



**OTIMIZAÇÃO META-HEURÍSTICA MULTIOBJETIVO APLICADA AO MÉTODO SEM
MALHA LOCAL**

ELVIS PEREIRA DE SANTANA

**TESE DE DOUTORADO EM ESTRUTURAS E CONSTRUÇÃO CIVIL
DEPARTAMENTO DE ENGENHARIA CIVIL E AMBIENTAL**

FACULDADE DE TECNOLOGIA

UNIVERSIDADE DE BRASÍLIA

**UNIVERSIDADE DE BRASÍLIA
FACULDADE DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA CIVIL E AMBIENTAL**

**OTIMIZAÇÃO META-HEURÍSTICA MULTI OBJETIVO
APLICADA AO MÉTODO SEM MALHA LOCAL**

ELVIS PEREIRA DE SANTANA

ORIENTADOR: ARTUR PORTELA

TESE DE DOUTORADO EM ESTRUTURAS E CONSTRUÇÃO CIVIL

PUBLICAÇÃO: E.TD-05A/21

BRASÍLIA/DF: MARÇO - 2021

**UNIVERSIDADE DE BRASÍLIA
FACULDADE DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA CIVIL E AMBIENTAL**

**OTIMIZAÇÃO META-HEURÍSTICA MULTIOBJETIVO APLICADA
AO MÉTODO SEM MALHA LOCAL**

ELVIS PEREIRA DE SANTANA

**TESE SUBMETIDA AO DEPARTAMENTO DE ENGENHARIA CIVIL
E AMBIENTAL DA FACULDADE DE TECNOLOGIA DA
UNIVERSIDADE DE BRASÍLIA COMO PARTE DOS REQUISITOS
NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE DOUTOR EM
ESTRUTURAS E CONSTRUÇÃO CIVIL.**

APROVADO POR:

**Prof. Artur Portela, PhD (ENC-UnB)
(Orientador)**

**Prof. Luciano Mendes Bezerra, PhD (ENC-UnB)
(Examinador Interno)**

**Prof. Luis Alejandro Perez Peña, PhD (FAU-UnB)
(Examinador Externo)**

**Prof. Tiago da Silva Oliveira, PhD (ENC-IESB)
(Examinador Externo)**

BRASÍLIA/DF, 17 DE MARÇO DE 2021.

FICHA CATALOGRÁFICA

SANTANA, ELVIS P.

Otimização Meta-heurística Multiobjetivo Aplicada ao Método sem Malha Local. [Distrito Federal] 2021.

xxiv, 209p., 297 mm (ENC/FT/UnB, Doutor, Estruturas e Construção Civil, 2021). Tese de Doutorado – Universidade de Brasília. Faculdade de Tecnologia.

Departamento de Engenharia Civil e Ambiental.

1.Otimização meta-heurística multiobjetivo	2.Métricas de performance
3.Método sem malha local	4.Elasticidade linear
5.Mecânica da fratura	6.Subtração da singularidade
7.Fator de intensidade de tensão	
I.ENC/FT/UnB	II.Título (Doutor)

REFERÊNCIA BIBLIOGRÁFICA

SANTANA, E. P. (2021). Otimização Meta-heurística Multiobjetivo Aplicada ao Método sem Malha Local. Tese de Doutorado em Estruturas e Construção Civil, Publicação E.TD-05A/21, Departamento de Engenharia Civil e Ambiental, Universidade de Brasília, Brasília, DF, 209p.

CESSÃO DE DIREITOS

AUTOR: Elvis Pereira de Santana.

TÍTULO: Otimização Meta-heurística Multiobjetivo Aplicada ao Método sem Malha Local.

GRAU: Doutor

ANO: 2021

É concedida à Universidade de Brasília permissão para reproduzir cópias desta tese de doutorado e para emprestar ou vender tais cópias somente para propósitos acadêmicos e científicos. O autor reserva outros direitos de publicação e nenhuma parte desta tese de doutorado pode ser reproduzida sem autorização por escrito do autor.

Elvis Pereira de Santana
Rua 25 Norte, Lote 5, Bloco D, Apto 2406
71.917-180 - Águas Claras - Brasília - DF - Brasil.

Há verdadeiramente duas coisas diferentes: saber e crer que se sabe. A ciência consiste em saber; em crer que se sabe reside a ignorância.

– Hipócrates

AGRADECIMENTOS

Em primeiro lugar, sou grato a Deus por me permitir trilhar e concluir essa jornada, me dando disposição e capacidade para suplantar os obstáculos. Além de colocar pessoas inestimáveis ao meu redor, que foram e são fundamentais nas minhas conquistas.

À minha família, principalmente meus pais, Anita P. de Santana e José S. Sousa; à minha irmã Juliane P. S. Peres; e à minha companheira Jennainy A. P. Rosa, que me apoiam e incentivam incondicionalmente nos meus desafios. Sempre preocupados em me ajudarem no que podem, sem medirem esforços.

Ao meu orientador e amigo, Professor Artur Portela, a quem tenho grande admiração e apreço. Sempre sereno, disponível e disposto a compartilhar seu vasto conhecimento e experiência. Sem ele, a jornada teria sido muito mais árdua. Ao meu parceiro nessa jornada acadêmica, Tiago S. Oliveira, companheiro de batalhas e conquistas, sempre pronto para compartilhar conhecimento e auxílio.

A todos os professores e funcionários do PECC - *Pós-Graduação em Estruturas e Construção Civil*, Departamento de Engenharia Civil, Faculdade de Tecnologia, Universidade de Brasília, imprescindíveis nessa caminhada.

Aos meus colegas de trabalho, sempre compreensivos, prestativos e grandes incentivadores durante todo esse período. Também, à Companhia de Saneamento Ambiental do Distrito Federal - CAESB, por permitir e incentivar que seus funcionários busquem conhecimento e crescimentos profissional e pessoal.

Obrigado a todos que direta ou indiretamente contribuíram na realização desse sonho. Só o fato de acreditarem em mim, me impeliu a vencer.

RESUMO

OTIMIZAÇÃO META-HEURÍSTICA MULTIOBJETIVO APLICADA AO MÉTODO SEM MALHA LOCAL

Autor: Elvis Pereira de Santana

Orientador: Artur Portela

Programa de Pós-graduação em Estruturas e Construção Civil - PECC

Brasília, março de 2021

Esta pesquisa trata da otimização meta-heurística multiobjetivo dos parâmetros de discretização do método sem malha local, para a solução de problemas bidimensionais da elasticidade linear e da mecânica da fratura linear elástica. Para uma discretização nodal, o método sem malha local usa um processo nó a nó para gerar o sistema global das equações de equilíbrio; no domínio local de cada nó, as respectivas equações de equilíbrio são geradas com uma integração numérica reduzida e usadas como base da formulação do método numérico. A aplicação do método numérico local, para cada nó da discretização, requer o uso de dois parâmetros arbitrários que especificam o tamanho do suporte compacto e do domínio local de integração, cujos valores dependem do modelo estudado. A determinação desses parâmetros deve ocorrer dentro de certos limites, encontrados na literatura, para que não causem singularidades na solução numérica. Ao encontro disso, a popularidade dos algoritmos meta-heurísticos pode ser atribuída às suas excelentes características, como: a habilidade de otimizar problemas não suaves, não convexos, multimodais e altamente não lineares; além de simplicidade, flexibilidade e eficiência. Tais vantagens os fazem versáteis para lidarem com uma grande gama de problemas de otimização. Aqui, uma nova abordagem, com algoritmos meta-heurísticos multiobjetivo (NSGA-II, NSGA-III, CMAES, MOEA/D, IBEA, Epsilon-MOEA, SPEA2, GDE3, OMOPSO e SMPSO), já consagrados na comunidade científica, será utilizada na otimização dos parâmetros arbitrários do método sem malha local aplicado na solução de problemas da elasticidade linear e da mecânica da fratura linear elástica. Onde, o desempenho de cada algoritmo utilizado será mensurado a partir das métricas de performance (hipervolume, distância geracional, distância geracional invertida, indicador ϵ_+ e espaçamento) normalmente empregadas na avaliação dos algoritmos multiobjetivo. No final, um *ranking* de performance mostrará a adequação de cada algoritmo na otimização dos modelos/exemplos abordados. A influência da densidade nodal e do tipo (se regular ou randômica) da discretização do domínio, nos resultados da otimização dos modelos, também será avaliada.

Palavras-chave: Otimização meta-heurística multiobjetivo. Métricas de performance. Método sem malha local. Elasticidade linear. Mecânica da fratura. Subtração da singularidade. Fator de intensidade de tensão.

ABSTRACT

MULTIOBJECTIVE METAHEURISTIC OPTIMIZATION APPLIED TO LOCAL MESH FREE NUMERICAL METHOD

Author: Elvis Pereira de Santana

Supervisor: Artur Portela

Postgraduate Program in Structures and Civil Construction - PECC

Brasília, march of 2021

This research deals with multiobjective metaheuristic optimization of the discretization parameters of the local mesh free numerical method, to solve two-dimensional problems of linear elasticity and linear elastic fracture mechanics. For a nodal discretization, the local mesh free method uses a node-by-node process to generate the global system of equilibrium equations; in the local domain of each node, the respective equilibrium equations are generated with a reduced numerical integration and used as the formulation basis of the numerical method. The application of the local numerical method, for each node of discretization, requires the use of two arbitrary parameters that specify the size of the compact support and the local integration domain, whose values depend on the model studied. The determination of these parameters must occur within certain limits, found in the literature, such that they do not cause singularities in the numerical solution. In this context, the popularity of metaheuristic algorithms can be attributed to their excellent characteristics, such as: the ability to optimize non-smooth, non-convex, multimodal and highly non-linear problems; in addition to simplicity, flexibility and efficiency. These advantages make them versatile to deal with a wide range of optimization problems. Here, a new approach, with multiobjective metaheuristic algorithms (NSGA-II, NSGA-III, CMAES, MOEA/D, IBEA, Epsilon-MOEA, SPEA2, GDE3, OMOPSO e SMPSO), already well established in the scientific community, will be used in the optimization of the arbitrary parameters of the local mesh free numerical method applied to solve linear elasticity and linear elastic fracture mechanics problems. Where, the performance of each algorithm will be measured from the performance metrics (hypervolume, generational distance, inverted generational distance, ϵ_+ indicator and spacing) normally used in the multiobjective algorithms evaluation. In the end, a performance ranking will show the suitability of each algorithm in optimizing the models/examples covered. The influence of nodal density and the type (whether regular or random) of the domain discretization, on the models optimization results, will be assessed too.

Keywords: Multiobjective metaheuristic optimization. Performance metrics. Integrated local meshfree method. Linear elasticity. Fracture mechanics. Singularity subtraction. Stress intensity factor.

SUMÁRIO

LISTA DE TABELAS	xiii
LISTA DE FIGURAS	xiv
LISTA DE SÍMBOLOS E ABREVIATURAS	xx
1 INTRODUÇÃO	1
1.1 OBJETIVOS	7
1.1.1 Objetivos Gerais	7
1.1.2 Objetivos Específicos	8
1.2 ORGANIZAÇÃO DO TRABALHO	9
2 REVISÃO BIBLIOGRÁFICA	10
2.1 OTIMIZAÇÃO	10
2.1.1 Histórico	10
2.1.2 Problemas de Otimização	11
2.1.3 Métodos de Otimização	11
2.1.4 Meta-heurísticas - Breve Histórico	15
2.1.5 Descrição Sucinta dos Principais Métodos Meta-heurísticos	16
2.1.6 Comparativo dos Métodos de Otimização Estocásticos e Determinísticos	22
2.1.7 Otimização Mono-objetivo	25
2.1.8 Otimização Multiobjetivo	26
2.1.9 Métricas de Performance de um MMA	27
2.1.10 Otimização Estrutural	34
2.1.11 O Impacto Econômico e Ambiental da Otimização Estrutural	36
2.2 <i>PACKAGES, FRAMEWORKS E TOOLBOXES</i> DE OTIMIZAÇÃO	36
2.2.1 <i>Python Packages</i> para Otimização Multiobjetivo	36
2.2.2 <i>Paradiseo Framework</i>	37
2.2.3 <i>Toolbox</i> de Otimização Global do MATLAB	40
2.2.4 <i>Solver Add-in</i> do Excel	43
2.2.5 Linguagem de Programação e <i>Framework</i> de Otimização Adotadas	43
3 MÉTODOS SEM MALHA	48
3.1 DEFINIÇÃO	48
3.2 FUNÇÕES DE APROXIMAÇÃO	49
3.3 FUNÇÕES DE FORMA	50

3.4	FUNÇÕES PONDERADORAS	51
3.5	SUORTE COMPACTO	51
3.6	DOMÍNIO DE DEFINIÇÃO	52
3.7	IMPOSIÇÃO DAS CONDIÇÕES DE CONTORNO ESSENCIAIS	52
3.8	AVALIAÇÃO DAS INTEGRAIS	53
3.9	MÉTODOS SEM MALHA BASEADOS NA FORMA FRACA GLOBAL	54
3.10	MÉTODOS SEM MALHA BASEADOS NA FORMA FRACA LOCAL	55
3.11	FORMULAÇÃO DO MÉTODO SEM MALHA LOCAL	55
3.11.1	Método dos Mínimos Quadrados Móveis	56
3.11.2	Forma Local do Teorema do Trabalho	62
3.11.3	Campo Elástico Generalizado (GSMF)	65
4	MECÂNICA DA FRATURA LINEAR ELÁSTICA	70
4.1	CONCEITOS FUNDAMENTAIS	70
4.1.1	Modos de Deslocamento na Ponta da Trinca	70
4.1.2	Campo Elástico Local	70
4.1.3	Fator de Intensidade de Tensão	77
4.1.4	Postulado Fundamental	78
4.1.5	Campo Elástico na Ponta da Trinca	78
4.2	MÉTODO DA VISIBILIDADE, DA DIFRAÇÃO E DA TRANSPARÊNCIA	83
4.3	TÉCNICA DA SUBTRAÇÃO DA SINGULARIDADE	85
4.3.1	Campo Elástico Original	85
4.3.2	Campo Elástico Regularizado	86
4.3.3	Solução Particular Singular de Williams	87
4.3.4	Análise com a Formulação do Campo Elástico Generalizado (GSMF)	88
4.3.5	Equações Adicionais	90
5	FUNÇÕES OBJETIVO PARA MODELOS DO MÉTODO SEM MALHA LOCAL	92
5.1	ERRO RELATIVO	92
5.2	<i>COMPLIANCE</i>	92
5.3	FATOR DE INTENSIDADE DE TENSÃO (SIF)	93
5.4	COEFICIENTE DE CORRELAÇÃO DE KARL PEARSON	95
6	METODOLOGIA	99
7	RESULTADOS E DISCUSSÕES	102
7.1	ABORDAGEM INICIAL PARA A OTIMIZAÇÃO DOS PARÂMETROS DE DISCRETIZAÇÃO DO MÉTODO SEM MALHA LOCAL	102
7.1.1	Otimização Baseada no Erro Relativo	103

7.1.2	Otimização Baseada na Conformidade Estrutural	109
7.1.3	Otimização Baseada no Domínio Local	113
7.1.4	Otimização Automática dos Parâmetros de Discretização	114
7.2	NOVA ABORDAGEM PROPOSTA PARA A OTIMIZAÇÃO DOS MODELOS DO MÉTODO SEM MALHA LOCAL	117
7.2.1	Equivalência Entre o <i>Compliance</i> e o Erro Relativo ao Campo de Deslocamentos	118
7.2.2	Influência da Discretização Nodal do Domínio nos Resultados do ILMF e da Otimização	129
7.2.3	Ranqueamento dos MMAs Através da <i>Utility Function</i> e as Métricas de Performance	129
7.3	APLICAÇÃO DA NOVA ABORDAGEM DE OTIMIZAÇÃO NA MECÂNICA DA FRATURA LINEAR ELÁSTICA	136
7.3.1	Placa com Trinca Horizontal de Borda - Modo-I	136
7.3.2	Placa com Trinca Horizontal de Borda - Modo-II	152
7.3.3	Placa com Trinca Inclinada de Borda – Modo Misto	168
7.3.4	<i>Ranking</i> de Performance dos MMAs - Fratura	184
7.4	<i>RANKING</i> GLOBAL DE PERFORMANCE DOS MMAs - ELASTICIDADE LINEAR + MECÂNICA DA FRATURA	185
8	CONCLUSÕES E RECOMENDAÇÕES	187
8.1	CONCLUSÕES	187
8.2	SUGESTÕES PARA TRABALHOS FUTUROS	188
	REFERÊNCIAS BIBLIOGRÁFICAS	196
	APÊNDICES	197
A	PARÂMETROS DE DISCRETIZAÇÃO α_s e α_q	198
B	TIPOS INCOMUNS DE GRÁFICOS UTILIZADOS	199
B.1	<i>PARALLEL COORDINATES</i>	199
B.2	<i>BOXPLOT</i>	201
B.3	<i>VIOLIN</i>	202
B.4	<i>HEATMAP</i>	203
B.5	DISPERSÃO COM REGRESSÃO LINEAR E DISTRIBUIÇÃO DE FREQUÊNCIAS	203
B.6	<i>PAIRWISE</i>	204
C	PUBLICAÇÕES	207
C.1	ARTIGOS PUBLICADOS	208

LISTA DE TABELAS

2.1	Classificação dos problemas de otimização.	13
2.2	Comparativo sintético dos métodos de otimização estocásticos e determinísticos.	23
2.3	Funções <i>benchmark</i> clássicas para a avaliação de algoritmos de otimização. Modificado de Saad et al. (2017).	25
2.4	<i>Frameworks</i> de otimização mais populares, escritas em Python (modificado de Benítez-Hidalgo et al. (2019)).	38
2.5	Seleção do <i>solver</i> na <i>toolbox</i> a partir dos tipos da função objetivo e das restrições e, da solução desejada.	41
3.1	Métodos sem malha classificados. Adaptado de Liu e Gu (2005).	49
3.2	Principais funções de forma e suas classificações.	50
6.1	Variáveis e funções objetivo para cada problema estudado.	99
7.1	Os resultados da frente de Pareto multiobjetivo para uma distribuição nodal regular com 52 nós, referentes à Figura 7.3a	106
7.2	Os resultados da frente de Pareto multiobjetivo para uma distribuição nodal regular com 165 nós, referentes à Figura 7.3b	107
7.3	Os resultados da frente de Pareto multiobjetivo para uma distribuição nodal regular com 585 nodes, referente à Figura 7.3c	107
7.4	Os resultados da frente de Pareto multiobjetivo para uma distribuição nodal regular com 52 nós para $\alpha_q = 0.5$ fixo, referente à Figura 7.9a	112
7.5	Os resultados da frente de Pareto multiobjetivo para uma distribuição nodal regular com 165 nós para $\alpha_q = 0.5$ fixo, referente à Figura 7.9b	112
7.6	Os resultados da frente de Pareto multiobjetivo para uma distribuição nodal regular com 585 nós para $\alpha_q = 0.5$ fixo, referente à Figura 7.9c	112
7.7	Os resultados da frente de Pareto multiobjetivo da rotina totalmente automatizada, para uma distribuição nodal regular com 52 nós e diferentes valores de α_q para cada nó da discretização sem malha, referente à Figura 7.11a	115
7.8	Os resultados da frente de Pareto multiobjetivo da rotina totalmente automatizada, para uma distribuição nodal regular com 165 nós e diferentes valores de α_q para cada nó da discretização sem malha, referente à Figura 7.11b	115

7.9	Os resultados da frente de Pareto multiobjetivo da rotina totalmente automatizada, para uma distribuição nodal regular com 585 nós e diferentes valores de α_q para cada nó da discretização sem malha, referente à Figura 7.11c	116
7.10	Descrição estatística dos resultados escalados - 1ª fase.	121
7.11	Descrição estatística dos resultados escalados - 2ª fase.	121
7.12	<i>Ranking</i> dos MMAs a partir das métricas de performance escaladas.	135
7.13	Placa retangular com uma trinca de borda única sob carregamento em modo-I. GSMF representa os valores obtidos na presente pesquisa e J-DBEM representa os valores obtidos com o DBEM utilizando a integral J (Aliabadi e Portela, 1999). A porcentagem dos erros foi calculada a partir do valor de referência obtido em Civelek e Erdogan (1982a).	138
7.14	Descrição estatística dos resultados escalados - 1ª fase.	141
7.15	Descrição estatística dos resultados escalados - 2ª fase.	141
7.16	<i>Ranking</i> dos MMAs a partir das métricas de performance escaladas.	149
7.17	Placa retangular com uma trinca de borda única sob carregamento em modo-II. GSMF representa os valores obtidos na presente pesquisa e J-DBEM representa os valores obtidos com o DBEM utilizando a integral J (Aliabadi e Portela, 1999). A porcentagem dos erros foi calculada a partir dos valores obtidos com o J-DBEM.	154
7.18	Descrição estatística dos resultados escalados - 1ª fase.	157
7.19	Descrição estatística dos resultados escalados - 2ª fase.	157
7.20	<i>Ranking</i> dos MMAs a partir das métricas de performance escaladas.	166
7.21	Placa retangular com uma trinca inclinada de $\alpha = 30^\circ$. GSMF representa os valores obtidos na presente pesquisa e J-DBEM representa os valores obtidos com o DBEM utilizando a integral J (Aliabadi e Portela, 1999). A porcentagem dos erros foi calculada a partir do valor de referência obtido em Murakami (1986).	170
7.22	Placa retangular com uma trinca inclinada de $\alpha = 30^\circ$. GSMF representa os valores obtidos na presente pesquisa e J-DBEM representa os valores obtidos com o DBEM utilizando a integral J (Aliabadi e Portela, 1999). A porcentagem dos erros foi calculada a partir do valor de referência obtido em Murakami (1986).	170
7.23	Placa retangular com uma trinca inclinada de $\alpha = 60^\circ$. GSMF representa os valores obtidos na presente pesquisa e J-DBEM representa os valores obtidos com o DBEM utilizando a integral J (Aliabadi e Portela, 1999). A porcentagem dos erros foi calculada a partir do valor de referência obtido em Murakami (1986).	170

7.24	Placa retangular com uma trinca inclinada de $\alpha = 60^\circ$. GSMF representa os valores obtidos na presente pesquisa e J-DBEM representa os valores obtidos com o DBEM utilizando a integral J (Aliabadi e Portela, 1999). A porcentagem dos erros foi calculada a partir do valor de referênci obtido em Murakami (1986).	170
7.25	Descrição estatística dos resultados escalados - 1ª fase.	173
7.26	Descrição estatística dos resultados escalados - 2ª fase.	173
7.27	<i>Ranking</i> dos MMAs a partir das métricas de performance escaladas.	181
7.28	<i>Ranking</i> de performance dos MMAs - fratura.	185
7.29	<i>Ranking</i> global de performance dos MMAs - elasticidade linear + mecânica da fratura linear elástica	185

LISTA DE FIGURAS

2.1	Cronologia sucinta das técnicas de otimização, segundo Gandomi et al. (2013).	12
2.2	Classificação das técnicas de otimização, segundo Yang (2010).	14
2.3	Fluxograma geral dos passos executados em algoritmo genético.	18
2.4	Processo de ACO na forma de grafo multicamadas (Gandomi et al., 2013).	20
2.5	Representação esquemática do movimento de uma partícula em PSO, movendo-se em direção ao ótimo global g^* e à melhor posição corrente x_i^* de cada partícula i (Yang, 2010).	21
2.6	Funções <i>benchmark</i> clássicas para a avaliação de algoritmos de otimização. Modificado de Saad et al. (2017).	24
2.7	Representação de várias regiões da dominância de Pareto, no caso de minimização de um MOP. Estas regiões são relativas a cada solução, que é centrada na figura. A região dominada é inferior em todos os objetivos, a região dominante é superior em todos os objetivos e a região não dominada é superior em um objetivo, mas, inferior no outro, (Hadka, 2016).	27
2.8	Representação do mapeamento hipotético entre um <i>Pareto optimal set</i> 3-dimensional e sua <i>Pareto front</i> 2-dimensional associada. A área sombreada/cinza (<i>Pareto front</i>) mostra o espaço dominado pela <i>Pareto front</i> , (Hadka, 2016).	28
2.9	Representação esquemática do hipervolume (Hadka, 2016).	29
2.10	Representação esquemática da distância geracional (Hadka, 2016).	30
2.11	Representação esquemática da distância geracional invertida (Hadka, 2016).	30
2.12	Representação esquemática do indicador ϵ_+ (Hadka, 2016).	31
2.13	Representação esquemática do espaçamento (Hadka, 2016).	31
2.14	Análise conjunta das métricas de performance consideradas (Hadka, 2016).	32
2.15	Módulos da <i>Paradiseo framework</i> e suas interações.	39
2.16	Interface da <i>toolbox</i> de otimização global do MATLAB.	42
2.17	Interface do <i>Solver, add-in</i> do Excel.	44
3.1	Representação do domínio nos métodos sem malha e no Método dos Elementos Finitos (MEF).	48
3.2	Suporte compacto do ponto de interesse em I em diferentes modelos de métodos sem malha. Adaptado de Chen et al. (2006).	52
3.3	Uso de <i>background cells</i> em métodos sem malha baseados na forma fraca global. Adaptado de Liu e Gu (2005).	53
3.4	Domínio local Ω_S de um nó x . Adaptado de Atluri et al. (2004).	54

3.5	Representação de uma discretização do domínio global Ω usando o método sem malha, com um contorno $\Gamma = \Gamma_u + \Gamma_t$, com uma distribuição nodal \mathbf{x}_i . Ω_s , representado como Ω_P , Ω_Q e Ω_R , é o suporte compacto do nó; Ω_x é o domínio de definição de um ponto qualquer \mathbf{x} e Ω_q é o domínio da forma fraca local ou domínio de quadratura do nó \mathbf{x}_i	56
3.6	Exemplo unidimensional da aproximação pelos MQM, onde $u^h(x_i) \neq \hat{u}_i$	57
3.7	Típica função peso e função de forma de uma aproximação pelos MQM para um nó $x = [1/2 \ 0]^T$	59
3.8	Representação do domínio global Ω , suas fronteiras natural Γ_t e essencial Γ_u e os subdomínios Ω_Q associados ao nó Q , com contorno interno $\Gamma_Q = \Gamma_{Qi} \cup \Gamma_{Qt} \cup \Gamma_{Qu}$. Nós P e R, de forma similar, possuem domínios locais correspondentes a Ω_P e Ω_R	63
3.9	Esquema representando o deslocamento $\mathbf{u}^*(\mathbf{x})$, dado pela equação (3.40), um deslocamento unitário de corpo rígido discreto definido nos pontos de colocação, da formulação do GSMF, para um domínio local arbitrário associado ao nó Q	67
3.10	Esquema representando o equilíbrio das forças de corpo e forças de superfície, da equação (3.46), pontualmente definidas nos pontos de colocação de um domínio local associado ao nó Q , da formulação do GSMF.	69
4.1	Modos de deslocamento da ponta da trinca básicos de um corpo trincado.	71
4.2	Geometria de um entalhe anguloso no sistema de coordenadas polares e cartesianas.	72
4.3	Representação gráfica das singularidades de Williams, na ponta da trinca. Caso de trinca horizontal de borda - Modo-I puro. Tensões σ_{xx} , σ_{yy} e σ_{xy}	80
4.4	Representação gráfica das singularidades de Williams, na ponta da trinca. Caso de trinca horizontal de borda - Modo-II puro. Tensões σ_{xx} , σ_{yy} e σ_{xy}	81
4.5	Domínio local de um nó x_i , truncado na face de uma trinca, e um nó x_j , truncado na face de uma trinca e próximo à ponta da trinca, para o método da visibilidade.	83
4.6	Domínio local de um nó x_i , truncado na face de uma trinca e próximo à ponta da trinca, para o método da difração.	84
4.7	Domínio local de um nó x_i , truncado na face de uma trinca e próximo à ponta da trinca, para o método da transparência.	85
4.8	Componentes das forças de superfície de um campo elástico regularizado na ponta de uma trinca.	91
5.1	O mínimo de $f(x)$ é o mesmo que o máximo de $-f(x)$ (Rao, 2019).	94
5.2	Vários conjuntos de pontos de dispersões de x e y , com os respectivos valores de ρ	97

7.1	Estado Plano de Tensão - Viga de Timoshenko engastada de Largura Unitária.	102
7.2	Distribuição regular dos nós de uma viga engastada em balanço, discretizada com 52, 165 e 585 nós.	104
7.3	A frente de Pareto da otimização multiobjetivo para três distribuições nodais regulares da viga engastada em balanço, em função do erro relativo r_ε , do tempo de processamento (<i>CPU Time</i>) em segundos e do erro relativo r_u ; apenas os dois primeiros foram apresentados graficamente.	106
7.4	Deslocamento vertical e horizontal normalizados em uma viga engastada com uma discretização de $33 \times 5 = 165$ nós.	108
7.5	Distribuição de tensões de uma viga engastada em $x_1 = L/2$ e $x_2 \in [-D/2, D/2]$ com uma discretização de $33 \times 5 = 165$ nós.	108
7.6	Comparação entre o indicador de conformidade (C) e o Erro relativo r_ε , para uma viga engastada em balanço com uma discretização de $13 \times 4 = 52$ nós, para um valor fixo $\alpha_q = 0.5$	109
7.7	Comparação entre o indicador de conformidade (C) e o Erro relativo r_ε , para uma viga engastada em balanço com uma discretização de $33 \times 5 = 165$ nós, para um valor fixo $\alpha_q = 0.5$	109
7.8	Comparação entre o indicador de conformidade (C) e o Erro relativo r_ε , para uma viga engastada em balanço com uma discretização de $65 \times 9 = 585$ nós, para um valor fixo $\alpha_q = 0.5$	110
7.9	A frente de Pareto da otimização multiobjetivo para três distribuições nodais regulares da viga engastada em balanço, em função do indicador de conformidade C e do tempo de processamento (<i>CPU Time</i>) em segundos.	111
7.10	Erro relativo r_ε em função da relação A_q , entre o somatório das áreas dos domínios locais dos nós na discretização sem malha com a área do domínio global do problema, para $\alpha_s = 4.5$ fixo e número máximo de gerações de 100, 200, 400 e 800; para uma distribuição nodal com 165 nós.	114
7.11	A frente de Pareto da otimização multiobjetivo para três distribuições nodais regulares da viga engastada em balanço, em função do indicador de conformidade C , do tempo de processamento (<i>CPU Time</i>) em segundos e da relação A_q , para a rotina totalmente automatizada de otimização.	116
7.12	Deslocamento vertical e horizontal normalizados em uma viga engastada com uma discretização de $33 \times 5 = 165$ nós, utilizando a rotina totalmente automatizada de otimização.	117
7.13	Distribuição de tensões de uma viga engastada em $x_1 = L/2$ e $x_2 \in [-D/2, D/2]$ com uma discretização de $33 \times 5 = 165$ nós, utilizando a rotina totalmente automatizada de otimização.	118
7.14	Discretizações nodais regulares e randômicas utilizadas.	119
7.15	Escalagem dos dados - 1ª fase.	120

7.16	Escalagem dos dados - 2ª fase.	120
7.17	Frequências relativas dos parâmetros da otimização.	122
7.18	Mapa de calor dos parâmetros da otimização em função do coeficiente de correlação.	124
7.19	<i>Pairwise plot</i> - 1ª fase.	125
7.20	<i>Pairwise plot</i> - 2ª fase.	126
7.21	Correlação entre err_U e <i>compliance</i>	127
7.22	Coordenadas paralelas dos parâmetros de otimização.	128
7.23	Influência no <i>compliance</i> , da densidade nodal da discretização do domínio (meshSize = 0.0/0.5/1.0 para 52/165/585 nós, respectivamente).	130
7.24	Influência no <i>compliance</i> e nas métricas de performance; devida ao tipo da discretização nodal do domínio (meshType = 0.0, discretização regular; meshType = 1.0, discretização randômica).	131
7.25	Quantidade de soluções únicas, não dominadas e ótimas de Pareto, por MMA utilizado - 2ª fase.	132
7.26	<i>Pairwise plot</i> dos parâmetros da otimização, 2ª fase, com as soluções discriminadas por MMA.	133
7.27	Exploração dos espaços das variáveis e das soluções pelos MMAs.	134
7.28	Métricas de performance dos MMAs plotadas em coordenadas paralelas.	135
7.29	Placa retangular com uma única trinca de borda sob carregamento em modo-I ($h/w = 0.5$).	137
7.30	Discretização de uma placa com uma distribuição nodal regular de $10 \times 10 = 100$ nós, com nós sobrepostos adicionais ao longo da trinca, para $a/w = 0.5$. A linha vermelha representa as faces da trinca.	137
7.31	Configuração deformada da placa, para $a/w = 0.5$, sob carregamento em modo-I. Os nós vermelhos representam os nós sobrepostos ao longo da trinca e na ponta da trinca.	138
7.32	Escalagem dos dados - 1ª fase.	140
7.33	Escalagem dos dados - 2ª fase.	140
7.34	Frequências relativas dos parâmetros da otimização.	142
7.35	Mapa de calor dos parâmetros da otimização em função do coeficiente de correlação.	144
7.36	<i>Pairwise plot</i> - 1ª fase.	145
7.37	<i>Pairwise plot</i> - 2ª fase.	146
7.38	Correlação entre K_I e <i>compliance</i>	147
7.39	Coordenadas paralelas dos parâmetros de otimização - 2ª fase.	148
7.40	Quantidade de soluções únicas, não dominadas e ótimas de Pareto, por MMA utilizado - 2ª fase.	148

7.41	<i>Pairwise plot</i> dos parâmetros da otimização, 2ª fase, com as soluções discriminadas por MMA.	150
7.42	Exploração dos espaços das variáveis e das soluções pelos MMAs.	151
7.43	Métricas de performance dos MMAs plotadas em coordenadas paralelas. . .	152
7.44	Placa retangular com uma única trinca de borda sob carregamento em modo-II ($w = 2h$).	153
7.45	Discretização de uma placa com uma distribuição nodal regular de $16 \times 16 = 256$ nós, com nós sobrepostos adicionais ao longo da trinca, para $a/w = 0.5$. A linha vermelha representa as faces da trinca.	154
7.46	Configuração deformada da placa, para $a/w = 0.5$, sob carregamento em modo-II. Os nós vermelhos representam os nós sobrepostos ao longo da trinca e na ponta da trinca.	155
7.47	Escalagem dos dados - 1ª fase.	156
7.48	Escalagem dos dados - 2ª fase.	156
7.49	Frequências relativas dos parâmetros da otimização.	158
7.50	Mapa de calor dos parâmetros da otimização em função do coeficiente de correlação.	160
7.51	<i>Pairwise plot</i> - 1ª fase.	161
7.52	<i>Pairwise plot</i> - 2ª fase.	162
7.53	Correlação entre K_{II} e <i>compliance</i>	163
7.54	Coordenadas paralelas dos parâmetros de otimização - 2ª fase.	164
7.55	Quantidade de soluções únicas, não dominadas e ótimas de Pareto, por MMA utilizado - 2ª fase.	164
7.56	<i>Pairwise plot</i> dos parâmetros da otimização, 2ª fase, com as soluções discriminadas por MMA.	165
7.57	Exploração dos espaços das variáveis e das soluções pelos MMAs.	167
7.58	Métricas de performance dos MMAs plotadas em coordenadas paralelas. . .	168
7.59	Placa retangular com uma única trinca inclinada de borda sob tração remota. .	169
7.60	Discretização de uma placa com uma distribuição nodal regular de $16 \times 16 = 256$ nós, com nós sobrepostos adicionais ao longo da trinca, para $a/w = 0.5$. A linha vermelha representa as faces da trinca.	169
7.61	Configuração deformada da placa, para $a/w = 0.5$, sob carregamento em modo misto. Os nós vermelhos representam os nós sobrepostos ao longo da trinca e na ponta da trinca.	171
7.62	Escalagem dos dados - 1ª fase.	172
7.63	Escalagem dos dados - 2ª fase.	173
7.64	Frequências relativas dos parâmetros da otimização.	175
7.65	Mapa de calor dos parâmetros da otimização em função do coeficiente de correlação.	176

7.66	<i>Pairwise plot</i> - 1ª fase.	177
7.67	<i>Pairwise plot</i> - 2ª fase.	178
7.68	Correlação entre K_I (e K_{II}) e <i>compliance</i>	179
7.69	Coordenadas paralelas dos parâmetros de otimização - 2ª fase.	180
7.70	Quantidade de soluções únicas, não dominadas e ótimas de Pareto, por MMA utilizado - 2ª fase.	181
7.71	<i>Pairwise plot</i> dos parâmetros da otimização, 2ª fase, com as soluções discriminadas por MMA.	182
7.72	Exploração dos espaços das variáveis e das soluções pelos MMAs.	183
7.73	Métricas de performance dos MMAs plotadas em coordenadas paralelas.	184
B.1	Exemplo de gráfico com coordenadas paralelas.	200
B.2	Exemplo de gráfico com coordenadas paralelas e frequências das ocorrências. (Modificado de Data Viz Project).	201
B.3	Exemplo de diagrama de caixa e suas medidas descritivas. (Em KDnuggets).	201
B.4	Similaridades entre o diagrama de caixa e uma curva de distribuição. (Em KDnuggets).	202
B.5	Exemplo de gráfico violino. (Em MDPI).	203
B.6	Exemplo de gráfico mapa de calor.	204
B.7	Exemplo de gráfico de dispersão com regressão linear e distribuição de frequências.	205
B.8	Exemplo de gráficos emparelhados.	206

LISTA DE SÍMBOLOS E ABREVIATURAS

Símbolos do Alfabeto Grego

Γ_t	Contorno estático
Γ_u	Contorno cinemático
μ	modulo de cisalhamento
ν	Coefficiente de Poisson
$\phi_i(\mathbf{x})$	Função de forma da aproximação com mínimos quadrados móveis
α_q	Parâmetro de discretização do domínio do método sem malha, relacionado ao domínio local (ou domínio da forma fraca)
α_s	Parâmetro de discretização do domínio do método sem malha, relacionado ao suporte compacto
ϵ^*	Campo de deformações cinematicamente admissível
ϵ_+	Indicador de performance epsilon aditivo
σ_{ij}^R	Componentes regulares do campo de tensões
σ_{ij}^S	Componentes singulares do campo de tensões

Símbolos do Alfabeto Latino

\hat{u}_i	Parâmetros nodais
$\mathbf{a}(\mathbf{x})$	Vetor de coeficientes
d_i	Distância entre um ponto qualquer x e o nó i
r_i	Tamanho do suporte compacto para o nó i
$w(\mathbf{x})$	Função peso
\mathbf{B}	Matriz dos operadores lineares
\mathbf{D}	Matriz constitutiva do material
\mathbf{J}^{-1}	inverso do Jacobiano
\mathbf{J}	Jacobiano

\mathbf{k}_m	Vetor contendo os fatores de intensidade de tensão (SIF)
\mathbf{n}	Matriz das componentes do vetor normal unitário
\mathbf{p}	Base polinomial
\mathbf{U}	Vetor contendo os deslocamentos nodais prescritos
a	Comprimento da trinca
E	Módulo de elasticidade longitudinal
G	Módulo de elasticidade transversal
h	Altura da placa
I	Momento de inércia da seção transversal
$u^h(\mathbf{x})$	Deslocamento aproximado
W	Largura da placa
\mathbf{b}	Vetor forças de corpo
\mathbf{t}	Componentes das forças superficiais
\mathbf{u}^*	Campos de deslocamento admissível
$\bar{\mathbf{t}}$	Valores das forças superficiais prescritas
$\bar{\mathbf{u}}$	Valores dos deslocamentos prescritos
$H(d)$	Heaviside step function
K_{II}	Fator de intensidade de tensão (SIF) do Modo II - cisalhamento
K_I	Fator de intensidade de tensão (SIF) do Modo I - abertura
u_i^R	Componentes regulares do campo de deformações
u_i^S	Componentes singulares do campo de deformações
\mathbf{F}	Matriz das componentes de Williams do deslocamento singular no contorno, dada uma ponta de trinca
\mathbf{G}	Matriz das componentes de Williams das forças de superfície singulares no contorno, dada uma ponta de trinca

Abreviaturas

ϵ_+ Additive ϵ -indicator

p_c Probabilidade de crossover, em GA

p_m Probabilidade de mutação, em GA

a.C. Antes de Cristo

ABC Colônia artificial de abelhas

ACO Ant Colony Optimization

BA Bat Algorithm

BEA Bee Algorithms

BEM Boundary Element Method

C Compliance ou conformidade estrutural

CAPES Coordenação de Aperfeiçoamento de Pessoal de Nível Superior

CMA-ES Algoritmo de estratégia de evolução e adaptação com matriz de covariância

CS Cuckoo Search

DE Differential Evolution

EEM Eigenfunction Expansion Method

ELSEVIER Editora holandesa de literatura médica e científica

EO Evolving objects

Epsilon-MOEA Algoritmo evolucionário multiobjetivo de estado estacionário que usa arquivo de dominância

FA Firefly Algorithm

FEM Finite Element Method

FPA Flower Pollination Algorithm

GA Genetic Algorithms

GD Generational distance

GDE3 Terceira versão do algoritmo de evolução diferencial generalizada

GEE Gases de efeito estufa

GFDM Generalized Finite Difference Method ou Método da diferença finita generalizada

GP Genetic Programming

GRG Generalized Reduced Gradient

GSMF Generalized-Strain Mesh-Free formulation ou formulação do campo elástico generalizado do método sem malha local

HS Harmony Search

IBEA Algoritmo evolucionário multiobjetivo à base do indicador hipervolume

IGD Inverted generational distance

ILMF Integrated Local Meshfree Method

MATLAB Matrix Laboratory

MEC Método dos Elementos de Contorno

MEF Método dos Elementos Finitos

MLPG Meshless Local Petrov–Galerkin

MLPG-FVM Meshless Local Petrov-Galerkin Finite Volume Method

MLS Mean least square

MMA Multiobjective metaheuristic algorithm ou algoritmo meta-heurístico multiobjetivo

MO Moving objects

MOEA/D Algoritmo evolucionário multiobjetivo com decomposição do problema em formulações mono-objetivo

MOEO Multiobjective evolving objects

MOP Multiobjective Problem

MP Mathematical Programming

MPI Message passing interface

MQM Mínimos quadrados móveis

NSGA-II Algoritmo evolucionário multiobjetivo

NSGA-III Algoritmo evolucionário multiobjetivo, sucessor do NSGA-II

OC Optimality Criteria

OMOPSO Algoritmo multiobjetivo de enxame de partículas que usa arquivo de dominância

OOP Object-Oriented Programming

PECC Pós-graduação em Estruturas e Construção Civil / UnB

PEO Parallel evolving objects

PIM Point Interpolation Method

PSO Particle Swarm Optimization

RBDMF Rigid-Body Displacement Mesh-Free formulation ou formulação de deslocamento de corpo rígido do método sem malha local

RPIM Radial Point Interpolation Method

SA Simulated Annealing

SIF Stress Intensity Factor

SMPSO Algoritmo multiobjetivo de enxame de partículas

SPEA2 Algoritmo evolucionário multiobjetivo com ranqueamento *strength-based*

SST Singularity Subtraction Technique

TS Tabu Search

VBA Visual Basic for Applications

1 - INTRODUÇÃO

Otimização é a ciência da escolha de valores para variáveis, em conformidade com dadas restrições, tal que uma função, por exemplo, custo total seja minimizado ou a confiabilidade global do sistema seja maximizada (Kaveh, 2017).

Quase todos os problemas de projeto em engenharia podem ser vistos como problemas de otimização e, portanto, requerem técnicas de otimização para a sua adequada solução. Todavia, como a maioria dos problemas do mundo real são altamente não lineares, métodos tradicionais de otimização não são muito eficazes na sua solução. A tendência atual é usar algoritmos evolucionários meta-heurísticos para lidar com esses problemas de otimização não lineares.

A palavra *heurística* tem sua origem no antigo trabalho grego *heuriskein*, que significa a arte de descobrir novas estratégias ou regras para resolver problemas. O prefixo *meta*, também um termo grego, passou a significar um maior nível de abstração, no idioma inglês. O termo *metaheuristic* foi introduzido por Glover e Kochenberger (2003) e denota uma estratégia de resolver um problema usando níveis mais altos de abstrações para guiar uma busca heurística no espaço de soluções (Kaveh, 2017).

Algoritmos meta-heurísticos ganharam uma grande popularidade nos últimos anos. Estes incluem algoritmos genéticos (GA - *Genetic Algorithms*), otimização com enxame de partículas (PSO - *Particle Swarm Optimization*), busca cuco (CS - *Cuckoo Search*), algoritmo de morcegos (BA - *Bat Algorithm*), evolução diferencial (DE - *Differential Evolution*), algoritmo de vaga-lumes (FA - *Firefly Algorithm*), busca harmônica (HS - *Harmony Search*), algoritmo de polinização de flores (FPA - *Flower Pollination Algorithm*), otimização com colônia de formigas, (ACO - *Ant Colony Optimization*), algoritmos de abelhas (BEA - *Bee Algorithms*) e muitos outros. A popularidade dos algoritmos meta-heurísticos inspirados na natureza pode ser atribuída às suas excelentes características, como: simplicidade, flexibilidade, eficiência, adaptabilidade e ainda são de fácil implementação. Tais vantagens os fazem versáteis para lidarem com uma grande gama de problemas de otimização sem a exigência, a priori, de muita informação pertinente aos mesmos, conforme relata Yang et al. (2016).

Os Algoritmos meta-heurísticos desempenham um importante papel quando se trata de projeto ótimo de problemas complexos da engenharia, onde abordagens analíticas e métodos tradicionais não são efetivos na solução de problemas não lineares. De forma geral, esses problemas de projeto são altamente não lineares e com restrições complexas e, portanto, são

também multimodais. Estas restrições de projeto normalmente são requisitos técnicos e de medidas de segurança, tais como tensões em membros estruturais devidas a carregamento externo, fatores ambientais e usabilidade sob carregamentos de serviço. Uma solução matemática pode ser a melhor abordagem em um mundo ideal, mas, em projetos de engenharia, os valores de uma variável de projeto, tal como a massa ou comprimento, deve ser realista; por exemplo, algumas quantidades devem ser não negativas. Em adição, tais valores de projeto devem corresponder a algo que possa ser fabricado na prática.

Para Yang et al. (2016), otimização é crucialmente importante no processo de projeto para encontrar um bom equilíbrio entre economia e segurança, que são as metas primárias de projetos, em todas as disciplinas de engenharia. Estética e praticidade são importantes também nas aplicações do mundo real. Otimização pode ser mais relevante na engenharia civil, principalmente em estruturas, do que em outros ramos da engenharia. Por exemplo, no projeto de um componente não crítico de uma máquina em engenharia mecânica, as tensões na peça não podem exceder certos limites. Utilizar uma peça mais resistente pode ser muito dispendioso. Por outro lado, uma peça menos resistente pode ser capaz de fazer a máquina funcionar adequadamente, mas, com o tempo, tais peças podem se desgastarem ou serem danificadas. Entretanto, pode ser fácil substituí-las e a baixo custo. Se for este o caso, a usabilidade da máquina pode ser mantida na prática. Mas, em engenharia civil a integridade estrutural e a segurança podem impor restrições rigorosas nos membros estruturais que podem não ser de fácil substituição. Nesses casos, todas as restrições de projeto e o melhor equilíbrio possível entre segurança e economia deve ser encontrado sem colocar vidas em risco. Além disso, às vezes, a pequena melhora pode não ser tão importante quanto a robustez das aplicações. Um projeto robusto deveria ser capaz de lidar com incertezas em termos de propriedades de materiais, tolerâncias de fabricação e irregularidade de carregamento em serviço. Devido à complexidade e ao grande número de restrições no projeto de estruturas, métodos tradicionais lutam para lidarem com a alta não linearidade e multimodalidade, e muitas vezes fracassam. Assim, os métodos de otimização meta-heurística tornaram-se ferramentas importantes no projeto ótimo de engenharia estrutural.

Algumas das técnicas de otimização algorítmicas clássicas, que foram mais comumente utilizadas, são baseadas em gradiente e permitem uma busca no espaço de solução próxima a um dado ponto de partida onde informações de gradiente da função objetivo estão disponíveis. Métodos baseados em gradiente, em geral, convergem mais rapidamente e obtêm soluções mais precisas do que as abordagens estocásticas modernas. Entretanto, a aquisição de informação de gradiente para a função objetivo pode ser difícil ou mesmo impossível. Além disso, só a convergência para um mínimo local é garantida nesses tipos de algoritmos. Também, um bom ponto de partida pode ser vital para se ter sucesso na execução desses métodos. Em muitos problemas de otimização, *feasible/infeasible regions*, limites laterais e funções não suaves e não convexas precisam ser levados em consideração,

aumentando com isso a dificuldade na obtenção de soluções ótimas. Existe uma enorme quantidade de métodos de otimização meta-heurísticos, desenvolvidos mais recentemente, que não são restritos da maneira acima mencionada. Estes métodos são adequados para buscas globais sobre todo o espaço de busca devido à sua capacidade de explorar e encontrar regiões promissoras no espaço de busca com um esforço computacional razoável. Algoritmos meta-heurísticos tendem a apresentar um bom desempenho para a maioria dos problemas de otimização. Isso ocorre porque esses métodos evitam simplificar ou fazer suposições sobre o problema original. Evidências disso podem ser vistas em aplicações bem sucedidas a uma vasta variedade de áreas, como engenharia, física, química, artes, economia, marketing, genética, pesquisa operacional, robótica, ciências sociais e políticas, entre outras.

Um método heurístico pode ser considerado como um procedimento que provavelmente descobrirá uma solução viável muito boa, mas, não necessariamente uma solução ótima, para um problema específico considerado. Na maioria dos casos nenhuma garantia é fornecida quanto à qualidade da solução obtida, mas, um método heurístico bem projetado geralmente pode fornecer uma solução que está nas proximidades da ótima. O procedimento também deve ser suficientemente eficiente para lidar com problemas de larga escala. Os métodos heurísticos são geralmente algoritmos iterativos, em que cada iteração envolve a busca por uma nova solução que pode ser melhor do que a melhor solução encontrada na iteração anterior. Após um período de tempo razoável, quando o algoritmo é finalizado, a solução fornecida é a melhor encontrada durante todas as iterações. Uma meta-heurística pode ser definida também como um processo de geração iterativo que guia uma heurística subordinada, combinando inteligentemente conceitos diferentes para explorar (busca global) e explorar (busca local) o espaço de busca para encontrar eficientemente soluções quase ótimas. Estratégias de aprendizado podem ser empregadas para adicionar a “inteligência” a essas heurísticas de busca guiada, veja Kaveh (2017).

Segundo Gandomi et al. (2013), as meta-heurísticas geralmente desempenham melhor do que simples heurísticas. Todos os algoritmos meta-heurísticos usam uma combinação de busca local e exploração global. A variedade de soluções é sempre alcançada via randomização. Apesar da popularidade dos métodos meta-heurísticos, não há acordo sobre a definição de heurísticas e meta-heurísticas na literatura. Alguns pesquisadores usam heurísticas e meta-heurísticas de forma intercambiável. Entretanto, a recente tendência é de nomear todos os algoritmos estocásticos com randomização e exploração global como meta-heurísticos. A randomização fornece uma boa maneira de se afastar da busca local em direção à busca na escala global. Portanto, quase todos os algoritmos meta-heurísticos são normalmente adequados para a modelagem não linear e otimização global.

Meta-heurísticas podem ser uma forma eficiente de usar tentativa e erro para produzir

soluções aceitáveis para um problema complexo em um prático e razoável tempo. A complexidade do problema de interesse faz com que seja impossível buscar cada solução possível ou combinação; o objetivo é encontrar soluções boas e viáveis em uma escala de tempo aceitável. Não existe garantia que as melhores soluções possam ser encontradas; não se sabe nem se o algoritmo funcionará, e se funcionar, o porquê; conforme Yang (2010). A ideia é ter um algoritmo eficiente e prático que funcionará na maioria das vezes e seja capaz de produzir soluções com boa qualidade. Entre as soluções de qualidade encontradas, pode-se esperar que algumas delas estejam próximas do ótimo, embora não há garantia de tal otimalidade (Gandomi et al., 2013).

Os principais componentes de qualquer algoritmo meta-heurístico são: intensificação e diversificação, ou exploração e exploração, respectivamente. Diversificação significa gerar soluções diversas, assim como explorar o espaço de busca em uma escala global, enquanto intensificação significa focar na busca em uma região local, explorando a informação de que uma boa solução corrente seja encontrada nesta região. Isto está em consonância com a seleção das melhores soluções. A seleção das melhores garante que as soluções convergirão para a otimalidade. Em contrapartida, a diversificação via randomização aumenta a diversidade das soluções, ao mesmo tempo que impede que as soluções sejam capturadas em ótimos locais. Uma boa combinação desses dois componentes principais geralmente garante que a solução global seja alcançável.

De acordo com Yang (2010), os algoritmos meta-heurísticos podem ser classificados de muitas formas. Uma delas é classificá-los como baseados em população ou em trajetória. Por exemplo, GA e programação genética (GP - *Genetic Programming*) são à base de população, pois, eles usam um conjunto de cromossomos / *strings*; otimização com algoritmo de enxame de partículas (PSO), que usa múltiplas partículas ou agentes, é também à base de população. Em contrapartida, o recozimento simulado (SA) usa uma única solução que move através do espaço de projeto ou espaço de busca.

Nas últimas três décadas houve uma explosão de novas metodologias aplicadas ao problema de otimização, mas, há também a evidência da ressurgência de algoritmos clássicos melhorados e um número crescente de problemas de engenharia onde a otimização algorítmica e heurística tem superado e, em alguns casos, substituído especialistas e métodos de otimização clássicos da engenharia (Kaveh, 2017).

Gandomi et al. (2013) avalia que modelagem e otimização podem ter uma ênfase diferente, mas, para resolver problemas do mundo real, tem de se usar ambos os processos, porque a modelagem garante que as funções objetivo sejam avaliadas usando o modelo numérico / matemático correto para o problema de interesse, enquanto a otimização pode obter os valores ótimos dos parâmetros de projeto. Para a otimização, as partes essenciais

são os algoritmos de otimização.

Na engenharia de estruturas a maioria dos problemas de projeto e análise estruturais podem ser tratados como problemas de otimização. Essa abordagem é muito oportuna, uma vez que a estrutura representa um dos custos mais significativos na construção civil, e isso se acentua ainda mais quando se trata de obras onde a parte estrutural é preponderante em importância, como pontes, viadutos, torres de transmissão, edifícios, galpões, etc. Partindo desse princípio, a engenharia de estruturas pode se beneficiar muito com os métodos de otimização e obter projetos que apresentem um uso mais eficiente dos recursos, sem perderem em performance e em segurança. Dentre os vários objetivos que podem ser atingidos a partir do uso de técnicas de otimização no projeto das estruturas, pode-se citar a obtenção de melhores resultados para as variáveis de projeto (que podem ser traduzidos em estruturas com menor peso total, por exemplo) subordinando com meta-heurística os métodos numéricos de análise estrutural já consagrados, como é o caso do MEF (Método dos Elementos Finitos), do MEC (Método dos Elementos de Contorno), entre outros. Um outro objetivo tangível pode ser a otimização do método numérico em si, para que a partir disso, seu uso naturalmente gere melhores resultados e projetos quando utilizados pelos engenheiros calculistas.

Seguindo a vertente de otimizar o método numérico em si, esta pesquisa lidará com a otimização dos parâmetros de discretização do método sem malha local. Buscando melhorar a eficiência computacional do método sem malha, assim como a precisão, duas formulações foram apresentadas por Oliveira e Portela (2016), a formulação de deslocamento de corpo rígido do método sem malha local ou *Rigid-Body Displacement Mesh-Free formulation* (RBDMF), onde a forma local do teorema do trabalho resulta em uma forma fraca apenas com termos de contorno (ausência de forças de corpo); e a formulação do campo elástico generalizado do método sem malha local ou *Generalized-Strain Mesh-Free formulation* (GSMF), onde a forma local do teorema do trabalho resulta em uma forma fraca completamente livre de integração numérica.

A precisão e a eficiência do método sem malha local são determinadas por dois fatores adimensionais, respectivamente o suporte compacto de cada nó e o tamanho do domínio local de cada nó. O primeiro parâmetro está primariamente ligado com a precisão do modelo, devido ao número total de nós utilizados na construção das funções de forma de cada nó da distribuição nodal, enquanto o segundo está intimamente ligado com a eficiência do modelo, onde a forma fraca da equação dos resíduos ponderados é definida computando-se por quadratura numérica ou colocação.

Esses dois parâmetros geralmente são arbitrariamente definidos e podem variar bastante, dependendo do método sem malha local utilizado; sendo esta a razão por sempre serem

apresentados juntamente com novos métodos sem malha desenvolvidos. Moussaoui e Bouziane (2013) realizaram um estudo sobre os efeitos na precisão e na convergência de diferentes parâmetros adimensionais para o MLPG (*Meshless Local Petrov–Galerkin*). A principal desvantagem em definir os parâmetros arbitrariamente é que eles não são únicos, sendo assim, não podem ser implementados em procedimentos automatizados, necessitando sempre da solução analítica para sua obtenção.

Dessa forma, uma otimização utilizando algoritmo genético no MLPG foi apresentada por Baradaran e Mahmoodabadi (2009), para uma problema bidimensional de condução de calor, e por Bagheri et al. (2011), para um problema elasto-plástico tridimensional. Uma otimização similar foi proposta por Ebrahimnejad et al. (2015), adicionando uma técnica de refinamento adaptativo ao processo de otimização, usando o MLPG-FVM. Apesar desses autores terem sido bem sucedidos em suas propostas, suas tentativas levaram a um método com alto custo computacional e dependente da solução analítica do problema em questão.

Uma estratégia inovadora onde os parâmetros de discretização do método sem malha local, α_s e α_q , são encontrados de forma totalmente automatizada realizando-se uma otimização multiobjetivo, usando algoritmo genético, foi apresentada por Santana et al. (2020), com o método ILMF (*Integrated Local Meshfree*). Nessa abordagem a maior vantagem é o uso da conformidade estrutural C na definição de uma das funções objetivo, o que dispensa o conhecimento da solução analítica do problema; que é um grande diferencial dessa estratégia. Neste caso, o ILMF foi aplicado na solução de problema da elasticidade linear.

No âmbito da aplicação do ILMF na solução de problemas da mecânica da fratura linear elástica, Oliveira e Portela (2019) apresentaram uma abordagem muito eficaz e precisa, onde α_s e α_q também foram definidos de forma automática por meio de otimização multiobjetivo com algoritmos genéticos. Foi proposta a técnica da subtração da singularidade, ou *Singularity Subtraction Technique* (SST), que regulariza o campo elástico antes da solução numérica, assim introduzindo o fator de intensidade de tensão, ou *Stress Intensity Factor* (SIF), como variáveis primárias do problema. Sendo assim, o modelo numérico realiza o cálculo do SIF diretamente, sem a necessidade de uma discretização nodal refinada para se obter resultados precisos, o que faz desta uma eficiente estratégia de modelagem.

Esta pesquisa propõe uma nova abordagem de otimização do ILMF, para a obtenção dos parâmetros α_s e α_q , utilizando dez algoritmos meta-heurísticos multiobjetivo (denominados aqui MMAs, por praticidade): NSGA-II, NSGA-III, CMAES, MOEA/D, IBEA, Epsilon-MOEA, SPEA2, GDE3, OMOPSO e SMPSO, largamente utilizados no meio científico. Aqui, a adequação de cada MMA na otimização dos modelos ILMF será avaliada a partir das métricas de performance: hipervolume, distância geracional, distância

geracional invertida, indicador ϵ_+ e espaçamento; normalmente utilizadas na avaliação de algoritmos multiobjetivo. As formulações do ILMF para a elasticidade linear serão as apresentadas em Oliveira e Portela (2016) e Santana et al. (2020); para a mecânica da fratura, serão as constantes em Oliveira e Portela (2019), devido suas eficácias comprovadas. O *compliance* será utilizado na definição das funções objetivo para as duas áreas de aplicação, onde será mostrada sua correlação perfeita positiva com o erro (solução numérica versus analítica) no campo de deslocamentos, para a elasticidade linear. E que, o mesmo ocorre para a mecânica da fratura, só que ao invés do erro, a correlação é entre o *compliance* e os fatores de intensidade de tensão (SIFs). Serão implementadas rotinas para a discretização automática do domínio, permitindo variar a densidade nodal e o tipo da discretização, se regular ou randômica; com isso, a influência da discretização na precisão dos resultados do ILMF e na performance dos MMAs será avaliada.

É imprescindível a avaliação do desempenho dos MMAs a partir das métricas de performance, pois, assim, além de obter um *ranking* mostrando quais algoritmos potencialmente gerarão os melhores resultados para cada tipo de modelo; propicia a comparação de desempenho com resultados de outras pesquisas, já que são indicadores amplamente empregados no meio científico.

Nesse texto, as siglas GSMF e ILMF são usadas intercambiavelmente para descrever o método sem malha local. Também, o ponto é usado como separador decimal em números decimais, ao invés de vírgula como é usado no Brasil. Isso é adotado por questão de praticidade, uma vez que a maioria dos *softwares* utilizados nessa pesquisa usam o padrão americano (ou seja, ponto como separador decimal e vírgula como separador de milhar).

1.1 - OBJETIVOS

1.1.1 - Objetivos Gerais

Apresentar um nova abordagem para a otimização dos parâmetros adimensionais (α_s e α_q) do ILMF aplicado em modelos da elasticidade linear e da mecânica da fratura linear elástica, utilizando algoritmos meta-heurísticos multiobjetivo (designados nesse texto como MMAs, por praticidade), já consagrados na comunidade científica; e, ranqueá-los em função da performance na otimização.

Contribuir na sedimentação e estabelecimento das metodologias para a otimização do ILMF, que vem sendo desenvolvidas pelo grupo de pesquisa do professor Portela, no âmbito do programa de Pós-graduação em Estruturas e Construção Civil (PECC), nos últimos sete anos; deixando-as ainda mais robustas e gerais. Consequentemente, contribuir

na pavimentação do caminho para um futuro próximo onde o método sem malha local será utilizado alternativamente a métodos numéricos consagrados, como o método dos elementos finitos (FEM); inclusive, sendo mais eficiente e preciso.

Estimular o uso de otimização na engenharia de estruturas, o que comprovadamente gera uma grande economia de recursos e um ganho significativo no desempenho estrutural. A otimização pode ser empregada para aperfeiçoar os métodos numéricos que normalmente são utilizados na solução e na análise estrutural (FEM, BEM, ILMF, etc.), como é o caso dessa pesquisa; ou, para otimizar o projeto de estruturas diretamente, tendo como objetivo, por exemplo: a minimização do peso total (seção dos elementos estruturais); o aprimoramento na distribuição de parafusos e soldas nas ligações metálicas; o posicionamento e bitolas ótimos dos cabos em estruturas protendidas; o balanceamento de massas e ajustes nos modos de vibração de estruturas sujeitas a carregamentos dinâmicos; entre uma infinidade de outras aplicações. Estes temas são sugestões de pesquisa e aprofundamento, e não foi feita busca por artigos existentes sobre os mesmos.

1.1.2 - Objetivos Específicos

Como objetivos específicos deste trabalho, pretende-se:

- Implementar uma classe (via OOP - Programação Orientada a Objetos) dedicada à discretização automática, regular ou randômica, do domínio do modelo a ser otimizado e, acoplá-la ao código de otimização;
- Utilizar os *solvers* do ILMF já implementados, em MATLAB, tanto para a elasticidade linear como para a mecânica da fratura linear elástica (conforme Santana et al. (2020) e Oliveira e Portela (2019), respectivamente), acoplados com o código desenvolvido em Python e o *package* Platypus de otimização meta-heurística multiobjetivo (que dispõe dos algoritmos: NSGA-II, NSGA-III, CMAES, MOEA/D, IBEA, Epsilon-MOEA, SPEA2, GDE3, OMOPSO e SMPSO); para otimizar os parâmetros adimensionais (α_s e α_q) do ILMF. Os *solvers* serão preparados para cumprirem o papel de funções objetivo no processo de otimização de cada modelo;
- Gerar uma frente de Pareto de referência e a frente de Pareto da otimização propriamente dita, então, avaliar o desempenho de cada MMA em relação à frente de Pareto de referência, ranqueando-os a partir da aplicação de uma *utility function* nas métricas de performance: hipervolume, distância geracional, distância geracional invertida, indicador ϵ_+ e espaçamento. Repetir essa abordagem para cada modelo estudado/otimizado;
- Demonstrar estatisticamente e graficamente, por meio do coeficiente de correlação de

Pearson, a eficiência do uso do *compliance* estrutural C na construção de função objeto para a otimização do ILMF aplicado na modelagem de problemas da elasticidade linear e da mecânica da fratura linear elástica;

- Avaliar a influência da densidade nodal e do tipo (regular ou randômica) da discretização nodal do domínio, na precisão dos resultados e na performance dos MMAs;
- Demonstrar a grande utilidade do emprego de ferramentas de *data science* na exploração, análise e apresentação dos resultados obtidos com a otimização.

1.2 - ORGANIZAÇÃO DO TRABALHO

Esse texto está estruturado da seguinte forma: após essa introdução, é apresentada no Capítulo 2, uma revisão bibliográfica sobre otimização de forma ampla, as condições de otimalidade, as métricas de performance dos algoritmos multiobjetivo, os algoritmos meta-heurísticos multiobjetivo, os *packages, frameworks e toolboxes* de otimização disponíveis, entre outros assuntos. Na sequência, Capítulo 3, são mostrados os fundamentos teóricos dos métodos sem malha, que fornecem a formulação base para a solução dos modelos abordados nessa pesquisa. Por sua vez, o Capítulo 4, traz as formulações relativas à mecânica da fratura linear elástica com a técnica da subtração da singularidade. Já o Capítulo 5, apresenta os critérios utilizados para a definição das funções objetivo imprescindíveis na otimização dos modelos aqui estudados. O Capítulo 6 relata a metodologia seguida para o atingimento dos objetivos estabelecidos. No Capítulo 7, são apresentados os resultados obtidos ao longo do desenvolvimento da pesquisa, as apreensões e comparações com resultados da literatura correlata. Fechando o texto, o Capítulo 8 apresenta as conclusões e as recomendações para futuros desenvolvimentos. Por fim, o "Apêndice A" explana sobre os parâmetros de discretização α_s e α_q do ILMF. O "Apêndice B" traz uma explicação sucinta sobre os tipos de gráficos incomuns utilizados nesse estudo e, o "Apêndice C" apresenta as publicações feitas ao longo do desenvolvimento dessa pesquisa.

2 - REVISÃO BIBLIOGRÁFICA

2.1 - OTIMIZAÇÃO

2.1.1 - Histórico

Há evidências de que as técnicas de otimização remontam a 300 a.C., desde o início da matemática. Gandomi et al. (2013) ilustra de forma prática e didática um histórico sucinto dessas técnicas e os principais pesquisadores envolvidos (Figura 2.1), até os recentes avanços nessa área. Pode-se observar que o marco matemático formal da otimização se deu com o advento do cálculo variacional de 1660 a 1670, proposto por Newton e Leibniz. Que posteriormente contou com as contribuições de Bernoulli, Taylor, Euler, Lagrange, Legendre, Hamilton, Cauchy, entre outros.

O cálculo de variações é um problema matemático que consiste em buscar máximos e mínimos (ou, mais geralmente, extremos relativos) de funções contínuas definidas sobre algum espaço funcional. Constituem uma generalização do cálculo elementar de máximos e mínimos de funções reais de uma variável. Ao contrário deste, o cálculo das variações lida com os funcionais, enquanto o cálculo ordinário trata de funções. Funcionais podem, por exemplo, ser formados por integrais envolvendo uma função incógnita e suas derivadas. O interesse está em funções extremas - aquelas que fazem o funcional atingir um valor máximo ou mínimo - ou de funções fixas - aquelas onde a taxa de variação do funcional é precisamente nula. Talvez o exemplo mais simples seja o de encontrar a curva com o menor comprimento possível ligando dois pontos. Se não houver restrições, a solução é (obviamente) uma linha reta ligando estes pontos. No entanto, se as possibilidades para esta curva estiverem restritas a uma determinada superfície no espaço, então a solução é menos óbvia e, possivelmente, muitas soluções podem existir. Tais soluções são conhecidas como geodésicas.

A despeito das contribuições fundamentais desses grandes cientistas/matemáticos com o cálculo variacional (e indiretamente à otimização como é vista hoje), pouco progresso foi feito até meados do século 20, quando os computadores de alta performance surgiram e tornaram possível a implementação de técnicas de otimização para o projeto de grandes estruturas. Então, técnicas modernas de otimização, principalmente as relacionadas à meta-heurística, emergiram como métodos poderosos e populares para a solução de problemas de otimização complexos da engenharia, principalmente no passado recente.

Apesar de otimização ser um ramo da matemática, a história mostra o notável papel de engenheiros no desenvolvimento de técnicas de otimização, para a aplicação prática em seus projetos.

Historicamente a otimização estrutural esteve muito associada à otimização de forma e de topologia, mas, é importante destacar que os métodos de otimização são capazes de buscar valores para qualquer variável do projeto na obtenção da solução ótima, desde que a modelagem seja feita adequadamente. Isto é, pode-se por exemplo, encontrar a carga crítica e sua distribuição na estrutura para que os deslocamentos nela obedeçam restrições pré-estabelecidas. Deixando assim o analista livre para modelar o problema estrutural da forma mais conveniente para a consecução dos seus objetivos.

2.1.2 - Problemas de Otimização

Otimização é um ramo da matemática relacionado à obtenção de condições que deem o extremo de uma função, ou várias funções, sob dadas circunstâncias. A otimização desempenha um papel fundamental na busca de soluções viáveis para os problemas da vida real, desde a programação matemática à pesquisa operacional, economia, ciência gerencial, negócios, medicina, ciências da vida e inteligência artificial, para mencionar apenas alguns. Um problema de otimização pode ser definido como: encontrar os valores das variáveis de uma função objetivo que a torne um mínimo (ou máximo), obedecendo algumas restrições; se o problema for de otimização restrita. Essas restrições representam limitações no comportamento ou performance do sistema. Portanto, elas são chamadas restrições comportamentais ou funcionais. Restrições laterais podem ocorrer, e estas restringem o intervalo aceitável de soluções potenciais do problema, que foram obtidas considerando as restrições não comportamentais.

Dependendo da escolha específica das variáveis de projeto, funções objetivo e restrições, vários tipos de problemas de otimização podem ser criados. Com base nisso, Gandomi et al. (2013) propõem uma classificação dos problemas de otimização, conforme apresentada na Tabela 2.1.

2.1.3 - Métodos de Otimização

As técnicas de otimização podem ser classificadas de diversas formas. De uma forma bem geral, elas podem ser divididas em duas grandes categorias: otimização de função ou otimização de parâmetros. No primeiro caso, um objeto sob consideração é descrito por um número de funções desconhecidas e, através de um processo de otimização, a forma ótima

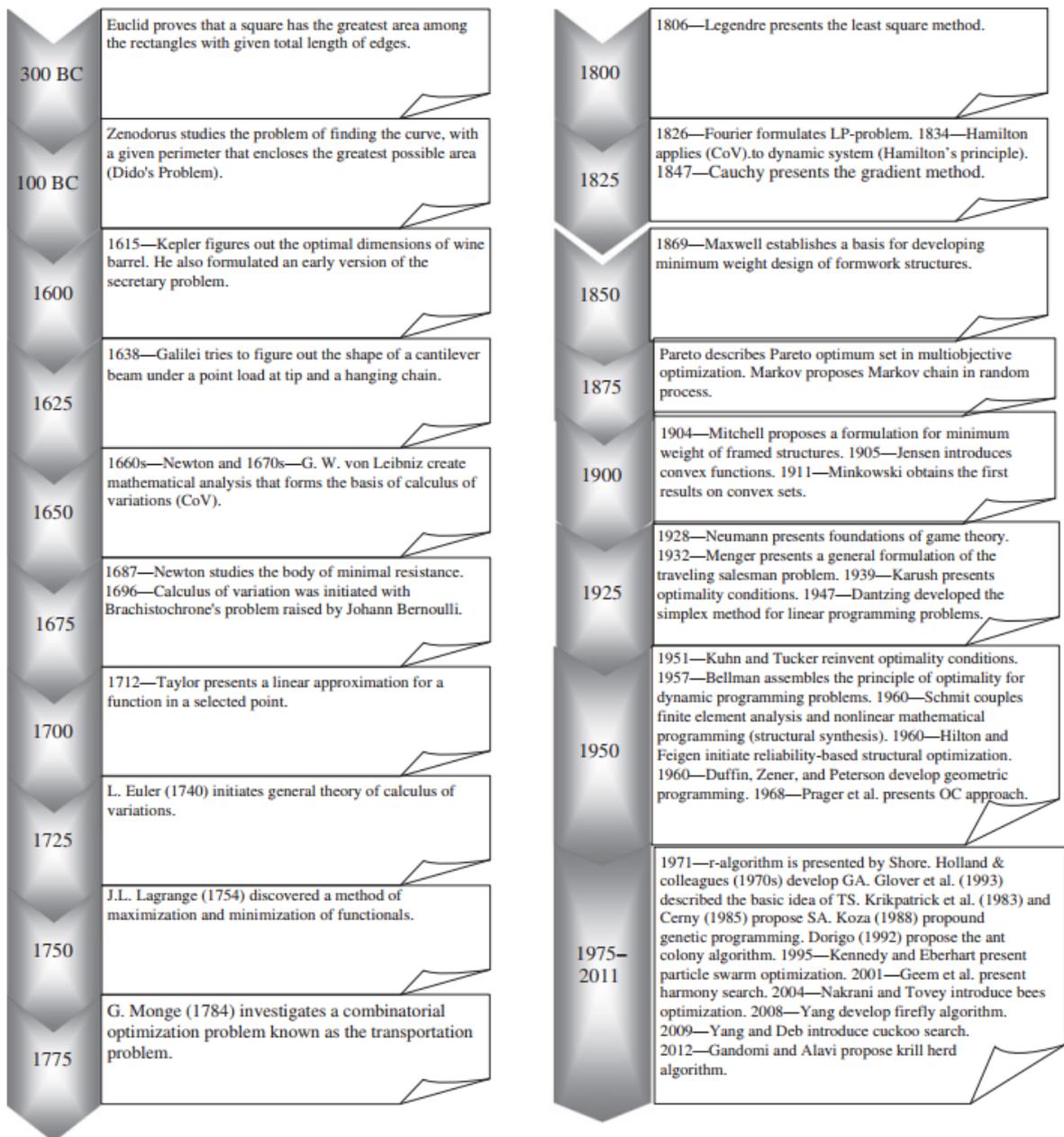


Figura 2.1 – Cronologia sucinta das técnicas de otimização, segundo Gandomi et al. (2013).

Tabela 2.1 – Classificação dos problemas de otimização.

Base de Classificação	Categoria	Especificações
Número de variáveis de projeto	Uma variável	O vetor de variáveis de projeto tem uma única variável.
	Multivariáveis	O vetor de variáveis de projeto tem mais de uma variável.
Número de funções objetivo	Mono-objetivo	Há um critério expresso como uma função objetivo.
	Multiobjetivos	Há muitos critérios que são considerados em conjunto para determinar uma solução ótima.
Presença de restrições	Irrestritos	É tentado um mínimo ou máximo de uma função objetivo sem qualquer limitação.
	Restritos	Algumas restrições definem o conjunto de soluções viáveis.
Características das restrições e das funções objetivo	Programação linear	As funções objetivo e as restrições são lineares.
	Programação não linear	Algumas das funções objetivo e restrições podem ser não lineares. Problemas de programação quadrática e geométrica são dois tipos específicos de problemas de otimização não linear.
Natureza das variáveis de projeto	Estáticas	As variáveis de projeto são independentes. Elas não são função de outros parâmetros.
	Dinâmicas	As variáveis de projeto são função de outros parâmetros, por exemplo, o tempo.
Tipo das variáveis de projeto	Discretas	As variáveis de projeto podem receber somente valores inteiros ou discretos.
	Contínuas ou Mistras	As variáveis de projeto podem receber qualquer valor real. Algumas podem receber valores inteiros e outras real.
Natureza das variáveis e dados de entrada do projeto	Determinísticos	Todas as variáveis de projeto ou parâmetros pré-atribuídos, tais como cargas agindo na estrutura, são assumidos ser determinísticos.
	Probabilísticos	Todas ou algumas das variáveis de projeto ou parâmetros pré-atribuídos são descritos como variáveis randômicas ou probabilísticas dentro de um dado intervalo.
Natureza das funções objetivo e restrições de projeto	<i>Crisp</i>	As restrições e as funções objetivo são descritas por expressões ou respostas <i>nonfuzzy</i> e inequívocas.
	<i>Fuzzy</i>	Algumas das restrições e das funções objetivo são descritas por expressões ou respostas <i>fuzzy</i> .

dessas funções será encontrada. A técnica para otimização de função usa o cálculo diferencial (Cálculo Variacional, etc.) para encontrar a função ótima. Enquanto que na otimização de parâmetros, ao invés de buscar uma função contínua ótima, os valores ótimos das variáveis de projeto são obtidos para um problema específico. Programação matemática, critérios de otimalidade (OC - *Optimality Criteria*) e métodos meta-heurísticos são alguns subconjuntos das técnicas de otimização de parâmetros, como reportado por Gandomi et al. (2013).

Já Yang (2010) prefere classificar os métodos de otimização em dois tipos: determinísticos e estocásticos, veja o diagrama esquemático na Figura 2.2. Os algoritmos determinísticos seguem um procedimento rigoroso e seu caminho e valores das variáveis de projeto e das funções, têm repetibilidade. Por exemplo, *hill climbing* é um algoritmo determinístico, e para o mesmo ponto de partida, ele seguirá o mesmo caminho independentemente se for executado hoje ou amanhã. Em contrapartida, os algoritmos estocásticos sempre possuem alguma aleatoriedade. Algoritmos genéticos são um bom exemplo, as *strings* ou soluções na população serão sempre diferentes a cada vez que são executados, pois, usam números pseudo-aleatórios; embora os resultados finais não serão muito diferentes, mas, os caminhos de cada indivíduo provavelmente não apresentarão repetibilidade. Também, existe um terceiro tipo de método que é um híbrido de algoritmos determinísticos e estocásticos. *Hill climbing* com reinício aleatório é um bom exemplo. A ideia básica é usar um algoritmo determinístico, mas, começar com diferentes pontos de partida. Isso, apresenta certas vantagens em relação à técnica *hill climbing* isolada, que pode ser capturada em um pico (ou depressão) local. Entretanto, uma vez que existe um componente randômico nesse algoritmo híbrido, faz com que ele seja normalmente classificado na literatura de otimização como um tipo de algoritmo estocástico.

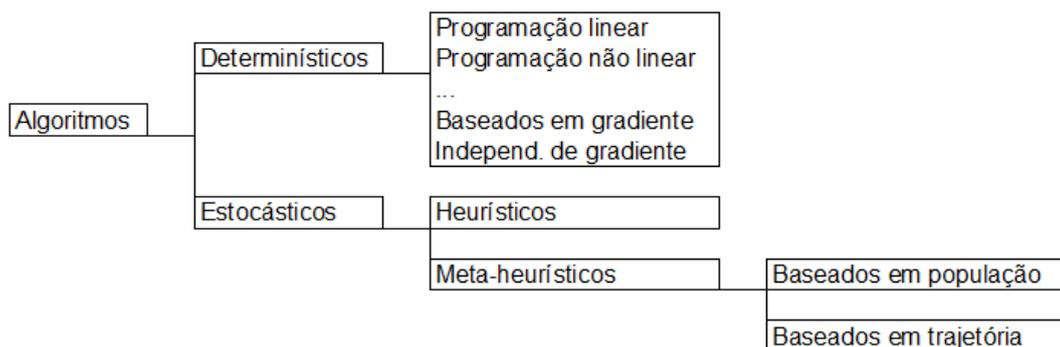


Figura 2.2 – Classificação das técnicas de otimização, segundo Yang (2010).

2.1.4 - Meta-heurísticas - Breve Histórico

Muitos processos para a solução de problemas tenderam ser heurísticos ao longo da história da humanidade; entretanto, heurística como um método científico de otimização é um fenômeno moderno (Yang, 2011). De 1940 a 1960, métodos heurísticos foram usados em várias aplicações, mas, o primeiro marco veio com o advento dos algoritmos evolucionários. Em 1963, na Universidade Técnica de Berlim, Ingo Rechenberg and Hans-Paul Schwefel desenvolveram o método "Estratégias Evolucionárias", enquanto L.J. Fogel et al. desenvolveram "Programação Evolucionária", em 1966. Algoritmos Genéticos foram desenvolvidos por J.H. Holland nas décadas de 1960 e 1970, embora seu livro seminal tenha sido publicado em 1975 (Holland, 1975).

As décadas de 1980 e 1990 se destacaram quanto à atenção despertada pelos algoritmos meta-heurísticos. Um grande passo foi o desenvolvimento do "Recozimento Simulado", em 1983, uma técnica de otimização iniciada por S. Kirkpatrick et al., e inspirada no processo de recozimento de metais. Outro passo importante foi o desenvolvimento dos "Sistemas Imunológicos Artificiais", por Farmer et al., em 1986.

O uso de memória em meta-heurísticas foi iniciado por Glover em "Busca Tabu", na década de 1980; embora seu livro correlato tenha sido publicado posteriormente em 1997 (Glover e Laguna, 1998). Movimentos de busca são gravados em uma lista Tabu, e os futuros movimentos evitam visitar as soluções prévias.

Em 1992, Marco Dorigo concluiu sua tese em otimização e algoritmos naturais, na qual descreve seu trabalho inovador sobre otimização com "Colônia de Formigas"(ACO), cujo livro foi publicado um pouco depois (Dorigo et al., 2004). Esta técnica de busca foi inspirada pela inteligência de enxame de formigas sociais usando feromônio como um mensageiro químico. Então, também em 1992, John R. Koza da Universidade de *Stanford* publicou um livro sobre programação genética, que lançou as bases de uma área totalmente nova de aprendizado de máquina, revolucionando a programação de computadores. Pouco depois, em 1995, outro progresso significativo foi o desenvolvimento da otimização com enxame de partículas, por James Kennedy e Russell C. Eberhart. Por volta de 1996, e depois em 1997, R. Storn e K. Price desenvolveram o algoritmo evolucionário à base de vetor, chamado evolução diferencial (DE). Este algoritmo provou ser mais eficiente do que algoritmos genéticos em várias aplicações.

Na virada do século XXI, as coisas se tornaram ainda mais atrativas. Primeiro, Zong Woo Geem et al. desenvolveram a "Busca Harmônica"(HS), em 2001, que é um algoritmo inspirado em música. Por volta de 2002, o algoritmo "FORAGEAMENTO BACTERIANO" foi desenvolvido por Passino. Em 2004, S. Nakrani e C. Tovey propuseram o algoritmo *Honey*

Bee e sua aplicação para otimização de centrais de hospedagem na internet, que foi seguido pelo desenvolvimento do novo algoritmo *Bee*, por D.T. Pham et al., em 2005 e, do algoritmo "Colônia Artificial de Abelhas"(ABC), por D. Karaboga, também em 2005. Em 2008, Xin-She Yang desenvolveu o algoritmo "Vaga-lume"(FA). Em 2009, Xin-She Yang e Suash Deb introduziram o algoritmo "Busca Cuckoo"(CS).

2.1.5 - Descrição Sucinta dos Principais Métodos Meta-heurísticos

Os métodos meta-heurísticos são principalmente intuitivos e não têm base teórica, de acordo com Gandomi et al. (2013). Dentre eles, GA, SA e TS fornecem maneiras gerais de buscar uma boa, mas não necessariamente a melhor, solução. GA e SA usam regras estatísticas em seu processo de busca e podem ser classificados como métodos estocásticos de busca. Nas últimas três décadas os métodos meta-heurísticos se desenvolveram rapidamente para a solução de problemas de otimização. A literatura afim reporta que esses métodos têm sido empregados em problemas de otimização estrutural e produzido boas soluções em um tempo de processamento razoável. Dentre eles, GA tem uma vantagem adicional, que é o fato de não exigir um "chute" inicial no processo de busca pelo ótimo. As subseções seguintes apresentam uma abordagem sucinta e informativa sobre os principais métodos meta-heurísticos utilizados em problemas de otimização.

2.1.5.1 - Algoritmo genético

Algoritmo genético (GA) é uma técnica de otimização inspirada na evolução biológica baseada na teoria da seleção natural de Charles Darwin. GA foi desenvolvido por John Holland e seus colaboradores nos anos de 1960 a 1970. Ele foi o primeiro a usar *crossover* e recombinação, mutação e seleção no estudo de sistemas adaptativos e artificiais. Esses operadores genéticos formam a parte essencial do algoritmo genético como uma estratégia de solução de problema. Desde sua criação, muitas variantes de GA foram desenvolvidas e aplicadas a um grande número de problemas de otimização, de pintura gráfica a reconhecimento de padrões, de sistemas discretos, como o problema do caixeiro-viajante, a sistemas contínuos, como o projeto eficiente de aerofólios em engenharia aeronáutica, e de mercado financeiro a otimização multiobjetivo em engenharia, veja Yang (2010).

Os algoritmos genéticos apresentam muitas vantagens em relação aos algoritmos tradicionais de otimização, e duas mais notáveis são: a habilidade de lidar com problemas de otimização complexos e paralelismo. GA pode resolver vários tipos de otimização, independente se a função objetivo é estacionária ou não (muda com o tempo), se é linear ou não linear, contínua ou descontínua, ou com ruídos aleatórios. Como múltiplas proles na população atuam

como agentes independentes, a população (ou qualquer subgrupo) pode explorar o espaço de busca em muitas direções simultaneamente. Esta característica tornam eles ideais para implementação de algoritmos com paralelização. Diferentes parâmetros, e mesmo grupos diferentes de *strings* codificadas, podem ser manipuladas ao mesmo tempo.

Todavia, GA também tem algumas desvantagens. A formulação da função *fitness*, o uso do parâmetro de tamanho da população, a escolha de parâmetros importantes como as taxas de mutação e *crossover* e os critérios seleção de nova população devem ser feitos cuidadosamente; pois, qualquer escolha inapropriada tornará difícil a convergência do algoritmo, ou simplesmente produzirá resultados inócuos. Apesar dessas desvantagens, GA continua sendo um dos algoritmos mais amplamente utilizados de otimização na moderna otimização não linear.

A essência dos algoritmos genéticos envolve a codificação de uma função de otimização na forma de vetor de *bits* ou *strings* de caracteres para representar os cromossomos, para executar as operações genéticas nas *strings* e realizar seleção de acordo com sua aptidão / *fitness*, com o objetivo de encontrar uma solução para o problema de interesse. Normalmente, os passos que são executados em algoritmo genético, são apresentados esquematicamente no fluxograma da Figura (2.3).

Uma iteração de criação de uma nova população é chamada geração. *Strings* de caracteres de comprimento fixo são usadas na maioria dos algoritmos genéticos durante cada geração, embora existam pesquisas significativas sobre *strings* de comprimento variável e estruturas de codificação. Usualmente em algoritmos genéticos adaptativos, a codificação da função objetivo é feita na forma vetores de dados binários ou reais.

