021)	ESORA-IDE (Khodam et al., 2021)	SORA-DM (Kaveh; Zaerreza, 2022)	Presente Estudo
A_1	$2,2163 \times 10^{-2}$	$2,2580 \times 10^{-2}$	$2,2580 \times 10^{-2}$	$2,2574 \times 10^{-2}$	$2,2569 \times 10^{-2}$
A_2	$6,4516 \times 10^{-5}$	$6,4516 \times 10^{-5}$	$6,4516 \times 10^{-5}$	$6,7548 \times 10^{-5}$	$6,4516 \times 10^{-5}$
A_3	$1,9150 \times 10^{-2}$	$1,7941 \times 10^{-2}$	$1,8031 \times 10^{-2}$	$1,7607 \times 10^{-2}$	$1,8240 \times 10^{-2}$
A_4	$1,6950 \times 10^{-2}$	$1,2835 \times 10^{-2}$	$1,2792 \times 10^{-2}$	$1,2632 \times 10^{-2}$	$1,2836 \times 10^{-2}$
A_5	$6,4516 \times 10^{-5}$	$6,4516 \times 10^{-5}$	$6,4516 \times 10^{-5}$	$6,4581 \times 10^{-5}$	$6,4516 \times 10^{-5}$
A_6	$6,4516 \times 10^{-5}$	$6,4516 \times 10^{-5}$	$6,4516 \times 10^{-5}$	$6,4839 \times 10^{-5}$	$6,4516 \times 10^{-5}$
A_7	$2,1529 \times 10^{-3}$	$2,2163 \times 10^{-3}$	$2,1890 \times 10^{-3}$	$2,1663 \times 10^{-3}$	$2,2435 \times 10^{-3}$
A_8	$1,8293 \times 10^{-2}$	$1,8844 \times 10^{-2}$	$1,8832 \times 10^{-2}$	$1,8952 \times 10^{-2}$	$1,8474 \times 10^{-2}$
A_9	$1,6863 \times 10^{-2}$	$1,8336 \times 10^{-2}$	$1,8341 \times 10^{-2}$	$1,8290 \times 10^{-2}$	$1,8488 \times 10^{-2}$
A_{10}	$6,4516 \times 10^{-5}$	$6,4516 \times 10^{-5}$	$6,4516 \times 10^{-5}$	$6,4516 \times 10^{-5}$	$6,4516 \times 10^{-5}$
$G(u^*)$	$2,3968 \times 10^{-4}$	$6,7142 \times 10^{-7}$	$9,9357 \times 10^{-6}$	$-2,4394 \times 10^{-4}$	$6,6041 \times 10^{-7}$
Massa (kg)	$2,8174 \times 10^3$	$2,7678 \times 10^3$	$2,7678 \times 10^3$	$2,7546 \times 10^3$	$2,7683 \times 10^3$
Massa média (kg)	-	-	-	$2,7603 \times 10^3$	$2,7697 \times 10^3$
Desv. padrão	-	-	-	4,7219	$8,6183 \times 10^{-1}$

Fonte: O autor, 2024.

5.3 Robustez e Confiabilidade

Esta seção discute os resultados obtidos pela aplicação de metamodelagem, combinando otimização baseada em robustez e análise de confiabilidade, utilizando o ASOSL em conjunto com a projeção estereográfica. Inicialmente, o Problema 1 é modificado para incorporar a confiabilidade como um objetivo a ser maximizado. Ou seja, o problema é ajustado para ser tratado como um problema RBRDO, onde as amostras utilizadas no cálculo da função de média efetiva são avaliadas em termos de confiabilidade em cada geração da execução do algoritmo NSGA-II.

Posteriormente, o problema de engenharia associado à treliça de 10 barras, discutido anteriormente, é resolvido considerando-se simultaneamente robustez e confiabilidade. Para tal finalidade, emprega-se o método SORA, no qual a análise de confiabilidade é realizada de forma desacoplada.

Problema 5 Considere o problema de otimização levando em consideração robustez e confiabilidade, descrito por:

$$\begin{aligned}
 & \underset{\mathbf{d}}{\operatorname{argmin}} && f(\mathbf{d}, X, \delta) \\
 & \max && \beta_t \\
 \text{sujeito a} &&& P[g_i(\mathbf{d}, X, \delta) \leq 0] \leq \Phi(-\beta_t), \quad i = 1, 2, 3, \\
 &&& -14 \leq d_1, d_2 \leq 0, \\
 &&& 1 \leq \beta_t \leq 4,
 \end{aligned} \tag{91}$$

onde

$$\begin{aligned}
 f(\mathbf{x}) &= \exp\left(\frac{1}{100}(\bar{x}_1 + \bar{x}_2)^2\right) - 0,8 \exp(-6,25((\bar{x}_1 + 2)^2 + (\bar{x}_2 + 1)^2)) \\
 g_1(\mathbf{x}) &= (\bar{x}_1^2 + \bar{x}_2^2 + 2r_1\bar{x}_1)^2 - 4r_1^2(\bar{x}_1^2 + \bar{x}_2^2) + 2 \geq 0 \\
 g_2(\mathbf{x}) &= (\bar{x}_1^2 + \bar{x}_2^2)^3 - 4r_2\bar{x}_1^2\bar{x}_2^2 \geq 0 \\
 g_3(\mathbf{x}) &= ((\bar{x}_1 + 1)^2 + \bar{x}_2^2 - 2r_3(\bar{x}_1 + 1))^2 + 4r_3^2((\bar{x}_1 + 1)^2 + \bar{x}_2^2) - 2 \geq 0 \\
 &- 14 \leq x_1, x_2 \leq 0,
 \end{aligned}$$

com $r_1 = 1,2054$, $r_2 = 1,6936 \times 10^1$, $r_3 = 1,2054$, $f : \mathbb{R}^2 \rightarrow \mathbb{R}$ e $\mathbf{x} = (\bar{x}_1, \bar{x}_2) = (\frac{x_1}{4}, \frac{x_2}{4})$.

Nos resultados previamente discutidos, o procedimento de seleção das amostras para serem incluídas no conjunto de treinamento e atualização do metamodelo, em cada geração, foi conduzido aleatoriamente dentre os descendentes obtidos com a metaheurística populacional. Todavia, em substituição ao método aleatório, pode-se considerar uma opção alternativa que envolve a utilização de funções de aquisição que foram abordadas na seção 3.3, o que constitui a aprendizagem ativa (*active learning*).

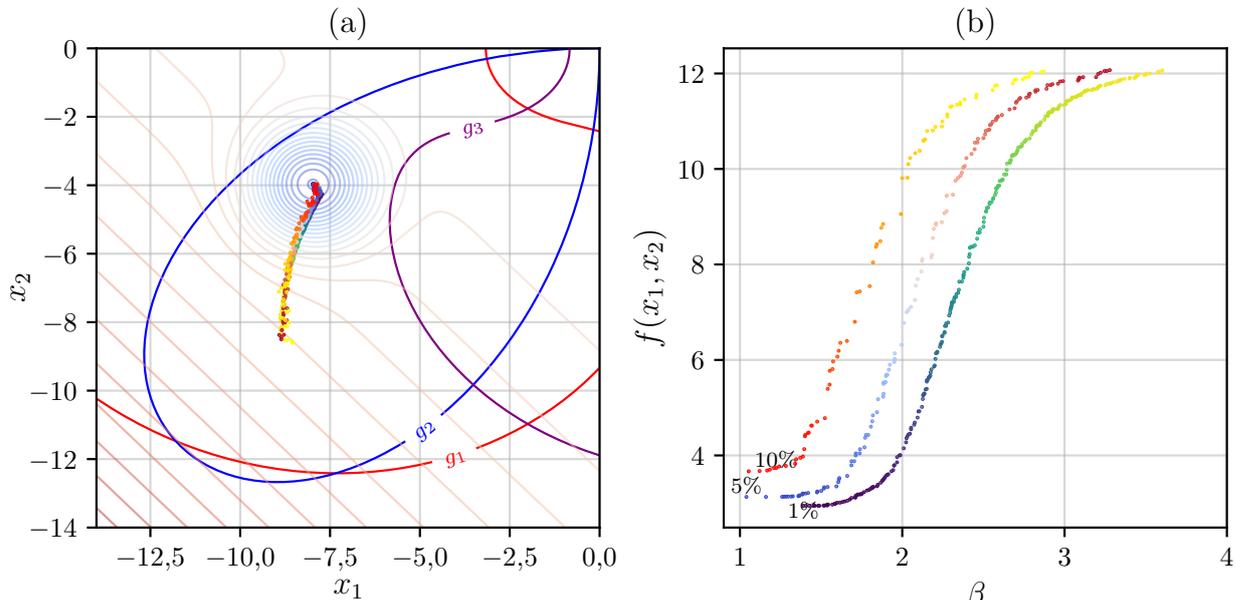
Os resultados apresentados a seguir utilizaram a função de aquisição melhoria esperada (EI) para selecionar 20% dos novos indivíduos, gerados pela metaheurística, para o conjunto de treinamento e atualização, conforme ilustrado no fluxograma 24. A população do algoritmo NSGA-II foi fixada em 100 indivíduos, com 20 descendentes, e as taxas de mutação e *crossover* foram definidas em 0,1 e 0,8, respectivamente, adotando como critério de parada a estabilidade no valor médio dos objetivos no espaço objetivo por 100 gerações, com uma tolerância de 0,005 tanto para a fase de metamodelagem quanto para a fase de refinamento.

No que se refere à integração do ASOSL com a estratégia de projeção estereográfica para a análise de confiabilidade adotando a abordagem PMA, a taxa de aprendizagem foi estabelecida em $s = 0.8$. Como critério de parada, adotou-se o valor da norma do gradiente ser inferior a 10^{-6} ou a diferença relativa absoluta entre as imagens das funções de desempenho em pontos consecutivos ser inferior a 10^{-12} , o que ocorrer primeiro. As soluções obtidas são exibidas na Figura 41.

