



TIAGO FERNANDO CHÍCHARO DAS NEVES
Licenciado em Ciências da Engenharia Mecânica

OTIMIZAÇÃO MULTIOBJETIVO COM E SEM DERIVADAS APLICADA A ESTRUTURAS MECÂNICAS MULTICARGA

MESTRADO EM ENGENHARIA MECÂNICA

Universidade NOVA de Lisboa
Abril, 2025



OTIMIZAÇÃO MULTIOBJETIVO COM E SEM DERIVADAS APLICADA A ESTRUTURAS MECÂNICAS MULTICARGA

TIAGO FERNANDO CHÍCHARO DAS NEVES

Licenciado em Ciências da Engenharia Mecânica

(pt)**Orientador:** Pedro Samuel Gonçalves Coelho
Professor Associado, Universidade NOVA de Lisboa

Coorientadora: Ana Luísa Custódio
Professora Associada, Universidade NOVA de Lisboa

Júri

Presidente: Doutor José Manuel Cardoso Xavier
Professor Associado, FCT-NOVA

Arguente: Doutor Fábio Rúben Monteiro Conde
Professor Auxiliar, FCT-NOVA

Orientador: Doutor Pedro Samuel Gonçalves Coelho
Professor Associado, FCT-NOVA

Otimização multiobjetivo com e sem derivadas aplicada a estruturas mecânicas multicarga

Copyright © Tiago Fernando Chícharo das Neves, Faculdade de Ciências e Tecnologia, Universidade NOVA de Lisboa.

A Faculdade de Ciências e Tecnologia e a Universidade NOVA de Lisboa têm o direito, perpétuo e sem limites geográficos, de arquivar e publicar esta dissertação através de exemplares impressos reproduzidos em papel ou de forma digital, ou por qualquer outro meio conhecido ou que venha a ser inventado, e de a divulgar através de repositórios científicos e de admitir a sua cópia e distribuição com objetivos educacionais ou de investigação, não comerciais, desde que seja dado crédito ao autor e editor.

AGRADECIMENTOS

A conclusão desta dissertação é algo que nunca teria sido possível sem o auxílio e incentivo de várias pessoas.

Ao meu orientador, Professor Pedro Coelho, e à minha co-orientadora, Professora Ana Luísa Custódio, os meus mais profundos agradecimentos por todo o apoio que deram e todo o esforço que cometeram à realização desta dissertação.

Aos meus avós, por nunca terem duvidado que isto ia acontecer.

E principalmente à minha mãe, por ter acreditado em mim mesmo quando não tinha razões para isso.

RESUMO

A otimização estrutural desempenha um papel crucial na Engenharia Mecânica, permitindo o desenvolvimento de estruturas mais eficientes. De entre as suas diferentes variantes, a otimização topológica é particularmente relevante, já que permite determinar a distribuição ótima de material num dado domínio de projeto. Quando múltiplos critérios de desempenho devem ser considerados simultaneamente, a otimização topológica multiobjetivo torna-se uma ferramenta essencial na análise de objetivos conflituosos, como a rigidez e a redução de peso. No decorrer desta dissertação procurou-se estudar a adequação de diferentes algoritmos de otimização quando aplicados a esta classe de problemas, em diferentes contextos. A maior contribuição desta dissertação consiste na exploração de métodos e algoritmos até agora não considerados na resolução de problemas em otimização topológica.

Inicialmente, foi analisado o desempenho numérico do algoritmo *Direct Multisearch* (DMS), numa classe de problemas académicos, permitindo dimensões variáveis tendo a sua performance sido comparada com uma sua variante, o BoostDMS. Seguidamente, consideraram-se duas formulações distintas para problemas de otimização topológica biobjetivo. Foi resolvido em primeiro lugar um problema formulado em domínio contínuo, sendo que a viabilidade do uso do DMS e do BoostDMS foi estudada utilizando valores obtidos pelo *Optimality Criteria Method* (OC) como referência. Os resultados obtidos permitem concluir que um problema formulado em domínio contínuo terá associado um número demasiado elevado de variáveis de projeto, inviabilizando a utilização do DMS e BoostDMS, devido aos tempos computacionais associados. Os problemas formulados em domínio discreto, com menor número de variáveis de projeto, revelam o potencial do DMS em otimização topológica. As aproximações das frentes de Pareto obtidas por este algoritmo, apesar de parcialmente dominadas, apresentam uma elevada densidade de pontos, quando comparadas às aproximações obtidas pelo *Method of Moving Asymptotes* (MMA), tornando esta classe de algoritmos uma interessante alternativa para problemas de otimização topológica formulados em domínio discreto.