Os operadores genéticos incluem *crossover*, mutação e seleção da população. O *crossover* de duas *strings* é o principal operador com uma alta probabilidade p_c e, é realizado através da troca de um segmento de um cromossomo com o segmento correspondente em outro cromossomo em uma posição aleatória. Realizado desta forma é chamado de *crossover* de ponto único. *Crossover* de múltiplos pontos é muito usado também para aumentar a eficiência do GA.

A operação de mutação é obtida ao se alterar bits selecionados aleatoriamente, e a probabilidade de mutação p_m é geralmente pequena. A seleção de um indivíduo na população é realizada pela avaliação da sua aptidão (*fitness*), e pode permanecer na nova geração se um certo limiar da aptidão for alcançado; também, pode-se usar *crossover* proporcional à aptidão. Ou seja, os indivíduos com maior aptidão são mais propensos a se reproduzirem. Essas operações genéticas são realizadas iterativamente até que uma condição de parada / convergência pré-estabelecida seja alcançada e o algoritmo finalizado.

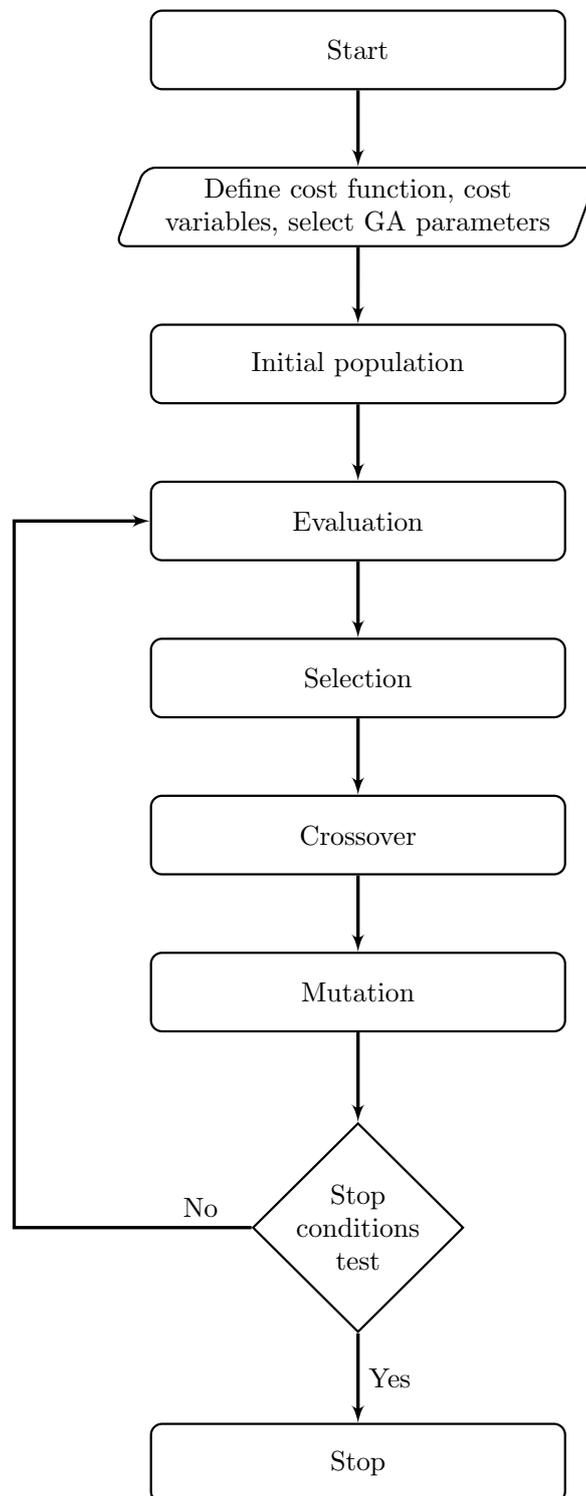


Figura 2.3 – Fluxograma geral dos passos executados em algoritmo genético.

2.1.5.2 - Recozimento simulado

O algoritmo de recozimento simulado (SA) é baseado na analogia entre a forma que a estrutura cristalina de um metal atinge estados de energia próximos do mínimo global, durante o processo de recozimento, e o modo no qual uma função pode atingir o mínimo durante uma busca estatística no espaço de projeto. A função objetivo corresponde ao estado de energia e o movimento para qualquer novo conjunto de variáveis de projeto equivale a uma variação do estado de energia. Embora o método tenha sido desenvolvido basicamente para problemas discretos, pode ser usado em problemas contínuos, como acontece com GA.

2.1.5.3 - Busca tabu

A busca tabu (TS) é um procedimento iterativo apropriado para problemas discretos de otimização. Todavia, este método, similarmente a SA e GA, pode ser aplicado em problemas contínuos. O método consiste de uma sequência de movimentos de exploração sobre o espaço de busca do projeto, que é iniciada a partir de um ponto / projeto viável até alcançar um ótimo. Para mover de uma solução para outra, o algoritmo explora a vizinhança do ponto e escolhe a melhor solução disponível. Para evitar a repetição de etapas realizadas anteriormente, o método registra como proibidos (lista tabu) os movimentos recentes de uma iteração particular. Movimentos Tabu são baseados no histórico de curto e longo prazo da sequência de movimentos. As aplicações de TS na otimização estrutural é bem recente. Já foi aplicada no projeto de treliça, otimização de problemas de topologia estrutural e otimização de pórticos (Gandomi et al., 2013).

2.1.5.4 - Otimização com colônia de formigas

Otimização com colônia de formigas (ACO) foi desenvolvida por Dorigo e seus associados, no início dos anos 90. A ideia principal de otimização com ACO é imitar o comportamento cooperativo de uma colônia de formigas, que encontra o caminho mais curto até à fonte de comida. Neste método, um problema de otimização combinatorial com n variáveis de projeto, $(x_1 - x_n)$ são modeladas como grafos multicamadas, conforme mostrado na Figura 2.4.

O número de camadas é igual ao número de variáveis de projeto e o número de nós em cada camada é igual ao número de valores discretos permitidos para a correspondente variável de projeto. Assim, cada nó em uma posição específica do grafo é associado com um valor discreto admissível de uma variável de projeto. Formigas artificiais caminham através desse grafo, procurando por caminhos até à comida. Uma colônia de formigas tem N formigas.

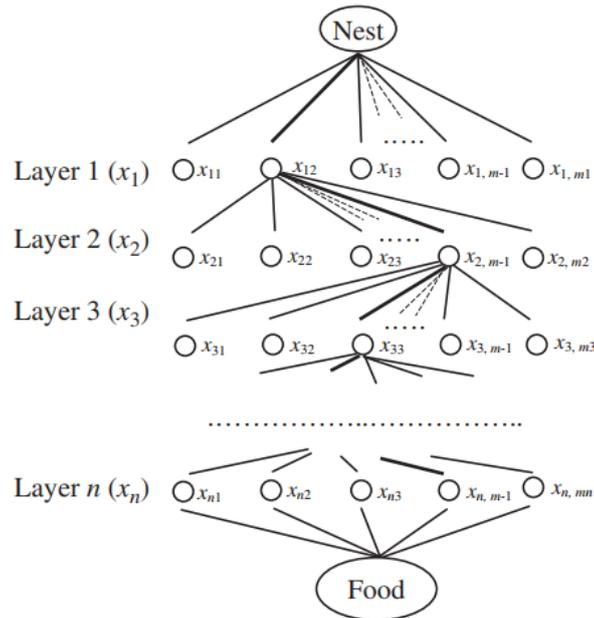


Figura 2.4 – Processo de ACO na forma de grafo multicamadas (Gandomi et al., 2013).

As formigas partem do nó correspondente ao ninho, viaja através de várias camadas, da primeira à última, e termina no nó de destino em cada ciclo ou iteração. Cada formiga pode selecionar somente um nó em cada camada, de acordo com a regra de transição de estado dada pela informação meta-heurística. Os nós selecionados ao longo do caminho visitados por uma formiga representam uma solução candidata. Um caminho típico visitado por uma formiga é mostrado como uma linha espessa na Figura 2.4. Uma vez que o caminho esteja completo, a formiga deposita feromônio nele com base em uma regra de atualização local. No início do processo de otimização (ou seja, na iteração 1), todas as arestas ou raios são inicializados com uma quantidade igual de feromônio. Assim, na iteração 1, todas as formigas partem do nó ninho inicial e terminam no nó fonte de comida, selecionando aleatoriamente um nó em cada camada. Pequenas quantidades de feromônio são depositadas durante a fase de construção, enquanto maiores quantidades são depositadas no final de cada iteração proporcionalmente à qualidade da solução. O processo de otimização é finalizado se qualquer uma das condições de parada definidas for satisfeita. Os valores das variáveis de projeto indicadas pelos nós no caminho com a maior quantidade de feromônio são considerados como os componentes do vetor da solução ótima. Em geral, na solução ótima, todas as formigas viajam pelo mesmo caminho ótimo / convergido (Gandomi et al., 2013).

ACO tem sido utilizada na otimização do projeto de treliças espaciais, no projeto ótimo de estruturas esqueléticas, na otimização de projeto de pórticos, no projeto ótimo da forma de torres de transmissão, na identificação de falhas em treliças e inúmeras outras aplicações.

2.1.5.5 - Otimização com enxame de partículas

A otimização com enxame de partículas (PSO) foi desenvolvida por Kennedy e Eberhart em 1995, com base no comportamento de cardumes de peixes e bandos de pássaros na natureza. Muitos algoritmos como colônia de formigas e algoritmos de formigas virtuais, usam o comportamento da chamada inteligência de enxame. Embora a otimização com enxame de partículas tenha muitas semelhanças com algoritmos genéticos e algoritmos de formigas virtuais, é muito mais simples porque não usa operadores de mutação / *crossover* ou feromônio. Em vez disso, ele usa a aleatoriedade com número real e a comunicação global entre as partículas do enxame. Nesse sentido, também é mais fácil de implementar, pois não há codificação ou decodificação dos parâmetros em *strings* binárias ou de números reais, como em algoritmos genéticos (Yang, 2010).

Esse algoritmo faz busca no espaço de uma função objetivo, ajustando as trajetórias dos agentes individuais, chamados partículas, uma vez que estas trajetórias formam caminhos segmentados de uma maneira semi-estocástica. O movimento das partículas tem dois componentes principais: um componente estocástico e um componente determinístico. Cada partícula é atraída em direção ao ótimo global corrente g^* e à sua própria melhor posição x_i^* histórica e, ao mesmo tempo, tem uma tendência a se mover aleatoriamente.

Quando uma partícula encontra um local melhor que qualquer outro encontrado anteriormente, ela se atualiza com a nova posição ótima corrente (para a partícula i). Existe um ótimo corrente para todas as partículas n em qualquer tempo t durante as iterações. O objetivo é encontrar o ótimo global entre todas as ótimas soluções individuais correntes, até que o objetivo não melhore mais ou após um certo número de iterações. O movimento das partículas é representado esquematicamente na Figura 2.5, onde x_i^* é o ótimo corrente para a partícula i e $g^* \approx \min \{f(x_i)\}$ para $(i = 1, 2, \dots, n)$ é o ótimo global corrente.

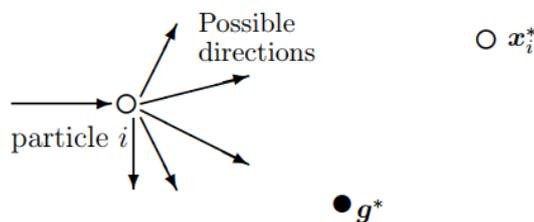


Figura 2.5 – Representação esquemática do movimento de uma partícula em PSO, movendo-se em direção ao ótimo global g^* e à melhor posição corrente x_i^* de cada partícula i (Yang, 2010).

2.1.5.6 - Algoritmo de vaga-lumes

O algoritmo de otimização de vaga-lumes (FA) é baseado em população e imita a atração de um vaga-lume pela luz intermitente. Foi proposto por Xin-She Yang, em 2008. É naturalmente um algoritmo multimodal, portanto, pode ser usado apropriadamente em problemas de otimização da engenharia, especialmente quando são necessárias algumas alternativas de solução em problemas multimodais (Gandomi et al., 2013).

2.1.5.7 - Busca cuco

A busca cuco (CS) é um algoritmo meta-heurístico inspirado em algumas espécies de uma família de aves chamada cuco, devido ao seu estilo de vida especial e estratégia de reprodução agressiva. Essas espécies depositam seus ovos com habilidades incríveis nos ninhos de outras aves hospedeiras: como selecionar os ninhos, recentes e com ovos recém postos, e remover os ovos existentes; o que aumenta a probabilidade de nascimento de seus ovos. A ave hospedeira cuida dos ovos, presumindo que os ovos são seus. No entanto, algumas das aves hospedeiras são capazes de combater este comportamento parasita dos cucos e lançam fora os ovos estranhos descobertos ou constroem seus novos ninhos em novos locais. A analogia com a reprodução do cuco foi usada para desenvolver um novo algoritmo de otimização de projeto. Uma geração é representada por um conjunto de ninhos hospedeiros. Cada ninho carrega um ovo (solução). A qualidade das soluções é melhorada gerando uma nova solução a partir de uma solução existente e modificando certas características. O número de soluções permanece fixo em cada geração (Kaveh, 2017). Gandomi et al. (2013) reporta que CS tem sido aplicado com sucesso em otimização de pórticos e diferentes tipos de treliças, e até supera a maioria dos algoritmos meta-heurísticos mais conhecidos.

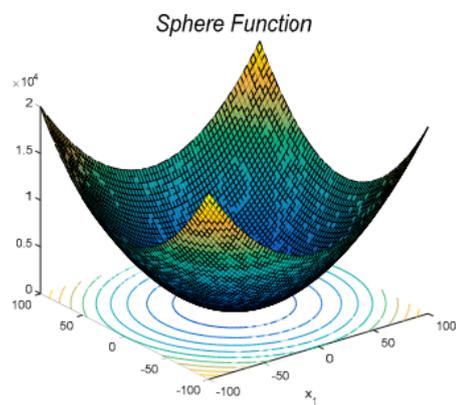
2.1.6 - Comparativo dos Métodos de Otimização Estocásticos e Determinísticos

Conforme já visto até aqui, os métodos meta-heurísticos (estocásticos) apresentam uma série de vantagens em relação aos clássicos (determinísticos). Ratificando isso, a Tabela 2.2 apresenta de forma sintética um comparativo entre esses métodos.

Os métodos determinísticos são bastante limitados quando se trata da otimização de funções multimodais, não convexas, não suaves, entre outras. Eles normalmente ficam presos em mínimos (ou máximos) locais quando aplicados em funções multimodais, como é o caso das funções *Alpine* e *Griewank*, conforme visto na Figura 2.6 e na Tabela 2.3. Elas apresentam algumas funções *benchmark* clássicas utilizadas na avaliação de algoritmos de otimização.

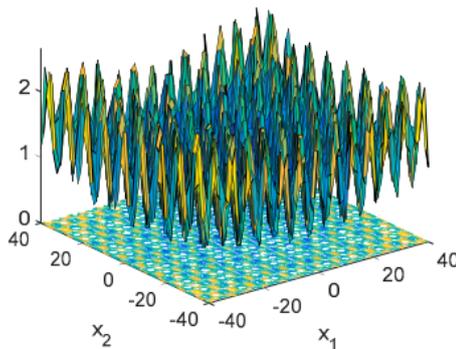
Tabela 2.2 – Comparativo sintético dos métodos de otimização estocásticos e determinísticos.

Critério	Métodos de Otimização	
	Determinísticos / Tradicionais	Estocásticos / Meta-heurísticos
Função unimodal	Eficaz	Eficaz
Função multimodal	Ineficaz	Eficaz
Função convexa	Eficaz	Eficaz
Função não convexa	Ineficaz	Eficaz
Função suave	Eficaz	Eficaz
Função não suave	Ineficaz, geralmente	Eficaz
Função e restrições altamente não lineares	Ineficaz	Eficaz
À base de gradiente	Sim, geralmente	Não, geralmente
À base de população / processamento paralelo	Não	Geralmente
Busca local / exploração	Eficaz	Eficaz
Busca global / exploração	Ineficaz	Grande probabilidade
Randomização para busca global e escape de mínimos locais	Não	Sim
Velocidade de convergência	Rápido	Depende do problema e método
Informações sobre o problema	Exigente	Não é exigente
Precisa de simplificações do problema	Geralmente	Não
Dependente do ponto de partida	Sim	Não

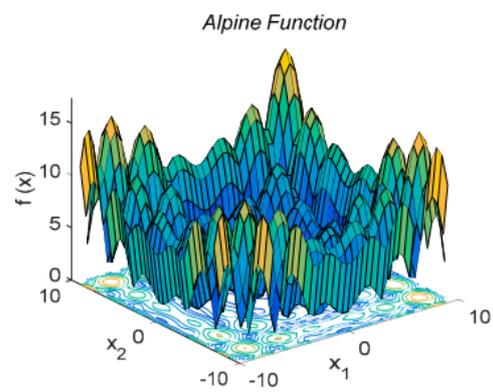


(a) multimodal function (f_1)

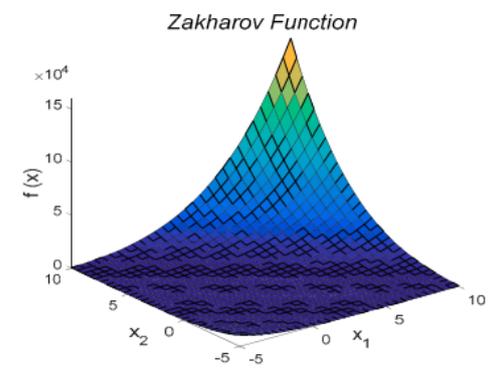
Griewank Function



(b) Non-smooth multimodal function (f_2)



(c) multimodal function (f_3)



(d) Hard unimodal function (f_4)

Figura 2.6 – Funções *benchmark* clássicas para a avaliação de algoritmos de otimização. Modificado de Saad et al. (2017).

Por outro lado, os métodos estocásticos são muito robustos na otimização dessas funções complexas.

Os métodos determinísticos quando aplicados em funções como a *Sphere* são bem eficientes e rápidos na convergência, pois normalmente são à base de gradiente e muito adequados para a otimização de funções suaves e unimodais. Nesses casos eles são mais precisos e mais rápidos que os métodos estocásticos. Mas, infelizmente funções desse tipo são pouco representativas dos problemas reais normalmente encontrados.

Tabela 2.3 – Funções *benchmark* clássicas para a avaliação de algoritmos de otimização. Modificado de Saad et al. (2017).

No	Function	Formula	f^*	D	Space	Properties
f_1	Sphere	$f(x) = \sum_{i=1}^D x_i^2$	0	50	$[-5.12 \ 5.12]^{50}$	Unimodal
f_2	Griewank	$f(x) = \sum_{i=1}^D \frac{x_i^2}{4000} - \prod \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	0	30	$[-100 \ 100]^{30}$	Multi-modal
f_3	Alpine	$f(x) = \sum_{i=1}^D x_i \sin(x_i) + 0.1x_i $	0	30	$[-10 \ 10]$	Multi-modal
f_4	Zakharov	$f(x) = \sum_{i=1}^D x_i^2 \left(\frac{1}{2} \sum_{i=1}^D ix_i \right)^2 + \left(\frac{1}{2} \sum_{i=1}^D ix_i \right)^4$	0	25	$[-5 \ 10]$	Hard convergence unimodal

2.1.7 - Otimização Mono-objetivo

Um problema de otimização mono-objetivo pode ser definido como

$$\begin{aligned}
 &\text{minimizar} && \mathbf{y} = \mathbf{f}(\mathbf{x}) \\
 &\text{sujeito a} && \mathbf{e}(\mathbf{x}) = (e_1(\mathbf{x}), e_2(\mathbf{x}), \dots, e_m(\mathbf{x})) \leq \mathbf{0} \\
 &\text{onde} && \mathbf{x} = (x_1, x_2, \dots, x_n) \in \mathbf{X}.
 \end{aligned} \tag{2.1}$$

A solução minimiza o escalar $f(\mathbf{x})$ onde \mathbf{x} é um vetor de variável de decisão de n -dimensões do universo \mathbf{X} . Observe que $\mathbf{e}(\mathbf{x})$ representa as restrições que precisam ser atendidas enquanto se otimiza (minimiza ou maximiza) $f(\mathbf{x})$. \mathbf{X} contém todos os possíveis \mathbf{x} que podem ser utilizados para satisfazer uma avaliação de $f(\mathbf{x})$ e suas respectivas restrições. Naturalmente, \mathbf{x} pode ser um vetor de variáveis discretas ou contínuas, assim como f pode ser contínua ou discreta.

O método para encontrar o ótimo global (pode não ser único) de qualquer função é chamado de otimização global. Dada a função $f : \mathbf{X} \subseteq \mathbb{R}^n \rightarrow \mathbb{R}$, $\mathbf{X} \neq \emptyset$, para $\mathbf{x} \in \mathbf{X}$ o valor $f^* \triangleq f(\mathbf{x}^*) > -\infty$ é considerado o mínimo global se, e apenas se, $\forall \mathbf{x} \in \mathbf{X} : f(\mathbf{x}^*) \leq f(\mathbf{x})$. \mathbf{x}^* é a definição da solução mínima global, f é a função objetivo, e o conjunto \mathbf{X} é a região viável de \mathbf{x} , veja Coelho et al. (2007).

2.1.8 - Otimização Multiobjetivo

Um problema geral de otimização multiobjetivo inclui um conjunto de n parâmetros, que são as variáveis de decisão, um conjunto de k funções objetivo, e um conjunto de m restrições. As funções objetivo e as restrições são funções das variáveis de decisão. O principal objetivo da otimização é

$$\begin{aligned} \text{minimizar} \quad & \mathbf{y} = \mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_k(\mathbf{x})) \\ \text{sujeito a} \quad & \mathbf{e}(\mathbf{x}) = (e_1(\mathbf{x}), e_2(\mathbf{x}), \dots, e_m(\mathbf{x})) \leq \mathbf{0} \\ \text{onde} \quad & \mathbf{x} = (x_1, x_2, \dots, x_n) \in \mathbf{X} \\ & \mathbf{y} = (y_1, y_2, \dots, y_k) \in \mathbf{Y}, \end{aligned} \tag{2.2}$$

no qual \mathbf{x} é o vetor de decisão, \mathbf{y} é o vetor objetivo, \mathbf{X} representa o espaço de decisão e \mathbf{Y} representa o espaço objetivo; as restrições $\mathbf{e}(\mathbf{x}) \leq \mathbf{0}$ determinam o conjunto de soluções aceitáveis.

2.1.8.1 - Otimalidade e dominância de Pareto

A noção de otimalidade usada atualmente é adotada do trabalho de Francis Ysidro Edgeworth e Vilfredo Pareto (Coello Coello et al., 2007) e é comumente referida como otimalidade de Pareto. Esta, considera que soluções são superiores ou inferiores a outra solução somente quando são superiores em todos os objetivos ou inferiores em todos os objetivos, respectivamente. As compensações (*tradeoffs*) em um problema multiobjetivo (MOP) são capturadas por soluções que são superiores em alguns objetivos, mas, inferiores em outros. Tais pares de soluções que são ambos superiores e inferiores com relação a certos objetivos, são chamados não-dominados (Hadka, 2016), como mostrado na Figura 2.7.

Mais formalmente, a noção de otimalidade de Pareto é definida pela relação de dominância de Pareto:

Um vetor $\mathbf{u} = (u_1, u_2, \dots, u_M)$, Pareto domina outro vetor $\mathbf{v} = (v_1, v_2, \dots, v_M)$, se e somente se, $\forall i \in \{1, 2, \dots, M\}, u_i \leq v_i$ e $\exists j \in \{1, 2, \dots, M\}, u_j < v_j$. Isto é denotado por $\mathbf{u} \prec \mathbf{v}$. Normalmente, a relação de dominância é aplicada aos objetivos.

Duas soluções são não dominadas se nenhuma Pareto domina a outra (isto é, $\mathbf{u} \not\prec \mathbf{v}$ e $\mathbf{v} \not\prec \mathbf{u}$). O conjunto de todas as soluções não dominadas é capturado pelo conjunto ótimo de Pareto (*Pareto optimal set*) e a frente de Pareto (*Pareto front*). O primeiro contém as variáveis de decisão, enquanto a última contém os objetivos.

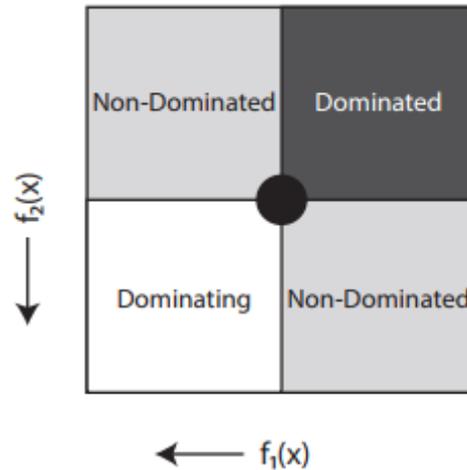


Figura 2.7 – Representação de várias regiões da dominância de Pareto, no caso de minimização de um MOP. Estas regiões são relativas a cada solução, que é centrada na figura. A região dominada é inferior em todos os objetivos, a região dominante é superior em todos os objetivos e a região não dominada é superior em um objetivo, mas, inferior no outro, (Hadka, 2016).

Para um dado problema multiobjetivo, o conjunto ótimo de Pareto (*Pareto optimal set*) é definido por:

$$P^* = \{x \in \Omega \mid \neg \exists x' \in \Lambda, F(x') \prec F(x)\} \quad (2.3)$$

Onde Ω é o espaço de decisão; é o conjunto de todas as variáveis de decisão. O conjunto de todas as variáveis de decisão em Ω , que são possíveis (isto é, satisfazem todas as restrições) define a região possível (*feasible region*), Λ .

Para um dado problema multiobjetivo com um *Pareto optimal set* P^* , a *Pareto front* é definida por:

$$PF^* = \{F(x) \mid x \in P^*\} \quad (2.4)$$

A Figura 2.8 ilustra um exemplo de *Pareto optimal set* e *Pareto front*, e o mapeamento resultante entre os dois. O *Pareto optimal set* define as variáveis de decisão, enquanto a *Pareto front* captura os objetivos e seus *tradeoffs*, via otimalidade de Pareto.

2.1.9 - Métricas de Performance de um MMA

Quando otimizando problemas multiobjetivo (MOPs), os algoritmos meta-heurísticos multiobjetivo (MMAs) retornam uma aproximação do conjunto ótimo de Pareto e da frente de Pareto. A aproximação da frente de Pareto, também chamada conjunto de aproximação, pode ser usada para medir a qualidade de um MMA em um problema particular. Em algumas situações, como em problemas definidos específicos para teste, um conjunto de

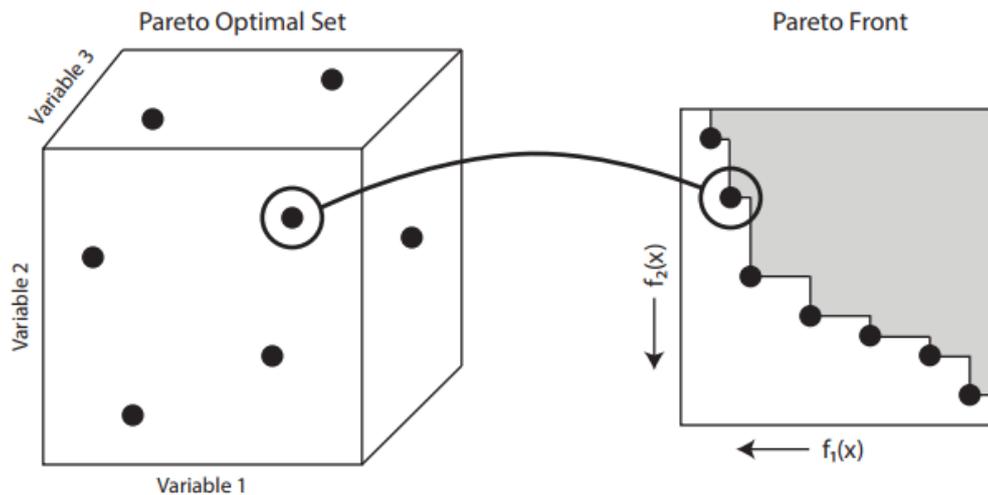


Figura 2.8 – Representação do mapeamento hipotético entre um *Pareto optimal set* 3-dimensional e sua *Pareto front* 2-dimensional associada. A área sombreada/cinza (*Pareto front*) mostra o espaço dominado pela *Pareto front*, (Hadka, 2016).

referência das soluções ótimas globais pode ser conhecido. Se conhecido, este conjunto pode ser usado para medir a performance absoluta de um MMA. Se não conhecido, os conjuntos de aproximação obtidos de múltiplos MMAs podem ser comparados para determinar sua qualidade relativa (Hadka, 2016).

Não há consenso na literatura do procedimento apropriado para a comparação dos conjuntos de aproximação. Esses procedimentos, chamados métricas de performance, são classificados em duas formas: métricas unárias e binárias de performance (Zitzler et al., 2002b). Métricas unárias de performance produz um único valor numérico com o qual se compara os conjuntos de aproximação. Métricas unárias de performance tem a vantagem de permitir a comparação de conjuntos de aproximação sem exigir o conjunto de aproximação real, pois, é necessário apenas comparar os valores numéricos. Métricas binárias de performance, por outro lado, compara pares de conjuntos de aproximação, identificando qual dos dois conjuntos de aproximação é superior. Para permitir comparações futuras com essa pesquisa, serão adotadas nela somente métricas unárias de performance.

Zitzler et al. (2002c) afirmam que o número de métricas unárias de performance exigido para determinar se um conjunto de aproximação é melhor do que outro deve ser pelo menos igual à quantidade de objetivos do problema. Porque diferentes MMAs tendem performarem melhor em diferentes métricas, Deb e Jain (2002) sugerem usar métricas somente para os dois objetivos funcionais principais dos MMAs: proximidade e diversidade. A seguir são descritas várias métricas unárias de desempenho comumente utilizadas no meio científico.

2.1.9.1 - Hipervolume

O hipervolume representa o volume do espaço dominado pelo conjunto de aproximação, conforme mostra a Figura 2.9. Este volume é delimitado por um ponto de referência adotado, que é usualmente o pior ponto (ou seja, o pior caso/valor de cada objetivo) do conjunto de referência, acrescido de um incremento fixo. Este incremento fixo é necessário para permitir pontos externos no conjunto de aproximação contribuir para o hipervolume. Knowles e Corne (2002) recomenda o uso do hipervolume, pois, é compatível com relações de superioridade, independe de escala, é intuitivo e pode refletir o grau de superioridade entre dois conjuntos de aproximação. Ao aplicar vários MMAs na minimização de um dado MOP, de acordo com essa métrica de performance, será considerado de desempenho superior o MMA que gerar o maior hipervolume.

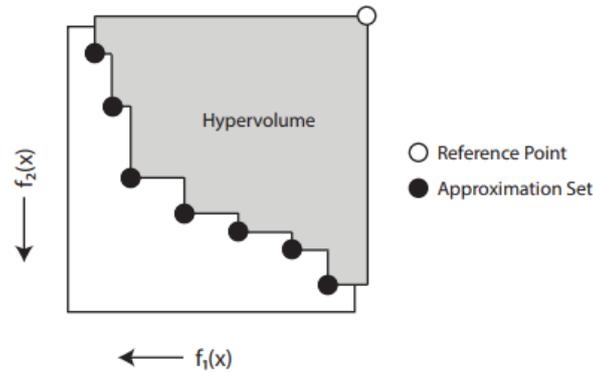


Figura 2.9 – Representação esquemática do hipervolume (Hadka, 2016).

A maior desvantagem do hipervolume é sua complexidade em tempo de processamento, dada por $O(n^{M-1})$, onde n é tamanho do conjunto não dominado. Todavia, Beume e Rudolph (2006) forneceram uma implementação com complexidade $O(n \log n + n^{M/2})$. Esta implementação permite computar o hipervolume em conjuntos não dominados moderadamente grandes, com até $M = 8$ objetivos, em um intervalo de tempo razoável. Aprimoramentos adicionais feitos por While et al. (2012) reduziram ainda mais o tempo de execução esperado, permitindo o cálculo eficiente do hipervolume de MOP com dez ou mais objetivos.

2.1.9.2 - Distância geracional

Distância geracional (GD) é a média das distâncias de cada solução no conjunto de aproximação à solução mais próxima no conjunto de referência, conforme mostrado na Figura 2.10. Dessa forma, ela mede a proximidade ao conjunto de referência. A GD por si só pode não ser conclusiva, já que um conjunto de aproximação contendo uma simples solução bem próxima ao conjunto de referência produz baixas medidas de GD; por isso, na

prática, essa métrica é normalmente combinada com medidas de diversidade (Hadka e Reed, 2012).

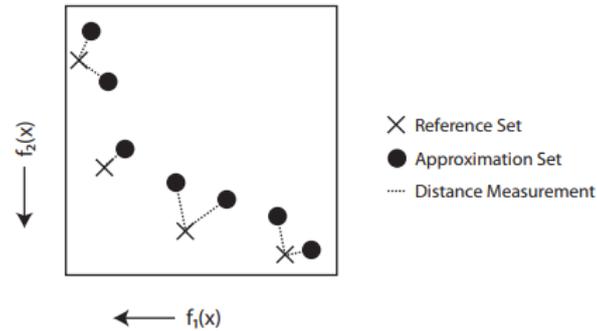


Figura 2.10 – Representação esquemática da distância geracional (Hadka, 2016).

2.1.9.3 - Distância geracional invertida

Conforme o nome sugere, a distância geracional invertida (IGD) é obtida de forma inversa à da GD. Ela é a distância média de cada solução no conjunto de referências à solução mais próxima no conjunto de aproximação. A IGD mede diversidade, como mostra a Figura 2.11, uma vez que requer que um conjunto de aproximação tenha soluções próximas a cada ponto do conjunto de referência para se obter baixas medidas de IGD (Coello Coello et al., 2007).

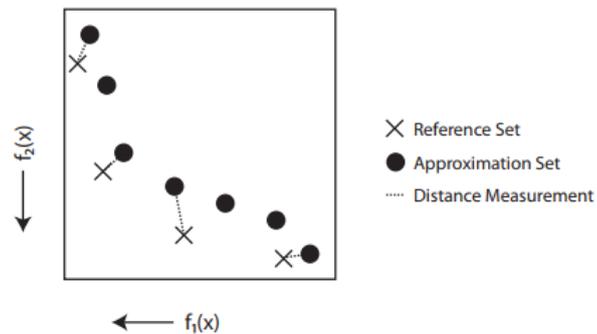


Figura 2.11 – Representação esquemática da distância geracional invertida (Hadka, 2016).

2.1.9.4 - Indicador ϵ_+

O indicador épsilon aditivo (ϵ_+) mede a menor distância ϵ que um conjunto de aproximação tem que ser transladado para dominar completamente o conjunto de referência, como apresenta a Figura 2.12. Observa-se que boa proximidade e boa diversidade resultam em baixos valores de ϵ , uma vez que a distância que o conjunto de aproximação precisa ser transladado é reduzida. Entretanto, se existir uma região do conjunto de referência que é pobremente aproximada pelas soluções no conjunto de aproximação, um grande ϵ será

requerido. Portanto, o indicador ϵ_+ mede a consistência de um conjunto de aproximação. Um conjunto de aproximação não deve ter grandes *gaps* ou regiões de aproximação pobre para ser considerado consistente (Hadka e Reed, 2013).

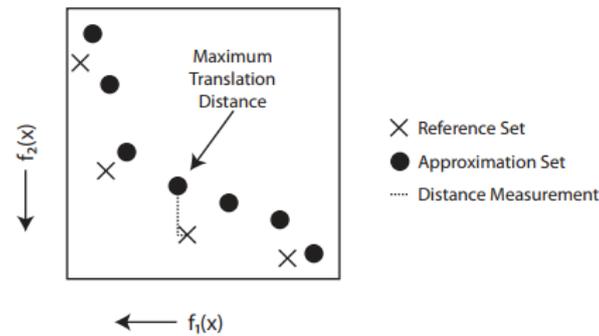


Figura 2.12 – Representação esquemática do indicador ϵ_+ (Hadka, 2016).

2.1.9.5 - Espaçamento

Espaçamento, esquematizado na Figura 2.13, mede a uniformidade dos espaços entre as soluções em um conjunto de aproximação (Coello Coello et al., 2007). Um conjunto de aproximação que é bem espaçado não contém aglomerados densos de soluções separados por grandes extensões vazias. Uma vez que o espaçamento não envolve um conjunto de referência em seu cálculo, um conjunto de aproximação pode apresentar bom espaçamento, mas, ter uma proximidade pobre em relação ao conjunto de referência. Portanto, é recomendado usar essa métrica em conjunto com uma métrica de performance de proximidade.

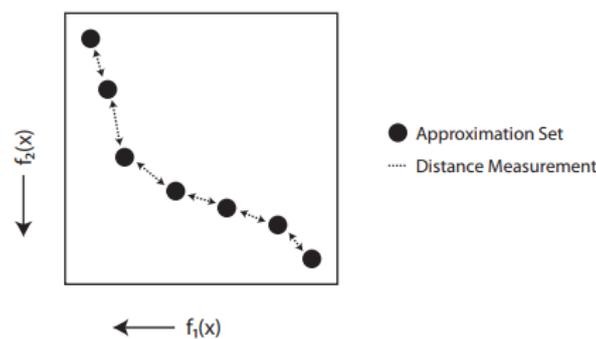


Figura 2.13 – Representação esquemática do espaçamento (Hadka, 2016).

2.1.9.6 - Considerações para a aplicação das métricas de performance

Segundo Fonseca e Fleming (1996), em trabalhos acadêmicos, é comum ver os resultados publicados usando GD, hipervolume e o indicador ϵ_+ . Estas três métricas avaliam proximidade, diversidade e consistência, respectivamente; também, eles consideram que são os três principais objetivos funcionais dos MMAs.

A Figura 2.14 fornece uma representação gráfica da importância do indicador ϵ_+ como uma métrica de consistência. A Figura 2.14a mostra uma boa aproximação ao conjunto de referência, representado pela linha tracejada. A distância geracional apresenta uma média da distância entre o conjunto de aproximação e o conjunto de referência, reduzindo o impacto de grandes *gaps*, veja a Figura 2.14b, onde os pontos hipoteticamente faltantes estão em cinza claro. A Figura 2.14c ilustra que há uma pequena variação no hipervolume, devida a um *gap*, em relação ao hipervolume total. Mas, em contrapartida, o indicador ϵ_+ facilmente identifica o *gap*, Figura 2.14d, reportando uma métrica 2 a 3 vezes pior neste exemplo (Hadka, 2016). É esperado que os MMAs produzam soluções de alta qualidade cobrindo toda a extensão da superfície de *tradeoff*, com poucos *gaps* ou regiões de aproximação pobre.

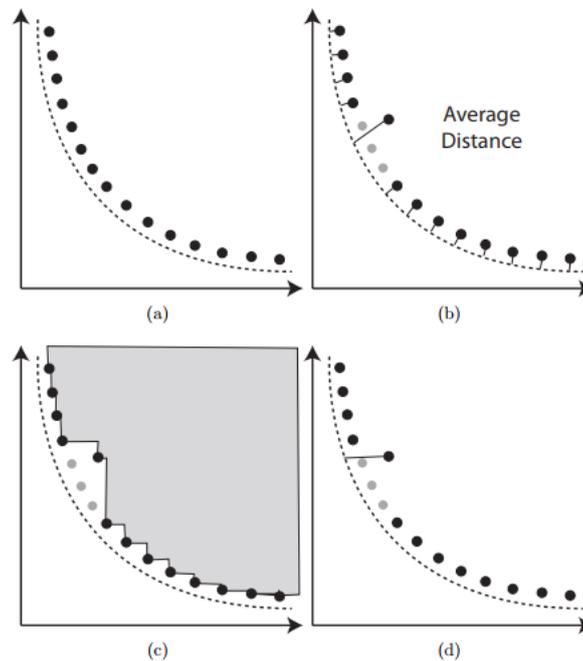


Figura 2.14 – Análise conjunta das métricas de performance consideradas (Hadka, 2016).

Normalmente, para reportar consistentemente essas métricas de performance, todos os seus valores são normalizados. Esta normalização converte todas as métricas de performance para residirem no intervalo $[0, 1]$, com 1 representando o valor ótimo. Primeiro, o conjunto de referência é normalizado pelos seus limites máximo e mínimo tal que todos os pontos no conjunto de referência estejam em $[0, 1]^N$, o hipercubo unitário N-dimensional. Segundo, cada conjunto de aproximação é normalizado usando os mesmos limites. Terceiro, as métricas de performance são calculadas usando esses conjuntos normalizados. Finalmente, as métricas de performance são transformadas pelas equações seguintes, para garantir que um valor 1 represente o valor ótimo obtido pela métrica (Hadka, 2016). O hipervolume é transformado com:

$$M(A_P^S) = \widehat{M}(A_P^S)/M^*, \quad (2.5)$$

onde \widehat{M} representa o valor bruto da métrica. A GD e o indicador ϵ_+ são transformados com:

$$M(A_P^S) = \max(1 - \widehat{M}(A_P^S), 0). \quad (2.6)$$

Quando se resolve problemas teste, o conjunto de referência é normalmente conhecido analiticamente. Entretanto, para a maioria dos problemas do mundo real, o conjunto de referência não está disponível. Nessas situações, é necessário construir um conjunto de referência a partir da união de todos os conjuntos de aproximação gerados durante os experimentos. Então, as métricas de performance podem ser avaliadas em relação a este conjunto de referência combinado.

2.1.9.7 - Ranqueamento dos MMAs com função utilidade e as métricas de performance

Conforme apresentado na Seção 2.1.8 e em consonância com Rao (2019), em geral, não existe um vetor de decisão (solução) que minimiza simultaneamente todas as funções objetivo, para um dado problema de otimização multiobjetivo. Os algoritmos normalmente geram um conjunto de soluções ótimas de Pareto e usam algum critério (ou regra) adicional para selecionar uma solução ótima de Pareto particular como a solução do problema. Um método muito utilizado para essa tarefa é o da função utilidade. Nele, uma função utilidade é definida $U_i(f_i)$ para cada objetivo, dependendo da importância de f_i comparada às outras funções objetivo. Após, uma função utilidade geral (ou total) é definida, por exemplo, como

$$U = \sum_{i=1}^k U_i(f_i). \quad (2.7)$$

O vetor solução x^* é então encontrado maximizando a função utilidade total U , sujeita às restrições $e_j(x) \leq 0$, $j = 1, 2, \dots, m$. Uma forma simples da equação (2.7) é dada por

$$U = \sum_{i=1}^k U_i = - \sum_{i=1}^k w_i f_i(x), \quad (2.8)$$

onde w_i é um fator de ponderação escalar associado com a i -ésima função objetivo. Este método [equação (2.8)] é conhecido também como método da função de ponderação. Este método será utilizado para classificar os MMAs quanto à performance quando aplicados na otimização dos modelos propostos nessa pesquisa.

As métricas de performance: hipervolume, distância geracional, distância geracional invertida, indicador ϵ_+ e espaçamento, serão as funções objetivo utilizadas na equação (2.8) para classificar o desempenho dos MMAs na otimização dos modelos ILMF. Com exceção do hipervolume, para todas as outras métricas listadas, quanto menor o valor, melhor foi a performance do algoritmo na otimização. Portanto, será assumido o hipervolume com sinal negativo na soma dada pela função de ponderação. Assim, o MMA que apresentar o menor

valor para a equação (2.8) terá a melhor performance para a otimização do modelo em questão. Todas as métricas terão a mesma importância nessa análise, ou seja, $w_i = 1, i = 1, \dots, k$.

2.1.10 - Otimização Estrutural

2.1.10.1 - Conceitos gerais

Na otimização estrutural os objetivos de projeto são critérios estruturais usados para avaliar o mérito dele, como: custo mínimo de construção, custo mínimo do ciclo de vida, peso mínimo e rigidez máxima. As recomendações das normas, que fornecem requisitos de segurança e usabilidade para a estrutura, geralmente aparecem como restrições de projeto. Além disso, algumas restrições não comportamentais, como o tipo e o tamanho dos elementos estruturais disponíveis, podem limitar os projetos aceitáveis. As variáveis de projeto podem descrever a forma, a topologia e a geometria da estrutura, ou podem definir o tamanho ou as propriedades dos elementos estruturais, Gandomi et al. (2013), entre outras aplicações.

2.1.10.2 - Método dos critérios de otimalidade (OC)

Os métodos de programação matemática (MP) e OC são duas grandes categorias de técnicas de otimização numérica. Usando métodos MP, a função objetivo é minimizada diretamente; mas, com o método OC, um critério é definido tal que quando ele é satisfeito, o ótimo é obtido. Neste método podem ser definidos vários critérios também.

A essência desses métodos é primeiro estabelecer os critérios que definem um ótimo e então criar um algoritmo recursivo que atenda a esses critérios. Esses métodos podem ser muito eficientes quando o número de restrições é pequeno comparado com o número de variáveis de projeto. Eles são dependentes do problema porque usam características conhecidas ou assumidas de um ótimo ou de um comportamento especial da estrutura. Os métodos OC podem ser expressões matemáticas rigorosas ou intuitivas, como a assunção de que a densidade de energia de deformação na estrutura ótima é uniforme. Foram desenvolvidos nas décadas de 70 e 80 com o propósito de otimização de grandes estruturas.

2.1.10.3 - Otimização baseada na confiabilidade

Esse método de otimização foi iniciado na década de 60. Ele se fundamenta no fato de que as propriedades de resistência dos elementos estruturais e a magnitude dos carregamentos são

probabilísticos, portanto, podem ser tratados como variáveis aleatórias com uma distribuição estatística especificada. Nesse método, os valores ótimos das variáveis de projeto são obtidos tal que a função objetivo seja minimizada considerando a probabilidade geral de falha, enquanto os carregamentos atuantes na estrutura e as resistências estruturais são variáveis aleatórias. A probabilidade geral de falha é calculada considerando diferentes modos de falha sob diferentes condições de carga.

Adeli e Sarma (2006) afirmam que a probabilidade de falha e o custo esperado da falha não podem ser calculados de forma consistente devido à insuficiência de dados estatísticos; eles têm de ser escolhidos meio que arbitrariamente. Além disso, a existência de um grande número de modos de falha possíveis, especificamente para grandes estruturas, faz da avaliação da confiabilidade do sistema, praticamente uma tarefa impossível. Eles concluíram que todos os trabalhos publicados sobre otimização de custos baseada em confiabilidade de estruturas adotam uma abordagem acadêmica, teórica e idealista para o problema.

2.1.10.4 - Otimização *fuzzy*

Em otimização *fuzzy*, que é baseada na teoria dos conjuntos *fuzzy*, proposta por Lotfi A. Zadeh em 1965, valores numéricos das funções de pertinência (*membership functions*) são usados, ao invés das probabilidades empregadas na otimização baseada na confiabilidade. No projeto estrutural, as duas maiores fontes de *fuzziness*, imprecisão, ou incertezas podem ser identificadas, uma na avaliação do comportamento e resistência estruturais, e a outra na determinação dos carregamentos atuantes na estrutura. Na otimização do custo da estrutura, uma terceira fonte de *fuzziness* e imprecisão aparece, que é inerente à formulação e avaliação da função de custo, vide Adeli e Sarma (2006).

Um conjunto *fuzzy* Y para qualquer conjunto Z é caracterizado por sua função de pertinência $\mu_Y(Z)$ que avalia cada ponto em Z com um valor no intervalo $[0, 1]$. Esta *membership function* é a nota da pertinência de Z em Y . Quanto mais próximo da unidade for o valor de $\mu_Y(Z)$, maior é a nota da pertinência de Z em Y . Dessa forma, um conjunto *fuzzy* é definido como $Y = \{z, \mu_Y(Z)\} | z \in Z$.

A teoria dos conjuntos *fuzzy* pode ser usada para modelar julgamentos em situações ambíguas, imprecisas ou difusas. As funções de pertinência de um conjunto *fuzzy* são usadas para desenvolver uma transição difusa da aceitação total para a rejeição total de certos processos de decisão.

2.1.11 - O Impacto Econômico e Ambiental da Otimização Estrutural

Lagaros (2018) apresenta um estudo muito detalhado a respeito desse tema. Ele aponta que o setor de construção é o maior emissor de gases de efeito estufa (GEE). Este setor consome globalmente mais de 40% da energia, 25% da água, 40% dos recursos de forma geral e emite quase 30% de GEE. Também, é responsável por 10% do PIB global (US\$ 7.5 trilhões) e emprega mais de 10 milhões de pessoas e, tem uma expectativa de expansão de 85% até 2030. Por isso, a demanda de material para a construção civil representa um dos maiores desafios do uso de recursos atualmente e, apesar desse consumo nem sempre se manifestar como um problema direto e observável, questões como a perda de diversidade, mudanças climáticas, desertificação e erosão no solo estão relacionados ao uso extensivo de material. Os números mostram que a demanda por aço na construção e infraestrutura responde por mais da metade da demanda global por aço e, a demanda por cimento (consequentemente concreto) é praticamente toda oriunda desses setores. A partir desses dados, foi feita uma simulação assumindo que a otimização de projetos reduz cerca de 5 a 10% do consumo de materiais e, a economia e a redução do impacto ambiental calculadas e projetadas até 2050, foram astronômicas.

A partir de todos esses dados, o autor conclui que apesar de atualmente existir um uso tímido de técnicas de otimização na construção civil, seja pelo ceticismo dos engenheiros projetistas quanto à eficácia e eficiência dos métodos quando do projeto de estruturas e construção, de forma geral; seja pela pouca oferta de softwares robustos com modelagem e otimização integradas (citou o MIDAS NFX 2017, entre outros, que apresentam capacidade insipiente de otimização de topologia, e que não cumprem os requisitos dos engenheiros estruturais profissionais); até 2030 haverá uma mudança cultural forçada e os processos de projeto, principalmente o estrutural, via de regra, incluirão otimização; o que já acontece naturalmente, segundo ele, nas engenharias mecânica e aeroespacial de ponta.

2.2 - PACKAGES, FRAMEWORKS E TOOLBOXES DE OTIMIZAÇÃO

2.2.1 - Python Packages para Otimização Multiobjetivo

Existe um crescente interesse dentro da comunidade científica em *frameworks* implementadas em Python, uma vez que esta linguagem oferece um vasto ecossistema de *packages* (bibliotecas), sendo a maioria delas dedicadas à análise, processamento e visualização de dados. Quando se trata de algoritmos de otimização, um conjunto de *frameworks* representativas é listado na Tabela 2.4, onde Benítez-Hidalgo et al. (2019) as analisam de acordo com: os domínios dos algoritmos, status de manutenção, versão do

Python, tipo de licença, variantes apresentadas, facilidades no pós-processamento e algoritmos disponíveis.

A partir dessa tabela resumo, pode-se observar que os *packages* PyGMO, Platypus e jMetalPy são os mais robustos para a tarefa de otimização de problemas multiobjetivo, pois, além de oferecerem uma vasta quantidade de algoritmos multiobjetivo representativos do estado da arte em otimização, permitem o processamento paralelo, otimização restrita, facilidades no pós-processamento e visualização dos resultados, entre outras vantagens. Além do fato de serem *softwares* livres.

2.2.2 - Paradise Framework

Paradiseo é uma *framework* "caixa-branca" (código aberto), orientada a objetos, baseada em uma clara separação conceitual da meta-heurística e dos problemas a que é destinada a resolver. Essa separação e a grande variedade de recursos de otimização implementados permitem um máximo reuso do código. A separação é expressa a nível de implementação dividindo as classes em duas categorias: classes fornecidas e classes requeridas. As classes fornecidas constituem uma hierarquia de classes implementando a parte invariante do código. Usuários experientes podem estender a estrutura por herança / especialização. As classes requeridas, que codificam a parte específica do problema, são classes abstratas que precisam ser especializadas e implementadas pelo usuário (Talbi, 2009).

Alterar componentes existentes, e adicionar novos, pode ser feito facilmente sem afetar o restante da aplicação. Essa flexibilidade é possível devido ao uso da tecnologia de desenvolvimento orientado a objetos. *Templates* (característicos da linguagem C++) são usados para modelar as características de meta-heurística: estruturas de codificação, operadores de transformação, critérios de parada e assim por diante. Esses *templates* podem ser instanciados pelo usuário de acordo com os parâmetros dependentes do problema. Os mecanismos de orientação a objetos, como herança, polimorfismo, e assim por diante, são maneiras poderosas de projetar novos algoritmos ou evoluir os já existentes. Além disso, Paradiseo integra vários serviços que a torna mais fácil de usar, incluindo aplicativos de visualização, definição online de parâmetros, aplicativos de pontos de verificação, entre outros.

Paradiseo disponibiliza os modelos paralelos e distribuídos mais comuns. Estes modelos dizem respeito aos três principais modelos paralelos: nível algorítmico, nível de iteração e nível de solução. Eles são portáteis em diferentes tipos de arquiteturas: máquinas de memória distribuída e multiprocessadores de memória compartilhada, uma vez que são implementados usando bibliotecas padrão, como *Message Passing Interface* (MPI),

Tabela 2.4 – *Frameworks* de otimização mais populares, escritas em Python (modificado de Benítez-Hidalgo et al. (2019)).

<i>Name</i>	<i>Status</i>	<i>Python version</i>	<i>License</i>	<i>Parallel processing</i>	<i>Dynamic optimization</i>	<i>Constrained optimization</i>	<i>Decision making</i>	<i>Post-processing facilities</i>	<i>Algorithms</i>
DEAP 1.2.2	Active	≥2.7	LGPL-3.0	✓		✓		Statistics	GA, GP, CMA-ES, NSGA-II, SPEA2, MO-CMA-ES
Geatpy 1.1.5	Active	≥3.5	MIT						GA, MOEA
Inspyred 1.0.1	Inactive	≥2.6	MIT			✓			GA, ES, PSO, ACO, SA, PAES, NSGA-II
PyGMO 2.10	Active	3.x	GPL-3.0	✓		✓		Visualization, statistics	GA, DE, PSO, SA, ABC, IHS, MC, CMA-ES, NSGA-II, MOEA/D
Platypus 1.0.3	Active	3.6	GPL-3.0	✓		✓		Visualization, statistics	CMA-ES, NSGA-II, NSGA-III, GDE3, IBEA, MOEA/D, OMOPSO, EpsMOEA, SPEA2
Pymoo 0.2.4	Active	3.6	Apache 2.0				✓	Visualization, statistics	GA, DE, NSGA-II, NSGA-III, U-NSGA-III, reference point (R-NSGA-III)
jMetalPy 1.5.0	Active	≥3.6	MIT	✓	✓	✓	✓	Visualization, statistics	GA, EA, SPEA2, NSGA-II, NSGA-III, SMPSO, GDE3, OMOPSO, HypE, IBEA, MOCcell, MOEA/D-DE, MOEA/D-DRA, constrained (MOEA/D-Epsilon) reference point (G-NSGA-II, SMPSO/RF, G-GDE3), dynamic (NSGA-II, SMPSO, GDE3)

multithreading (Pthreads) ou *grid middlewares (Condor ou Globus)*. Os modelos podem ser explorados de forma transparente, para isso, é preciso apenas instanciar seus componentes Paradiseo associados. O usuário tem a possibilidade de escolher por uma instanciação simples para a camada de comunicação. Os modelos foram validados em problemas acadêmicos e industriais. Os resultados experimentais demonstram sua eficiência. Os testes também demonstram a alta capacidades de reutilização e que o usuário refaz pouco código. Além disso, a *framework* fornece os mecanismos de hibridização mais comuns, que permitem fazer meta-heurísticas cooperativas pertencentes à mesma família ou a famílias diferentes, vide Talbi (2009).

ParadisEO é uma *framework* de código aberto C++ LGPL (*STL-Template*). Isto é, portátil no Windows, sistemas semelhantes ao Unix, como Linux e MacOS. Ela inclui o seguinte conjunto de módulos (Figura 2.15):

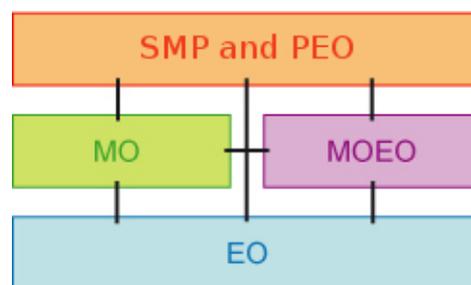


Figura 2.15 – Módulos da Paradiseo *framework* e suas interações.

- *Evolving Objects (EO)*: A biblioteca EO foi desenvolvida inicialmente para algoritmos evolucionários (algoritmos genéticos, estratégias de evolução, programação evolucionária, programação genética e algoritmos de distribuição de estimativas). Ele foi estendido para meta-heurísticas baseadas em populações, como otimização com enxame de partículas e otimização com colônia de formigas;
- *Moving Objects (MO)*: inclui meta-heurísticas baseadas em solução única, como busca local, recozimento simulado, busca tabu e busca local iterada;
- *Multiobjective Evolving Objects (MOEO)*: Inclui os mecanismos de busca para resolver problemas de otimização multiobjetivo, como atribuição de aptidão / *fitness*, diversificação e elitismo. A partir desse conjunto de mecanismos, algoritmos clássicos como NSGA-II, SPEA2 e IBEA foram implementados e estão disponíveis;
- *Parallel Evolving Objects (PEO)*: Inclui os modelos paralelos e distribuídos para meta-heurísticas e sua hibridação.

Outro fator essencial é o desempenho na execução das rotinas pesadas de otimização, pois, essa biblioteca é implementada em C++, que é uma linguagem compilada. A literatura correlata reporta que a performance das linguagens de programação compiladas costuma ser muitas vezes maior do que a das linguagens interpretadas, como é o caso de VBA (*Visual Basic for Applications*), MATLAB, Maple, entre outras.

Linguagens compiladas, como: Assembler, COBOL, PL/I, C/C++, C#, VB e etc., são todas traduzidas executando o código-fonte através de um compilador. Isso resulta em um código muito eficiente que pode ser executado qualquer número de vezes, sem a necessidade de interpretação em tempo de execução (*runtime*). A sobrecarga para a tradução é feita apenas uma vez, quando o código-fonte é compilado; depois disso, só precisa ser carregado e executado. Ao contrário, nas linguagens interpretadas, cada linha de um programa deve ser analisada, interpretada e executada toda vez que o programa for executado, aumentando assim o custo de execução do programa. Por essa razão, os programas interpretados geralmente são menos eficientes que os programas compilados.

2.2.3 - Toolbox de Otimização Global do MATLAB

A *toolbox* de otimização global do MATLAB fornece métodos que buscam soluções globais para problemas que contêm múltiplos máximos ou mínimos. Inclui busca global, *multistart*, busca de padrões, algoritmo genético e *solvers* de recozimento simulado. Esses *solvers* podem ser usados para resolver problemas de otimização em que a função objetivo ou restrição é contínua, descontínua, estocástica, não possui derivada ou inclui simulações ou funções "caixa-preta" com valores indefinidos para algumas configurações de parâmetros. Os *solvers* de algoritmo genético e de busca de padrões suportam a customização algorítmica. Pode ser criada uma variante de algoritmo genético personalizada modificando a população inicial e as opções de *fitness scaling* ou definindo as funções de seleção de matrizes, *crossover* e mutação. A busca de padrões pode ser personalizada definindo *polling*, busca e outras funções (MathWorks, 2014).

Características principais:

- Ferramentas interativas para definir e resolver problemas de otimização e monitorar o progresso da solução;
- *Solvers* de busca global e *multistart* para encontrar ótimos globais únicos ou múltiplos;
- *Solver* de algoritmo genético que suporta restrições lineares, não lineares e laterais;
- Algoritmo genético multiobjetivo com identificação de frente de Pareto, incluindo restrições lineares e laterais;

- *Solver* de busca de padrões que suporta restrições lineares, não lineares e laterais;
- Ferramentas de recozimento simulado que implementam um método de busca aleatória, com opções para definir o processo de recozimento, programação de temperatura e critérios de aceitação;
- Suporte para computação paralela nos *solvers* de *multistart*, de algoritmo genético e de busca de padrões;
- Suporte para tipo de dados personalizado nos *solvers* de algoritmo genético, algoritmo genético multiobjetivo e recozimento simulado.

No total há sete *solvers* na *toolbox*: GA mono-objetivo (função *ga*), GA multiobjetivo (função *gamultiobj*), busca global (função *GlobalSearch*), *multistart* (função *MultiStart*), busca de padrões (função *patternsearch*), enxame de partículas (função *particleswarm*) e recozimento simulado (função *simulannealbnd*).

A Tabela 2.5 apresenta uma indicação do *solver* a ser utilizado, a partir dos tipos da função objetivo e das restrições e, da solução desejada; facilitando a seleção do mais adequado.

Tabela 2.5 – Seleção do *solver* na *toolbox* a partir dos tipos da função objetivo e das restrições e, da solução desejada.

Solução Desejada	Função Objetivo e Restrições Suaves	Função Objetivo e Restrições não Suaves
Solução local única	Funções de otimização na <i>toolbox</i>	<i>fminbnd</i> , <i>patternsearch</i> , <i>fminsearch</i> , <i>ga</i> , <i>particleswarm</i> , <i>simulannealbnd</i>
Múltiplas soluções locais	<i>GlobalSearch</i> , <i>MultiStart</i>	-
Solução global única	<i>GlobalSearch</i> , <i>MultiStart</i> , <i>patternsearch</i> , <i>ga</i> , <i>simulannealbnd</i>	<i>patternsearch</i> , <i>ga</i> , <i>particleswarm</i> , <i>Simulannealbnd</i>
Solução local única usando processamento paralelo	<i>MultiStart</i> , Funções de otimização na <i>toolbox</i>	<i>patternsearch</i> , <i>ga</i> , <i>particleswarm</i>
Múltiplas soluções locais usando processamento paralelo	<i>MultiStart</i>	-
Solução global única usando processamento paralelo	<i>MultiStart</i>	<i>patternsearch</i> , <i>ga</i> , <i>particleswarm</i>

A Figura 2.16 mostra a interface da *toolbox* de otimização global do MATLAB.

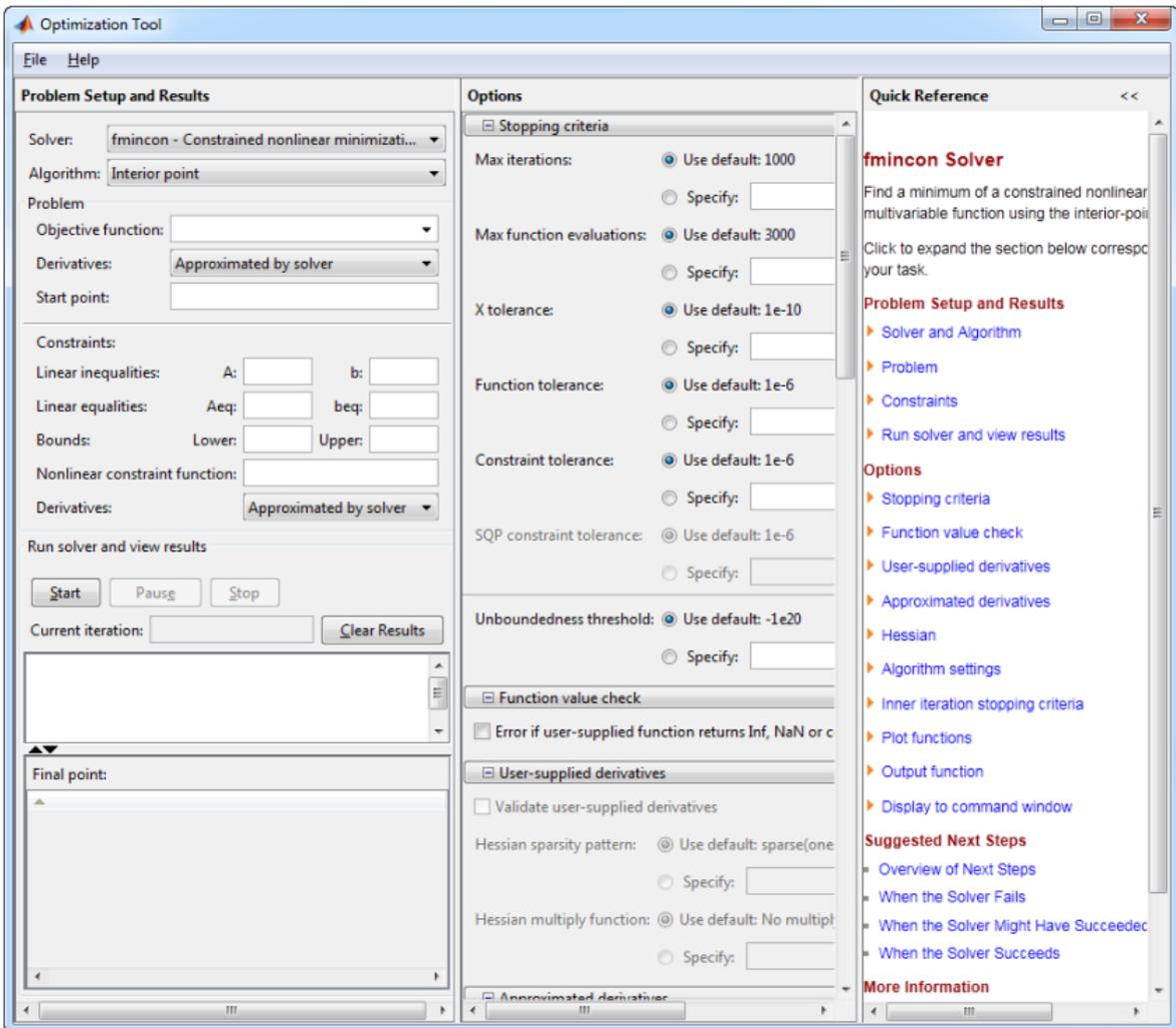


Figura 2.16 – Interface da *toolbox* de otimização global do MATLAB.

2.2.4 - Solver Add-in do Excel

O *Solver* é um suplemento (*add-in*) do Microsoft Excel para a otimização e a análise de hipóteses ou modelos. Ele busca um valor ótimo (máximo ou mínimo) para uma fórmula em uma célula, chamada de célula objetivo, sujeita a restrições ou limites nos valores de outras células / fórmulas, em uma planilha. O *Solver* funciona com um grupo de células, chamado de variáveis de decisão, ou simplesmente células variáveis, usadas no cálculo das fórmulas nas células objetivo e de restrição. O Solver ajusta os valores nas células variáveis para satisfazer os limites das células de restrição e produzir o resultado desejado para a célula objetivo. Simplificando, o *Solver* é útil para determinar o valor máximo ou mínimo de uma célula alterando outras células.

As células variáveis, de restrição e decisão, e as fórmulas que as interrelacionam formam o modelo para o *Solver*; os valores finais encontrados pelo *Solver* são uma solução para esse modelo. O *Solver* usa três métodos, desde programação linear à otimização não linear para a solução dos modelos. Em programação linear, o método disponível é o *Simplex*. Já para a otimização não linear, há dois métodos. O primeiro é o GRG (*Generalized Reduced Gradient*), que apesar de ser um método determinístico e dependente de gradiente, o *Solver* apresenta uma versão melhorada dele, agregando *multistart*; o que o torna um método populacional, capaz de escapar de mínimos locais, portanto, adequado na busca de máximo ou mínimo global. O segundo método é o GA / evolucionário, que por natureza é um método à base de população e muito adequado na obtenção de ótimos globais.

Uma limitação da versão gratuita do *Solver* é pertinente ao número de células variáveis e de restrições, que é de 200 e 100, respectivamente, que um modelo pode ter. Esse limite pode ser aumentado na versão paga. Há pacotes que até retiram esses limites. Vale ressaltar que todas as funcionalidades disponíveis no *Solver* podem ser automatizadas via programação VBA, que já vem embarcada no Excel. A Figura 2.17 mostra a interface do *Solver, add-in* do Excel.

2.2.5 - Linguagem de Programação e Framework de Otimização Adotadas

A linguagem de programação adotada nesta pesquisa é o Python, dada a riqueza de *packages* (termo usado no dialeto do Python para *frameworks*/bibliotecas) *open source*, de alto nível, disponíveis e crescente a cada dia. Além disso, a sintaxe do Python é muito simples e prática, o que poupa tempo e permite focar na modelagem e solução do problema estudado e não nas questões de programação computacional. Outro fator, é a versatilidade de integração com outras linguagens de programação como o matlab, C (que é a linguagem mãe do Python), C++, etc.

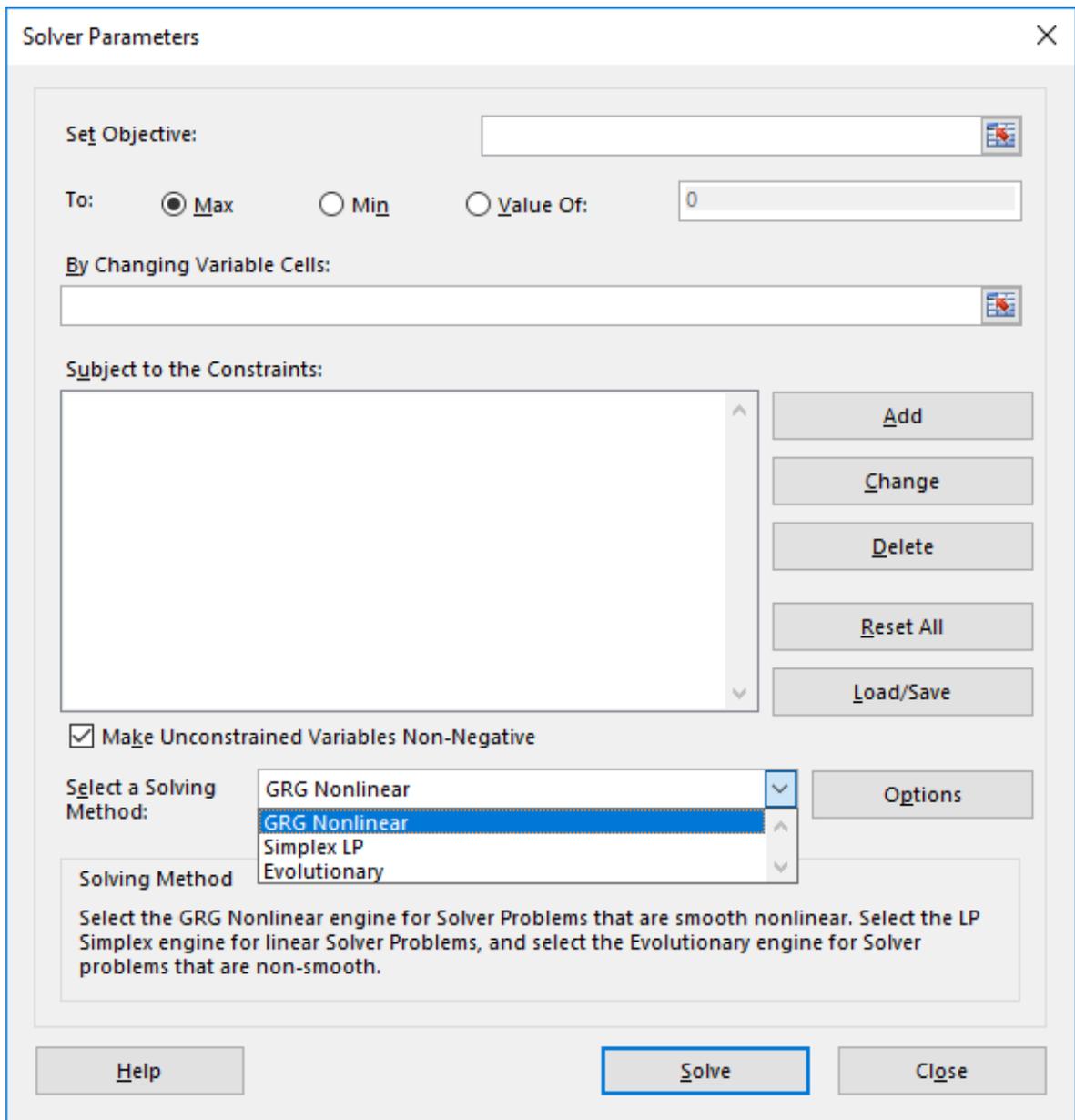


Figura 2.17 – Interface do *Solver, add-in* do Excel.

Quanto à ferramenta de otimização, o Platypus será o utilizado. De acordo com seu desenvolvedor, D. Hadka, Platypus é um *framework* para computação evolucionária, em Python, com foco em algoritmos meta-heurísticos multiobjetivo (MMAs) e que difere de bibliotecas de otimização existentes, incluindo PyGMO, Inspyred e Scipy, porque fornece algoritmos de otimização e ferramentas de análise para otimização multiobjetivo. Ele correntemente disponibiliza os algoritmos: NSGA-II, NSGA-III, CMAES, MOEA/D, IBEA, Epsilon-MOEA, SPEA2, GDE3, OMOPSO e SMPSO, já consagrados na comunidade científica. A seguir está uma descrição sucinta de cada algoritmo e a respectiva referência para mais detalhes:

- **NSGA-II:** é um algoritmo evolucionário multiobjetivo. Um dos primeiros e mais largamente utilizados algoritmos genéticos. Ele aprimora seu predecessor, NSGA, introduzindo uma classificação não dominada rápida e usa uma métrica de distância populacional mais eficiente computacionalmente durante a seleção dos indivíduos sobreviventes, (Deb et al., 2000);
- **NSGA-III:** é o sucessor multiobjetivo do NSGA-II e usa pontos de referência para direcionar as soluções rumo a um conjunto diferente. O número desses pontos é controlado pelos parâmetros: número de objetivos e divisões. Seus autores também propõem uma abordagem de duas camadas para as divisões em problemas multiobjetivo, onde números de divisão externa e interna são especificados, (Deb e Jain, 2014);
- **CMA-ES:** é um algoritmo de estratégia de evolução e adaptação com matriz de covariância sofisticado. Ele produz proles amostrando uma distribuição formada por uma matriz de covariância, por isso o nome; e atualiza esta matriz a partir das proles sobreviventes. Existem variantes mono e multiobjetivo na literatura, (Hansen e Kern, 2004) e (Igel et al., 2007);
- **MOEA/D:** é um algoritmo relativamente novo, baseado no conceito de decompor o problema em várias formulações mono-objetivo. Existem várias versões dele na literatura, mas, a variante mais comum vista é proposta por Li e Zhang (2009);
- **IBEA:** é um algoritmo evolucionário multiobjetivo à base de indicador, que usa o indicador de performance hipervolume como um meio de ranquear as soluções. Algoritmos à base de indicador surgem da ideia de que um indicador de performance, tal como hipervolume ou ϵ_+ , etc., destaca soluções com qualidades desejáveis. A desvantagem principal dessa metodologia é que o cálculo do indicador de performance pode se tornar custoso computacionalmente, particularmente quando o número de objetivos cresce, (Zitzler e Künzli, 2004);

- **Epsilon-MOEA:** é um algoritmo evolucionário multiobjetivo de estado estacionário que usa arquivo com dominância ϵ para registrar um conjunto diversificado de soluções ótimas de Pareto. O termo estado estacionário significa que o algoritmo desenvolve um solução de cada vez. Isto, contrasta com algoritmos geracionais, que desenvolve uma população inteira a cada iteração. Os arquivos de dominância ϵ são úteis uma vez que garantem convergência e diversidade ao longo da busca. Entretanto, o algoritmo requer um parâmetro ϵ adicional que é dependente do problema. O parâmetro ϵ controla a granularidade ou resolução das soluções no espaço dos objetivos. Valores menores produzem conjuntos maiores e mais densos, enquanto valores maiores produzem conjuntos menores. Em geral, os valores de ϵ deveriam ser escolhidos de forma a gerar um tamanho moderado para o conjunto de aproximação de Pareto, (Deb et al., 2003);
- **SPEA2:** é um antigo mas popular algoritmo evolucionário multiobjetivo de *benchmark*, que usa o método chamado *strength-based* para ranquear as soluções. A ideia geral é que a força ou qualidade de uma solução está relacionada à força das soluções que ela domina, (Zitzler et al., 2002a);
- **GDE3:** é a terceira versão do algoritmo de evolução diferencial generalizada. O nome evolução diferencial vem de como o algoritmo desenvolve a prole. Ele randomicamente seleciona três pais. Em seguida, ele calcula a diferença (o diferencial) entre dois dos pais. Finalmente, ele compensa o pai remanescente por este diferencial, (Kukkonen e Lampinen, 2005);
- **OMOPSO:** é um algoritmo multiobjetivo de enxame de partículas que inclui um arquivo de dominância ϵ , para descobrir um conjunto diferente das soluções ótimas de Pareto. Esta implementação do OMOPSO difere ligeiramente da implementação original do autor, na *framework* jMetal, devido a uma discrepância entre os código do autor e o *paper*. O *paper* retorna o arquivo de dominância ϵ , enquanto que o código retorna os líderes. Esta discrepância causa uma pequena diferença na performance, (Sierra e Coello Coello, 2005);
- **SMPSO:** é um algoritmo multiobjetivo de enxame de partículas, (Nebro et al., 2009);

O Platypus foi escolhido, porque além dessas características, e das já apresentadas na Seção 2.2.1, é de fácil integração com a modelagem do problema a ser otimizado e conta com um *experimenter*, o que torna fácil a otimização de vários problemas ao mesmo tempo, utilizando simultaneamente todos os algoritmos disponíveis, facilitando a avaliação da adequação de cada algoritmo a cada problema de otimização. O *experimenter* utiliza processamento paralelo, o que é fundamental no tratamento de problemas exigentes

computacionalmente, e com alto grau de complexidade dos algoritmos, como os abordados aqui.

O Platypus é a versão moderna, e em Python, da *MOEA Framework*. Esta última, também foi desenvolvida por Hadka (2016), mas, em linguagem Java. Segundo o autor, o Platypus foi criado para acompanhar a preferência recente pelo Python e tirar proveito das vantagens e bibliotecas que esta linguagem oferece. Um diferencial que o Platypus apresenta (seguindo sua precursora: MOEA) é que o desenvolvedor adicionalmente aos seus algoritmos utiliza nela uma série de *frameworks* e algoritmos robustos, *open source*, disponíveis, como o jMetal (acrônimo de *Metaheuristic Algorithms in Java* e, é uma *framework* para otimização multiobjetivo com meta-heurísticas. Esta é a precursora da *framework* jMetalPy), entre outras. Isso deu robustez à MOEA, e por consequência, ao Platypus.

3 - MÉTODOS SEM MALHA

Nesse capítulo serão apresentados os métodos sem malha mais conhecidos e os recentes avanços no método. Alguns termos e conceitos fundamentais para a formulação do método são apresentados.

3.1 - DEFINIÇÃO

Os métodos sem malha, diferentemente do MEF, usam um conjunto de nós espalhados no domínio e no contorno do problema (chamados de nós de campo ou *field nodes*), de forma que estes não contêm nenhuma informação sobre a conectividade entre eles para a aproximação ou interpolação das incógnitas do campo, ou seja, não formam uma malha. A Figura 3.1 mostra a principal diferença entre esses dois métodos. Essa distribuição nodal não necessariamente precisa ser uniforme e pode ser controlada, assim como a densidade, ou quantidade total de nós (Atluri e Zhu, 1998).

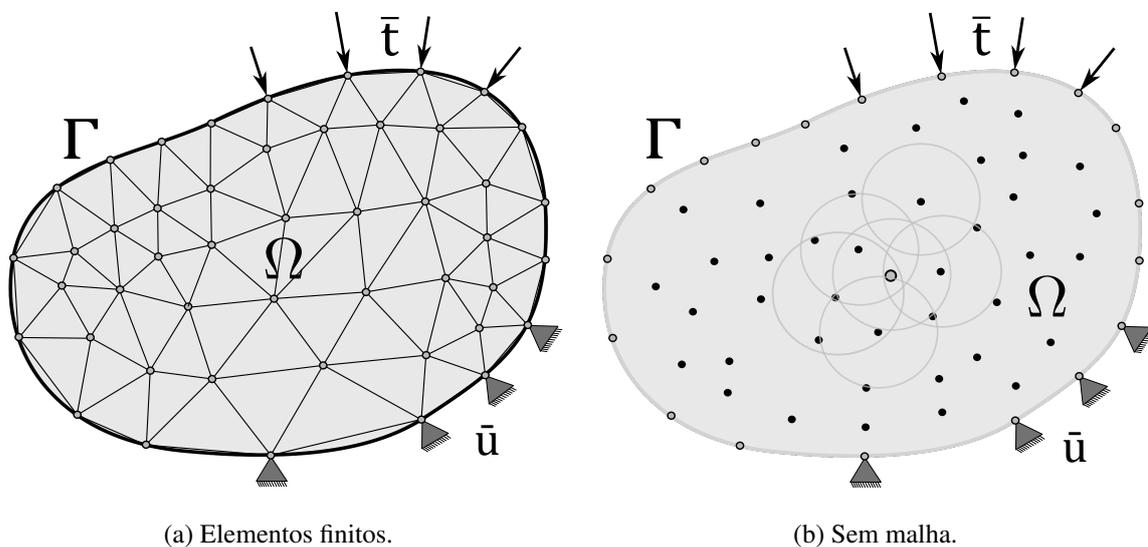


Figura 3.1 – Representação do domínio nos métodos sem malha e no Método dos Elementos Finitos (MEF).

Ele surgiu como uma alternativa ao MEF tradicional, buscando eliminar os problemas relacionados à malha, como o alto custo para a geração da mesma, a baixa precisão na recuperação das tensões e a dificuldade de resolução de problemas que exijam análise adaptativa, como fratura e grandes deformações.

Muitos métodos sem malha obtiveram bons resultados e possuem um grande potencial para se tornarem poderosas ferramentas numéricas. Mas, em compensação, esses métodos estão em fase de desenvolvimento e possuem muitos problemas técnicos que ainda precisam ser contornados.

Esses métodos podem ser classificados de acordo com a formulação ou método de aproximação utilizado, bem como o tipo de representação do domínio. Os principais métodos e suas classificações estão resumidos esquematicamente na Tabela 3.1.

Tabela 3.1 – Métodos sem malha classificados. Adaptado de Liu e Gu (2005).

Classificação	Categoria	Exemplos de métodos
Baseado nas formulações utilizadas	Forma forte	Colocação, FPM etc.
	Forma fraca	EFG, RPIM, MLPG, LRPIM, etc.
	Combinação dessas duas formas	MWS, etc.
Baseado no método de interpolação/aproximação	Mínimos quadrados móveis (MQM)	EFG, MLPG, etc.
	Método da representação da integral	SPH, etc.
	PIM	RPIM, LRPIM, etc.
	Outras interpolações	PUFEM, hp-cloud, etc.
Baseado na representação do domínio	Domínio	SPH, EFG, RPIM, MLPG, LRPIM, etc.
	Contorno	BNM, LBIE, BPIM, BRPIM, HBRPIM, etc.

Alguns métodos recentes foram deixados de lado por ainda estarem em sua fase inicial de estudos e por não possuírem uma base sólida.

3.2 - FUNÇÕES DE APROXIMAÇÃO

As funções de forma e as funções peso ou ponderadoras desempenham um papel essencial na performance dos métodos sem malha. O campo de deslocamentos u em qualquer ponto x dentro do domínio é interpolado inteiramente em função dos termos nodais, dentro do suporte compacto da função ponderadora desse ponto x , como

$$u(x) = \sum_{i=1}^n \phi_i(x) u_i = \mathbf{\Phi}(\mathbf{x}) \hat{\mathbf{u}}, \quad (3.1)$$

onde n é o número de nós dentro do suporte compacto do domínio do ponto x ; u_i é a variável do campo nodal no nó i ; $\hat{\mathbf{u}}$ é o vetor que contém as variáveis de campo desses n nós e $\phi_i(x)$ é a função de forma do nó i . O suporte compacto, também conhecido como domínio de influência nodal, garante a localidade da aproximação e será definido posteriormente neste capítulo.

3.3 - FUNÇÕES DE FORMA

As funções de forma utilizadas nos métodos sem malha devem satisfazer alguns requisitos básicos, sendo eles:

- Nós arbitrariamente distribuídos, sendo suficientemente robusto para permitir uma distribuição razoavelmente arbitrária;
- Numericamente estável;
- Deve satisfazer uma certa ordem de consistência;
- Deve ter suporte compacto, devendo ser considerado zero quando fora do mesmo;
- As funções de aproximação ligadas às funções de forma devem ser compatíveis ao longo do domínio do problema quando a forma fraca global é utilizada, ou deve ser compatível no domínio local da quadratura na forma fraca local;
- É ideal que a função de forma possua propriedades do delta de Kronecker;
- Deve ser computacionalmente eficiente.

O desenvolvimento de funções de forma especiais para serem utilizadas nos métodos sem malha é uma das áreas de pesquisa mais procuradas nesses últimos anos e, sendo assim, várias aproximações foram propostas. Liu et al. (2002b) classificam essas formulações em três grandes categorias, baseados nos tipos de teoria usadas na aproximação. As principais funções de forma e suas classificações estão resumidos esquematicamente na Tabela 3.2.

Tabela 3.2 – Principais funções de forma e suas classificações.

Categoria	Técnicas de aproximação
Representação por integral	<i>Smoothed Particle Hydrodynamics</i> (SPH) <i>Reproducing Kernel Particle Method</i> (RKPM)
Representação por séries	Mínimos Quadrados Móveis (MQM) <i>Point Interpolation Method</i> (PIM, RPIM) <i>Partition of Unity</i> (PoU)
Representação por diferencial	<i>Generalized Finite Difference Method</i> (GFDM)

No método de representação por integral, um dos mais antigos métodos utilizados na discretização sem malha, a sua função é representada usando suas informações no domínio local por meio de uma operação com integrais ponderadas. A consistência é conseguida escolhendo-se uma função peso adequada.

Os métodos de representação em série possuem um longo histórico. Eles foram bem desenvolvidos no MEF e agora estão sendo utilizados nos métodos sem malha, sendo que seus dois métodos mais conhecidos são dos MQM ou MLS e o PIM usando funções bases radiais (RPIM). Nessa pesquisa é empregado o MQM.

O método de representação por diferencial vem sendo utilizado por um bom tempo no método das diferenças finitas. Essa aproximação não é compatível globalmente e sua consistência é garantida pela teoria da série de Taylor. Esse método é geralmente utilizado para estabelecer sistemas de equações baseados na forma forte (Zienkiewicz et al., 2000).

3.4 - FUNÇÕES PONDERADORAS

A continuidade das funções de forma $\Phi(\mathbf{x})$ são governadas pela continuidade da base polinomial escolhida, assim como a continuidade das funções ponderadoras utilizadas. A escolha das funções ponderadoras é na maioria das situações arbitrária, desde que a localidade da mesma seja garantida; ou seja, que seu valor decresça monotonicamente à medida que se afasta do ponto de interesse x , e seja nula quando fora do suporte compacto ou domínio de influência.

As funções exponenciais e do tipo *spline* são as mais utilizadas na prática, mas muitos autores optam por construir suas próprias funções ponderadoras, com a ordem de continuidade desejada, dependendo do problema em questão. Devido a arbitrariedade dessas funções elas podem ser facilmente construídas, desde que a propriedade de partição da unidade seja garantida e a primeira e segunda derivada sejam equivalentes a zero no contorno do suporte compacto. Geralmente as funções exponenciais são computacionalmente mais custosas, mas são menos sensíveis ao tamanho do suporte compacto.