Com o aumento do nível de robustez, observa-se uma crescente dificuldade na obtenção de soluções que mantenham o correspondente índice de confiabilidade, devido ao fato de que as amostras utilizadas no cálculo da função de média efetiva se aproximam cada vez mais das restrições impostas pelo problema. Isto se reflete na diminuição da densidade de pontos na fronteira de Pareto para 10% de robustez, por exemplo. Para a execução sem o uso de metamodelagem, os parâmetros foram os mesmos citados, porém o critério de parada foi fixado em um período de estabilidade de 200 gerações. Para a função de média efetiva e a análise de confiabilidade, foram utilizadas 20 amostras em ambas as abordagens.

Para a obtenção da fronteira de Pareto aproximada, foram conduzidas 30 execuções

Figura 41 - Fronteiras e conjuntos de Pareto obtidos com o uso do NSGA-II para diferentes níveis de robustez (1%, 5% e 10%).



Legenda: (a) Região factível do problema, com os respectivos conjuntos de Pareto. (b) Fronteiras de Pareto com a influência da análise de confiabilidade para diferentes níveis de robustez.

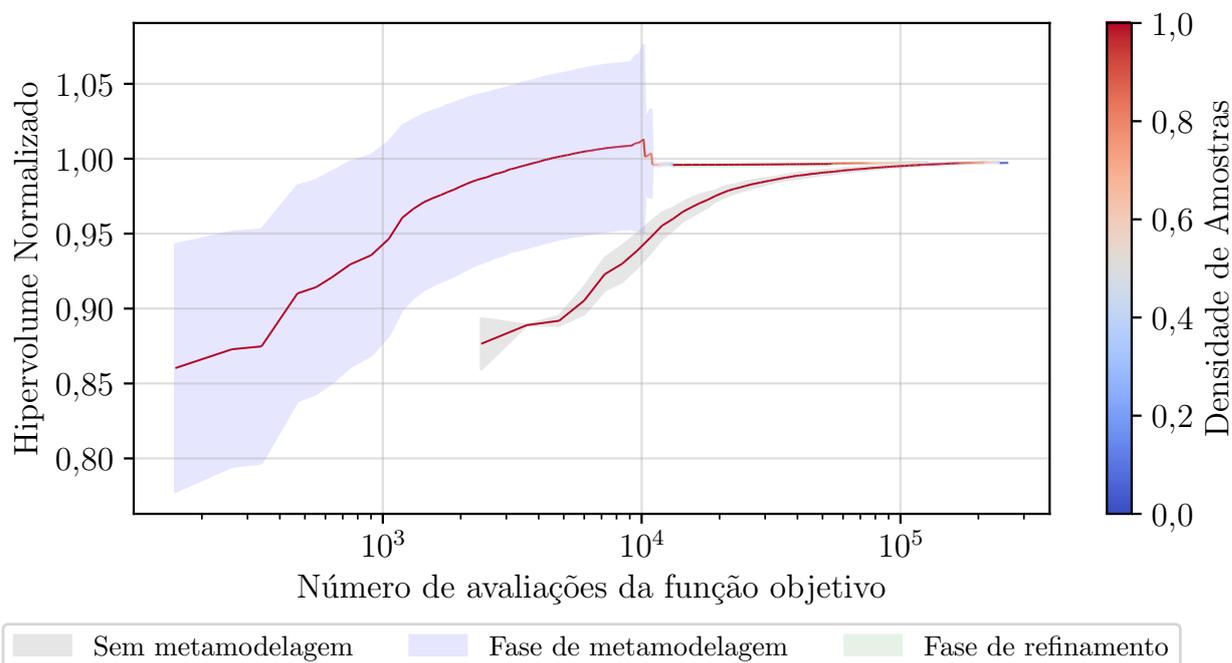
Fonte: O autor, 2024.

para cada nível de robustez (1%, 5% e 10%). Após estas execuções, as soluções não dominadas de cada execução, para cada nível de robustez, foram agregadas a fim de constituir uma aproximação da fronteira de Pareto. Após a definição dessas fronteiras, o algoritmo NSGA-II foi novamente executado 30 vezes, porém realizou-se o cálculo do hipervolume normalizado e da distância geracional em cada geração, considerando a comparação entre a fronteira de Pareto obtida na iteração e a fronteira aproximada.

As Figuras 42 e 44 ilustram a progressão dos valores do hipervolume normalizado e da distância geracional em relação ao número de vezes que a função objetivo foi avaliada por geração, considerando um nível de robustez de 1%. Nas métricas de convergência apresentadas, a utilização do metamodelo resultou em valores extremamente próximos aos alcançados sem o uso de metamodelagem, ao mesmo tempo em que exigiu um número consideravelmente reduzido de avaliações da função objetivo. Isso é corroborado pela distribuição de valores para o número total de avaliações da função objetivo e hipervolume apresentados na Figura 43.

Conforme pode ser observado nas tabelas 12 e 13, que exibem os valores finais referentes ao total de avaliações da função objetivo e ao cálculo do hipervolume para as 30 execuções do experimento, a aplicação da técnica de metamodelagem não resultou em alterações significativas no valor final do hipervolume quando comparada à ausência dessa estratégia, mesmo adicionando-se aprendizagem ativa ao processo de treinamento. No entanto, a utilização da metamodelagem proporcionou uma redução significativa no

Figura 42 - Evolução do número de avaliações da função objetivo e do valor do hipervolume normalizado para 1% de robustez.



Legenda: A região sombreada e a curva central correspondem, respectivamente, ao desvio padrão e à média do hipervolume.

Fonte: O autor, 2024.

número de avaliações da função objetivo, alcançando uma diminuição de pelo menos 48,72%.

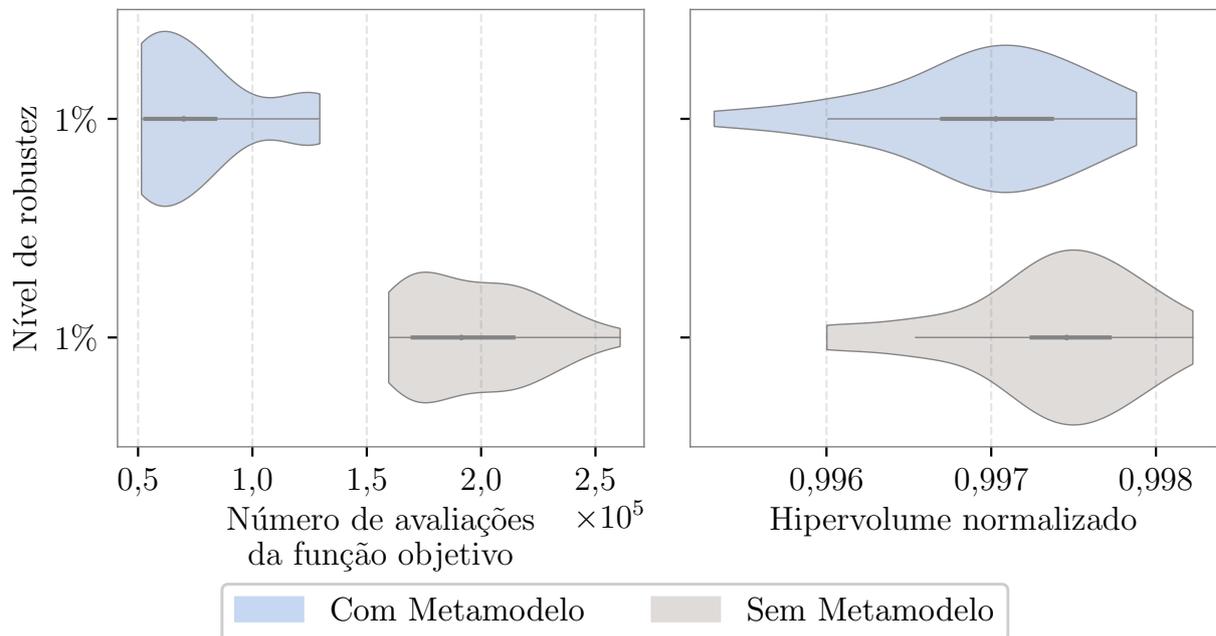
Tabela 12 - Número de avaliações da função objetivo para com e sem metamodelagem para diferentes níveis de robustez.

Nível de Robustez	Usando Metamodelagem			Variação percentual entre as médias
	Sem o uso de Metamodelagem			
	Intervalo	Média	Desvio Padrão (C.V.)	
1%	$(5,1481 \times 10^4, 1,2946 \times 10^5)$	$7,6117 \times 10^4$	$2,6029 \times 10^4$ (34,19%)	61,31%
	$(1,5960 \times 10^5, 2,6080 \times 10^5)$	$1,9677 \times 10^5$	$2,7136 \times 10^4$ (13,79%)	
5%	$(5,1783 \times 10^4, 1,8024 \times 10^5)$	$9,9596 \times 10^4$	$2,8016 \times 10^4$ (28,12%)	54,04%
	$(1,6000 \times 10^5, 3,3600 \times 10^5)$	$2,1671 \times 10^5$	$4,1350 \times 10^4$ (19,08%)	
10%	$(5,1515 \times 10^4, 1,8012 \times 10^5)$	$1,1775 \times 10^5$	$3,4520 \times 10^4$ (29,31%)	48,72%
	$(1,6840 \times 10^5, 2,9600 \times 10^5)$	$2,2964 \times 10^5$	$3,9396 \times 10^4$ (17,15%)	

Fonte: O autor, 2024.