Palavras-chave: Otimização topológica multiobjetivo, *Direct MultiSearch* (DMS), *Method*

of Moving Asymptotes (MMA), Optimality Criteria Method (OC), Formulação em domínio contínuo, Formulação em domínio discreto

ABSTRACT

Structural optimization plays a crucial role in Mechanical Engineering, enabling the development of more efficient structures. Among its different variants, topology optimization is particularly relevant, as it determines the optimal material distribution within a given design domain. When multiple performance criteria must be considered simultaneously, multiobjective topology optimization becomes an essential tool for analyzing conflicting objectives, such as stiffness and weight reduction. This dissertation aims to study the suitability of different optimization algorithms when applied to this class of problems in various contexts. The main contribution of this dissertation lies in the exploration of methods and algorithms that have not yet been considered for solving problems in Mechanical Engineering.

Initially, the numerical performance of the *Direct Multisearch* (DMS) algorithm was analyzed in a class of academic problems, allowing for variable dimensions. Its performance was compared with its variant, BoostDMS. Subsequently, two distinct formulations for biobjective topology optimization problems were considered. First, a problem formulated in a continuous domain was solved, with the feasibility of using DMS and BoostDMS studied using as reference values obtained by the *Optimality Criteria Method* (OC). The results indicate that a problem formulated in a continuous domain involves an excessively high number of design variables, making the use of DMS and BoostDMS impractical due to associated computational times. However, problems formulated in a discrete domain, with fewer design variables, highlight the potential of DMS in topology optimization. The approximated Pareto front obtained by this algorithm, although partially dominated, exhibit a high density of points compared to those obtained by the *Method of Moving Asymptotes* (MMA), making this class of algorithms an interesting alternative for topology optimization problems formulated in a discrete domain.

Keywords: Multiobjective topology optimization, *Direct MultiSearch* (DMS), *Method of Moving Asymptotes* (MMA), *Optimality Criteria Method* (OC), Formulation in a continuous domain, Formulation in a discrete domain

ÍNDICE

Índice de Figuras	ix
Índice de Tabelas	xi
1 Motivação e objetivos	1
1.1 Introdução	1
1.2 Objetivos	1
2 Otimização multiobjetivo	3
2.1 Abordagem de escalarização	4
Algoritmos de otimização associados a escalarização - Método do Critério de Otimalidade e Método das Assíntotas Móveis	5
2.2 Abordagem de Pareto	6
Algoritmos de otimização que recorrem à abordagem de Pareto- DMS e BoostDMS	7
3 Otimização topológica	9
3.1 Formulação do problema multicarga num domínio contínuo	9
3.2 Formulação do problema multicarga num domínio discreto	12
4 Simulação numérica	15
4.1 Problemas académicos	15
4.2 Análise no contexto de otimização estrutural	20
4.2.1 Resolução de um problema de otimização topológica no contínuo	20
4.2.2 Resolução de um problema de otimização topológica no domínio discreto	27
5 Conclusões e trabalho futuro	44
Apêndices	46

A Problemas académicos	47
B Otimização topológica em domínio discreto	56
Bibliografia	59