3.5 - SUPORTE COMPACTO

O suporte compacto é uma região de influência do nó x_i onde a função ponderadora é diferente de zero. A precisão da interpolação para um ponto de interesse depende dos nós no interior do suporte compacto. Sendo assim, um suporte compacto apropriado tem que ser escolhido para garantir uma aproximação eficiente e precisa, geralmente controlado por coeficientes adimensionais, que aumentam o raio de cobertura do suporte c_i , controlando o tamanho desse domínio.

O suporte compacto tem geometria arbitrária, dessa forma por simplicidade, geralmente são escolhidos suportes circulares ou retangulares, como demonstrado na Figura 3.2.

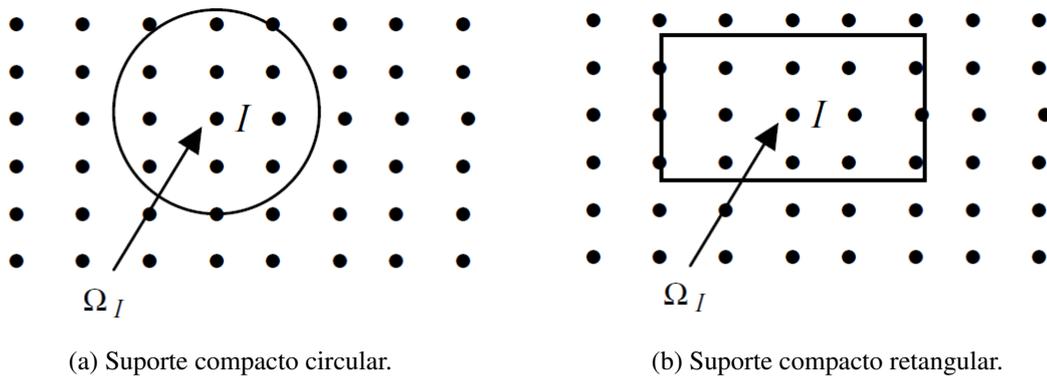


Figura 3.2 – Suporte compacto do ponto de interesse em I em diferentes modelos de métodos sem malha. Adaptado de Chen et al. (2006).

Esse número é totalmente experimental, deve ser testado e aperfeiçoado, já que, até o atual momento, não existe uma fórmula particular para calcular este valor. Geralmente valores de $\alpha_s = [2.0, 3.0]$ resultam em bons resultados para a maioria dos problemas, como visto em Liu e Gu (2005) e Atluri e Zhu (2000).

3.6 - DOMÍNIO DE DEFINIÇÃO

O domínio de definição (local) de cada nó é a região local, centrada no nó, onde está definido a integral dos resíduos ponderados (Figura 3.5). Por outro lado, o domínio de definição de um ponto qualquer x é um subdomínio no qual a aproximação é definida. Esse domínio cobre todos os nós cujos respectivos suportes compactos incluem o ponto qualquer x . Assim como o suporte compacto, a precisão da interpolação depende da quantidade de nós cujo suporte compacto inclua o ponto de interesse. Sendo assim, o tamanho do suporte compacto influencia diretamente na quantidade de nós que fará parte desta interpolação.

3.7 - IMPOSIÇÃO DAS CONDIÇÕES DE CONTORNO ESSENCIAIS

As aproximações utilizadas pelos métodos sem malha resultam em valores que não passam por parâmetros nodais. Sendo assim, uma imposição das condições de contorno essenciais é necessária para se obter a solução do problema, aumentando a dificuldade de implementação e o esforço computacional. Para tal propósito os multiplicadores Lagrangeanos são os mais utilizados, mas são também os mais custosos computacionalmente, devido a não positividade das equações discretas (Chen et al., 2006).

3.8 - AVALIAÇÃO DAS INTEGRAIS

Uma das grandes diferenças dentre os diversos métodos sem malha baseados na forma fraca é a forma como as integrais são avaliadas.

Nos métodos sem malha baseados na forma fraca global geralmente são utilizados *background cells* ou malhas de fundo, onde a quadratura de Gauss é realizada dentro dessas células ou elementos. Essa malha de fundo pode ser um arranjo regular dentro do domínio ou pode ser uma discretização do domínio, da mesma forma que no MEF, como mostra a Figura 3.3.

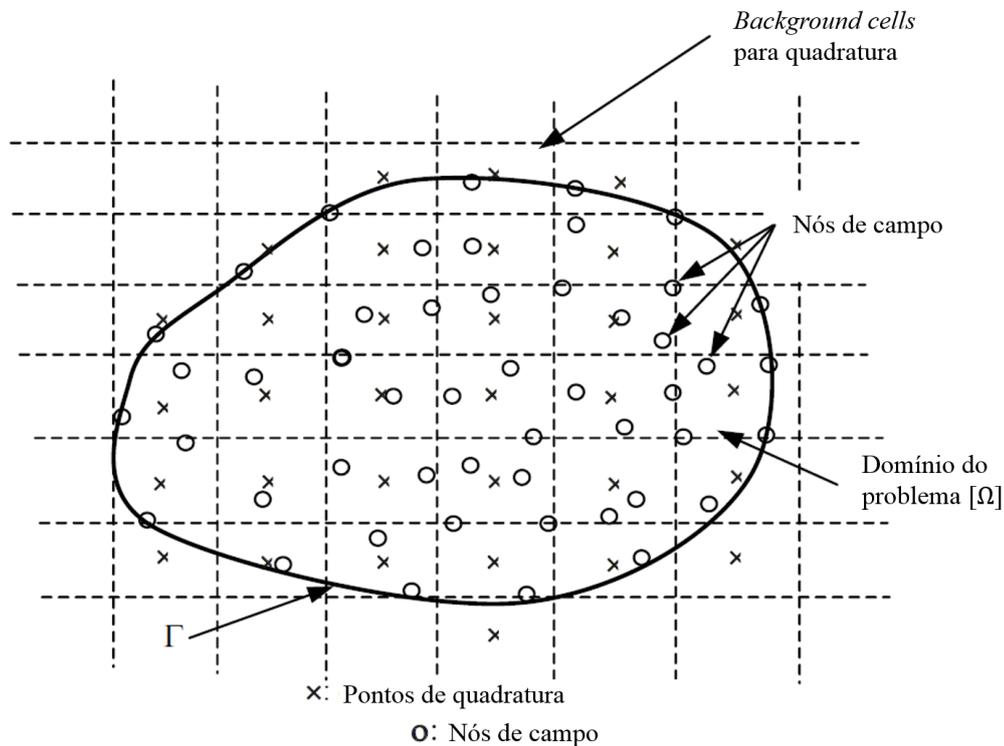


Figura 3.3 – Uso de *background cells* em métodos sem malha baseados na forma fraca global. Adaptado de Liu e Gu (2005).

Em contrapartida, nos métodos sem malha baseados na forma fraca local a integração numérica é realizada em um domínio de quadratura local predefinida para o nó, também chamado de domínio local.

Domínio local ou domínio da forma fraca de um nó x_i é o subdomínio onde a forma fraca local é definida e onde a quadratura de Gauss será realizada, como mostra a Figura 3.4, para um nó qualquer x . Geralmente o tamanho do domínio local é controlado por coeficientes adimensionais, que reduzem o raio de cobertura do suporte compacto. Esse número, assim como o tamanho do suporte compacto, é totalmente experimental, deve ser testado e aperfeiçoado. Geralmente valores de $\alpha_q = [0.4, 0.6]$ conduzem a bons resultados para a

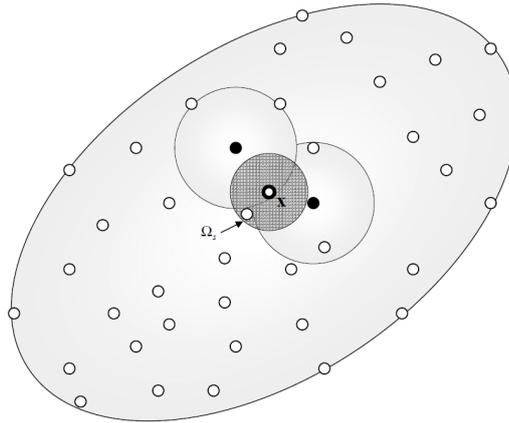


Figura 3.4 – Domínio local Ω_S de um nó x . Adaptado de Atluri et al. (2004).

maioria dos problemas, como visto em Liu e Gu (2005) e Atluri e Zhu (2000).

3.9 - MÉTODOS SEM MALHA BASEADOS NA FORMA FRACA GLOBAL

Os métodos sem malha baseado na forma fraca global são geralmente baseados na forma fraca de Galerkin e definida sobre o domínio global de um problema, usando funções de forma localmente definidas. Multiplicadores Lagrangeanos são utilizados para impor as condições de contorno essenciais e *background cells*, ou malha de fundo, são utilizadas no processo de quadratura para avaliar as integrais.

O primeiro método sem malha baseado na forma fraca global foi o *Diffuse Element Method* (DEM) proposto por Nayroles et al. (1992). Nesse método o MQM proposto por Lancaster e Salkauskas (1981) foi criado para gerar as funções de forma e a forma fraca de Galerkin, empregada para construir o sistema de equações.

Em 1994, Belytschko et al. (1994b) propuseram o método *Element-free Galerkin* (EFG), onde o mesmo MQM foi utilizado para construir funções de forma baseadas em apenas um grupo de nós distribuídos arbitrariamente em um domínio local. Esse método utiliza a mesma forma fraca de Galerkin, mas precisa de um conjunto de *background cells* para avaliar essas integrais.

O EFG se mostrou muito mais preciso que o MEF, com um bom fator de convergência, tendo sido aplicado a uma gama de problemas, como fratura e problemas não lineares. Todas essas aplicações tornaram o EFG um método bem atrativo e sólido, mas não isento de falhas. O método é computacionalmente mais custoso do que o MEF, além de precisar de *background cells* para se calcular o sistema de matrizes, não sendo, portanto, verdadeiramente sem malha.

Liu e Gu (1999) propuseram o *Point Interpolation Method* (PIM) baseado na forma fraca de Galerkin, onde o domínio era representado por nós propriamente distribuídos. O PIM também é usado para construir as funções de forma polinomiais, baseado em um grupo de nós arbitrariamente distribuídos no domínio local. Posteriormente, Liu e Gu (2001b) desenvolveram o RPIM, que utiliza uma base radial, sendo bem mais estável e robusto, mas que, assim como o EFG, precisa de *background cells*, não sendo assim verdadeiramente sem malha.

3.10 - MÉTODOS SEM MALHA BASEADOS NA FORMA FRACA LOCAL

Para evitar o uso de *background cells*, a forma fraca global é usada por Atluri e Zhu (1998) para desenvolver o *Meshless Local Petrov-Galerkin* (MLPG), que também foi desenvolvido com outras variações (Atluri e Zhu, 2000).

Quando a forma fraca local é usada em um campo nodal, a integração numérica é realizada em um domínio de quadratura local predefinida para o nó, que também pode ser o domínio local onde as funções ponderadas são definidas. Desta forma, estes métodos não necessitam de *background cells*, sendo assim, verdadeiramente sem malha.

O procedimento desse método é similar aos métodos numéricos baseados na formulação na forma forte, como o FDM. Entretanto, por causa da aproximação MQM empregada, tratamentos especiais são necessários para aplicar as condições de contorno essenciais.

Outro método bastante conhecido é o *Local Point Interpolation Method* (LPIM), desenvolvido por Liu e Gu (2001a), e o *Local Radial Point Interpolation Method* (LRPIM), desenvolvido por Liu et al. (2002a). Nesses métodos, funções de forma polinomiais PIM são utilizadas, o que é responsável pela singularidade da matriz de interpolação do momento, sendo assim, um algoritmo de triangulação é necessário para evitar o problema. Já as funções de forma RPIM contornam esse problema, tornando-se uma alternativa robusta, notadamente, quando domínios definidos com nós aleatoriamente posicionados são utilizados no problema.

3.11 - FORMULAÇÃO DO MÉTODO SEM MALHA LOCAL

A formulação proposta na presente pesquisa tem fortes influências da equação dos resíduos ponderados e do teorema do trabalho, que estabelece uma relação energética entre campos de tensão estaticamente admissíveis e campos de deformação cinematicamente admissíveis.

3.11.1 - Método dos Mínimos Quadrados Móveis

A aproximação pelo método dos Mínimos Quadrados Móveis (MQM) ou *Moving Least Square (MLS)* foi concebida por Lancaster e Salkauskas (1981) para a reconstrução de superfícies e ajuste de dados pontuais arbitrariamente dispersos. Nayroles et al. (1992) introduziram o uso dos MQM para a construção de funções de forma, para a solução de problemas de mecânica dos sólidos deformáveis.

A aproximação pelos MQM é composta por três componentes: uma função peso do suporte compacto associada a cada nó da discretização, uma base polinomial completa e um conjunto de coeficientes em função das coordenadas espaciais, conforme apresentado por Atluri e Zhu (1998). Todas as terminologias utilizadas na presente pesquisa seguem o modelo proposto por Atluri e Zhu (2000).

Considere Ω o domínio de um corpo com contorno Γ e seja $N = \{x_1, x_2, \dots, x_N\} \in \Omega$ o conjunto de nós espalhados ao longo desse domínio, que representam uma discretização sem malha, no qual alguns deles estão localizados no contorno Γ , como pode ser visto na Figura 3.5. Os suportes locais circulares ou retangulares, centralizados em cada ponto nodal, podem ser utilizados. Nos arredores de um ponto x qualquer, o domínio de definição da aproximação pelos MQM é o subdomínio $\Omega_x \in \Omega$, com um contorno local Γ_x .

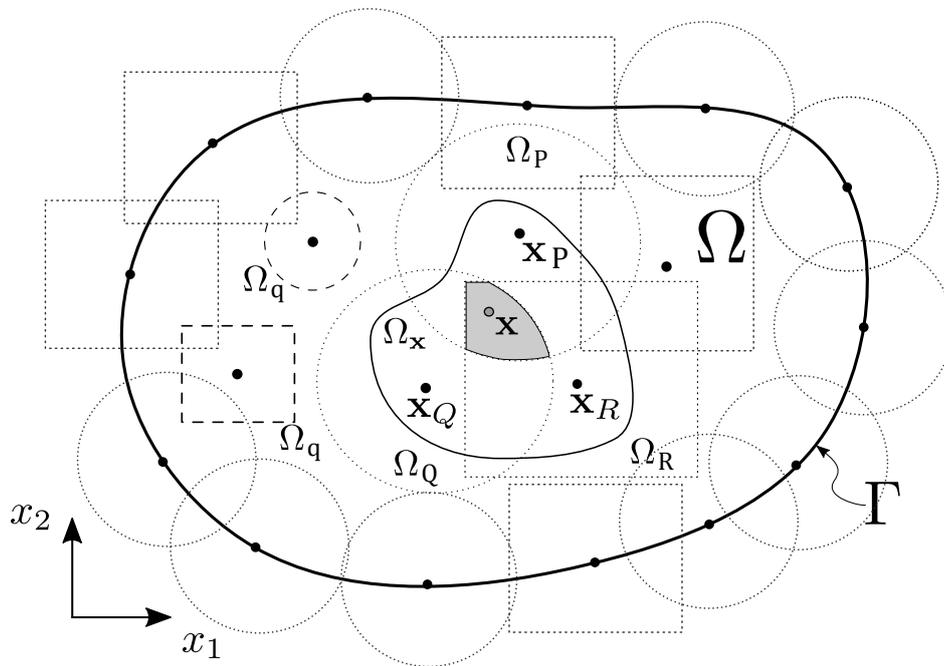


Figura 3.5 – Representação de uma discretização do domínio global Ω usando o método sem malha, com um contorno $\Gamma = \Gamma_u + \Gamma_t$, com uma distribuição nodal \mathbf{x}_i . Ω_s , representado como Ω_P , Ω_Q e Ω_R , é o suporte compacto do nó; Ω_x é o domínio de definição de um ponto qualquer x e Ω_q é o domínio da forma fraca local ou domínio de quadratura do nó \mathbf{x}_i .

3.11.1.1 - Funções de forma

Considere Ω_x o domínio de definição de uma aproximação pelos MQM em uma região nos arredores de um ponto \mathbf{x} qualquer, demonstrado para uma aproximação unidimensional na Figura 3.6.

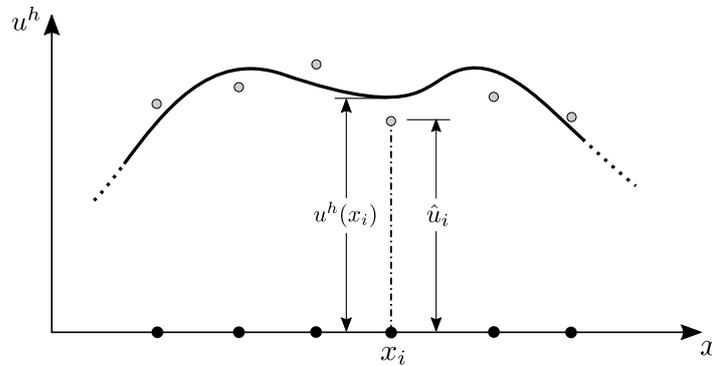


Figura 3.6 – Exemplo unidimensional da aproximação pelos MQM, onde $u^h(x_i) \neq \hat{u}_i$.

Para aproximar o deslocamento $u(\mathbf{x}) \in \Omega_x$, a partir de um certo número de nós espalhados ao longo do domínio \mathbf{x}_i , $i = 1, 2, \dots, n$, onde esses parâmetros nodais \hat{u}_i são definidos, a aproximação pelos MQM é dada por

$$u^h(\mathbf{x}) = \mathbf{p}^T(\mathbf{x})\mathbf{a}(\mathbf{x}) \quad (3.2)$$

para $\mathbf{x} \in \Omega_x$, em que

$$\mathbf{p}^T(\mathbf{x}) = [p_1(\mathbf{x}), p_2(\mathbf{x}), \dots, p_m(\mathbf{x})], \quad (3.3)$$

é o vetor da base polinomial de ordem m e $\mathbf{a}(\mathbf{x})$ é o vetor de coeficientes $a_j(\mathbf{x})$, $j = 1, 2, \dots, m$, que são funções das coordenadas espaciais $\mathbf{x} = [x_1, x_2]^T$, em problemas 2-D.

O vetor de coeficientes $\mathbf{a}(\mathbf{x})$ é determinado a partir da minimização da norma ponderada discreta L_2

$$J(\mathbf{x}) = \frac{1}{2} \sum_{i=1}^n w_i(\mathbf{x}) [u^h(\mathbf{x}_i) - \hat{u}_i]^2 = \frac{1}{2} \sum_{i=1}^n w_i(\mathbf{x}) [\mathbf{p}^T(\mathbf{x}_i)\mathbf{a}(\mathbf{x}) - \hat{u}_i]^2 \quad (3.4)$$

com respeito a cada termo de $\mathbf{a}(\mathbf{x})$, onde $w_i(\mathbf{x})$ é a função peso associada ao nó i , com suporte compacto $w_i(\mathbf{x}) > 0$, para todo \mathbf{x} no suporte de $w_i(\mathbf{x})$. A Figura 3.5 demonstra esquematicamente o suporte compacto das funções peso dos MQM associadas a alguns nós, que nada mais são do que os domínios de influência de cada um desses nós. Encontrando o

extremo de $J(\mathbf{x})$ com respeito a cada termo de $\mathbf{a}(\mathbf{x})$ obtemos

$$\mathbf{A}(\mathbf{x})\mathbf{a}(\mathbf{x}) = \mathbf{B}(\mathbf{x})\hat{\mathbf{u}}, \quad (3.5)$$

no qual

$$\mathbf{A}(\mathbf{x}) = \sum_{i=1}^n w_i(\mathbf{x})\mathbf{p}(\mathbf{x}_i)\mathbf{p}^T(\mathbf{x}_i), \quad (3.6)$$

$$\mathbf{B}(\mathbf{x}) = [w_1(\mathbf{x})\mathbf{p}(\mathbf{x}_1), w_2(\mathbf{x})\mathbf{p}(\mathbf{x}_2), \dots, w_n(\mathbf{x})\mathbf{p}(\mathbf{x}_n)] \quad (3.7)$$

e

$$\hat{\mathbf{u}} = [\hat{u}_1, \hat{u}_2, \dots, \hat{u}_n]. \quad (3.8)$$

Resolvendo a equação (3.5) para $\mathbf{a}(\mathbf{x})$, tem-se

$$\mathbf{a}(\mathbf{x}) = \mathbf{A}^{-1}(\mathbf{x})\mathbf{B}(\mathbf{x})\hat{\mathbf{u}}, \quad (3.9)$$

desde que $n \geq m$, para cada ponto qualquer \mathbf{x} , como uma condição necessária para uma aproximação bem definida. Substituindo $\mathbf{a}(\mathbf{x})$, obtido na equação (3.9), na equação (3.2) obtém-se a equação final da aproximação pelos MQM

$$u^h(\mathbf{x}) = \sum_{i=1}^n \phi_i(\mathbf{x})\hat{u}_i, \quad (3.10)$$

onde

$$\phi_i(\mathbf{x}) = \sum_{j=1}^m p_j(\mathbf{x}) [\mathbf{A}^{-1}(\mathbf{x})\mathbf{B}(\mathbf{x})]_{ji} \quad (3.11)$$

é a função de forma da aproximação pelos MQM correspondente ao nó \mathbf{x}_i , esquematicamente representado na Figura 3.7b. É importante notar que as funções de forma dos MQM não são interpolantes nodais, não apresentando assim a propriedade do delta de *Kronecker*, ou seja, $\phi_i(\mathbf{x}_j) \neq \delta_{ij}$.

Como $\phi_i(\mathbf{x})$ desaparece para qualquer \mathbf{x} que não esteja no domínio local do nó \mathbf{x}_i , como pode ser visto na Figura 3.7b, o caráter local da aproximação pelos MQM é preservada.

As derivadas espaciais da função de forma $\phi_i(\mathbf{x})$ são obtidas com

$$\phi_{i,k} = \sum_{j=1}^m [p_{j,k}(\mathbf{A}^{-1}\mathbf{B})_{ji} + p_j(\mathbf{A}^{-1}\mathbf{B}_{,k} - \mathbf{A}^{-1}\mathbf{A}_{,k}\mathbf{A}^{-1}\mathbf{B})_{ji}], \quad (3.12)$$

onde $(\cdot)_{,k} = \partial(\cdot)/\partial x_k$.

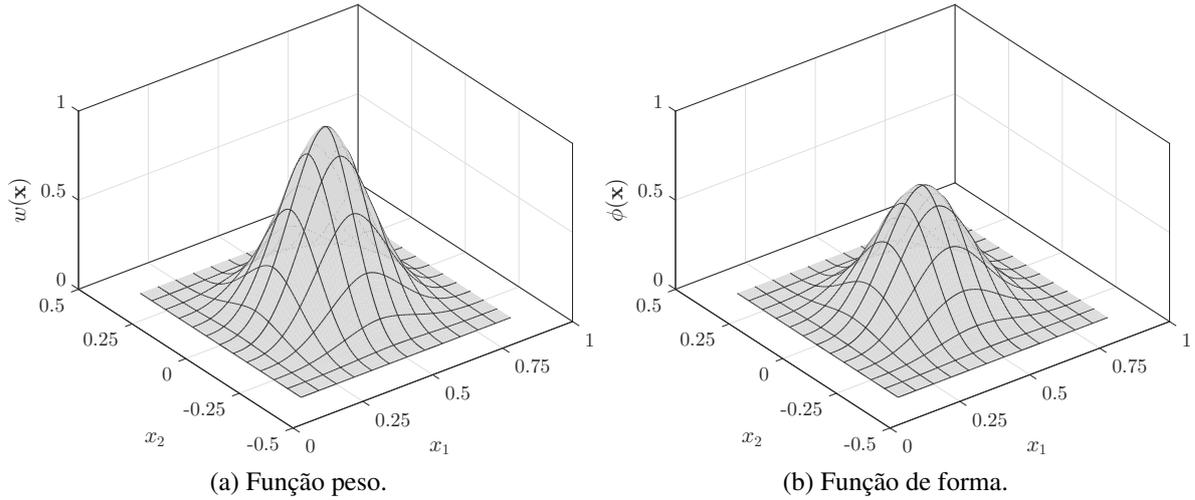


Figura 3.7 – Típica função peso e função de forma de uma aproximação pelos MQM para um nó $x = [1/2 \ 0]^T$

3.11.1.2 - Funções de ponderação

As funções peso $w_i(\mathbf{x})$, ilustrada na Figura 3.7a, introduzidas na equação (3.4) para cada nó \mathbf{x}_i , tem suporte compacto que define o subdomínio, onde $w_i(\mathbf{x}) > 0$ para todos os \mathbf{x} . Por simplicidade, um suporte compacto retangular ou circular serão considerados nas análises desta pesquisa, com suas funções peso definidas por

$$w_i(\mathbf{x}) = w_{i_x}(\mathbf{x}) w_{i_y}(\mathbf{x}) \quad (3.13)$$

com funções ponderadoras *spline* de quarta ordem dadas por

$$w_{i_x}(\mathbf{x}) = \begin{cases} 1 - 6 \left(\frac{d_{i_x}}{r_{i_x}} \right)^2 + 8 \left(\frac{d_{i_x}}{r_{i_x}} \right)^3 - 3 \left(\frac{d_{i_x}}{r_{i_x}} \right)^4 & \text{para } 0 \leq d_{i_x} \leq r_{i_x} \\ 0 & \text{para } d_{i_x} > r_{i_x} \end{cases} \quad (3.14)$$

e

$$w_{i_y}(\mathbf{x}) = \begin{cases} 1 - 6 \left(\frac{d_{i_y}}{r_{i_y}} \right)^2 + 8 \left(\frac{d_{i_y}}{r_{i_y}} \right)^3 - 3 \left(\frac{d_{i_y}}{r_{i_y}} \right)^4 & \text{para } 0 \leq d_{i_y} \leq r_{i_y} \\ 0 & \text{para } d_{i_y} > r_{i_y}, \end{cases} \quad (3.15)$$

onde $d_{i_x} = |\mathbf{x} - \mathbf{x}_i|$ e $d_{i_y} = |\mathbf{y} - \mathbf{y}_i|$ são as distâncias entre as coordenadas $[\mathbf{x}, \mathbf{y}]$ e os nós $[\mathbf{x}_i, \mathbf{y}_i]$, onde r_{i_x} e r_{i_y} são os tamanhos do suporte nodal para a função peso $w_i(\mathbf{x})$. A função das equações (3.14) e (3.15) tem continuidade C^1 e, portanto, as funções de forma em (3.12) também tem a mesma ordem de continuidade em todo o domínio.

3.11.1.3 - Campo elástico

Agora o campo elástico deve ser aproximado nas proximidades de \mathbf{x} . Assim, considerando a equação (3.11), pode-se montar a aproximação pelos MQM do campo de deslocamentos

$$\mathbf{u}(\mathbf{x}) = \begin{bmatrix} u^h(\mathbf{x}) \\ v^h(\mathbf{x}) \end{bmatrix} = \begin{bmatrix} \phi_1(\mathbf{x}) & 0 & \dots & \phi_n(\mathbf{x}) & 0 \\ 0 & \phi_1(\mathbf{x}) & \dots & 0 & \phi_n(\mathbf{x}) \end{bmatrix} \begin{bmatrix} \hat{u}_1 \\ \hat{v}_1 \\ \vdots \\ \hat{u}_n \\ \hat{v}_n \end{bmatrix} = \mathbf{\Phi}(\mathbf{x}) \hat{\mathbf{u}} \quad (3.16)$$

e do campo de deformações específicas

$$\boldsymbol{\varepsilon} = \mathbf{L} \mathbf{u} = \mathbf{L} \mathbf{\Phi} \hat{\mathbf{u}} = \mathbf{B} \hat{\mathbf{u}}, \quad (3.17)$$

assumindo linearidade geométrica no operador diferencial $\mathbf{L}(\cdot)$, então tem-se que

$$\mathbf{B} = \begin{bmatrix} \phi_{1,1} & 0 & \dots & \phi_{n,1} & 0 \\ 0 & \phi_{1,2} & \dots & 0 & \phi_{n,2} \\ \phi_{1,2} & \phi_{1,1} & \dots & \phi_{n,2} & \phi_{n,1} \end{bmatrix}. \quad (3.18)$$

Finalmente, o campo de tensões pode ser aproximado pela lei de Hooke generalizada

$$\boldsymbol{\sigma} = \mathbf{D} \boldsymbol{\varepsilon} = \mathbf{D} \mathbf{B} \hat{\mathbf{u}} \quad (3.19)$$

assim como as componentes das forças de superfície

$$\mathbf{t} = \mathbf{n} \boldsymbol{\sigma} = \mathbf{n} \mathbf{D} \mathbf{B} \hat{\mathbf{u}}, \quad (3.20)$$

onde \mathbf{D} representa a matriz constitutiva do material, dada por

$$\mathbf{D} = \underbrace{\frac{E}{(1+\nu)(1-2\nu)} \begin{bmatrix} 1-\nu & \nu & 0 \\ \nu & 1-\nu & 0 \\ 0 & 0 & \frac{1-2\nu}{2} \end{bmatrix}}_{\text{Estado Plano de Tensão}} \quad \text{ou} \quad \underbrace{\frac{E}{1-\nu^2} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{bmatrix}}_{\text{Estado Plano de Deformação}} \quad (3.21)$$

e \mathbf{n} é a matriz das componentes do vetor normal unitário, definida por

$$\mathbf{n} = \begin{bmatrix} n_1 & 0 & n_2 \\ 0 & n_2 & n_1 \end{bmatrix}. \quad (3.22)$$

As equações (3.16) a (3.22) mostram que, em um ponto x qualquer, as variáveis do campo elástico são definidas em termo das incógnitas nodais \hat{u} .

3.11.2 - Forma Local do Teorema do Trabalho

A solução para um determinado problema físico nada mais é do que um campo elástico totalmente admissível, que satisfaça tanto a admissibilidade cinemática quanto a estática. Sendo assim, se essa solução existe, como visto em Fichera (2006), ela é a única capaz de promover a estabilidade e a linearidade admissíveis pelo material.

Baseando-se na teoria fundamental do cálculo variacional, essa solução única proposta leva diretamente ao teorema do trabalho virtual, que nada mais é do que a minimização da energia potencial de um problema físico (Reddy, 2006).

O teorema geral do trabalho estabelece uma relação energética entre um campo de tensões estaticamente admissível qualquer, com um campo de deformações cinematicamente admissível qualquer, em um mesmo corpo ou domínio. Derivado diretamente da equação dos resíduos ponderados, o teorema do trabalho serve como uma base unificadora para as formulações numéricas de modelos contínuos (Brebbia e Walker, 2013).

Seja Ω um domínio bidimensional com contorno definido por $\Gamma = \Gamma_u \cup \Gamma_t$. Considera-se um campo de tensões estaticamente admissível, ou seja, que satisfaça a condição de equilíbrio

$$\mathbf{L}^T \boldsymbol{\sigma} + \mathbf{b} = \mathbf{0}, \quad (3.23)$$

válida no domínio Ω , com a condição de contorno na fronteira natural Γ_t

$$\mathbf{t} = \mathbf{n} \boldsymbol{\sigma} = \bar{\mathbf{t}} \quad (3.24)$$

onde o vetor $\boldsymbol{\sigma}$ representa as componentes de tensão, \mathbf{L} é o operador de derivadas matricial, o vetor \mathbf{t} representa as componentes das forças superficiais, $\bar{\mathbf{t}}$ são valores das forças superficiais prescritas e \mathbf{n} é o vetor unitário normal ao contorno Γ .

Seja Q um ponto do domínio Ω contido em um subdomínio arbitrário Ω_Q com fronteira constituída por $\Gamma_Q = \Gamma_{Qi} \cup \Gamma_{Qt} \cup \Gamma_{Qu}$, onde Γ_{Qi} designa o contorno interno do subdomínio, $\Gamma_{Qt} = \Gamma_t \cap \Gamma_Q$ e $\Gamma_{Qu} = \Gamma_u \cap \Gamma_Q$, como mostrado na Figura 3.8. Devido a sua arbitrariedade, o domínio local pode se sobrepor a outros domínios locais similares.

A forma forte da equação dos resíduos ponderados para o domínio local do ponto Q pode ser escrita da forma

$$\int_{\Omega_Q} (\mathbf{L}^T \boldsymbol{\sigma} + \mathbf{b})^T \mathbf{W}_\Omega d\Omega + \int_{\Gamma_{Qt}} (\mathbf{t} - \bar{\mathbf{t}})^T \mathbf{W}_\Gamma d\Gamma = \mathbf{0}, \quad (3.25)$$

onde \mathbf{W}_Ω e \mathbf{W}_Γ são funções peso arbitrárias definidas em Ω e Γ , respectivamente. Integrando

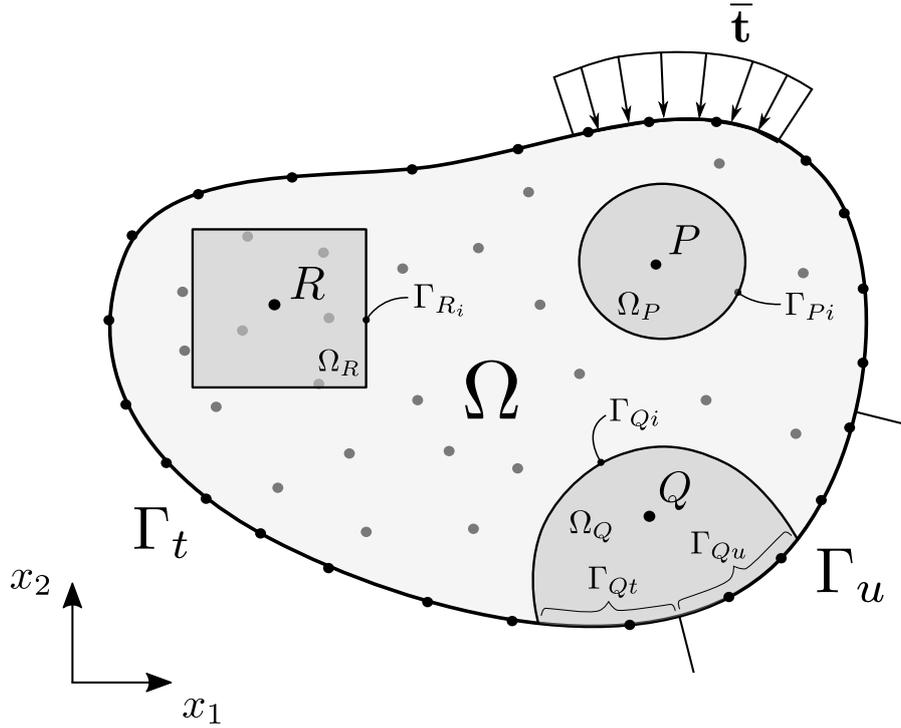


Figura 3.8 – Representação do domínio global Ω , suas fronteiras natural Γ_t e essencial Γ_u e os subdomínios Ω_Q associados ao nó Q , com contorno interno $\Gamma_Q = \Gamma_{Qi} \cup \Gamma_{Qt} \cup \Gamma_{Qu}$. Nós P e R , de forma similar, possuem domínios locais correspondentes a Ω_P e Ω_R .

o termo de domínio da equação (3.25) por partes (teorema da divergência), obtém-se a forma fraca local da equação dos resíduos ponderados

$$\int_{\Gamma_Q} (\mathbf{n}\boldsymbol{\sigma})^T \mathbf{W}_\Omega d\Gamma - \int_{\Omega_Q} (\boldsymbol{\sigma}^T \mathbf{L}\mathbf{W}_\Omega - \mathbf{b}^T \mathbf{W}_\Omega) d\Omega + \int_{\Gamma_{Qt}} (\mathbf{t} - \bar{\mathbf{t}})^T \mathbf{W}_\Gamma d\Gamma = 0. \quad (3.26)$$

Por conveniência, a função peso do contorno, \mathbf{W}_Γ , é avaliada como

$$\mathbf{W}_\Gamma = -\mathbf{W}_\Omega, \quad (3.27)$$

no contorno Γ_{Qt} . Assim, a equação (3.26) resulta em

$$\int_{\Gamma_Q - \Gamma_{Qt}} \mathbf{t}^T \mathbf{W}_\Omega d\Gamma + \int_{\Gamma_{Qt}} \bar{\mathbf{t}}^T \mathbf{W}_\Omega d\Gamma - \int_{\Omega_Q} (\boldsymbol{\sigma}^T \mathbf{L}\mathbf{W}_\Omega - \mathbf{b}^T \mathbf{W}_\Omega) d\Omega = 0 \quad (3.28)$$

Percebe-se que \mathbf{W}_Ω deve ter ordem de continuidade compatível com o operador \mathbf{L} , pois é uma condição de admissibilidade para a integração.

Seja $\boldsymbol{\varepsilon}^*$ um campo de deformações cinematicamente admissível, associado a um campo de deslocamentos \mathbf{u}^* com derivadas suficientemente pequenas para garantir-se a linearidade

geométrica, dado por

$$\boldsymbol{\varepsilon}^* = \mathbf{L} \mathbf{u}^*, \quad (3.29)$$

no domínio Ω , com condições de contorno

$$\mathbf{u}^* = \bar{\mathbf{u}}, \quad (3.30)$$

válidas em Γ_u^* . Considerando as funções peso arbitrárias \mathbf{W}_Ω como

$$\mathbf{W}_\Omega = \mathbf{u}^*, \quad (3.31)$$

a equação dos resíduos ponderados (3.28) reduz-se a

$$\int_{\Gamma_Q - \Gamma_{Qt} - \Gamma_{Qu}} \mathbf{t}^T \mathbf{u}^* \, d\Gamma + \int_{\Gamma_{Qu}} \mathbf{t}^T \bar{\mathbf{u}}^* \, d\Gamma + \int_{\Gamma_{Qt}} \bar{\mathbf{t}}^T \mathbf{u}^* \, d\Gamma - \int_{\Omega_Q} (\boldsymbol{\sigma}^T \mathbf{L} \mathbf{u}^* - \mathbf{b}^T \mathbf{u}^*) \, d\Omega = \mathbf{0} \quad (3.32)$$

que pode ser escrita de de forma compacta

$$\int_{\Gamma_Q} \mathbf{t}^T \mathbf{u}^* \, d\Gamma + \int_{\Omega_Q} \mathbf{b}^T \mathbf{u}^* \, d\Omega = \int_{\Omega_Q} \boldsymbol{\sigma}^T \boldsymbol{\varepsilon}^* \, d\Omega. \quad (3.33)$$

Essa equação, que expressa a dualidade estático-cinemática, é a forma local do conhecido teorema do trabalho, uma das identidades fundamentais da mecânica dos sólidos deformáveis, como proposto por Sokolnikoff (1956).

A equação (3.33) é o ponto de partida da formulação cinematicamente admissível do método desenvolvido neste trabalho, assim como de outras formulações, como visto em Oliveira e Portela (2016); que merece algumas ressalvas, sendo elas:

- O campo de tensões $\boldsymbol{\sigma}$, é qualquer campo que satisfaça o equilíbrio das forças externas \mathbf{b} e \mathbf{t} aplicadas, não sendo necessariamente o campo de tensões que efetivamente está atuando no corpo;
- O campo de deformações $\boldsymbol{\varepsilon}^*$, é qualquer campo gerado por \mathbf{u}^* que seja compatível com as restrições $\mathbf{u}^* = \bar{\mathbf{u}}$, não sendo necessariamente o campo de deformações que efetivamente está atuando no corpo;
- Tanto o campo de tensões $\boldsymbol{\sigma}$ quanto o campo de deformações $\boldsymbol{\varepsilon}^*$, não estão conectados por qualquer relação constitutiva. Esses campos são completamente independentes, como uma consequência da arbitrariedade da função ponderadora \mathbf{W}_Ω ;
- Por fim, o domínio local Ω_Q é qualquer subdomínio arbitrário atuando no corpo, onde os campos independentes $\boldsymbol{\sigma}$ e $\boldsymbol{\varepsilon}^*$ podem ser definidos.

A independência dos campos relacionados na equação (3.33), é a característica que torna possível e inédita as formulações do método sem malha local, pela escolha de um campo de deslocamentos cinematicamente admissível. É importante notar que nenhuma relação constitutiva foi adotada na dedução da equação (3.33). Portanto, pode-se tratar tanto de problemas da elasticidade quanto da plasticidade, desde que se garanta a linearidade geométrica.

Como estratégia de modelagem, considera-se que o campo local estaticamente admissível é um campo totalmente admissível. Assim, além de satisfazer as condições das equações (3.23) e (3.24), também satisfaz a condição de admissibilidade cinemática, ou seja

$$\boldsymbol{\varepsilon} = \mathbf{L} \mathbf{u} \quad (3.34)$$

válida no domínio Ω , com condição de contorno

$$\mathbf{u} = \bar{\mathbf{u}} \quad (3.35)$$

na fronteira cinemática Γ_u . Assume-se que o campo de deslocamentos \mathbf{u} é contínuo e tem derivadas suficientemente pequenas para que seja válida a hipótese da linearidade geométrica do campo de deformações $\boldsymbol{\varepsilon}$. Sendo assim, a equação (3.35) deve ser imposta no modelo numérico, para que se obtenha a solução única para o determinado problema.

Para uma discretização de um corpo, a forma fraca local do domínio ou domínio de quadratura Ω_Q , centralizado em um nó Q , pode ser definido como o subdomínio local daquele nó, podendo este ser circular ou retangular, como mostra a Figura 3.8.

3.11.3 - Campo Elástico Generalizado (GSMF)

A formulação do campo elástico generalizado segue uma tendência de redução do esforço computacional nos métodos sem malha, eliminando todo o processo de integração numérica da montagem das matrizes do domínio local. Para tanto, é necessário abordar a descrição matemática dos campos elásticos sob uma nova perspectiva.

Na forma local do teorema do trabalho assumiu-se que o campo de deslocamentos cinematicamente admissíveis \mathbf{u}^* é uma função contínua, o que acarreta em um campo de deformações cinematicamente admissível $\boldsymbol{\varepsilon}^*$ contínuo, regular e integrável. No entanto, a condição de continuidade dos campos elásticos não é absolutamente obrigatória e pode ser violada, visto que é vantajoso representar $\boldsymbol{\varepsilon}^*$ em termos de funções generalizadas, no sentido da teoria das distribuições (Gelfand e Shilov, 1964).

Assim, o campo de deslocamentos é representado por uma função contínua segmentada definida em termos da função *Heaviside step function* gerando, conseqüentemente, uma função generalizada correspondente a um campo de deformações definido pela função delta de *Dirac*.

Com o intuito de simplificar as expressões matemáticas, quando se trabalha com funções *Heaviside* e delta de *Dirac* em duas dimensões, considera-se uma função distância escalar d , dada por

$$d = \| \mathbf{x} - \mathbf{x}_Q \| \quad \text{que é} \quad \begin{cases} d = 0 & \text{se } \mathbf{x} \equiv \mathbf{x}_Q \\ d > 0 & \text{se } \mathbf{x} \neq \mathbf{x}_Q, \end{cases} \quad (3.36)$$

que representa a função de valor absoluto da distância entre um ponto \mathbf{x} e um ponto de referência \mathbf{x}_Q , no domínio local $\Omega_Q \cup \Gamma_Q$ pertencente a um nó Q . Assim, essa definição sempre assume que $d = d(\mathbf{x}, \mathbf{x}_Q) \geq 0$ tem valor positivo, ou nulo toda vez que \mathbf{x} e \mathbf{x}_Q são pontos coincidentes. É importante lembrar que, na equação (3.36), tanto o ponto \mathbf{x} quanto o ponto de referência \mathbf{x}_Q não precisam ser necessariamente pontos nodais do domínio local.

Considerando a função escalar $d \supset d(\mathbf{x}, \mathbf{x}_Q)$, a função *Heaviside* pode ser definida como

$$H(d) = \begin{cases} 1 & \text{if } d \leq 0 \text{ (} d = 0 \text{ for } \mathbf{x} \equiv \mathbf{x}_Q \text{),} \\ 0 & \text{if } d > 0 \text{ that is } \mathbf{x} \neq \mathbf{x}_Q, \end{cases} \quad (3.37)$$

onde assume-se a descontinuidade em \mathbf{x}_Q . Conseqüentemente, define-se a função delta de *Dirac* com as seguintes propriedades

$$\delta(d) = H'(d) = \begin{cases} \infty & \text{if } d = 0 \text{ that is } \mathbf{x} \equiv \mathbf{x}_Q, \\ 0 & \text{if } d \neq 0 \text{ (} d > 0 \text{ for } \mathbf{x} \neq \mathbf{x}_Q \text{)} \end{cases} \quad \text{and} \quad \int_{-\infty}^{+\infty} \delta(d) d\Gamma = 1, \quad (3.38)$$

onde $H'(d)$ representa a derivada de $H(d)$ com respeito a d . Note que a derivada de $H(d)$, com respeito a coordenada x_i , pode ser definida como

$$H(d)_{,i} = H'(d) d_{,i} = \delta(d) d_{,i} = \delta(d) n_i. \quad (3.39)$$

Como o resultado dessa equação não é afetado pelo valor da constante n_i , ela será convenientemente redefinida posteriormente. Considere agora que d_l , d_j e d_k representam a função distância d , definida na equação (3.36), para os correspondentes pontos de colocação \mathbf{x}_l , \mathbf{x}_j e \mathbf{x}_k . Assim, quando as equações (3.36) a (3.38) são consideradas, o campo de

deslocamentos cinematicamente admissível pode ser convenientemente definido como

$$\mathbf{u}^*(\mathbf{x}) = \left[\frac{L_i}{n_i} \sum_{l=1}^{n_i} H(d_l) + \frac{L_t}{n_t} \sum_{j=1}^{n_t} H(d_j) + \frac{S}{n_\Omega} \sum_{k=1}^{n_\Omega} H(d_k) \right] \mathbf{e}, \quad (3.40)$$

onde $\mathbf{e} = [1 \ 1]^T$ representa a métrica da direção ortogonal; n_i , n_t e n_Ω representam o número de pontos de colocação, respectivamente no contorno local $\Gamma_{Q_i} = \Gamma_Q - \Gamma_{Q_t} - \Gamma_{Q_u}$ com comprimento L_i , no contorno local estático Γ_{Q_t} com comprimento L_t e no domínio local Ω_Q com área S . Esse campo de deslocamentos definido $\mathbf{u}^*(\mathbf{x})$, é um deslocamento unitário de corpo rígido discreto definido nos pontos de colocação, esquematicamente representado na Figura 3.9. Sendo assim, quando a equação (3.39) é considerada, o campo de deformações

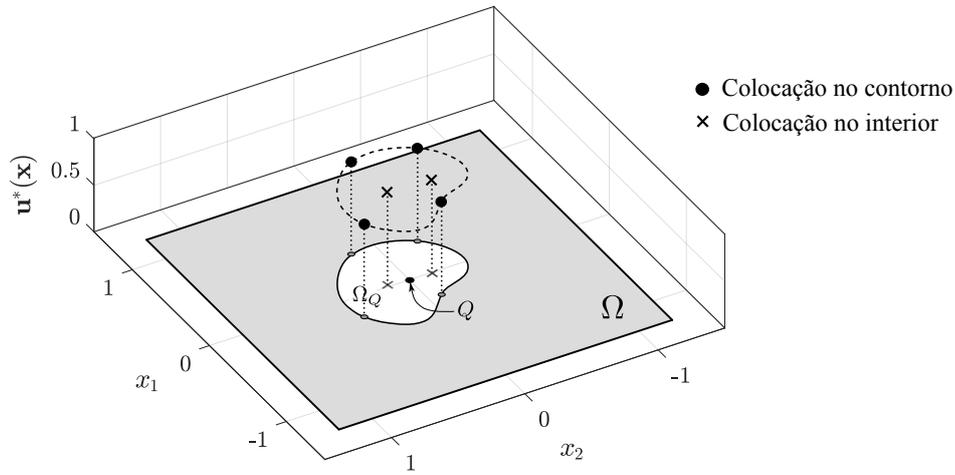


Figura 3.9 – Esquema representando o deslocamento $\mathbf{u}^*(\mathbf{x})$, dado pela equação (3.40), um deslocamento unitário de corpo rígido discreto definido nos pontos de colocação, da formulação do GSMF, para um domínio local arbitrário associado ao nó Q .

$\boldsymbol{\varepsilon}^*(\mathbf{x})$ pode ser dado por

$$\begin{aligned} \boldsymbol{\varepsilon}^*(\mathbf{x}) = \mathbf{L} \mathbf{u}^*(\mathbf{x}) &= \left[\frac{L_i}{n_i} \sum_{l=1}^{n_i} \mathbf{L} H(d_l) + \frac{L_t}{n_t} \sum_{j=1}^{n_t} \mathbf{L} H(d_j) + \frac{S}{n_\Omega} \sum_{k=1}^{n_\Omega} \mathbf{L} H(d_k) \right] \mathbf{e} = \\ &= \left[\frac{L_i}{n_i} \sum_{l=1}^{n_i} \delta(d_l) \mathbf{n}^T + \frac{L_t}{n_t} \sum_{j=1}^{n_t} \delta(d_j) \mathbf{n}^T + \frac{S}{n_\Omega} \sum_{k=1}^{n_\Omega} \delta(d_k) \mathbf{n}^T \right] \mathbf{e}, \end{aligned} \quad (3.41)$$

Tendo definido os deslocamentos e as componentes da deformação do campo cinematicamente admissível, respectivamente as equações (3.40) e (3.41), o teorema local do trabalho pode ser escrito como

$$\int_{\Gamma_Q - \Gamma_{Q_t}} \mathbf{t}^T \mathbf{u}^* d\Gamma + \int_{\Gamma_{Q_t}} \bar{\mathbf{t}}^T \mathbf{u}^* d\Gamma + \int_{\Omega_Q} \mathbf{b}^T \mathbf{u}^* d\Omega = \int_{\Omega_Q} \boldsymbol{\sigma}^T \boldsymbol{\varepsilon}^* d\Omega \quad (3.42)$$

resultando em

$$\begin{aligned} \frac{L_i}{n_i} \sum_{l=1}^{n_i} \int_{\Gamma_Q - \Gamma_{Qt}} \mathbf{t}^T H(d_l) \mathbf{e} \, d\Gamma + \frac{L_t}{n_t} \sum_{j=1}^{n_t} \int_{\Gamma_{Qt}} \bar{\mathbf{t}}^T H(d_j) \mathbf{e} \, d\Gamma + \frac{S}{n_\Omega} \sum_{k=1}^{n_\Omega} \int_{\Omega_Q} \mathbf{b}^T H(d_k) \mathbf{e} \, d\Omega = \\ = \frac{S}{n_\Omega} \sum_{k=1}^{n_\Omega} \int_{\Omega_Q} \boldsymbol{\sigma}^T \delta(d_k) \mathbf{n}^T \mathbf{e} \, d\Omega. \end{aligned} \quad (3.43)$$

Considerando as propriedades da função *Heaviside*, definidas na equação (3.37), a equação (3.43) resulta em

$$\mathbf{e}^T \left[\frac{L_i}{n_i} \sum_{l=1}^{n_i} \mathbf{t}_{\mathbf{x}_l} + \frac{L_t}{n_t} \sum_{j=1}^{n_t} \bar{\mathbf{t}}_{\mathbf{x}_j} + \frac{S}{n_\Omega} \sum_{k=1}^{n_\Omega} \mathbf{b}_{\mathbf{x}_k} - \frac{S}{n_\Omega} \sum_{k=1}^{n_\Omega} \mathbf{n} \int_{\Omega_Q} \delta(d_k) \boldsymbol{\sigma} \, d\Omega \right] = \mathbf{0} \quad (3.44)$$

que, após considerar as propriedades seletivas da função delta de *Dirac*, resulta em

$$\frac{L_i}{n_i} \sum_{l=1}^{n_i} \mathbf{t}_{\mathbf{x}_l} - \frac{S}{n_\Omega} \mathbf{n} \sum_{k=1}^{n_\Omega} \boldsymbol{\sigma}_{\mathbf{x}_k} = - \frac{L_t}{n_t} \sum_{j=1}^{n_t} \bar{\mathbf{t}}_{\mathbf{x}_j} - \frac{S}{n_\Omega} \sum_{k=1}^{n_\Omega} \mathbf{b}_{\mathbf{x}_k}. \quad (3.45)$$

Finalmente, quando a variável \mathbf{n} é arbitrariamente definida com componentes nulas $n_i = 0$, como apresentado na equação (3.39), a equação (3.45) resulta em

$$\frac{L_i}{n_i} \sum_{l=1}^{n_i} \mathbf{t}_{\mathbf{x}_l} = - \frac{L_t}{n_t} \sum_{j=1}^{n_t} \bar{\mathbf{t}}_{\mathbf{x}_j} - \frac{S}{n_\Omega} \sum_{k=1}^{n_\Omega} \mathbf{b}_{\mathbf{x}_k} \quad (3.46)$$

A equação (3.46) demonstra o equilíbrio das forças de superfície e das forças de corpo, pontualmente definida nos pontos de colocação, como demonstra a Figura 3.10; que na verdade é equivalente a versão pontual do princípio das tensões de Euler e Cauchy. Essa é a equação utilizada na formulação do campo elástico generalizado do método sem malha local ou *Generalized-Strain Mesh-free formulation* (GSMF), que é livre de integração numérica. Como o teorema do trabalho é a forma fraca advinda dos resíduos ponderados, é facilmente perceptível que essa formulação nada mais é do que uma colocação na forma fraca. A colocação na forma fraca supera as dificuldades conhecidas da colocação na forma forte, como baixa precisão e instabilidade da solução, o que torna esta uma formulação confiável e robusta.

Discretiza-se a equação (3.46), introduzindo-se as equações do campo elástico, gerando um sistema linear de duas equações referentes ao nó $Q \in \Omega_Q$ em função das incógnitas nodais $\hat{\mathbf{u}}$

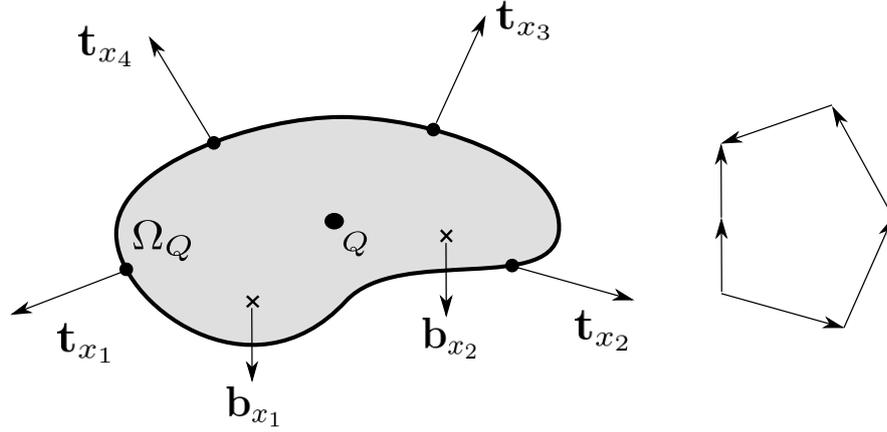


Figura 3.10 – Esquema representando o equilíbrio das forças de corpo e forças de superfície, da equação (3.46), pontualmente definidas nos pontos de colocação de um domínio local associado ao nó Q , da formulação do GSMF.

$$\frac{L_i}{n_i} \sum_{l=1}^{n_i} \mathbf{n}_{\mathbf{x}_l} \mathbf{D} \mathbf{B}_{\mathbf{x}_l} \hat{\mathbf{u}} = - \frac{L_t}{n_t} \sum_{j=1}^{n_t} \bar{\mathbf{t}}_{\mathbf{x}_j} - \frac{S}{n_\Omega} \sum_{k=1}^{n_\Omega} \mathbf{b}_{\mathbf{x}_k} \quad (3.47)$$

que pode ser reescrito como

$$\mathbf{K}_Q \hat{\mathbf{u}} = \mathbf{F}_Q, \quad (3.48)$$

onde \mathbf{K}_Q é a matriz de rigidez nodal do domínio local Ω_Q , uma matriz $2 \times 2n$ dada por

$$\mathbf{K}_Q = \frac{L_i}{n_i} \sum_{l=1}^{n_i} \mathbf{n}_{\mathbf{x}_l} \mathbf{D} \mathbf{B}_{\mathbf{x}_l} \quad (3.49)$$

e \mathbf{F}_Q , é o respectivo vetor de forças dado por

$$\mathbf{F}_Q = - \frac{L_t}{n_t} \sum_{j=1}^{n_t} \bar{\mathbf{t}}_{\mathbf{x}_j} - \frac{S}{n_\Omega} \sum_{k=1}^{n_\Omega} \mathbf{b}_{\mathbf{x}_k} \quad (3.50)$$

A equação (3.48) é aplicada aos M domínios internos, gerando um sistema global $2M \times 2N$

$$\mathbf{K} \hat{\mathbf{u}} = \mathbf{F}. \quad (3.51)$$

Finalmente, para um nó no contorno cinemático, uma interpolação direta pode ser utilizada para impor as condições de contorno como

$$\mathbf{u}_k = \Phi_k \hat{\mathbf{u}} = \bar{\mathbf{u}}_k, \quad (3.52)$$

com $k = 1, 2$, onde $\bar{\mathbf{u}}_k$ é a componente do deslocamento nodal prescrito. A equação (3.52) é diretamente adicionada no sistema global, equação (3.51).

4 - MECÂNICA DA FRATURA LINEAR ELÁSTICA

Este capítulo reflete fielmente o respectivo tema no trabalho de Oliveira (2019).

4.1 - CONCEITOS FUNDAMENTAIS

Nessa seção serão apresentados os conceitos fundamentais para o desenvolvimento e aplicação da técnica da subtração da singularidade e a análise de múltiplas trincas, no âmbito da mecânica da fratura linear elástica.

4.1.1 - Modos de Deslocamento na Ponta da Trinca

Considere o corpo trincado representado na Figura 4.1, no qual o plano da trinca se encontra no plano x_1x_3 e a frente da trinca (*crack front*) é paralela a direção da coordenada x_3 . Os modos básicos de deslocamento na ponta da trinca são apresentados na Figura 4.1. O modo de abertura, ou modo de tensão, é aquele no qual a trinca é separada simetricamente com respeito aos planos x_1x_2 e x_1x_3 , denominado *modo I*. O modo de cisalhamento no plano, ou *modo II*, é aquele no qual os deslocamentos são simétricos ao plano x_1x_2 e antissimétricos ao plano x_1x_3 . O modo de cisalhamento fora do plano, ou *modo III*, é aquele em que os deslocamentos que são antissimétricos aos planos x_1x_2 e x_1x_3 . A partir desses três modos básicos de deslocamento, qualquer deslocamento de trinca pode ser representado por meio de uma superposição linear, fazendo-se assim uma combinação desses modos, como mostra Irwin (1957).

4.1.2 - Campo Elástico Local

A formulação do critério de crescimento de trinca é baseada na forma analítica do campo elástico na ponta da trinca. Desta forma, uma investigação do campo elástico nos arredores da ponta da trinca também se faz necessária para a presente pesquisa.

Nos limites da elasticidade linear, o campo de tensões é infinito na ponta da trinca. Brahtz (1933) foi o primeiro a discutir o assunto e posteriormente Williams (1952a), que depois de realizar uma investigação sobre a forma analítica dessa singularidade demonstrou que, sob todas as possíveis combinações de condições de contorno, a tensão se torna infinita na ponta

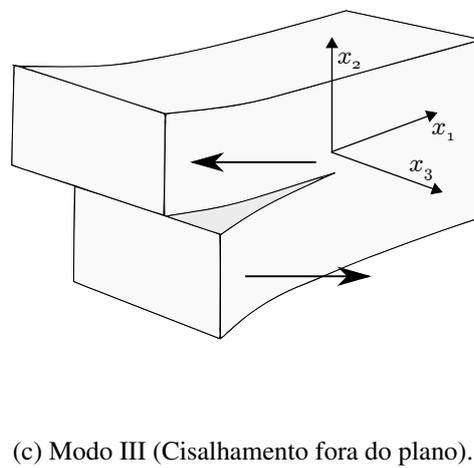
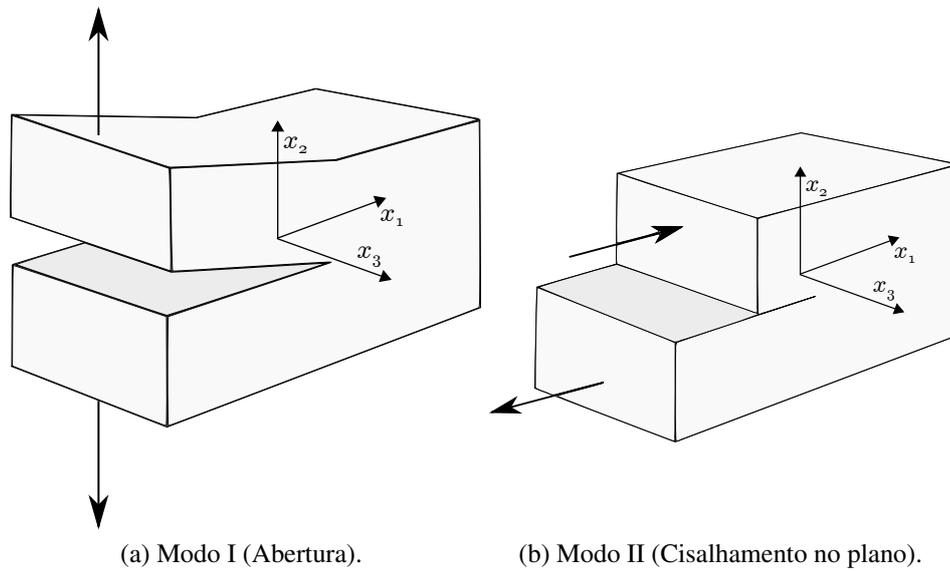


Figura 4.1 – Modos de deslocamento da ponta da trinca básicos de um corpo trincado.

da trinca.

4.1.2.1 - O problema plano

O *Eigenfunction Expansion Method* (EEM) foi utilizado por Williams (1952a) para obter soluções locais para problemas planos, lineares, elásticos, semi-infinitos e de fatias homogêneas (*homogeneous wedge*). O caso da fatia plana (*plane wedge*) com trações nulas em ambas as faces do entalhe (*notch*) será destacado aqui.

Seja r, θ um sistema de coordenadas polares centralizado na ponta de uma fatia semi-infinita, de tal forma que o eixo $\theta = 0$, o bissetor da fatia de ângulo 2α , é coincidente com o eixo cartesiano x , como mostra a Figura 4.2.

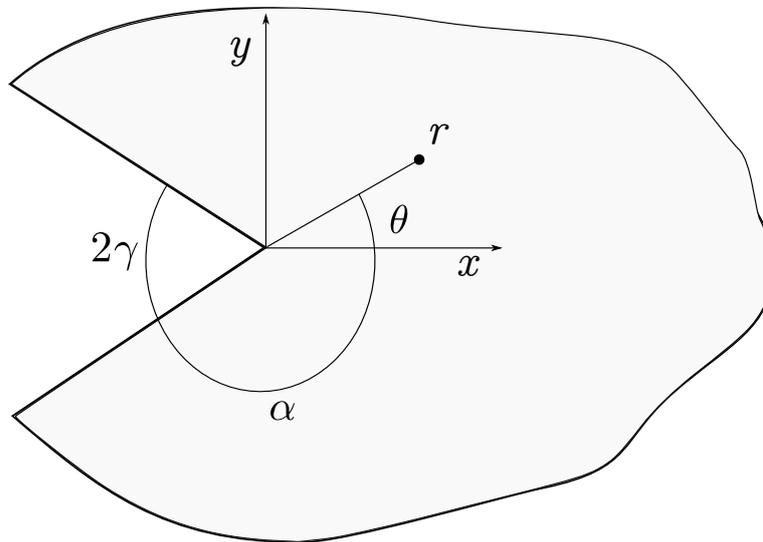


Figura 4.2 – Geometria de um entalhe angular no sistema de coordenadas polares e cartesianas.

O ângulo total ocupado pelo entalhe é $2\gamma = 2(\pi - \alpha)$. As funções de tensão de Airy χ , que satisfazem a equação bi-harmônica, podem ser utilizadas para expressar os campos de tensão e de deslocamento nos arredores da ponta do entalhe (*notch-tip*). No geral

$$\sigma_{rr} = \frac{1}{r} \frac{\partial^2 \chi}{\partial \theta^2} + \frac{1}{r} \frac{\partial \chi}{\partial r}, \quad (4.1)$$

$$\sigma_{\theta\theta} = \frac{\partial^2 \chi}{\partial r^2}, \quad (4.2)$$

$$\sigma_{r\theta} = -\frac{1}{r} \frac{\partial^2 \chi}{\partial r \partial \theta} + \frac{1}{r^2} \frac{\partial \chi}{\partial \theta}, \quad (4.3)$$

$$u_r = \frac{1}{2\mu} \left[-\frac{\partial\chi}{\partial r} + (1-\eta)r \frac{\partial\psi}{\partial\theta} \right], \quad (4.4)$$

e

$$u_\theta = \frac{1}{2\mu} \left[-\frac{\partial\chi}{\partial\theta} + (1-\eta)r^2 \frac{\partial\psi}{\partial r} \right], \quad (4.5)$$

com $\eta = \nu$ para o caso de estado plano de deformação e $\eta = \nu/(1+\nu)$ para o caso de estado plano de tensão; onde ν é o coeficiente de Poisson e μ é o módulo de cisalhamento. A função ψ é harmônica e está relacionada com a função de tensão χ por

$$\nabla^2\chi = \frac{\partial}{\partial r} \left(r \frac{\partial\psi}{\partial\theta} \right). \quad (4.6)$$

Seguindo os procedimentos apresentados por Williams (1952a), soluções para χ podem ser obtidas na forma

$$\chi = \mathbb{R} [r^{\lambda+1} F(\theta; \lambda)] \quad (4.7)$$

e de forma similar, soluções para ψ na forma

$$\psi = \mathbb{R} [r^m G(\theta; m)], \quad (4.8)$$

onde λ e m podem ser números complexos. As funções $F(\theta; \lambda)$ e $G(\theta; m)$ são soluções gerais das equações diferenciais, obtidas quando (4.7) e (4.8) são introduzidas na equação harmônica e bi-harmônica, respectivamente. A solução geral $F(\theta; \lambda)$ é válida apenas para $\lambda \neq 0, 1$. Vale ressaltar que esses valores excepcionais também serão examinados posteriormente. Usando a relação (4.6), pode-se encontrar que $m = \lambda - 1$ e $G(\theta; m)$ são transformados em $G(\theta; \lambda)$. Dessa forma, as tensões são dados por

$$\sigma_{rr} = \mathbb{R} \left\{ r^{\lambda-1} [F'' + (\lambda+1)F] \right\}, \quad (4.9)$$

$$\sigma_{\theta\theta} = \mathbb{R} \left\{ r^{\lambda-1} (\lambda+1) \lambda F \right\} \quad (4.10)$$

e

$$\sigma_{r\theta} = -\mathbb{R} \left\{ r^{\lambda-1} \lambda F' \right\}. \quad (4.11)$$

Os deslocamentos são dados por

$$u_r = \mathbb{R} \left\{ \frac{1}{2\mu} r^\lambda [-(\lambda+1)F + (1-\eta)G''] \right\} \quad (4.12)$$

e

$$u_\theta = \mathbb{R} \left\{ \frac{1}{2\mu} r^\lambda [-F'' + (1-\eta)(\lambda-1)G] \right\}, \quad (4.13)$$

onde as aspas representam a diferenciação com respeito a θ . A solução geral apresentada contém quatro constantes desconhecidas e um parâmetro λ indeterminado.

Para estabelecer uma solução particular para o problema em questão, um conjunto de condições de contorno são necessárias. Essas condições devem ser prescritas ao longo do contorno da constante θ , porque as funções F e G são independentes de r . Os contornos da placa com entalhe correspondem às linhas radiais, como mostra a Figura 4.2. Ao longo desse contorno, tanto deslocamentos nulos quanto trações nulas podem ser prescritas para a maioria dos casos.

Para que se obtenha deslocamentos livres nas duas faces do entalhe, trações nulas são prescritas ao longo de $\theta = \pm\alpha$, que nada mais são do que, $\sigma_{\theta\theta}(r, \pm\alpha) = \sigma_{r\theta}(r, \pm\alpha) = 0$, que a partir das equações (4.10) e (4.11), implicam que

$$F(\theta = \pm\alpha; \lambda) = F'(\theta = \pm\alpha; \lambda) = 0, \quad (4.14)$$

onde $\lambda \neq 0$. Quando $\lambda = 0$, as condições de contorno nas faces do entalhe são automaticamente satisfeitas, tendo em vista que $\sigma_{\theta\theta} = \sigma_{r\theta} = 0$ na solução do domínio. Substituindo a solução geral $F(\theta; \lambda)$ em (4.14) e depois de algumas manipulações algébricas, dois conjuntos de equações independentes são obtidos em função das constantes de integração e de λ . Um desses sistemas resulta na solução angular simétrica enquanto o outro resulta na solução angular antissimétrica, sendo esta uma consequência natural da simetria angular completa, resultante da geometria e das condições de contorno. Uma solução não-trivial desse sistema de equações somente é possível quando o determinante dos seus respectivos sistemas desaparece, originando assim a equação característica para os autovalores de λ .

Com relação a parte simétrica da solução, a equação característica é dada por

$$\lambda \sin 2\alpha + \sin 2\lambda\alpha = 0 \quad (4.15)$$

e a parte antissimétrica da solução é dada por

$$\lambda \sin 2\alpha - \sin 2\lambda\alpha = 0, \quad (4.16)$$

dado $\lambda \neq 1$. Para $\lambda = 1$ a equação obtida é $\sin 2\alpha = 0$ para a parte simétrica e $2\alpha = \tan 2\alpha$ para a parte antissimétrica, ao invés das equações (4.15) e (4.16), respectivamente. Isso ocorre devido à forma degenerada da solução geral $F(\theta; \lambda)$ e devido à solução do sistema de equações, que é não-trivial apenas quando $\alpha = 90^\circ$ e $\alpha = 180^\circ$, e $\alpha = 128.725^\circ$, respectivamente para a parte simétrica e antissimétrica (Fenner, 1973). Aliás, esse ângulo é associado a um paradoxo na teoria da elasticidade, descrito por Sternberg e Koiter (1958).

Autovalores fisicamente admissíveis, resultando em tensões singulares na ponta do entalhe, devem satisfazer a condição

$$0 < \mathbb{R}[\lambda_1] < 1, \quad (4.17)$$

sendo λ_1 o autovalor dominante, como visto em Fenner (1973).

As equações características (4.15) e (4.16) podem ser resolvidas numericamente utilizando o método de interação de Muller, como proposto por Portela et al. (1991). A vantagem desse método em relação aos outros possíveis é que, mesmo que a interação se inicie com raízes reais, para um dado ângulo α , ela pode convergir para uma raiz complexa.

Um estudo abrangente sobre o comportamento das raízes das equações características (4.15) e (4.16) pode ser encontrando em Rösel (1987) e Vasilopoulos (1988). Mesmo assim é importante ressaltar que a solução angular simétrica é identificada com o *modo I*, enquanto a solução angular antissimétrica corresponde ao *modo II*.

Para uma dada coordenada angular θ , existe um conjunto infinito de autovalores $\lambda = \lambda_n$ na equação (4.15) para o *modo I* e $\lambda = \zeta_n$ na equação (4.16) para o *modo II*; e um conjunto infinito de autovetores correspondentes. A distribuição das tensões pode ser escrita em coordenadas cartesianas como:

$$\begin{aligned} \sigma_{xx} = & \\ & \sum_{n=1}^{\infty} \mathbb{R}\{r^{\lambda_n-1} \lambda_n \beta_n [(2 + \lambda_n \cos 2\alpha + \cos 2\lambda_n \alpha) \cos(\lambda_n - 1)\theta - (\lambda_n - 1) \cos(\lambda_n - 3)\theta] \\ & + r^{\zeta_n-1} \zeta_n \delta_n [-(2 + \zeta_n \cos 2\alpha - \cos 2\zeta_n \alpha) \sin(\zeta_n - 1)\theta + (\zeta_n - 1) \sin(\zeta_n - 3)\theta]\}, \end{aligned} \quad (4.18)$$

$$\begin{aligned} \sigma_{yy} = & \\ & \sum_{n=1}^{\infty} \mathbb{R}\{r^{\lambda_n-1} \lambda_n \beta_n [(2 - \lambda_n \cos 2\alpha - \cos 2\lambda_n \alpha) \cos(\lambda_n - 1)\theta + (\lambda_n - 1) \cos(\lambda_n - 3)\theta] \\ & + r^{\zeta_n-1} \zeta_n \delta_n [(-2 + \zeta_n \cos 2\alpha - \cos 2\zeta_n \alpha) \sin(\zeta_n - 1)\theta - (\zeta_n - 1) \sin(\zeta_n - 3)\theta]\}, \end{aligned} \quad (4.19)$$

e

$$\begin{aligned} \sigma_{xy} = & \\ & \sum_{n=1}^{\infty} \mathbb{R}\{r^{\lambda_n-1} \lambda_n \beta_n [-(\lambda_n \cos 2\alpha + \cos 2\lambda_n \alpha) \cos(\lambda_n - 1)\theta + (\lambda_n - 1) \sin(\lambda_n - 3)\theta] \\ & + r^{\zeta_n-1} \zeta_n \delta_n [-(\zeta_n \cos 2\alpha - \cos 2\zeta_n \alpha) \cos(\zeta_n - 1)\theta + (\zeta_n - 1) \cos(\zeta_n - 3)\theta]\}. \end{aligned} \quad (4.20)$$

O campo de deslocamentos associado pode ser descrito como

$$u = \sum_{n=1}^{\infty} \text{Re} \left\{ \frac{1}{2\mu} r^{\lambda_n} \beta_n [(\eta + \lambda_n \cos 2\alpha + \cos 2\lambda_n \alpha) \cos \lambda_n \theta - \lambda_n \cos(\lambda_n - 2)\theta] \right. \\ \left. - \frac{1}{2\mu} r^{\zeta_n} \delta_n [(\eta + \zeta_n \cos 2\alpha - \cos 2\zeta_n \alpha) \sin \zeta_n \theta - \zeta_n \sin(\zeta_n - 2)\theta] \right\}, \quad (4.21)$$

e

$$v = \sum_{n=1}^{\infty} \text{Re} \left\{ \frac{1}{2\mu} r^{\lambda_n} \beta_n [(\eta - \lambda_n \cos 2\alpha - \cos 2\lambda_n \alpha) \sin \lambda_n \theta + \lambda_n \sin(\lambda_n - 2)\theta] \right. \\ \left. + \frac{1}{2\mu} r^{\zeta_n} \delta_n [(\eta - \zeta_n \cos 2\alpha + \cos 2\zeta_n \alpha) \cos \zeta_n \theta + \zeta_n \sin(\zeta_n - 2)\theta] \right\}, \quad (4.22)$$

onde β_n e δ_n são autovetores da expansão para o *modo I* e *modo II*, respectivamente.

4.1.2.2 - Movimento de corpo rígido

O campo de deslocamentos de William, equações (4.21) e (4.22), não inclui movimentos de corpo rígido e conseqüentemente não podem ser utilizados na técnica de subtração da singularidade, na forma como está apresentada.

Os parâmetros $\lambda = 0$ e $\zeta = 0$ não são autovalores do problema, já que eles resultam em tensões igualmente nulas ao longo do domínio da solução. Dessa forma, esses parâmetros podem ser utilizados para representar um movimento de corpo rígido, que neste caso, nada mais é do que a translação do corpo. Efetivamente, quando $\lambda = 0$ e $\zeta = 0$ são introduzidos nas equações (4.21) e (4.22), obtêm-se os seguintes deslocamentos

$$u_T = \frac{\eta + 1}{2\mu} \beta_0 = \beta_T \quad (4.23)$$

e

$$v_T = \frac{\eta - 1}{2\mu} \delta_0 = \delta_T, \quad (4.24)$$

onde β_0 e δ_0 são constantes arbitrárias que definem a translação de corpo rígido nas coordenadas x e y , respectivamente.

Um campo de rotação de corpo rígido também pode ser definido a partir dos deslocamentos nas equações (4.21) e (4.22), que correspondem às equações características (4.15) para o modo de deformação simétrico e (4.16) para o modo de deformação antissimétrico. O parâmetro $\zeta = 1$ não é um autovalor do problema, já que este resulta em tensões igualmente nulas de *modo II* ao longo do domínio da solução. Dessa forma, esse parâmetro pode ser utilizado para representar um movimento de corpo rígido, que neste caso, nada mais é do

que a rotação do corpo. Efetivamente, quando $\zeta = 1$ é introduzido nas equações (4.21) e (4.22), obtêm-se os seguintes deslocamentos

$$u_R = -\frac{\eta + 1}{2\mu} r \delta_1 \sin \theta = -\delta_R r \sin \theta \quad (4.25)$$

e

$$v_R = \frac{\eta + 1}{2\mu} r \delta_1 \cos \theta = \delta_R r \cos \theta, \quad (4.26)$$

onde δ_R é uma constante arbitrária que define o movimento.

Por fim, os campos de deslocamento de corpo rígido (4.23) a (4.26) podem ser usados para generalizar o campo de William como

$$u = u_T + u_R + u_W \quad (4.27)$$

e

$$v = v_T + v_R + v_W, \quad (4.28)$$

onde as componentes u_W e v_W representam o campo de deslocamentos de William obtidos pelas equações (4.21) e (4.22), respectivamente. Pode-se observar que as equações (4.27) e (4.28) introduzem três novas variáveis desconhecidas (β_T , δ_T e δ_R).

4.1.3 - Fator de Intensidade de Tensão

Na ponta de um entalhe anguloso (*tip of a sharp notch*), os fatores de intensidade de tensão são os coeficientes das tensões singulares e podem ser definidas de forma similar à utilizada por Irwin (1957), como

$$K_I = \sqrt{2\pi} \lim_{r \rightarrow 0} r^{1-\lambda_1} \sigma_{yy}(r, \theta = 0, \alpha) \quad (4.29)$$

e

$$K_{II} = \sqrt{2\pi} \lim_{r \rightarrow 0} r^{1-\zeta_1} \sigma_{xy}(r, \theta = 0, \alpha) \quad (4.30)$$

que nada mais é do que

$$K_I = \sqrt{2\pi} \lambda_1 \beta_1 (1 + \lambda_1 - \lambda_1 \cos 2\alpha - \cos 2\lambda_1 \alpha) \quad (4.31)$$

e

$$K_{II} = \sqrt{2\pi} \zeta_1 \delta_1 (-1 + \zeta_1 - \zeta_1 \cos 2\alpha - \cos 2\zeta_1 \alpha), \quad (4.32)$$

onde λ_1 , β_1 e ζ_1 , δ_1 são os pares dominantes do modo de abertura e de cisalhamento, respectivamente.

Como apontado por Portela (1993), vale ressaltar que, de um ponto de vista estritamente matemático, os fatores de intensidade de tensão também podem ser definidos a partir do primeiro termo da série do campo de deslocamentos, equações (4.27) e (4.28). Nesse caso, o processo de limitação é realizado ao longo da superfície do entalhe (coordenada $\theta = \pm\alpha$).

4.1.4 - Postulado Fundamental

A expressão analítica da série de expansão do campo de tensões na ponta da trinca, equações (4.19) a (4.21), mostra que apenas o primeiro termo da série, para cada modo de deformação, tem uma contribuição não-trivial para o estado de tensão na ponta da trinca. Em virtude da forma analítica desses termos, o campo de tensões é da ordem $O(r^{\lambda_1-1})$, para o *modo I* e $O(r^{\zeta_1-1})$ para o *modo II*, tornando-se assim singular à medida que r tende a zero. Isso significa que a força dessas singularidades é o único parâmetro que pode caracterizar o carregamento e a deformação sofrida pela ponta da trinca. Por consequência, o fator de intensidade de tensão, que mede a força dessas singularidades, determina completamente o comportamento da trinca. Esse é o postulado fundamental da mecânica da fratura linear elástica introduzido por Irwin (1957).

4.1.5 - Campo Elástico na Ponta da Trinca

Para a análise de problemas envolvendo trinca no âmbito da mecânica da fratura linear elástica, a série de expansão de Williams do campo elástico na ponta de um entalhe, equações (4.18) a (4.20), serão apresentadas para o caso de uma trinca ($\alpha = 180^\circ$). Além do mais, como o comportamento de uma trinca é completamente determinado pelos fatores de intensidade de tensão, equações (4.29) a (4.32), apenas o primeiro termo da série de expansão precisa ser considerado. Dessa forma, para o caso em que $\alpha = 180^\circ$, a tensão na ponta da trinca é dada por

$$\sigma_{xx} = \frac{K_I}{\sqrt{2\pi r}} \cos \frac{\theta}{2} \left(1 - \sin \frac{\theta}{2} \sin \frac{3\theta}{2} \right) + \frac{K_{II}}{\sqrt{2\pi r}} \sin \frac{\theta}{2} \left(2 + \cos \frac{\theta}{2} \cos \frac{3\theta}{2} \right), \quad (4.33)$$

$$\sigma_{yy} = \frac{K_I}{\sqrt{2\pi r}} \cos \frac{\theta}{2} \left(1 + \sin \frac{\theta}{2} \sin \frac{3\theta}{2} \right) - \frac{K_{II}}{\sqrt{2\pi r}} \sin \frac{\theta}{2} \cos \frac{\theta}{2} \cos \frac{3\theta}{2} \quad (4.34)$$

e

$$\sigma_{xy} = \frac{K_I}{\sqrt{2\pi r}} \cos \frac{\theta}{2} \sin \frac{\theta}{2} \cos \frac{3\theta}{2} + \frac{K_{II}}{\sqrt{2\pi r}} \cos \frac{\theta}{2} \left(1 - \sin \frac{\theta}{2} \sin \frac{3\theta}{2} \right) \quad (4.35)$$

e os deslocamentos dados por

$$u = \frac{K_I}{4\mu} \sqrt{\frac{r}{2\pi}} \left[(2k - 1) \cos \frac{\theta}{2} \cos \frac{3\theta}{2} \right] + \frac{K_{II}}{4\mu} \sqrt{\frac{r}{2\pi}} \left[(2k + 3) \sin \frac{\theta}{2} \sin \frac{3\theta}{2} \right] \quad (4.36)$$

e

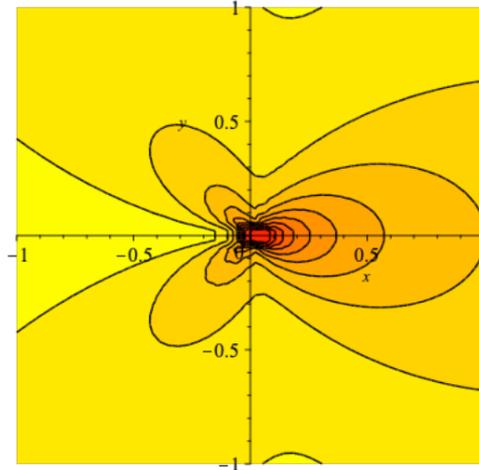
$$v = \frac{K_I}{4\mu} \sqrt{\frac{r}{2\pi}} \left[(2k + 1) \sin \frac{\theta}{2} \sin \frac{3\theta}{2} \right] + \frac{K_{II}}{4\mu} \sqrt{\frac{r}{2\pi}} \left[(2k - 3) \cos \frac{\theta}{2} \cos \frac{3\theta}{2} \right], \quad (4.37)$$

onde K_I e K_{II} são os fatores de intensidade de tensão de abertura e de cisalhamento, respectivamente. A constante μ é o módulo de cisalhamento e a constante k é dada por $3 - 4\nu$, para o estado plano de deformação, e $(3 - \nu)/(1 + \nu)$, para o caso de estado plano de tensão, onde ν é o coeficiente de Poisson. Com relação ao campo elástico de Williams vale ressaltar que, como o campo elástico foi formulado a partir de uma trinca semi-infinita e análise com autovalores, ele é válido nos arredores da ponta de uma trinca de borda arbitrária (*tip of an arbitrary edge crack*) e é válido para carregamentos arbitrários remotos.

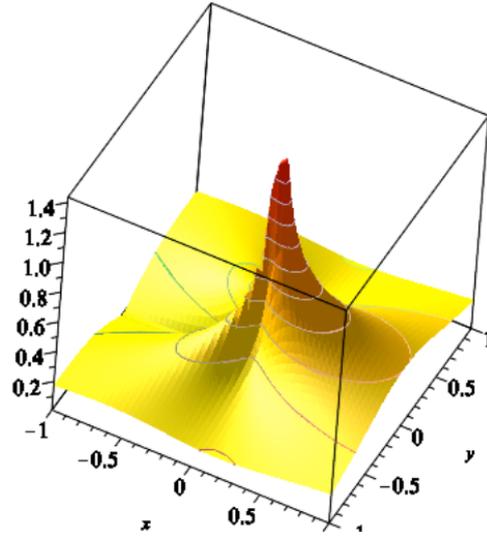
A Figura 4.3 ilustra as singularidades de Williams, para as tensões dadas pelas Equações 4.33 à 4.35, para o caso de trinca horizontal de borda, em Modo-I puro. Enquanto a Figura 4.4 apresenta o caso idêntico, exceto que para Modo-II puro.

A característica intrínseca que distingue uma trinca de borda (*edge crack*) de uma trinca interna (*internal crack*) é que, no primeiro caso, todo o caminho percorrido pelo carregamento (um padrão de curvas que indicam como o carregamento é transferido ao longo do objeto em estudo) passa à frente da ponta da trinca. Além disso, uma trinca emanando de um furo em uma placa retangular carregada é considerada uma trinca interna, porque apenas parte do caminho percorrido pelo carregamento passa à frente da ponta da trinca.

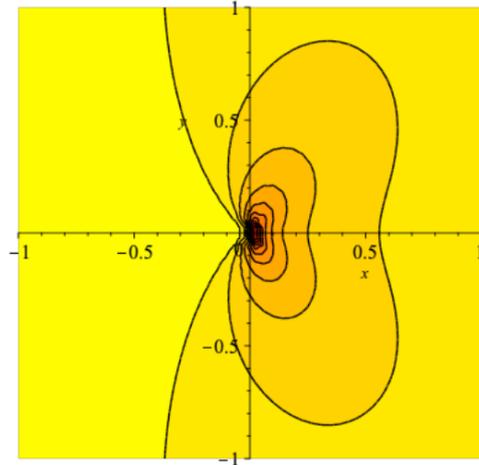
O campo elástico ao redor da ponta de uma trinca interna pode ser formulado a partir de uma análise do valor limite com variáveis complexas, como apresentado por Sih e Liebowitz (1968). Nesse método de análise, o carregamento remoto desempenha um papel fundamental na derivação da solução elástica. A base desse estudo foi o caso de uma placa retangular infinita contendo uma trinca inclinada interna, livre de forças de superfície, de ângulo α , sujeita a uma tensão biaxial remota com componentes σ e $k\sigma$, aplicadas uniformemente em relação a borda da placa, com ângulos de $\alpha = \pi/2$ e $\alpha = 0$, respectivamente. Esse caso foi estudado por Eftis et al. (1990) por meio de uma análise do valor limite com uma representação complexa do campo elástico obtido por Muskhelishvili (2010). Nesse caso, a aproximação de primeira ordem do campo de tensões na região da ponta da trinca é idêntica à correspondente aproximação do campo de tensões nos arredores da ponta da trinca de borda (*edge crack*), equações (4.33) a (4.37), com uma constante



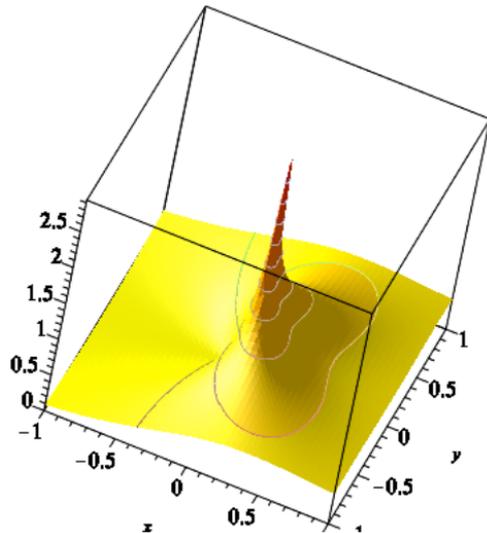
(a) σ_{xx} - 2D.