Embora as Figuras 42, 43 e 44 apresentem dados para o caso específico de uma robustez de 1%, é importante notar que o comportamento para níveis de robustez de 5%

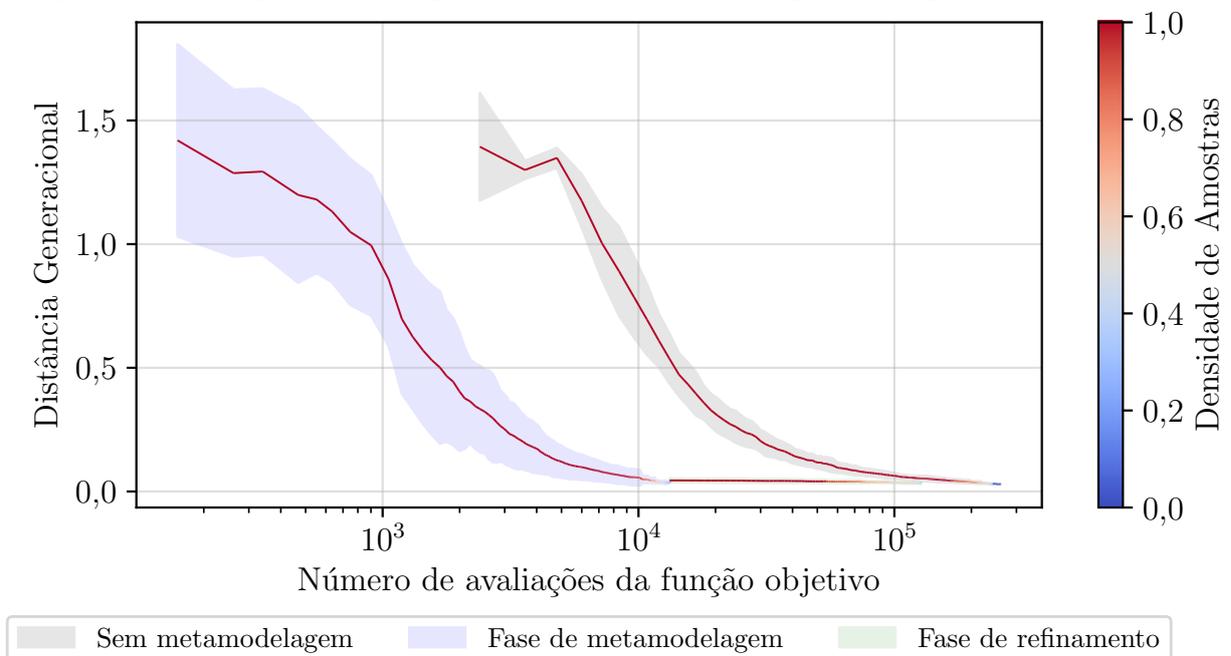
Figura 43 - Distribuição do hipervolume normalizado para 1% de Robustez usando aprendizagem ativa.



Legenda: A região sombreada e a curva central correspondem, respectivamente, ao desvio padrão e à média do hipervolume.

Fonte: O autor, 2024.

Figura 44 - Evolução das avaliações e do valor da distância geracional para 1% de robustez.



Legenda: A área sombreada e a curva central representam, respectivamente, o desvio padrão e a média da GD.

Fonte: O autor, 2024.

e 10% mostra uma elevada semelhança, como é demonstrado pelos valores nas tabelas 12 e 13. Essa semelhança entre os níveis de robustez analisados indica que a inclusão de figuras adicionais para os casos de 5% e 10% de robustez no corpo principal do texto não é necessária para a compreensão dos resultados discutidos.

Tabela 13 - Hipervolume para diferentes níveis de robustez.

Nível de Robustez	Usando Metamodelagem			Variação percentual entre as médias
	Sem o uso de Metamodelagem			
	Intervalo	Média	Desvio Padrão (C.V.)	
1%	(0,9953; 0,9979)	0,9969	0,0006 (0,06%)	0,04%
	(0,9960; 0,9982)	0,9974	0,0005 (0,05%)	
5%	(0,9948; 1,0020)	0,9961	0,0013 (0,12%)	0,04%
	(0,9944; 0,9974)	0,9957	0,0007 (0,07%)	
10%	(0,9904; 1,0134)	0,9937	0,0039 (0,39%)	0,01%
	(0,9914; 0,9961)	0,9938	0,0011 (0,11%)	

Fonte: O autor, 2024.

Problema 6 Seja $f(A) = \rho \sum_{i=1}^{10} A_i L_i$, onde $A = \{A_1, A_2, \dots, A_{10}\}$ denota as áreas das seções transversais das hastes na estrutura de treliças mostrada na Figura 45 e os comprimentos das hastes são dados pelo conjunto $L = \{L_1, L_2, \dots, L_{10}\}$. O parâmetro ρ representa a densidade do material. Sem levar em consideração robustez e confiabilidade, o objetivo é a minimização da massa da estrutura, mantendo o deslocamento do nó n_5 inferior ou igual a $5,0800 \times 10^{-2}$ m. Este problema de otimização pode ser formulado como:

Minimizar $f(A)$

Sujeito a $g(A, P, E) = |u_2^y(A, P, E)| - 5,0800 \times 10^{-2} \leq 0$

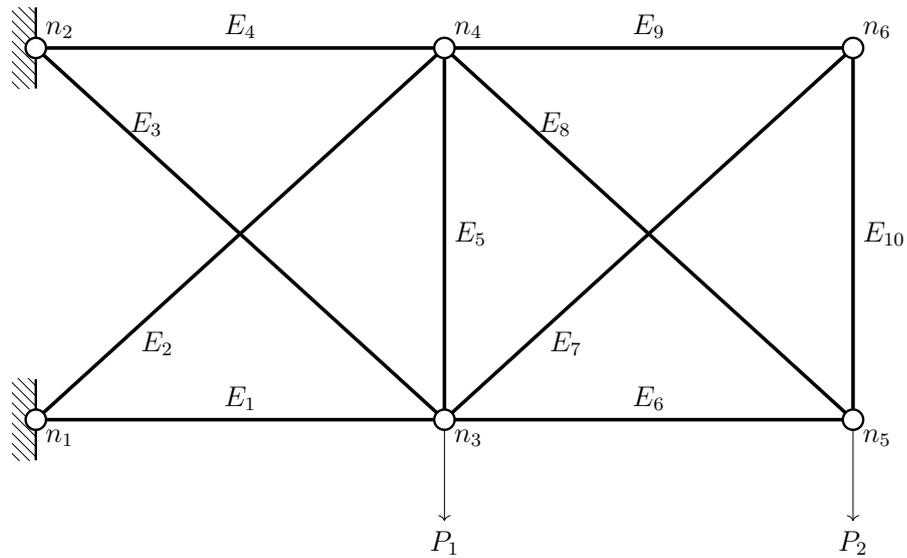
$$6,4516 \times 10^{-5} \text{ m}^2 \leq A_i \leq 2,2581 \times 10^{-2} \text{ m}^2, \quad i = 1, 2, \dots, 10$$

Os valores do módulo de elasticidade (E) e das cargas aplicadas P_1 e P_2 estão enumerados na Tabela 10.

Dada a solução apresentada na Tabela 14, obtem-se que $f(A^*) \approx 2215,3900$ kg como valor obtido sem levar em consideração robustez e confiabilidade.

Ao incorporar elementos de robustez e confiabilidade ao problema por meio de uma abordagem de Otimização Baseada em Projeto Robusto e Confiável (RBRDO), o

Figura 45 - A estrutura de treliça composta por 10 hastes.



Fonte: Adaptado de Renkavieski e Parpinelli (2021).

Tabela 14 - Solução Determinística para o Problema para o problema da telha de 10 barras.

A_1	A_2	A_3	A_4	A_5
$2,0672 \times 10^{-2} \text{ m}^2$	$6,4516 \times 10^{-5} \text{ m}^2$	$1,3561 \times 10^{-2} \text{ m}^2$	$9,8234 \times 10^{-3} \text{ m}^2$	$6,4516 \times 10^{-5} \text{ m}^2$
A_6	A_7	A_8	A_9	A_{10}
$6,4516 \times 10^{-5} \text{ m}^2$	$2,0901 \times 10^{-3} \text{ m}^2$	$1,4401 \times 10^{-2} \text{ m}^2$	$1,4047 \times 10^{-2} \text{ m}^2$	$6,4516 \times 10^{-5} \text{ m}^2$

Fonte: O autor, 2024.

problema pode ser reformulado como:

$$\text{Determinar}_{\mu_A} \quad \mu_A = \{\mu_{A_1}, \mu_{A_2}, \dots, \mu_{A_{10}}\}$$

$$\text{Minimizar}_{\mu_A} \quad f(\mu_A, \delta)$$

$$\text{Sob a condição de que} \quad P_f = P[g(\mu_A, P, E) \leq 0] \leq \Phi(-\beta^t)$$

$$\text{Onde} \quad g(\mu_A, P, E) = 0,0508 - |u_2^y(\mu_A, P, E)|$$

$$6,4516 \times 10^{-5} \text{ m}^2 \leq \mu_A \leq 2,2581 \times 10^{-2} \text{ m}^2$$

Para abordar o problema de robustez e confiabilidade, foi empregado o método SORA, em 30 execuções independentes. Este método foi implementado considerando diferentes índices de confiabilidade, especificamente $\beta_t = 1,0$, $2,0$ e $3,0$, utilizando a estratégia de projeção estereográfica e níveis de robustez de 1%, 5% e 10%. O algoritmo genético foi utilizado com taxas de mutação e *crossover* estabelecidas em 0,2 e 0,9, respectivamente.

Como critério de parada, adotou-se a estabilidade no valor ótimo da função de média efetiva por 100 gerações durante a fase de metamodelagem e 100 gerações na fase

de refinamento. Os mesmos parâmetros de mutação e *crossover* citados anteriormente foram utilizados para obter soluções sem o uso de metamodelagem, mas com 200 gerações de estabilidade no valor ótimo da função de média efetiva.

A Tabela 15 permite visualizar uma comparação dos números de avaliações da função objetivo para diferentes níveis de robustez e confiabilidade. A aplicação de metamodelagem resultou em uma redução de pelo menos 60,20% no número de avaliações da função objetivo, mantendo a diferença em relação ao valor médio ótimo inferior a 0,95%, conforme demonstrado na Tabela 16.

Tabela 15 - Número de avaliações da função objetivo com o uso da metamodelagem para diferentes níveis de robustez e índices de confiabilidade.