ÍNDICE DE FIGURAS

2.1	Passo de sondagem no algoritmo DMS realizado a partir do centro x_k , a resultar na geração de novos pontos não dominados (a verde escuro), num problema biobjetivo.	8
3.1	Exemplo do processo de otimização topológica num problema de elasticidade.	10
3.2	Domínio de projeto Ω e representação do esquema de forças considerado. .	11
3.3	Problema de otimização topológica formulado num domínio discreto. . . .	12
4.1	Aproximações da frente de Pareto obtidas pelos algoritmos DMS e BoostDMS, para o problema DTLZ2, com $n = 10, 20, 40, 60$	16
4.2	Perfis de desempenho para a coleção de problemas com dimensão $n = 10$. .	18
4.3	Perfis de desempenho para a coleção de problemas com dimensão $n = 20$. .	18
4.4	Perfis de desempenho para a coleção de problemas com dimensão $n = 40$. .	19
4.5	Perfis de desempenho para a coleção de problemas com dimensão $n = 60$. .	19
4.6	Frente de Pareto esperada para um domínio de dimensões 60x60, obtida pelo OC.	20
4.7	Aproximações da frente de Pareto obtidas com o <i>top88</i> e o <i>top99</i> para malhas de diferentes dimensões.	21
4.8	Aproximações da frente de Pareto obtidas pelo <i>top88</i> para os dois tipos de filtro em malhas de diferentes dimensões.	23
4.9	Aproximação da frente de Pareto obtida pelo OC, para uma malha de 10x10, considerando 100 valores de peso distintos.	24
4.10	Aproximação da frente de Pareto obtida pelo DMS, permitindo os 230 segundos de tempo computacional utilizados pelo OC.	24
4.11	<i>Zoom-in</i> da aproximação da frente de Pareto e algumas estruturas obtidas pelo DMS para uma malha de 10x10, permitido 230 segundos de tempo computacional.	25
4.12	Aproximação da frente de Pareto obtida pelo DMS para uma malha de 10x10, permitindo 100000 avaliações da função objetivo.	26

4.13	<i>Zoom-in</i> da aproximação da frente de Pareto e algumas estruturas obtidas pelo DMS para uma malha de 10x10, permitindo 100000 avaliações da função objetivo.	27
4.14	Estruturas de barras num domínio retangular com um arranjo regular de 5 por 3 nós, para transmitir uma força vertical para uma linha vertical de suportes: (a) domínio simplificado, (b) domínio com variáveis de projeto adicionais, (c) todas as conexões possíveis entre nós estão incluídas [4].	28
4.15	Informação associada ao problema de estrutura de barras considerado. . . .	29
4.16	Soluções esperadas para (a) $w = 1$ e (b) $w = 0$, considerando uma fração volúmica de aproximadamente 0.4.	31
4.17	Aproximação da frente de Pareto obtida pelo MMA para a formulação em domínio discreto.	32
4.18	Evolução das áreas dos elementos barra em função do valor de peso.	32
4.19	Aproximação da frente de Pareto obtida pelo DMS, permitindo 10000 avaliações da função objetivo.	33
4.20	Aproximação da frente de Pareto obtida pelo DMS, permitindo 20000 avaliações da função objetivo.	34
4.21	Aproximações das frentes de Pareto obtidas pelo DMS e pelo MMA, permitindo um tempo computacional máximo de 25 segundos.	35
4.22	Aproximações das frentes de Pareto obtidas pelo DMS e pelo BoostDMS, permitindo um tempo computacional de 25 segundos.	36
4.23	Informação associada ao problema de 20 elementos de barra considerado. . .	37
4.24	Aproximação da frente de Pareto obtida pelo MMA para o problema de 20 elementos barra.	38
4.25	Aproximações da frente de Pareto obtidas pelo MMA e pelo DMS para o problema de 20 elementos barra, permitindo um tempo computacional máximo de 42 segundos.	38
4.26	Aproximações da frente de Pareto obtidas pelo DMS e pelo BoostDMS para o problema de 20 elementos barra, permitindo um tempo computacional de 42 segundos.	39
4.27	Informação associada ao problema de 28 elementos de barra considerado. . .	40
4.28	Aproximação da frente de Pareto obtida pelo MMA para o problema de 28 elementos barra.	41
4.29	Aproximações das frentes de Pareto obtidas pelo MMA e pelo DMS para o problema de 28 elementos barra, permitindo um tempo computacional máximo de 50 segundos.	41
4.30	Aproximação da frente de Pareto obtida pelo DMS para o problema de 28 elementos barra, permitindo um máximo de 100000 avaliações da função objetivo e comparação com a solução obtida com o MMA.	42