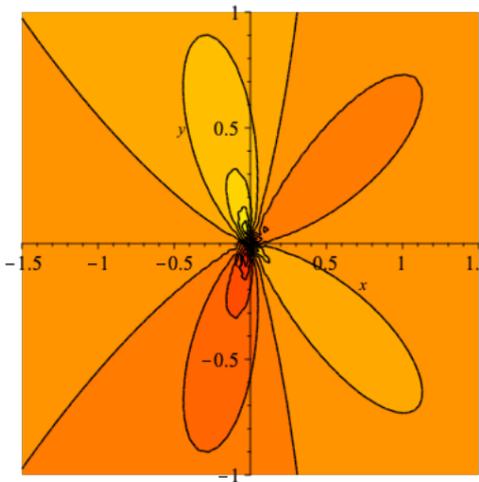
(b) σ_{xx} - 3D.



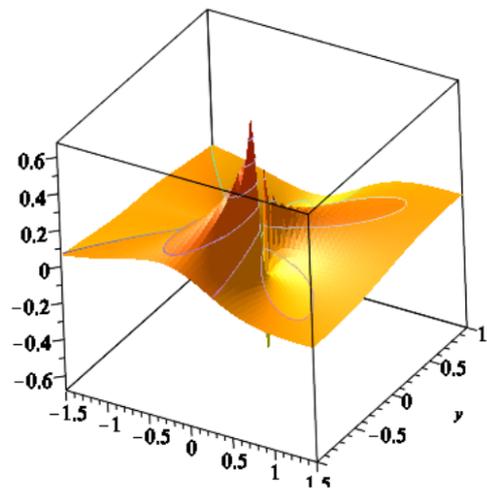
(c) σ_{yy} - 2D.



(d) σ_{yy} - 3D.

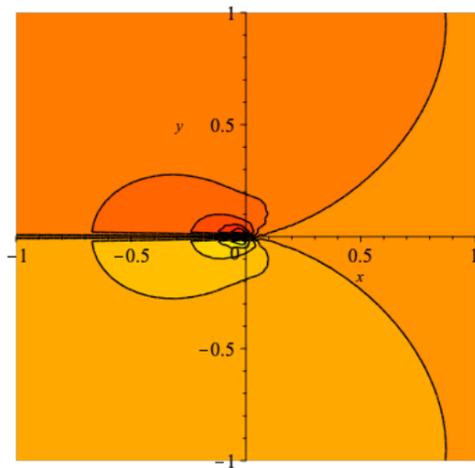


(e) σ_{xy} - 2D.

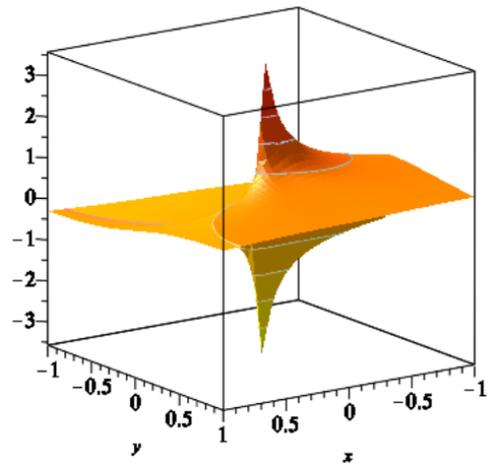


(f) σ_{xy} - 3D.

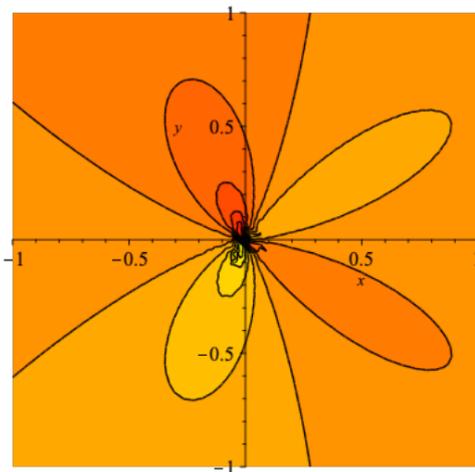
Figura 4.3 – Representação gráfica das singularidades de Williams, na ponta da trinca. Caso de trinca horizontal de borda - Modo-I puro. Tensões σ_{xx} , σ_{yy} e σ_{xy} .



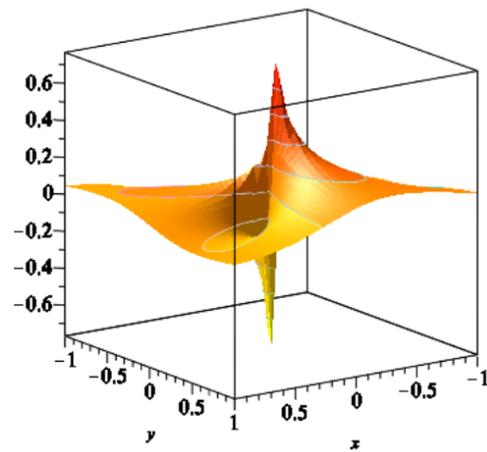
(a) σ_{xx} - 2D.



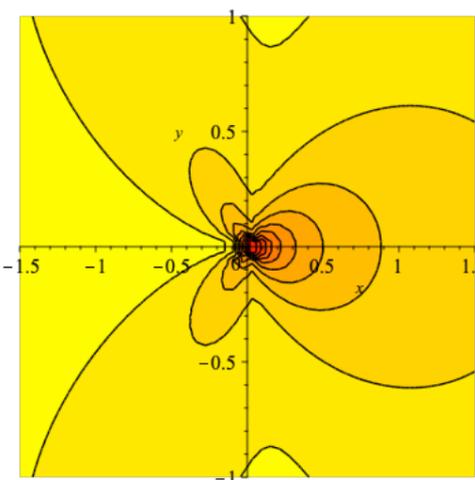
(b) σ_{xx} - 3D.



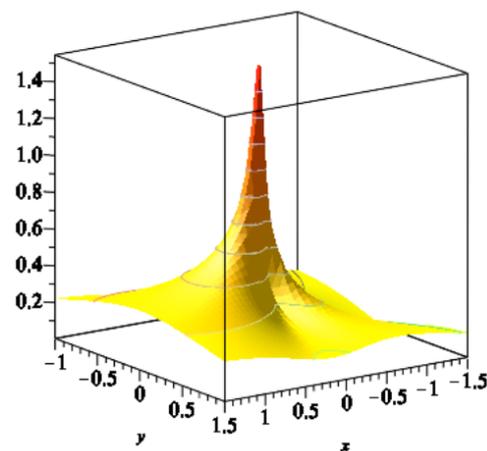
(c) σ_{yy} - 2D.



(d) σ_{yy} - 3D.



(e) σ_{xy} - 2D.



(f) σ_{xy} - 3D.

Figura 4.4 – Representação gráfica das singularidades de Williams, na ponta da trinca. Caso de trinca horizontal de borda - Modo-II puro. Tensões σ_{xx} , σ_{yy} e σ_{xy} .

adicional na expressão da tensão normal na direção do eixo da trinca. Essa tensão normal é definida por ambos os termos referentes ao carregamento, σ e $k\sigma$, como também pelo ângulo de orientação da trinca α como sendo

$$\sigma_{xx} = \sigma(1 - k) \cos 2\alpha. \quad (4.38)$$

Para $\alpha \neq \pi/4$, esse termo adicional é cancelado apenas para o caso em que $k = 1$, que corresponde a estados de tensão idênticos aplicados nos contornos remotos da placa. As implicações desse resultado são aparentemente dramáticas, tendo em vista que as mesmas invalidam o postulando fundamental da mecânica da fratura linear elástica. Efetivamente, o carregamento ou a deformação sofrida pela ponta de uma trinca interna é caracterizada pelo termo constante (4.38), adicionalmente aos termos singulares usuais (4.33) a (4.35) e conseqüentemente, os fatores de intensidade de tensão não podem determinar completamente o comportamento da trinca. Contudo, ao invocar o princípio da superposição da elasticidade linear, o postulando fundamental da mecânica da fratura linear elástica pode ser restaurado, já que o termo constante da tensão, dada pela equação (4.38), representa uma constante subjacente simples do estado de tensão em uma placa infinita sem trincas, que pode ser considerada separadamente como um problema independente, cujos resultados são impostos no estado de tensão na ponta da trinca.

Os resultados expressos pela equação (4.38) são válidos para o caso de uma trinca interna em uma placa retangular infinita, carregada por uma tensão de tração uniforme aplicada na borda da placa ou contorno da placa. No caso de uma trinca interna em uma placa de geometria arbitrária sujeita a um carregamento arbitrário, a solução elástica não pode ser derivada analiticamente e sendo assim, os termos não singulares adicionais correspondentes ao estado de tensão subjacente da placa são desconhecidos. Neste caso, uma aproximação é introduzida na análise, que considera apenas os termos singulares na representação do campo elástico na ponta da trinca. Isso significa que o campo elástico na ponta da trinca de uma trinca interna pode ser representada por um campo elástico na ponta de uma trinca de borda, apenas como uma primeira aproximação.

Este capítulo contém informações pertinentes à implementação dos conceitos relacionadas à mecânica da fratura linear elástica na formulação do campo elástico generalizado do método sem malha local (GSMF), apresentada anteriormente. Os diferentes métodos para simular a descontinuidade são apresentados juntamente com a técnica da subtração da singularidade, ambos usados na análise de crescimento de trincas.

4.2 - MÉTODO DA VISIBILIDADE, DA DIFRAÇÃO E DA TRANSPARÊNCIA

Existem basicamente quatro abordagens para se modelar descontinuidade nos métodos sem malha, sendo: (i) modificação das funções ponderadoras de forma intrínseca, como o método da visibilidade, da difração e da transparência; (ii) modificação de bases intrínsecas para incorporar funções especiais; (iii) métodos baseados no enriquecimento extrínseco dos mínimos quadrados móveis (MQM); e (iv) métodos baseados no enriquecimento extrínseco do método de partição da unidade.

O método dos mínimos quadrados móveis (MQM) usado no GSMF permite que o número de bases polinomiais seja independente do número de graus de liberdade. Dessa forma, o campo pode ser enriquecido de forma intrínseca, sem a adição de variáveis extras, o que é desejável do ponto de vista computacional. Outra vantagem do enriquecimento intrínseco é que ele elimina o problema do mau condicionamento no cálculo da matriz de rigidez.

Uma das formas mais simples de realizar esse procedimento é o método da visibilidade, apresentado por Belytschko et al. (1994a), onde o domínio local é truncado na face de uma trinca. O domínio local de um nó é restringido para as áreas do domínio visíveis pelo nó, com a face da trinca funcionando como uma barreira opaca. Se uma linha entre o nó e o ponto de interesse intersectar uma trinca, e se a ponta da trinca estiver contida no domínio local daquele nó, o nó não terá influência sobre aquele ponto; em termos matemáticos, a linha entre aquele ponto e o nó é modificada para

$$d_i = \infty. \quad (4.39)$$

Por exemplo, considere um domínio local intersectado por uma trinca, como o da Figura 4.5. O domínio local do nó x_i é truncado por essa trinca, e apenas a área mais escura está incluída

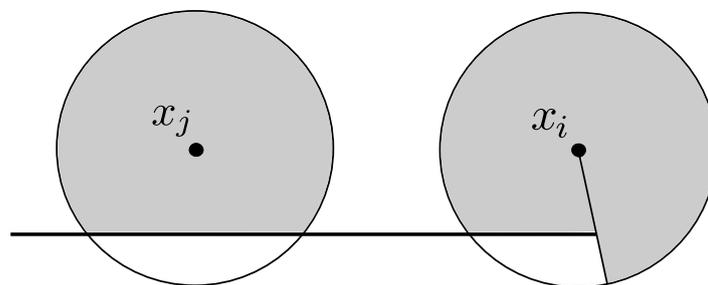


Figura 4.5 – Domínio local de um nó x_i , truncado na face de uma trinca, e um nó x_j , truncado na face de uma trinca e próximo à ponta da trinca, para o método da visibilidade.

no domínio de influência daquele nó. Aqueles nós que se encontram no lado oposto da trinca são excluídos da aproximação do campo de descolamentos. Esse truncamento irá criar uma descontinuidade na função de forma do nó x_i e assim resultar em um deslocamento

descontínuo através da trinca. Para o nó próximo à ponta da trinca, como o nó x_j , algumas dificuldades surgem, tendo em vista que uma descontinuidade adicional indesejada talvez seja produzida por esse método. Apesar disso, Krysl e Belytschko (1997) mostrou que o método, mesmo assim, resulta em soluções convergentes.

Uma abordagem alternativa é o método da difração, apresentado por Belytschko et al. (1996), onde ao invés de utilizar um truncamento total, o domínio local pode difratar ou deformar ao redor da ponta da trinca, em que a distância é definida como o caminho difratado entre o ponto de interesse e o nó. Por exemplo, como demonstrado na Figura 4.6, se a linha $r_0(x)$,

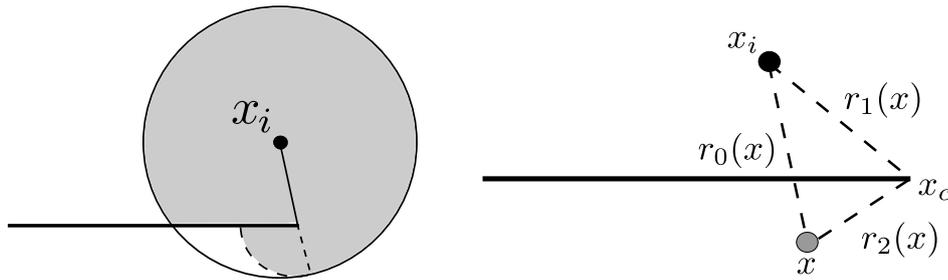


Figura 4.6 – Domínio local de um nó x_i , truncado na face de uma trinca e próximo à ponta da trinca, para o método da difração.

que junta o ponto de interesse x e o nó x_i , se intersectar com a face da trinca e a ponta da trinca estiver contida no domínio local daquele nó, então a distância $d_i(x)$, na equação (3.14), será modificada para

$$d_i = \left(\frac{r_1 + r_2(x)}{r_0(x)} \right)^\lambda r_0(x), \quad (4.40)$$

onde $r_1 = |x_i - x_c|$ é independente de x , as igualdades $r_2 = |x - x_c|$ e $r_0 = |x - x_i|$ são válidas, x_c é a coordenada da ponta da trinca e λ é o parâmetro de dilatação, que controla a distância $w_i(\mathbf{x})$ no lado oposto da trinca. Bons resultados podem ser obtidos quando $1 \leq \lambda \leq 1.5$, como pode ser visto em Belytschko et al. (1996).

Esse método é um aperfeiçoamento do método da visibilidade, retirando as descontinuidades indesejadas próximas à ponta da trinca. Apesar de ser conceitualmente simples, o método da difração se torna computacionalmente complexo para trincas não planares em 3D ou múltiplas trincas em 2D e 3D.

O método da transparência foi desenvolvido como uma alternativa para o método da difração, desenvolvido por Organ et al. (1996), criado especialmente para ser computacionalmente mais fácil de implementar em problemas tridimensionais. No método da transparência, a trinca é transparente próxima à ponta da trinca, assim como mostra a Figura 4.7, necessitando de uma condição adicional imposta nos nós próximos a trinca. Como o ângulo entre a trinca e a linha que liga o nó até a ponta da trinca é pequeno, ocorre um gradiente acentuado nas funções ponderadoras, através da linha à frente da trinca. Buscando reduzir esse efeito,

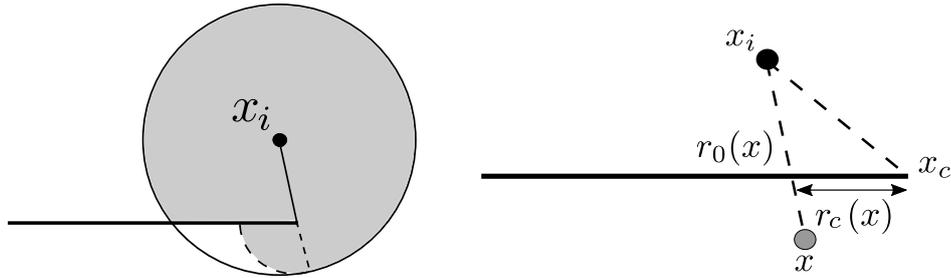


Figura 4.7 – Domínio local de um nó x_i , truncado na face de uma trinca e próximo à ponta da trinca, para o método da transparência.

Organ et al. (1996) impôs que todos os nós devem ter uma distância mínima em relação à face da trinca.

4.3 - TÉCNICA DA SUBTRAÇÃO DA SINGULARIDADE

A técnica da subtração da singularidade ou *singularity subtraction technique (SST)* é uma técnica muito eficiente e precisa na análise de placas com a presença de trincas em modo misto, que realiza o cálculo direto do fator de intensidade de tensão. A técnica da subtração da singularidade será propriamente introduzido no GSMF nessa seção, sendo implementada pela primeira vez em um método sem malha.

Uma das principais vantagens dessa técnica é que ela regulariza por completo o campo elástico, por meio da subtração das singularidades existentes na ponta de todas as trincas, antes que a análise numérica seja realizada, o que adiciona o fator de intensidade de tensão como uma variável primária do modelo numérico. Como essa formulação elimina a singularidade da ponta da trinca do problema, a análise numérica usando o GSMF é realizada apenas no campo regularizado, o que resulta em soluções altamente precisas e sem problemas de convergência.

4.3.1 - Campo Elástico Original

Seja Ω um domínio bidimensional com uma trinca plana e com contorno definido por $\Gamma = \Gamma_u \cup \Gamma_t$. Considera-se um campo elástico na ausência de forças de corpo, que satisfaça a condição de equilíbrio

$$\mathbf{L}^T \boldsymbol{\sigma} = \mathbf{0}, \quad (4.41)$$

$$\boldsymbol{\varepsilon} = \mathbf{L} \mathbf{u}, \quad (4.42)$$

e

$$\boldsymbol{\sigma} = \mathbf{D} \boldsymbol{\varepsilon}, \quad (4.43)$$

válido no domínio Ω , com as condições de contorno

$$\mathbf{u} = \bar{\mathbf{u}}, \quad (4.44)$$

no contorno cinemático Γ_u e

$$\mathbf{t} = \mathbf{n} \boldsymbol{\sigma} = \bar{\mathbf{t}}, \quad (4.45)$$

no contorno estático Γ_t ; onde o vetor $\boldsymbol{\sigma}$ representa as componentes de tensão, \mathbf{L} é o operador de derivadas matricial, o vetor $\boldsymbol{\varepsilon}$ representa o campo de deformações, \mathbf{D} é a matriz contendo as constantes elásticas, o vetor \mathbf{u} representa as componentes dos deslocamentos, $\bar{\mathbf{u}}$ os valores dos deslocamentos prescritos, o vetor \mathbf{t} representa as componentes das forças superficiais, $\bar{\mathbf{t}}$ são valores das forças superficiais prescritas e \mathbf{n} é o vetor unitário normal ao contorno Γ .

4.3.2 - Campo Elástico Regularizado

Em virtude das dificuldades que surgem em modelar numericamente problemas elastoestáticos com campos singulares, cujo campo de tensões é singular nos arredores da ponta da trinca, é conveniente subtrair essa singularidade do problema original, antes mesmo de realizar a análise com o método sem malha local.

Considere, então o caso de uma placa qualquer com m pontas de trinca e que esse campo elástico possui comportamento linear, onde o princípio da superposição é válido, e com isso podemos decompor esse campo em uma parte regular (R) e outra singular (S), como pode ser visto em

$$\sigma_{ij} = (\sigma_{ij} - \sigma_{ij}^{S_1} - \dots - \sigma_{ij}^{S_m}) + (\sigma_{ij}^{S_1} + \dots + \sigma_{ij}^{S_m}) = \sigma_{ij}^R + \sigma_{ij}^S \quad (4.46)$$

e

$$u_i = (u_i - u_i^{S_1} - \dots - u_i^{S_m}) + (u_i^{S_1} + \dots + u_i^{S_m}) = u_i^R + u_i^S, \quad (4.47)$$

onde $\sigma_{ij}^R = \sigma_{ij} - \sigma_{ij}^{S_1} - \dots - \sigma_{ij}^{S_m}$ e $u_i^R = u_i - u_i^{S_1} - \dots - u_i^{S_m}$ são as componentes regulares, respectivamente, do campo de tensões e de deformações do problema original; $\sigma_{ij}^S = \sigma_{ij}^{S_1} + \dots + \sigma_{ij}^{S_m}$ e $u_i^S = u_i^{S_1} + \dots + u_i^{S_m}$ são, respectivamente, as componentes do campo de tensões e deformações da solução particular do problema original, representando o campo elástico singular. Se as funções apropriadas forem escolhidas para esse campo singular particular, então, as equações (4.46) e (4.47) regularizam completamente o problema original, tendo em vista que as componentes de tensão σ_{ij}^R são não singulares agora. A análise do problema elástico agora pode ser realizada apenas no campo regular, representada pelas componentes

σ_{ij}^R e u_i^R , uma consequência do processo de regularização. As componentes σ_{ij}^S e u_i^S do campo singular automaticamente satisfazem as equações de campo, porque elas são definidas como a solução particular do problema original. Dessa forma, as equações elásticas (4.41) a (4.43) podem ser escritas como

$$\mathbf{L}^T \boldsymbol{\sigma}^R = \mathbf{0}, \quad (4.48)$$

$$\boldsymbol{\varepsilon}^R = \mathbf{L} \mathbf{u}^R \quad (4.49)$$

e

$$\boldsymbol{\sigma}^R = \mathbf{D} \boldsymbol{\varepsilon}^R, \quad (4.50)$$

no domínio Ω , com condições de contorno

$$\mathbf{u}^R = \bar{\mathbf{u}} - \mathbf{u}^S \quad (4.51)$$

no contorno cinemático Γ_u e

$$\mathbf{t}^R = \bar{\mathbf{t}} - \mathbf{t}^S \quad (4.52)$$

no contorno estático Γ_t .

É importante observar que esse campo elástico regularizado é governado pelas mesmas equações do campo original, exceto pela condição de contorno (4.51) e (4.52), onde os termos adicionais \mathbf{u}^S e \mathbf{t}^S , estão agora, respectivamente, inclusos. Esses termos adicionais, componentes de uma solução particular do problema original, representam o campo elástico singular.

4.3.3 - Solução Particular Singular de Williams

A solução particular utilizada nas equações (4.46) e (4.47), definidas pelas componentes σ_{ij}^S e u_i^S , representam o campo elástico singular nos arredores da ponta da trinca. O primeiro termo da série de expansão de William, para um entalhe semi-infinito (*semi-infinite edge crack*), é considerado. As componentes de tensão são dadas pelas equações (4.33) a (4.35) e os deslocamentos associados são dados pelas equações (4.36) e (4.37). O sistema de coordenadas polares (r, θ) é centralizado na ponta da trinca, de forma que $\theta = 0$ é o eixo da trinca, à frente da ponta da trinca, como apresentado na Figura 4.2. Note que o campo de tensões é da ordem de $r^{-1/2}$, que se torna singular à medida que r tende a zero. Vale ressaltar, também, que o campo de deslocamentos não inclui termos de corpo rígido, dessa forma, resultando em componentes nulas na ponta da trinca.

Caicedo e Portela (2015) demonstraram que o primeiro termo da série de expansão de

Williams, a partir da ponta de um entalhe, também pode ser utilizado para representar o campo elástico ao redor da ponta da trinca, onde a tensão singular é dominante, para o caso de múltiplas trincas internas planas em uma placa infinita, sob deformação em modo misto. Dessa forma, a estratégia de modelagem da técnica da subtração da singularidade, que é baseada na superposição do campo elástico de Williams em cada ponta de trinca, pode ser usada para lidar com várias trincas simultaneamente.

O campo de tensões singular, definido pelas equações (4.33) a (4.35), é usado para definir as componentes das forças de superfície no contorno como

$$\mathbf{t}^S = \begin{bmatrix} t_1^S \\ t_2^S \end{bmatrix} = \begin{bmatrix} \sigma_{11}^S & \sigma_{21}^S \\ \sigma_{12}^S & \sigma_{22}^S \end{bmatrix} \begin{bmatrix} n_1 \\ n_2 \end{bmatrix} = \begin{bmatrix} g_{11} & g_{21} \\ g_{12} & g_{22} \end{bmatrix} \begin{bmatrix} K_I \\ K_{II} \end{bmatrix} = \mathbf{g} \mathbf{k}, \quad (4.53)$$

onde n_i denota a componente do vetor unitário normal ao contorno e as funções $g_{ij} = g_{ij}(r^{-1/2}, \theta)$ são introduzidas na equação como uma notação conveniente advinda das equações (4.33) a (4.35) e o vetor \mathbf{k} contém o fator de intensidade de tensão. Para o caso de uma placa com m pontas de trinca, com a equação (4.53) válida para cada ponta de trinca, o princípio da superposição resulta em

$$\mathbf{t}^S = \mathbf{g}_1 \mathbf{k}_1 + \cdots + \mathbf{g}_m \mathbf{k}_m. \quad (4.54)$$

De forma similar, o campo de deslocamentos, equações (4.36) e (4.37), pode ser definido como um vetor

$$\mathbf{u}^S = \begin{bmatrix} u_1^S \\ u_2^S \end{bmatrix} = \begin{bmatrix} f_{11} & f_{21} \\ f_{12} & f_{22} \end{bmatrix} \begin{bmatrix} K_I \\ K_{II} \end{bmatrix} = \mathbf{f} \mathbf{k}, \quad (4.55)$$

onde as funções $f_{ij} = f_{ij}(r^{1/2}, \theta)$ são introduzidas na equação como uma notação conveniente das equações (4.36) e (4.37). Para o caso de uma placa com m pontas de trinca, com a equação (4.55) válida para cada ponta de trinca, a componente singular é dada por

$$\mathbf{u}^S = \mathbf{f}_1 \mathbf{k}_1 + \cdots + \mathbf{f}_m \mathbf{k}_m. \quad (4.56)$$

4.3.4 - Análise com a Formulação do Campo Elástico Generalizado (GSMF)

Tendo em vista que a solução formal do modelo elástico regularizado, definida pelas equações de campo (4.48) a (4.50), com condições de contorno (4.51) e (4.52), não pode ser obtida para problemas práticos, uma discretização deve ser utilizada para se obter a solução aproximada.

A análise utilizando a formulação do campo elástico generalizado (GSMF) do método sem malha local é realizada em termo das variáveis regularizadas u_i^R . Temos que a formulação do campo elástico generalizado, dado pela equação (3.46), na ausência de forças de corpo, pode ser escrita como

$$\frac{L_i}{n_i} \sum_{l=1}^{n_i} \mathbf{t}_{\mathbf{x}_l}^R = -\frac{L_t}{n_t} \sum_{j=1}^{n_t} (\bar{\mathbf{t}} - \mathbf{t}^S)_{\mathbf{x}_j}. \quad (4.57)$$

A equação (3.46) e conseqüentemente, a equação (4.57), demonstram o equilíbrio das forças de superfície e as forças de corpo, pontualmente definidas nos pontos de colocação, obtidas no contorno do domínio local, onde a equação dos resíduos ponderados está definida.

Discretiza-se a equação (4.57), introduzindo-se as equações do campo elástico, dada pelas equações (3.16) a (3.22), gerando um sistema linear de três equações referentes ao nó $Q \in \Omega_Q$ em função das incógnitas nodais $\hat{\mathbf{u}}$, resultando em

$$\frac{L_i}{n_i} \sum_{l=1}^{n_i} \mathbf{n}_{\mathbf{x}_l} \mathbf{D} \mathbf{B}_{\mathbf{x}_l} \hat{\mathbf{u}} - \frac{L_t}{n_t} \sum_{j=1}^{n_t} \mathbf{t}_{\mathbf{x}_j}^S = -\frac{L_t}{n_t} \sum_{j=1}^{n_t} \bar{\mathbf{t}}_{\mathbf{x}_j}, \quad (4.58)$$

que pode ser reescrito como

$$\mathbf{K}_Q \hat{\mathbf{u}}^R + \mathbf{G}_{Q1} \mathbf{k}_1 + \cdots + \mathbf{G}_{Qm} \mathbf{k}_m = \mathbf{P}_Q, \quad (4.59)$$

onde \mathbf{K}_Q é a matriz de rigidez nodal do domínio local Ω_Q , uma matriz $2 \times 2n$, onde n é o número de nós incluídos no domínio de influência do nó Q , dada por

$$\mathbf{K}_Q = \frac{L_i}{n_i} \sum_{l=1}^{n_i} \mathbf{n}_{\mathbf{x}_l} \mathbf{D} \mathbf{B}_{\mathbf{x}_l}, \quad (4.60)$$

\mathbf{G}_{Qm} , com m pontas de trinca, é a respectiva matriz nodal $2n \times 2$ contendo as funções de Williams e \mathbf{k}_m é o vetor contendo os fatores de intensidade de tensão K_I e K_{II} , dados por

$$\mathbf{G}_{Qm} \mathbf{k}_m = -\frac{L_t}{n_t} \sum_{j=1}^{n_t} \mathbf{t}_{\mathbf{x}_j}^S = -\frac{L_t}{n_t} \sum_{j=1}^{n_t} \begin{bmatrix} g_{11} & g_{21} \\ g_{12} & g_{22} \end{bmatrix} \begin{bmatrix} K_I \\ K_{II} \end{bmatrix} \quad (4.61)$$

e \mathbf{P}_Q , é o respectivo vetor de forças dado por

$$\mathbf{P}_Q = -\frac{L_t}{n_t} \sum_{j=1}^{n_t} \bar{\mathbf{t}}_{\mathbf{x}_j}. \quad (4.62)$$

Vale ressaltar, para o caso dos nós interiores, que as matrizes \mathbf{G}_{Qm} e o vetor \mathbf{P}_Q são nulos.

Para os $N - M$ nós no contorno cinemático, uma interpolação direta pode ser utilizada para impor as condições de contorno como

$$\mathbf{u}_Q^R = \Phi_Q \hat{\mathbf{u}} = \bar{\mathbf{u}}_Q - \mathbf{u}_Q^S = \bar{\mathbf{u}}_Q - \frac{L_t}{n_t} \sum_{j=1}^{n_t} \begin{bmatrix} f_{11} & f_{21} \\ f_{12} & f_{22} \end{bmatrix} \begin{bmatrix} K_I \\ K_{II} \end{bmatrix}, \quad (4.63)$$

que pode ser reescrito como

$$\mathbf{K}'_Q \hat{\mathbf{u}} + \mathbf{F}_{Q1} \mathbf{k}_1 + \dots + \mathbf{F}_{Qm} \mathbf{k}_m = \mathbf{U}_Q, \quad (4.64)$$

onde \mathbf{K}'_Q corresponde a matriz nodal contendo Φ_Q , \mathbf{F}_Q é a matriz contendo as funções de Williams e \mathbf{U}_Q é o vetor contendo os deslocamentos nodais prescritos. A equação (4.59) é aplicada aos M domínios internos e contornos estáticos, gerando um sistema

$$\mathbf{K} \hat{\mathbf{u}} + \mathbf{G} \mathbf{k} = \mathbf{P}, \quad (4.65)$$

enquanto a equação (4.64) é aplicada aos M contornos cinemáticos, gerando um sistema

$$\mathbf{K}' \hat{\mathbf{u}} + \mathbf{F} \mathbf{k} = \mathbf{U}, \quad (4.66)$$

que juntos formam o sistema global $2(M + 2) \times 2N$ de equações

$$\begin{bmatrix} \mathbf{K} & \mathbf{G} \\ \mathbf{K}' & \mathbf{F} \end{bmatrix} \begin{bmatrix} \hat{\mathbf{u}} \\ \mathbf{k} \end{bmatrix} = \begin{bmatrix} \mathbf{P} \\ \mathbf{U} \end{bmatrix}. \quad (4.67)$$

Destaca-se que esse sistema global de equações adiciona duas novas variáveis, sendo elas os fatores de intensidade de tensão K_I e K_{II} , representadas pelo vetor $\mathbf{k}_1 \dots \mathbf{k}_m$, para cada ponta de trinca. Consequentemente, para que se tenha um problema bem condicionado com uma solução única, devemos especificar duas restrições (*constraints*) adicionais, necessitando-se de uma para cada modo de deformação incluído na análise.

4.3.5 - Equações Adicionais

Restrições adicionais podem ser especificadas de várias formas, embora elas tenham que resultar no cancelamento da singularidade no campo regularizado, introduzido nas equações (4.46) e (4.47). A forma mais simples de se garantir essa condição é considerar que o campo de tensões é nulo na ponta da trinca, ou seja

$$\sigma_{ij}^R = \mathbf{0} \rightarrow \sigma_{ij} = \sigma_{ij}^S, \quad (4.68)$$

que garante que o campo elástico original é singular na ponta da trinca.

Dessa forma, como o problema é formulado em termos das componentes do deslocamento regularizado u_i^R , as restrições adicionais devem ser definidas em termos das componentes nodais incluídas no vetor de variáveis $\hat{\mathbf{u}}$, para que sejam efetivas. De certa maneira, a condição (4.68) não é definida em termos dessas componentes, u_i^R e $\hat{\mathbf{u}}$, e sendo assim, não pode ser utilizada simplesmente na forma como está, necessitando de tratamento para satisfazer de forma efetiva as restrições exigidas pela equação (4.67). Para superar essa dificuldade, a condição (4.68) primeiramente deve ser reescrita em termos das respectivas componentes das forças de superfície, definidas na ponta da trinca como

$$\mathbf{t}_t^R = \mathbf{n}\boldsymbol{\sigma} = \mathbf{0} \rightarrow \mathbf{t} = \mathbf{t}^S, \quad (4.69)$$

onde o vetor \mathbf{n} representa as componentes unitárias normais as faces da trinca, assim como esta esquematicamente representado na Figura 4.8. Essa equação resulta em

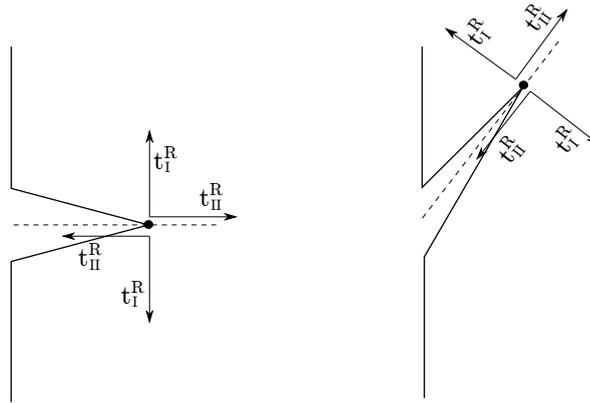


Figura 4.8 – Componentes das forças de superfície de um campo elástico regularizado na ponta de uma trinca.

$$\mathbf{t}_t^R = (\mathbf{nDB})_t \hat{\mathbf{u}}^R = \mathbf{0}, \quad (4.70)$$

onde

$$\mathbf{t}_t = (\mathbf{nDB})_1 \cdots (\mathbf{nDB})_m. \quad (4.71)$$

Com a introdução das restrições adicionais (4.70), para m pontas de trinca, o sistema de equações (4.67) resulta no sistema global $(2M + 2m) \times (2N + 2m)$

$$\left[\begin{array}{c|c} \mathbf{K} & \mathbf{G} \\ \mathbf{K}' & \mathbf{F} \\ \hline \mathbf{t}_t & \mathbf{0} \end{array} \right] \left[\begin{array}{c} \hat{\mathbf{u}} \\ \mathbf{k} \end{array} \right] = \left[\begin{array}{c} \mathbf{P} \\ \mathbf{U} \\ \mathbf{0} \end{array} \right], \quad (4.72)$$

que agora pode ser resolvido.

5 - FUNÇÕES OBJETIVO PARA MODELOS DO MÉTODO SEM MALHA LOCAL

A definição da função objetivo apropriada tem um grande impacto na performance geral do processo de otimização. Este capítulo apresenta as funções objetivo consideradas nessa pesquisa, para a otimização dos parâmetros de discretização α_s e α_q do ILMF.

5.1 - ERRO RELATIVO

A estimativa do erro, na norma de deslocamento, pode ser usada como uma função objetivo, assim definida

$$\|\mathbf{u}\| = \left[\int_{\Omega} \mathbf{u}^T \mathbf{u} \, d\Omega \right]^{1/2} \quad (5.1)$$

e, para o campo de deformações

$$\|\boldsymbol{\varepsilon}\| = \left(\frac{1}{2} \int_{\Omega} \boldsymbol{\varepsilon}^T \mathbf{D} \boldsymbol{\varepsilon} \, d\Omega \right)^{1/2}. \quad (5.2)$$

O erro relativo para $\|\mathbf{u}\|$ é dado por

$$r_u = \frac{\|\mathbf{u}_{num} - \mathbf{u}_{exact}\|}{\|\mathbf{u}_{exact}\|} \quad (5.3)$$

e, para $\|\boldsymbol{\varepsilon}\|$

$$r_\varepsilon = \frac{\|\boldsymbol{\varepsilon}^{num} - \boldsymbol{\varepsilon}^{analítico}\|}{\|\boldsymbol{\varepsilon}^{analítico}\|}. \quad (5.4)$$

O valor mínimo de r_u ou r_ε advém do resultado mais próximo possível da solução analítica, refletindo o campo elástico exato que se estabelece no corpo do problema proposto.

5.2 - COMPLIANCE

A definição desta função objetivo resulta das características do parâmetro α_s combinado com α_q . Considerando um estado qualquer do campo elástico real do corpo, a energia de deformação U , e a energia potencial P , dos carregamentos aplicados, são dadas,

respectivamente, por

$$U = \int_{\Omega} \frac{1}{2} \boldsymbol{\sigma}^T \boldsymbol{\varepsilon} \, d\Omega \quad (5.5)$$

e

$$P = - \int_{\Gamma_t} \bar{\mathbf{t}}^T \mathbf{u} \, d\Gamma, \quad (5.6)$$

e são utilizadas no cômputo da energia potencial total T . A aplicação do teorema do trabalho ao domínio global do corpo, para o caso do campo elástico real estabelecido nele, conduz a $P = -2U$ e, portanto, $T = -U$, assim como $T = P/2$. Estes resultados implicam que o valor mínimo da energia potencial total do corpo corresponde ao valor mínimo da energia potencial P ou ao máximo valor da energia de deformação U . A avaliação da energia de deformação U do corpo, requer o cálculo do campo de tensão para todos os valores nodais, o que é computacionalmente ineficiente, uma vez que exige a avaliação das derivadas das funções de forma e pode degradar a acurácia numérica.

Todavia, para calcular a energia potencial P , é necessário avaliar o campo de deslocamentos, mas, somente para os nós de contorno estáticos, aqueles com carregamentos não nulos aplicados, o que é computacionalmente muito eficiente. Este processo é executado somente em poucos nós e não requer o cálculo das derivadas das funções de forma. Por esta razão, uma função objetivo eficiente pode ser definida como o *compliance* estrutural C , dado por

$$C = \frac{1}{2} \int_{\Gamma_t} \bar{\mathbf{t}}^T \mathbf{u} \, d\Gamma = -\frac{1}{2}P. \quad (5.7)$$

Consequentemente, o valor mínimo da energia potencial P corresponde ao valor máximo de C . Isto é, maximizar a função dada pela equação (5.7) implica na mínima energia potencial total do corpo. Por questões práticas de implementação computacional, a otimização dos modelos estudados nessa pesquisa executa a minimização das funções objetivo. Assim sendo, matematicamente, maximizar C ($-1/2P$) equivale a minimizar $-C$ ($1/2P$). Pode ser visto na Figura 5.1 que se um ponto x^* corresponde ao valor mínimo da função $f(x)$, o mesmo ponto também corresponde ao valor máximo do negativo desta função, $-f(x)$. Então, sem perda de generalidade, otimização pode ser entendida como significando minimização, uma vez que o máximo de uma função pode ser encontrado buscando o mínimo do negativo da mesma função (Rao, 2019).

5.3 - FATOR DE INTENSIDADE DE TENSÃO (SIF)

A característica principal da Técnica de Subtração da Singularidade (SST), apresentada por Oliveira e Portela (2019), é a completa regularização do campo elástico, obtida através da

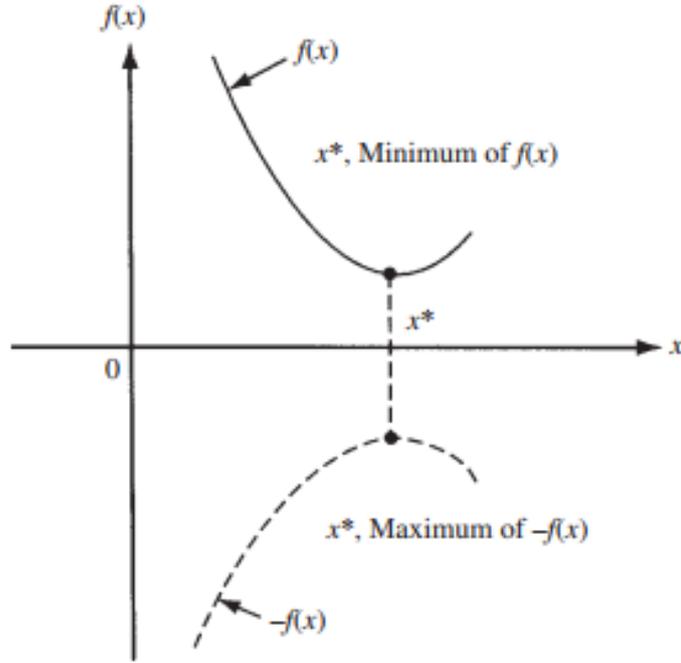


Figura 5.1 – O mínimo de $f(x)$ é o mesmo que o máximo de $-f(x)$ (Rao, 2019).

subtração das singularidades de todas as pontas de trincas, antes que a solução numérica do problema dado seja executada, o que introduz os fatores de intensidade de tensão (SIF) como variáveis primárias adicionais do modelo numérico. A formulação remove completamente as singularidades na ponta de trinca, do modelo, o que permite uma análise numérica sem malha ser executada no campo elástico regularizado, com soluções suaves e altamente precisas.

A definição dessa função objetivo resolve a questão de acurácia do modelo numérico ILMF com relação aos problemas da mecânica da fratura linear elástica. O fator de intensidade de tensão (SIF) introduzido na SST, efetivamente representa o campo singular em volta de cada ponta de trinca, o que pode ser definido através do primeiro termo da *eigenexpansion* de Williams (1952b). Portanto, os SIFs dos modos de abertura e deslizamento são respectivamente

$$k_I = \frac{K_I}{\mathbf{t}\sqrt{\pi a}}, \quad (5.8)$$

e

$$k_{II} = \frac{K_{II}}{\mathbf{t}\sqrt{\pi a}}. \quad (5.9)$$

A função objetivo pode ser definida como

$$\phi_{k_I} = \frac{k_{n_I}}{k_{a_I}}, \quad (5.10)$$

para o Modo I de fratura, e

$$\phi_{k_{II}} = \frac{k_{n_{II}}}{k_{a_{II}}}, \quad (5.11)$$

para o Modo II de fratura, onde k_{n_I} e $k_{n_{II}}$ são os resultados obtidos pelo modelo ILMF, e k_{a_I} e $k_{a_{II}}$ são os resultados analíticos obtidos pela técnica do DBEM com a integral J (J-DBEM), que é um método numérico muito preciso. Como apresentado por Portela e Aliabadi (1993), a estratégia de modelagem DBEM considera trincas segmentadas retas que são discretizadas com elementos de contorno retos quadráticos descontínuos. Elementos de contorno quadráticos contínuos são usados para os demais contornos do problema, exceto na intersecção entre uma trinca e uma borda, onde elementos de contorno semi-descontínuos são usados na borda. Elementos de contorno descontínuos *self-point* são integrados analiticamente, enquanto quadratura Gaussiana, com integração subelemento, é executada para as demais integrações. Também, valores de referência introduzidos por Portela e Aliabadi (1993), estão em consonância com aqueles obtidos por Civelek e Erdogan (1982b); Murakami (1987) e Y. (1987) e, aqueles publicados na referência H. (2000).

O valores mínimos obtidos da equação (5.10) e da equação (5.11) efetivamente correspondem ao campo singular em volta da ponta da trinca e, portanto, mais precisamente, representam o problema de fratura linear elástica.

A relação entre as funções objetivo apresentadas na Seção 5.2 e nesta Seção, será ilustrada nos resultados dos problemas analisados e mostrará que os resultados analíticos não serão mais necessários para se obter campos singulares precisos na trinca, já que uma excelente precisão pode ser obtida dependendo apenas do *compliance*, que por sua vez não requer nenhuma referência.

5.4 - COEFICIENTE DE CORRELAÇÃO DE KARL PEARSON

Conforme demonstrado por Santana et al. (2020) e nos resultados dessa pesquisa, a conformidade estrutural C substitui as funções de erro (para os campos de deslocamentos ou deformações), quando da definição das funções objetivo; dispensando assim o conhecimento das soluções analíticas dos modelos a serem resolvidos numericamente ou otimizados com o ILMF. Isso deixa o ILMF mais robusto e capaz de solucionar problemas com domínios de geometria qualquer. Todavia, para realçar essa relação direta e linear do *compliance* com os erros já mencionados, será aprofundada nesse trabalho essa investigação. Nesse sentido, é relevante uma demonstração matemática/estatística dessa relação.

Para isso, será utilizado o coeficiente de correlação de Karl Pearson. Em estatística descritiva esse coeficiente também é chamado de "coeficiente de correlação produto-momento" ou simplesmente de " ρ de Pearson", e mede o grau da correlação (e a direção dessa correlação - se positiva ou negativa) entre duas variáveis ("Coeficiente de correlação de Pearson" 2021).

Este coeficiente, normalmente representado por ρ para uma população e r para uma amostra, assume apenas valores entre -1 e 1, inclusive, de acordo com Boslaugh (2012):

- $\rho = 1$ significa uma correlação perfeita positiva entre as duas variáveis;
- $\rho = -1$ significa uma correlação negativa perfeita entre as duas variáveis, isto é, se uma aumenta, a outra sempre diminui;
- $\rho = 0$ significa que as duas variáveis não dependem linearmente uma da outra. No entanto, pode existir uma dependência não linear. Assim, o resultado $\rho = 0$ deve ser investigado por outros métodos.

O coeficiente de correlação de Pearson é dado por

$$\rho = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2} \sqrt{\sum_{i=1}^n (y_i - \bar{y})^2}} = \frac{cov(X, Y)}{\sqrt{var(X)var(Y)}}, \quad (5.12)$$

onde n é o tamanho da amostra, x_1, x_2, \dots, x_n e y_1, y_2, \dots, y_n são os valores medidos de ambas as variáveis. Além disso,

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i, \quad (5.13)$$

e

$$\bar{y} = \frac{1}{n} \sum_{i=1}^n y_i, \quad (5.14)$$

são as médias aritméticas de ambas as variáveis.

A análise correlacional indica a relação entre duas variáveis lineares e os valores sempre serão entre +1 e -1. O sinal indica a direção, se a correlação é positiva ou negativa, e o valor da variável indica a força de correlação.

Para Mukaka (2012) o valor de ρ pode ser interpretado da seguinte forma:

- 0.9 para mais (se positivo) ou para menos (se negativo) indica uma correlação muito forte;
- 0.7 a 0.9 positivo ou negativo indica uma correlação forte;
- 0.5 a 0.7 positivo ou negativo indica uma correlação moderada;

- 0.3 a 0.5 positivo ou negativo indica uma correlação fraca;
- 0 a 0.3 positivo ou negativo indica uma correlação desprezível.

Essa interpretação é de certa forma arbitrária, pois, depende do contexto e propósitos da análise correlacional. Uma correlação com $\rho = 0.80$ pode ser muito pequena quando se está verificando uma lei física utilizando instrumentos de alta precisão, mas, pode ser considerada muito grande em ciências sociais, onde pode haver uma maior contribuição de fatores complicadores. Todavia, nesse trabalho, correlações com $\rho \geq 0.80$ serão consideradas satisfatórias quando da comparação da relação linear entre duas variáveis/séries.

A Figura 5.2 ilustra com vários casos de dispersão de x e y, como a distribuição dos valores das variáveis/séries sendo analisadas impactam no valor do coeficiente de correlação ρ .

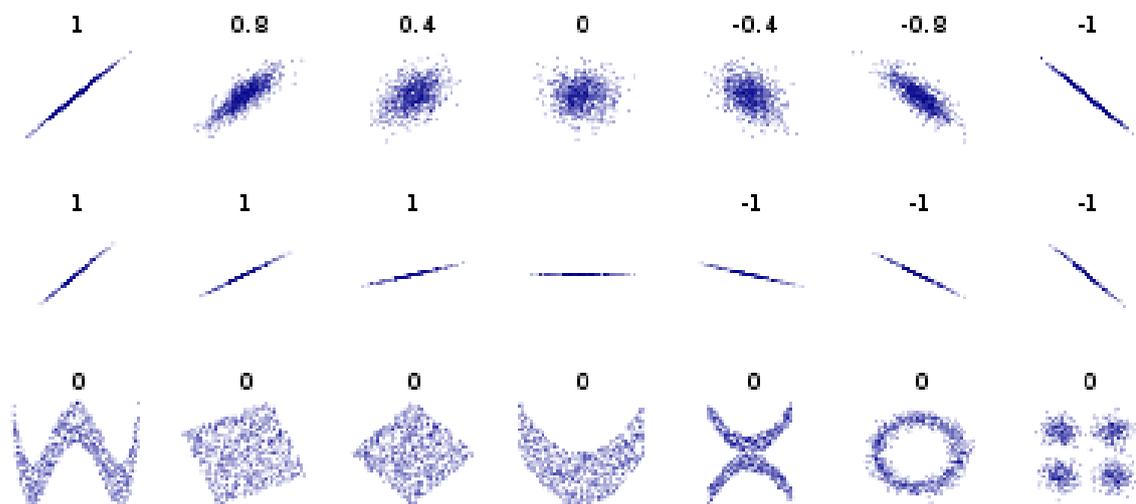


Figura 5.2 – Vários conjuntos de pontos de dispersões de x e y, com os respectivos valores de ρ ¹.

Geometricamente, as duas séries de valores $X(x_1, \dots, x_n)$ e $Y(y_1, \dots, y_n)$ podem ser consideradas como vetores em um espaço n-dimensional: $X(x_1 - \bar{x}, \dots, x_n - \bar{x})$ e $Y(y_1 - \bar{y}, \dots, y_n - \bar{y})$. O cosseno do ângulo α entre estes dois vetores é dado pelo produto escalar deles, normalizado, como

$$\cos(\alpha) = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2} \sqrt{\sum_{i=1}^n (y_i - \bar{y})^2}}, \quad (5.15)$$

¹A correlação reflete o ruído e a direção de uma relação linear (linha superior), mas, não a inclinação dessa relação (linha do meio), nem muitos aspectos das relações não lineares (linha inferior). A dispersão no centro da figura tem uma inclinação de 0, mas, nesse caso, o coeficiente de correlação é indefinido porque a variância de Y é nula. (Em Wikipedia, acessado em 13/02/2021 no link: https://en.wikipedia.org/wiki/Correlation_and_dependence).

portanto, $\cos(\alpha) = \rho$.

O coeficiente de correlação não é outro senão o cosseno do ângulo α entre os dois vetores.

Logo:

- se $\rho = 1$, o ângulo $\alpha = 0$, os dois vetores são colineares/paralelos;
- se $\rho = 0$, o ângulo $\alpha = 90^\circ$, os dois vetores são ortogonais;
- se $\rho = -1$, o ângulo $\alpha = 180^\circ$, os dois vetores são colineares, mas, com sentidos opostos.

Generalizando, $\alpha = \arccos(\rho)$.

6 - METODOLOGIA

Foram definidos quatro problemas para a aplicação da abordagem de otimização proposta para o ILMF. O primeiro, se trata de uma viga engastada, em balanço, representado a elasticidade linear. Os demais, no âmbito da mecânica da fratura linear elástica, são: placa com trinca horizontal de borda - Modo-I, placa com trinca horizontal de borda - Modo-II e placa com trinca inclinada de borda - Modo Misto.

A Tabela 6.1 resume a composição de variáveis e funções objetivo para cada problema estudado nesse trabalho. Detalhes e formulação para a definição das funções objetivo são apresentados no Capítulo 5.

Tabela 6.1 – Variáveis e funções objetivo para cada problema estudado.

Problema	Descrição	Variáveis				Funções Objetivo			
		α_s	α_q	densidade nodal da discretização	tipo da discretização (regular ou randômica)	errU	C	k_I	k_{II}
1	Viga em balanço, engastada	x	x	x	x	x	x		
2	Trinca horizontal de borda - Modo-I	x	x				x	x	
3	Trinca horizontal de borda - Modo-II	x	x				x		x
4	Trinca inclinada de borda - Modo Misto	x	x				x	x	x

Só um problema não possui solução analítica conhecida. Mas, todos são amplamente utilizados na literatura como *benchmark*. Conforme demonstrado ao longo desse texto, as soluções analíticas são prescindíveis utilizando-se a abordagem proposta. Mas, em fase de pesquisa é oportuno o uso de problemas *benchmark*.

Os *solvers* do ILMF (Santana et al. (2020), para a elasticidade linear; e, Oliveira e Portela (2019), para a mecânica da fratura), implementados em MATLAB, foram preparados para cumprirem o papel das funções objetivo. Que por sua vez, são lincadas às rotinas de otimização (em Python) e chamadas em tempo de execução, durante o processo de otimização. Esses *solvers* já estavam concluídos e aptos a resolverem modelos ILMF, aqui é feita a integração deles com a *framework* de otimização Platypus.

No processo de otimização de modelos matemáticos com algoritmos meta-heurísticos, que normalmente são estocásticos por natureza, é necessário a mensuração da performance para qualificação das respostas/soluções, uma vez que para cada batelada de iterações são obtidos resultados diferentes, mesmo que na mesma ordem de precisão e mesma vizinhança no espaço de soluções. Isso, não é um problema para algoritmos determinísticos, já que seus resultados são exatos e oriundos de funções/soluções matemáticas fechadas, mas, que infelizmente, são inviáveis para a solução da maioria dos modelos matemáticos reais onde

surtem vários complicadores, como: domínios irregulares, funções singulares, funções não convexas, entre outros. Nesse aspecto, as métricas de performance são as ferramentas ideais para a avaliação das soluções de otimização com os MMAs. A Seção 2.1.9 explora bem esse tema.

O processo de otimização proposto foi dividido em duas fases. A 1ª fase trata da geração dos resultados de referência do modelo em estudo, onde, a partir deles é medido o desempenho dos MMAs. Já na 2ª fase, os resultados obtidos com cada MMA é confrontado com os resultados obtidos na 1ª.

Na 1ª fase, os dez algoritmos meta-heurísticos da Platypus foram utilizados concomitantemente no processo de otimização, com processamento paralelo para abreviar o tempo computacional. Foram feitas 2.000 iterações (sendo 1.000 para cada *seed* do processo estocástico de ponto de partida para a população de soluções iniciais) para cada algoritmo, o que totaliza 20.000 iterações por modelo otimizado. Os resultados gerados nesta fase não foram discriminados por algoritmo, já que são para a formação da frente de Pareto de referência, e portanto, sem importância aqui tal discriminação.

Já na 2ª fase, foram utilizados os mesmos parâmetros de otimização para os algoritmos, exceto que somente 400 iterações (sendo 200 para cada *seed*) para cada MMA. E aqui, foi feita a discriminação das soluções geradas, por algoritmo. Uma vez que nessa fase, a partir das métricas, é feito um *ranking* de performance por MMA, destacando a adequação de cada um ao tipo do modelo otimizado. O número de iterações bem menor do que na 1ª fase serve para testar a capacidade dos algoritmos em estudo de gerar resultados precisos, com poucas iterações (para fins de simplificação, o "número de iterações" aqui é utilizado ao invés do "número de avaliações das funções objetivo"), para problemas complexos computacionalmente como os modelos ILMF. Além disso, cada algoritmo (MMA) tem seus respectivos resultados comparados, via métricas, aos resultados gerados por vários MMAs simultaneamente (1ª fase) e com um número de iterações muito superior.

Para as duas fases, foi utilizada uma população de 100 indivíduos para cada MMA.

A partir dos resultados da otimização, a influência do tipo da discretização nodal do domínio, se regular ou randômica, e a densidade nodal, são avaliadas para o Problema 1. Para tanto, foram desenvolvidas rotinas em *Python* para que essa discretização seja feita automaticamente dentro da otimização.

Os resultados das duas fases são mostrados em paralelo nesse texto, na medida do possível, para facilitar a compreensão e comparação dos mesmos.

Os valores das variáveis e das funções objetivo do modelo ILMF possuem escalas (ordens

de grandeza) diferentes e, portanto, para que os resultados sejam apresentados de forma clara e inteligível é necessário escalar os dados de forma unificada. Para essa tarefa, é empregado o pacote *Seaborn* (pacote em Python para a visualização de dados estatísticos). Em suas rotinas de pré-processamento há disponíveis três métodos de escalagem de dados: *MinMaxScaler*, *StandardScaler* e *RobustScaler*. O método *MinMaxScaler* será o utilizado aqui. Para mais detalhes, a documentação do pacote pode ser acessada através do link <https://seaborn.pydata.org>.

A partir da investigação e representação gráfica dos resultados das duas fases, com o auxílio de ferramentas de *data science*, são feitas várias depreensões, conforme apresentado nos resultados, Capítulo 7.

Por fim, é gerado um *ranking* global para a classificação dos MMAs em função da performance. O ranqueamento global, como nas duas fases de otimização, é feito aplicando-se uma função *utility* nas métricas de performance dos MMAs.

7 - RESULTADOS E DISCUSSÕES

Para testar as rotinas da nova abordagem de otimização proposta para os parâmetros de discretização dos métodos sem malha, um exemplo clássico foi escolhido: uma viga em balanço com uma carga concentrada na ponta, sem forças de corpo, para o caso de estado plano de tensão. As soluções analíticas encontradas na literatura servem de base de comparação para as soluções numéricas encontradas com a formulação. Esse exemplo foi apresentado em Santana et al. (2020), e, é considerado fundamental na validação das metodologias de otimização aqui empregadas. Naquela ocasião, foi empregado somente o método dos algoritmos genéticos e a *toolbox* do MATLAB.

Após essa validação, a nova abordagem será aplicada em três problemas *benchmark* da mecânica da fratura linear elástica. Sendo os dois primeiros relativos a trincas horizontais, em modo-I e modo-II, respectivamente. E o terceiro, uma trinca inclinada, em modo misto. A parte inicial de cada um desses problemas reflete fielmente o trabalho de Oliveira (2019).

7.1 - ABORDAGEM INICIAL PARA A OTIMIZAÇÃO DOS PARÂMETROS DE DISCRETIZAÇÃO DO MÉTODO SEM MALHA LOCAL

O primeiro exemplo se trata de uma viga em balanço com uma carga parabólica na ponta, sem forças de corpo atuando, ilustrada na Figura 7.1, com carga parabólica dada por

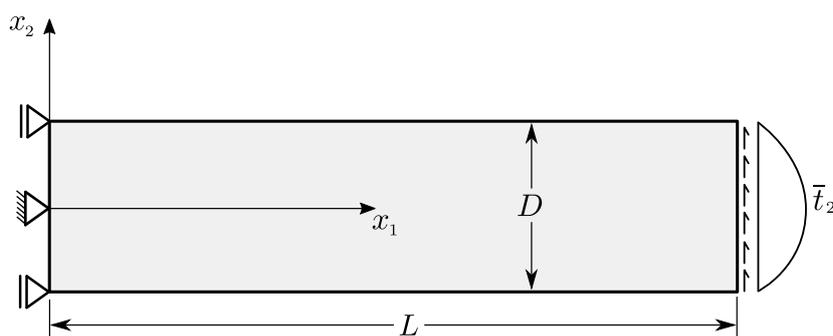


Figura 7.1 – Estado Plano de Tensão - Viga de Timoshenko engastada de Largura Unitária.

$$\bar{t}_2(x_2) = -\frac{P}{2I} \left(\frac{D^2}{4} - x_2^2 \right). \quad (7.1)$$

A solução analítica pode ser obtida utilizando-se polinômios de Airy de quarta ordem e aplicando as respectivas condições de contorno, como visto em Mase (1999). Com isso, a

solução analítica do campo de deslocamentos é dada por

$$\begin{bmatrix} u_1(x_1, x_2) \\ u_2(x_1, x_2) \end{bmatrix} = \begin{bmatrix} -\frac{Px_2}{6EI} \left[(6L - 3x_1)x_1 + (2 + \nu) \left(x_2^2 - \frac{D^2}{4} \right) \right] \\ \frac{P}{6EI} \left[3\nu x_2^2(L - x_1) + (4 + 5\nu) \frac{D^2 x_1}{4} + (3L - x_1)x_1^2 \right] \end{bmatrix}, \quad (7.2)$$

onde I é momento de inércia da seção transversal, ν é o coeficiente de Poisson, E é módulo de elasticidade longitudinal e G é módulo de elasticidade transversal. Sendo assim, a solução analítica para o campo de tensões é dado por:

$$\begin{bmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{12} \end{bmatrix} = \begin{bmatrix} -\frac{P(L - x_1)x_2}{I} \\ 0 \\ -\frac{P}{2I} \left(\frac{D^2}{4} - x_2^2 \right) \end{bmatrix}, \quad (7.3)$$

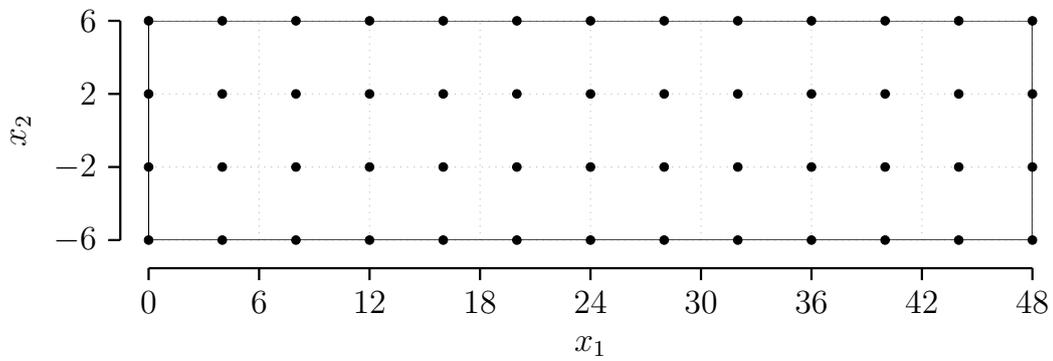
onde σ_{11} é a tensão na direção x , σ_{22} é a tensão na direção y e σ_{12} é a tensão transversal.

Para a aplicação desse exemplo foi considerada uma viga com comprimento de $L = 48$ e altura de $D = 12$, com uma carga aplicada na ponta de $P = 1000$, módulo de elasticidade longitudinal $E = 3.10^7$ e coeficiente de Poisson de $\nu = 0.3$.

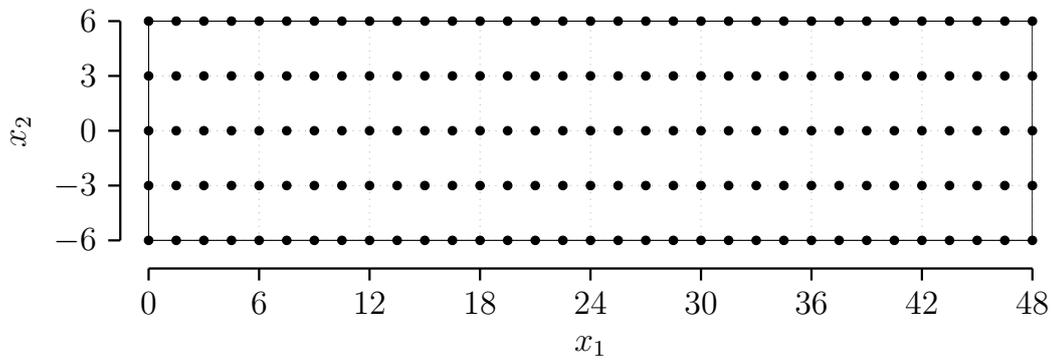
Uma distribuição regular dos nós ao longo do domínio foi escolhida, como mostra a Figura 7.2, com um total de $13 \times 4 = 52$, $33 \times 5 = 165$ e $65 \times 9 = 585$ nós. Foram considerados domínios locais retangulares, com apenas 1 ponto de colocação centralizado em cada contorno do domínio local Ω_q , para computar a integração da forma fraca local. Na aproximação pelos MQM, bases polinomiais de primeira ordem e funções ponderadoras *quartic spline* foram consideradas. Para esse exemplo, todas as rotinas foram executadas em MATLAB 2015a com um computador Intel Core I7-4700MQ com CPU de 2.4GHz e 16 GB de memória RAM.

7.1.1 - Otimização Baseada no Erro Relativo

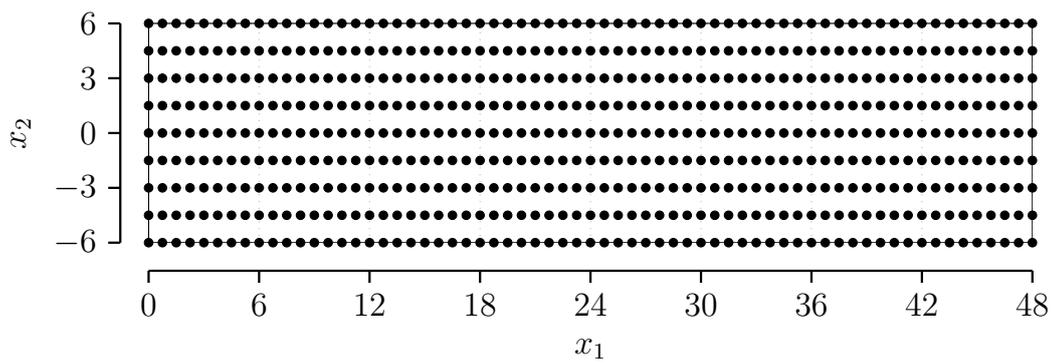
Na abordagem inicial de otimização, o algoritmo genético (AG) é implementado considerando a minimização do tempo de processamento (*CPU Time*) e os erros relativos r_ε e r_u , escolhidos como as funções objetivo desta rotina. O principal objetivo desta abordagem é avaliar o comportamento dos parâmetros adimensionais com relação à precisão e o esforço computacional. Apenas os maiores custos computacionais foram considerados, sendo estes, o custo de gerar a matriz de rigidez e resolver as equações



(a) Distribuição nodal regular com $13 \times 4 = 52$ nós.



(b) Distribuição nodal regular com $33 \times 5 = 165$ nós.



(c) Distribuição nodal regular com $65 \times 9 = 585$ nós.

Figura 7.2 – Distribuição regular dos nós de uma viga engastada em balanço, discretizada com 52, 165 e 585 nós.

algébricas. As variáveis a serem otimizadas, α_s e α_q , são definidas como contínuas e seguem o intervalo

$$\alpha_s = [1.1 \ 10] \quad e \quad \alpha_q = [0.1 \ 0.9]. \quad (7.4)$$

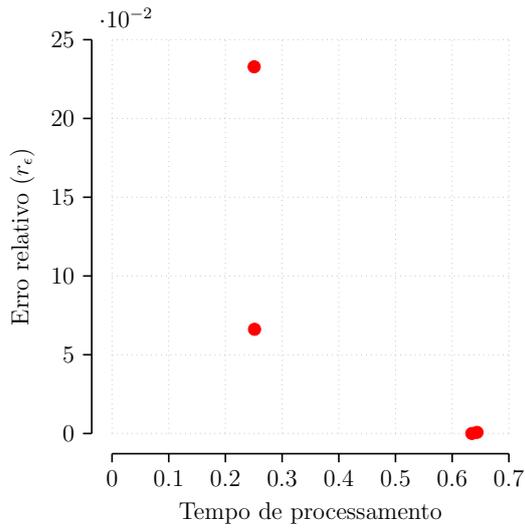
Geralmente, o parâmetro α_s é tomado como maior do que 1, para garantir que o suporte compacto terá a quantidade mínima necessária de nós para a construção das funções de forma. Neste exemplo, o limite superior é definido como 10, para reduzir o esforço computacional, limitando o espaço de busca com valores razoáveis. Além do mais, o parâmetro α_s é tomado como menor do que 1, para garantir que o domínio local interno de um nó esteja completamente dentro do domínio da solução, sem intersectar o contorno global.

As funções objetivo estão diretamente relacionadas com as variáveis a serem otimizadas. Isso significa que, ao selecionar diferentes valores para as variáveis α_s e α_q no espaço de busca amostral, diferentes resultados para o tempo de processamento (*CPU Time*), r_ε e r_u são obtidos, para cada análise com GSMF. Claramente esse é um processo de otimização com três funções objetivo e assim, uma frente de Pareto é obtida no final do processo de otimização, mas apenas duas funções são demonstradas graficamente.

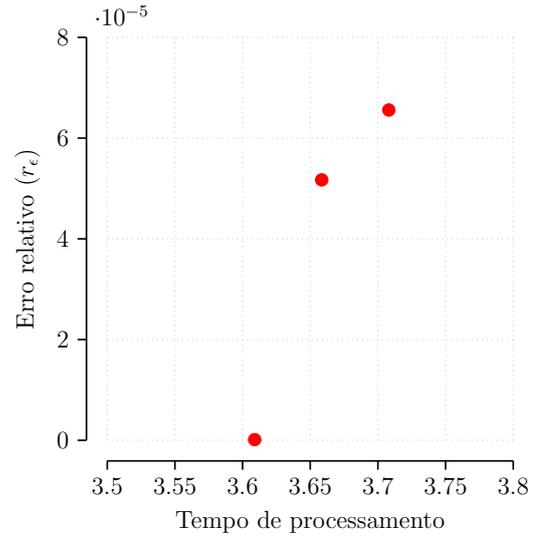
A população inicial é aleatoriamente gerada de acordo com o tamanho predefinido de 20 indivíduos. Em seguida, a função de aptidão é calculada para cada membro da população e escalonada utilizando um processo de classificação (*rank*), que será posteriormente usado no processo de seleção. O operador de reprodução é implementado baseando-se em uma seleção do tipo torneio, onde a mutação e o crossover são limitados pelas restrições impostas, equação (7.4). Esse AG descrito gera uma sequência de variáveis, que são avaliadas pelas funções objetivo escolhidas. Finalmente, o processo de otimização termina quando o número máximo de gerações é atingido ou quando a mudança média na função custo for menor que um valor determinado, nesse caso 100 e 1×10^{-6} , respectivamente.

7.1.1.1 - Frente de Pareto

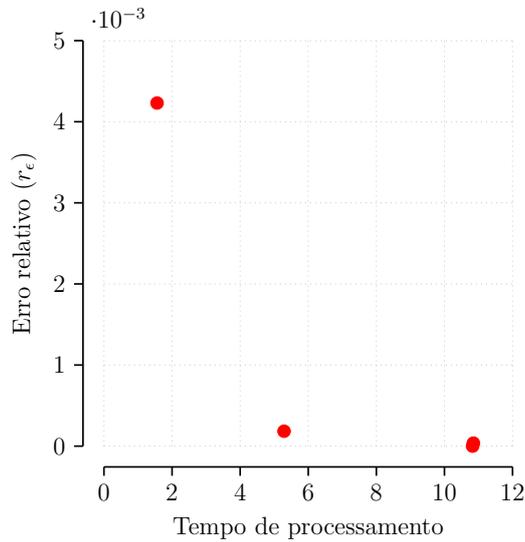
Os resultados da otimização multiobjetivo são apresentados na Figura 7.3 e nas Tabelas 7.1 a 7.3, onde todos os pontos apresentados são ótimos e não dominados. Para cada um desses pontos, um par de α_s e α_q otimizado é apresentado. Os resultados da otimização demonstram uma boa concordância com a solução analítica, resultando em erros relativos r_ε e r_u baixos para todas as configurações nodais, até mesmo em pontos onde o tempo de processamento foi mais baixo. Outro ponto que vale destacar é que a convergência é garantida, tendo em vista que um aumento na quantidade de nós na discretização sem malha, também aumenta a precisão.



(a) Distribuição nodal regular com $13 \times 4 = 52$ nós.



(b) Distribuição nodal regular com $33 \times 5 = 165$ nós.



(c) Distribuição nodal regular com $65 \times 9 = 585$ nós.

Figura 7.3 – A frente de Pareto da otimização multiobjetivo para três distribuições nodais regulares da viga engastada em balanço, em função do erro relativo r_ϵ , do tempo de processamento (*CPU Time*) em segundos e do erro relativo r_u ; apenas os dois primeiros foram apresentados graficamente.

Tabela 7.1 – Os resultados da frente de Pareto multiobjetivo para uma distribuição nodal regular com 52 nós, referentes à Figura 7.3a

Índice	Tempo de processamento (s)	r_ϵ	r_u	α_s	α_q
1	0.2513	0.06615	0.06682	1.3628	0.4930
2	0.6347	1.413E-5	4.772E-4	3.9200	0.506
3	0.6437	7.194E-4	4.211E-4	3.7666	0.5109
4	0.2507	0.2327	0.2531	1.3818	0.5067

Tabela 7.2 – Os resultados da frente de Pareto multiobjetivo para uma distribuição nodal regular com 165 nós, referentes à Figura 7.3b

Índice	Tempo de processamento (s)	r_ε	r_u	α_s	α_q
1	3.7081	6.557E-5	2.009E-7	5.2789	0.5138
2	3.6090	1.336E-7	7.075E-5	5.2775	0.5137
3	3.6585	5.171E-5	1.552E-5	5.2788	0.5139

Tabela 7.3 – Os resultados da frente de Pareto multiobjetivo para uma distribuição nodal regular com 585 nodes, referente à Figura 7.3c

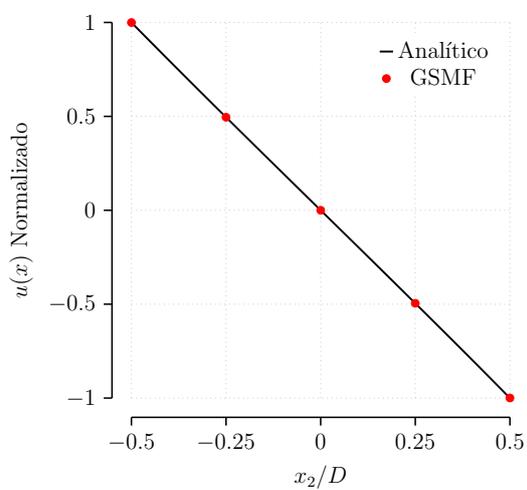
Índice	Tempo de processamento (s)	r_ε	r_u	α_s	α_q
1	10.8267	6.190E-10	0.0001	7.2685	0.4971
2	5.2903	0.0002	1.314E-07	4.2103	0.4971
3	10.8503	3.733E-05	6.978E-05	7.2686	0.4973
4	1.5595	0.0042	0.0046	1.7677	0.4995

Quanto ao comportamento dos parâmetros adimensionais, observa-se que o α_s muda constantemente para diferentes configurações nodais, geralmente aumentando à medida que mais nós são adicionados à análise. Em contrapartida, o α_q tem seu melhor resultado sempre associado a valores próximos de 0.5.

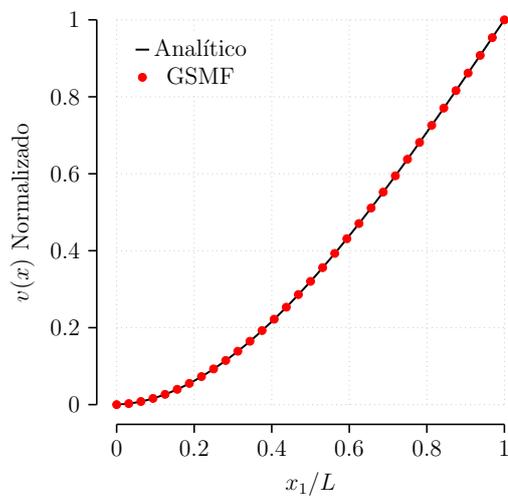
A principal desvantagem dessa abordagem de otimização é o esforço computacional extremamente alto necessário para realizar essa análise, que pode levar horas e, dependendo da discretização nodal utilizada, dias para ser realizada, mesmo quando processada em paralelo. Outra grande desvantagem é que as soluções analíticas são necessárias para o cálculo do erro relativo, e dessa forma, não estão disponíveis para problemas mais complexos. Ambas as desvantagens apresentadas são inaceitáveis e demonstram a necessidade de uma abordagem de otimização mais eficiente.

7.1.1.2 - Deslocamentos e tensões

Os deslocamentos obtidos com o GSMF, para uma distribuição nodal com 165 nós, considerando $\alpha_s = 5.2775$ e $\alpha_q = 0.5137$ obtidos pelo processo de otimização, representados na Figura 7.4, mostraram uma boa concordância com os resultados analíticos. As tensões, calculadas no centro da viga, em $x_1 = L/2$ e $x_2 \in [-D/2, D/2]$, também apresentaram bons resultados quando comparadas com as respostas analíticas, como mostra a Figura 7.5.



(a) Para $x_2 = 0$



(b) Para $x_1 = L/2$

Figura 7.4 – Deslocamento vertical e horizontal normalizados em uma viga engastada com uma discretização de $33 \times 5 = 165$ nós.

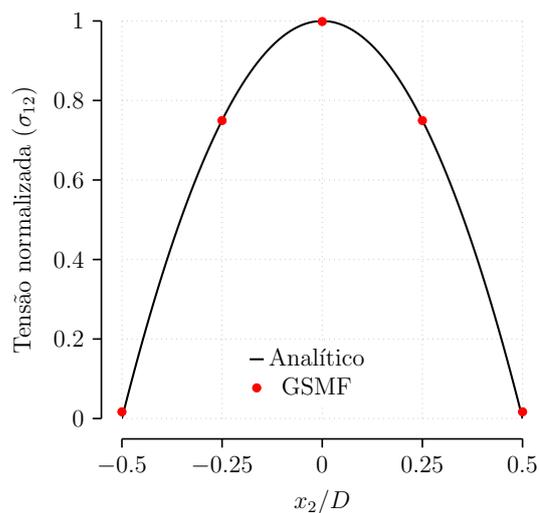
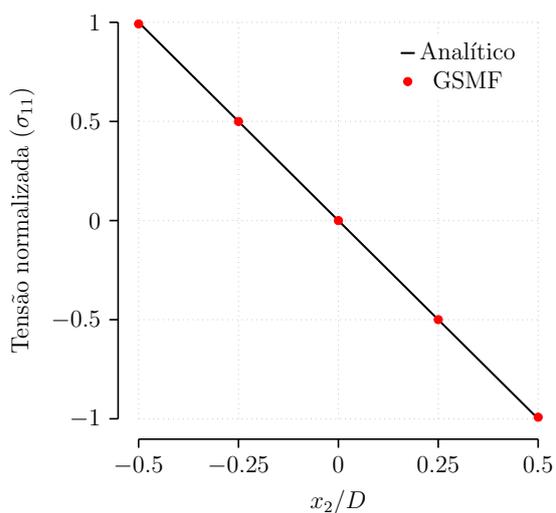


Figura 7.5 – Distribuição de tensões de uma viga engastada em $x_1 = L/2$ e $x_2 \in [-D/2, D/2]$ com uma discretização de $33 \times 5 = 165$ nós.

7.1.2 - Otimização Baseada na Conformidade Estrutural

Buscando solucionar as principais desvantagens apresentadas pela abordagem anterior, uma nova função objetivo se faz necessária. A conformidade estrutural (*Compliance*), também conhecida como flexibilidade estrutural do material, derivada do teorema do trabalho local e calculada apenas no contorno estático Γ_t , foi escolhida como a principal função objetivo dessa abordagem, juntamente com o tempo de processamento. As Figuras 7.6 a 7.8

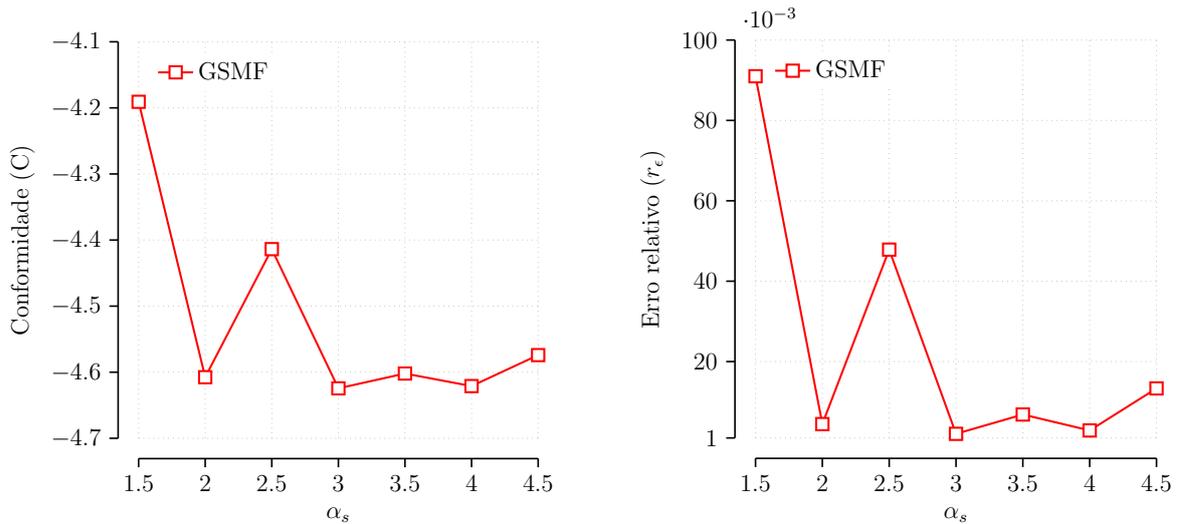


Figura 7.6 – Comparação entre o indicador de conformidade (C) e o Erro relativo r_e , para uma viga engastada em balanço com uma discretização de $13 \times 4 = 52$ nós, para um valor fixo $\alpha_q = 0.5$.

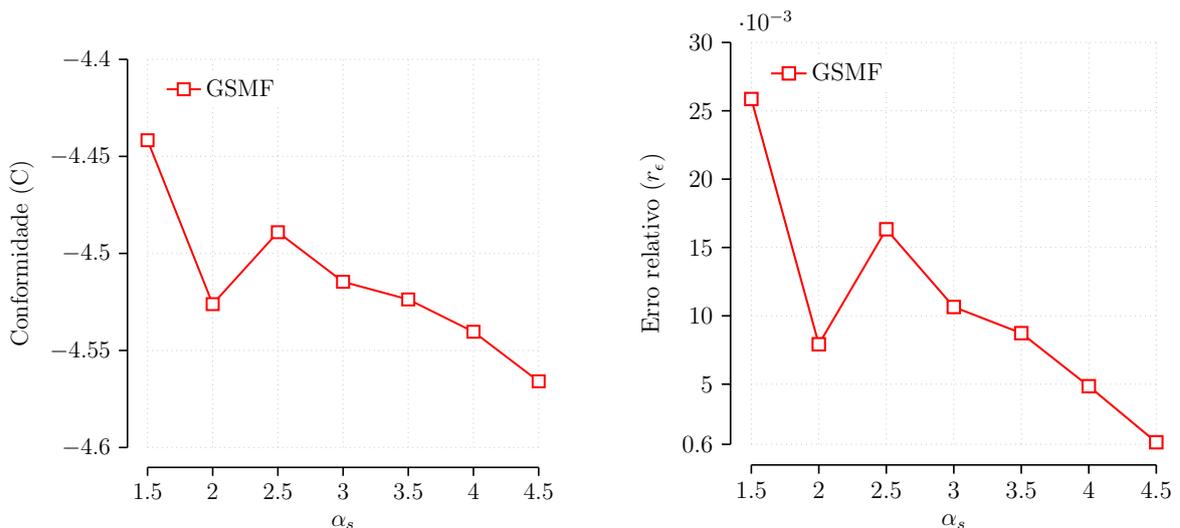


Figura 7.7 – Comparação entre o indicador de conformidade (C) e o Erro relativo r_e , para uma viga engastada em balanço com uma discretização de $33 \times 5 = 165$ nós, para um valor fixo $\alpha_q = 0.5$.

mostram o comportamento do indicador de conformidade (C) e o erro relativo r_ε , para

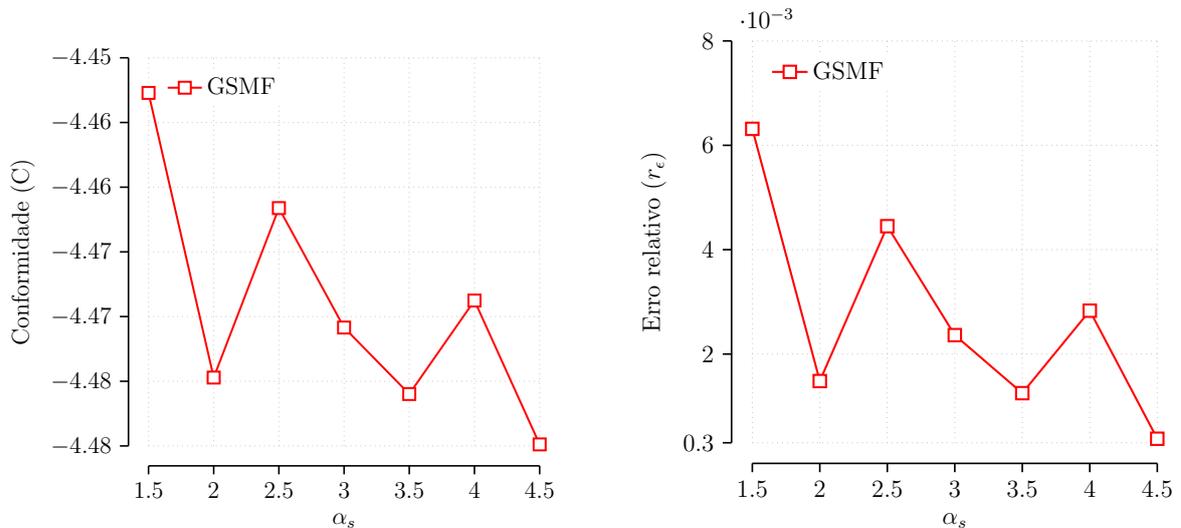


Figura 7.8 – Comparação entre o indicador de conformidade (C) e o Erro relativo r_ε , para uma viga engastada em balanço com uma discretização de $65 \times 9 = 585$ nós, para um valor fixo $\alpha_q = 0.5$.

diferentes valores de α_s , considerando $\alpha_q = 0.5$ fixo. Essa análise demonstra a similaridade entre ambos os parâmetros com relação ao α_s , e comprova a capacidade do indicador de conformidade como potencial função objetivo a ser minimizada. Uma das vantagens do indicador de conformidade (C) como função objetivo é a sua alta eficiência, que reduz o esforço computacional necessário para realizar a otimização em aproximadamente 15 vezes, se comparado com a otimização baseada no erro relativo r_ε .