Nível de Robustez	Usando Metamodelagem								
	Sem o uso de Metamodelagem								
	$\beta_t = 1$		$\beta_t = 2$		$\beta_t = 3$				
	Média	Desvio Padrão (C.V.)	Média	Desvio Padrão (C.V.)	Média	Desvio Padrão (C.V.)			
1%	67,97% Dif	$1,1445 \times 10^6$	$5,5144 \times 10^5$ (48,18%)	76,89% Dif	$9,1131 \times 10^5$	$4,6388 \times 10^5$ (50,90%)	69,09% Dif	$1,4057 \times 10^6$	$5,4456 \times 10^5$ (38,73%)
		$3,5741 \times 10^6$	$7,1895 \times 10^5$ (20,11%)		$3,9449 \times 10^6$	$1,6454 \times 10^6$ (41,70%)		$4,5485 \times 10^6$	$1,2448 \times 10^6$ (27,36%)
5%	66,99% Dif	$8,4743 \times 10^5$	$2,8727 \times 10^5$ (33,89%)	63,09% Dif	$1,0704 \times 10^6$	$9,6197 \times 10^5$ (89,87%)	65,40% Dif	$2,3862 \times 10^6$	$2,1839 \times 10^6$ (91,52%)
		$2,5677 \times 10^6$	$9,3436 \times 10^5$ (36,38%)		$2,9000 \times 10^6$	$1,8125 \times 10^6$ (65,50%)		$6,8973 \times 10^6$	$6,2750 \times 10^6$ (90,97%)
10%	81,03% Dif	$7,7511 \times 10^5$	$7,5469 \times 10^5$ (97,36%)	85,72% Dif	$1,0385 \times 10^6$	$8,8671 \times 10^5$ (85,38%)	60,20% Dif	$1,2344 \times 10^6$	$1,0453 \times 10^6$ (84,68%)
		$4,0861 \times 10^6$	$3,8116 \times 10^6$ (93,28%)		$7,2725 \times 10^6$	$7,0089 \times 10^6$ (96,37%)		$3,1021 \times 10^6$	$3,0505 \times 10^6$ (98,33%)

Fonte: O autor, 2024.

Analisando as distribuições mostradas nas Figuras 46, 47 e 48, verifica-se que a metamodelagem não causou mudanças significativas nos valores ótimos médios obtidos. Isso é evidenciado pela semelhança entre esses valores e os resultados sem uso de metamodelagem, indicando que a técnica preserva a integridade dos resultados. Isto corrobora para a adoção da metamodelagem em contextos que requerem elevada complexidade numérica, principalmente os que envolvem robustez e confiabilidade.

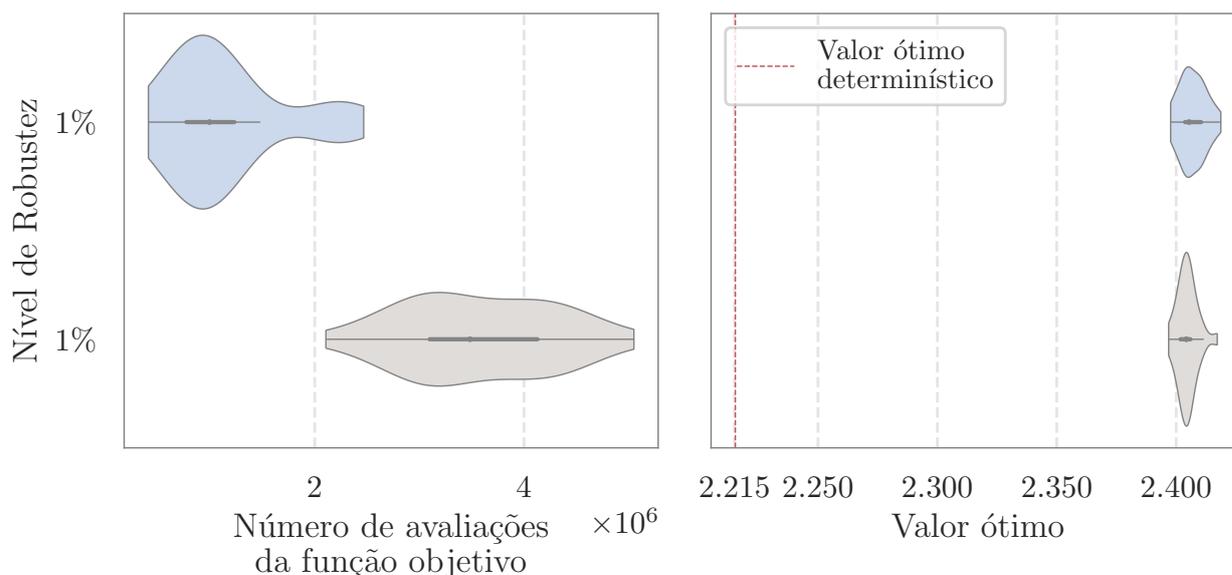
Ao analisar o valor ótimo obtido em função do número de avaliações por geração da função objetivo em cada ciclo do SORA, conforme ilustrado nas imagens da Figura 49, observa-se uma redução no número de avaliações da função objetivo em comparação à otimização sem metamodelagem. Essa redução permite alcançar valores próximos à estabilidade com menos avaliações da função objetivo, o que pode ser extremamente relevante em cenários onde a análise da função objetivo requer um esforço computacional substancial, mas a resposta deve ser fornecida em um período de tempo aceitável, como previsões meteorológicas, que podem demandar um tempo considerável. No primeiro ciclo do SORA, conforme visualizado na Figura 49a, é realizada a otimização robusta sem levar em conta as restrições probabilísticas. Ambas as abordagens atingem, com e sem metamodelagem, valores ótimos similares. No entanto, a abordagem que utiliza metamodelagem no primeiro ciclo do SORA alcança valores ótimos com uma redução

Tabela 16 - Valores ótimos obtidos para com o uso da metamodelagem para diferentes níveis de robustez e índices de confiabilidade.

Nível de Robustez	Usando Metamodelagem								
	Sem o uso de Metamodelagem								
	$\beta_t = 1$		$\beta_t = 2$		$\beta_t = 3$				
	Média	Desvio Padrão (C.V.)	Média	Desvio Padrão (C.V.)	Média	Desvio Padrão (C.V.)			
1%	$0,09\%$ Dif	2406,9371	5,3585 (0,22%)	$0,10\%$ Dif	2590,0193	6,1040 (0,23%)	$0,01\%$ Dif	2792,8050	6,6808 (0,23%)
		2404,6337	4,1188 (0,17%)		2587,1943	6,1925 (0,23%)		2792,3049	6,4611 (0,23%)
5%	$0,95\%$ Dif	2530,5406	101,8992 (0,63%)	$0,44\%$ Dif	2719,8377	102,4031 (3,76%)	$0,24\%$ Dif	2915,2658	67,8463 (2,32%)
		2506,6767	15,8905 (2,17%)		2707,8737	71,9784 (2,65%)		2908,0229	68,4050 (2,35%)
10%	$0,2\%$ Dif	2697,9106	110,6919 (4,10%)	$1,56\%$ Dif	2917,6516	152,4063 (5,22%)	$0,13\%$ Dif	3254,5428	222,2268 (6,82%)
		2703,4649	136,4796 (5,04%)		2964,1066	176,5686 (5,95%)		3250,0824	204,5817 (6,29%)

Fonte: O autor, 2024.

Figura 46 - Distribuição do número de avaliações e resultado ótimo obtido em 30 execuções com 1% de robustez e índice de confiabilidade $\beta_t = 1,0$.



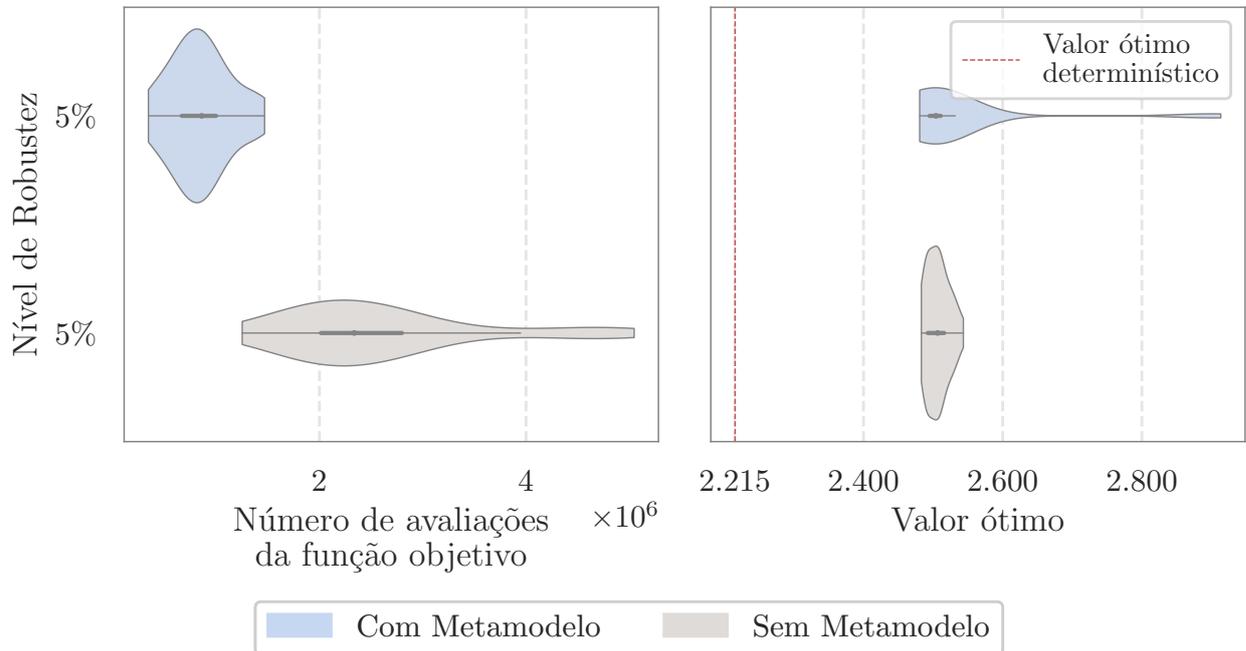
Fonte: O autor, 2024.

considerável no número de avaliações da função objetivo.