ÍNDICE DE TABELAS

4.1	Coordenadas dos pontos de Pareto obtidos para malhas de dimensões 10x10 e 30x30, utilizando o <i>top88</i> e o <i>top99</i>	22
4.2	Algumas soluções e valores de variáveis de projeto correspondentes obtidos pelo DMS, considerando um máximo de 100000 avaliações da função objetivo.	35

MOTIVAÇÃO E OBJETIVOS

1.1 Introdução

Enquanto área, a Engenharia desempenha um papel fundamental no desenvolvimento da civilização e evolução científica. Engenheiros encontram-se na linha da frente em inúmeros avanços tecnológicos, contribuindo para o desenvolvimento de infraestruturas, transportes, comunicação e várias outras indústrias. No projeto de estruturas, o engenheiro deve ter em consideração os materiais que estão a ser utilizados e as suas propriedades mecânicas específicas, assim como as forças e os deslocamentos que estarão aplicados na estrutura em condições de funcionamento. A presente dissertação insere-se no âmbito da Engenharia Mecânica, com ênfase na área de otimização de estruturas mecânicas. A Otimização Estrutural consiste num processo sequencial de análises estruturais onde, de forma iterativa, se melhora uma medida de desempenho da estrutura fazendo variar de alguma forma as suas características. Tal envolve a utilização de métodos matemáticos, que permitem a descrição das características e propriedades principais de um dado sistema através de um conjunto de equações cujas variáveis se relacionam entre si. A Otimização Estrutural pode ser dividida em três grandes categorias: Otimização de Forma, Otimização Dimensional e Otimização Topológica. Esta última categoria é abordada na presente dissertação e, por isso, encontra-se explicada em maior detalhe no Capítulo 3. O objetivo desta dissertação é a otimização topológica de estruturas recorrendo a formulações multiobjetivo. Para tal, aplica-se o Método dos Elementos Finitos para simular numericamente o comportamento estrutural. Trata-se de um trabalho que combina duas grandes áreas da otimização: a Otimização Multiobjetivo e a Otimização Topológica.

1.2 Objetivos

No decorrer da presente dissertação serão propostos vários problemas de otimização topológica multiobjetivo, com o objetivo de avaliar diferentes abordagens para a sua resolução. Numa primeira fase, estes problemas serão resolvidos utilizando algoritmos de otimização com um histórico prévio de utilização na área da Engenharia Mecânica,

nomeadamente o Optimality Criteria Method (OC) [2] e o Method of Moving Asymptotes (MMA) [3], sendo ambos métodos baseados no gradiente e frequentemente aplicados em otimização estrutural. Posteriormente, os mesmos problemas serão resolvidos utilizando o Direct Multisearch (DMS), um método de otimização não baseado no gradiente. A aplicação do DMS a problemas de otimização topológica multiobjetivo é inédita, tornando esta investigação relevante para explorar a sua aplicabilidade, assim como quaisquer potenciais limitações quando aplicado a esta classe de problemas. A comparação entre os métodos será realizada considerando, principalmente, o tempo computacional necessário para a obtenção das soluções, assim como a qualidade das aproximações das frentes de Pareto geradas. Com base nesses resultados, serão retiradas conclusões acerca da viabilidade do DMS na área de otimização topológica, analisando as suas vantagens e desafios em relação às abordagens tradicionais baseadas no gradiente.

No Capítulo 2 será elaborado um enquadramento teórico do processo de otimização multiobjetivo. Aqui serão descritos alguns dos conceitos mais importantes, sendo estabelecidas as distinções entre a abordagem de escalarização, tipicamente utilizada no campo da otimização topológica, e a abordagem de Pareto, cuja inclusão na área de Engenharia Mecânica é menos comum. Associados a estas duas abordagens, serão explicados resumidamente os principais algoritmos estudados ao longo desta dissertação.

O Capítulo 3 focar-se-á sobre a otimização topológica, detalhando os principais objetivos deste tipo de otimização. Serão descritos pela primeira vez, ainda que de forma geral, o tipo de problemas que se procurará resolver ao longo desta dissertação, sendo feita uma divisão entre problemas que são formulados em domínio contínuo e aqueles que são resolvidos considerando um domínio discreto.