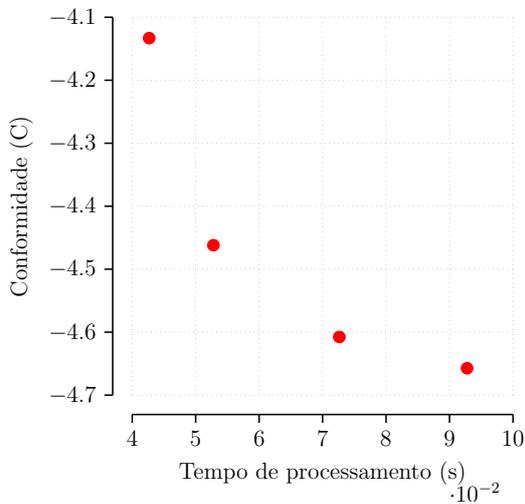
Sendo assim, o GA é implementado e preparado para minimizar o tempo de processamento e o indicador de conformidade (C), escolhidos como funções objetivo dessa abordagem de otimização. Apenas os maiores custos computacionais foram considerados, assim como na abordagem anterior. A variável α_s a ser otimizada é classificada como contínua e segue o intervalo

$$\alpha_s = [1.5 \ 10]. \quad (7.5)$$

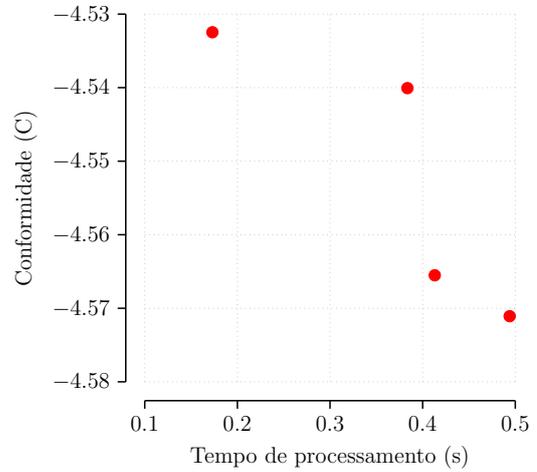
A população inicial de 20 indivíduos é aleatoriamente gerada. Em seguida, a função de aptidão é calculada para cada membro da população e "escalada" utilizando um processo de classificação (*rank*), que será posteriormente usado no processo de seleção. O operador de reprodução é implementado baseando-se em uma seleção do tipo torneio, onde a mutação e o crossover são limitados pelas restrições impostas, equação (7.5). Finalmente, o processo de otimização termina quando o número máximo de gerações de 100 é atingido ou quando a mudança média na função aptidão for menor que 1×10^{-6} .

7.1.2.1 - Frente de Pareto

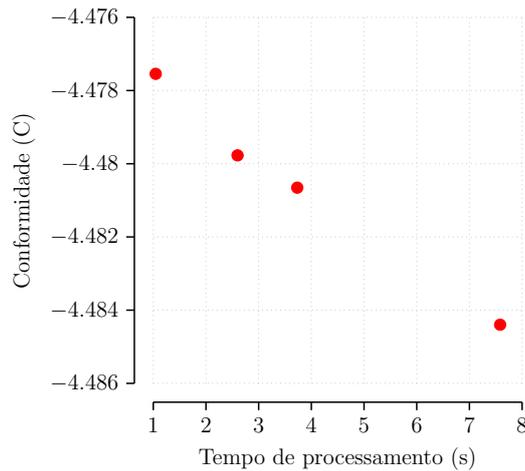
Os resultados da otimização multiobjetivo são apresentados na Figura 7.9 e nas Tabelas 7.4



(a) Distribuição nodal regular com 52 nós.



(b) Distribuição nodal regular com 165 nós.



(c) Distribuição nodal regular com 585 nós.

Figura 7.9 – A frente de Pareto da otimização multiobjetivo para três distribuições nodais regulares da viga engastada em balanço, em função do indicador de conformidade C e do tempo de processamento ($CPU\ Time$) em segundos.

a 7.6, onde todos os pontos apresentados são ótimos e não dominados.

Para cada um desses pontos, um α_s é apresentado, considerando $\alpha_q = 0.5$ fixo e os erros relativos r_ε e r_u calculados posteriormente, apenas para efeito de comparação. Pode-se observar que a presente otimização gera resultados altamente precisos em todos os pontos da frente de Pareto, diferente da abordagem anterior, em que o tempo de processamento era priorizado e os erros relativos eram ligeiramente maiores. Esse resultado demonstra que o conjunto ótimo de Pareto obtido com essa otimização é mais eficaz para essa função objetivo. Assim como na abordagem anterior, a convergência é garantida.

Tabela 7.4 – Os resultados da frente de Pareto multiobjetivo para uma distribuição nodal regular com 52 nós para $\alpha_q = 0.5$ fixo, referente à Figura 7.9a

Índice	Tempo de processamento (s)	Conformidade (C)	r_ε	r_u	α_s
1	0.0427	-4.1332	0.0993	0.1069	1.3585
2	0.0923	-4.6573	0.0051	0.0042	3.1515
3	0.0726	-4.6077	0.0044	0.0067	2.0053
4	0.0528	-4.4619	0.0346	0.0373	1.7414

Tabela 7.5 – Os resultados da frente de Pareto multiobjetivo para uma distribuição nodal regular com 165 nós para $\alpha_q = 0.5$ fixo, referente à Figura 7.9b

Índice	Tempo de processamento (s)	Conformidade (C)	r_ε	r_u	α_s
1	0.4131	-4.5655	6.660E-04	5.510E-04	4.4932
2	0.4940	-4.5711	0.0019	0.0017	4.7085
3	0.1730	-4.5325	0.0064	0.0066	1.8935
4	0.3835	-4.5401	0.0049	0.0050	3.9966

Tabela 7.6 – Os resultados da frente de Pareto multiobjetivo para uma distribuição nodal regular com 585 nós para $\alpha_q = 0.5$ fixo, referente à Figura 7.9c

Índice	Tempo de processamento (s)	Conformidade (C)	r_ε	r_u	α_s
1	2.5991	-4.4798	4.022E-04	3.821E-04	4.4965
2	3.7328	-4.4806	2.040E-04	1.883E-04	4.5330
3	1.0455	-4.4775	8.209E-04	8.768E-04	1.8733
4	7.5833	-4.4844	6.614E-04	6.352E-04	9.6232

Vale ressaltar que o menor erro relativo não é sempre obtido para o menor valor do indicador de conformidade (C), embora isso não seja necessariamente um problema, já que todos os pontos na frente de Pareto geram bons resultados. Outra vantagem dessa abordagem é que o esforço computacional necessário para realizar a análise é muito menor que o necessário pela abordagem baseada no erro relativo r_ε , levando minutos para a maioria das distribuições nodais analisadas ao invés de horas, como era anteriormente.

7.1.3 - Otimização Baseada no Domínio Local

Uma otimização completa do problema em questão requer também uma otimização do parâmetro α_q . A forma mais simples de realizar essa otimização é garantindo que a área do domínio local de todos os nós de uma discretização sem malha, onde está definida a forma fraca local, seja o mais próximo possível da área total do domínio do problema. Dessa forma, o GA foi implementado considerando o somatório das áreas dos domínios locais de cada nó, dado pela equação 5.8, como a função objetivo dessa otimização mono-objetivo. Considerando que o problema tem um total de N nós, a variável α_q será um vetor contendo N componentes e seguindo o intervalo

$$\alpha_q = [0.5 \ 0.95] , \quad (7.6)$$

para garantir que o domínio local dos nós internos está contido inteiramente dentro do domínio da solução, sem intersectar o contorno global; e também para garantir que o domínio local esteja no mínimo intersectando outros domínios locais de outros nós. Ambas as restrições são impostas para garantir uma boa precisão, tendo em vista que valores muito pequenos de α_q degeneram rapidamente a solução.

Nessa abordagem, a população inicial é aleatoriamente gerada de acordo com o tamanho predefinido de 200 indivíduos. Especificamente nessa abordagem, populações de análises anteriores são aleatoriamente inseridas nas otimizações posteriores, buscando assim melhorar a eficiência computacional em distribuições nodais maiores. Em seguida, a função aptidão é calculada para cada membro da população e "escalada" utilizando um processo de classificação (*rank*), que será posteriormente usado no processo de seleção. O operador de reprodução é implementado baseando-se em uma seleção estocasticamente uniforme, onde a mutação e o crossover são limitados pelas restrições impostas, equação (7.6). Finalmente, o processo de otimização termina quando o número máximo de gerações de 400 é atingido ou quando a mudança média na função aptidão for menor que 1×10^{-6} .

7.1.3.1 - Convergência do domínio local de colocação do GSMF

Os resultados obtidos para essa otimização mono-objetivo, considerando $\alpha_s = 4.5$ fixo e número máximo de gerações de 100, 200, 400 e 800 nos critérios de parada, são apresentados na Figura 7.10, onde observa-se que a maior precisão é obtida quando valores

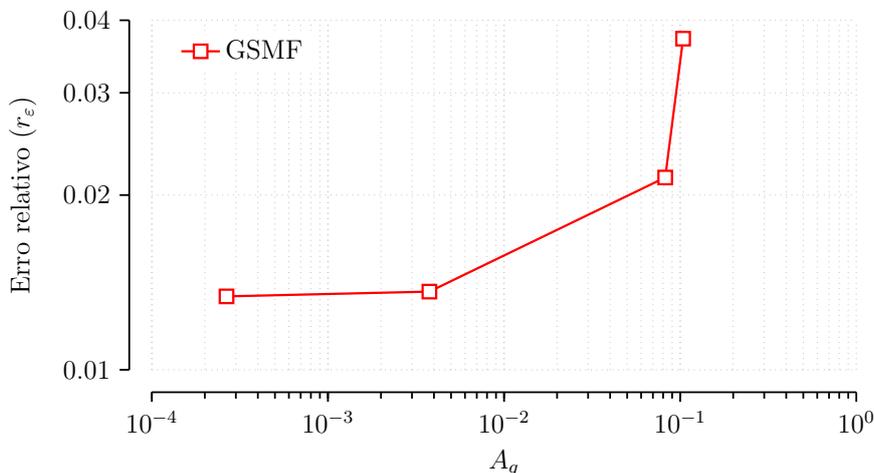


Figura 7.10 – Erro relativo r_ϵ em função da relação A_q , entre o somatório das áreas dos domínios locais dos nós na discretização sem malha com a área do domínio global do problema, para $\alpha_s = 4.5$ fixo e número máximo de gerações de 100, 200, 400 e 800; para uma distribuição nodal com 165 nós.

de A_q são menores, convergindo para a área do domínio do problema em questão. O esforço computacional reduzido necessário para realizar essa análise é uma característica chave dessa abordagem, gerando boas aproximações em questão de minutos, sendo até mesmo mais rápida que a abordagem anterior.

O esforço computacional pode ser reduzido ainda mais se forem consideradas populações aleatórias de análises anteriores. Com essa estratégia, uma primeira análise, de preferência com uma distribuição nodal reduzida, pode ser realizada de forma rápida e a população gerada salva na memória interna do MATLAB. Em seguida, quando uma distribuição nodal maior for considerada, a otimização realiza a análise mais rapidamente utilizando a população da análise anterior, acelerando ainda mais todo o processo de otimização.

Esse processo de otimização pode ser facilmente combinado com o processo de otimização anterior, para gerar uma rotina totalmente automatizada, assim como demonstrado a seguir.

7.1.4 - Otimização Automática dos Parâmetros de Discretização

A abordagem de otimização apresentada combina as características chaves das duas metodologias apresentadas anteriormente, seções 7.1.2 e 7.1.3, em uma única rotina para

calcular ambos os parâmetros adimensionais dos métodos sem malha locais, α_s e α_q , automaticamente e de forma eficiente.

A implementação do GA é dividida em duas etapas. Primeiramente, α_q é otimizado usando A_q como função objetivo, que resulta em um vetor contendo um valor escalar de α_q para cada nó que compõe a distribuição nodal em questão. Em seguida, o α_s é otimizado usando a conformidade (C) e o tempo de processamento como funções objetivo, considerando os resultados obtidos na análise anterior. É importante ressaltar que as mesmas opções e parâmetros das análises anteriores foram utilizadas nessa abordagem.

7.1.4.1 - Frente de Pareto

Os resultados obtidos com essa otimização multiobjetivo dos parâmetros do GSMF são apresentados na Figura 7.11 e nas Tabelas 7.7 a 7.9, no qual todos os pontos apresentados

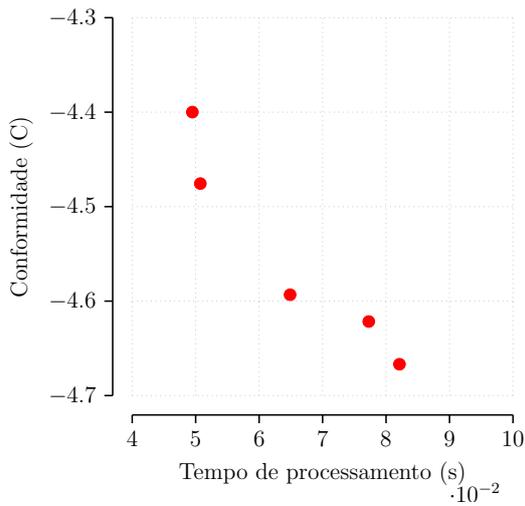
Tabela 7.7 – Os resultados da frente de Pareto multiobjetivo da rotina totalmente automatizada, para uma distribuição nodal regular com 52 nós e diferentes valores de α_q para cada nó da discretização sem malha, referente à Figura 7.11a

Índice	Tempo de processamento (s)	Conformidade (C)	r_ε	r_u	α_s
1	0.0495	-4.3999	0.0495	0.0522	1.6919
2	0.0773	-4.6216	0.0046	0.0051	2.9747
3	0.0821	-4.6668	0.0052	0.0043	3.1443
4	0.0649	-4.5932	0.0091	0.0117	1.9415
5	0.0507	-4.4756	0.0091	0.0117	1.7464

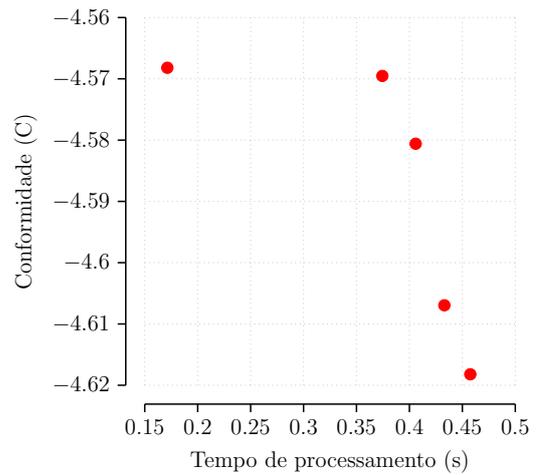
Tabela 7.8 – Os resultados da frente de Pareto multiobjetivo da rotina totalmente automatizada, para uma distribuição nodal regular com 165 nós e diferentes valores de α_q para cada nó da discretização sem malha, referente à Figura 7.11b

Índice	Tempo de processamento (s)	Conformidade (C)	r_ε	r_u	α_s
1	0.3744	-4.5695	0.0083	0.0083	3.8145
2	0.4575	-4.6182	0.0023	0.0022	4.6639
3	0.1713	-4.5682	0.0084	0.0087	1.9470
4	0.4330	-4.6069	1.281E-04	1.980E-04	4.3474
5	0.4059	-4.5806	0.0059	0.0059	3.9606

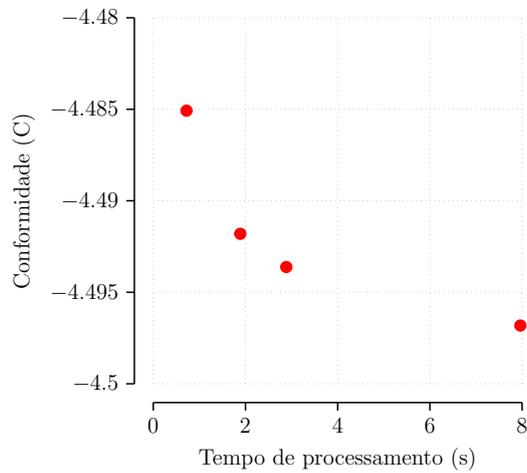
são ótimos e não dominados. Para cada um dos pontos, um α_s otimizado é apresentado, considerando o α_q obtido na primeira etapa da análise, garantindo que $A_q < 1 \times 10^6$. Os resultados mostram que a rotina gera soluções altamente precisas para todos os pontos na frente de Pareto, de forma rápida e eficiente; além de garantir a convergência, como nas abordagens anteriores.



(a) Distribuição nodal regular com 52 nós.



(b) Distribuição nodal regular com 165 nós.



(c) Distribuição nodal regular com 585 nós.

Figura 7.11 – A frente de Pareto da otimização multiobjetivo para três distribuições nodais regulares da viga engastada em balanço, em função do indicador de conformidade C , do tempo de processamento ($CPU\ Time$) em segundos e da relação A_q , para a rotina totalmente automatizada de otimização.

Tabela 7.9 – Os resultados da frente de Pareto multiobjetivo da rotina totalmente automatizada, para uma distribuição nodal regular com 585 nós e diferentes valores de α_q para cada nó da discretização sem malha, referente à Figura 7.11c

Índice	Tempo de processamento (s)	Conformidade (C)	r_ε	r_u	α_s
1	0.7201	-4.4851	0.0016	9.535E-04	1.5
2	2.8858	-4.4936	1.660E-04	2.557E-04	4.5010
3	1.8870	-4.4918	2.463E-04	6.637E-05	3.4838
4	7.9617	-4.4968	9.110E-04	9.074E-04	9.5744

Assim, como na abordagem que deu origem à rotina, o menor erro relativo não é sempre obtido para o menor valor do indicador de conformidade, gerando bons resultados em todos os pontos na frente de Pareto. O esforço computacional reduzido é fruto da combinação da eficiência das duas abordagens, resultando em uma otimização eficiente.

7.1.4.2 - Deslocamentos e tensões

Os deslocamentos obtidos com o GSMF, para uma distribuição nodal com 165 nós, considerando $\alpha_s = 4.3474$ e o vetor contendo valores nodais de α_q obtidos pelo processo de otimização, representados na Figura 7.12, mostraram uma boa concordância com os resultados analíticos. As tensões, calculadas no centro da viga, em $x_1 = L/2$ e

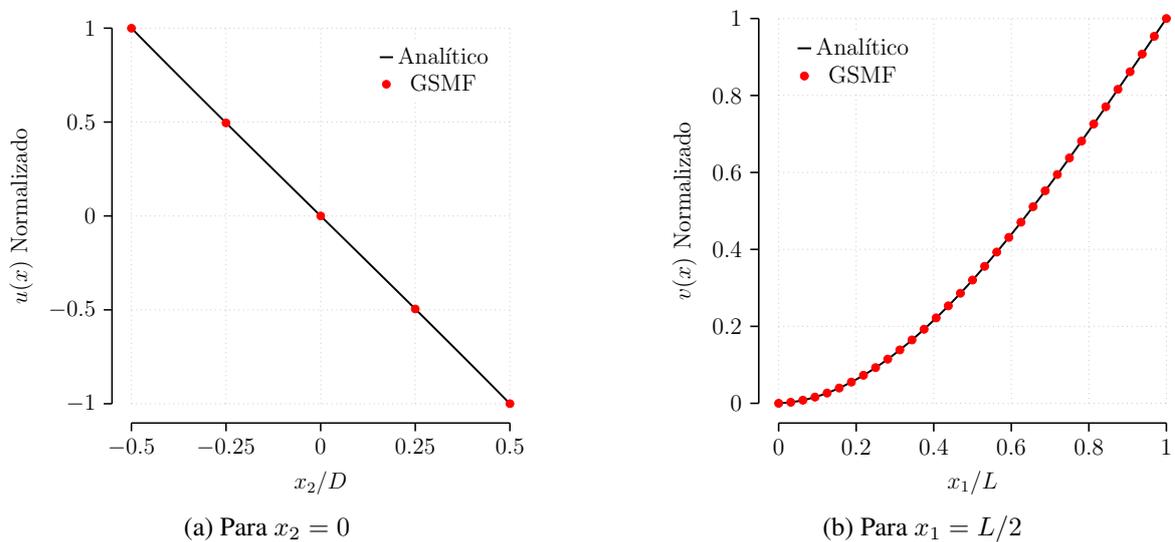


Figura 7.12 – Deslocamento vertical e horizontal normalizados em uma viga engastada com uma discretização de $33 \times 5 = 165$ nós, utilizando a rotina totalmente automatizada de otimização.

$x_2 \in [-D/2, D/2]$, também apresentaram bons resultados quando comparadas com as respostas analíticas, como mostra a Figura 7.13.

7.2 - NOVA ABORDAGEM PROPOSTA PARA A OTIMIZAÇÃO DOS MODELOS DO MÉTODO SEM MALHA LOCAL

Uma série de metodologias de otimização dos parâmetros adimensionais do método sem malha local (ILMF), foi empregada ao longo dessa pesquisa e dos estudos desenvolvidos pelo grupo do professor Artur Portela, no PECC, nos últimos sete anos. Com o intuito de explorar essas metodologias clarificando-as ainda mais, além de contribuir na sedimentação

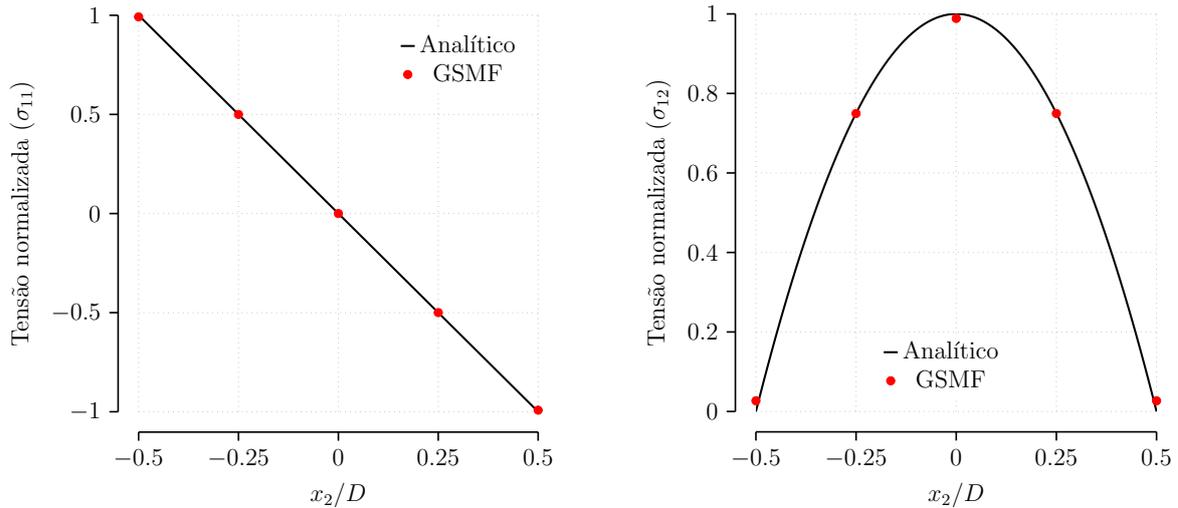


Figura 7.13 – Distribuição de tensões de uma viga engastada em $x_1 = L/2$ e $x_2 \in [-D/2, D/2]$ com uma discretização de $33 \times 5 = 165$ nós, utilizando a rotina totalmente automatizada de otimização.

e estabelecimento dos parâmetros adequados no processo de otimização, uma nova abordagem é aqui proposta e uma série de investigações feitas, aplicando ferramentas estatísticas, como correlação (Seção 5.4), de visualização de dados, métricas clássicas de performance (Seção 2.1.9) dos MMAs, entre outras.

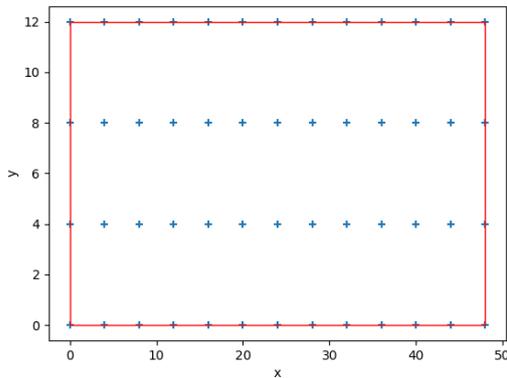
Até agora, nesse texto, os resultados foram gerados otimizando os modelos com GA e a *toolbox* do MATLAB. Mas, como o objeto dessa pesquisa é o emprego amplo dos algoritmos meta-heurísticos multiobjetivo na otimização do ILMF, doravante será utilizado também o Platypus, que dispõe de dez algoritmos meta-heurísticos multiobjetivo já consagrados na comunidade científica (vide Seções 2.2.1 e 2.2.5). Isso possibilita ampliar o ângulo de visão sobre os modelos do ILMF e traz uma riqueza na qualidade e quantidade dos resultados gerados pela otimização. A partir desses resultados será definido quais algoritmos são mais apropriados para cada área em que o ILMF se aplica, além de várias outras deduções.

7.2.1 - Equivalência Entre o *Compliance* e o Erro Relativo ao Campo de Deslocamentos

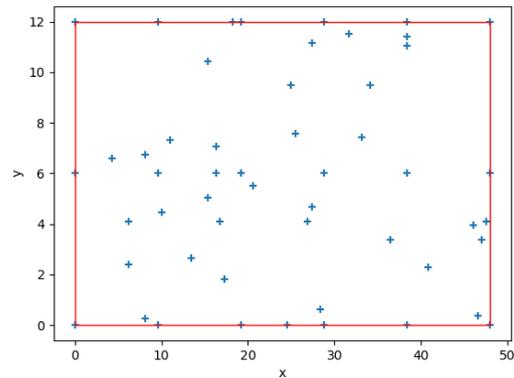
Nesta seção será tratado o mesmo caso da seção anterior, da viga engastada e em balanço.

A influência do tipo da discretização nodal do domínio, se regular ou randômica, foi considerada. A quantidade de nós e os limites das variáveis foram os mesmos utilizados e apresentados na Seção 7.1, com pequenas discrepâncias para as discretizações regulares de 165 e 585 nós, em virtude do critério de atribuição do número de faixas horizontais e

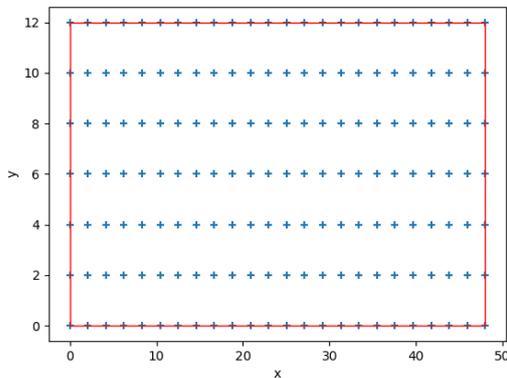
verticais. Amostras das discretizações utilizadas podem ser vistas na Figura 7.14.



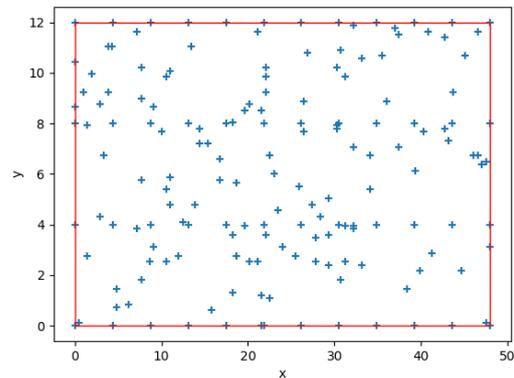
(a) Discretização regular com 52 nós.



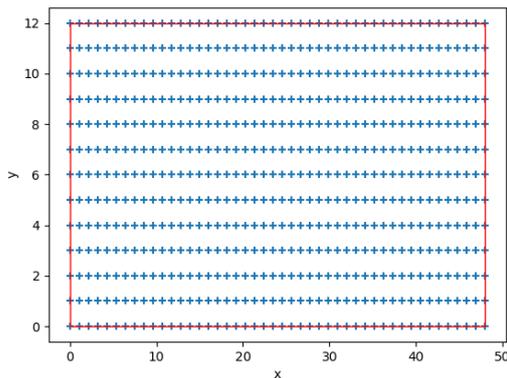
(b) Discretização randômica com 52 nós.



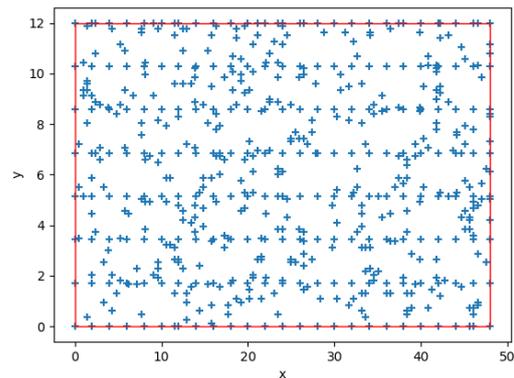
(c) Discretização regular com 168 nós.



(d) Discretização randômica com 165 nós.



(e) Discretização regular com 598 nós.



(f) Discretização randômica com 585 nós.

Figura 7.14 – Discretizações nodais regulares e randômicas utilizadas.

O resultado da escalagem dos dados pode ser visto na Figura 7.15, para a 1ª fase e, na Figura 7.16, para os resultados da otimização (2ª fase). Como pode ser observado, é notória a diferença na visualização que há entre os dados originais e os escalados, mostrando a

importância das funções de escalagem apresentadas anteriormente.

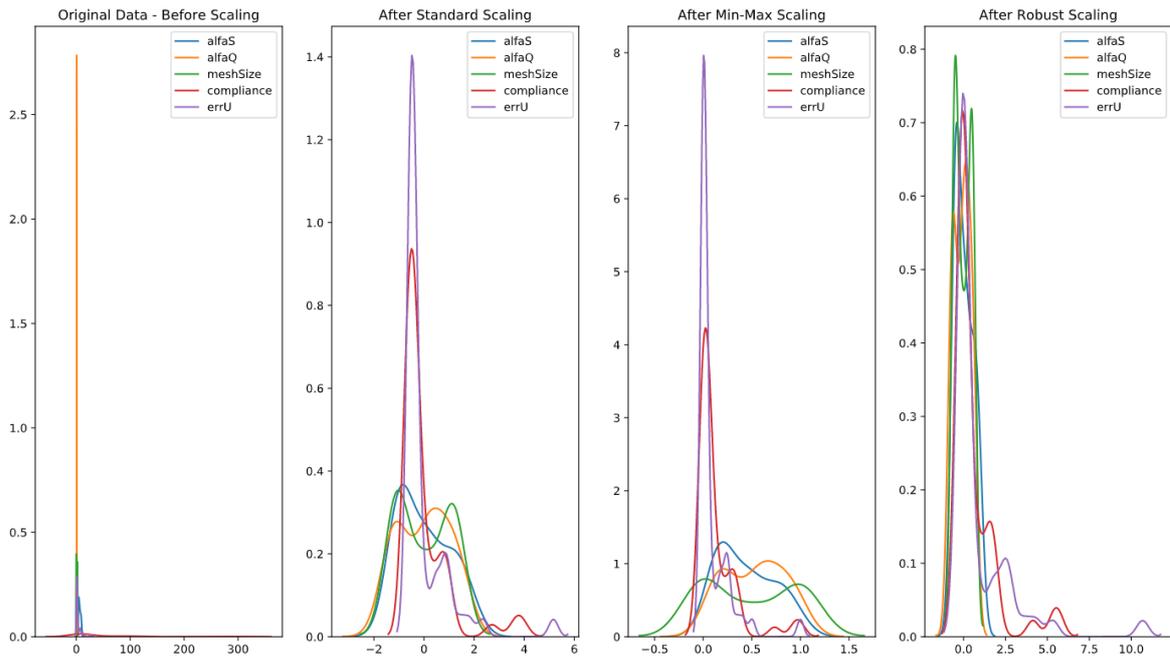


Figura 7.15 – Escalagem dos dados - 1ª fase.

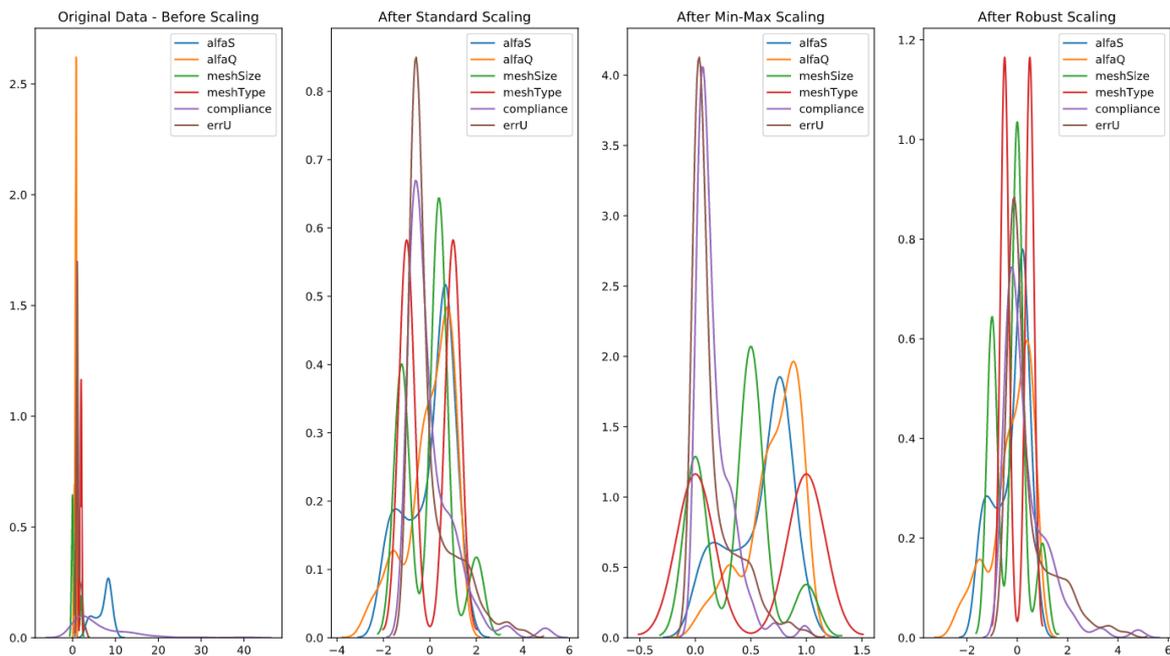


Figura 7.16 – Escalagem dos dados - 2ª fase.

Os resultados da otimização referentes à 1ª e 2ª fases estão descritos estatisticamente nas Tabelas 7.10 e 7.11, respectivamente. Esta descrição estatística compreende, na sequência dos cabeçalhos das linhas: quantidade de soluções, média, desvio padrão, valor mínimo, 1º quartil, 2º quartil (ou mediana), 3º quartil e valor máximo, respectivamente. Os valores também estão escalados para facilitar a apresentação dos resultados. Observa-se que foram encontradas 50 soluções únicas e não dominadas quando da busca pela frente de Pareto de

referência (1ª fase) e 296 na otimização propriamente dita (2ª fase). A quantidade menor de soluções na frente de Pareto de referência se dá pelo fato de os algoritmos terem executado mais iterações nessa fase e terem apurado a convergência, diminuindo assim as soluções aptas a integrarem a frente de Pareto.

Tabela 7.10 – Descrição estatística dos resultados escalados - 1ª fase.

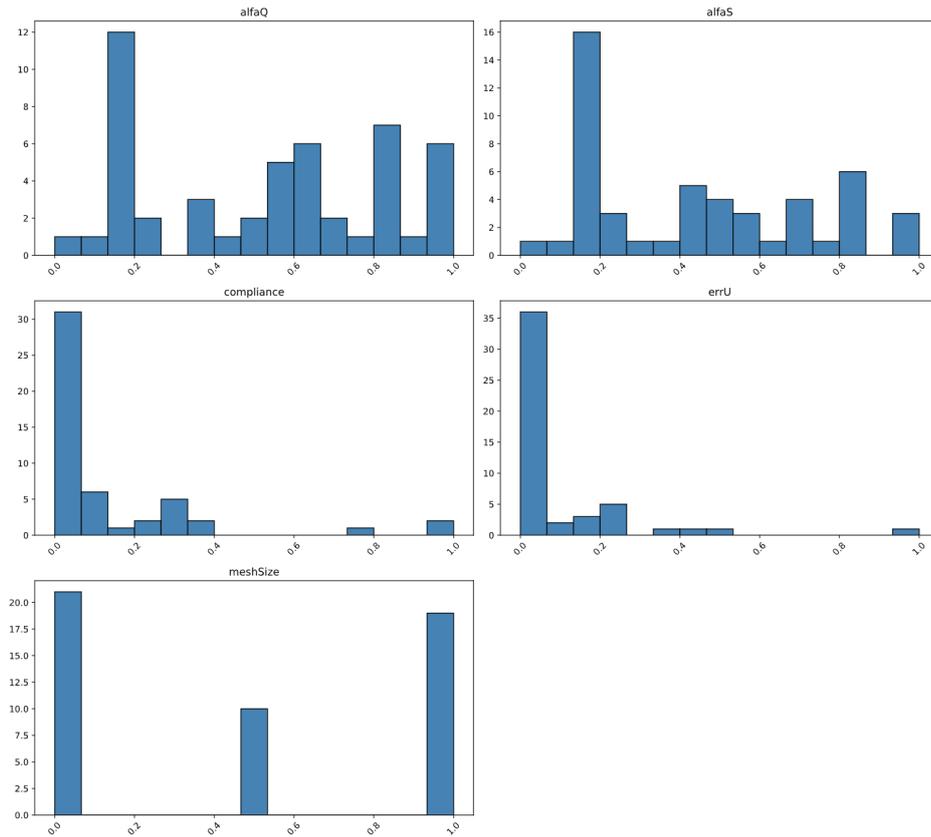
Statistic	alfaS	alfaQ	compliance	errU
count	50	50	50	50
mean	0.44079	0.52858	0.13289	0.08972
std	0.28585	0.30199	0.22373	0.17805
min	0.00000	0.00000	0.00000	0.00000
25%	0.16499	0.18225	0.00650	0.00060
50%	0.41763	0.58126	0.03309	0.00767
75%	0.70520	0.80398	0.17591	0.09352
max	1.00000	1.00000	1.00000	1.00000

Tabela 7.11 – Descrição estatística dos resultados escalados - 2ª fase.

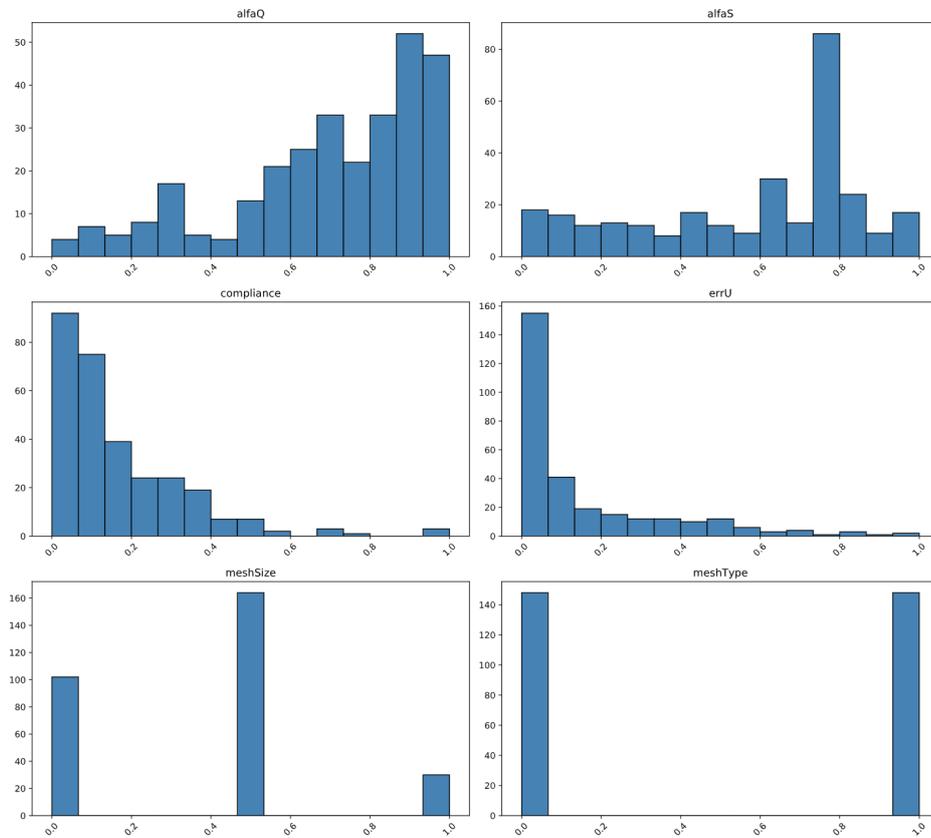
Statistic	alfaS	alfaQ	compliance	errU
count	296	296	296	296
mean	0.57619	0.69749	0.16617	0.15715
std	0.27923	0.24688	0.16520	0.20628
min	0.00000	0.00000	0.00000	0.00000
25%	0.35832	0.58866	0.05282	0.01772
50%	0.67336	0.76515	0.10601	0.06060
75%	0.77859	0.89228	0.23595	0.23199
max	1.00000	1.00000	1.00000	1.00000

As Figuras 7.17a e 7.17b apresentam as frequências relativas para cada variável e cada função objetivo, das soluções pertinentes às frentes de Pareto da 1ª e 2ª fases da otimização, respectivamente. Pode ser observado que em ambas as frentes de Pareto há uma concentração das soluções na região de valor mínimo para o *compliance* e o erro no campo de deslocamentos (*errU*), mostrando a eficiência dos MMAs na otimização do ILMF. Na 1ª fase, a com um maior número de iterações, as discretizações com 52 e 585 nós foram priorizadas pelos MMAs, enquanto que na 2ª fase, maior ênfase foi dada à discretização de 165 nós. Nestas figuras, *meshSize* = 0.0 se refere à discretização de 52 nós, e *meshSize* = 0.5 e *meshSize* = 1.0 se referem a 165 e 585 nós, respectivamente. Similarmente, *meshType* = 0.0 representa a discretização regular, enquanto *meshType* = 1.0 representa a randômica. Cabe destacar que o termo "mesh"às vezes é utilizado nessa pesquisa por questão de praticidade no desenvolvimento do código, mas, como o nome diz, são métodos sem malha. O que ocorre é uma discretização nodal do domínio e não a geração de uma malha propriamente dita.

Com base na fundamentação apresentada na Seção 5.4, foram gerados mapas de calor dos



(a) Frente de Pareto - 1ª fase.



(b) Frente de Pareto - 2ª fase.

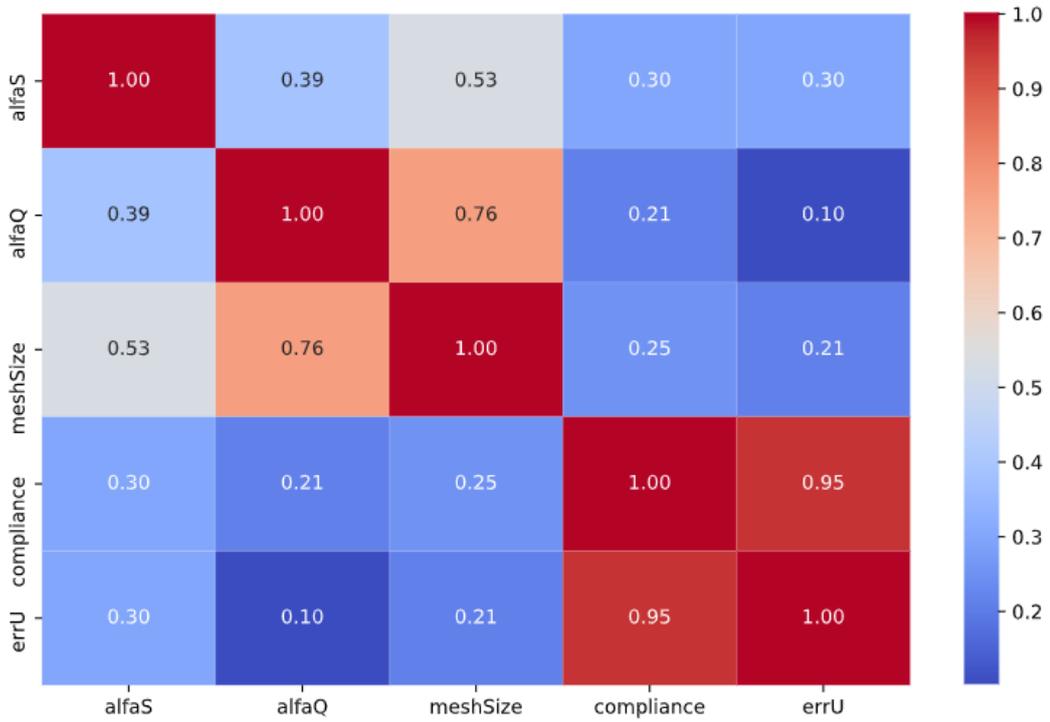
Figura 7.17 – Frequências relativas dos parâmetros da otimização.

parâmetros da otimização em função do coeficiente de correlação de Pearson, como mostram as Figuras 7.18a e 7.18b. Aqui fica demonstrado estatisticamente que a função objetivo baseada no erro no campo de deslocamentos pode ser substituída pela baseada na conformidade estrutural C , uma vez que o coeficiente de correlação entre C e $errU$ é 0.95 na 1ª fase de otimização e, de 0.84 na 2ª. Esses coeficientes indicam que os vetores que representam as variáveis em um espaço n -dimensional são praticamente colineares e possuem uma correlação muito forte. Depreende-se também que os MMAs são muito robustos na otimização dos modelos ILMF, pois, mesmo com poucas iterações na 2ª fase de otimização, a correlação obtida entre C e $errU$ foi forte, como esperado. Presume-se também que para um número de iterações muito maior, esse coeficiente tende a 1, ou seja, uma correlação perfeita positiva. Como os parâmetros α_S e α_Q estão diretamente ligados à discretização do domínio, não é surpresa que exista correlação entre eles e o $meshSize$ (Figura 7.18a, coeficientes de correlação 0.53 e 0.76, respectivamente).

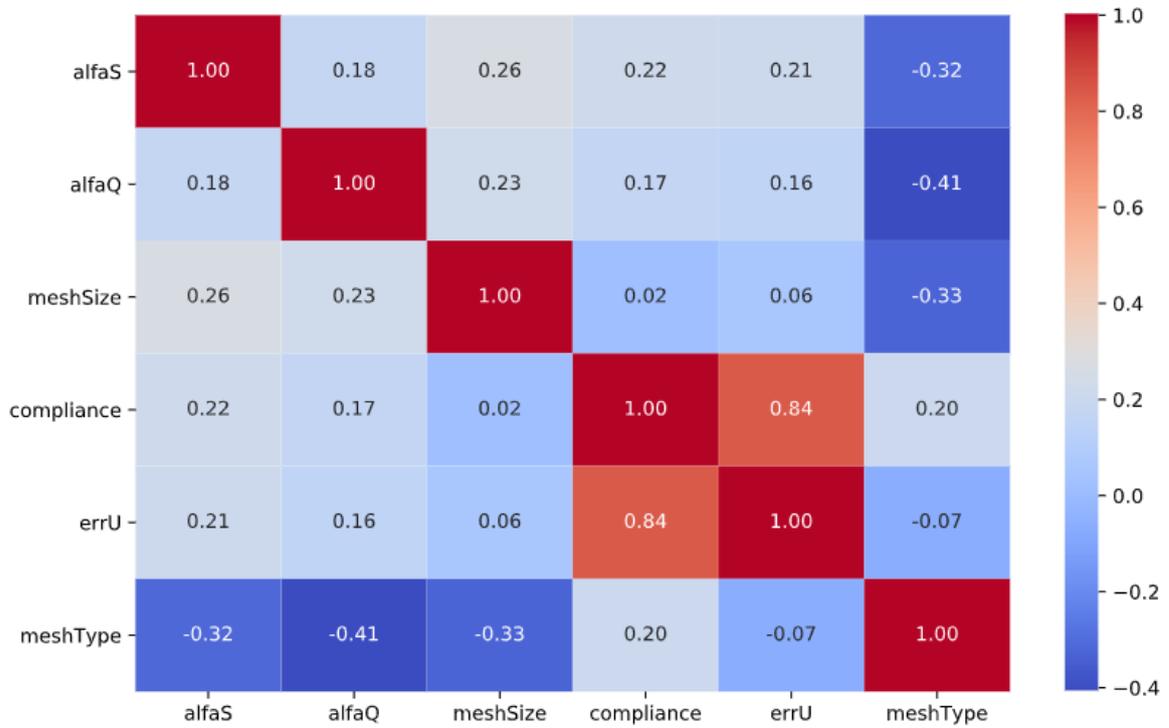
Nas Figuras 7.19 e 7.20, além de se observar se há correlação entre os parâmetros, através da linha de regressão linear que é plotada, também pode-se observar a forma da distribuição das soluções encontradas no processo de otimização. Confrontando duas variáveis quaisquer (uma da linha e outra da coluna) pode-se analisar imediatamente a relação que há entre elas. São gráficos emparelhados, do inglês: *pairwise plot*. Mais uma vez é notória a similaridade entre as funções objetivo baseadas no $errU$ e no *compliance*. Inclusive na forma em que seus valores são distribuídos nas soluções constituintes da frente de Pareto (vide os cruzamentos $errU \times errU$ e *compliance* \times *compliance*).

Uma vista ampliada da correlação dos dois parâmetros mais relevantes da otimização (*compliance* e $errU$) é dada pelas Figuras 7.21a e 7.21b. Note a concentração de soluções na vizinhança do mínimo do *compliance* e do $errU$ (que são os resultados pretendidos, já que é feita uma minimização destes parâmetros), em ambas as fases da otimização.

A plotagem dos resultados em coordenadas paralelas permite analisar o impacto que os valores dos parâmetros de otimização (variáveis e funções objetivo) exercem uns sobre os outros. Para isso, as Figuras 7.22a e 7.22b apresentam plotagens em coordenadas paralelas para a 1ª e 2ª fases da otimização. Lembrando que na 1ª fase não foi feita a discriminação das soluções por algoritmo/MMA; só na 2ª. No primeiro gráfico há a discriminação pelo $meshSize$ (0.0/0.5/1.0 para 52/165/585 nós, respectivamente).



(a) Resultados - 1ª fase.



(b) Resultados - 2ª fase.

Figura 7.18 – Mapa de calor dos parâmetros da otimização em função do coeficiente de correlação.

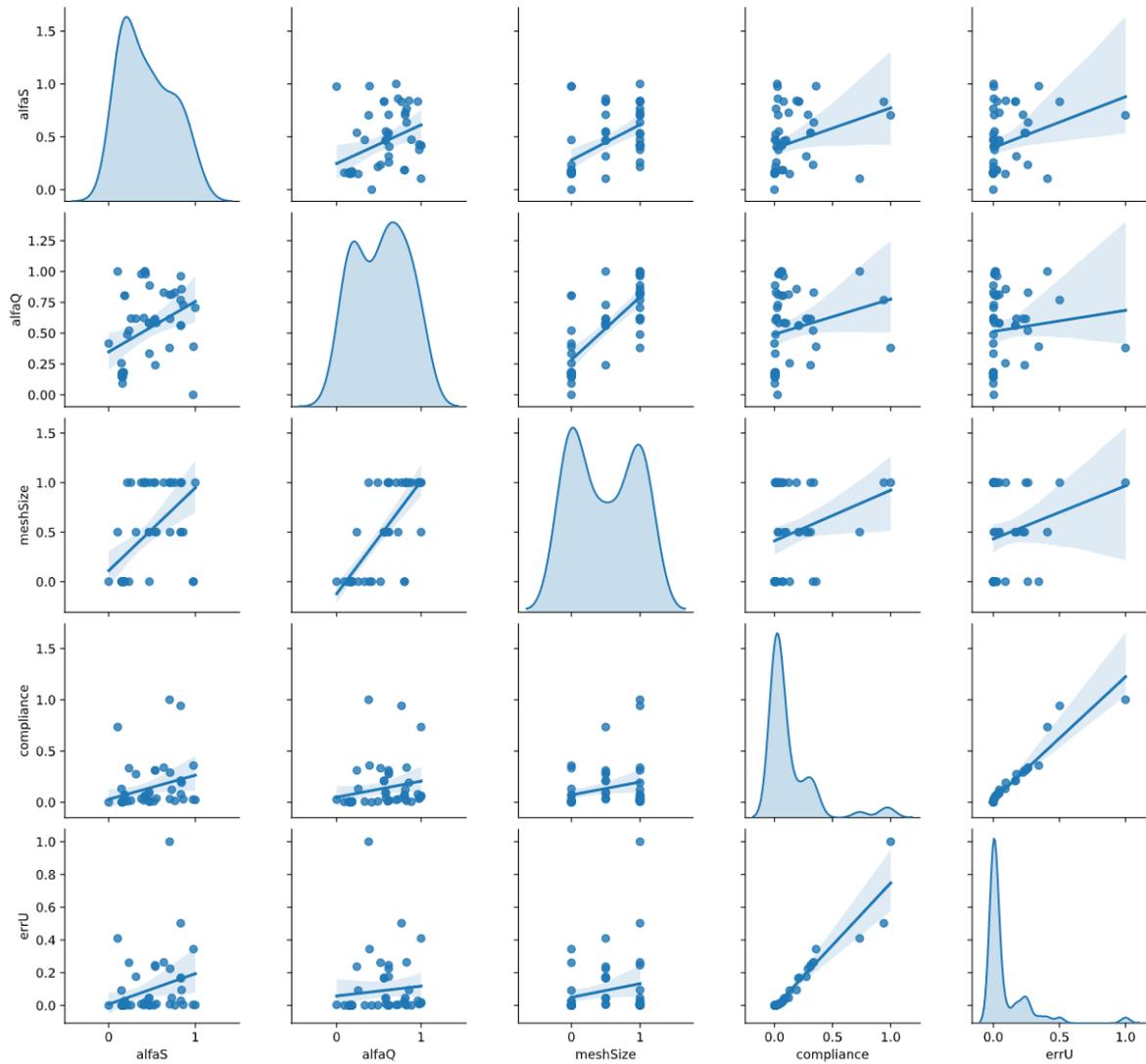


Figura 7.19 – Pairwise plot - 1ª fase.

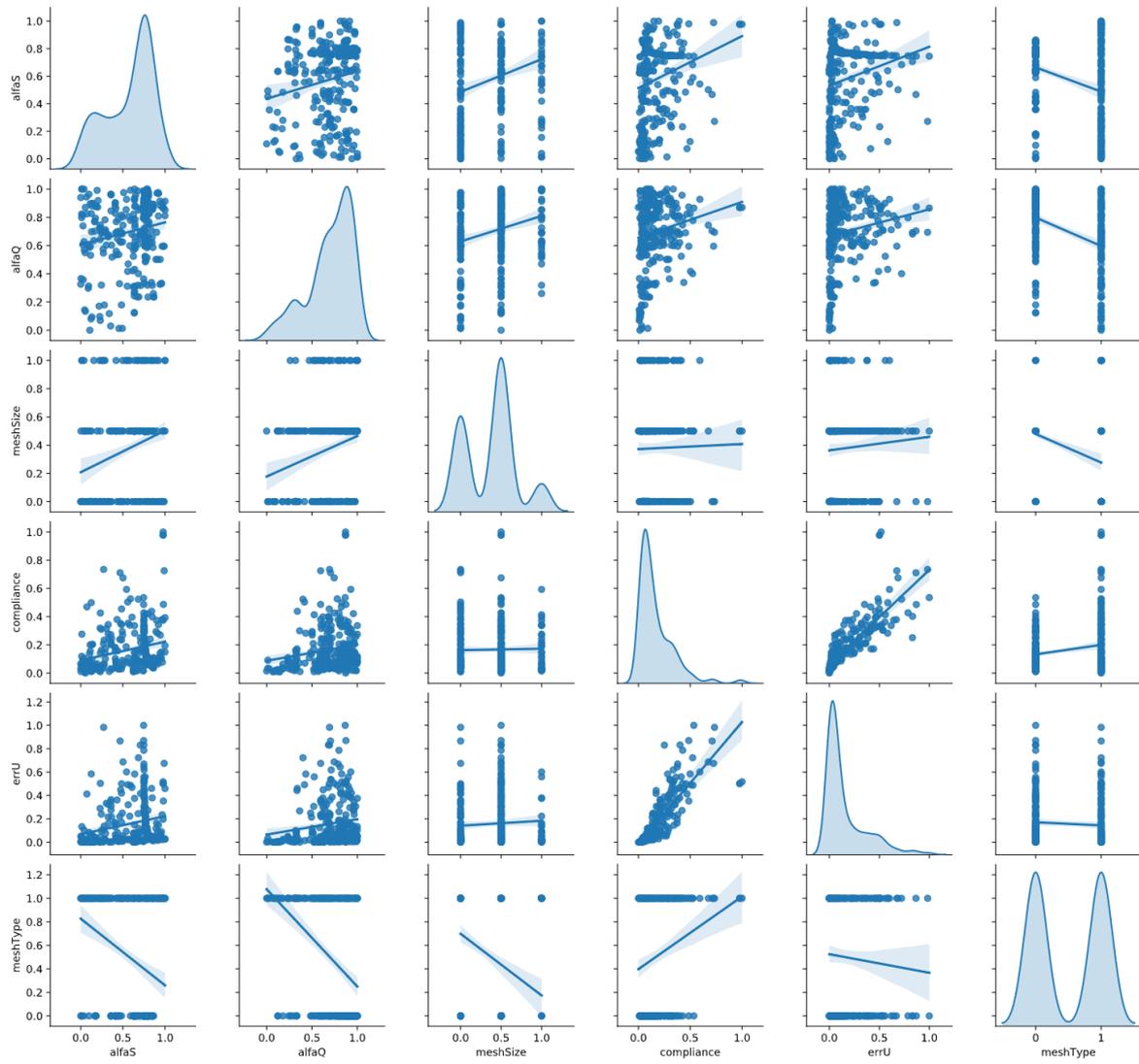
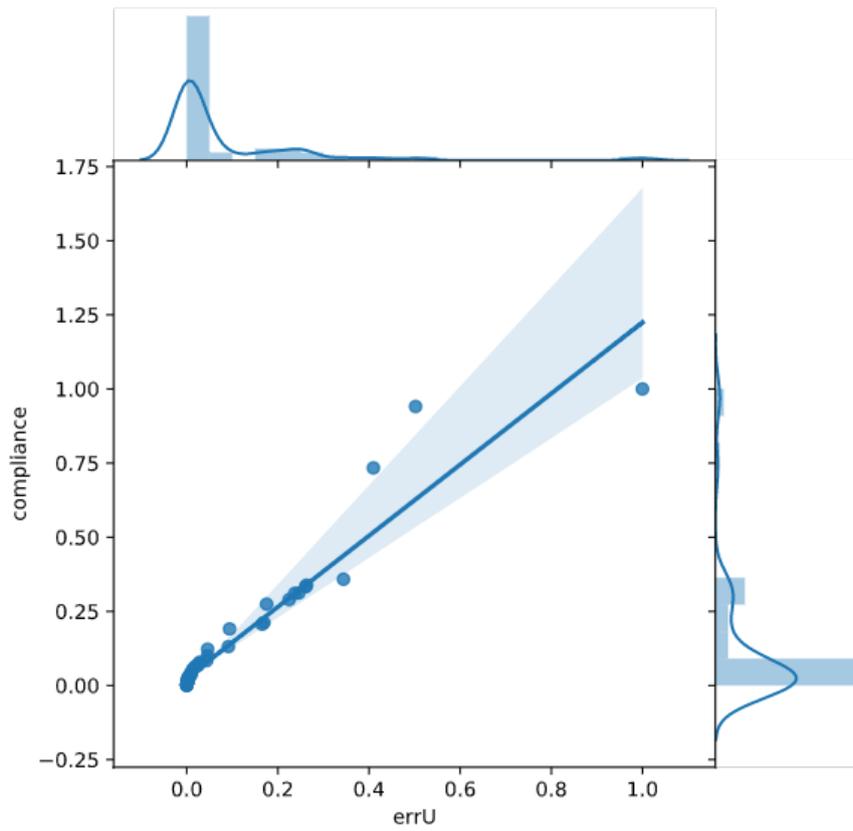
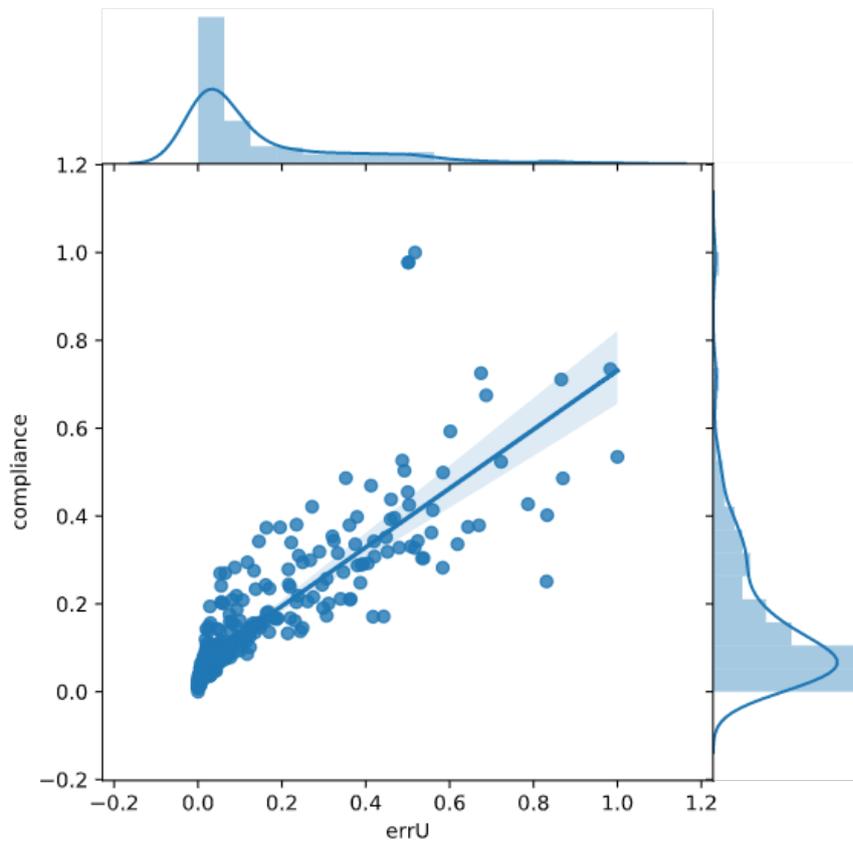


Figura 7.20 – Pairwise plot - 2^a fase.

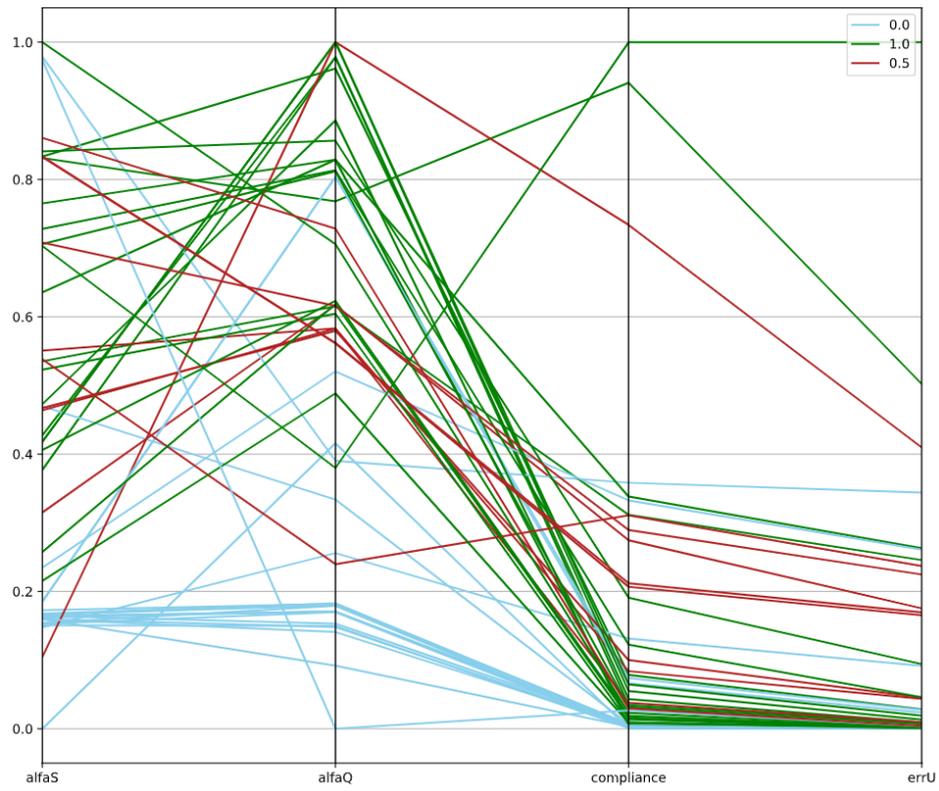


(a) Resultados - 1ª fase.

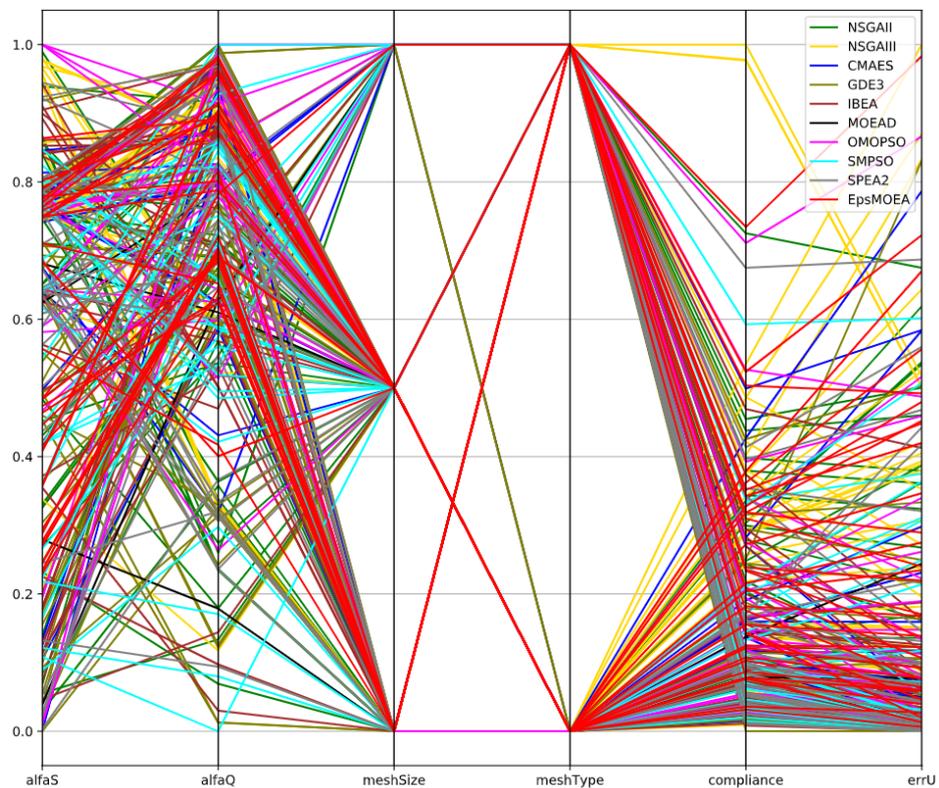


(b) Resultados - 2ª fase.

Figura 7.21 – Correlação entre *errU* e *compliance*.



(a) Resultados - 1ª fase - discriminados por meshSize (0.0/0.5/1.0 para 52/165/585 nós, respectivamente).



(b) Resultados - 2ª fase - discriminados por MMA.

Figura 7.22 – Coordenadas paralelas dos parâmetros de otimização.

7.2.2 - Influência da Discretização Nodal do Domínio nos Resultados do ILMF e da Otimização

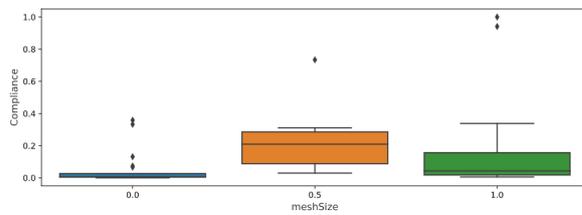
A influência da densidade nodal da discretização do domínio, no *compliance*, pode ser vista na Figura 7.23, para as duas fases da otimização. Nota-se que os MMAs e o ILMF performam muito bem na otimização do modelo da viga, independente da densidade nodal. A densidade nodal está mais relacionada com a forma da distribuição das soluções das frentes de Pareto e com a existência de *outliers* (pontos/soluções discrepantes, situados muito fora dos limites mínimo e máximo dos *boxplots*, e que não apresentam valor estatístico) do que com a precisão dos resultados; já que para todas as densidades nodais (*meshSize*), os coeficientes angulares das linhas de correlação $\text{errU} \times \text{compliance}$, apresentam valores próximos, isto é, as retas oriundas das diversas densidades apresentam inclinações quase idênticas.

A influência do tipo de discretização nodal (se regular ou randômica) do domínio, no *compliance* e nas métricas de performance, é apresentada na Figura 7.24. Aqui, observações análogas às feitas para a densidade nodal, são válidas. Exceto que há uma diferença mais significativa na inclinação das retas de regressão linear. A Figura 7.24d mostra que as métricas de performance são praticamente indiferentes ao tipo de discretização (regular ou randômica).

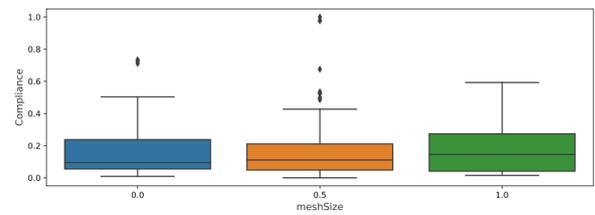
7.2.3 - Ranqueamento dos MMAs Através da *Utility Function* e as Métricas de Performance

Ao lado da demonstração que o errU e o *compliance* podem ser utilizados de forma intercambiável quando da definição das funções objetivo para a otimização dos modelos ILMF, o ranqueamento dos MMAs pela performance na otimizações destes modelos constitui os principais objetivos desta pesquisa. Para a consecução desse ranqueamento, mais algumas análises dos resultados necessitam ser feitas. Nesse sentido, a Figura 7.25 mostra a quantidade de soluções únicas, não dominadas e ótimas de Pareto, geradas por algoritmo utilizado. Os algoritmos CMAES e o MOEAD, não geraram um número significativo de soluções na frente de Pareto, quando comparado com os demais MMAs.

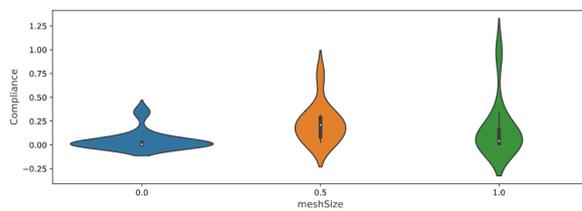
A Figura 7.26 apresenta gráficos emparelhados (*pairwise plot*) dos parâmetros da otimização, 2ª fase, onde as soluções da frente de Pareto são discriminadas por MMA. Todos os MMAs apresentaram soluções satisfatórias quanto à precisão e exploraram bem o espaço de soluções, apresentando distribuições com formas muito semelhantes, como pode ser visto nos gráficos da diagonal descendente (do canto superior esquerdo ao canto inferior direito).



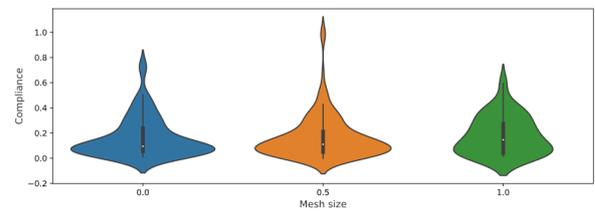
(a) *Box plot - 1ª fase.*



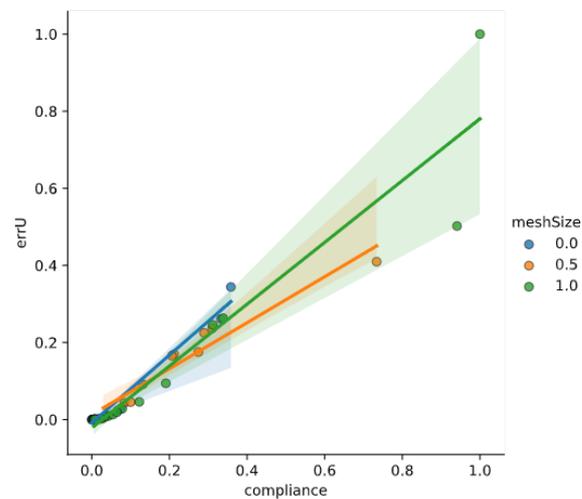
(b) *Box plot - 2ª fase.*



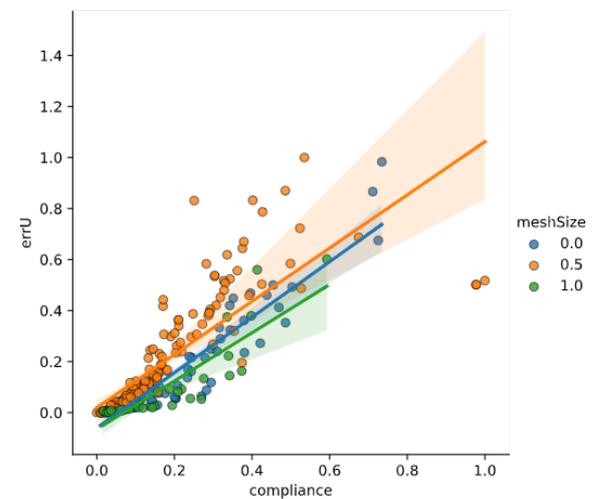
(c) *Violin plot - 1ª fase.*



(d) *Violin plot - 2ª fase.*

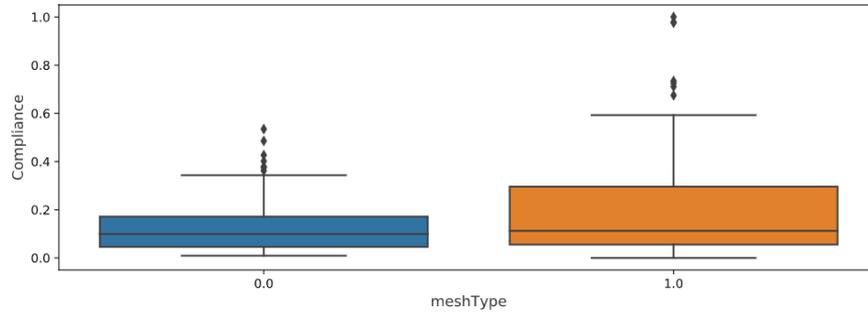


(e) *Regressão linear - 1ª fase.*

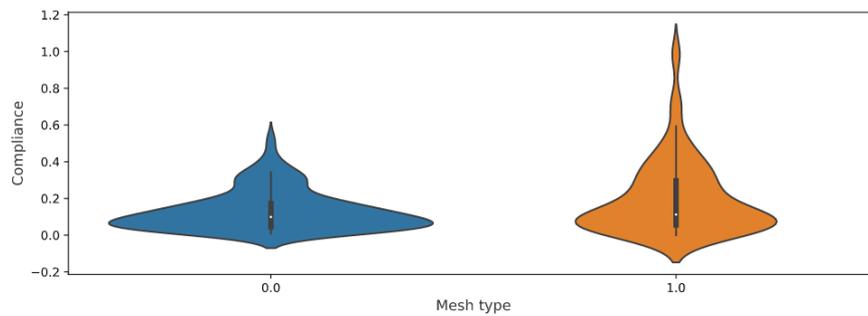


(f) *Regressão linear - 2ª fase.*

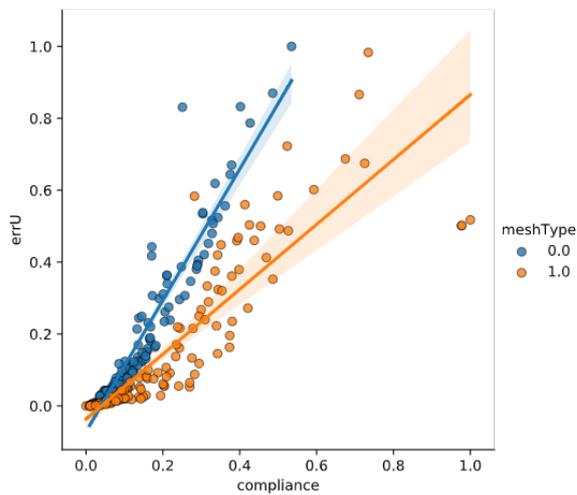
Figura 7.23 – Influência no *compliance*, da densidade nodal da discretização do domínio (meshSize = 0.0/0.5/1.0 para 52/165/585 nós, respectivamente).



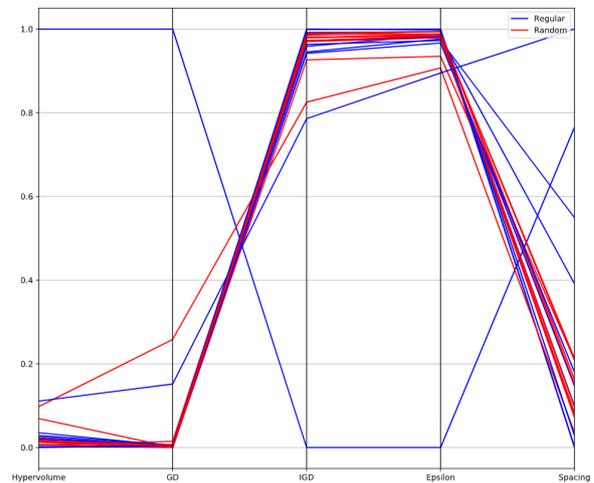
(a) *Box plot - 2ª fase.*



(b) *Violin plot - 2ª fase.*



(c) *Regressão linear - 2ª fase.*



(d) *Coordenadas paralelas das métricas - 2ª fase.*

Figura 7.24 – Influência no *compliance* e nas métricas de performance; devida ao tipo da discretização nodal do domínio (meshType = 0.0, discretização regular; meshType = 1.0, discretização randômica).

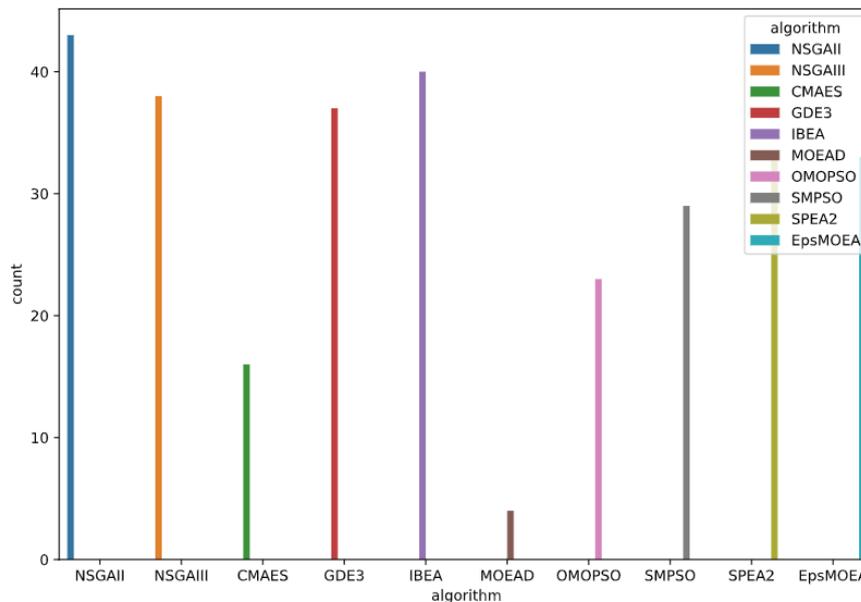


Figura 7.25 – Quantidade de soluções únicas, não dominadas e ótimas de Pareto, por MMA utilizado - 2ª fase.

Uma observação mais detalhada é propiciada pelas Figuras 7.27a e 7.27b, onde as inclinações muito próximas das retas de regressão linear, obtidas a partir das soluções para cada frente de Pareto, por MMA, confirmam a boa performance dos MMAs empregados na otimização. Pode-se observar também, que tanto o espaço das variáveis (alfaS e alfaQ), quanto o espaço das soluções (errU e compliance) foram explorados pelos MMAs em regiões muito próximas, refletindo uma quase colinearidade das retas de regressão linear.

Todos os MMAs performaram muito bem conforme mostrado até agora, porém, ainda não é possível ranqueá-los quanto à performance. Por isso, as métricas de performance são fundamentais no mensuramento da qualidade das soluções e do desempenho dos MMAs. A Tabela 7.12 traz as métricas da otimização, que foram obtidas comparando-se as soluções da 2ª fase da otimização com as soluções de referência (1ª fase). Nessa tabela contém também o ranqueamento dos MMAs para esse exemplo (mecânica linear elástica - viga engastada em balanço). O ranqueamento foi feito aplicando-se uma *utility function* (Seção 2.1.9) nas métricas de performance individuais por algoritmo. Então, os três MMAs com melhor desempenho na otimização do ILMF aplicado nesse exemplo, foram: MOEAD, IBEA e NSGAIII, nesta ordem. Isso não desabilita o uso dos demais MMAs na otimização desse tipo de modelo, uma vez que, conforme demonstrado, obtiveram excelentes e precisas soluções na frente de Pareto. Tanto que há uma alternância dos algoritmos que dominam cada métrica, por exemplo, o MMA obteve valor mínimo (preferível) para a IGD e Epsilon, e o pior resultado para a GD, todavia, foi o melhor ranqueado já que a *utility function* considera todas as métricas simultaneamente em seu cálculo.

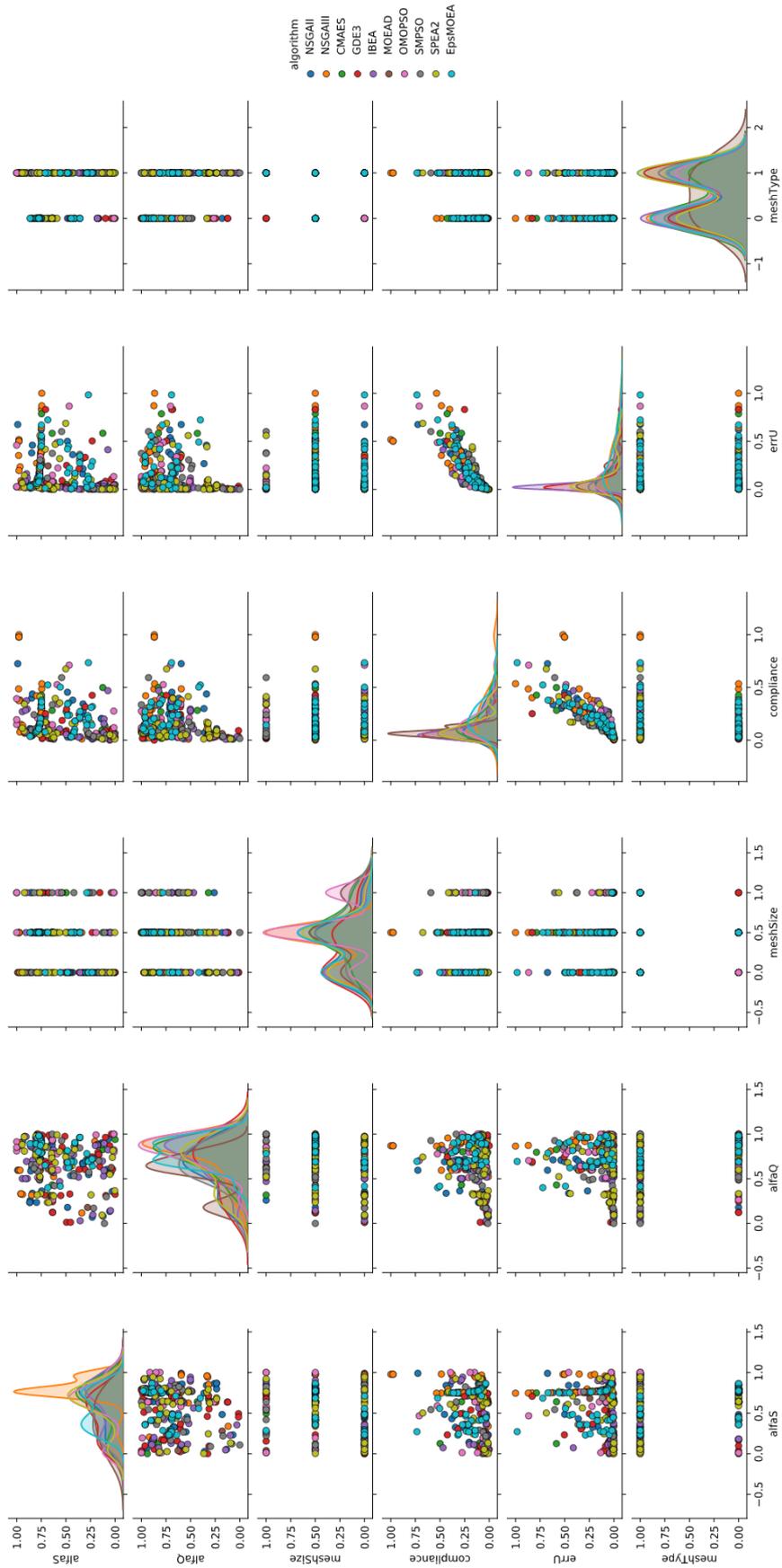
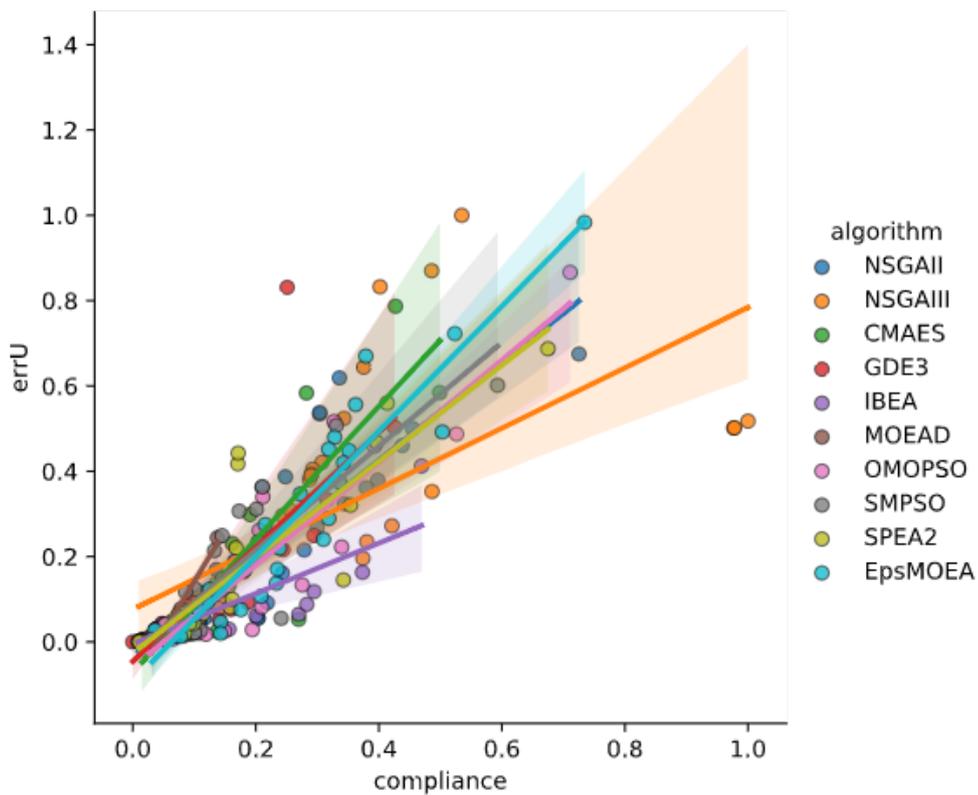
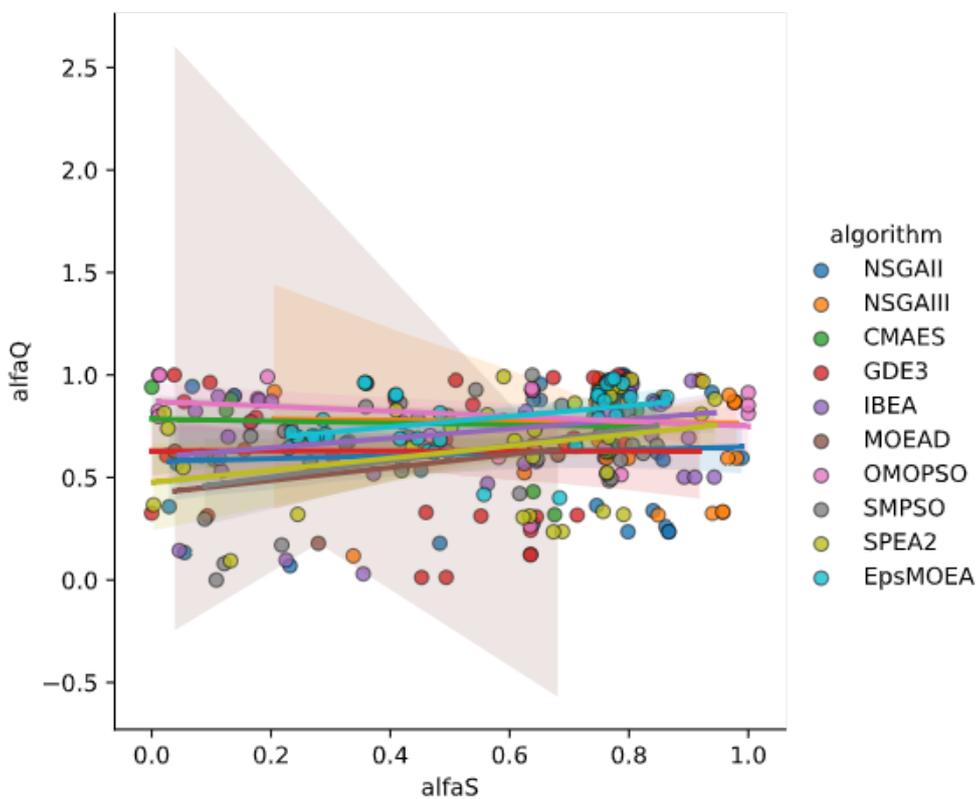


Figura 7.26 – *Pairwise plot* dos parâmetros da otimização, 2ª fase, com as soluções discriminadas por MMA.