No segundo ciclo do SORA, representado na Figura 49b, observa-se que o uso da metamodelagem resultou em um desvio padrão menor na fase de refinamento, quando comparado com a otimização sem o uso de metamodelagem. Isso se refletiu em uma menor necessidade de ciclos adicionais (Figuras 49c e 49d) para obter soluções ótimas robustas que satisfaçam as restrições probabilísticas, visualizado na menor quantidade de execuções que demandaram os terceiros e quarto ciclos.

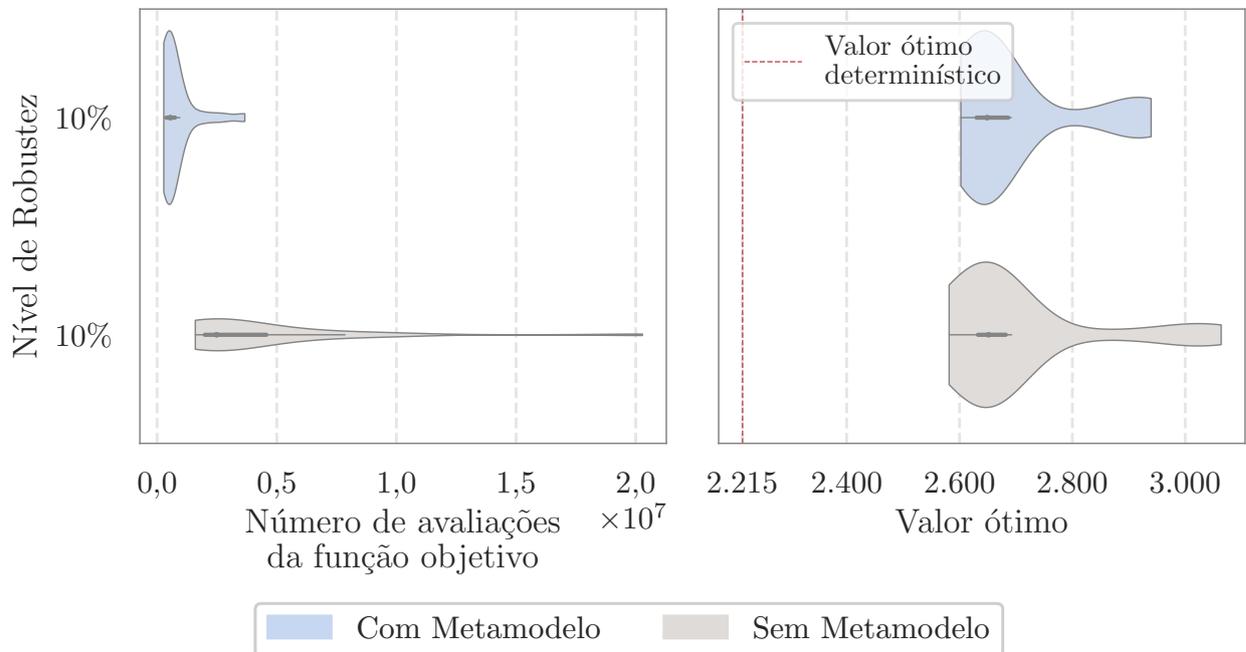
Os gráficos dos ciclos do SORA foram apresentados exclusivamente para $\beta_t = 1$,

Figura 47 - Distribuição do número de avaliações e valor ótimo obtido para 30 execuções com 5% de Robustez e índice de Confiabilidade $\beta_t = 1,0$.



Fonte: O autor, 2024.

Figura 48 - Distribuição do número de avaliações e valor ótimo obtido para 30 execuções com 10% de Robustez e índice de Confiabilidade $\beta_t = 1,0$.

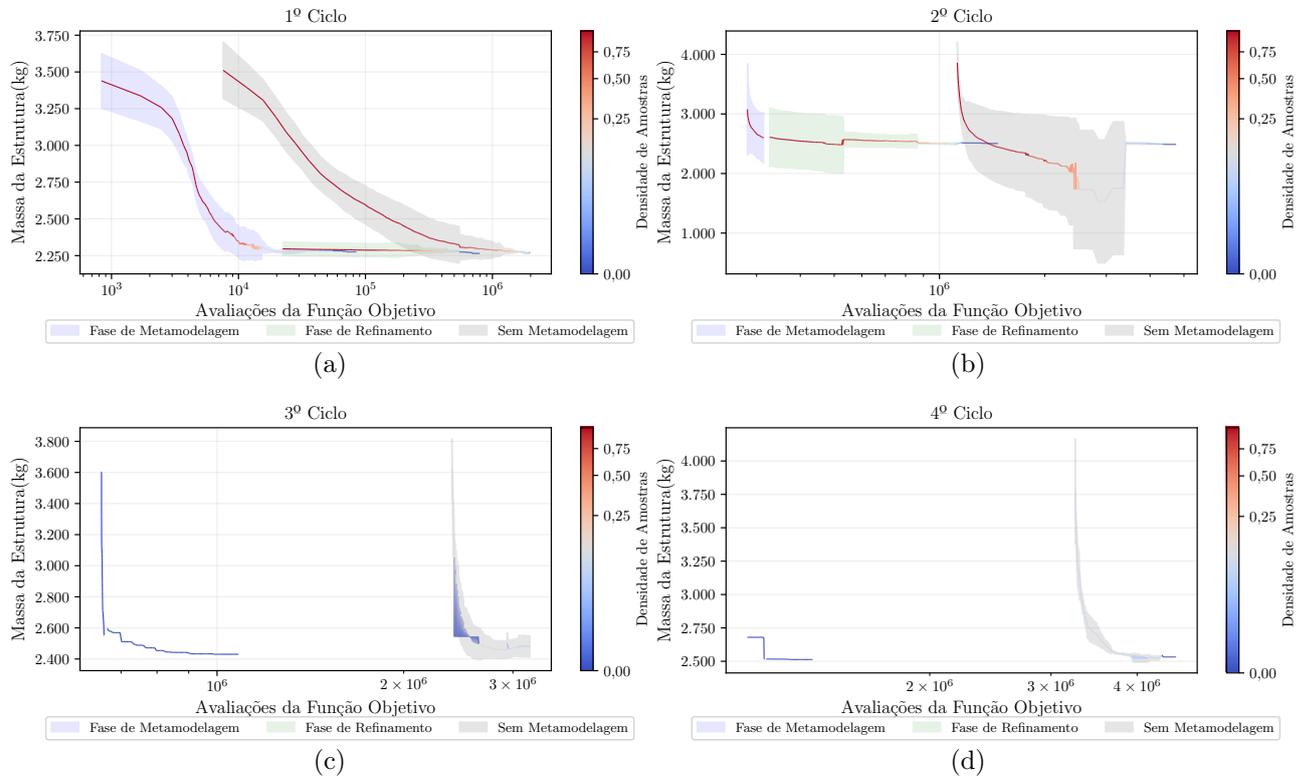


Fonte: O autor, 2024.

uma vez que as representações gráficas correspondentes a outros valores de β_t se mostraram bastante similares, não apresentando comportamentos distintos que justifiquem uma análise mais detalhada.

Para permitir uma compreensão das interações entre índices de confiabilidade e

Figura 49 - Ciclos do SORA com as etapas de metamodelagem e refinamento para alcançar 5% de robustez e índice de confiabilidade $\beta_t = 1,0$. As áreas sombreadas representam o respectivo desvio padrão.



Fonte: O autor, 2024.

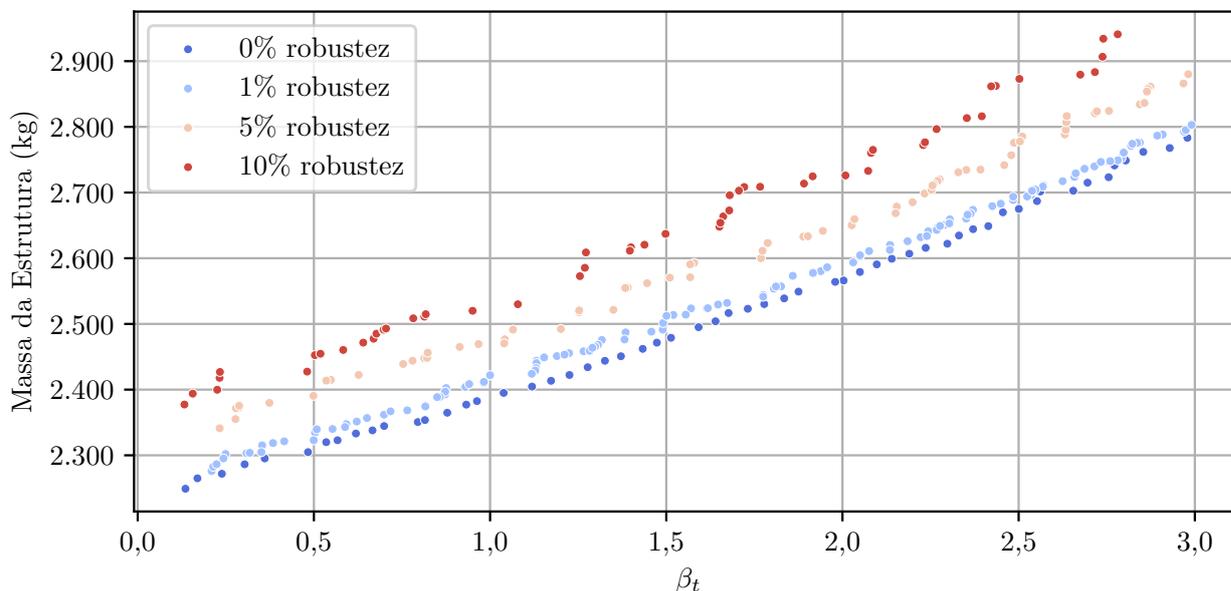
graus de robustez, o processo de otimização multiobjetivo foi executado utilizando a projeção estereográfica, em conjunto com a técnica ASOSL e o algoritmo evolutivo NSGA-II. Ao considerar o método ASOSL com projeção estereográfica, os parâmetros foram definidos da seguinte forma: a taxa de aprendizado foi fixada em $s = 0,5$, com $\alpha = 1 \times 10^{-4}$. O critério para interromper a busca pelo MPP foi estipulado quando a norma do gradiente atingisse um valor inferior a $\varepsilon = 10^{-6}$.