Finalmente, no Capítulo 4 é efetuada a resolução dos problemas introduzidos no Capítulo 3, quer em domínio discreto, quer em domínio contínuo, recorrendo aos diversos algoritmos explicados nos capítulos anteriores. É ainda feita com uma análise extensa dos resultados obtidos, com o objetivo de concluir adequadamente acerca da viabilidade do uso dos algoritmos anteriormente apresentados na área da otimização topológica.

A dissertação termina no Capítulo 5, com a apresentação das conclusões gerais e de linhas de trabalho futuro. Os anexos complementam os resultados numéricos apresentados no texto principal.

OTIMIZAÇÃO MULTIOBJETIVO

De forma sistemática, o projeto estrutural pode ser formulado como um problema de otimização, compreendendo três passos [4]:

- 1) Identificação das variáveis de projeto, que correspondem aos parâmetros dos quais a solução final irá depender. Estas devem ser tão independentes umas das outras quanto possível.
- 2) Identificação da função objetivo, que corresponde ao critério que é analisado para decidir se um dado projeto é ou não melhor do que outro. Será esta função que se pretende maximizar ou minimizar.
- 3) Identificação dos constrangimentos ou restrições de projeto, que podem refletir limitações económicas, físicas, etc., sendo que um projeto só é considerado admissível se satisfazer todos os constrangimentos.

O projeto estrutural está diretamente relacionado com a otimização multiobjetivo já que muitos problemas de engenharia envolvem múltiplas grandezas físicas que devem ser consideradas simultaneamente e frequentemente são conflituosas entre si. Uma vez que muitos projetos desta área visam aumentar a rigidez de estruturas, minimizando simultaneamente, por exemplo, o peso das mesmas, torna-se necessário transcender a otimização uniobjetivo, passando para a multiobjetivo.

Um problema de otimização multiobjetivo distingue-se de um problema de otimização uniobjetivo pelo número de funções que se pretende minimizar ou maximizar. Um problema de otimização multiobjetivo pode ser formulado matematicamente como [5]:

$$\begin{aligned}
 \min_{x \in \mathbb{R}^n} \quad & F(x) = (f_1(x), f_2(x), \dots, f_m(x)) \\
 \text{s.a.} \quad & g_j(x) \leq 0, j = 1, 2, \dots, n_d, \\
 & h_l(x) = 0, l = 1, 2, \dots, n_i
 \end{aligned} \tag{2.1}$$

onde o natural $m \geq 2$ representa o número de funções objetivo, n_d o número de restrições

de desigualdade e n_i o número de restrições de igualdade. Uma diferença importante entre problemas uniobjetivo e multiobjetivo consiste na definição de solução. Enquanto que num problema uniobjetivo a solução tende a ser um único ponto, num problema multiobjetivo tipicamente não existe uma solução única, sendo possível identificar um conjunto de pontos que podem ser todos interpretados como ótimos. O conceito predominantemente utilizado para definir um ponto como ótimo é o da otimalidade de Pareto. Um ótimo de Pareto, x^* , é um ponto pertencente ao espaço de soluções admissíveis, ou seja, que satisfaz todas as restrições impostas, tal que não existe nenhum outro ponto admissível, x , para o qual $F(x) \leq F(x^*)$ e $\exists i \in \{1, \dots, m\}, f_i(x) < f_i(x^*)$. A imagem por F do conjunto de todos estes pontos ótimos forma aquilo que é designado por frente ou fronteira de Pareto, que em otimização multiobjetivo constitui a solução do problema. Diversos métodos podem ser adotados para resolver problemas de otimização multiobjetivo e obter aproximações da frente de Pareto [5]. Os mais relevantes, no contexto desta dissertação, são a abordagem de escalarização e a abordagem de Pareto.