(a) Espaço das soluções: $errU$ x $compliance$ - 2ª fase.



(b) Espaço das variáveis: $alfaQ$ x $alfaS$ - 2ª fase.

Figura 7.27 – Exploração dos espaços das variáveis e das soluções pelos MMAs.

Tabela 7.12 – Ranking dos MMAs a partir das métricas de performance escaladas.

Ranking	Algorithm	Hypervolume	GD	IGD	Epsilon	Spacing	Utility
1	MOEAD	1.0000000000	1.0000000000	0.0000000000	0.0000000000	0.7647957619	0.7647957619
2	IBEA	0.0062502766	0.0000000000	0.9903139936	0.9940989717	0.0020503952	1.9802130838
3	NSGAIII	0.0018324915	0.0006549827	1.0000000000	0.9982708033	0.0000000000	1.9970932944
4	NSGAII	0.0153279638	0.0019294022	0.9587327818	0.9855124252	0.0991897816	2.0300364270
5	EpsMOEA	0.0000000000	0.0050208892	0.9987344691	1.0000000000	0.0282412074	2.0319965657
6	GDE3	0.0251306852	0.0026133197	0.9450135955	0.9762155525	0.1471861852	2.0458979677
7	SPEA2	0.0354494434	0.0039778457	0.9419702686	0.9664741191	0.1802563688	2.0572291589
8	OMOPSO	0.0198530504	0.0069371173	0.9725974141	0.9812495186	0.3915635453	2.3324945449
9	CMAES	0.0289341011	0.0055381843	0.9632695848	0.9726728806	0.5493119436	2.4618584921
10	SMPSO	0.1107970647	0.1517065274	0.7857910963	0.8948532265	1.0000000000	2.7215537854

A Figura 7.28 mostra essas métricas plotadas em coordenadas paralelas, para uma visualização mais inteligível.

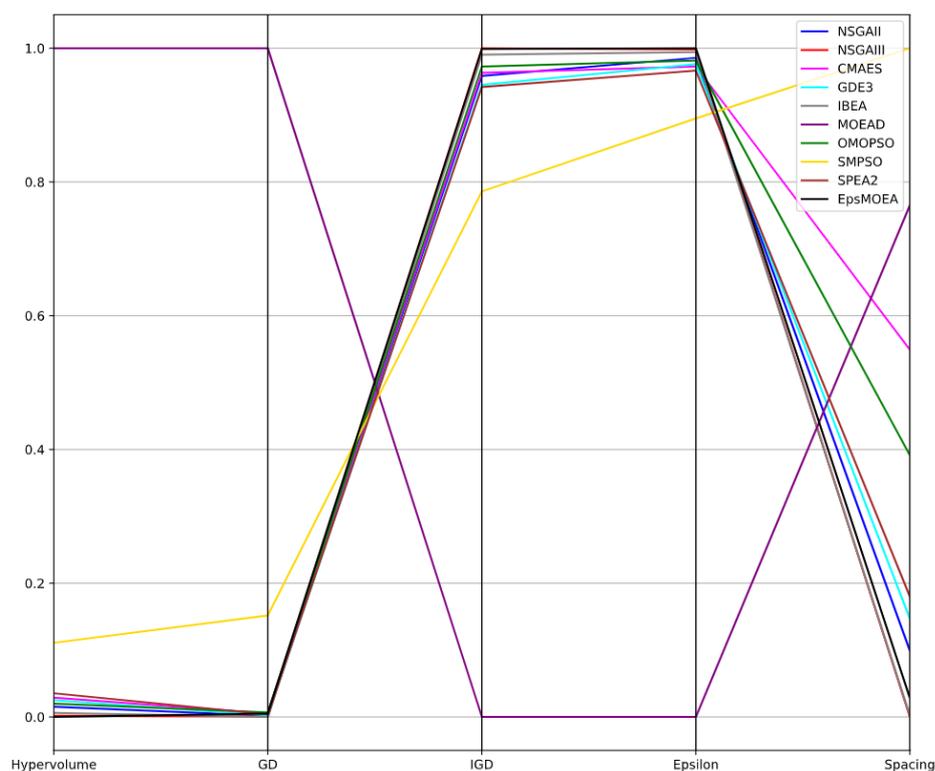


Figura 7.28 – Métricas de performance dos MMAs plotadas em coordenadas paralelas.

A função objetivo que leva em conta o tempo de processamento (*CPU time*) de cada iteração, para cada algoritmo, foi desconsiderada nesse exemplo, pois, como os MMAs possuem heurística intrínseca, tendem enviesar as soluções favorecendo as discretizações com menor densidade nodal, já que estas tomam menos tempo computacional e também fornecem bons resultados. Portanto, somente duas funções objetivo foram utilizadas nesse exemplo: as relativas ao *compliance* e ao *errU*. E não três, (*compliance* + *errU* + *CPU time*) como tentado inicialmente. Muito importante salientar também que a função objetivo que considera o *errU*

só foi considerada aqui para fins de demonstração da sua correlação com a função objetivo baseada no *compliance*, mas, a estratégia principal reside em, após essa demonstração, só se utilizar esta última para a otimização dos modelos. Até porque, como já enfatizado ao longo desse texto, para a maioria dos problemas não se tem soluções analíticas conhecidas, o que tornaria impossível calcular o err_U que surge entre as soluções analítica e numérica.

7.3 - APLICAÇÃO DA NOVA ABORDAGEM DE OTIMIZAÇÃO NA MECÂNICA DA FRATURA LINEAR ELÁSTICA

Oliveira (2019) comparou os resultados gerados utilizando o GSMF com os resultados obtidos pelo *Dual Boundary Element Method* (DBEM) com a técnica da integral J (J-DBEM), que é um método numérico muito preciso. Aliabadi e Portela (1999) apresentaram a estratégia de modelagem do DBEM, que considera trincas retas por partes, que são modeladas com elementos de contorno quadráticos descontínuos retos. Elementos de contorno quadráticos contínuos são utilizados ao longo dos contornos restantes do problema, exceto na interseção entre a trinca e uma borda, onde elementos de contorno semi-descontínuos são utilizados na borda.

Para as aplicações deste exemplo, Oliveira e Portela (2019) configuraram GA no MATLAB para minimizar o tempo de processamento e o indicador de conformidade (C), escolhidos como funções objetivo. A variável α_q foi fixada em 0.5. A variável α_s foi classificada como contínua e no intervalo $\alpha_s = 1.5$ a 10. A população inicial foi de 20 indivíduos, a função de aptidão foi escalonada utilizando um processo de classificação (*rank*), o operador de reprodução foi implementado baseando-se em uma seleção do tipo torneio, onde a mutação e o *crossover* foram limitados pelas restrições impostas. Por fim, a condição de parada foi para o número máximo de gerações de 50 ou uma mudança média na função de aptidão menor que 1×10^{-6} . A seção seguinte reflete sua abordagem e os resultados obtidos e, a comparação dos mesmos com os resultados obtidos pelos autores citados.

7.3.1 - Placa com Trinca Horizontal de Borda - Modo-I

Uma placa retangular com uma trinca de borda, esquematicamente representada na Figura 7.29, foi considerada para a primeira análise. O comprimento da trinca é dado por a , a largura da placa é dada por W e a altura é dada por $h = w/2$. A placa é carregada por uma tração uniforme $\bar{t} = \sigma$, aplicada simetricamente nas extremidades. Resultados foram obtidos para o caso $h/w = 0.5$, para serem comparados com os valores altamente precisos obtidos por Civelek e Erdogan (1982a). Cinco casos foram considerados, com $a/w = 0.2, 0.3, 0.4, 0.5$ e 0.6 . O GSMF foi modelado com domínios locais de colocação

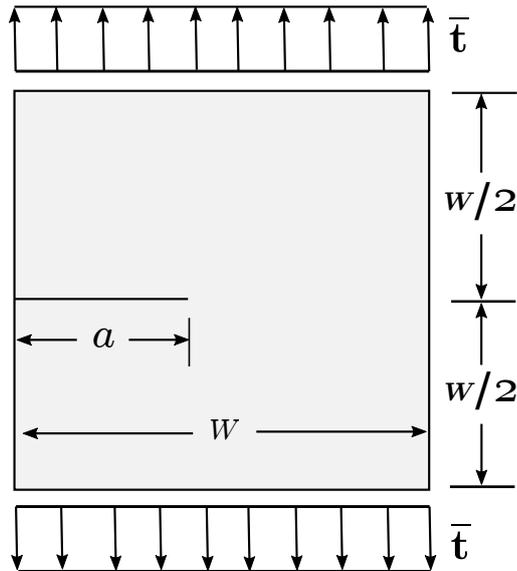


Figura 7.29 – Placa retangular com uma única trinca de borda sob carregamento em modo-I ($h/w = 0.5$).

retangulares e parâmetros de discretização $\alpha_s = 1.5$ a 3 e $\alpha_q = 0.5$, obtidos com uma rotina de otimização em AG, conforme apresentada anteriormente.

A aproximação com o MQM considera base polinomial de primeira ordem e funções ponderadoras do tipo *quartic spline*. A análise numérica foi realizada considerando-se a distribuição nodal representada na Figura 7.30, onde para todos os cinco casos de

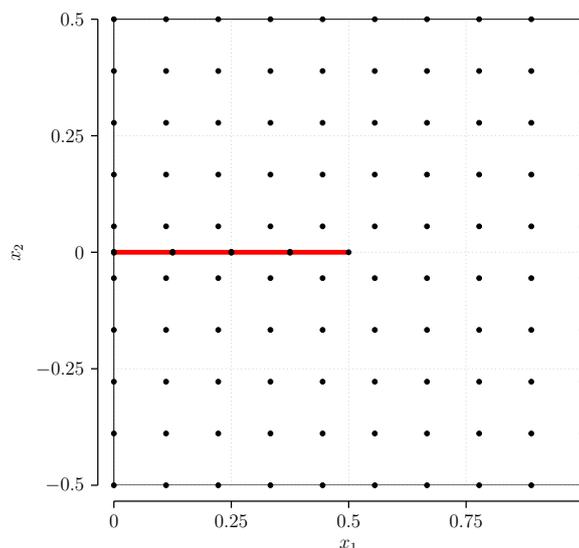


Figura 7.30 – Discretização de uma placa com uma distribuição nodal regular de $10 \times 10 = 100$ nós, com nós sobrepostos adicionais ao longo da trinca, para $a/w = 0.5$. A linha vermelha representa as faces da trinca.

comprimento de trinca considerados, apenas a distribuição nodal ao longo da trinca foi

modificada, sempre sem considerar nenhum refinamento ao redor da ponta da trinca.

Os resultados obtidos são apresentados na Tabela 7.13 e atestam a alta precisão do método,

Tabela 7.13 – Placa retangular com uma trinca de borda única sob carregamento em modo-I. GSMF representa os valores obtidos na presente pesquisa e J-DBEM representa os valores obtidos com o DBEM utilizando a integral J (Aliabadi e Portela, 1999). A porcentagem dos erros foi calculada a partir do valor de referência obtido em Civelek e Erdogan (1982a).

a/w	$K_I/(\bar{t}\sqrt{\pi a})$			% Erro	
	GSMF	J-DBEM	Civelek	GSMF	J-DBEM
0.2	1.520	1.495	1.488	0.0216	0.005
0.3	1.967	1.858	1.848	0.0647	0.005
0.4	2.413	2.338	2.324	0.0387	0.006
0.5	2.973	3.028	3.010	0.0122	0.006
0.6	3.991	4.184	4.152	0.0387	0.008

quando comparado com Civelek e Erdogan (1982a). Os resultados obtidos com o GSMF também são muito precisos quando comparados com os resultados obtidos pelo J-DBEM. Nessa análise, o fator de intensidade de tensão do modo-II é sempre menor do que 10^{-7} , tendo em vista que esse é um problema de modo-I. A configuração deformada da placa está esquematicamente representada na Figura 7.31.

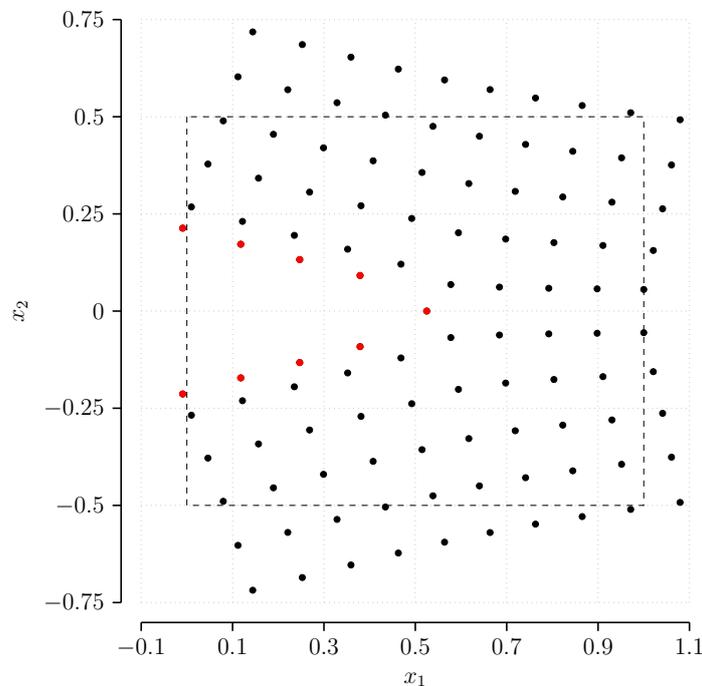


Figura 7.31 – Configuração deformada da placa, para $a/w = 0.5$, sob carregamento em modo-I. Os nós vermelhos representam os nós sobrepostos ao longo da trinca e na ponta da trinca.

É importante ressaltar a alta precisão obtida com essa análise, considerando uma configuração nodal simples, sem qualquer refinamento ao redor da ponta da trinca. Essa é uma consequência direta da técnica da subtração da singularidade, implementada no GSMF, que resulta em campos de tensão regularizados ao redor da ponta da trinca.

Outra grande vantagem do GSMF, quando comparado com outros métodos sem malha, é a composição da sua matriz de rigidez, que é esparsa e com valores em banda, o que contribui para sua alta eficiência; advinda do processo de construção nodal da mesma.

As seções seguintes apresentam a nova abordagem de otimização proposta, aplicada no exemplo corrente.

7.3.1.1 - Equivalência entre o *compliance* e os fatores de intensidade de tensão

Os parâmetros do modelo GSMF utilizados na nova abordagem serão os mesmos empregados por Oliveira (2019), exceto que α_q não será constante (variará de 0.2 a 0.95) e fará parte das variáveis a serem otimizadas pelos MMAs. Somente um caso de a/w será otimizado ($a/w = 0.6$), uma vez que a quantidade de gráficos, tabelas, resultados, etc., apresentados com a nova abordagem de otimização é significativamente grande e deixaria o texto muito extenso. E, não haveria ganho significativo; seria mais uma redundância de resultados.

Nesse estudo de caso também serão seguidos os passos descritos na metodologia (Capítulo 6); situações excepcionais ou destoantes serão explanadas.

Os valores das variáveis e das funções objetivo do modelo possuem escalas (ordens de grandeza) diferentes, portanto, para que os resultados sejam apresentados de forma clara e inteligível é necessário escalar os dados de forma unificada. O resultado da escalagem dos dados pode ser visto na Figura 7.32, para a 1ª fase e, na Figura 7.33, para os resultados da otimização (2ª fase). Como pode ser observado, é notória a diferença na visualização que há entre os dados originais e os escalados.

Os resultados da otimização referentes à 1ª e 2ª fases estão descritos estatisticamente nas Tabelas 7.14 e 7.15, respectivamente. Esta descrição estatística compreende, na sequência dos cabeçalhos das linhas: quantidade de soluções, média, desvio padrão, valor mínimo, 1º quartil, 2º quartil (ou mediana), 3º quartil e valor máximo, respectivamente. Os valores também estão escalados para facilitar a apresentação dos resultados. Observa-se que foram encontradas 32 soluções únicas e não dominadas quando da busca pela frente de Pareto de referência (1ª fase) e 117 na otimização propriamente dita (2ª fase). A quantidade menor de soluções na frente de Pareto de referência se dá pelo fato de os algoritmos terem executado

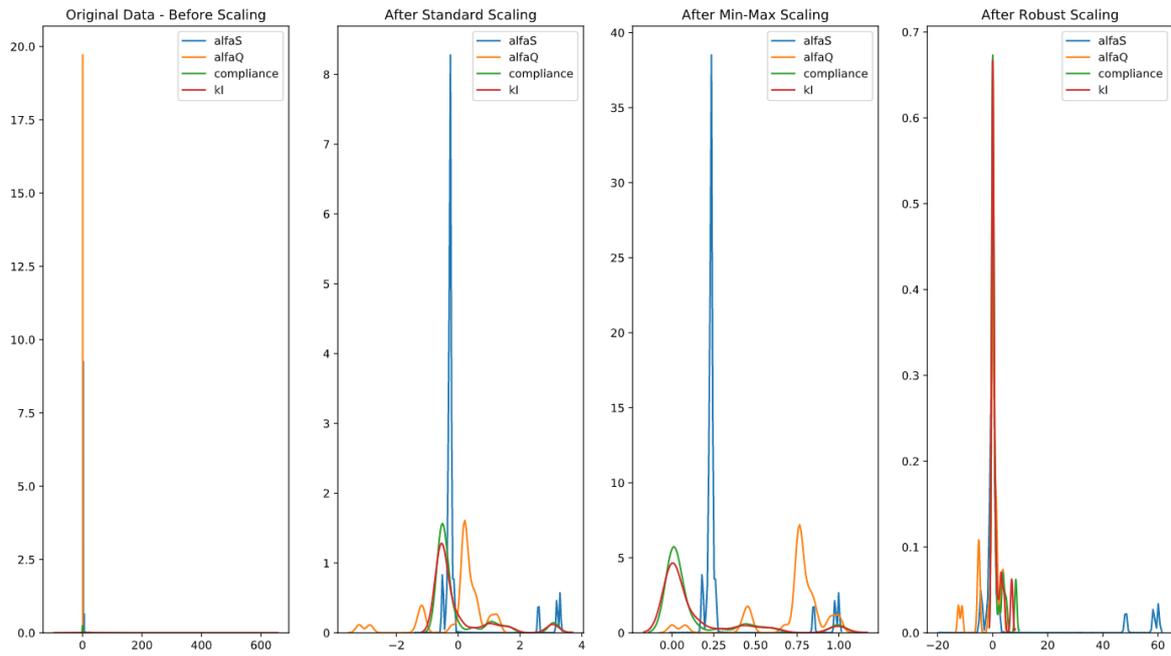


Figura 7.32 – Escalagem dos dados - 1ª fase.

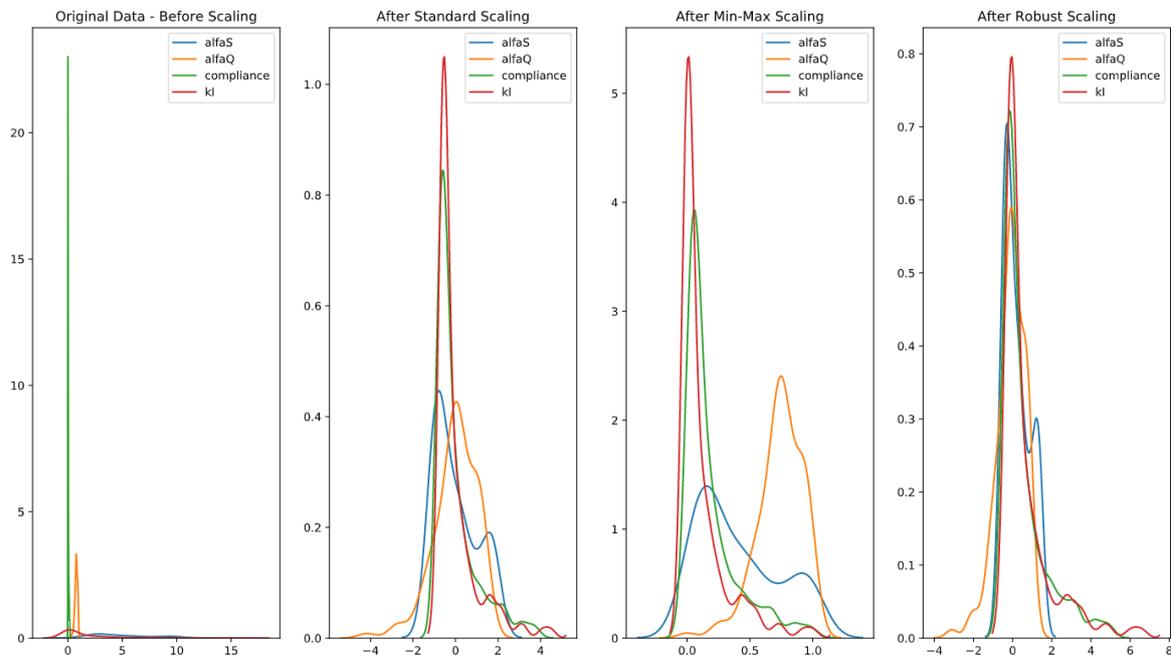


Figura 7.33 – Escalagem dos dados - 2ª fase.

mais iterações nessa fase e terem apurado a convergência, diminuindo assim as soluções aptas a integrarem a frente de Pareto.

Tabela 7.14 – Descrição estatística dos resultados escalados - 1ª fase.

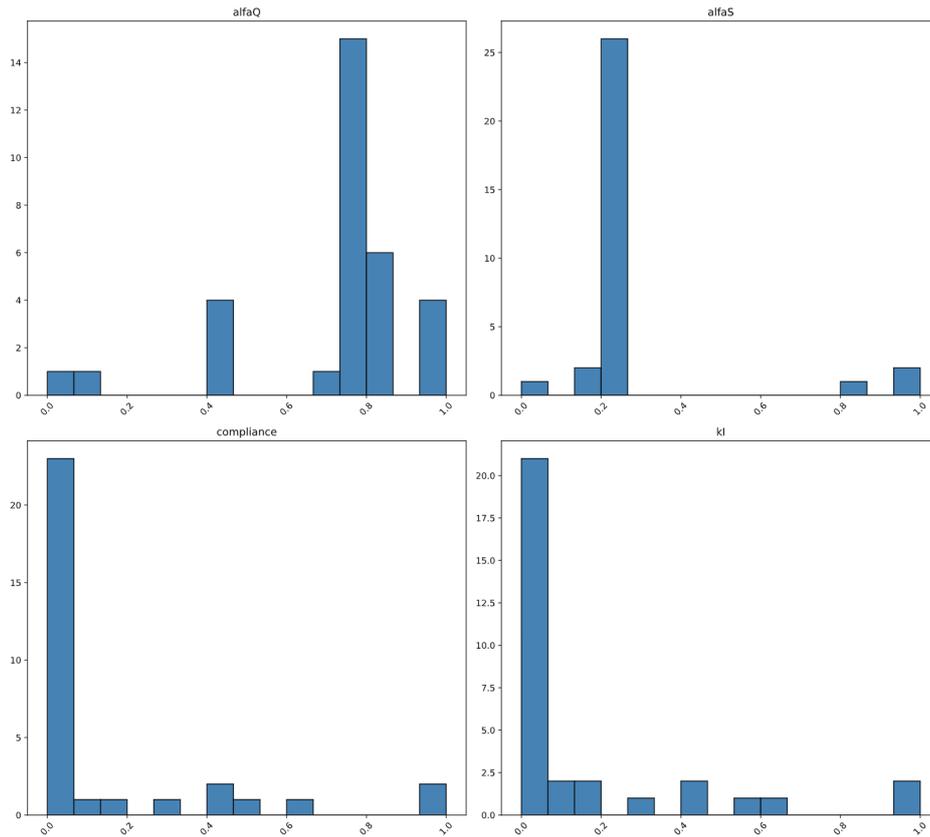
Statistic	alfaS	alfaQ	compliance	kI
count	32	32	32	32
mean	0.29037	0.71708	0.15014	0.15227
std	0.21837	0.22700	0.27709	0.27962
min	0.00000	0.00000	0.00000	0.00000
25%	0.22890	0.75779	0.00147	0.00002
50%	0.23485	0.76193	0.00794	0.00109
75%	0.24161	0.81903	0.11867	0.14346
max	1.00000	1.00000	1.00000	1.00000

Tabela 7.15 – Descrição estatística dos resultados escalados - 2ª fase.

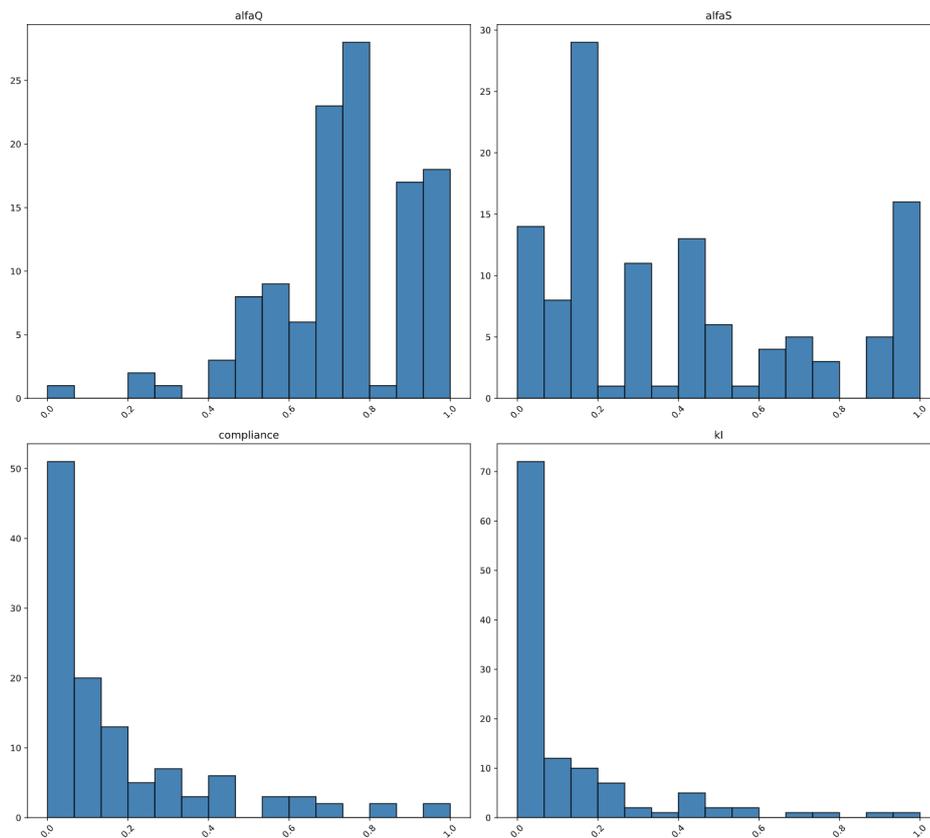
Statistic	alfaS	alfaQ	compliance	kI
count	117	117	117	117
mean	0.40630	0.74065	0.18514	0.11688
std	0.32149	0.17832	0.21573	0.19757
min	0.00000	0.00000	0.00000	0.00000
25%	0.14639	0.65842	0.04991	0.00057
50%	0.30037	0.76232	0.08435	0.01910
75%	0.65089	0.90377	0.23350	0.14972
max	1.00000	1.00000	1.00000	1.00000

As Figuras 7.34a e 7.34b apresentam as frequências relativas para cada variável e cada função objetivo, das soluções pertinentes às frentes de Pareto da 1ª e 2ª fases da otimização, respectivamente. Pode ser observado que em ambas as frentes de Pareto há uma concentração das soluções na região de valor mínimo para o *compliance* e o K_I , mostrando a eficiência dos MMAs na otimização de modelos ILMF.

Com base na fundamentação apresentada na Seção 5.4, foram gerados mapas de calor dos parâmetros da otimização em função do coeficiente de correlação de Pearson, como mostram as Figuras 7.35a e 7.35b. Aqui fica demonstrado estatisticamente que a função objetivo baseada no fator de intensidade de tensão K_I (SIF) pode ser substituída pela baseada na conformidade estrutural C , uma vez que o coeficiente de correlação entre C e K_I é 1.00 na 1ª fase de otimização e, de 0.81 na 2ª. Esses coeficientes indicam que os vetores que representam as variáveis em um espaço n-dimensional são praticamente colineares e possuem uma correlação muito forte. No caso da 1ª fase, é uma correlação perfeita positiva. Depreende-se também que os MMAs são muito robustos na otimização dos modelos ILMF, pois, mesmo com poucas iterações na 2ª fase de otimização, a correlação obtida entre C e K_I foi forte, como esperado. Presume-se também que para um



(a) Frente de Pareto - 1ª fase.



(b) Frente de Pareto - 2ª fase.

Figura 7.34 – Frequências relativas dos parâmetros da otimização.

número de iterações muito maior, esse coeficiente tende a 1, ou seja, uma correlação perfeita positiva, igual na 1ª fase.

Nas Figuras 7.36 e 7.37, além de se observar se há correlação entre os parâmetros, através da linha de regressão linear que é plotada, também pode-se observar a forma da distribuição das soluções encontradas no processo de otimização. Confrontando duas variáveis quaisquer (uma da linha e outra da coluna) pode-se analisar imediatamente a relação que há entre elas. São gráficos emparelhados, do inglês: *pairwise plot*. Mais uma vez é notória a similaridade entre as funções objetivo baseadas no K_I e no *compliance*. Inclusive na forma em que seus valores são distribuídos nas soluções constituintes da frente de Pareto (vide os cruzamentos $K_I \times K_I$ e *compliance* \times *compliance*).

Uma vista ampliada da correlação dos dois parâmetros mais relevantes da otimização (*compliance* e K_I) é dada pelas Figuras 7.38a e 7.38b. Note a concentração de soluções na vizinhança do mínimo do *compliance* e do K_I (que são os resultados pretendidos, já que é feita uma minimização destes parâmetros), em ambas as fases da otimização.

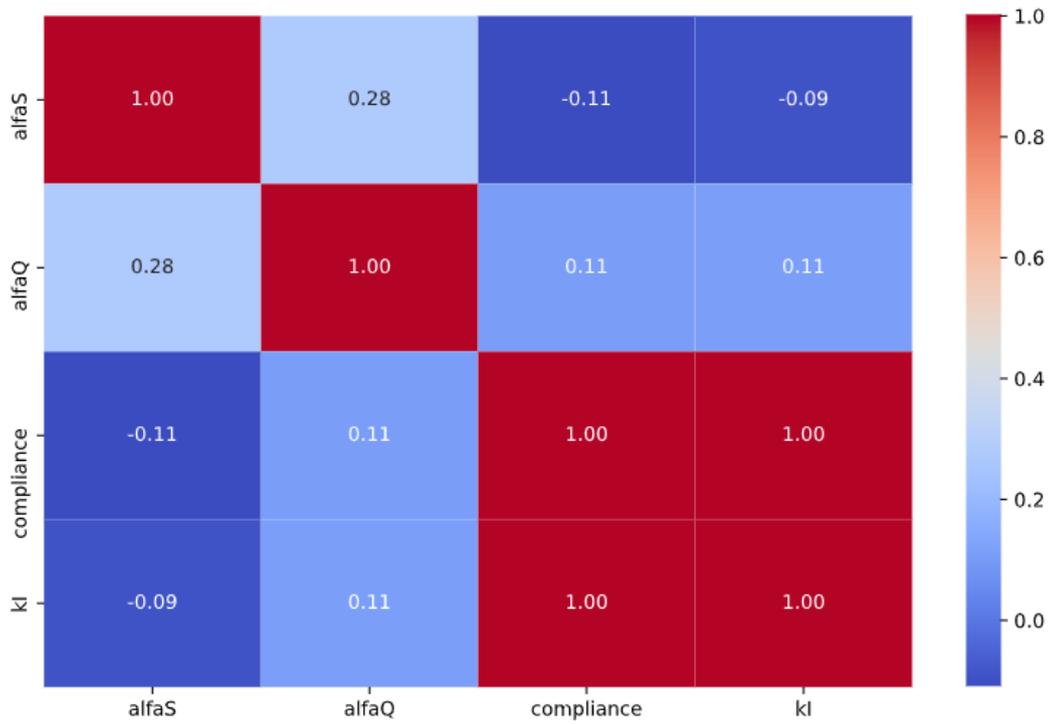
A plotagem dos resultados em coordenadas paralelas permite analisar o impacto que os valores dos parâmetros de otimização (variáveis e funções objetivo) exercem uns sobre os outros. Para isso, a Figura 7.39 apresentam plotagens em coordenadas paralelas para a 2ª fase da otimização. Lembrando que na 1ª fase não foi feita a discriminação das soluções por algoritmo/MMA.

7.3.1.2 - Influência da discretização nodal do domínio nos resultados do ILMF e da otimização

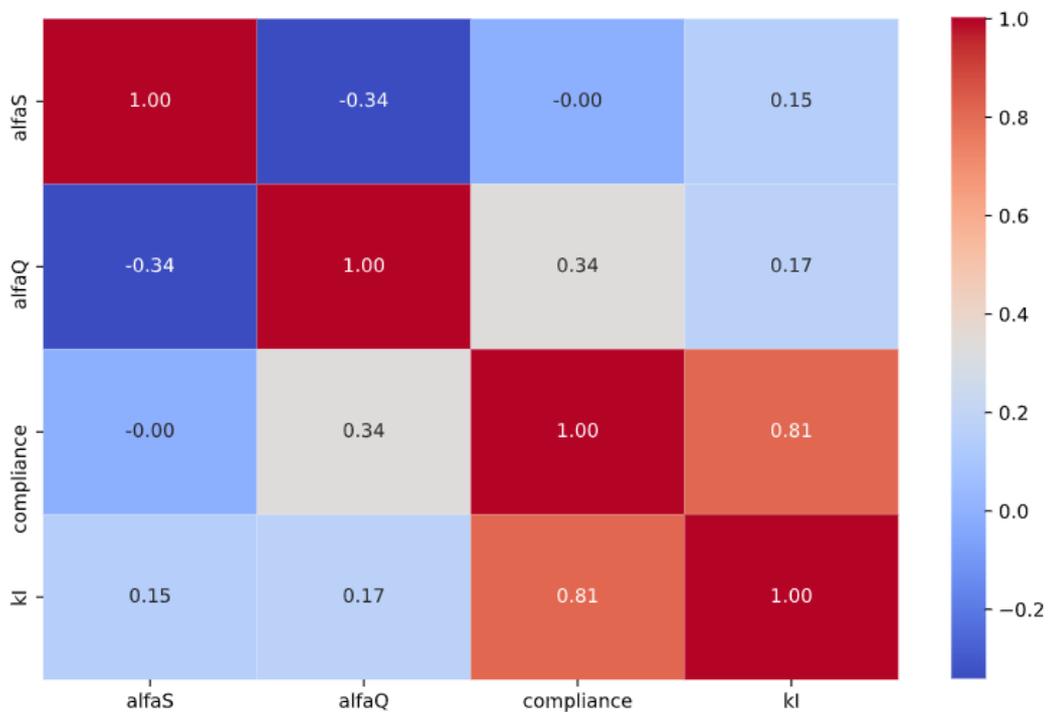
A Seção 7.2.2 (caso da viga engastada, em balanço) explorou a influência da densidade nodal e do tipo (regular ou randômica) da discretização do domínio, na solução e otimização do modelo ILMF e, não foram notados ganhos significativos na precisão dos resultados. Portanto, para o caso corrente será adotada somente a discretização regular, conforme Seção 7.3.1. Isto é uma grande vantagem propiciada pelo ILMF, pois a discretização regular facilita a implementação do método, melhora sua performance e simplifica a discretização do domínio do problema em estudo.

7.3.1.3 - Ranqueamento dos MMAs através da *utility function* e as métricas de performance

A Figura 7.40 mostra a quantidade de soluções únicas, não dominadas e ótimas de Pareto, geradas por algoritmo utilizado. Os algoritmos CMAES, OMOPSO e EpsMOEA, não geraram um número significativo de soluções na frente de Pareto, quando comparado com



(a) Resultados - 1ª fase.



(b) Resultados - 2ª fase.

Figura 7.35 – Mapa de calor dos parâmetros da otimização em função do coeficiente de correlação.

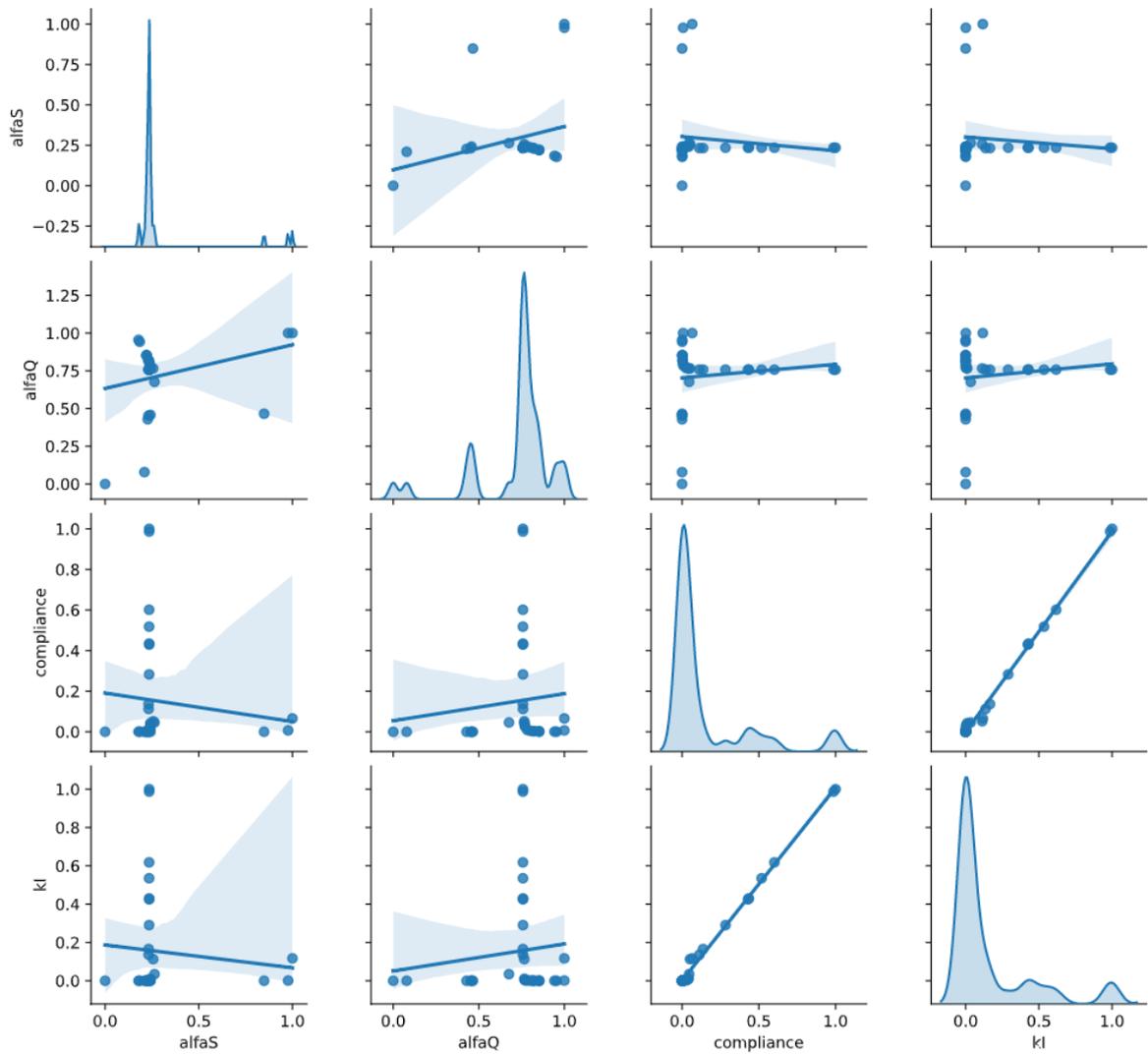


Figura 7.36 – *Pairwise plot* - 1ª fase.

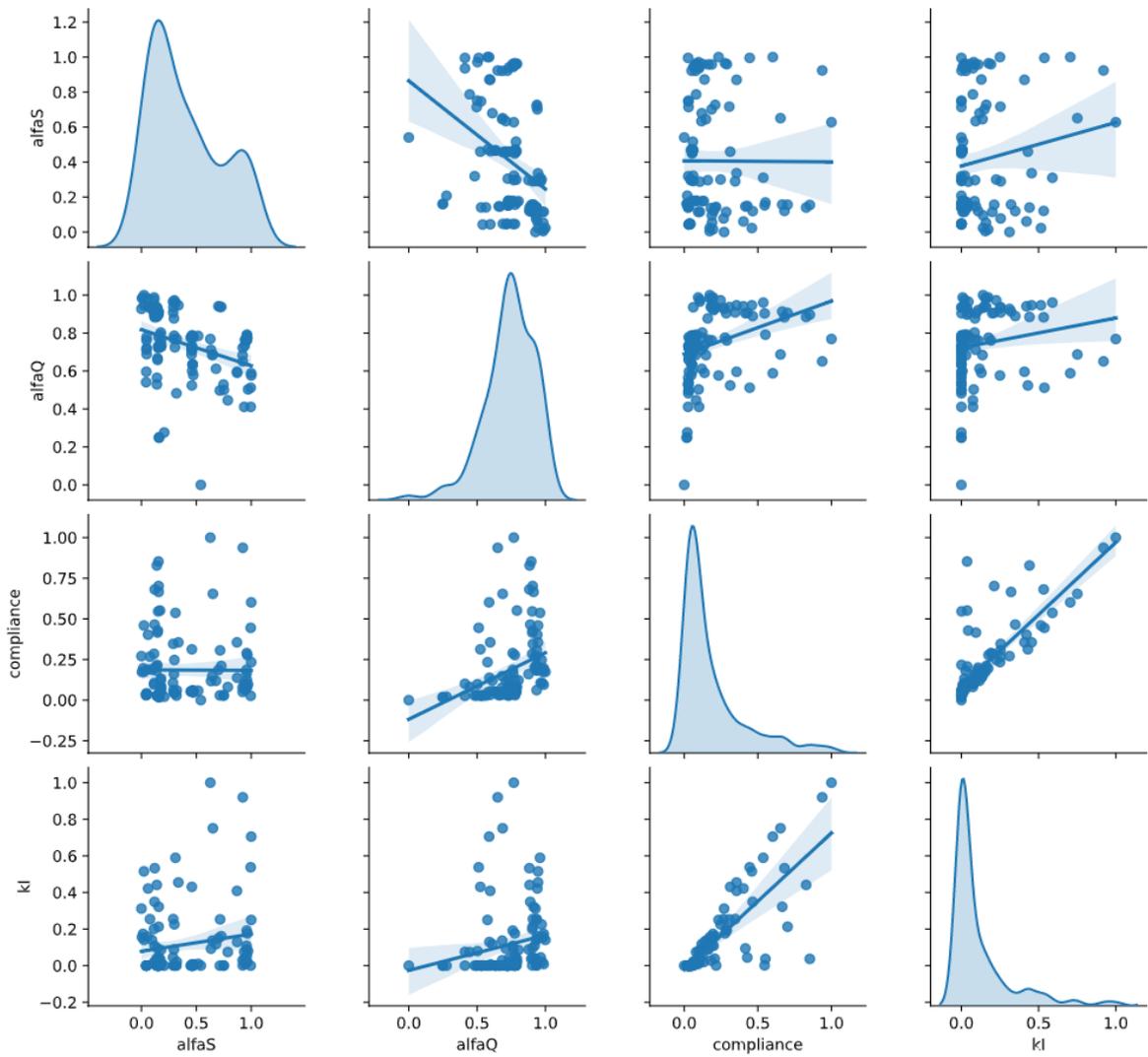
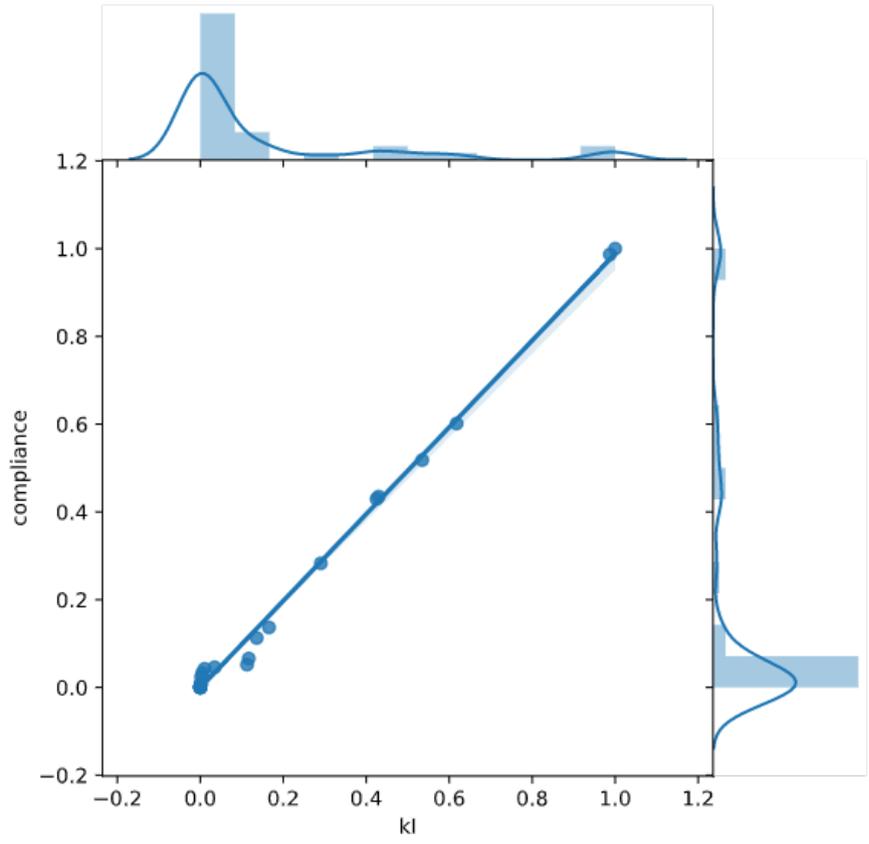
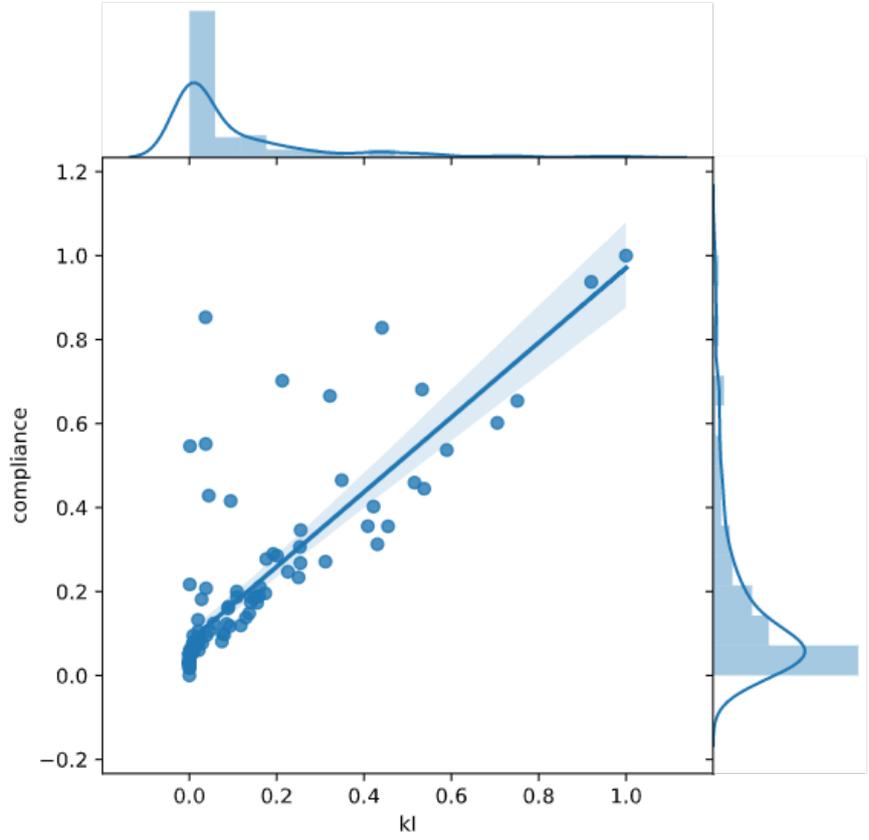


Figura 7.37 – Pairwise plot - 2^a fase.



(a) Resultados - 1ª fase.



(b) Resultados - 2ª fase.

Figura 7.38 – Correlação entre K_I e *compliance*.

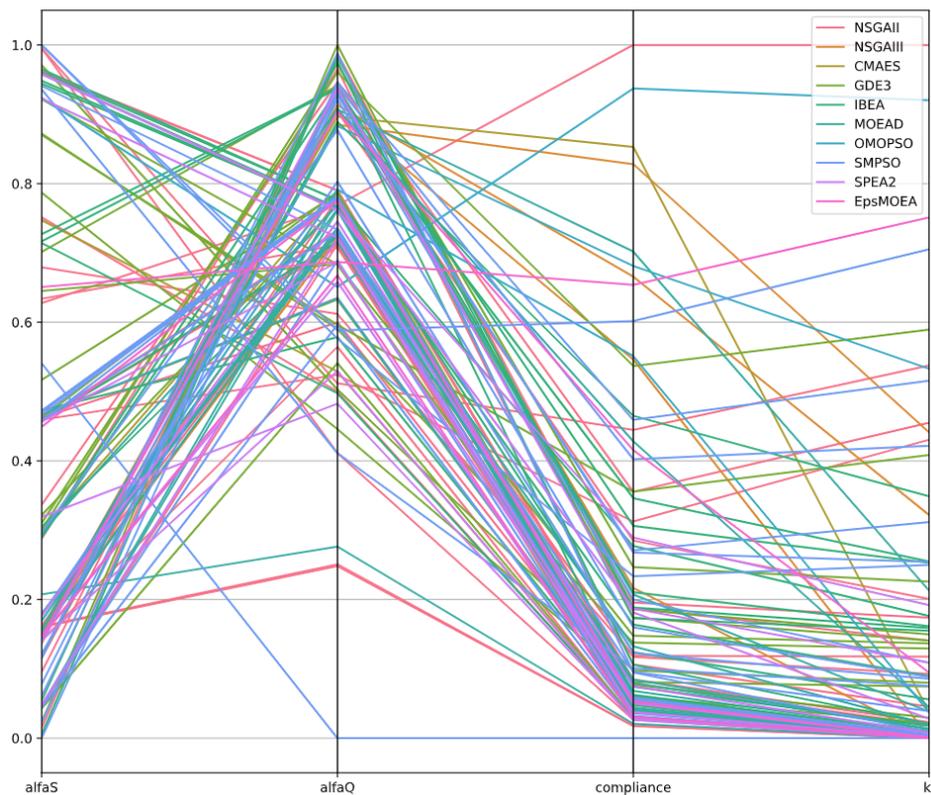


Figura 7.39 – Coordenadas paralelas dos parâmetros de otimização - 2ª fase.

os demais MMAs. Nesse exemplo, diferente do que ocorreu no exemplo anterior, o MOEAD apresentou um número significativo de soluções na frente de Pareto.

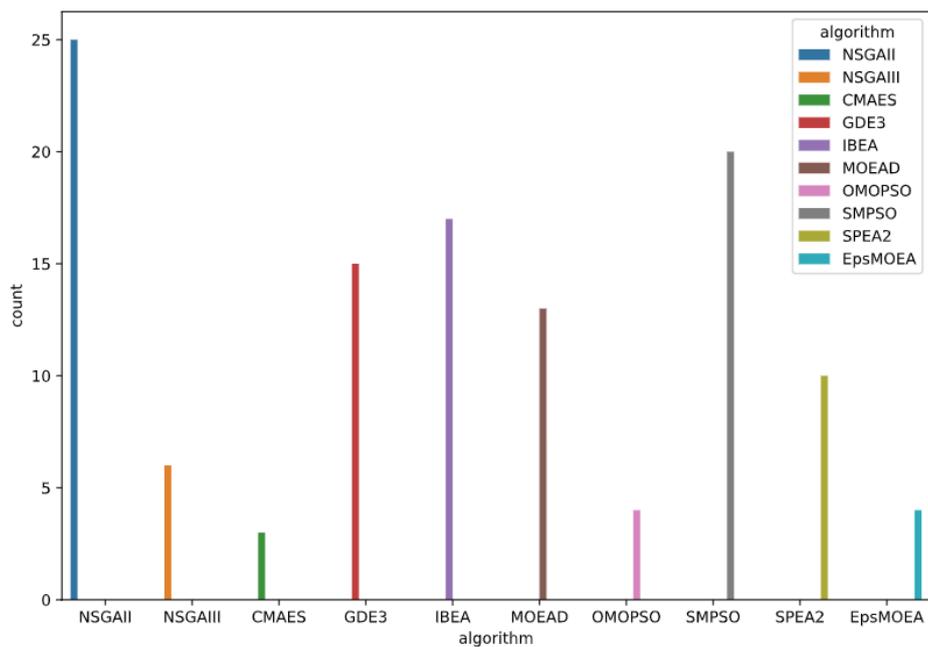


Figura 7.40 – Quantidade de soluções únicas, não dominadas e ótimas de Pareto, por MMA utilizado - 2ª fase.

A Figura 7.41 apresenta gráficos emparelhados (*pairwise plot*) dos parâmetros da otimização, 2ª fase, onde as soluções da frente de Pareto são discriminadas por MMA. Todos os MMAs apresentaram soluções satisfatórias quanto à precisão e exploraram bem o espaço de soluções, apresentando distribuições com formas muito semelhantes, como pode ser visto nos gráficos da diagonal descendente (do canto superior esquerdo ao canto inferior direito). Exceto o OMOPSO, o NSGAIII e o EpsMOEA que destoaram um pouco nesse quesito.

Uma observação mais detalhada é propiciada pelas Figuras 7.42a e 7.42b, onde as inclinações muito próximas das retas de regressão linear, obtidas a partir das soluções para cada frente de Pareto, por MMA, confirmam a boa performance dos MMAs empregados na otimização. Pode-se observar também, que tanto o espaço das variáveis (α_S e α_Q), quanto o espaço das soluções (K_I e *compliance*) foram explorados pelos MMAs em regiões muito próximas, refletindo uma quase colinearidade das retas de regressão linear.

A Tabela 7.16 traz as métricas da otimização, que foram obtidas comparando-se as soluções da 2ª fase da otimização com as soluções de referência (1ª fase). Nessa tabela contém também o ranqueamento dos MMAs para esse exemplo (mecânica da fratura linear elástica - placa com trinca de borda - Modo-I). O ranqueamento foi feito aplicando-se uma *utility function* (Seção 2.1.9) nas métricas de performance individuais por algoritmo. Então, os três MMAs com melhor desempenho na otimização do ILMF aplicado nesse exemplo, foram: GDE3, SPEA2 e NSGAIII (3ª posição, como no exemplo anterior, demonstrando consistência), nesta ordem. Isso não desabilita o uso dos demais MMAs na otimização desse tipo de modelo, uma vez que, conforme demonstrado, obtiveram excelentes e precisas soluções na frente de Pareto. Tanto que há uma alternância dos algoritmos que dominam cada métrica, por exemplo, o GDE3 obteve o maior hypervolume (preferível), o EpsMOEA obteve os menores IGD e Epsilon (preferível), e assim por diante. Mas, como definido nessa pesquisa, o que importa é o cálculo final da *utility function* que considera todas as métricas.

Tabela 7.16 – *Ranking* dos MMAs a partir das métricas de performance escaladas.

Ranking	Algorithm	Hypervolume	GD	IGD	Epsilon	Spacing	Utility
1	GDE3	0.8656712572	0.0426910447	0.1020742670	0.1343058678	0.2179337017	-0.3686663760
2	SPEA2	0.5695950453	0.0096541027	0.4206989010	0.4304107163	0.1262666313	0.4174353061
3	NSGAIII	0.5868573506	0.2115565828	0.3761642865	0.4131638299	0.2428852917	0.6569126403
4	EpsMOEA	1.0000000000	1.0000000000	0.0000000000	0.0000000000	1.0000000000	1.0000000000
5	IBEA	0.1359284303	0.0154551700	0.7729366756	0.8640748242	0.0786507277	1.5951889672
6	MOEAD	0.1468883046	0.0615584392	0.8551451103	0.8531183072	0.1989526686	1.8218862207
7	SMP SO	0.1266186616	0.1036694117	0.8424272308	0.8733857971	0.1307805278	1.8236443058
8	NSGAI	0.0113569033	0.0000000000	0.9882721038	0.9886434503	0.0000000000	1.9655586508
9	CMAES	0.3143836812	0.3342434905	0.5965594152	0.6856303253	0.6854895604	1.9875391101
10	OMOPSO	0.0000000000	0.2229688787	1.0000000000	1.0000000000	0.2719667438	2.4949356225

A Figura 7.43 mostra essas métricas plotadas em coordenadas paralelas, para uma

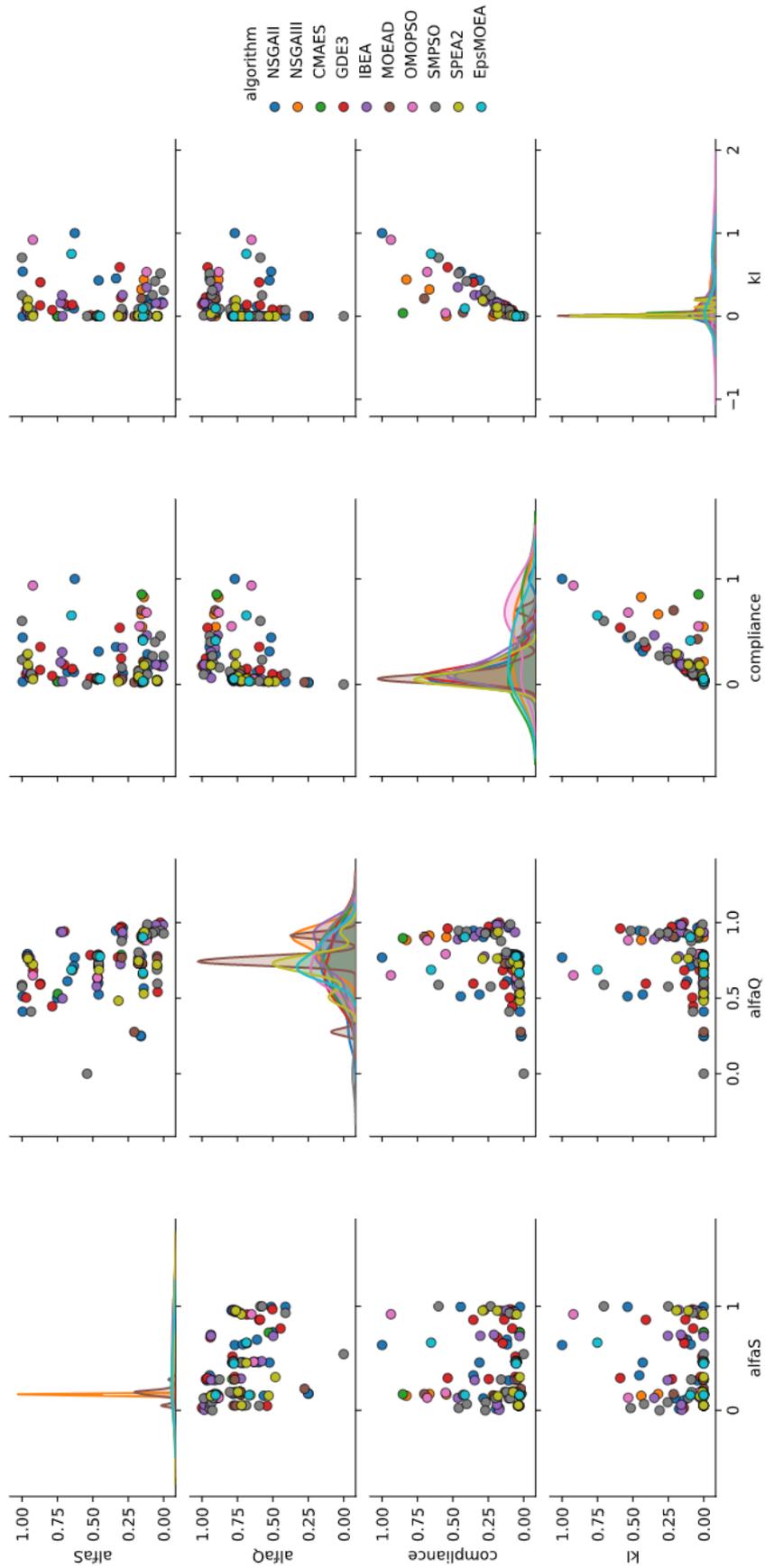
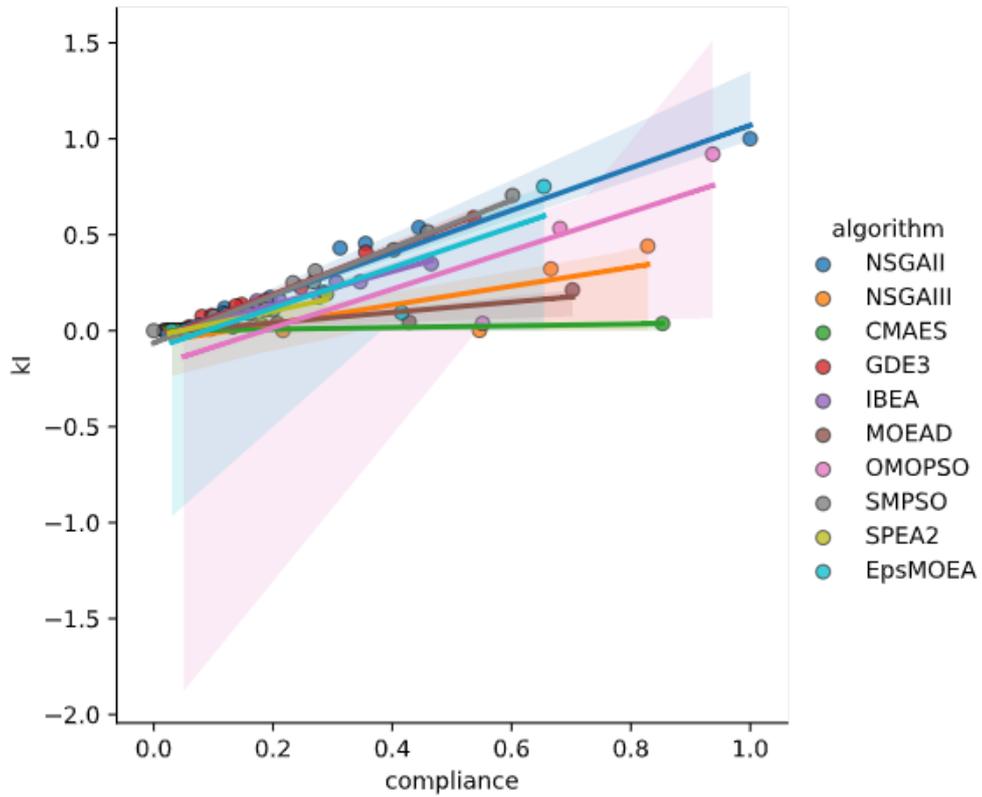
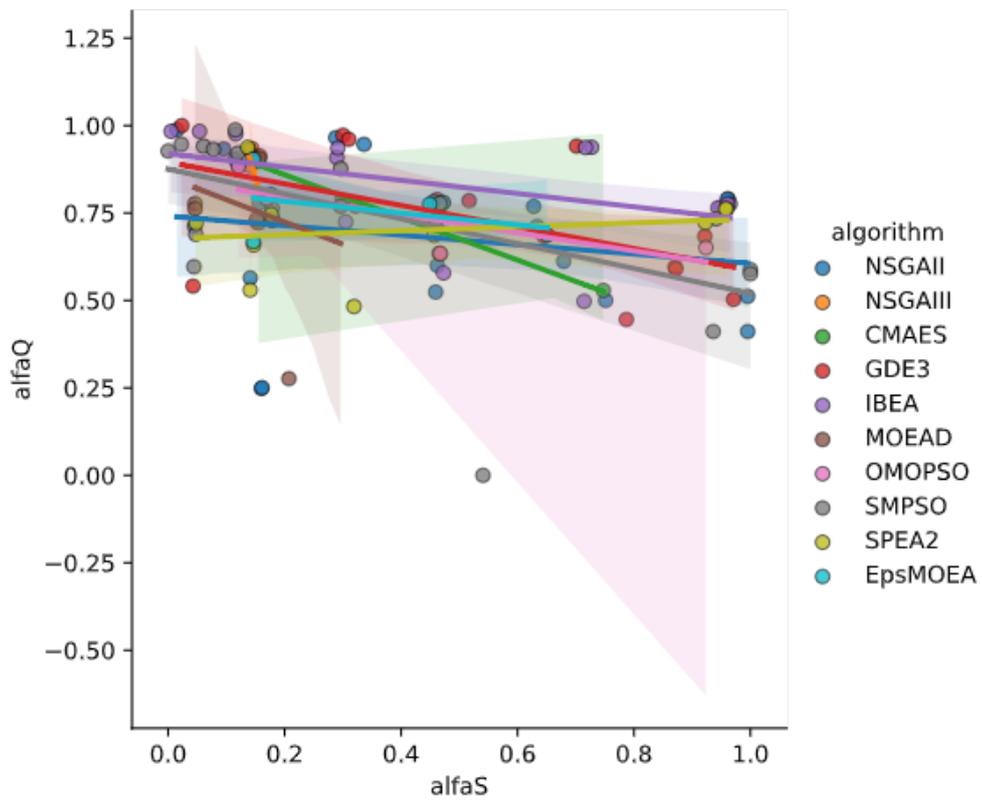


Figura 7.41 – *Pairwise plot* dos parâmetros da otimização, 2ª fase, com as soluções discriminadas por MMA.



(a) Espaço das soluções: K_I x *compliance* - 2ª fase.



(b) Espaço das variáveis: alfaQ x alfaS - 2ª fase.

Figura 7.42 – Exploração dos espaços das variáveis e das soluções pelos MMAs.

visualização mais inteligível.

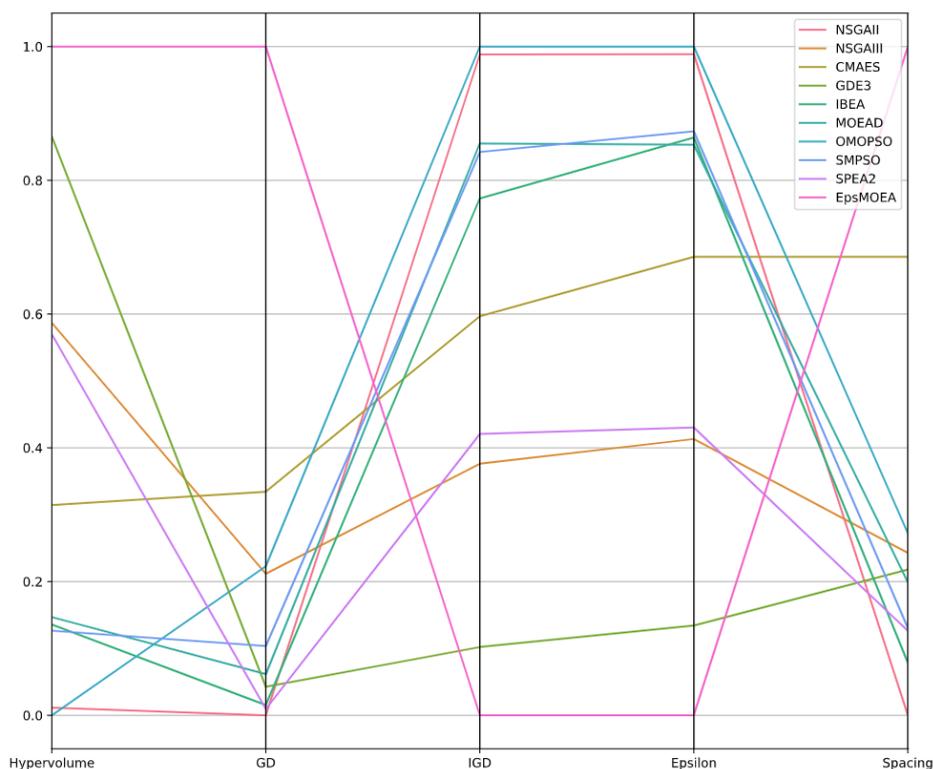


Figura 7.43 – Métricas de performance dos MMAs plotadas em coordenadas paralelas.

A função objetivo que leva em conta o tempo de processamento (*CPU time*) de cada iteração, para cada algoritmo, foi desconsiderada nesse exemplo, pois, como os MMAs possuem heurística intrínseca, tendem enviesar as soluções favorecendo as que gastam menos tempo computacional, mas, que não necessariamente fornecem bons resultados para as outras funções objetivo consideradas mais relevantes, que é o caso das relativas ao K_I e *compliance*. Portanto, somente duas funções objetivo foram utilizadas nesse exemplo: as relativas ao *compliance* e ao K_I . E não três, (*compliance* + K_I + *CPU time*) como tentado inicialmente.

7.3.2 - Placa com Trinca Horizontal de Borda - Modo-II

Uma placa retangular com uma trinca de borda, esquematicamente representada na Figura 7.44, foi considerada. O comprimento da trinca é dado por a e a relação entre a largura e o comprimento da placa é dada por $h/w = 0.5$. A placa é carregada por uma tração uniforme $\bar{t} = \sigma$, paralela à trinca e é aplicada anti-simetricamente nas laterais, caracterizando assim um carregamento de modo-II.

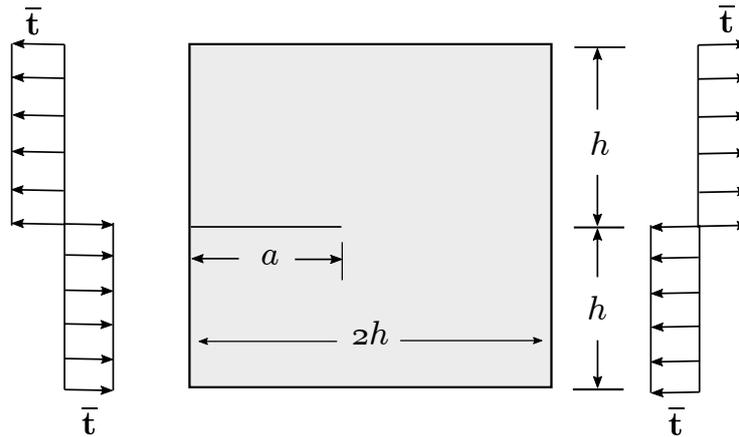


Figura 7.44 – Placa retangular com uma única trinca de borda sob carregamento em modo-II ($w = 2h$).

Este é um exemplo muito difícil, do qual não existem resultados analíticos de referência publicados. Desta forma, os resultados obtidos com este problema, utilizando o GSMF implementado com a técnica da subtração da singularidade, foram comparados com os resultados obtidos com o J-DBEM, utilizando o *software* apresentado por Aliabadi e Portela (1999).

Cinco casos são considerados, com $a/w = 0.2, 0.3, 0.4, 0.5$ e 0.6 . Uma base polinomial de primeira ordem e funções ponderadoras do tipo *quartic spline* foram consideradas na aproximação com os MQM.

O GSMF foi modelado com domínios locais de colocação retangulares e parâmetros de discretização $\alpha_s = 2 a 3.5$ e $\alpha_q = 0.5$, obtidos por otimização numérica. A análise numérica foi realizada considerando-se a distribuição nodal representada na Figura 7.45, onde para todos os cinco casos de comprimento de trinca considerados; apenas a distribuição nodal ao longo da trinca foi modificada, sempre sem considerar nenhum refinamento ao redor da ponta da trinca.

Os resultados obtidos apresentados na Tabela 7.17 demonstram a alta precisão do método. Nessa análise, o fator de intensidade de tensão do modo-I é sempre menor do que 10^{-3} , tendo em vista que este é um problema de modo-II. A configuração deformada da placa está esquematicamente representada na Figura 7.46.

É importante ressaltar a alta precisão obtida com essa análise, considerando uma configuração nodal relativamente simples, sem qualquer refinamento ao redor da ponta da trinca. Esta é uma característica chave do GSMF implementado com a técnica da subtração da singularidade.

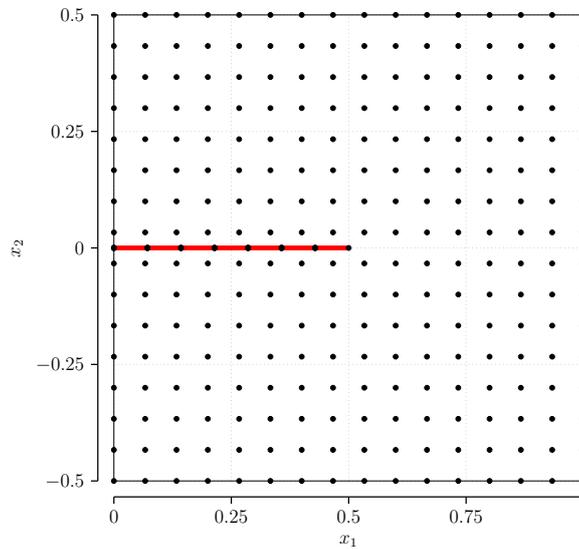


Figura 7.45 – Discretização de uma placa com uma distribuição nodal regular de $16 \times 16 = 256$ nós, com nós sobrepostos adicionais ao longo da trinca, para $a/w = 0.5$. A linha vermelha representa as faces da trinca.

Tabela 7.17 – Placa retangular com uma trinca de borda única sob carregamento em modo-II. GSMF representa os valores obtidos na presente pesquisa e J-DBEM representa os valores obtidos com o DBEM utilizando a integral J (Aliabadi e Portela, 1999). A porcentagem dos erros foi calculada a partir dos valores obtidos com o J-DBEM.

a/w	$K_{II}/(\bar{t}\sqrt{\pi a})$		
	GSMF	J-DBEM	% Erro
0.2	0.416	0.435	0.0436
0.3	0.338	0.358	0.0532
0.4	0.296	0.304	0.0261
0.5	0.248	0.262	0.0522
0.6	0.218	0.223	0.0218

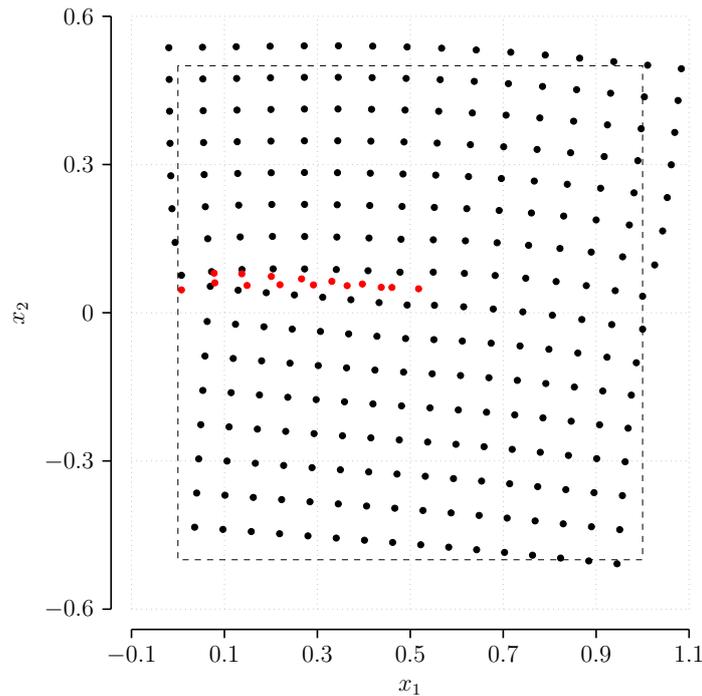


Figura 7.46 – Configuração deformada da placa, para $a/w = 0.5$, sob carregamento em modo-II. Os nós vermelhos representam os nós sobrepostos ao longo da trinca e na ponta da trinca.

A seções seguintes apresentam a nova abordagem de otimização proposta, aplicada no exemplo corrente.

7.3.2.1 - Equivalência entre o *compliance* e os fatores de intensidade de tensão

Os parâmetros do modelo GSMF utilizados na nova abordagem serão os mesmos empregados por Oliveira (2019), exceto que $\alpha_s = 1.5$ a 10.0 e α_q não será constante (variará de 0.20 a 0.95) e fará parte das variáveis a serem otimizadas pelos MMAs. Somente um caso de a/w será otimizado ($a/w = 0.2$), uma vez que a quantidade de gráficos, tabelas, resultados, etc., apresentados com a nova abordagem de otimização é significativamente grande e deixaria o texto muito extenso. E, não haveria ganho significativo; seria mais uma redundância de resultados.

Nesse estudo de caso também serão seguidos os passos descritos na metodologia (Capítulo 6); situações excepcionais ou destoantes serão explanadas.

Os valores das variáveis e das funções objetivo do modelo possuem escalas (ordens de grandeza) diferentes, portanto, para que os resultados sejam apresentados de forma clara e inteligível é necessário escalar os dados de forma unificada. O resultado da escalagem dos dados pode ser visto na Figura 7.47, para a 1ª fase e, na Figura 7.48, para os resultados da

otimização (2ª fase). Como pode ser observado, é notória a diferença na visualização que há entre os dados originais e os escalados.

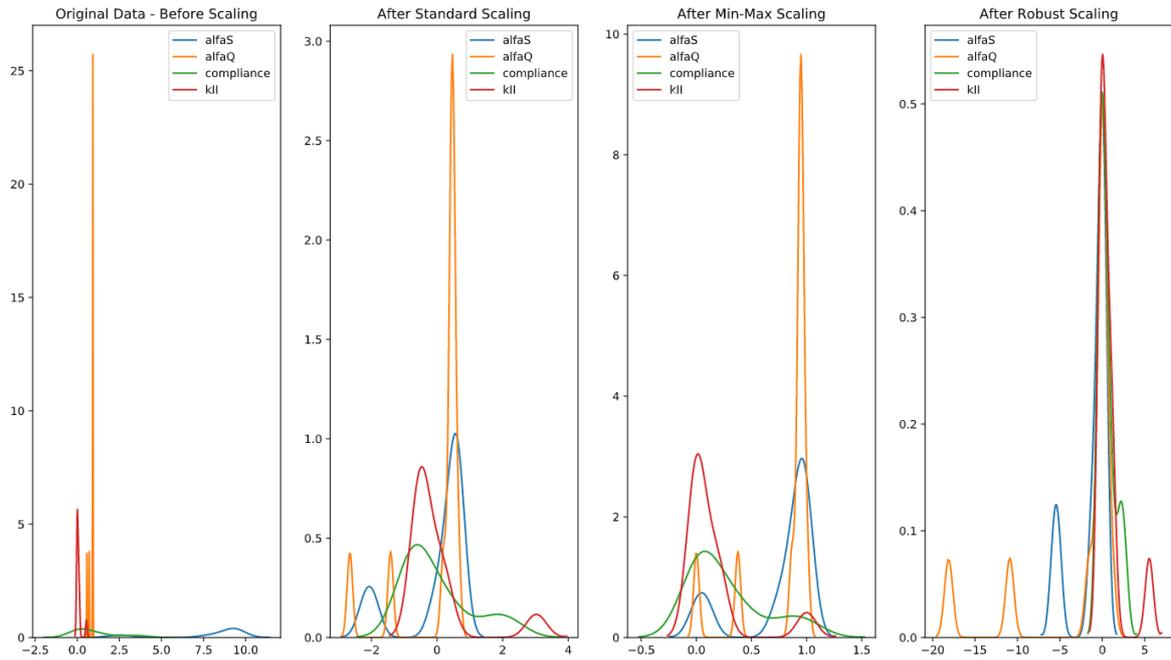


Figura 7.47 – Escalagem dos dados - 1ª fase.

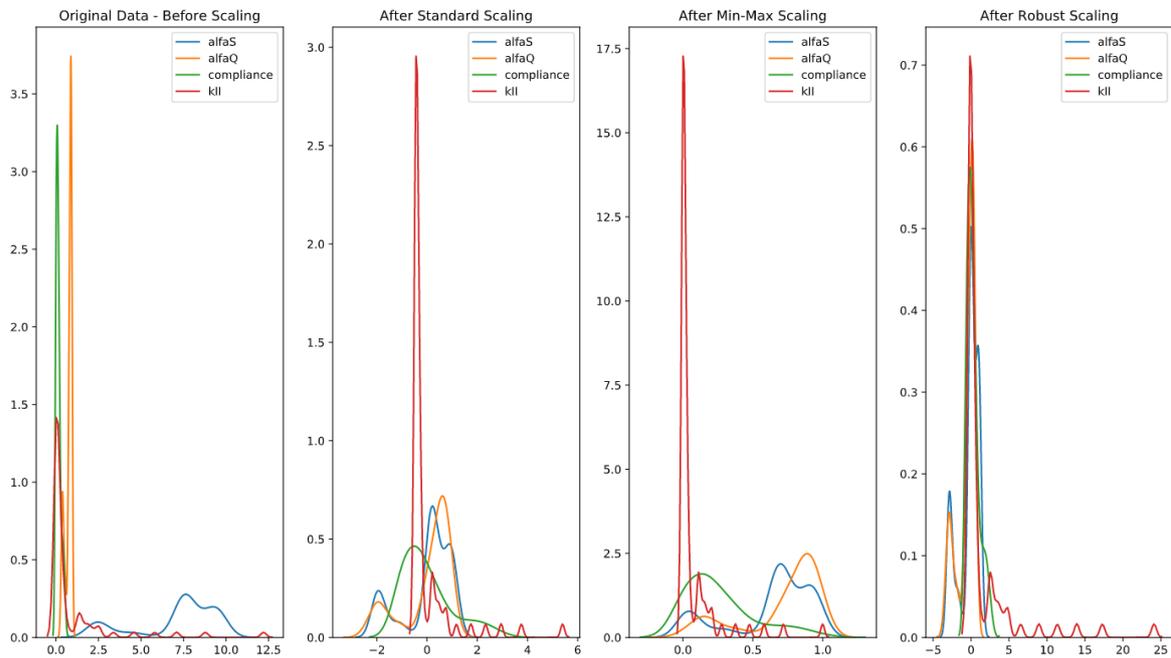


Figura 7.48 – Escalagem dos dados - 2ª fase.

Os resultados da otimização referentes à 1ª e 2ª fases estão descritos estatisticamente nas Tabelas 7.18 e 7.19, respectivamente. Esta descrição estatística compreende, na sequência dos cabeçalhos das linhas: quantidade de soluções, média, desvio padrão, valor mínimo, 1º quartil, 2º quartil (ou mediana), 3º quartil e valor máximo, respectivamente. Os valores

também estão escalados para facilitar a apresentação dos resultados. Observa-se que foram encontradas 11 soluções únicas e não dominadas quando da busca pela frente de Pareto de referência (1ª fase) e 73 na otimização propriamente dita (2ª fase). A quantidade menor de soluções na frente de Pareto de referência se dá pelo fato de os algoritmos terem executado mais iterações nessa fase e terem apurado a convergência, diminuindo assim as soluções aptas a integrarem a frente de Pareto.

Tabela 7.18 – Descrição estatística dos resultados escalados - 1ª fase.