No que se refere ao algoritmo evolutivo, o NSGA-II foi configurado com uma taxa de mutação de 0,2 e uma taxa de cruzamento de 0,8. Para o critério de parada, foi exigida uma estabilidade no espaço objetivo por 100 gerações consecutivas, com uma tolerância de 5×10^{-4} , limitando-se o máximo de gerações a 3000.

A Figura 50 apresenta uma visão sobre o impacto dos distintos níveis de robustez e confiabilidade nos valores associados à massa da estrutura da treliça de 10 barras. É importante destacar que um aumento no índice de confiabilidade está diretamente relacionado a uma redução na probabilidade de ocorrência de violação da restrição, mas impacta na massa da estrutura. Além disso, observa-se que a elevação do nível de robustez dificulta a identificação de soluções robustas que sejam compatíveis com o índice de confiabilidade analisado, evidenciando a complexidade do equilíbrio entre essas duas

naturezas conflitantes.

Figura 50 - Fronteiras de Pareto elaboradas com base na otimização com robustez e confiabilidade para o problema da treliça de 10 barras.



Fonte: O autor, 2024.

Neste capítulo, foram apresentados os resultados alcançados em problemas de benchmark com o objetivo de analisar a eficácia da metodologia proposta na diminuição do custo computacional ao realizar análises de robustez e confiabilidade. A implementação de metamodelos fundamentados em processos gaussianos permitiu a redução significativa do custo computacional associado às avaliações da função objetivo, sem comprometer a qualidade ou a precisão das soluções encontradas.

CONCLUSÕES

A presente tese desenvolveu e validou uma metodologia híbrida que integra metamodelagem e projeção estereográfica em problemas de otimização robusta e baseada em confiabilidade. A pesquisa respondeu a desafios associados ao custo computacional elevado e à necessidade de encontrar soluções robustas e confiáveis em sistemas de engenharia. Os resultados alcançados proporcionam suporte tanto para o avanço científico quanto para as aplicações práticas no campo da engenharia.

Inicialmente, destacou-se a importância dos metamodelos na redução do custo computacional. Tais metamodelos são capazes de aproximar funções que apresentam alta complexidade numérica e são computacionalmente exigentes, proporcionando uma abordagem mais eficiente. Com isso, diminuem a quantidade de vezes que é necessário avaliar a função objetivo, sem comprometer a qualidade das soluções. Os experimentos realizados indicaram que é possível alcançar uma diminuição de até 60% no número de avaliações nos problemas RBRDO, sendo que as variações percentuais observadas entre as soluções ótimas, com e sem a aplicação de metamodelos, permaneceram inferiores a 1%.

A adoção dos Processos Gaussianos para a construção dos metamodelos trouxe flexibilidade na capacidade de modelar funções não lineares, fornecendo estimativas da incerteza associada às predições, possibilitando o uso de funções de aquisição para guiar a exploração do espaço de busca. Essa abordagem pode resultar em um equilíbrio eficiente entre exploração e refinamento, maximizando o aproveitamento de recursos computacionais.

Um dos principais avanços metodológicos deste trabalho é a introdução da técnica de projeção estereográfica para análise de confiabilidade inversa. Essa técnica substitui as projeções sucessivas na hipersfera, utilizadas pelo ASOSL, por uma transformação bijetiva que mapeia a hipersfera para um espaço euclidiano de dimensões equivalentes. Essa abordagem simplifica a busca pelo ponto mais provável de falha, aproveitando a característica de que o gradiente é nulo no ponto em que a função de desempenho atinge seu valor mínimo. A adoção da projeção estereográfica não alterou os resultados obtidos em problemas de referência, sugerindo que essa abordagem continua a ser eficiente e possui potencial para aplicação em problemas de alta dimensionalidade.

O estudo também abordou a formulação de problemas de otimização como problemas multiobjetivo, equilibrando robustez, confiabilidade e custo computacional. Essa abordagem revelou-se promissora na identificação de *trade-offs* entre objetivos conflitantes, fornecendo soluções otimizadas e equilibradas para sistemas de engenharia.

A validação por meio de experimentos numéricos incluiu problemas de *benchmark* e casos práticos, como o projeto de estruturas de treliças, mostrando a aplicabilidade das metodologias sugeridas. Os testes confirmaram que a abordagem híbrida é eficiente em

lidar com incertezas aleatórias, fornecendo soluções seguras mesmo em condições desafiadoras.

No entanto, foram observadas algumas limitações. A efetividade do método depende de parâmetros críticos, como o nível de ruído na robustez e o índice de confiabilidade desejado. Além disso, problemas com alta dimensionalidade ainda podem exigir muitos recursos computacionais, embora a metamodelagem ajude a reduzir significativamente esse impacto.

Em termos práticos, a metodologia desenvolvida tem implicações consideráveis para a engenharia. Sistemas que exigem decisões robustas sob incertezas, como estruturas de grande porte ou dispositivos de alta precisão, podem se beneficiar diretamente das ferramentas propostas. Além disso, o *framework* contribui para uma maior sustentabilidade ao reduzir o número de simulações computacionalmente caras.

A robustez da metodologia foi reforçada pela sua capacidade de preservar o desempenho mesmo em condições adversas. Soluções encontradas foram consistentemente viáveis, confiáveis e eficientes, atendendo às demandas de segurança e desempenho em engenharia.

Por fim, a pesquisa contribui para o avanço científico ao propor inovações metodológicas que podem ser amplamente aplicadas em problemas de engenharia e outros domínios. A partir das bases estabelecidas, espera-se que os conceitos desenvolvidos inspirem novos estudos e ampliem o impacto das soluções robustas e confiáveis em diferentes áreas.

Em síntese, esta tese alcançou seus objetivos ao oferecer uma solução prática para a otimização robusta e baseada em confiabilidade. Os avanços apresentados consolidam a importância da metamodelagem e da projeção estereográfica como ferramentas adicionais em problemas de otimização sob incertezas, proporcionando resultados eficientes e aplicáveis a sistemas complexos.

Perspectivas Futuras

Uma direção promissora para futuras pesquisas é investigar as incertezas causadas por informações incompletas ou imprecisas. Técnicas, como análise intervalar e lógica *fuzzy*, podem ser combinadas para aumentar a aplicação prática da metodologia em contextos reais, onde a incerteza possui diversas formas.

Outro aspecto que merece aprofundamento é a análise de diferentes critérios de parada para a transição entre a fase de exploração com metamodelos e o refinamento com a função original. Estratégias como a estabilização da função de aquisição ou a análise de sensibilidade podem fortalecer a robustez da metodologia proposta.

Embora os Processos Gaussianos tenham se mostrado eficientes, há espaço para explorar outras técnicas de metamodelagem, como redes neurais artificiais e máquinas de vetores de suporte. Comparações entre essas abordagens podem revelar vantagens,

dependendo da natureza do problema de otimização.

A combinação de metamodelagem com técnicas de redução de ordem também se apresenta como uma possibilidade promissora. O uso combinado de metamodelos e modelos reduzidos, como os obtidos por métodos de elementos finitos reduzidos, pode abrir novos horizontes para lidar com problemas de alta dimensionalidade de maneira ainda mais eficiente.

A integração entre metamodelagem e projeção estereográfica também pode ser aprofundada, buscando adaptar a metodologia para problemas com distribuições de probabilidade não gaussianas. Esse avanço aumentaria a flexibilidade da abordagem para aplicações em cenários reais complexos.

O uso da projeção estereográfica demonstrou ser particularmente vantajoso para a análise de confiabilidade inversa, mas sua aplicação pode ser ampliada para outros domínios onde hiperesferas são utilizadas, como em problemas de classificação ou *clustering* em espaços de alta dimensionalidade.

A abordagem híbrida também oferece uma base para futuros desenvolvimentos em otimização adaptativa. A combinação de aprendizado de máquina com os conceitos apresentados pode impulsionar ainda mais a eficácia das ferramentas de otimização em cenários dinâmicos e de alta complexidade.