2.1 Abordagem de escalarização

De forma geral, a abordagem de escalarização consiste na reformulação do problema multiobjetivo como um problema uniobjetivo. Diversas técnicas podem ser utilizadas nesta tarefa. Um primeiro método, comumente utilizado, é o Método da Soma Ponderada [5], que reformula matematicamente o problema multiobjetivo como:

$$\min_{x \in \mathbb{R}^n} \varphi(x) = \min_{x \in \mathbb{R}^n} \sum_{i=1}^m w_i f_i(x), \quad (2.2)$$

onde, $w = (w_1, \dots, w_m) \geq 0$ consiste num vetor de pesos, cujos valores representam a importância que se dá à minimização de cada uma das componentes da função objetivo, traduzindo as preferências do agente de decisão. Tipicamente $\sum_{i=1}^m w_i = 1$. Este método permite obter a totalidade da frente de Pareto se o problema for resolvido para diferentes variações de pesos, assumindo que a mesma é convexa.

Outra possibilidade de escalarização consiste no Método do Produto Ponderado, tipicamente utilizado para resolver problemas cujas componentes da função objetivo apresentam semelhante importância, mas diferentes ordens de magnitude [5]:

$$\min_{x \in \mathbb{R}^n} \varphi(x) = \min_{x \in \mathbb{R}^n} \prod_{i=1}^m [f_i(x)^{w_i}]. \quad (2.3)$$

Uma desvantagem deste método é o carácter não linear da escalarização, trazendo dificuldades adicionais à resolução do problema.

Outros exemplos da abordagem de escalarização menos comumente utilizados são o Método Lexicográfico e o Método Min-Max Ponderado. No Método Lexicográfico [5], as diferentes componentes da função objetivo são ordenadas por importância, sendo resolvidos sequencialmente m problemas uniobjetivo, respeitando a ordem pré-estabelecida:

$$\begin{aligned} \min_{x \in \mathbb{R}^n} \quad & f_p(x) \\ \text{s.a} \quad & f_i(x) \leq f_i(x_{p-1}^*) \\ & i = 1, \dots, p-1, \end{aligned} \tag{2.4}$$

com p a tomar valores entre 1 e m , respeitando a ordem de importância atribuída às componentes da função objetivo, e sendo x_{p-1}^* o ótimo relativo ao problema anterior (se $i = 1$, considera-se a minimização irrestrita de f_1). Uma desvantagem deste método é que é muito sensível à ordem de importância pré-estabelecida. Diferentes ordenações irão explorar diferentes zonas da frente de Pareto. Consequentemente, podem não ser obtidas soluções que representam adequadamente a totalidade da frente de Pareto.

O Método Min-Max Ponderado [5] resolve o problema de otimização recorrendo ao conceito de ponto utopia. Num problema de otimização o ponto F^o é um ponto utopia se para cada componente da função objetivo $i = 1, 2, \dots, m$, $f_i^o = \min_x f_i(x)$. Esta condição torna o ponto utopia normalmente inatingível. A reformulação uniobjetivo do problema é dada do seguinte modo:

$$\min_{x \in \mathbb{R}^n} \varphi(x) = \min_{x \in \mathbb{R}^n} \max_i w_i [f_i(x) - f_i^o] \tag{2.5}$$

Este problema pode ser reformulado com a adição de uma variável λ :

$$\begin{aligned} \min_{x \in \mathbb{R}^n} \quad & \lambda \\ \text{s.a} \quad & w_i [f_i(x) - f_i^o(x)] - \lambda \leq 0, i = 1, 2, \dots, m \end{aligned} \tag{2.6}$$

A reformulação matemática uniobjetivo de um problema multiobjetivo tem a vantagem de simplificar o problema, permitindo a aplicação de métodos de otimização uniobjetivo na sua resolução, como o método da descida máxima, o método de Newton, etc [6]. Para além disto, a atribuição de pesos permite ao agente de decisão modelar explicitamente as suas preferências.

No entanto, a utilização deste tipo de métodos também apresenta desvantagens. Em primeiro lugar, cada resolução de uma escalarização do problema resulta, no melhor dos cenários, apenas num ótimo de Pareto. Se se pretender obter toda a fronteira de Pareto, como é o caso nesta dissertação, torna-se necessário resolver várias escalarizações, alterando os valores de w em cada início de resolução ou considerando diferentes ordens de prioridade para as componentes da função objetivo (embora sem garantias de sucesso).