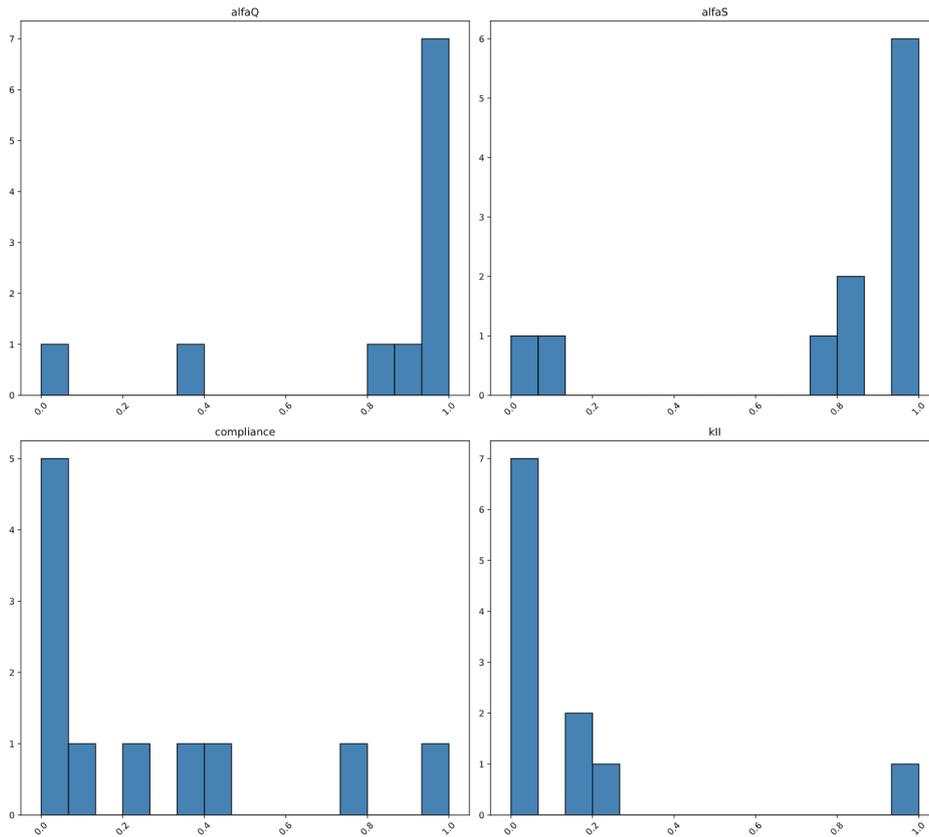
Statistic	alfaS	alfaQ	compliance	k_{II}
count	11	11	11	11
mean	0.76341	0.80621	0.27317	0.14505
std	0.36285	0.31838	0.34284	0.29634
min	0.00000	0.00000	0.00000	0.00000
25%	0.79520	0.89781	0.02877	0.00109
50%	0.96268	0.94641	0.08709	0.00781
75%	0.96311	0.95007	0.38681	0.18059
max	1.00000	1.00000	1.00000	1.00000

Tabela 7.19 – Descrição estatística dos resultados escalados - 2ª fase.

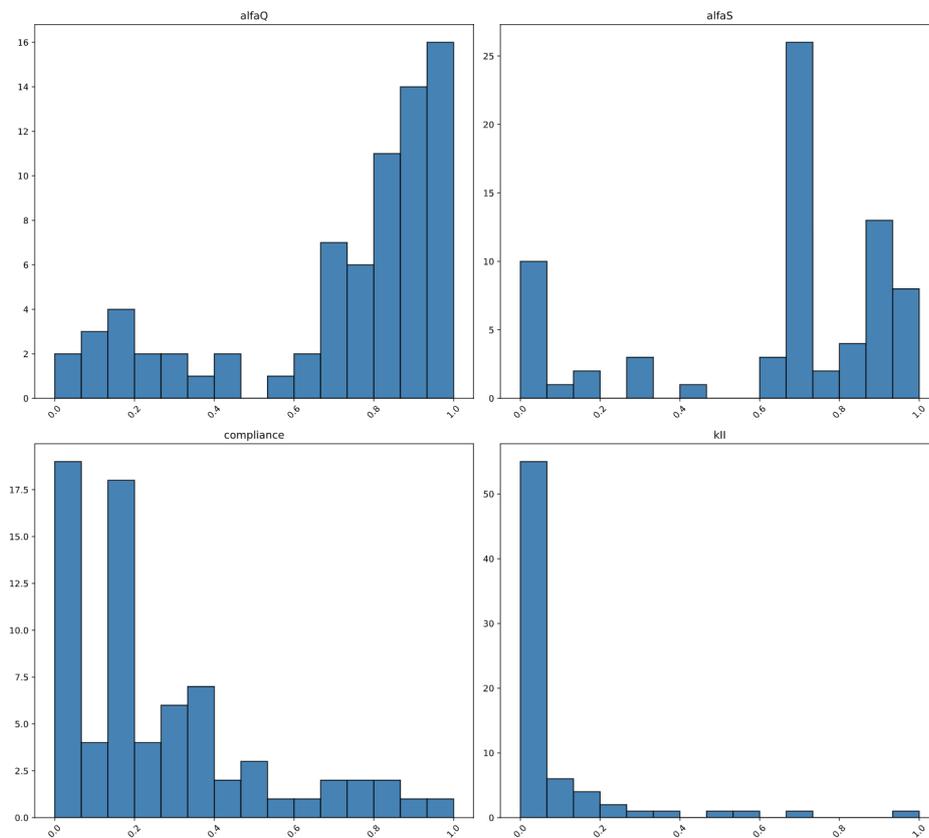
Statistic	alfaS	alfaQ	compliance	k_{II}
count	73	73	73	73
mean	0.63350	0.71190	0.25956	0.07678
std	0.30755	0.29095	0.24730	0.17222
min	0.00000	0.00000	0.00000	0.00000
25%	0.65473	0.66207	0.04765	0.00103
50%	0.69340	0.85394	0.17906	0.00762
75%	0.88435	0.90712	0.35182	0.04218
max	1.00000	1.00000	1.00000	1.00000

As Figuras 7.49a e 7.49b apresentam as frequências relativas para cada variável e cada função objetivo, das soluções pertinentes às frentes de Pareto da 1ª e 2ª fases da otimização, respectivamente. Pode ser observado que em ambas as frentes de Pareto há uma concentração das soluções na região de valor mínimo para o *compliance* e o K_{II} , mostrando a eficiência dos MMAs na otimização de modelos ILMF.

Com base na fundamentação apresentada na Seção 5.4, foram gerados mapas de calor dos parâmetros da otimização em função do coeficiente de correlação de Pearson, como mostram as Figuras 7.50a e 7.50b. Aqui fica demonstrado estatisticamente que a função objetivo baseada no fator de intensidade de tensão K_{II} (SIF) pode ser substituída pela baseada na conformidade estrutural C , uma vez que o coeficiente de correlação entre C e K_{II} é 0.86 na 1ª fase de otimização e, de 0.70 na 2ª. Esses coeficientes indicam que os vetores que representam as variáveis em um espaço n-dimensional são praticamente



(a) Frente de Pareto - 1ª fase.



(b) Frente de Pareto - 2ª fase.

Figura 7.49 – Frequências relativas dos parâmetros da otimização.

colineares e possuem uma correlação muito forte. No caso da 1ª fase, é uma correlação perfeita positiva. Depreende-se também que os MMAs são muito robustos na otimização dos modelos ILMF, pois, mesmo com poucas iterações na 2ª fase de otimização, a correlação obtida entre C e K_{II} foi forte, como esperado. Presume-se também que para um número de iterações muito maior, esse coeficiente tende a 1, ou seja, uma correlação perfeita positiva, igual na 1ª fase.

Nas Figuras 7.51 e 7.52, além de se observar se há correlação entre os parâmetros, através da linha de regressão linear que é plotada, também pode-se observar a forma da distribuição das soluções encontradas no processo de otimização. Confrontando duas variáveis quaisquer (uma da linha e outra da coluna) pode-se analisar imediatamente a relação que há entre elas. São gráficos emparelhados, do inglês: *pairwise plot*. Mais uma vez é notória a similaridade entre as funções objetivo baseadas no K_{II} e no *compliance*. Inclusive na forma em que seus valores são distribuídos nas soluções constituintes da frente de Pareto (vide os cruzamentos $K_{II} \times K_{II}$ e *compliance* \times *compliance*).

Uma vista ampliada da correlação dos dois parâmetros mais relevantes da otimização (*compliance* e K_{II}) é dada pelas Figuras 7.53a e 7.53b. Note a concentração de soluções na vizinhança do mínimo do *compliance* e do K_{II} (que são os resultados pretendidos, já que é feita uma minimização destes parâmetros), em ambas as fases da otimização.

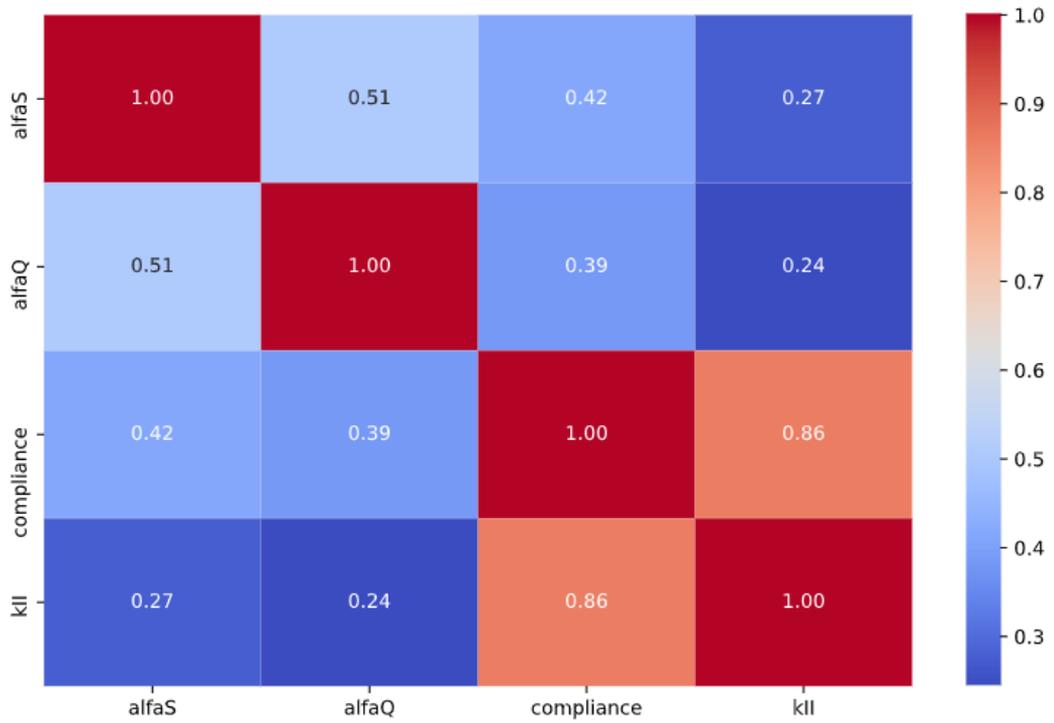
A plotagem dos resultados em coordenadas paralelas permite analisar o impacto que os valores dos parâmetros de otimização (variáveis e funções objetivo) exercem uns sobre os outros. Para isso, a Figura 7.54 apresentam plotagens em coordenadas paralelas para a 2ª fase da otimização. Lembrando que na 1ª fase não foi feita a discriminação das soluções por algoritmo/MMA.

7.3.2.2 - Ranqueamento dos MMAs através da *utility function* e as métricas de performance

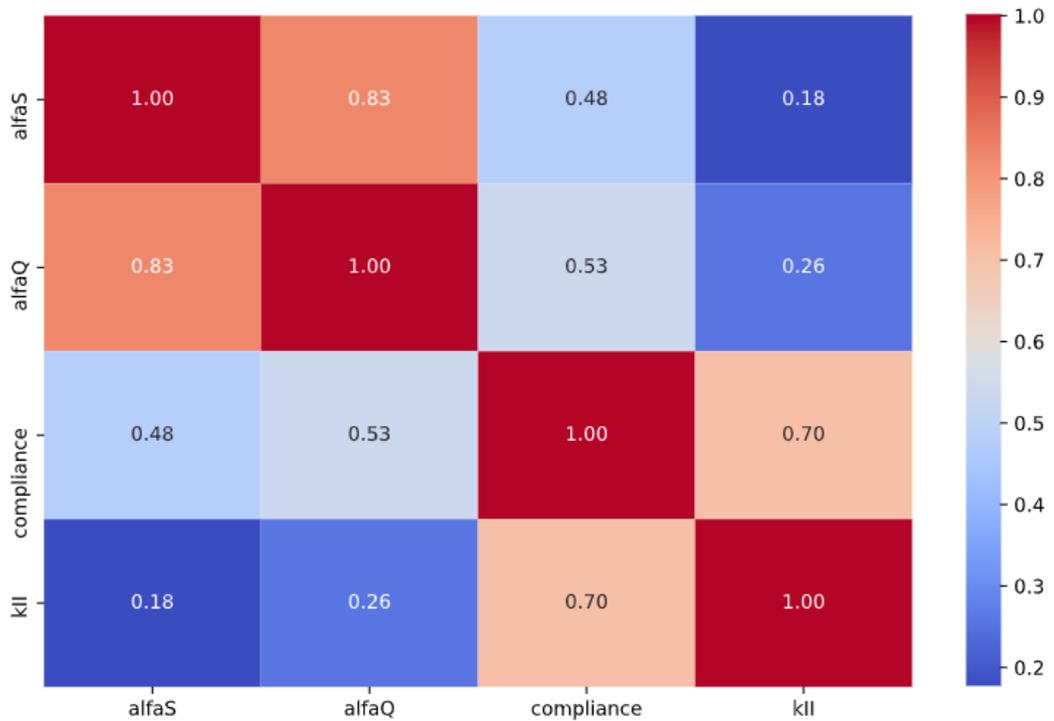
A Figura 7.55 mostra a quantidade de soluções únicas, não dominadas e ótimas de Pareto, geradas por algoritmo utilizado.

A Figura 7.56 apresenta gráficos emparelhados (*pairwise plot*) dos parâmetros da otimização, 2ª fase, onde as soluções da frente de Pareto são discriminadas por MMA. Todos os MMAs apresentaram soluções satisfatórias quanto à precisão e exploraram bem o espaço de soluções, apresentando distribuições com formas muito semelhantes, como pode ser visto nos gráficos da diagonal descendente (do canto superior esquerdo ao canto inferior direito).

Uma observação mais detalhada é propiciada pelas Figuras 7.57a e 7.57b, onde as



(a) Resultados - 1ª fase.



(b) Resultados - 2ª fase.

Figura 7.50 – Mapa de calor dos parâmetros da otimização em função do coeficiente de correlação.

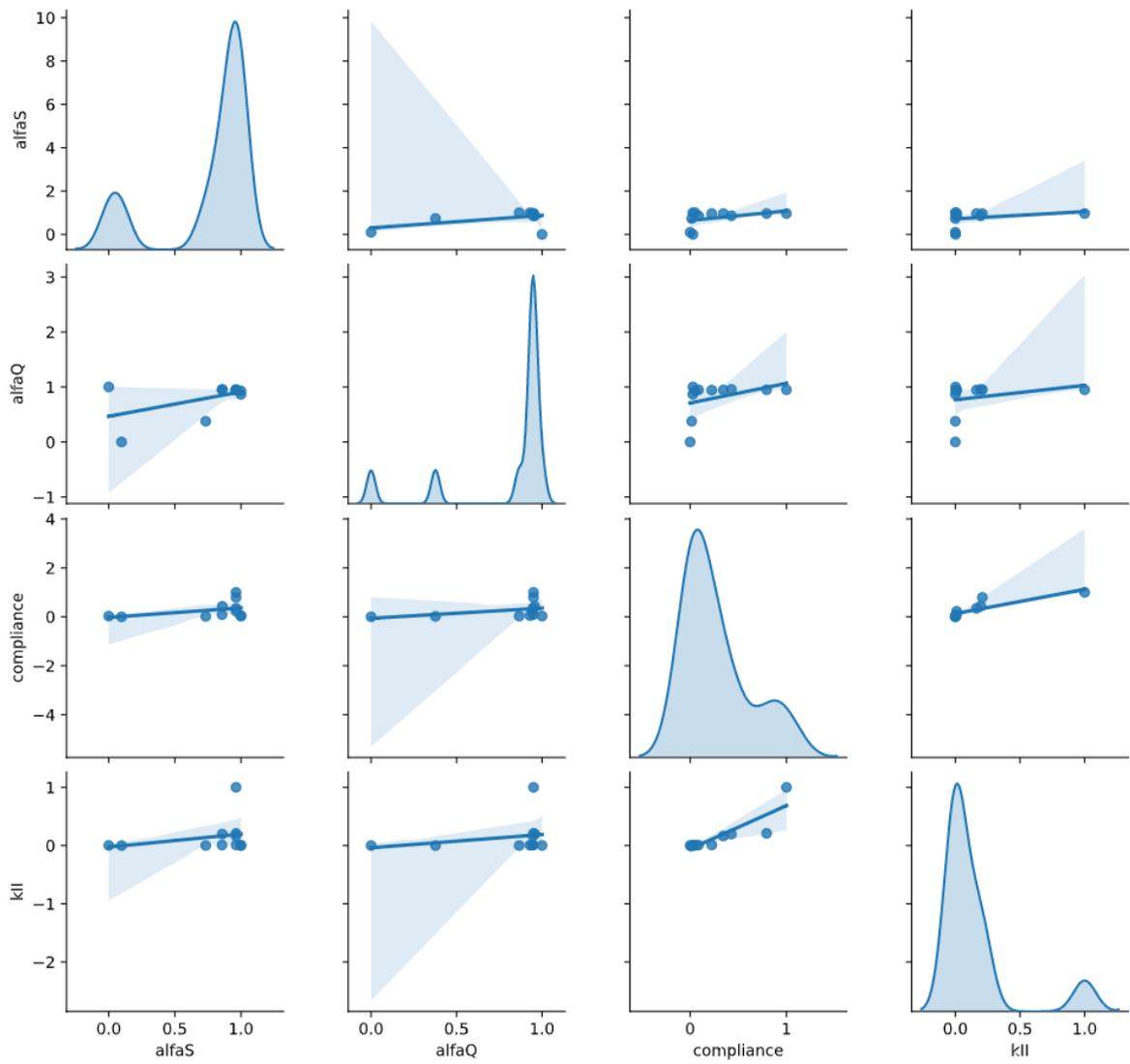


Figura 7.51 – *Pairwise plot* - 1ª fase.

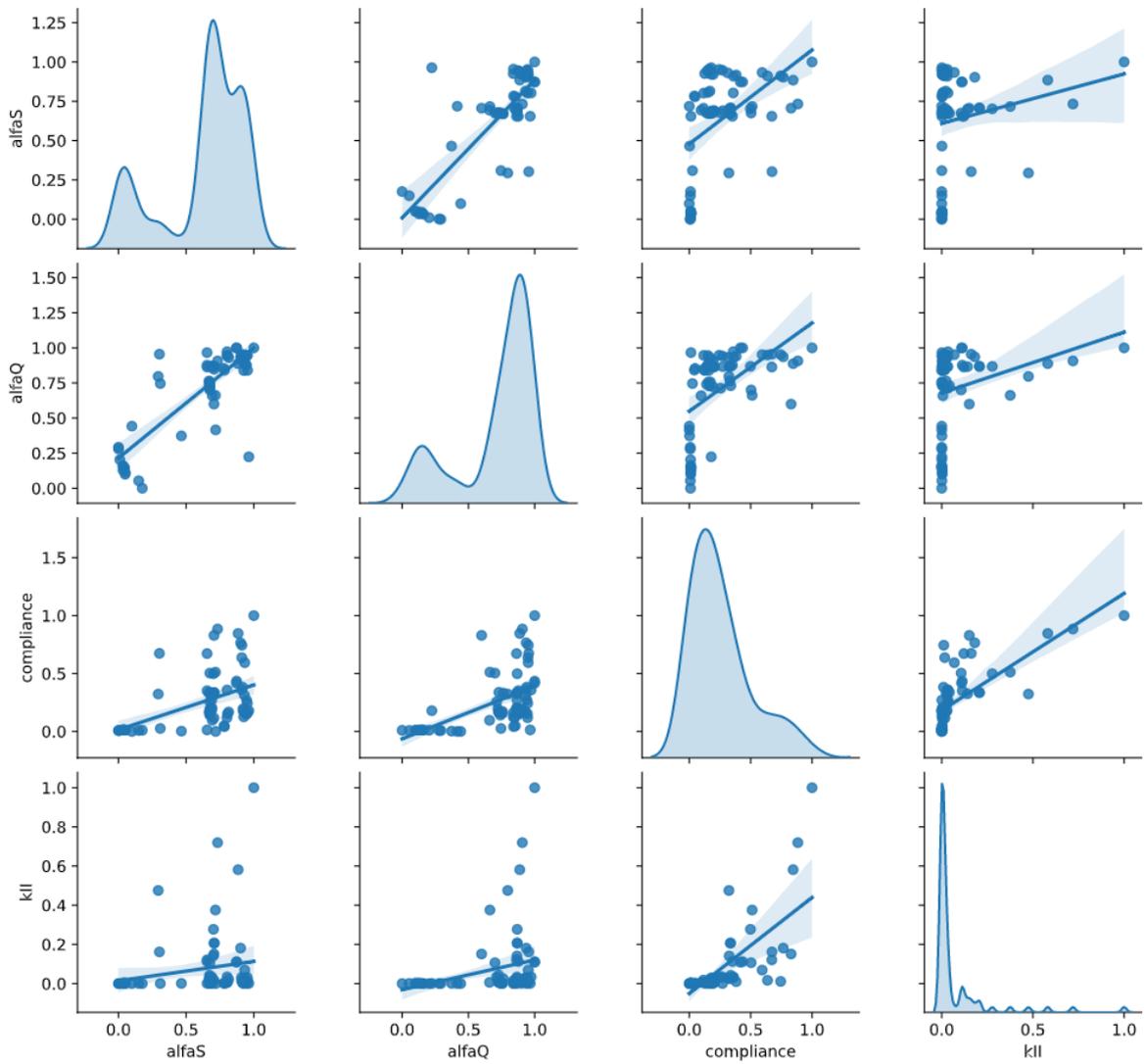
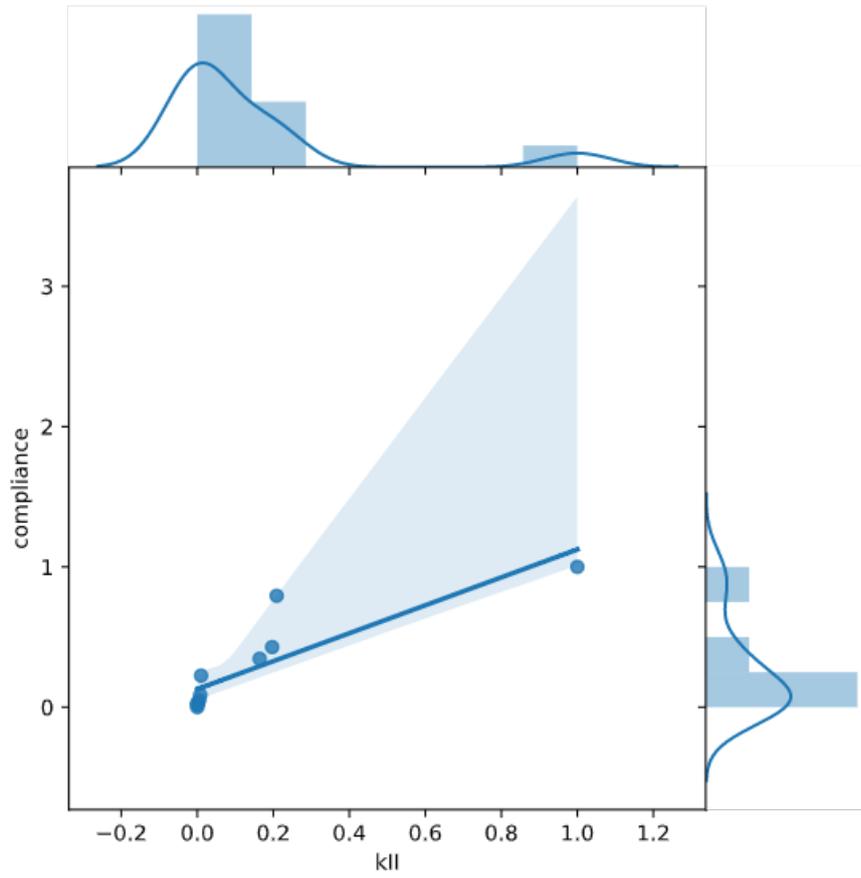
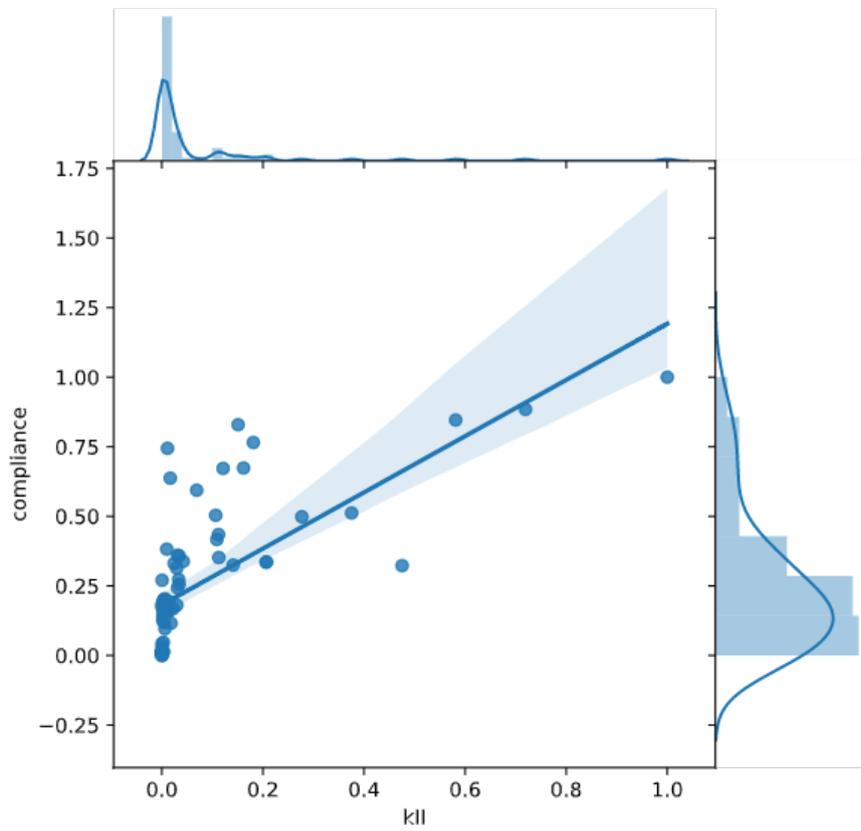


Figura 7.52 – Pairwise plot - 2^a fase.



(a) Resultados - 1ª fase.



(b) Resultados - 2ª fase.

Figura 7.53 – Correlação entre K_{II} e *compliance*.
163

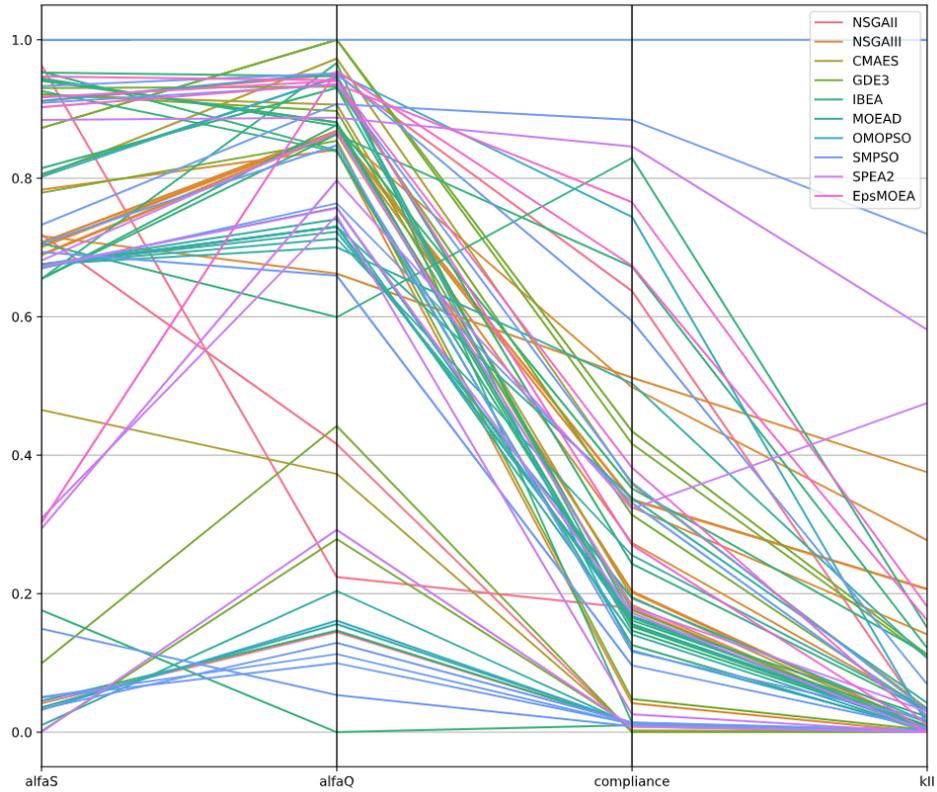


Figura 7.54 – Coordenadas paralelas dos parâmetros de otimização - 2ª fase.

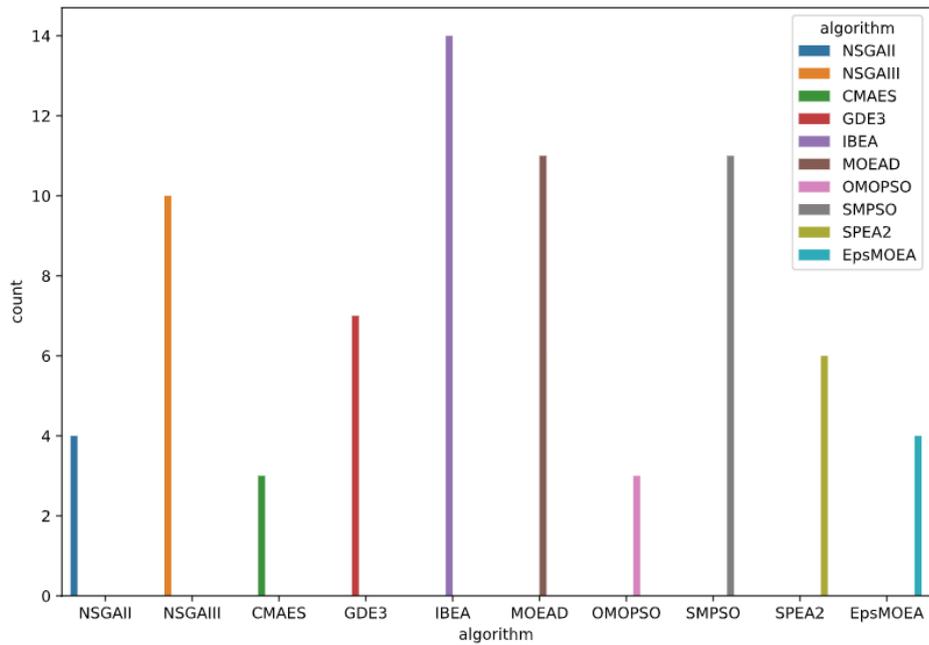


Figura 7.55 – Quantidade de soluções únicas, não dominadas e ótimas de Pareto, por MMA utilizado - 2ª fase.

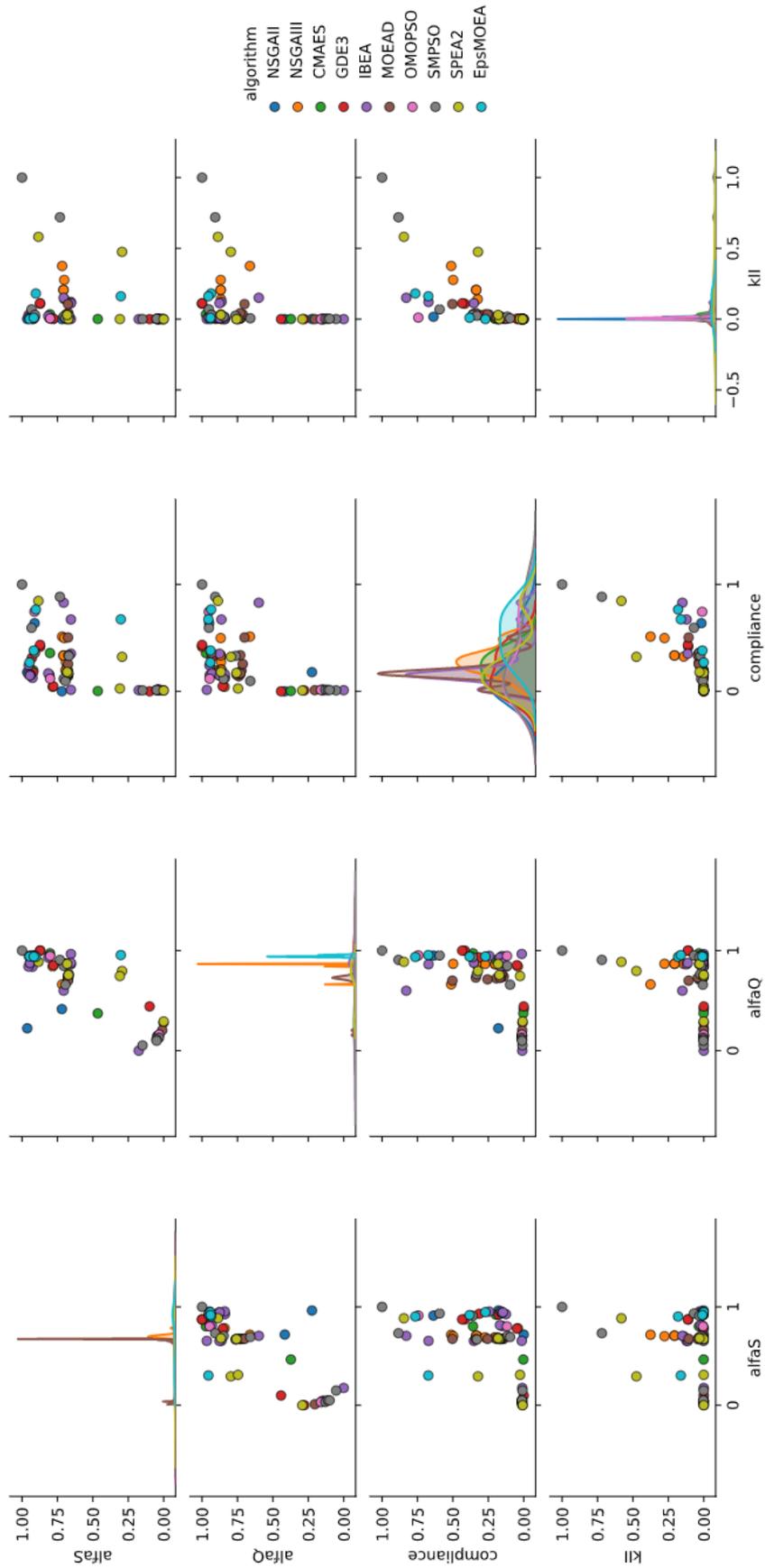


Figura 7.56 – *Pairwise plot* dos parâmetros da otimização, 2ª fase, com as soluções discriminadas por MMA.

inclinações muito próximas das retas de regressão linear, obtidas a partir das soluções para cada frente de Pareto, por MMA, confirmam a boa performance dos MMAs empregados na otimização. Pode-se observar também, que tanto o espaço das variáveis (alfaS e alfaQ), quanto o espaço das soluções (K_{II} e *compliance*) foram explorados pelos MMAs em regiões muito próximas, refletindo uma quase colinearidade das retas de regressão linear. Cabe ressaltar que alguns MMAs destoaram um pouco com relação à maioria, com inclinações um tanto discrepantes. Mas, mesmo assim, as soluções foram satisfatórias.

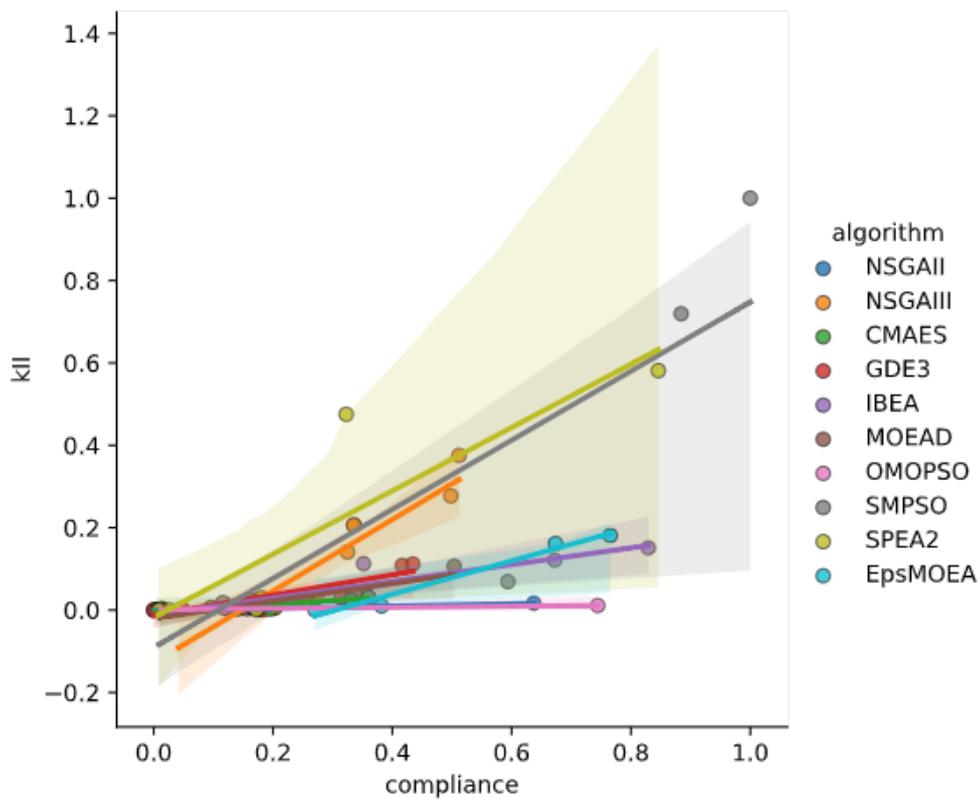
A Tabela 7.20 traz as métricas da otimização, que foram obtidas comparando-se as soluções da 2ª fase da otimização com as soluções de referência (1ª fase). Nessa tabela contém também o ranqueamento dos MMAs para esse exemplo (mecânica da fratura linear elástica - placa com trinca de borda - Modo-II). O ranqueamento foi feito aplicando-se uma *utility function* (Seção 2.1.9) nas métricas de performance individuais por algoritmo. Então, os três MMAs com melhor desempenho na otimização do ILMF aplicado nesse exemplo, foram: IBEA, GDE3 (foi o 1º do ranking no exemplo anterior) e NSGAI, nesta ordem. Isso não desabilita o uso dos demais MMAs na otimização desse tipo de modelo, uma vez que, conforme demonstrado, obtiveram excelentes e precisas soluções na frente de Pareto. Tanto que há uma alternância dos algoritmos que dominam cada métrica, por exemplo, o CMAES obteve o maior hypervolume (preferível), o IBEA obteve o menor Epsilon (preferível), e assim por diante. Mas, como definido nessa pesquisa, o que tem maior peso é o cálculo final da *utility function* que considera todas as métricas.

Tabela 7.20 – *Ranking* dos MMAs a partir das métricas de performance escaladas.

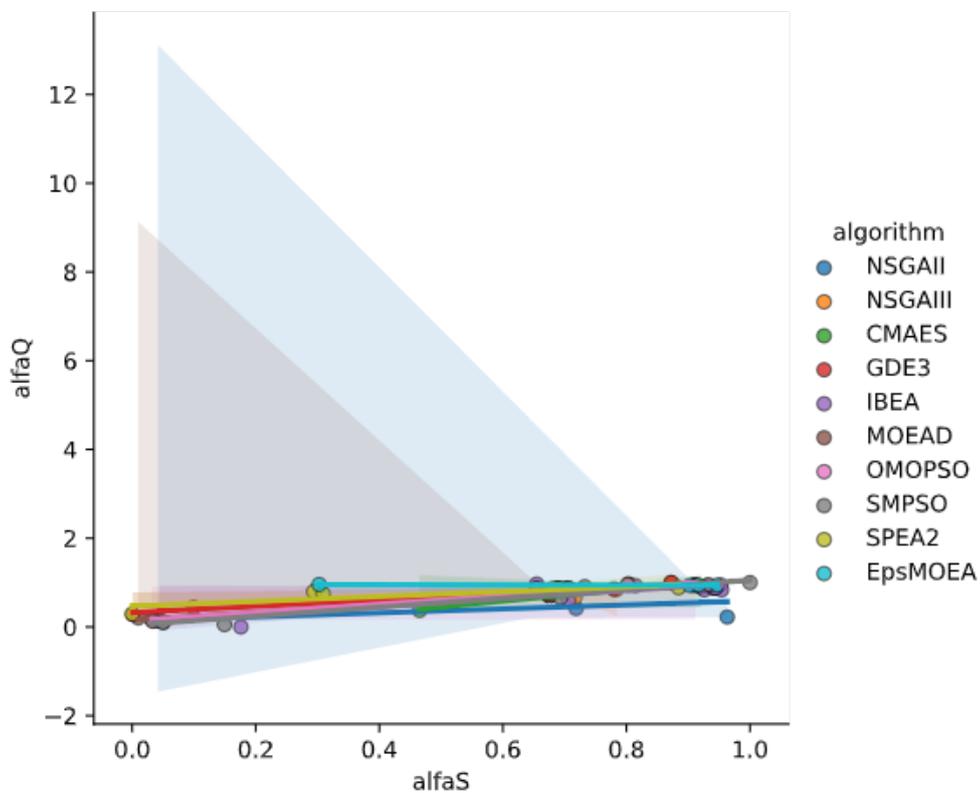
Ranking	Algorithm	Hypervolume	GD	IGD	Epsilon	Spacing	Utility
1	IBEA	0.3314407930	0.0401985520	0.1275799275	0.0000000000	0.0056640505	-0.1579982628
2	GDE3	0.4617663611	0.0071067603	0.7618752210	0.5417534130	0.0332094418	0.8821784750
3	NSGAI	0.4691015004	0.0049710189	0.7980697809	0.5408535777	0.0213760002	0.8961688773
4	CMAES	1.0000000000	1.0000000000	0.0000000000	0.0240195277	1.0000000000	1.0240195277
5	NSGAI	0.4303325206	0.0063447024	0.9241401276	0.5783662939	0.0515872641	1.1301058675
6	SPEA2	0.1549908171	0.0139629449	0.6074467965	0.8468071997	0.0529233855	1.3661495094
7	EpsMOEA	0.3711289108	0.1276991132	0.6963660034	0.6371262407	0.3046751499	1.3947375964
8	OMOPSO	0.2162287397	0.0152581714	0.9560253578	0.7886377556	0.0642557107	1.6079482558
9	MOEAD	0.0000000000	0.0000000000	1.0000000000	1.0000000000	0.0000000000	2.0000000000
10	SMPSO	0.0987005891	0.4604326546	0.9684702516	0.9029074671	0.7774110201	3.0105208043

A Figura 7.58 mostra essas métricas plotadas em coordenadas paralelas, para uma visualização mais inteligível.

A função objetivo que leva em conta o tempo de processamento (*CPU time*) de cada iteração, para cada algoritmo, foi desconsiderada nesse exemplo, pois, como os MMAs possuem heurística intrínseca, tendem enviar as soluções favorecendo as que gastam menos tempo computacional, mas, que não necessariamente fornecem bons resultados para as outras funções objetivo consideradas mais relevantes, que é o caso das relativas ao K_{II} e



(a) Espaço das soluções: K_{II} x *compliance* - 2ª fase.



(b) Espaço das variáveis: alfaQ x alfaS - 2ª fase.

Figura 7.57 – Exploração dos espaços das variáveis e das soluções pelos MMAs.

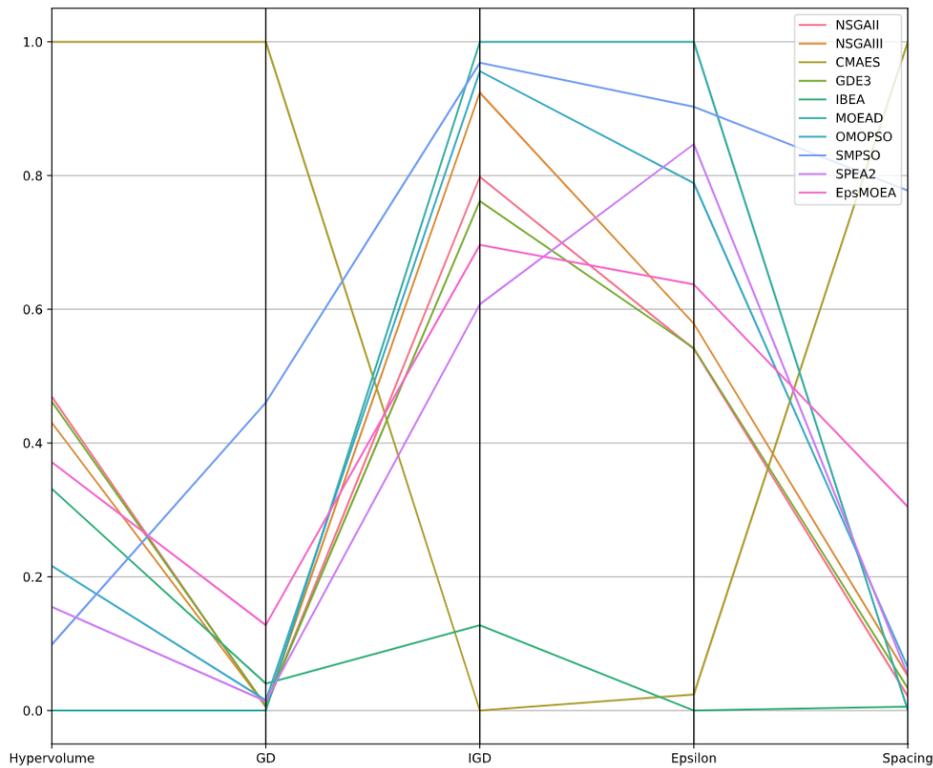


Figura 7.58 – Métricas de performance dos MMAs plotadas em coordenadas paralelas.

compliance . Portanto, somente duas funções objetivo foram utilizadas nesse exemplo: as relativas ao *compliance* e ao K_{II} . E não três, (*compliance* + K_{II} + CPU time) como tentado inicialmente.

7.3.3 - Placa com Trinca Inclinada de Borda – Modo Misto

Agora considere uma placa retangular com uma trinca de borda inclinada, esquematicamente representada na Figura 7.59 , em deformação de modo misto devido a uma tração remota $\bar{t} = \sigma$. O comprimento da trinca é dado por a , a largura da placa é dada por W e a altura é dada por $h = w/2$.

Para a análise desse problema, três casos foram considerados, com fatores de $a/w = 0.2, 0.4$ e 0.6 para $\alpha = 30^\circ$ e dois casos, com fatores de $a/w = 0.2$ e 0.4 , para $\alpha = 60^\circ$. A aproximação com o MQM considera base polinomial de primeira ordem e funções ponderadoras do tipo *quartic spline*, assim como domínios locais de colocação retangulares e parâmetros de discretização $\alpha_s = 2$ a 12 e $\alpha_q = 0.5$, obtidos por otimização numérica.

A análise numérica foi realizada considerando-se a distribuição nodal representada na

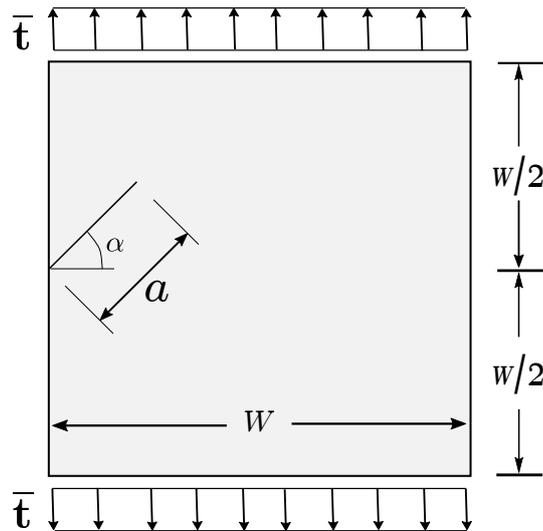


Figura 7.59 – Placa retangular com uma única trinca inclinada de borda sob tração remota.

Figura 7.60, onde para todos os casos de comprimento de trinca considerados, apenas a

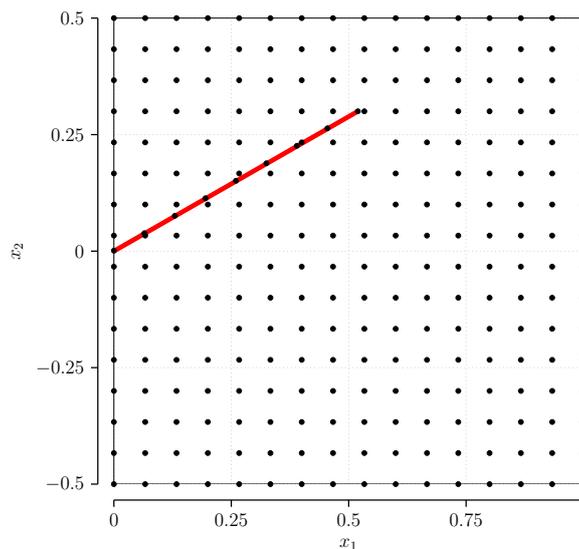


Figura 7.60 – Discretização de uma placa com uma distribuição nodal regular de $16 \times 16 = 256$ nós, com nós sobrepostos adicionais ao longo da trinca, para $a/w = 0.5$. A linha vermelha representa as faces da trinca.

distribuição nodal ao longo da trinca foi modificada, sempre sem considerar nenhum refinamento ao redor da ponta da trinca.

Os resultados obtidos são apresentados na Tabela 7.21 e Tabela 7.22 , respectivamente para os valores de $K_I/(\sigma\sqrt{\pi a})$ e $K_{II}/(\sigma\sqrt{\pi a})$, assim como a diferença relativa em porcentagem dos valores obtidos em Murakami (1986), para $\alpha = 30^\circ$. As tabelas 7.23 e 7.24 apresentam respectivamente os valores de $K_I/(\sigma\sqrt{\pi a})$, $K_{II}/(\sigma\sqrt{\pi a})$ e a diferença relativa em porcentagem dos valores obtidos em Murakami (1986), para $\alpha = 60^\circ$. Os resultados

Tabela 7.21 – Placa retangular com uma trinca inclinada de $\alpha = 30^\circ$. GSMF representa os valores obtidos na presente pesquisa e J-DBEM representa os valores obtidos com o DBEM utilizando a integral J (Aliabadi e Portela, 1999). A porcentagem dos erros foi calculada a partir do valor de referência obtido em Murakami (1986).

a/W	$K_I/(\sigma\sqrt{\pi a})$			% Diferença	
	GSMF	J-DBEM	Murakami	GSMF	J-DBEM
0.2	1.164	1.082	1.100	0.058	0.016
0.4	1.513	1.545	1.550	0.024	0.003
0.6	2.732	2.572	2.550	0.071	0.009

Tabela 7.22 – Placa retangular com uma trinca inclinada de $\alpha = 30^\circ$. GSMF representa os valores obtidos na presente pesquisa e J-DBEM representa os valores obtidos com o DBEM utilizando a integral J (Aliabadi e Portela, 1999). A porcentagem dos erros foi calculada a partir do valor de referência obtido em Murakami (1986).

a/W	$K_{II}/(\sigma\sqrt{\pi a})$			% Diferença	
	GSMF	J-DBEM	Murakami	GSMF	J-DBEM
0.2	0.325	0.351	0.350	0.071	0.003
0.4	0.471	0.474	0.470	0.002	0.009
0.6	0.580	0.700	0.700	0.171	0.000

Tabela 7.23 – Placa retangular com uma trinca inclinada de $\alpha = 60^\circ$. GSMF representa os valores obtidos na presente pesquisa e J-DBEM representa os valores obtidos com o DBEM utilizando a integral J (Aliabadi e Portela, 1999). A porcentagem dos erros foi calculada a partir do valor de referência obtido em Murakami (1986).

a/W	$K_I/(\sigma\sqrt{\pi a})$			% Diferença	
	GSMF	J-DBEM	Murakami	GSMF	J-DBEM
0.2	0.543	0.495	0.500	0.086	0.010
0.4	0.603	0.592	0.600	0.055	0.013

Tabela 7.24 – Placa retangular com uma trinca inclinada de $\alpha = 60^\circ$. GSMF representa os valores obtidos na presente pesquisa e J-DBEM representa os valores obtidos com o DBEM utilizando a integral J (Aliabadi e Portela, 1999). A porcentagem dos erros foi calculada a partir do valor de referência obtido em Murakami (1986).

a/W	$K_{II}/(\sigma\sqrt{\pi a})$			% Diferença	
	GSMF	J-DBEM	Murakami	GSMF	J-DBEM
0.2	0.327	0.356	0.360	0.092	0.011
0.4	0.439	0.413	0.420	0.045	0.017

obtidos estão em perfeita concordância com aqueles obtidos em Murakami (1986) e em Hiroshi et al. (2000), assim como os resultados obtidos com o J-DBEM. A alta precisão obtida pode ser facilmente evidenciada, considerando a configuração nodal simples e sem qualquer refinamento ao redor da ponta da trinca.

A configuração deformada da placa está esquematicamente representada na Figura 7.61.

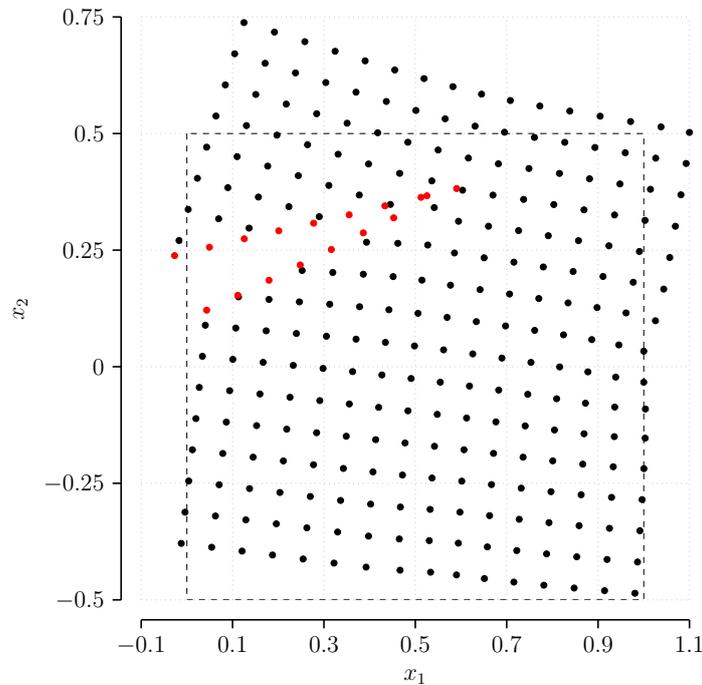


Figura 7.61 – Configuração deformada da placa, para $a/w = 0.5$, sob carregamento em modo misto. Os nós vermelhos representam os nós sobrepostos ao longo da trinca e na ponta da trinca.

As seções seguintes apresentam a nova abordagem de otimização proposta, aplicada no exemplo corrente.

7.3.3.1 - Equivalência entre o *compliance* e os fatores de intensidade de tensão

Os parâmetros do modelo GSMF utilizados na nova abordagem serão os mesmos empregados por Oliveira (2019), exceto que $\alpha_s = 1.5$ a 10.0 e α_q não será constante (variará de 0.20 a 0.95) e fará parte das variáveis a serem otimizadas pelos MMAs. Somente um caso de a/w será otimizado ($a/w = 0.6$) e $\alpha = 30^\circ$, uma vez que a quantidade de gráficos, tabelas, resultados, etc., apresentados com a nova abordagem de otimização é significativamente grande e deixaria o texto muito extenso. E, não haveria ganho significativo; seria mais uma redundância de resultados.

Nesse estudo de caso também serão seguidos os passos descritos na metodologia

(Capítulo 6); situações excepcionais ou destoantes serão explanadas.

Os valores das variáveis e das funções objetivo do modelo possuem escalas (ordens de grandeza) diferentes, portanto, para que os resultados sejam apresentados de forma clara e inteligível é necessário escalar os dados de forma unificada. O resultado da escalagem dos dados pode ser visto na Figura 7.62, para a 1ª fase e, na Figura 7.63, para os resultados da otimização (2ª fase). Como pode ser observado, é notória a diferença na visualização que há entre os dados originais e os escalados.

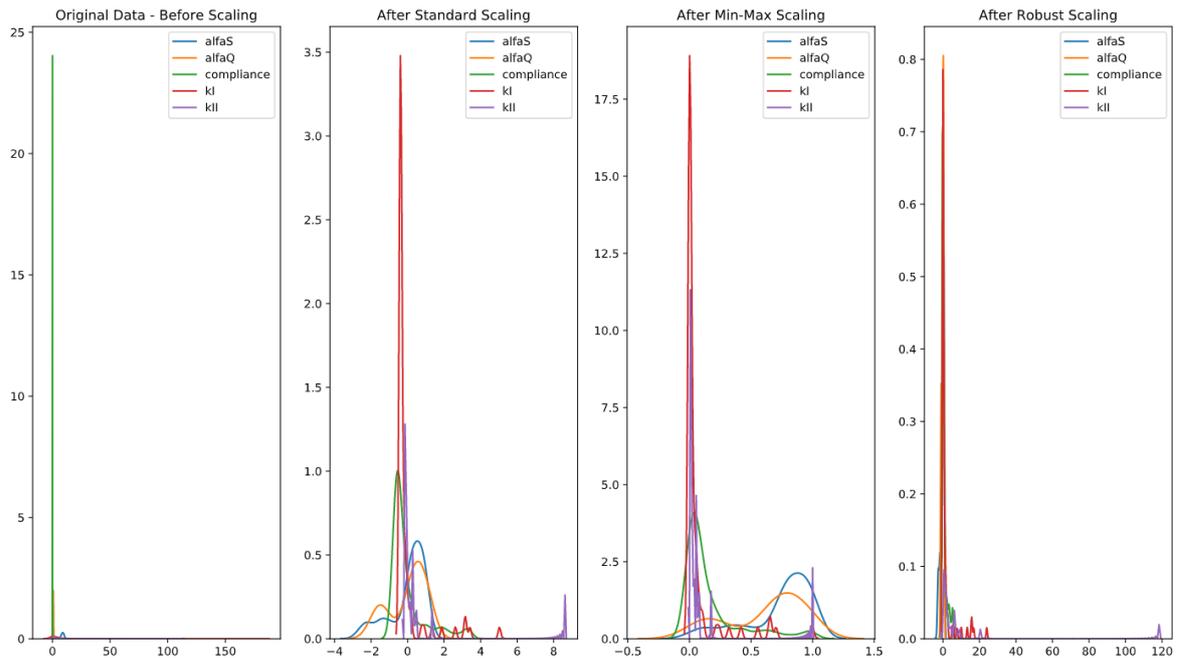


Figura 7.62 – Escalagem dos dados - 1ª fase.

Os resultados da otimização referentes à 1ª e 2ª fases estão descritos estatisticamente nas Tabelas 7.25 e 7.26, respectivamente. Esta descrição estatística compreende, na sequência dos cabeçalhos das linhas: quantidade de soluções, média, desvio padrão, valor mínimo, 1º quartil, 2º quartil (ou mediana), 3º quartil e valor máximo, respectivamente. Os valores também estão escalados para facilitar a apresentação dos resultados. Observa-se que foram encontradas 79 soluções únicas e não dominadas quando da busca pela frente de Pareto de referência (1ª fase) e 148 na otimização propriamente dita (2ª fase). A quantidade menor de soluções na frente de Pareto de referência se dá pelo fato de os algoritmos terem executado mais iterações nessa fase e terem apurado a convergência, diminuindo assim as soluções aptas a integrarem a frente de Pareto.

As Figuras 7.64a e 7.64b apresentam as frequências relativas para cada variável e cada função objetivo, das soluções pertinentes às frentes de Pareto da 1ª e 2ª fases da otimização, respectivamente. Pode ser observado que em ambas as frentes de Pareto há uma concentração

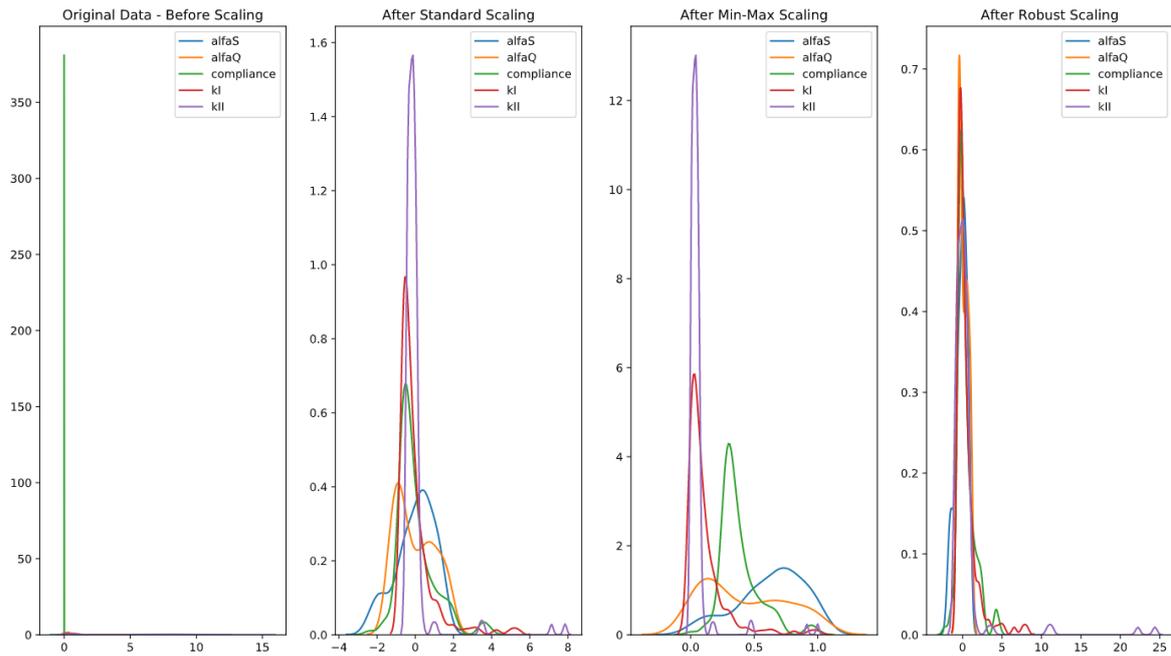


Figura 7.63 – Escalagem dos dados - 2ª fase.

Tabela 7.25 – Descrição estatística dos resultados escalados - 1ª fase.

Statistic	alfaS	alfaQ	compliance	k_I	k_{II}
count	79	79	79	79	79
mean	0.73105	0.61825	0.17067	0.07413	0.02394
std	0.27516	0.31231	0.24623	0.18515	0.11391
min	0.00000	0.00000	0.00000	0.00000	0.00000
25%	0.67139	0.31119	0.01604	0.00028	0.00128
50%	0.81271	0.72047	0.06858	0.00283	0.00335
75%	0.93966	0.85254	0.18829	0.04184	0.00971
max	1.00000	1.00000	1.00000	1.00000	1.00000

Tabela 7.26 – Descrição estatística dos resultados escalados - 2ª fase.

Statistic	alfaS	alfaQ	compliance	k_I	k_{II}
count	148	148	148	148	148
mean	0.63276	0.42706	0.38101	0.11246	0.05581
std	0.26141	0.32642	0.15840	0.16581	0.12070
min	0.00000	0.00000	0.00000	0.00000	0.00000
25%	0.46737	0.12966	0.28561	0.01754	0.01672
50%	0.68683	0.36713	0.33064	0.05806	0.03759
75%	0.82792	0.69790	0.43080	0.13311	0.05619
max	1.00000	1.00000	1.00000	1.00000	1.00000

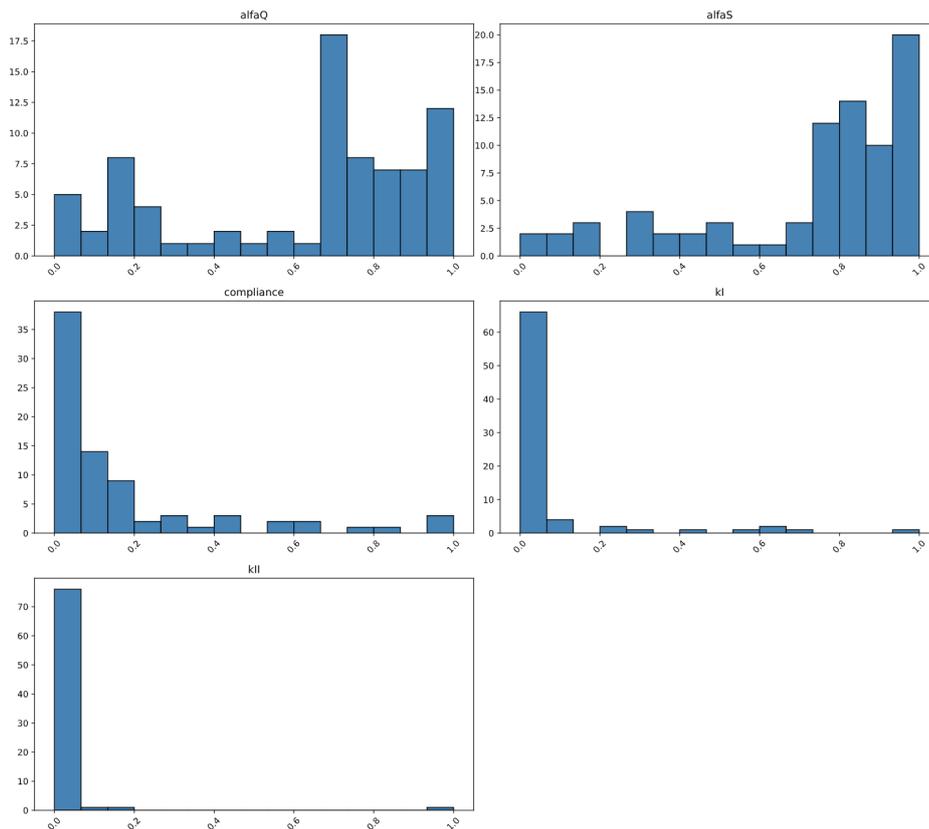
das soluções na região de valor mínimo para o *compliance*, K_I e K_{II} , mostrando a eficiência dos MMAs na otimização de modelos ILMF.

Com base na fundamentação apresentada na Seção 5.4, foram gerados mapas de calor dos parâmetros da otimização em função do coeficiente de correlação de Pearson, como mostram as Figuras 7.65a e 7.65b. Aqui fica demonstrado estatisticamente que a função objetivo baseada no fator de intensidade de tensão (SIF) pode ser substituída pela baseada na conformidade estrutural C , uma vez que o coeficiente de correlação entre C e K_I é 0.79 na 1ª fase de otimização e, 0.53 na 2ª fase. Esses coeficientes indicam que os vetores que representam as variáveis em um espaço n-dimensional são praticamente colineares e possuem uma correlação forte, na 1ª fase. Já a correlação entre C e K_{II} é 0.11 na 1ª fase e, 0.28 na 2ª. Ou seja, desprezível. Isso se dá pelo fato do modo de abertura da trinca (modo-I, relacionado a K_I) ser dominante nesse problema. Até porque, em problemas de propagação de trinca o incremento da trinca procura a direção que só tem Modo-I. Isso mostra que os MMAs foram capazes de "perceber" e lidar com essa peculiaridade do modelo.

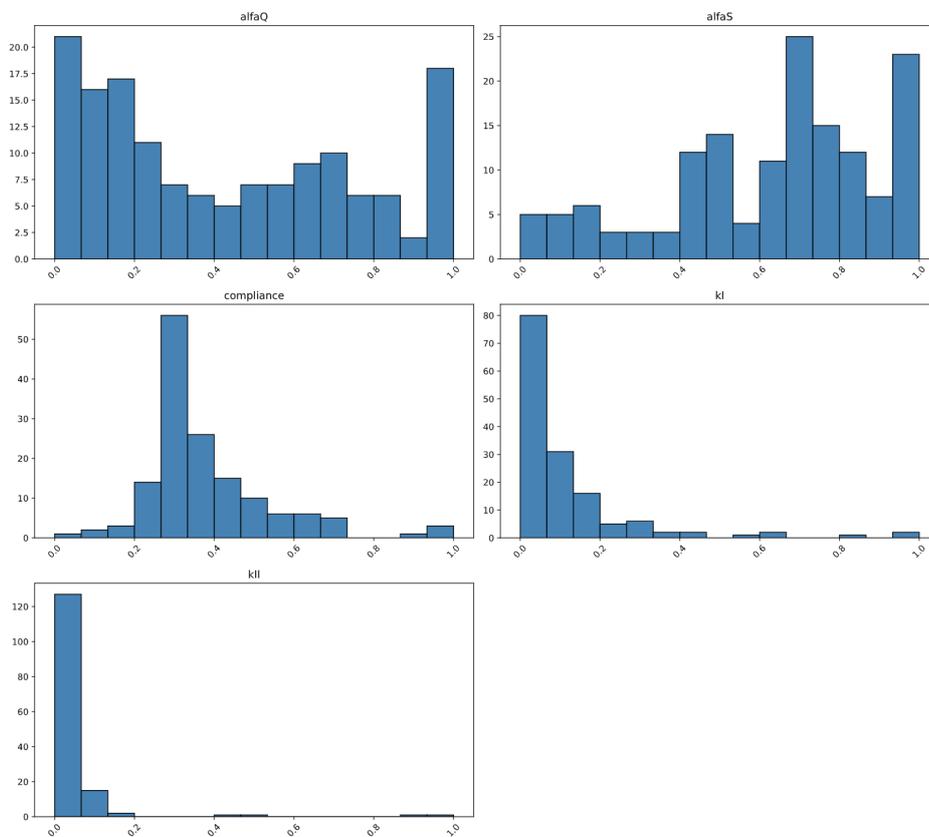
Nas Figuras 7.66 e 7.67, além de se observar se há correlação entre os parâmetros, através da linha de regressão linear que é plotada, também pode-se observar a forma da distribuição das soluções encontradas no processo de otimização. Confrontando duas variáveis quaisquer (uma da linha e outra da coluna) pode-se analisar imediatamente a relação que há entre elas. São gráficos emparelhados, do inglês: *pairwise plot*. Mais uma vez é notória a similaridade entre as funções objetivo baseadas no SIF e no *compliance*. Inclusive na forma em que seus valores são distribuídos nas soluções constituintes da frente de Pareto (vide os cruzamentos $K_I \times K_I$ e *compliance* x *compliance*).

Uma vista ampliada da correlação dos dois parâmetros mais relevantes da otimização (*compliance* e K_I) é dada pelas Figuras 7.68a e 7.68c. Note a concentração de soluções na vizinhança do mínimo do *compliance* e do K_I (que são os resultados pretendidos, já que é feita uma minimização destes parâmetros), em ambas as fases da otimização. Já o comportamento do K_{II} não é relevante para o modelo corrente.

A plotagem dos resultados em coordenadas paralelas permite analisar o impacto que os valores dos parâmetros de otimização (variáveis e funções objetivo) exercem uns sobre os outros. Para isso, a Figura 7.69 apresentam plotagens em coordenadas paralelas para a 2ª fase da otimização. Lembrando que na 1ª fase não foi feita a discriminação das soluções por algoritmo/MMA.

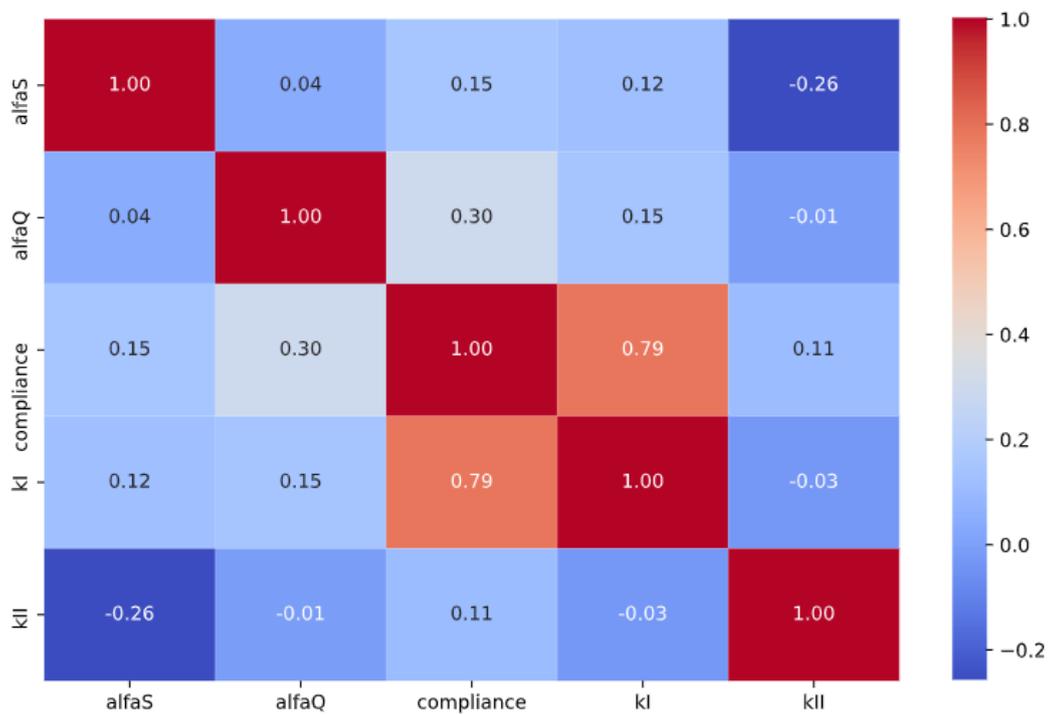


(a) Frente de Pareto - 1ª fase.

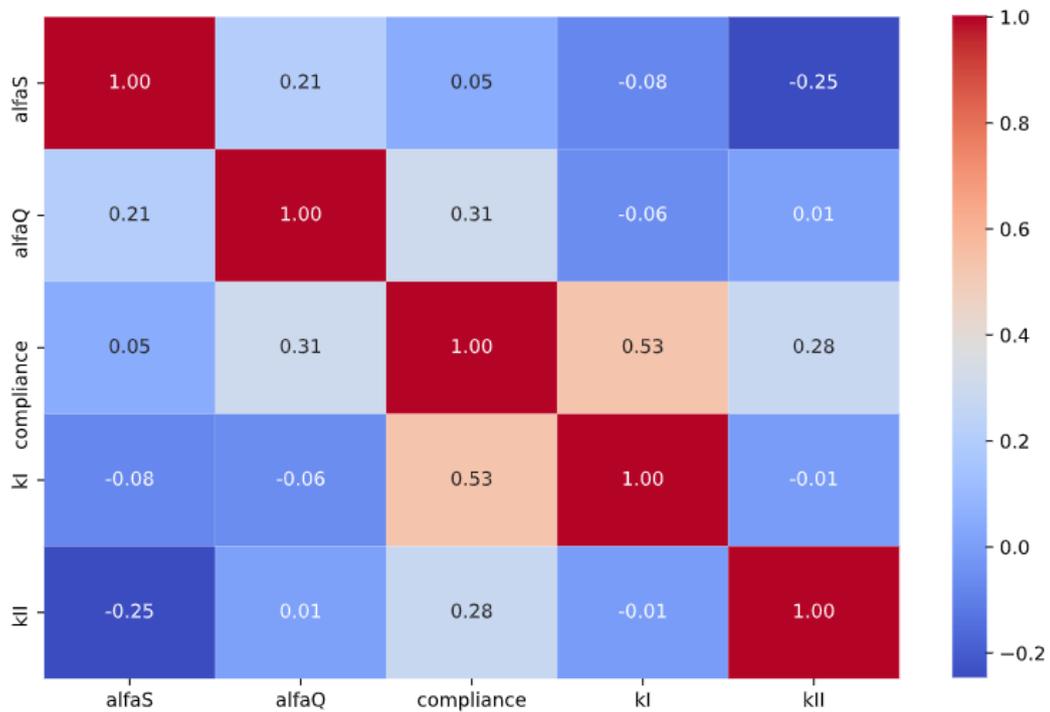


(b) Frente de Pareto - 2ª fase.

Figura 7.64 – Frequências relativas dos parâmetros da otimização.



(a) Resultados - 1ª fase.



(b) Resultados - 2ª fase.

Figura 7.65 – Mapa de calor dos parâmetros da otimização em função do coeficiente de correlação.

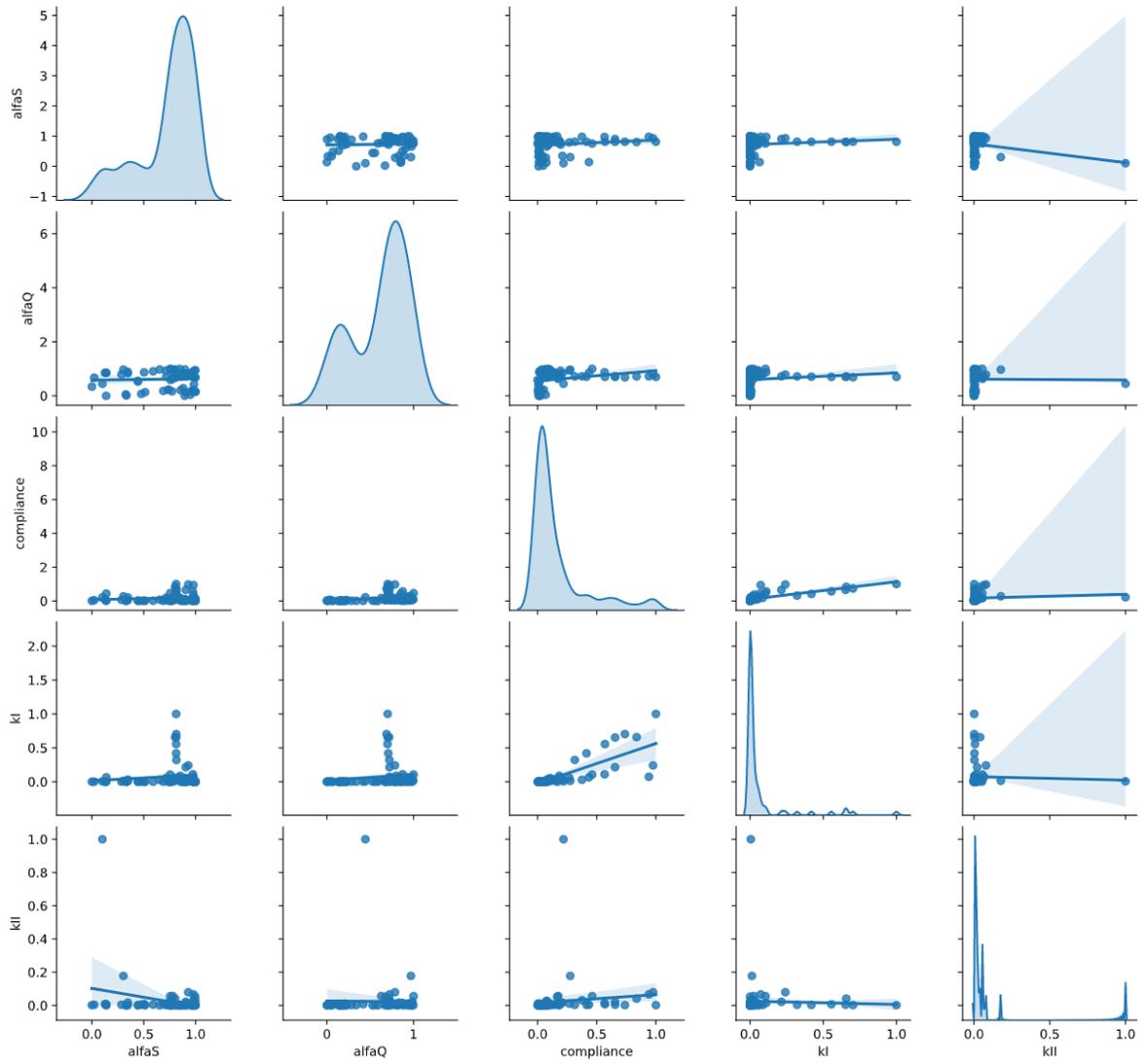


Figura 7.66 – *Pairwise plot - 1^a fase.*

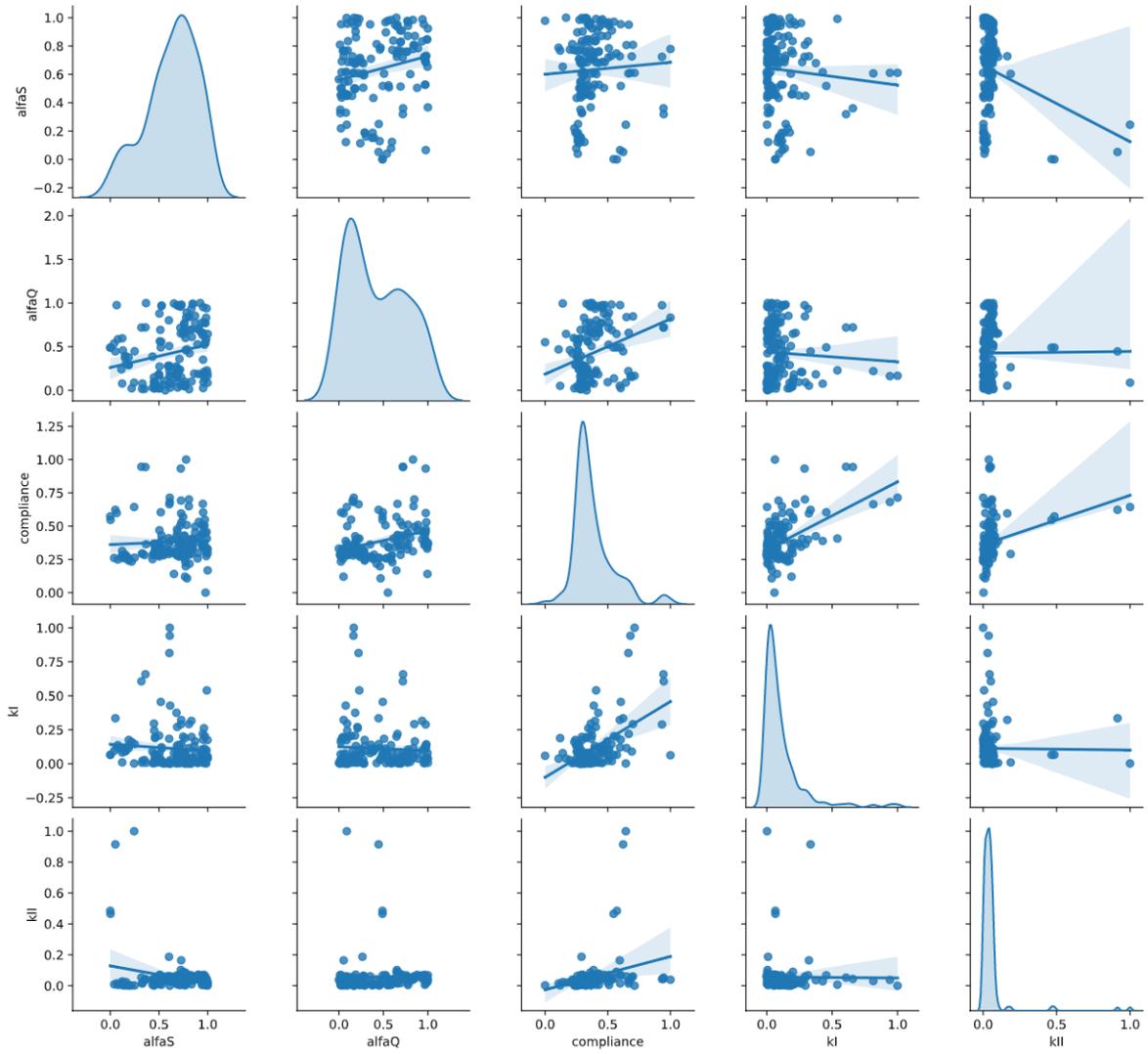
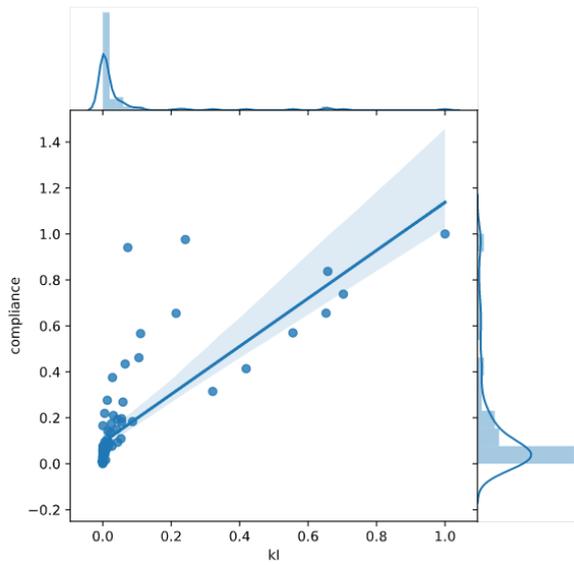
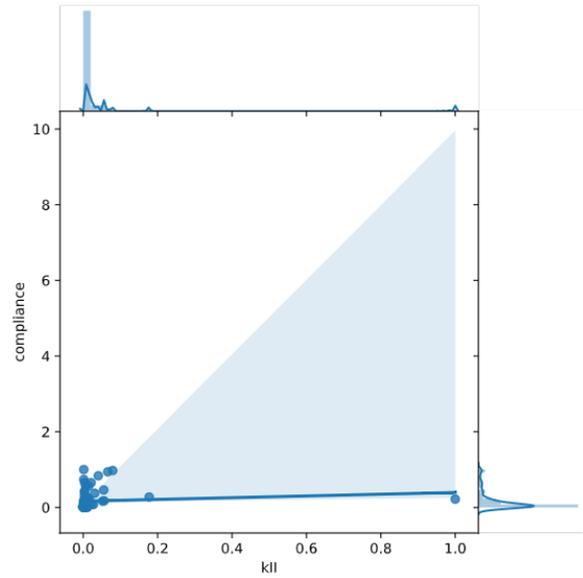


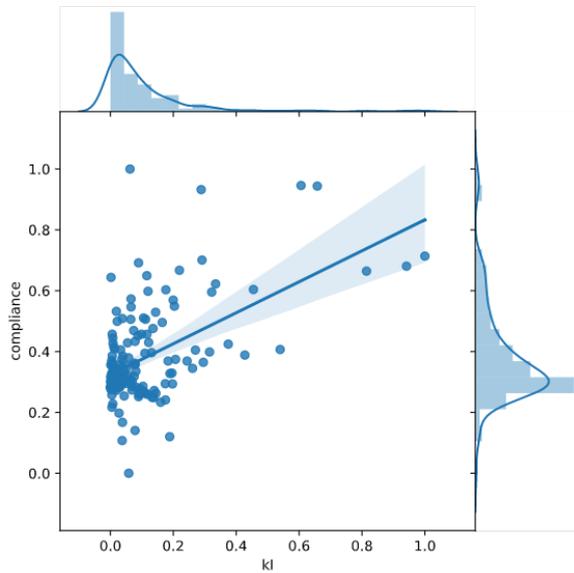
Figura 7.67 – *Pairwise plot - 2^a fase.*



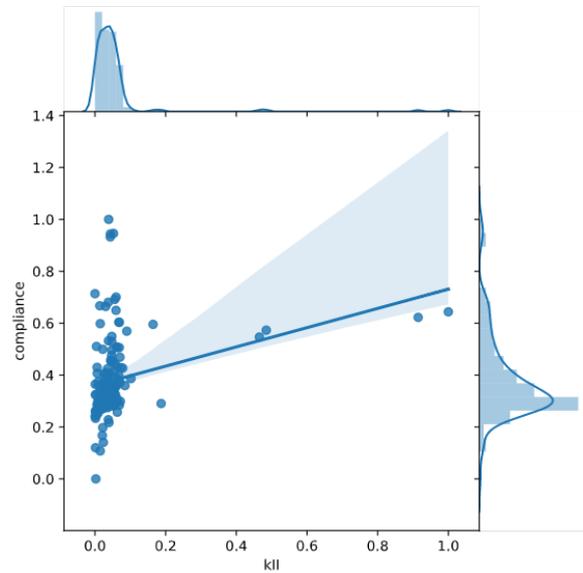
(a) Resultados - 1ª fase.