REFERÊNCIAS

- ABEBE, M.; KOO, B.; KIM, M. G.; KIM, H. S. Surrogate-assisted reliability-based design optimization of PEMFC serpentine flow channel. *Frontiers in Energy Research*, v. 12, n. set., p. 1–16, 2024. ISSN 2296598X. Disponível em: <https://doi.org/10.3389/fenrg.2024.1468702>. Acesso em: 29 nov. 2024.
- ADIL, B.; CENGIZ, B. Optimal design of truss structures using weighted superposition attraction algorithm. *Engineering with Computers*, Springer Science and Business Media LLC, v. 36, n. 3, p. 965–979, jun. 2019. ISSN 1435-5663. Disponível em: <https://doi.org/10.1007/s00366-019-00744-x>. Acesso em: 19 out. 2023.
- ANDREI, N. Relaxed gradient descent and a new gradient descent methods for unconstrained optimization. 2004. Submitted for publication to Journal of Mathematical Programming. Disponível em: <http://www.ici.ro/camo/neculai/newgrad.pdf>. Acesso em: 17 abr. 2023.
- ARORA, J. S. Global optimization concepts and methods. In: ARORA, J. S. (Ed.). *Introduction to Optimum Design (Third Edition)*. Third edition. Boston: Academic Press, 2012. p. 681–712. ISBN 978-0-12-381375-6. Disponível em: <https://doi.org/10.1016/B978-0-12-381375-6.00018-8>. Acesso em: 01 jan. 2023.
- BAQUELA, E. G.; OLIVERA, A. C. A novel hybrid multi-objective metamodel-based evolutionary optimization algorithm. *Operations Research Perspectives*, v. 6, p. 100098, 2019. ISSN 2214-7160. Disponível em: <https://doi.org/10.1016/j.orp.2019.100098>. Acesso em: 07 jul. 2023.
- BINH, T. T.; KORN, U. Mobes: A multi-objective evolution strategy for constrained optimization problems. In: *Proceedings of the 3rd International Conference on Genetic Algorithms (MENDEL97)*. San Francisco, CA: Morgan Kaufmann, 1997. p. 176–182. Disponível em: <https://api.semanticscholar.org/CorpusID:14315597>. Acesso em: 22 jul. 2024.
- BLANK, J.; DEB, K. pymoo: Multi-objective optimization in python. *IEEE Access*, IEEE, v. 8, p. 89497–89509, 2020. Disponível em: <https://doi.org/10.1109/ACCESS.2020.2990567>. Acesso em: 30 abr. 2022.
- BORRI, A.; SPERANZINI, E. Structural reliability analysis using a standard deterministic finite element code. *Structural Safety*, Elsevier BV, v. 19, n. 4, p. 361–382, jan. 1997. ISSN 0167-4730. Disponível em: [https://doi.org/10.1016/s0167-4730\(97\)00017-9](https://doi.org/10.1016/s0167-4730(97)00017-9). Acesso em: 26 ago. 2024.
- BRANKE, J.; DEB, K.; MIETTINEN, K.; SLOWIŃSKI, R. *Multiobjective Optimization: Interactive and Evolutionary Approaches*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008. (Lecture Notes in Computer Science). ISBN 9783540889083. Disponível em: <https://doi.org/10.1007/978-3-540-88908-3>. Acesso em: 14 ago. 2023.
- BRASIL, N. U. no. *Objetivos de Desenvolvimento Sustentável*. 2025. Disponível em: <https://brasil.un.org/pt-br/sdgs>. Acesso em: 22 dez. 2024.

- BRAUN, F. *Compactificação de Poincaré de Campos Polinomiais*. Dissertação (Mestrado) — Universidade Federal de São Carlos, São Carlos, 2007. Disponível em: <https://www.dm.ufscar.br/ppgm/attachments/article/739/braun.pdf>. Acesso em: 08 ago. 2023.
- CASTRO, H. F.; PAULA, E. H.; VISNADI, L. B. Reliability-Based Design Optimization Applied to a Rotor Supported by Hydrodynamic Bearings. *Machines*, v. 12, n. 4, 2024. ISSN 20751702. Disponível em: <https://doi.org/10.3390/machines12040233>. Acesso em: 30 nov. 2024.
- CHEN, J.; CHEN, Z.; JIANG, W.; GUO, H.; CHEN, L. A reliability-based design optimization strategy using quantile surrogates by improved PC-kriging. *Reliability Engineering and System Safety*, Elsevier Ltd, v. 253, n. set. 2024, p. 110491, 2025. ISSN 09518320. Disponível em: <https://doi.org/10.1016/j.ress.2024.110491>. Acesso em: 01 dez. 2024.
- COELLO, C. A. C.; LAMONT, G. B.; VELDHUIZEN, D. A. V. *Evolutionary Algorithms for Solving Multi-Objective Problems*. 2. ed. New York: Springer New York, NY, 2007. XXI, 800 p. (Genetic and Evolutionary Computation). ISBN 978-0-387-36797-2. Disponível em: <https://doi.org/10.1007/978-0-387-36797-2>. Acesso em: 12 maio 2022.
- DEB, K.; GUPTA, H. *Searching for Robust Pareto-Optimal Solutions in Multi-objective Optimization*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005. 150–164 p. ISSN 1611-3349. ISBN 9783540318804. Disponível em: https://doi.org/10.1007/978-3-540-31880-4_11. Acesso em: 30 abr. 2024.
- DEB, K.; PRATAP, A.; AGARWAL, S.; MEYARIVAN, T. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE Transactions on Evolutionary Computation*, v. 6, n. 2, p. 182–197, 2002. Disponível em: <https://doi.org/10.1109/4235.996017>. Acesso em: 30 abr. 2024.
- DU, X.; CHEN, W. Sequential optimization and reliability assessment method for efficient probabilistic design. *Journal of Mechanical Design*, ASME International, v. 126, n. 2, p. 225–233, mar. 2004. ISSN 1528-9001. Disponível em: <https://doi.org/10.1115/1.1649968>. Acesso em: 27 ago. 2024.
- DU, Y.; GU, J.; ZHANG, H.; HUA, W. Reliability-Based Robust Optimization of High-Speed PM Synchronous Machine with Local Surrogate Model Strategy. *IEEE Transactions on Transportation Electrification*, IEEE, PP, p. 1, 2024. ISSN 23327782. Disponível em: <https://doi.org/10.1109/TTE.2024.3360112>. Acesso em: 01 dez. 2024.
- EIBEN, A.; SMITH, J. *Introduction to Evolutionary Computing*. 2. ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2015. (Natural Computing Series). ISSN 1619-7127. ISBN 9783662448748. Disponível em: <https://doi.org/10.1007/978-3-662-44874-8>. Acesso em: 11 jun. 2022.
- FERREIRA, A.; FANTUZZI, N. *MATLAB Codes for Finite Element Analysis: Solids and Structures*. Singapore: Springer International Publishing, 2020. (Solid Mechanics and Its Applications). ISBN 9783030479527. Disponível em: <https://doi.org/10.1007/978-1-4020-9200-8>. Acesso em: 17 jul. 2023.
- FONSECA, C.; PAQUETE, L.; LOPEZ-IBANEZ, M. An improved dimension-sweep algorithm for the hypervolume indicator. In: *2006 IEEE International Conference on Evolutionary Computation*. Vancouver, BC, Canada: IEEE, 2006. p. 1157–1163. Disponível em: <https://doi.org/10.1109/CEC.2006.1688440>. Acesso em: 12 fev. 2024.

GOLDBERG, D. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Reading, MA: Addison-Wesley, 1989. (Addison Wesley series in artificial intelligence). ISBN 9780201157673.

HALDAR, A.; MAHADEVAN, S. *Probability, Reliability, and Statistical Methods in Engineering Design*. New York: Wiley, 2000. ISBN 9780471331193.

HUYNH, T. V.; TANGARAMVONG, S.; DO, B.; GAO, W. A novel decoupled approach combining invertible cross-entropy method with Gaussian process modeling for reliability-based design and topology optimization. *Computer Methods in Applied Mechanics and Engineering*, Elsevier B.V., v. 427, n. February, p. 117006, 2024. ISSN 00457825. Disponível em: <https://doi.org/10.1016/j.cma.2024.117006>. Acesso em: 01 dez. 2024.

JIANG, P.; ZHOU, Q.; SHAO, X. *Surrogate Model-Based Engineering Design and Optimization*. Singapore: Springer Singapore, 2020. ISBN 978-981-15-0730-4. Disponível em: <https://doi.org/10.1007/978-981-15-0731-1>. Acesso em: 02 ago. 2022.

JONES, D. R.; SCHONLAU, M.; WELCH, W. J. Efficient global optimization of expensive black-box functions. *Journal of Global Optimization*, Springer, v. 13, n. 1, p. 455–492, 1998. Disponível em: <https://doi.org/10.1023/A:1008306431147>. Acesso em: 04 out. 2023.

KAVEH, A.; ZAERREZA, A. A new framework for reliability-based design optimization using metaheuristic algorithms. *Structures*, Elsevier BV, v. 38, p. 1210–1225, abr. 2022. ISSN 2352-0124. Disponível em: <https://doi.org/10.1016/j.istruc.2022.02.069>. Acesso em: 06 maio 2024.

KHODAM, A.; MESBAHI, P.; SHAYANFAR, M.; AYYUB, B. M. Global decoupling for structural reliability-based optimal design using improved differential evolution and chaos control. *ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part A: Civil Engineering*, American Society of Civil Engineers (ASCE), v. 7, n. 1, mar. 2021. ISSN 2376-7642. Disponível em: <https://doi.org/10.1061/ajrua6.0001097>. Acesso em: 27 ago. 2024.

KIM, J.; YI, S. ri; SONG, J. Active learning-based optimization of structures under stochastic excitations with first-passage probability constraints. *Engineering Structures*, Elsevier Ltd, v. 307, n. March, p. 117873, 2024. ISSN 18737323. Disponível em: <https://doi.org/10.1016/j.engstruct.2024.117873>. Acesso em: 02 dez. 2024.

KUDELA, J.; MATOUSEK, R. Recent advances and applications of surrogate models for finite element method computations: a review. *Soft Computing*, v. 26, n. 24, p. 13709–13733, 2022. Disponível em: <https://doi.org/10.1007/s00500-022-07362-8>. Acesso em: 16 jul. 2023.

LARRAÑAGA, A.; MARTÍNEZ, J.; MÍGUEZ, J. L.; PORTEIRO, J. Robust optimization of heat-transfer-enhancing microtextured surfaces based on machine learning surrogate models. *International Communications in Heat and Mass Transfer*, v. 151, n. December 2023, 2024. ISSN 07351933. Disponível em: <https://doi.org/10.1016/j.icheatmasstransfer.2023.107218>. Acesso em: 01 dez. 2024.

LIBOTTE, G. B. *Otimização Multi-objetivo de Sistemas de Engenharia na Presença de Incerteza*. Tese (phdthesis) — Universidade do Estado do Rio de Janeiro - Instituto Politécnico, 2020. Disponível em: <https://www.bdtd.uerj.br:8443/handle/1/16820>. Acesso em: 22 abr. 2022.

LIBOTTE, G. B.; LOBATO, F. S.; NETO, F. D. M.; PLATT, G. M. Adaptive second order step length algorithm for inverse reliability analysis. *Advances in Engineering Software*, Elsevier BV, v. 146, p. 102831, ago. 2020. ISSN 0965-9978. Disponível em: <https://doi.org/10.1016/j.advengsoft.2020.102831>. Acesso em: 22 abr. 2022.