Algoritmos de otimização associados a escalarização - Método do Critério de Otimalidade e Método das Assíntotas Móveis

Embora qualquer método de otimização não linear com restrições possa ser utilizado para resolver o problema uniobjetivo resultante da escalarização, nesta dissertação consideraram-se dois algoritmos de otimização específicos: o *Optimality Criteria Method* (OC), ou Método do Critério de Otimalidade [2], e o *Method of Moving Asymptotes* (MMA), ou Método das

Assintotas Móveis [3]. Ambos são métodos baseados no gradiente, uma vez que recorrem às derivadas da função objetivo para melhorar iterativamente a aproximação corrente da solução do problema.

O *Optimality Criteria Method* utiliza um método de ponto fixo, que trata os constrangimentos com uma função Lagrangeana do tipo:

$$L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = \varphi(\mathbf{x}) + \sum_{j=1}^{n_d} \lambda_j g_j(\mathbf{x}) + \sum_{e=1}^{n_i} \mu_e h_e(\mathbf{x}), \lambda \geq 0, \mu \in \mathbb{R}, \quad (2.7)$$

na qual $\varphi(\mathbf{x})$ é a função uniobjetivo resultante da escalarização, $h_e(\mathbf{x}), e = 1, \dots, n_i$ estão associadas a restrições de igualdade do tipo $h_e(\mathbf{x}) = 0$ e $g_j(\mathbf{x}), j = 1, \dots, n_d$ estão associadas a constrangimentos de desigualdade do tipo $g_j(\mathbf{x}) \leq 0$. Os multiplicadores de Lagrange, λ_j e μ_e , associados a cada tipo de restrição, podem ser interpretados como a força requerida para que as mesmas não sejam violadas. No *Optimality Criteria Method*, o método da bissecção [7] é utilizado na atualização iterativa dos multiplicadores de Lagrange, por forma a satisfazer os constrangimentos impostos. O que torna o OC mais interessante é o facto de ser um método particularmente adaptado à otimização estrutural: problemas deste tipo tendem a envolver restrições simples, como a restrição de volume máximo que será mais tarde abordada, constrangimentos estes que são tratados de forma direta e eficiente utilizando o método da bissecção para ajustar iterativamente os multiplicadores de Lagrange. O OC é também particularmente eficiente em problemas de grande dimensão, tipicamente observados em otimização estrutural, em particular na otimização topológica.

O segundo método baseado no gradiente considerado na presente dissertação para resolver o problema resultante da escalarização trata-se do *Method of Moving Assymptotes* (MMA). O MMA baseia-se na aproximação iterativa da função objetivo e dos constrangimentos por funções convexas, fazendo uso de assíntotas móveis, que funcionam como limites ajustáveis, permitindo o controlo da região de pesquisa, evitando grandes distâncias entre iteradas sucessivas. Dado um problema geral do tipo:

$$\min_{\mathbf{x} \in \mathbb{R}^n} \varphi(\mathbf{x}), \quad s.a \quad g_j(\mathbf{x}) \leq 0, \mathbf{x} \in [\mathbf{x}_{min}, \mathbf{x}_{max}], j = 1, \dots, n_d \quad (2.8)$$

o MMA aproxima a função objetivo $\varphi(\mathbf{x})$ na vizinhança da iterada atual, x_k , utilizando funções convexas racionais da forma:

$$\varphi(\mathbf{x}) \approx \sum_j \left(\frac{a_j}{x_j - L_j} + \frac{b_j}{U_j - x_j} + c_j \right), \quad (2.9)$$

onde L_j e U_j são as assintotas móveis e a_j, b_j e c_j são coeficientes calculados a partir do valor da função e do gradiente da função na iteração atual [3].