(b) Resultados - 1ª fase.



(c) Resultados - 2ª fase.



(d) Resultados - 2ª fase.

Figura 7.68 – Correlação entre K_I (e K_{II}) e *compliance*.

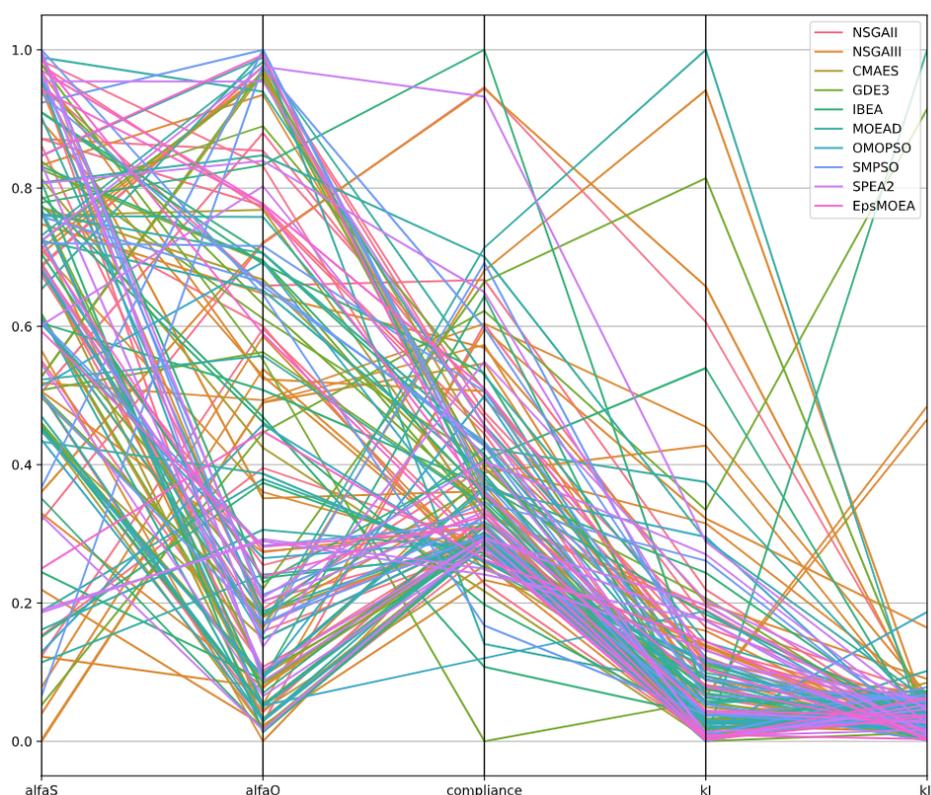


Figura 7.69 – Coordenadas paralelas dos parâmetros de otimização - 2ª fase.

7.3.3.2 - Ranqueamento dos MMAs através da *utility function* e as métricas de performance

A Figura 7.70 mostra a quantidade de soluções únicas, não dominadas e ótimas de Pareto, geradas por algoritmo utilizado.

A Figura 7.71 apresenta gráficos emparelhados (*pairwise plot*) dos parâmetros da otimização, 2ª fase, onde as soluções da frente de Pareto são discriminadas por MMA. Todos os MMAs apresentaram soluções satisfatórias quanto à precisão e exploraram bem o espaço de soluções, apresentando distribuições com formas muito semelhantes, como pode ser visto nos gráficos da diagonal descendente (do canto superior esquerdo ao canto inferior direito).

Uma observação mais detalhada é propiciada pela Figura 7.72. Pode-se observar que tanto o espaço das variáveis (alfaS e alfaQ), quanto o espaço das soluções (K_I , K_{II} e *compliance*) foram explorados pelos MMAs. A Figura 7.72c mostra que os algoritmos OMOPSO e SMPSO tiveram um comportamento diferente, apresentado correlação negativa entre alfQ e alfaS, enquanto todos os outros apresentaram correlação positiva.

A Tabela 7.27 traz as métricas da otimização, que foram obtidas comparando-se as soluções

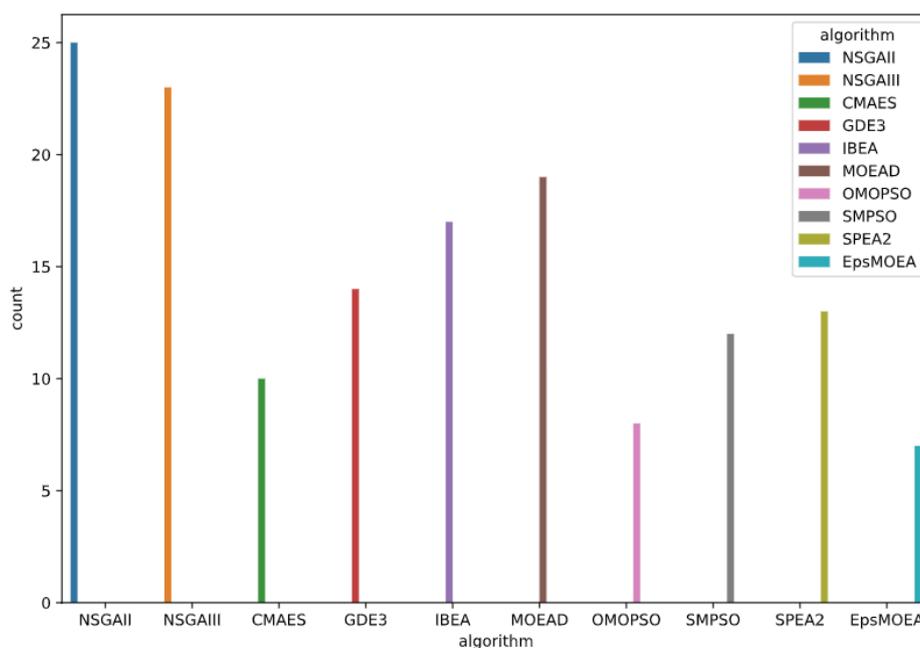


Figura 7.70 – Quantidade de soluções únicas, não dominadas e ótimas de Pareto, por MMA utilizado - 2ª fase.

da 2ª fase da otimização com as soluções de referência (1ª fase). Nessa tabela contém também o ranqueamento dos MMAs para esse exemplo (mecânica da fratura linear elástica - placa com trinca de borda - modo misto). O ranqueamento foi feito aplicando-se a *utility function* (Seção 2.1.9) nas métricas de performance individuais por algoritmo. Então, os três MMAs com melhor desempenho na otimização do ILMF aplicado nesse exemplo, foram: CMAES, EpsMOEA e SMPSO, nesta ordem. Isso não desabilita o uso dos demais MMAs na otimização desse tipo de modelo, uma vez que, conforme demonstrado, obtiveram excelentes e precisas soluções na frente de Pareto.

Tabela 7.27 – *Ranking* dos MMAs a partir das métricas de performance escaladas.

Ranking	Algorithm	Hypervolume	GD	IGD	Epsilon	Spacing	Utility
1	CMAES	1.0000000000	0.4937833494	0.4014804544	0.2383999889	0.5678015784	0.7014653711
2	EpsMOEA	0.8938287791	1.0000000000	0.0000000000	0.0000000000	1.0000000000	1.1061712209
3	SMPSO	0.1949240246	0.0506950094	0.6955499405	0.8720400676	0.2572696607	1.6806306536
4	NSGAIII	0.1063486884	0.0009987718	0.8441762324	0.9304383716	0.0307852514	1.7000499388
5	GDE3	0.1048521005	0.0024972795	0.8309093804	0.9327363576	0.0392394607	1.7005303776
6	OMOPSO	0.1802656167	0.0475780467	0.7164640819	0.8762910541	0.2805307637	1.7405983297
7	NSGAI	0.0317388620	0.0107109881	0.7621180195	0.9780642282	0.1066929051	1.8258472789
8	SPEA2	0.0374570500	0.0124972443	0.8389865389	0.9754846599	0.0893225140	1.8788339071
9	MOEAD	0.0427690711	0.0000000000	1.0000000000	0.9728251441	0.0000000000	1.9300560730
10	IBEA	0.0000000000	0.0338689200	0.9729608569	1.0000000000	0.1632099510	2.1700397278

A Figura 7.73 mostra essas métricas plotadas em coordenadas paralelas, para uma visualização mais inteligível.

A função objetivo que leva em conta o tempo de processamento (*CPU time*) de cada

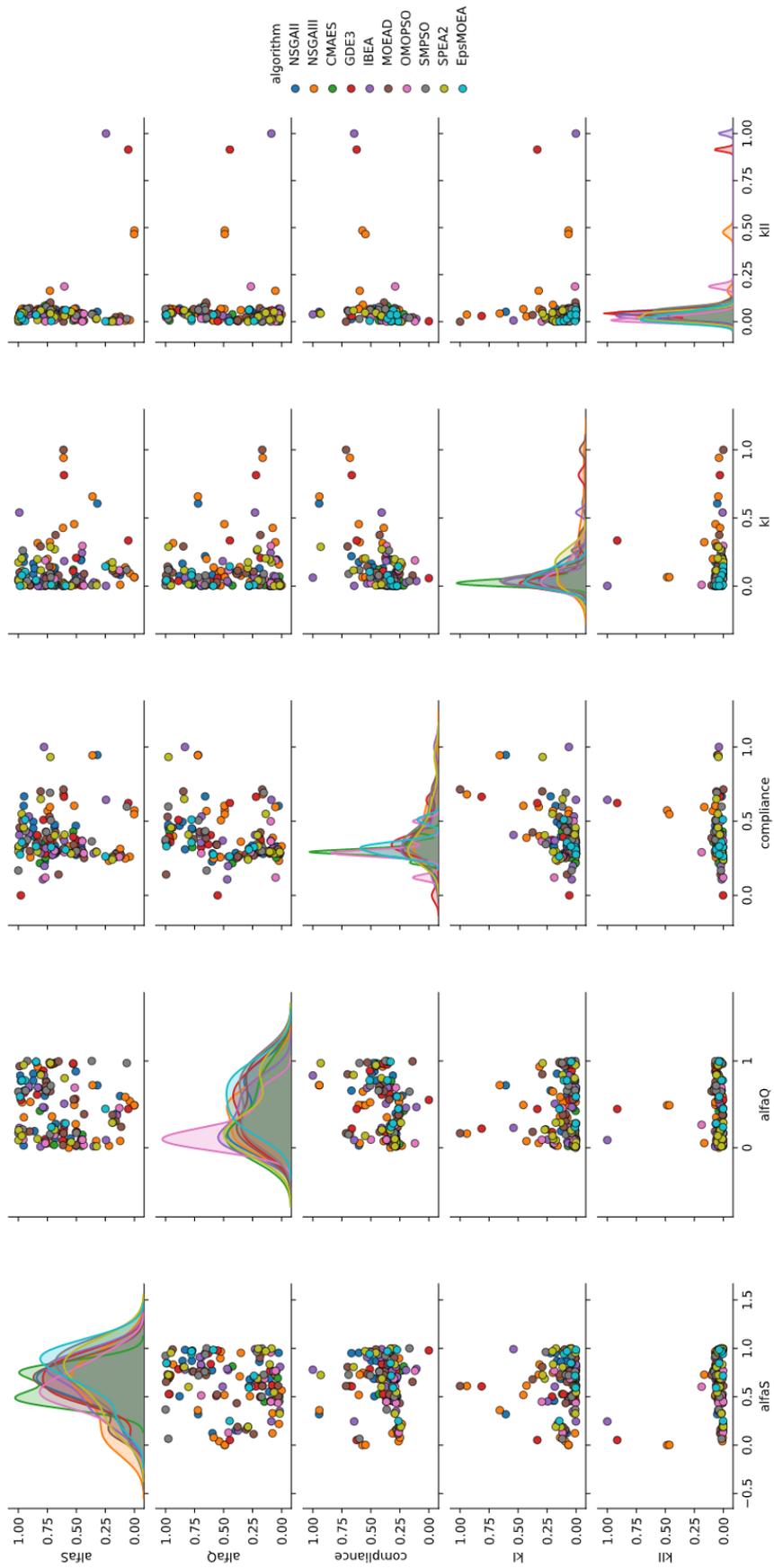
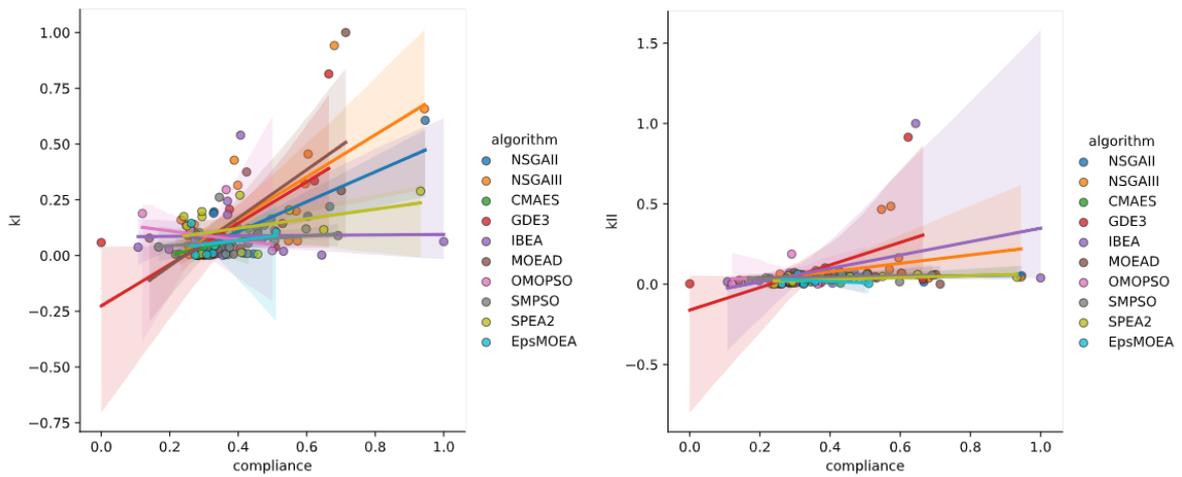
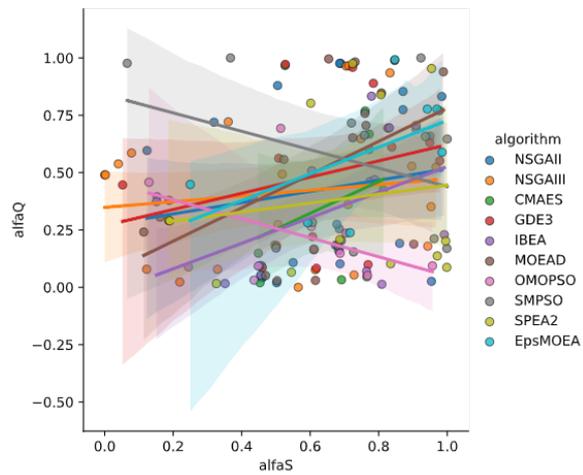


Figura 7.71 – *Pairwise plot* dos parâmetros da otimização, 2ª fase, com as soluções discriminadas por MMA.



(a) Espaço das soluções: K_I x compliance - 2ª fase. (b) Espaço das soluções: K_{II} x compliance - 2ª fase.



(c) Espaço das variáveis: α_Q x α_S - 2ª fase.

Figura 7.72 – Exploração dos espaços das variáveis e das soluções pelos MMAs.

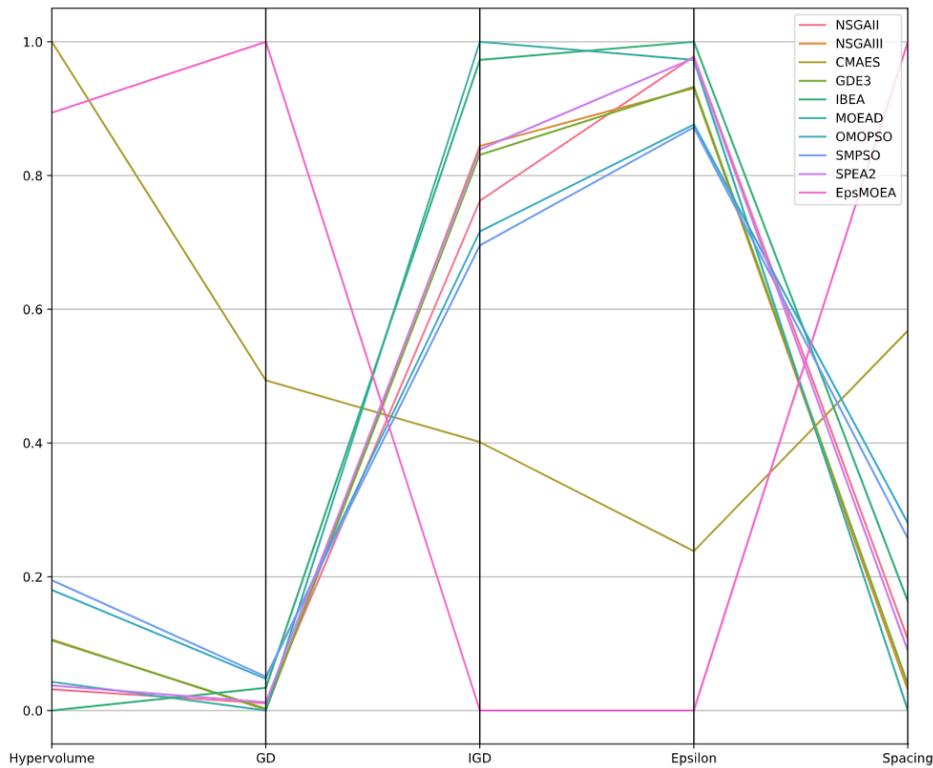


Figura 7.73 – Métricas de performance dos MMAs plotadas em coordenadas paralelas.

iteração, para cada algoritmo, foi desconsiderada nesse exemplo, pois, como os MMAs possuem heurística intrínseca, tendem a enviesar as soluções favorecendo as que gastam menos tempo computacional, mas, que não necessariamente fornecem bons resultados para as outras funções objetivo consideradas mais relevantes, que é o caso das relativas ao K_I e *compliance*. Portanto, somente três funções objetivo foram utilizadas nesse exemplo: as relativas ao *compliance*, K_I e K_{II} . E não quatro, (*compliance* + K_I + K_{II} + CPU time) como tentado inicialmente.

7.3.4 - Ranking de Performance dos MMAs - Fratura

A Tabela 7.28 apresenta o desempenho obtido por cada MMA quando otimizando o GSMF aplicado aos três problemas da mecânica da fratura linear elástica. Como pode ser observado, os MMAs com desempenho significativamente melhor (coluna: %do $\sum GlobalFratura$) que os demais, foram: GDE3, NSGAIII, EpsMOEA, IBEA e SPEA2; nesta ordem. Vale ressaltar que o IBEA e o NSGAIII também desempenharam muito bem no problema da elasticidade linear (viga em balanço e engastada), ocupando a 2ª e 3ª posições, respectivamente (Tabela 7.12).

Tabela 7.28 – *Ranking* de performance dos MMAs - fratura.

Ranking	MMA	Utility			Global Fratura	% do \sum Global Fratura
		Trinca Horiz. Modo I	Trinca Horiz. Modo II	Trinca Inclinada Modo Misto		
1	GDE3	-0.3686663760	0.8821784750	1.7005303776	2.2140424767	5.15%
2	NSGAI	0.6569126403	1.1301058675	1.7000499388	3.4870684466	8.11%
3	EpsMOEA	1.0000000000	1.3947375964	1.1061712209	3.5009088173	8.14%
4	IBEA	1.5951889672	-0.1579982628	2.1700397278	3.6072304322	8.39%
5	SPEA2	0.4174353061	1.3661495094	1.8788339071	3.6624187226	8.52%
6	CMAES	1.9875391101	1.0240195277	0.7014653711	3.7130240089	8.64%
7	NSGAI	1.9655586508	0.8961688773	1.8258472789	4.6875748070	10.91%
8	MOEAD	1.8218862207	2.0000000000	1.9300560730	5.7519422937	13.38%
9	OMOPSO	2.4949356225	1.6079482558	1.7405983297	5.8434822080	13.60%
10	SMPSO	1.8236443058	3.0105208043	1.6806306536	6.5147957637	15.16%

7.4 - RANKING GLOBAL DE PERFORMANCE DOS MMAS - ELASTICIDADE LINEAR + MECÂNICA DA FRATURA

As métricas de performance apresentadas até agora nesse trabalho, são escaladas, para facilitar seu entendimento e apresentação. Além disso, escalar os valores propicia o ranqueamento dos MMAs, pois, retira o viés de escala que existiria devido às diferentes ordens de grandezas dos parâmetros envolvidos no modelo. Com base nisso, um *ranking* global foi criado somando-se as performances (*utility*) de cada MMA, para cada caso estudado. Dessa forma, pode-se medir o desempenho global e a consistência de cada MMA na otimização dos modelos do ILMF. O *ranking* global obtido é mostrado na Tabela 7.29. Ela reflete o desempenho somado nas duas áreas de aplicação: elasticidade linear e mecânica da fratura linear elástica. Pelo fato de terem sido otimizados três modelos (contra um da primeira) desta última área, os resultados dela influencia o ranking global de performance. A análise de mais modelos da elasticidade linear é recomendável.

Tabela 7.29 – *Ranking* global de performance dos MMAs - elasticidade linear + mecânica da fratura linear elástica

Ranking	MMA	Utility			% do \sum Global
		Elasticidade Linear	Mec. Fratura Linear Elástica	Global	
1	GDE3	2.045897968	2.2140424767	4.2599404444	6.72%
2	NSGAI	1.997093294	3.4870684466	5.4841617410	8.65%
3	EpsMOEA	2.031996566	3.5009088173	5.5329053831	8.73%
4	IBEA	1.980213084	3.6072304322	5.5874435160	8.81%
5	SPEA2	2.057229159	3.6624187226	5.7196478814	9.02%
6	CMAES	2.461858492	3.7130240089	6.1748825010	9.74%
7	MOEAD	0.764795762	5.7519422937	6.5167380556	10.28%
8	NSGAI	2.030036427	4.6875748070	6.7176112340	10.59%
9	OMOPSO	2.332494545	5.8434822080	8.1759767529	12.89%
10	SMPSO	2.721553785	6.5147957637	9.2363495491	14.57%

Como se observa, os MMAs que apresentaram performance (coluna: %do \sum Global)

significativamente melhor comparada aos demais, são: GDE3, NSGAIII, EpsMOEA, IBEA e SPEA2. Relembrando, seguindo os critérios explanados na Seção 2.1.9, o MMA com menor *utility* é julgado de melhor desempenho, visto que é preferível que se tenha valor mínimo para as métricas GD, IGD, Epsilon e *Spacing*; e, valor máximo para o *hypervolume* (considerado com sinal negativo no cálculo da *utility*).

Conforme visto nos resultados dos estudos de caso dessa pesquisa, todos os MMAs apresentaram resultados satisfatórios, mas, se for necessário estabelecer uma preferência quando da otimização meta-heurística multiobjetivo dos modelos ILMF, deve-se dar prioridade aos cinco primeiros (GDE3, NSGAIII, EpsMOEA, IBEA e SPEA2) do ranking e preterir os três últimos (NSGAI, OMOPSO e SMPSO).

8 - CONCLUSÕES E RECOMENDAÇÕES

8.1 - CONCLUSÕES

Os algoritmos meta-heurísticos multiobjetivo empregados na otimização do método sem malha local, tanto quando aplicados na modelagem de problemas da elasticidade linear quanto na mecânica da fratura, apresentaram resultados precisos e performaram bem, como mostraram as métricas de performance. A nova abordagem propiciou a obtenção de frentes de Pareto ricas/densas de soluções, o que não ocorria na abordagem inicial, utilizando-se somente algoritmo genético puro. Entretanto, foi demonstrado que a quantidade de soluções existentes na frente de Pareto não está necessariamente vinculada à precisão e bom desempenho de um MMA. Se for necessário estabelecer uma preferência quando da otimização meta-heurística multiobjetivo dos modelos ILMF, deve-se dar prioridade aos cinco primeiros (GDE3, NSGAI, EpsMOEA, IBEA e SPEA2) do *ranking* global e, preferir os três últimos (NSGAI, OMOPSO e SMPSO).

O ranqueamento dos MMAs a partir da função *utility* e das métricas de performance permite selecionar os algoritmos mais adequados na otimização de cada tipo de modelo do ILMF. A partir deles, é possível criar um processo totalmente automatizado que trata internamente da convergência do *compliance* e deixa o ILMF muito robusto e adequado para a sua utilização em modeladores comerciais.

As métricas são imprescindíveis, pois, permite avaliar o quão próximo de um ótimo global as soluções da otimização provavelmente estão; uma vez que um algoritmo meta-heurístico pode ser considerado como um procedimento que provavelmente descobrirá uma solução viável muito boa, através de um processo iterativo com uma heurística subordinada de exploração (busca global) e exploração (busca local) do espaço de busca, mas, não necessariamente uma solução ótima, para um problema específico considerado. Na maioria dos casos nenhuma garantia é fornecida quanto à qualidade da solução obtida, mas, um método heurístico bem projetado geralmente pode fornecer uma solução que está nas proximidades da ótima.

O ILMF apresenta soluções muito precisas para as áreas estudadas, todavia, os parâmetros adimensionais α_s e α_q são variáveis e dependem do modelo sendo solucionado. Isso torna indispensável o uso de otimização eficiente para a sua determinação.

As formulações utilizadas são muito robustas e eficazes e trouxeram soluções para

limitações que existia no ILMF. Este, é baseado no método dos resíduos ponderados e resulta na forma fraca local, que nada mais é do que o teorema do trabalho advindo da teoria das estruturas. Em uma região local arbitrária, o teorema do trabalho estabelece uma relação de energia entre um campo de tensões estaticamente admissível e um campo de deformações cinematicamente admissível, resultando em uma formulação totalmente livre de integração numérica e com apenas termos de contorno. Para a modelagem de problemas da mecânica da fratura, a técnica da subtração da singularidade, permite o cálculo direto dos fatores de intensidade de tensão, como variáveis primárias do modelo numérico. Baseada no princípio da superposição, a técnica da subtração da singularidade regulariza o campo elástico através da subtração da singularidade de cada ponta de trinca, antes de sua solução com o método numérico. Como uma consequência dessa abordagem, a análise não necessita de distribuições nodais refinadas próximas da ponta da trinca, diferente de outros métodos numéricos conhecidos. Tudo isso, torna o ILMF muito menos exigente computacionalmente e adequado para soluções iterativas de modelos, como é necessário na otimização com os MMAs.

Uma das principais vantagens da abordagem empregada é a utilização do *compliance* na definição das funções objetivo, pois, torna dispensável o conhecimento das soluções analíticas para a aplicação do ILMF e, o deixa robusto e capaz de solucionar problemas diversos daqueles encontrados na literatura. Como demonstrado, a função objetivo advinda do *compliance* tem correlação perfeita positiva com o erro do campo dos deslocamentos (para problemas da elasticidade linear) e com o fatores de intensidade de tensão SIFs, para problemas da mecânica da fratura linear elástica.

A discretização do domínio com densidades nodais variadas, regulares ou randômicas, não impactam significativamente os resultados da abordagem de otimização proposta e do ILMF; pois, para todos os casos analisados, resultados precisos foram obtidos. Como esperado, só o tempo computacional gasto é maior para as discretizações nodais mais densas.

O uso de ferramentas de *data science* é fundamental para o entendimento adequado das inúmeras soluções geradas na otimização dos modelos, além de propiciar uma apresentação gráfica muito elucidativa das mesmas. Os *solvers* e a otimização geram as soluções, mas, os dados destas, se bem explorados, "contam toda a história" para o analista.

8.2 - SUGESTÕES PARA TRABALHOS FUTUROS

Conforme foi demonstrado com este estudo, a abordagem de otimização proposta é eficaz e eficiente na determinação dos parâmetros α_s e α_q do ILMF, quando aplicado em modelos da elasticidade linear e da mecânica da fratura linear elástica. Ainda há muito o que

ser explorado e aprimorado. Nesse sentido, como recomendações para trabalhos futuros, propõe-se:

- A implementação de uma *framework* do ILMF com otimização acoplada, para possibilitar a solução de modelos quaisquer da elasticidade linear e da mecânica da fratura; automatizando totalmente a abordagem proposta nesse estudo, para que a convergência no processo de otimização, através do *compliance*, seja feita internamente pela rotina, sem intervenção do analista. Isso deixará o método sem malha local a um passo de ser utilizado de forma generalizada pelos *softwares* de modelagem e análise estrutural; como acontece com o FEM atualmente. Mas, adicionando as inúmeras vantagens que o ILMF traz, conforme demonstrado ao longo desse texto. Consequentemente, a criação e disponibilização de um *package* Python *open source*;
- Conversão dos dois solvers do ILMF utilizados nesse estudo, para Python. Isso facilitará sua manutenção e propiciará um ganho enorme de velocidade de processamento, uma vez que paradigmas como o Cython, NumPy, Numba, etc., geram rotinas Python com velocidade que beiram à do C e C++ clássicos. Cabe ressaltar que os códigos do MATLAB são muito lentos, já que é uma linguagem interpretada;
- Adequação das formulações e metodologias empregadas nessa pesquisa para aplicação em modelos 3D. Isso permitirá ao *designer* não se preocupar mais com o tamanho do suporte compacto e do domínio de definição local. O que é um aspecto muito significativo que facilita a utilização dos métodos sem malha;
- Extensão da abordagem utilizada nesse estudo para a solução de problemas de propagação e crescimento de trincas, sendo guiadas pelo *compliance*;
- Extensão da abordagem utilizada nesse estudo para a geração automática de distribuições nodais, o que simplifica muito a modelagem; principalmente em regiões complexas com furos e trincas;
- Aplicação de otimização meta-heurística multiobjetivo diretamente em projetos de estruturais, por exemplo, para minimizar o peso total de estruturas metálicas. Uma possibilidade é utilizar softwares disponíveis e já consagrados no mercado, como: SAP2000, Robot, ABAQUS, ANSYS, etc., cumprindo o papel de *solvers* servindo o processo de otimização. Isso é facilmente conseguido integrando as rotinas por meio da OAPI (*Open Application Programming Interface*) que esses softwares disponibilizam.

REFERÊNCIAS BIBLIOGRÁFICAS

- Adeli, H. e K. C. Sarma (2006). *Cost Optimization of Structures: Fuzzy Logic, Genetic Algorithms, and Parallel Computing*. Wiley.
- Aliabadi, M.H. e A. Portela (1999). *Crack Growth Analysis Using Boundary Elements*. Sydney Grammar School Press.
- Atluri, S. N., Z. D. Han e A.M. Rajendran (2004). “A New Implementation of the Meshless Finite Volume Method Through the MLPG “Mixed” Approach”. Em: *CMES: Computer Modeling in Engineering and Sciences* 6.1, pp. 491–513.
- Atluri, S.N. e T. Zhu (1998). “A new Meshless Local Petrov-Galerkin (MLPG) approach in computational mechanics”. Em: *Computational Mechanics* 22.2, pp. 117–127.
- (2000). “New Concepts in Meshless Methods”. Em: *International Journal for Numerical Methods in Engineering* 47, 537—556.
- Bagheri, A. et al. (2011). “Optimization of meshless local Petrov-Galerkin using genetic algorithm for 3D elasto–static problems”. Em: *International Journal of Engineering* 24.2, pp. 143–153.
- Baradaran, G.H. e M.J. Mahmoodabadi (2009). “Optimal pareto parametric analysis of two dimensional steady–state heat conduction problems by MLPG method”. Em: *International Journal of Engineering* 22.4, pp. 387–406.
- Belytschko, T., L. Gu e Y. Y. Lu (1994a). “Fracture and crack growth by element-free Galerkin methods”. Em: *Modelling and Simulation in Materials Science and Engineering* 2.3A, pp. 519–534.
- Belytschko, T., Y. Y. Lu e L. Gu (1994b). “Element-free Galerkin methods”. Em: *International Journal for Numerical Methods in Engineering* 37.2, pp. 229–256. ISSN: 1097-0207.
- Belytschko, T. et al. (1996). “Smoothing and Accelerated Computations in the Element Free Galerkin Method”. Em: *Journal of Computational and Applied Mathematics* 74.1–2, pp. 111–126.
- Benítez-Hidalgo, Antonio et al. (2019). “jMetalPy: A Python framework for multi-objective optimization with metaheuristics”. Em: *Swarm and Evolutionary Computation* 51, p. 100598.

- Beume, N. e G. Rudolph (2006). “Faster S-metric calculation by considering dominated hypervolume as Klee’s measure problem”. Em: *Second International Association of Science and Technology for Development (IASTED) Conference on Computational Intelligence*.
- Boslaugh, S. (2012). *Statistics in a Nutshell*. O’Reilly Media, Incorporated.
- Brahtz, J. H. A. (1933). “Stress distribution in a reentrant corner”. Em: *Transactions of the American Society of Mechanical Engineers* 55.1, pp. 31–37.
- Brebbia, Carlos Alberto e Stephen Walker (2013). *Boundary element techniques in engineering*. Elsevier.
- Caicedo, J. e A. Portela (2015). “Cracked plate analysis with the Dual Boundary Element Method and Williams eigenexpansion”. Em: *Engineering analysis with boundary elements* 52.1, pp. 16–23.
- Chen, Youping, James D. Lee e Azim Eskandarian (2006). *Meshless Methods in Solid Mechanics*. Vol. 1. Springer Science Business Media, p. 200.
- Civelek, M. B. e F. Erdogan (1982a). “Crack problems for a rectangular plate and an infinite strip”. Em: *International Journal of Fracture* 19.2, pp. 139–159.
- Civelek, M.B. e F. Erdogan (1982b). “Crack problems for a rectangular plate and an infinite strip”. Em: *International Journal of Fracture* 19, pp. 139–159.
- “Coeficiente de correlação de Pearson” (2021). Em: *Wikipédia: a enciclopédia livre*. Wikimedia. URL: [\url{https : / / pt . wikipedia . org / wiki / Coeficiente_de_correla%C3%A7%C3%A3o_de_Pearson}](https://pt.wikipedia.org/wiki/Coeficiente_de_correla%C3%A7%C3%A3o_de_Pearson).
- Coelho, C.A.C., G.B. Lamont e D.A. Van Veldhuizen (2007). *Evolutionary Algorithms for Solving Multi-Objective Problems*. Springer New York.
- Coello Coello, C. A., G. B. Lamont e D. A. Van Veldhuizen (2007). *Evolutionary Algorithms for Solving Multi-Objective Problems*. Springer Science+Business Media LLC.
- Deb, K. e H. Jain (2014). “An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part i: Solving problems with box constraints”. Em: *IEEE Transactions on Evolutionary Computation* 18.4, pp. 577–601.
- Deb, K. e S. Jain (2002). “Running performance metrics for evolutionary multi-objective optimization”. Em: *Kanpur Genetic Algorithms Laboratory (KanGAL), Indian Institute of Technology*.

- Deb, K., M. Mohan e S. Mishra (2003). *A fast multi-objective evolutionary algorithm for finding well-spread Pareto-optimal solutions*. Rel. técn. 2003002. Indian Institute of Technology, Kanpur, India: Kanpur Genetic Algorithms Laboratory (KanGAL).
- Deb, Kalyanmoy et al. (2000). “A Fast and Elitist Multi-Objective Genetic Algorithm: NSGA-II”. Em: *IEEE Transactions on Evolutionary Computation* 6.2, pp. 182–197.
- Dorigo, M. et al. (2004). *Ant Colony Optimization*. BRADFORD BOOK.
- Ebrahimnejad, M., N. Fallah e A.R. Khoei (2015). “Adaptive refinement in the meshless finite volume method for elasticity problems”. Em: *Computers & Mathematics with Applications* 69.12, pp. 1420–1443.
- Eftis, J., D.L. Jones e H. Liebowitz (1990). “Load Biaxiality and Fracture: Synthesis and Summary”. Em: *Engineering Fracture Mechanics* 36.4, pp. 537–574.
- Fenner, D.G. (1973). “Elastic Stress Singularity Analysis Using the Eigenfunction-Expansion Method”. Ph.D. University of London, Imperial College.
- Fichera, G (2006). *Linear Elliptic Differential Systems and Eigenvalue Problems*. Springer.
- Fonseca, C. M. e P. J. Fleming (1996). “On the performance assessment and comparison of stochastic multiobjective optimizers”. Em: *Parallel Problem Solving from Nature (PPSN IV)*, pp. 584–593.
- Gandomi, A.H. et al. (2013). *Metaheuristic Applications in Structures and Infrastructures*. Elsevier Science.
- Gelfand, I.M. e G.E. Shilov (1964). *Generalized Functions, Vol. I: Properties and Operations*.
- Glover, F.W. e G.A. Kochenberger (2003). *Handbook of Metaheuristics*. Springer US.
- Glover, F.W. e M. Laguna (1998). *Tabu Search*. Springer US.
- H., Tada (2000). *The stress analysis of cracks handbook*. ASME Press.
- Hadka, D. (2016). *Beginner’s Guide to the Moea Framework*. CreateSpace Independent Publishing Platform.
- Hadka, D. e P. Reed (2012). “Diagnostic assessment of search controls and failure modes in many-objective evolutionary optimization”. Em: *Evolutionary Computation* 20.3, pp. 423–452.
- (2013). “Borg: An auto-adaptive many-objective evolutionary computing framework”. Em: *Evolutionary Computation* 21.2, pp. 231–259.

- Hansen e Kern (2004). “Evaluating the cma evolution strategy on multimodal test functions”. Em: *Eighth International Conference on Parallel Problem Solving from Nature PPSN VII*, pp. 282–291.
- Hiroshi, T., C.P. Paul e R.I. George (2000). *Stress Intensity Factors Handbook*. ASME.
- Holland, J.H. (1975). *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence*. University of Michigan Press.
- Igel, C., N. Hansen e S. Roth (2007). “Covariance matrix adaptation for multi-objective optimization”. Em: *Evolutionary Computation* 15, pp. 1–28.
- Irwin, G.R. (1957). “Analysis of Stresses and Strains Near the End of a Crack Traversing a Plate”. Em: *Journal of Applied Mechanics* 24, pp. 361–364.
- Kaveh, A. (2017). *Applications of Metaheuristic Optimization Algorithms in Civil Engineering*. Springer International Publishing.
- Knowles, J. e D Corne (2002). “On metrics for comparing non-dominated sets”. Em: *Congress on Evolutionary Computation (CEC 2002)*.
- Krysl, P. e T. Belytschko (1997). “Element-Free Galerkin Method: Convergence of the Continuous and Discontinuous Shape Functions”. Em: *Computer Methods in Applied Mechanics and Engineering* 148.3–4, pp. 257–277.
- Kukkonen, S. e J. Lampinen (2005). “GDE3: The third evolution step of generalized differential evolution”. Em: *The 2005 IEEE Congress on Evolutionary Computation (CEC 2005)*, pp. 443–450.
- Lagaros, Nikos D. (2018). “The environmental and economic impact of structural optimization”. Em.
- Lancaster, Peter e Kes Salkauskas (1981). “Surfaces generated by moving least squares methods”. Em: *Mathematics of computation* 37.155, pp. 141–158.
- Li, H. e Q. Zhang (2009). “Multiobjective optimization problems with complicated Pareto sets, MOEA/D and NSGA-II”. Em: *IEEE Transactions on Evolutionary Computation* 13.2, pp. 284–302.
- Liu, G. R. e Y. T. Gu (2005). *An introduction to meshfree methods and their programming*. Springer Science & Business Media.
- Liu, G.R. e Y.T. Gu (1999). “A point interpolation method”. Em: *Asia-Pacific Conference on Computational Mechanics*, pp. 1009–1014.

- Liu, G.R. e Y.T. Gu (2001a). “A Local Point Interpolation Method for Stress Analysis of Two-Dimensional Solids”. Em: *Structural Engineering and Mechanics* 11.2, 221—236.
- (2001b). “A local radial point interpolation method (LRPIM) for free vibration analyses of 2-D solids”. Em: *Journal of Sound and vibration* 246.1, pp. 29–46.
- Liu, G.R. et al. (2002a). “Point Interpolation Method Based on Local Residual Formulation Using Radial Basis Functions”. Em: *Structural Engineering and Mechanics* 14, 713—732.
- Liu, X. et al. (2002b). “Radial Basis Point Interpolation Collocation Method For 2D Solid Problem”. Em: *Proceedings of the 1st Asian Workshop on Meshfree Methods*, pp. 35–40.
- Mase, George Thomas (1999). *Continuum mechanics for engineers*. 2^a ed. CRC Press, p. 381. ISBN: 0849318556.
- MathWorks (2014). *Global Optimization Toolbox: User’s Guide*. The MathWorks, Inc. - USA.
- Moussaoui, A. e T. Bouziane (2013). “Comparative Study of the Effect of the Parameters of Sizing Data on Results by the Meshless Methods (MLPG)”. Em: *World Journal of Mechanics* 3.1, pp. 82–87.
- Mukaka, MM. (2012). “A guide to appropriate use of Correlation coefficient in medical research”. Em: *Malawi Medical Journal* 24, pp. 69 –71.
- Murakami, Y. (1986). *Stress Intensity Factors Handbook*. Pergamon Press.
- (1987). *Linear Elliptic Differential Systems and Eigenvalue Problems*. Pergamon Press.
- Muskhelishvili, N.I. (2010). *Some Basic Problems of the Mathematical Theory of Elasticity*. Springer Netherlands.
- Nayroles, B., G. Touzot e P. Villon (1992). “Generalizing the Finite Element Method: Diffuse Approximation and Diffuse Elements”. Em: *Computational Mechanics* 10, 307—318.
- Nebro, A. J. et al. (2009). “SMPSO: A new PSO-based metaheuristic for multi-objective optimization”. Em: *IEEE Symposium on Computational Intelligence in Multicriteria Decision-Making (MCDM 2009)*, pp. 66–73.
- Oliveira, T. e A. Portela (2016). “Weak-Form Collocation – a Local Meshless Method in Linear Elasticity”. Em: *Engineering Analysis with Boundary Elements* 6, pp. 491–513.
- (2019). “A local mesh free method with the singularity subtraction technique”. Em: *Engineering Analysis with Boundary Elements* 104, pp. 148–159.

- Oliveira, T. S. (2019). “Método sem malha local – Subtração da singularidade em mecânica da fratura linear elástica”. Tese de dout. ENC/FT/UnB.
- Organ, D.J., M.A. Fleming e T. Belytschko (1996). “Continuous Meshless Approximations for Nonconvex Bodies by Diffraction and Transparency”. Em: *Computational Mechanics* 18.3–4, pp. 225–235.
- Portela, A. (1993). *Dual boundary element analysis of crack growth*. reprint. Vol. 14. Computational Mechanics Publications, p. 176. ISBN: 9781853121876.
- Portela, A. e M.H. Aliabadi (1993). *Crack Growth Analysis Using Boundary Elements – Software*. Computational Mechanics Publications, Southampton, UK e Boston, USA.
- Portela, A., M.H. Aliabadi e D.P. Rooke (1991). “Efficient Boundary Element Analysis of Sharp Notched Plates”. Em: *International Journal for Numerical Methods in Engineering* 32.3, pp. 445–470.
- Rao, S.S. (2019). *Engineering Optimization: Theory and Practice*. Wiley.
- Reddy, N (2006). *An introduction to the Finite Element Method*. 3ª ed. Vol. 3. McGraw-Hill, p. 761. ISBN: 0070513554.
- Rösel, R. (1987). “On the Wedge/Notch Eigenvalues”. Em: *International Journal of Fracture* 33.1, pp. 61–67.
- Saad, A. El H., Z. Dong e M. Karimi (2017). “A Comparative Study on Recently-Introduced Nature-Based Global Optimization Methods in Complex Mechanical System Design”. Em: *Algorithms* 10.4.
- Santana, E. et al. (2020). “A local mesh free numerical method with automatic parameter optimization”. Em: *Engineering Analysis with Boundary Elements* 113, pp. 55–71.
- Sierra, M. R. e C. A. Coello Coello (2005). “Improving PSO-based multi-objective optimization using crowding, mutation and ϵ -dominance”. Em: *Evolutionary Multi-Criterion Optimization (EMO 2005)*, pp. 505–519.
- Sih, G.C. e H. Liebowitz (1968). *Mathematical Theories of Brittle Fracture*. Fracture – An Advanced Treatise. Academic Press, p. 124.
- Sokolnikoff, I. S. (1956). *Mathematical theory of elasticity*. Vol. 83. McGraw-Hill New York.
- Sternberg, E. e W.T. Koiter (1958). “The Wedge Under a Concentrated Couple: A Paradox in Two Dimensional Elasticity”. Em: *Journal of Applied Mechanics* 25, pp. 575–581.
- Talbi, E.G. (2009). *Metaheuristics: From Design to Implementation*. Wiley.

- Vasilopoulos, D. (1988). “On the Determination of Higher Order Terms of Singular Elastic Stress Fields Near Corners”. Em: *Numerische Mathematik* 53.1, pp. 51–95.
- While, L., L. Bradstreet e L. Barone (2012). “A fast way of calculating exact hypervolumes”. Em: *IEEE Transactions on Evolutionary Computation*.
- Williams, M. L. (1952a). “Stress singularities resulting from various boundary conditions in angular corners of plates in extension”. Em: *Journal of Applied Mechanics* 19.4, pp. 526–528.
- Williams, M.L. (1952b). “Stress Singularities Resulting from Various Boundary Conditions in Angular Corners of Plates in Extension”. Em: *Journal of Applied Mechanics*, pp. 526–528.
- Y., Murakami (1987). *Stress intensity factors handbook, 1st edition, vol 2*. Pergamon Press.
- Yang, X. (2011). “Metaheuristic Optimization”. Em: *Scholarpedia* 6.8. revision #91488, p. 11472. DOI: 10.4249/scholarpedia.11472.
- Yang, X.S. (2010). *Engineering Optimization: An Introduction with Metaheuristic Applications*. Wiley.
- Yang, X.S., G. Bekdaş e S.M. Nigdeli (2016). *Metaheuristics and Optimization in Civil Engineering*. Springer International Publishing.
- Zienkiewicz, O. C., R. L. Taylor e Robert Leroy Taylor (2000). *The Finite Element Method: The basis*. Butterworth-Heinemann, p. 689.
- Zitzler, E. e S. Künzli (2004). “Indicator-based selection in multiobjective search”. Em: *Parallel Problem Solving from Nature (PPSN VIII)*, pp. 832–842.
- Zitzler, E., M. Laumanns e L. Thiele (2002a). *SPEA2: Improving the Strength Pareto Evolutionary Algorithm For Multiobjective Optimization*. Rel. técn. IBarcelona, Spain: International Center for Numerical Methods in Engineering (CIMNE).
- Zitzler, E. et al. (2002b). “Performance assessment of multiobjective optimizers: An analysis and review”. Em: *IEEE Transactions on Evolutionary Computation* 7.2, pp. 117–132.
- Zitzler, E. et al. (2002c). “Why quality assessment of multiobjective optimizers is difficult”. Em: *Genetic and Evolutionary Computation Conference (GECCO 2002)*, pp. 666–674.

APÊNDICES

A - PARÂMETROS DE DISCRETIZAÇÃO α_s E α_q

Em uma discretização sem malha, o tamanho do suporte compacto onde as funções de forma nodais são definidas e, o tamanho do domínio de integração onde a matriz de rigidez nodal do domínio do modelo é computada, são parâmetros de discretização arbitrários que devem ser convenientemente definidos em uma aplicação, uma vez que seus valores afetam a performance da solução numérica.

No que diz respeito à precisão da solução a partir de uma discretização sem malha, os valores para a dimensão do suporte compacto Ω_s e a dimensão do domínio local de integração Ω_q , denotados respectivamente por r_{Ω_s} e r_{Ω_q} , devem ser otimizados para cada modelo/aplicação. Para um nó i , estes parâmetros são definidos como

$$r_{\Omega_s} = \alpha_s c_i \quad (\text{A.1})$$

e

$$r_{\Omega_q} = \alpha_q c_i \quad (\text{A.2})$$

onde c_i é a máxima distância do nó i aos nós vizinhos mais próximos; α_s e α_q são constantes arbitrárias que devem ser definidas para cada aplicação. Para melhorar a precisão da solução, um refinamento de r_{Ω_s} e r_{Ω_q} pode ser necessário, através dos parâmetros α_s e α_q especificados nas equações (A.1) e (A.2). Portanto, estes parâmetros, denominados parâmetros de discretização, são ambos considerados variáveis do domínio do modelo ILMF, e conseqüentemente variáveis do processo de otimização multiobjetivo, para a obtenção automática de seus valores ótimos.

A Seção 3.5 e a Figura 3.2 apresentam o suporte compacto, cujo tamanho é definido e controlado pelo α_s . De acordo com Liu e Gu (2005) e Atluri e Zhu (2000), geralmente um valor de $\alpha_s = 2.0$ a 3.0 propicia bons resultados para a maioria dos problemas. Nesta pesquisa, os limites impostos na otimização para esse parâmetro foram $\alpha_s = [1.5, 10.0]$.

A Seção 3.8 e a Figura 3.4 apresentam o domínio local ou domínio da forma fraca, cujo tamanho é controlado pelo α_q , que altera o raio de cobertura do suporte compacto. Esse número, assim como o tamanho do suporte compacto, é totalmente experimental, deve ser testado e aperfeiçoado. Geralmente um valor de $\alpha_q = 0.4$ a 0.6 conduz a bons resultados para a maioria dos problemas, conforme Liu e Gu (2005) e Atluri e Zhu (2000). Nesta pesquisa, os limites impostos na otimização para esse parâmetro foram $\alpha_q = [0.20, 0.95]$.

B - TIPOS INCOMUNS DE GRÁFICOS UTILIZADOS

Apesar de terem uso comum e frequente em áreas como estatística e ciência de dados, os gráficos do tipo: *parallel coordinates* (coordenadas paralelas), *boxplot* (diagrama de caixa), *violin* (violino), *heatmap* (mapa de calor), dispersão com regressão linear e distribuição de frequências e, *pairwise* (emparelhados); não são usualmente empregados na engenharia. Esses gráficos são muito úteis e indispensáveis para a exploração e compreensão de dados, principalmente quando estes são extensos. Por isso, a seguir está uma explanação sucinta sobre cada um desses tipos de gráficos.

B.1 - PARALLEL COORDINATES

Coordenadas paralelas são uma maneira comum de visualizar e analisar conjuntos de dados com muitas dimensões.

Para mostrar um conjunto de pontos em um espaço n-dimensional, um pano de fundo é desenhado consistindo de n linhas paralelas, normalmente verticais e igualmente espaçadas. Um ponto no espaço n-dimensional é representado como uma polilinha com vértices nos eixos paralelos; a posição do vértice no eixo i-ésimo corresponde à coordenada i-ésima do ponto.

Essa visualização está intimamente relacionada à visualização de séries temporais, exceto que é aplicada a dados onde os eixos não correspondem a pontos no tempo e, portanto, não têm uma ordem natural. Portanto, diferentes arranjos de eixos podem ser de interesse.

A Figura B.1 mostra um exemplo prático do emprego desse tipo de gráfico. Ele foi apresentado nos resultados dessa pesquisa, na Seção 7.3.3 (Placa com Trinca Inclinada de Borda – Modo Misto), e mostra os resultados da otimização para as variáveis: α_S , α_Q ; e para as funções objetivo: *compliance*, K_I e K_{II} . Cada solução, contendo um valor para cada variável e cada função objetivo, obtida no processo de otimização é representada por uma polilinha com vértices em cada linha vertical (coordenada) que representam esses parâmetros (variáveis e funções objetivo). Cada solução é discriminada por cor, de acordo com o algoritmo que a obteve. Com isso, fica fácil a visualização da performance de cada algoritmo utilizado, da relação entre os valores e da concentração deles em cada zona das coordenadas. Portanto, pode-se observar, nesse exemplo, que todos algoritmos obtiveram resultados similares para as funções objetivo K_I e K_{II} , dada a concentração das linhas na

proximidade de zero para as respectivas coordenadas. O mesmo não acontece quando se observa as outras três coordenadas (alfaS, alfaQ e *compliance*), pois há um espalhamento das polilinhas ao longo do intervalo das respectivas coordenadas.

Os valores a serem plotados devem ser escalados/normalizados, se não obedecerem ao mesmo intervalo para cada coordenada paralela utilizada.

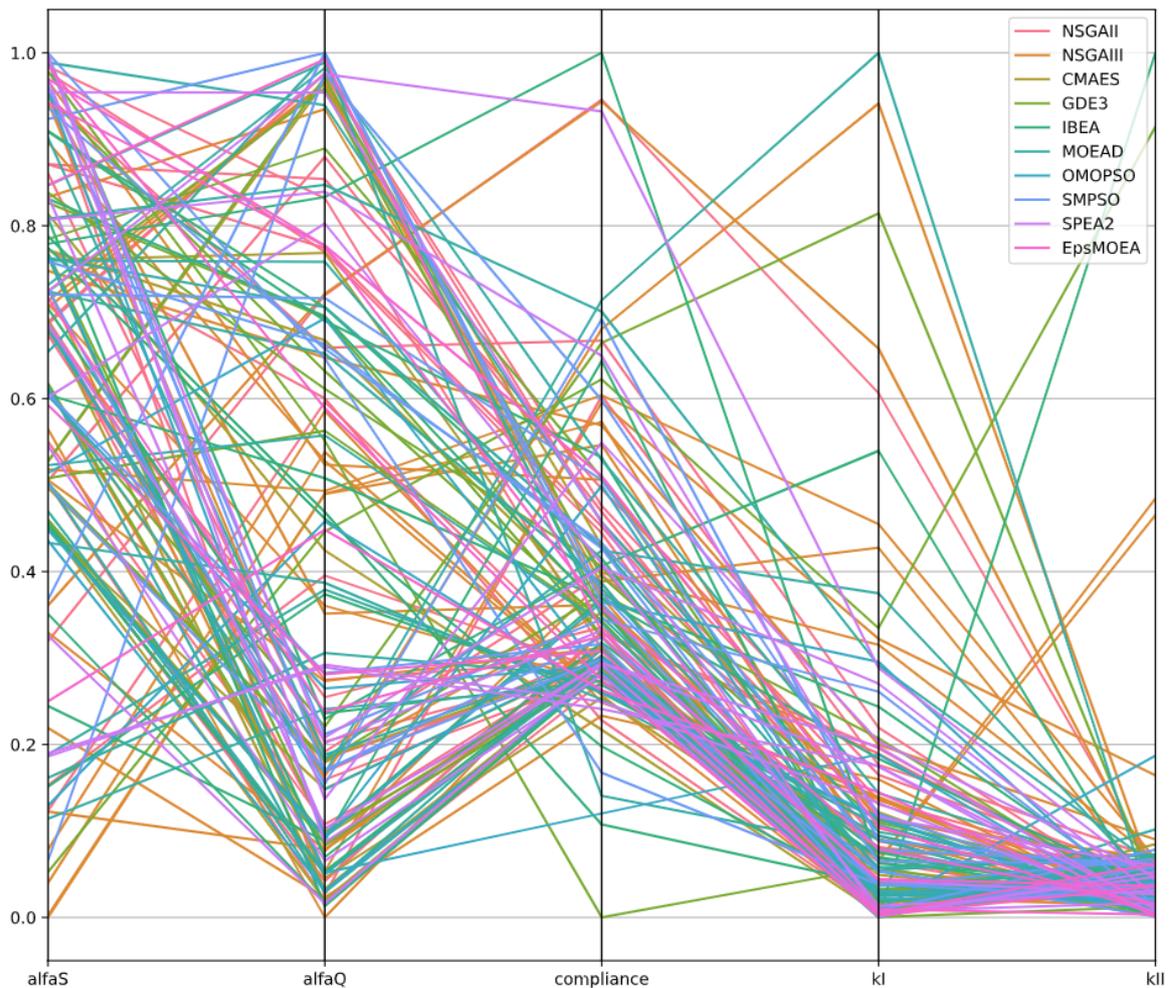


Figura B.1 – Exemplo de gráfico com coordenadas paralelas.

A Figura B.2 coloca esquematicamente um gráfico de distribuição de frequência em cada coordenada paralela apresentada, para passar a ideia da distribuição das ocorrências dos dados que o gráfico de coordenadas paralelas também passa, em função da concentração/dispersão das polilinhas em cada coordenada.

¹Acessado em 05/04/2021 no link: <https://datavizproject.com/data-type/parallel-coordinates/>

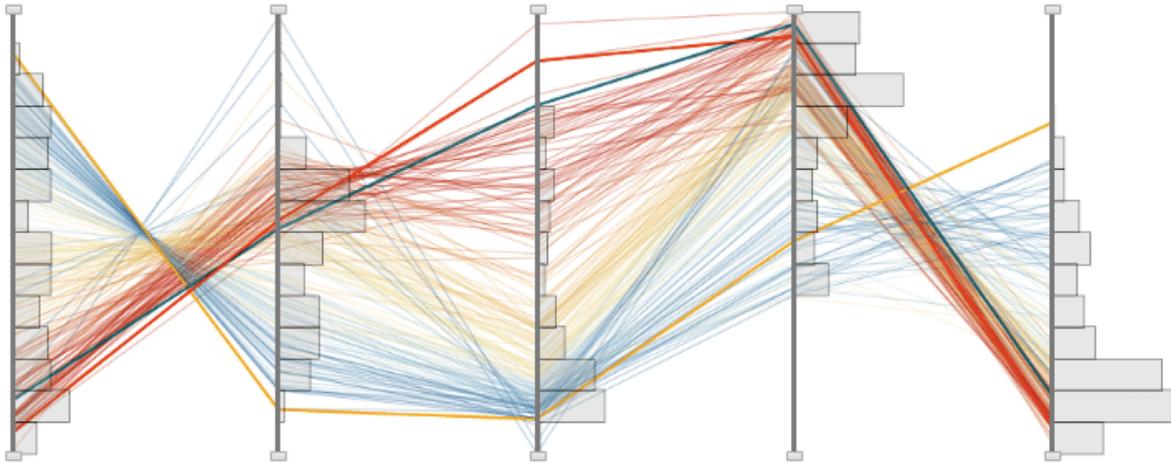


Figura B.2 – Exemplo de gráfico com coordenadas paralelas e frequências das ocorrências. (Modificado de Data Viz Project¹).

B.2 - BOXPLOT

Um diagrama de caixa é uma forma padronizada de exibir uma distribuição de dados baseada em medidas descritivas (mínimo, primeiro quartil - Q1, mediana, terceiro quartil - Q3 e máximo), conforme mostra a Figura B.3. Ele pode informar sobre seus valores discrepantes (*outliers*) e quais são seus valores. Ele também pode dizer se os dados são simétricos, o quanto os dados estão agrupados e se e como eles estão distorcidos.

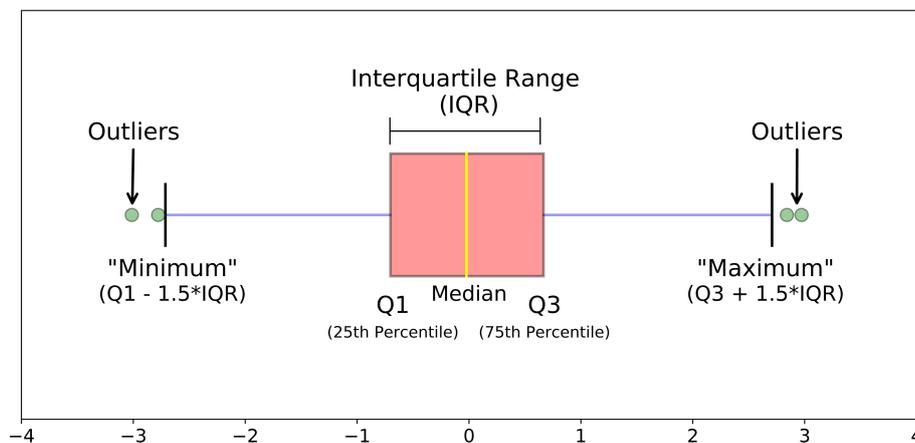


Figura B.3 – Exemplo de diagrama de caixa e suas medidas descritivas. (Em KDnuggets²).

Valores que residem fora dos limites mínimo e máximo são considerados *outliers* (discrepâncias). Esses limites são obtidos a partir do intervalo interquartil ($IQR = IQ3 - Q1$), conforme as equações apresentadas nessa mesma figura.

A Figura B.4 mostra as similaridades entre um *boxplot* de uma distribuição quase normal e

²Acessado em 05/04/2021 no link: <https://www.kdnuggets.com/2019/11/understanding-boxplots.html>

a função de densidade de probabilidade (pdf) para uma distribuição normal. É mais comum olhar para uma distribuição estatística do que para um diagrama de caixa.

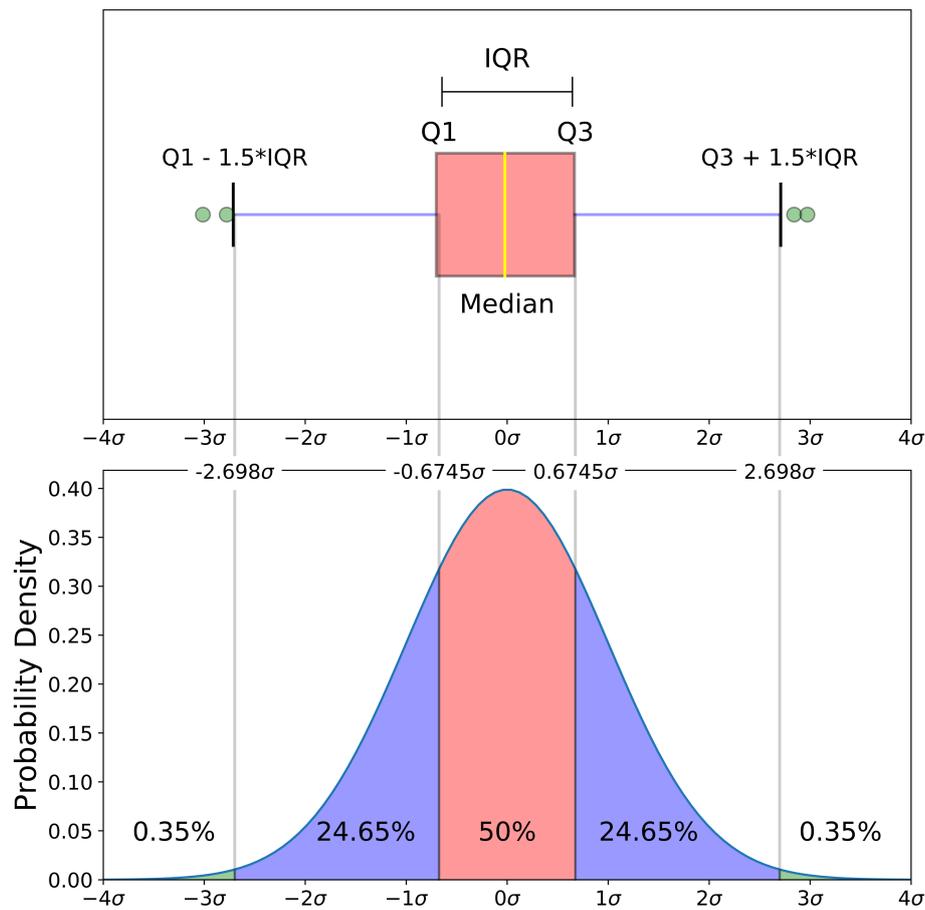


Figura B.4 – Similaridades entre o diagrama de caixa e uma curva de distribuição. (Em KDnuggets³).

B.3 - VIOLIN

O “violino” é delineado pelas curvas no entorno do gráfico *boxplot*, e representa a função densidade de probabilidade estimada via kernel (basicamente, as curvas mais largas representam maior densidade de pontos, ou seja, existe uma maior frequência de pontos). Esse tipo de representação pode ser útil para alguns conjuntos de dados, pois pelo fato do *boxplot* resumir os dados em cinco medidas descritivas, pode haver perda de informação. Já o “violino” resume os dados em uma função densidade, caracterizando melhor o conjunto de dados.

A Figura B.5 apresenta uma analogia entre o gráfico violino e o *boxplot*. Até porque o gráfico

³Acessado em 05/04/2021 no link: <https://www.kdnuggets.com/2019/11/understanding-boxplots.html>

violino traz um *boxplot* em seu interior.

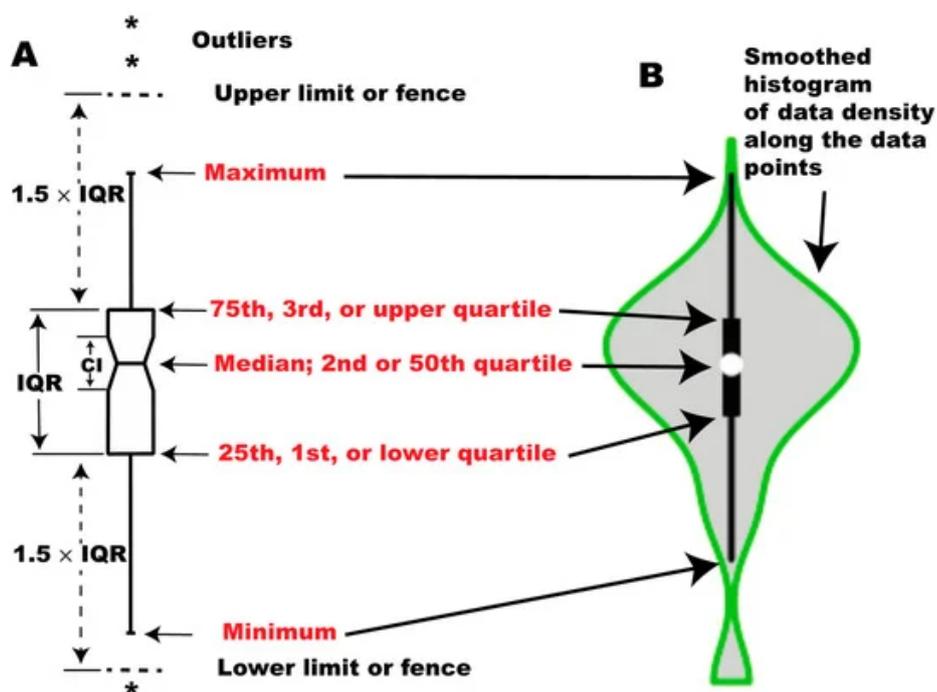


Figura B.5 – Exemplo de gráfico violino. (Em MDPI⁴).

B.4 - HEATMAP

O gráfico mapa de calor é uma técnica de visualização de dados que mostra a magnitude de um fenômeno como cor, em duas dimensões. A variação na cor pode ser por matiz ou intensidade, dando pistas visuais óbvias ao analista sobre como o fenômeno está agrupado.

A matiz de cores funciona como se "acrescentasse uma terceira dimensão em um gráfico 2D".

A Figura B.6 apresenta um exemplo de *heatmap* com resultados dessa pesquisa. É comum o uso desse tipo de gráfico quando da análise de correlação entre variáveis.

B.5 - DISPERSÃO COM REGRESSÃO LINEAR E DISTRIBUIÇÃO DE FREQUÊNCIAS

Esse tipo de gráfico propicia visualizar se há uma relação linear entre duas variáveis, pois a partir dos pontos de dispersão é obtida uma reta de regressão linear. Além disso, através

⁴Acessado em 05/04/2021 no link: <https://www.mdpi.com/2409-9279/3/4/64>

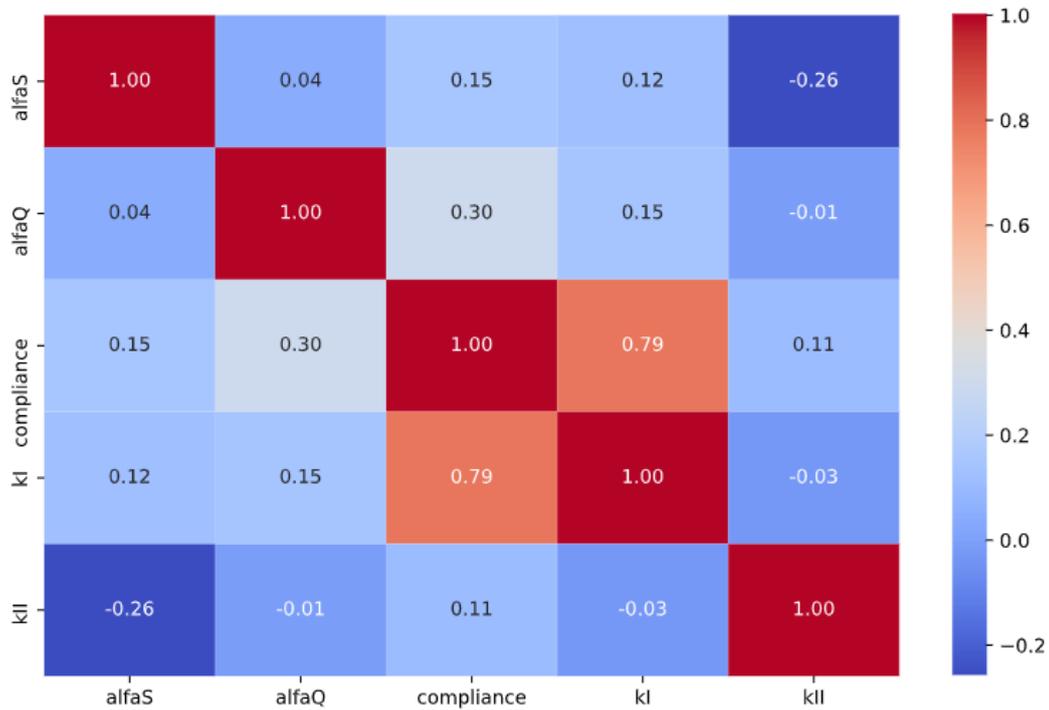


Figura B.6 – Exemplo de gráfico mapa de calor.

do histograma de distribuição de frequências plotado para cada eixo é possível entender a distribuição dos valores ao longo de cada eixo.

A Figura B.7 mostra um exemplo obtido a partir dos resultados dessa pesquisa. Ele indica claramente a correlação que há entre o *compliance* e o K_I , para o caso estudado. Também mostra, a partir dos gráficos de distribuição de frequências, que há uma concentração das soluções na proximidade de zero, para os dois parâmetros.

B.6 - PAIRWISE

Os gráficos emparelhados permitem a comparação entre as variáveis analisadas. Cruzando um parâmetro do eixo vertical com outro do eixo horizontal é possível visualizar a relação que há entre eles, se há uma correlação, etc. Os gráficos da diagonal descendente apresentam a forma com que os valores são distribuídos para cada parâmetro. A Figura B.8 traz um exemplo elaborado a partir de resultados de estudo de caso nesse texto.

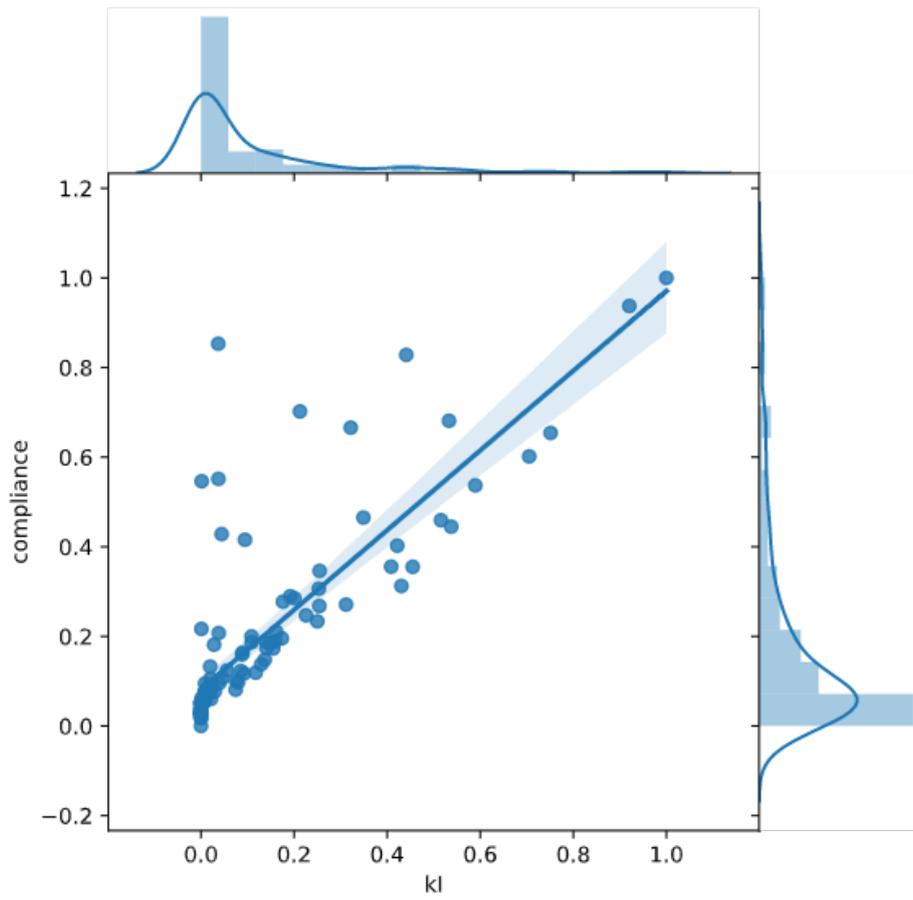


Figura B.7 – Exemplo de gráfico de dispersão com regressão linear e distribuição de frequências.

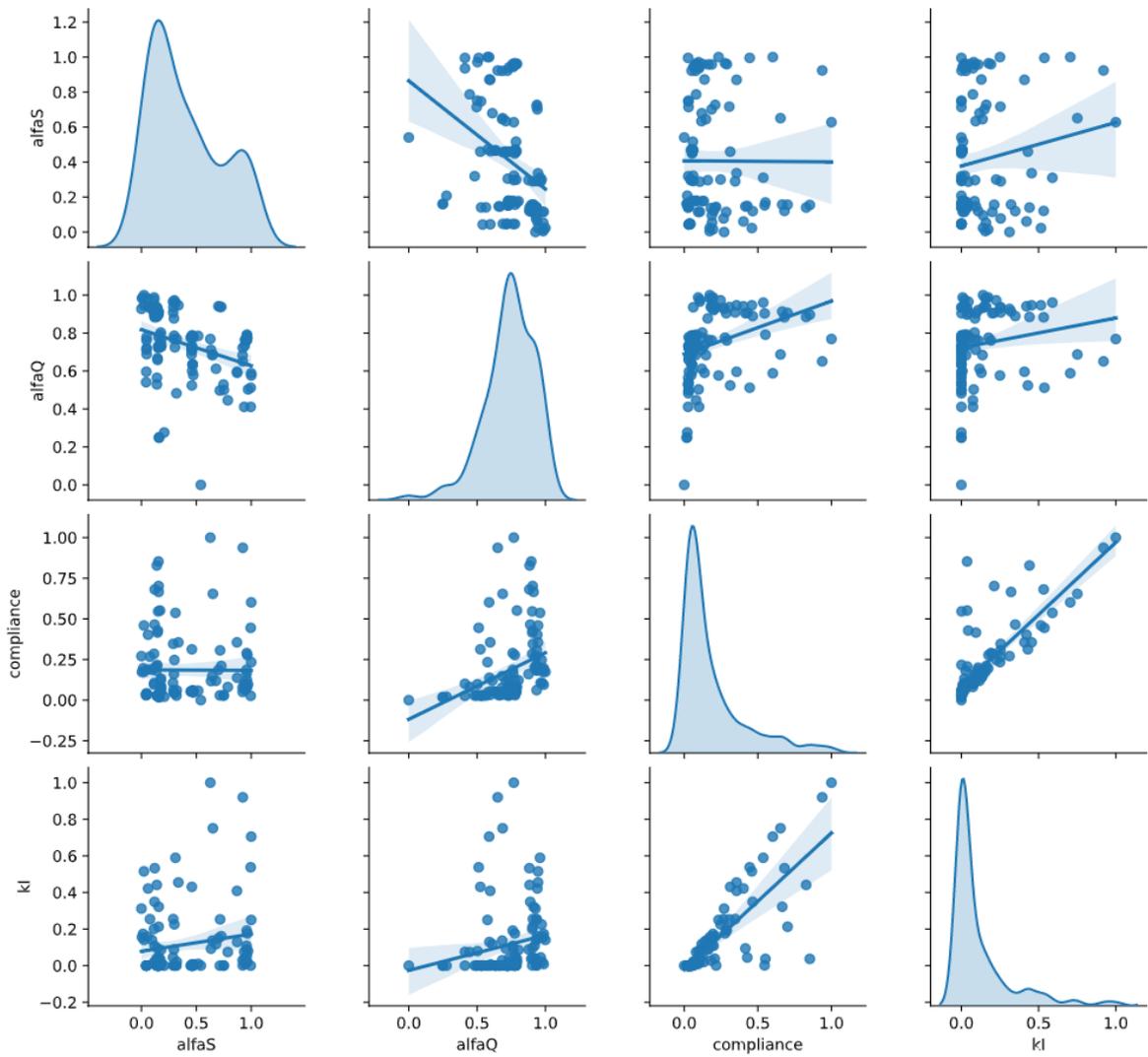


Figura B.8 – Exemplo de gráficos emparelhados.

C - PUBLICAÇÕES

Com o intuito de dar publicidade aos estudos que foram desenvolvidos ao longo desta tese e incentivar o desenvolvimento de novos trabalhos utilizando a abordagem proposta, foi elaborado em paralelo o artigo inovador: *A local mesh free numerical method with automatic parameter optimization*, com toda a parte de otimização numérica dos parâmetros adimensionais dos métodos sem malha.

Adicionalmente, outro artigo foi publicado, que também trata dos temas desenvolvidos nesta pesquisa: *A local mesh free method for linear elasticity and fracture mechanics*, como um tributo ao professor Brebbia, contendo as etapas iniciais do desenvolvimento das aplicações em mecânica da fratura.

Os artigos foram aceitos e publicados pela ELSEVIER¹ em seu *journal: Engineering Analysis with Boundary Elements*², que tem classificação A1 pela CAPES. A seção seguinte apresenta a primeira página desses artigos.

A ELSEVIER é a maior editora de literatura médica e científica do mundo, fazendo parte do grupo Reed Elsevier. Localizada em Amsterdã, a companhia tem grandes operações no Reino Unido, EUA, Europa e no Brasil.

¹<https://www.elsevier.com/>

²<http://www.journals.elsevier.com/engineering-analysis-with-boundary-elements/>



Contents lists available at ScienceDirect

Engineering Analysis with Boundary Elements

journal homepage: www.elsevier.com/locate/enganabound

A local mesh free numerical method with automatic parameter optimization



E. Santana, T. Oliveira, W. Vélez, A. Araújo, F. Martins, A. Portela*

University of Brasília, Faculty of Technology, Department of Civil Engineering, Brasília, DF 70910-900, Brazil

ARTICLE INFO

Keywords:

Local mesh less method
Local mesh free method
Parameter optimization
Genetic algorithms
Symbiotic organisms search
Local formulation

ABSTRACT

The concern of the paper is a local mesh free numerical method, implemented with automatic optimization of the discretization parameters, to solve problems of two-dimensional linear elasticity.

For a nodal discretization, the mesh free local method uses a node-by-node process to generate the global system of equilibrium equations; in the local domain of each node, the respective equilibrium equations are generated with a reduced numerical integration and used as the formulation basis of the numerical method.

Application of the local numerical method, to each node of a mesh free discretization, requires the use of two arbitrary parameters which specify the size of, respectively the compact support and the local domain of integration. In this paper, these parameters are automatically defined by means of a multi-objective optimization process, based on genetic algorithms, which is the novelty of the present numerical method. For the sake of efficiency a comparison between genetic algorithms and another robust evolutionary optimization method, the symbiotic organism search algorithm, is carried out.

A benchmark problem was analyzed to assess the accuracy and efficiency of these techniques. The results presented in the paper are in perfect agreement with those of analytical solutions and therefore, make reliable and robust this mesh free numerical method, implemented with automatic parameter optimization.

1. Introduction

The ILMF local mesh free numerical method, presented by Oliveira et al. [1], is considered in this paper, for the sake of simplicity. Derived from the local work theorem, the method is implemented with a reduced numerical integration with a single integration point on each boundary of the local domain of integration. As reported by Oliveira et al. [1], the linear reduced integration has the desirable effect of increasing the solution accuracy of the ILMF model, induced by a reduction of the nodal stiffness. This is the main feature of ILMF.

Mesh free numerical methods reached an extraordinary progress over the past few years. The essential attribute of these methods is that they discretize the domain and boundaries of the problem with a set of disperse nodes that do not need of any mesh for the approximation of variables. In the past, most of known mesh free methods count on background cells to perform the integration of the weighted residual weak form on the domain of the body, in the generation of the system of algebraic equations and consequently, they are not considered truly mesh free methods.

Local mesh free methods, based on weighted residual local weak forms, have been developed to avoid the background mesh generation in cells. The most popular of these methods is the Meshless Local Petrov-Galerkin (MLPG) method, see Atluri et al. [2,3], that is based on the well known moving least-squares (MLS) approximation. The main feature of this method is that local weak forms are used, for integration on regular-shaped local regions, instead of global weak forms and therefore, the method does not require using a global background mesh, but only a local background mesh.

Aiming to further improvements of the computational efficiency and the numerical accuracy of local mesh free methods, the ILMF method was recently presented by Oliveira et al. [1].

In general, the accuracy and efficiency of local mesh free methods is directly determined by two nodal parameters of the discretization which are, respectively the size of the nodal compact support, parameter primarily linked to the accuracy of the model through the total number of nodes used in the shape function definition, and the size of the local integration domain of each node, parameter primarily linked to the efficiency of the model. Therefore, these discretization parameters play an important role in the modeling strategy. Both parameters are usually defined arbitrarily and can vary depending on the nodal configuration

* Corresponding author.

E-mail address: aportela@unb.br (A. Portela).<https://doi.org/10.1016/j.enganabound.2019.12.013>Received 8 April 2019; Received in revised form 13 June 2019; Accepted 17 December 2019
0955-7997/© 2019 Elsevier Ltd. All rights reserved.



Contents lists available at ScienceDirect

Engineering Analysis with Boundary Elements

journal homepage: www.elsevier.com/locate/enganabound

A local mesh free method for linear elasticity and fracture mechanics

T. Oliveira, W. Vélez, E. Santana, T. Araújo, F. Mendonça, A. Portela*



Department of Civil and Environmental Engineering, Faculty of Technology, University of Brasília, Brasília-DF 70910-900, Brazil

ARTICLE INFO

Keywords:

Local mesh free methods
 Reduced numerical integration
 Multi objective optimization
 Genetic algorithms
 Linear elastic fracture mechanics
 Singularity subtraction technique

ABSTRACT

The paper presents recent developments on mesh free numerical methods, implemented at the Department of Civil Engineering and Environment of the University of Brasília.

The concern is a local mesh free numerical model developed to solve two-dimensional problems in elasticity and linear elastic fracture mechanics.

The model formulation is based in the work theorem of the theory of structures, kinematically formulated with a simple rigid-body displacement.

The discretization considers the approximation of the elastic field with moving least squares (MLS) and implements a reduced numerical integration which improves the model accuracy. Both regular and irregular nodal distributions can be considered, which makes it a reliable model.

The arbitrary discretization parameters, which determine the accuracy and efficiency of local mesh free methods, can be generated automatically, in a multi objective optimization environment, with genetic algorithms (GA), which makes it a robust model.

Applications of linear elastic fracture mechanics implement the singularity subtraction technique (SST) to regularize the elastic field, before the numerical solution, introducing the stress intensity factors (SIF) as additional unknowns of the problem. These features make it an efficient model.

Benchmark problems were solved for an assessment of the accuracy and efficiency of these techniques.

1. Introduction

The work theorem has been postulated as a unifying tool in the formulation of numerical methods in continuum mechanics, as early reported by Brebbia [1] and Portela [2].

This paper presents a local mesh free numerical model, derived from the work theorem, kinematically formulated with a simple rigid-body displacement. Using the MLS approximation, the model implements a reduced numerical integration, on the boundaries of the local domain associated to each node, with 1 single point per segment, which leads to a point-wise discrete form that represents equilibrium of tractions defined at integration points. The reduced integration, a key feature of this formulation, induces a reduction of the nodal stiffness which, in turn, increases the solution accuracy, presenting no instabilities at all, unlike nodal integration without stabilization, used by other mesh free methods, as reported by Chen et al. [3].

The issue of numerical stability is quite important. In the finite element method (FEM), elements with a reduced integration are commonly employed because they are computationally efficient and avoid locking of fully integrated elements. As a side effect, such elements with reduced integration are susceptible to spurious singular modes, named as hourglass modes, which are zero-energy modes in which the element

can deform without an associated increase of the internal energy. These spurious modes, generated by a reduced integration, can be prevented through stabilization techniques, known as hourglass stabilization, as reported by Zienkiewicz et al. [4] and Bathe [5]. Reduced integration has also been exploited in XFEM, as reported by Amiri et al. [6].

The reduced integration is the main source of the numerical instability of some mesh free methods, leading to unstable hourglass deformation and zero-energy modes. This is the case of the EFG method, see Beissel et al. [7], and the mesh free particle method, see Belytschko et al. [8]. As reported by Chen et al. [3], nodal integration in mesh free methods, without stabilization, leads to instabilities due to the fact that each node is associated with a support domain, where integrations are carried out, to compute the nodal stiffness. This means that each domain of integration is associated with just 1 integration point, that is the node and thus, when only 1 point is considered for the integration of higher order functions, the nodal integration inevitably causes fatal instabilities. In contrast, the mesh free method with reduced integration, presented in this paper, considers a total of 4 integration points on the boundaries of the local domain, with 1 point per segment which, therefore, conveniently prevents the generation of spurious zero-energy modes. In order to overcome solution instabilities that are present in direct nodal integration, Taylor series expansions have been used, to

* Corresponding author.

E-mail address: aportela@unb.br (A. Portela).<https://doi.org/10.1016/j.enganabound.2019.01.007>Received 26 October 2018; Received in revised form 23 December 2018; Accepted 8 January 2019
0955-7997/© 2019 Elsevier Ltd. All rights reserved.