LIU, J.; ZHANG, D.; LIU, J.; SUN, W.; PAN, H.; ZHU, Y.; JING, H.; FANG, Z. Robust Optimization under Geological Uncertainty Using a TransUNet-Based Surrogate Model with EnOpt Algorithm. *Mathematical Geosciences*, Springer Berlin Heidelberg, 2024. ISSN 18748953. Disponível em: <https://doi.org/10.1007/s11004-024-10165-2>. Acesso em: 01 dez. 2024.

LIU, X.; DENG, J.; CHEN, H.; ZHAI, G.; WU, J. An efficient and multi-fidelity reliability-based design optimization method based on a novel surrogate model local update strategy. *Computer Methods in Applied Mechanics and Engineering*, Elsevier B.V., v. 430, n. April, p. 117219, 2024. ISSN 00457825. Disponível em: <https://doi.org/10.1016/j.cma.2024.117219>. Acesso em: 30 nov. 2024.

LOPEZ, R. H.; BECK, A. T. Reliability-based design optimization strategies based on form: a review. *Journal of The Brazilian Society of Mechanical Sciences and Engineering*, v. 34, p. 506–514, 2012. Disponível em: <https://doi.org/10.1590/S1678-58782012000400012>. Acesso em: 22 mar. 2024.

MA, H.; ZHANG, Y.; SUN, S.; LIU, T.; SHAN, Y. A comprehensive survey on NSGA-II for multi-objective optimization and applications. *Artificial Intelligence Review*, v. 56, n. 12, p. 15217–15270, dec 2023. ISSN 0269-2821. Acesso em: 01 set. 2024.

MAI, H. T.; LEE, J.; KANG, J.; NGUYEN-XUAN, H.; LEE, J. An improved blind kriging surrogate model for design optimization problems. *Mathematics*, MDPI, v. 10, n. 16, p. 2906, 2022. Disponível em: <https://doi.org/10.3390/math10162906>. Acesso em: 27 dez. 2024.

MIRJALILI, S.; DONG, J. S. *Multi-Objective Optimization using Artificial Intelligence Techniques*. 1st. ed. Cham, Switzerland: Springer International Publishing, 2020. (SpringerBriefs in Applied Sciences and Technology). ISSN 2191-5318. ISBN 9783030248352. Disponível em: <https://doi.org/10.1007/978-3-030-24835-2>. Acesso em: 28 abr. 2022.

NOCEDAL, J.; WRIGHT, S. *Numerical Optimization*. New York: Springer New York, 2006. (Springer Series in Operations Research and Financial Engineering). ISBN 9780387227429. Disponível em: <https://doi.org/10.1007/978-0-387-40065-5>. Acesso em: 08 set. 2022.

PASSOS, A. G. *Otimização Multiobjetivo com Base em Processo Gaussiano de Regressão (Kriging)*. Tese (Doutorado) — Universidade Tecnológica Federal do Paraná. Programa de Pós-Graduação em Engenharia Mecânica e de Materiais, 2020. Disponível em: <https://repositorio.utfpr.edu.br/jspui/bitstream/1/5092/1/processogaussianoregressaokrigin.pdf>. Acesso em: 20 set. 2022.

PERICARO, G.; SANTOS, S.; RIBEIRO, A.; MATIOLI, L. HLRFBFGS optimization algorithm for structural reliability. *Applied Mathematical Modelling*, Elsevier BV, v. 39, n. 7, p. 2025–2035, abr. 2015. ISSN 0307-904X. Disponível em: <https://doi.org/10.1016/j.apm.2014.10.024>. Acesso em: 07 jun. 2022.

RASMUSSEN, C. E.; WILLIAMS, C. K. *Gaussian Processes for Machine Learning*. Cambridge, MA: The MIT Press, 2006. (Adaptive Computation and Machine Learning series). ISBN 9780262182539. Disponível em: <https://doi.org/10.7551/mitpress/3206.001.0001>. Acesso em: 01 set. 2022.

RENKAVIESKI, C.; PARPINELLI, R. S. Meta-heuristic algorithms to truss optimization: Literature mapping and application. *Expert Systems with Applications*, Elsevier BV, v. 182, p. 115197, nov. 2021. ISSN 0957-4174. Disponível em: <https://doi.org/10.1016/j.eswa.2021.115197>. Acesso em: 09 jul. 2023.

ROSENBLATT, M. Remarks on a multivariate transformation. *The Annals of Mathematical Statistics*, Institute of Mathematical Statistics, v. 23, n. 3, p. 470–472, set. 1952. ISSN 0003-4851. Disponível em: <https://doi.org/10.1214/aoms/1177729394>. Acesso em: 27 fev. 2022.

SCHULZ, E.; SPEEKENBRINK, M.; KRAUSE, A. A tutorial on gaussian process regression: Modelling, exploring, and exploiting functions. *Journal of Mathematical Psychology*, v. 85, p. 1–21, 2018. Disponível em: <https://doi.org/10.1016/j.jmp.2018.03.001>. Acesso em: 01 abr. 2024.

SHAHRIARI, B.; SWERSKY, K.; WANG, Z.; ADAMS, R. P.; FREITAS, N. de. Taking the human out of the loop: A review of bayesian optimization. *Proceedings of the IEEE*, v. 104, n. 1, p. 148–175, 2016. Disponível em: <https://doi.org/10.1109/JPROC.2015.2494218>. Acesso em: 22 jan. 2022.

SHAYANFAR, M.; ABBASNIA, R.; KHODAM, A. Development of a ga-based method for reliability-based optimization of structures with discrete and continuous design variables using opensees and tcl. *Finite Elements in Analysis and Design*, Elsevier BV, v. 90, p. 61–73, nov. 2014. ISSN 0168-874X. Disponível em: <https://doi.org/10.1016/j.finel.2014.06.010>. Acesso em: 29 ago. 2024.

SHI, J.; CHU, L.; BRAUN, R. A kriging surrogate model for uncertainty analysis of graphene based on a finite element method. *International Journal of Molecular Sciences*, MDPI, v. 20, n. 9, p. 2355, 2019. Disponível em: <https://doi.org/10.3390/ijms20092355>. Acesso em: 29 dez. 2024.

SHI, Y.; LIN, C.; MA, Y.; SHEN, J. Active learning Kriging-based multi-objective modeling and optimization for system reliability-based robust design. *Reliability Engineering and System Safety*, Elsevier Ltd, v. 245, n. February, p. 110007, 2024. ISSN 09518320. Disponível em: <https://doi.org/10.1016/j.res.2024.110007>. Acesso em: 01 dez. 2024.

SRINIVAS, N.; DEB, K. Multiobjective optimization using nondominated sorting in genetic algorithms. *Evolutionary Computation*, v. 2, n. 3, p. 221–248, 1994. Disponível em: <https://doi.org/10.1162/evco.1994.2.3.221>. Acesso em: 12 mar. 2022.

TALBI, E.-G. *Metaheuristics: From Design to Implementation*. Hoboken: Wiley Publishing, 2009. ISBN 0470278587. Disponível em: <https://doi.org/10.1002/9780470496916>. Acesso em: 07 dez. 2021.

TRZECIAK, T. M. *Some PGF/TikZ 3D drawings*. 2008. Disponível em: <https://latex.org/forum/viewtopic.php?f=4&t=2111>. Acesso em: 21 jul. 2024.

TSUTSUI, S.; GHOSH, A.; FUJIMOTO, Y. A robust solution searching scheme in genetic search. In: SPRINGER. *International Conference on Parallel Problem Solving from Nature*. Berlin, Heidelberg, 1996. p. 543–552. Disponível em: https://doi.org/10.1007/3-540-61723-X_1018. Acesso em: 06 abr. 2023.

TU, J.; CHOI, K. K.; PARK, Y. H. A new study on reliability-based design optimization. *Journal of Mechanical Design*, ASME International, v. 121, n. 4, p. 557–564, dez. 1999. ISSN 1528-9001. Disponível em: <https://doi.org/10.1115/1.2829499>. Acesso em: 07 nov. 2022.

VON MISES, R. *Mathematical Theory of Probability and Statistics*. New York: Academic Press, 1964. ISBN 9780127273563. Disponível em: <https://doi.org/10.1016/C2013-0-12460-9>. Acesso em: 27 ago. 2024.

WACKERNAGEL, H. *Multivariate Geostatistics: An Introduction with Applications*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003. ISBN 978-3-642-0791 1-5. Disponível em: <https://doi.org/10.1007/978-3-662-05294-5>. Acesso em: 03 mar. 2022.

WANG, Y.; SHIYUAN, E.; YANG, K.; XIE, B.; LU, F. Reliability-Based Robust Design Optimization with Fourth-Moment Method for Ball Bearing Wear. *Lubricants*, v. 12, n. 8, 2024. ISSN 20754442. Disponível em: <https://doi.org/10.3390/lubricants12080293>. Acesso em: 30 nov. 2024.

XUE, D.; LI, Y.; ZHANG, H.; TONG, X.; GAO, B.; YU, J. Reliability-based robust optimization design for tolerance of aerospace thin-walled components based on surrogate model. *Advanced Engineering Informatics*, Elsevier Ltd, v. 62, n. PB, p. 102754, 2024. ISSN 14740346. Disponível em: <https://doi.org/10.1016/j.aei.2024.102754>. Acesso em: 29 nov. 2024.

YANG, Y.; WANG, L.; ZHANG, Y.; ZHANG, H.; ZHOU, G. A computationally efficient surrogate model based robust optimization for permanent magnet synchronous machines. *IEEE Transactions on Energy Conversion*, v. 37, n. 3, p. 1520–1532, 2022. Disponível em: <https://doi.org/10.1109/TEC.2021.3140096>. Acesso em: 22 mar. 2023.

YU, S.; WU, X.; ZHAO, D.; LI, Y. A two-level surrogate framework for demand-objective time-variant reliability-based design optimization. *Reliability Engineering and System Safety*, Elsevier Ltd, v. 244, n. May 2023, p. 109924, 2024. ISSN 09518320. Disponível em: <https://doi.org/10.1016/j.ress.2023.109924>. Acesso em: 01 dez. 2024.