2.2 Abordagem de Pareto

Na abordagem de Pareto não é estabelecida qualquer relação de ordem entre as diferentes componentes da função objetivo, nem há qualquer agregação destas componentes. Como

tal, o conceito de “pesos” não é utilizado, pretendendo-se obter uma aproximação de toda a frente de Pareto do problema, incluindo as regiões não convexas, adiando a escolha da solução particular a ser adotada pelo agente de decisão para uma fase posterior à otimização. Esta abordagem providencia como solução uma lista de pontos não dominados, recorrendo à definição de ótimo de Pareto anteriormente introduzida. Qualquer ponto que não a satisfaça é considerado dominado e retirado da lista final de pontos, resultando numa aproximação da frente de Pareto que evidencia os *trade-offs* entre as diferentes componentes da função objetivo, permitindo ao agente de decisão retirar as soluções que melhor satisfaçam as suas preferências específicas. Os algoritmos de otimização que se baseiam nesta abordagem aproximam iterativamente a totalidade da frente de Pareto, um procedimento distinto da abordagem de escalarização, em que a frente de Pareto é aproximada ponto a ponto.

Algoritmos de otimização que recorrem à abordagem de Pareto- DMS e BoostDMS

O primeiro método de otimização estudado nesta dissertação que não é tradicionalmente utilizado na resolução de problemas de otimização topológica é o *Direct Multisearch* (DMS) [8]. Contrariamente aos algoritmos mencionados no Subcapítulo 2.1, este método não é baseado no gradiente, sendo tradicionalmente usado em problemas de otimização multiobjetivo sem derivadas, embora também se tenha mostrado competitivo em problemas multiobjetivo com derivadas [9].

Sendo um método que foi desenvolvido para problemas onde a avaliação da função objetivo é dispendiosa, os constrangimentos são abordados recorrendo ao conceito da barreira extrema: quando um ponto viola uma das restrições, não é avaliado na função objetivo, atribuindo-se o valor de $+\infty$ a cada uma das suas componentes. O algoritmo mantém uma lista de pontos admissíveis e não dominados, que é atualizada em todas as iterações. Cada iteração é organizada num passo de procura (*search step*) e num passo de sondagem (*poll step*). Em cada uma das iterações é escolhido um ponto da lista como centro da iteração, que será explorado em ambos os passos através da geração de novos pontos numa grelha que é implicitamente considerada pelo algoritmo.

O passo de procura tem muita flexibilidade, permitindo implementar uma qualquer estratégia, desde que gere um número finito de pontos na grelha. O passo de sondagem é responsável pelas propriedades teóricas de convergência do algoritmo e consiste numa procura local em torno do centro de iteração escolhido. Essa procura local baseia-se na avaliação de pontos correspondentes a direções com boas propriedades geométricas, escalonadas por um comprimento de passo. A boa geometria do conjunto de direções consiste na existência de uma direção de descida entre os seus vetores. Bases positivas [10] traduzem matematicamente esta propriedade, sendo as direções coordenadas $D = D_{\oplus} = [I - I]$, onde I representa a matriz identidade de ordem n , um exemplo.

A Figura 2.1 ilustra o passo de sondagem do algoritmo DMS num problema biobjetivo.

O ponto x_k é o centro a ser explorado localmente, testando as quatro direções representadas a preto. Como resultado, são gerados três novos pontos (dois representados a verde e um a vermelho). Uma das direções corresponde a um ponto que não faz parte da região admissível, pelo que o ponto correspondente não é avaliado na função objetivo. Após o processo de avaliação, a lista de pontos admissíveis não dominados é atualizada, sendo removidos os pontos dominados (representados a vermelho). Os pontos não dominados (representados a verde) constituem a aproximação corrente da frente de Pareto. No final de cada iteração, o comprimento de passo, diretamente associado ao comprimento das direções a serem testadas no passo de sondagem, é atualizado. Em iterações bem sucedidas, em que pelo menos um novo ponto não dominado é adicionado à lista, o comprimento de passo é mantido ou duplicado. Nas restantes iterações sem sucesso, o comprimento de passo é obrigatoriamente reduzido, tipicamente passando o seu valor a metade.

No que respeita ao passo de procura, este é opcional, sendo a sua inclusão aquilo que diferencia o BoostDMS [11] do DMS. O passo de procura implementado no BoostDMS consiste no cálculo de modelos locais de interpolação polinomial quadrática do tipo $\bar{m}_i(x) = c_i + \mathbf{g}_i^T x + \frac{1}{2} x^T \mathbf{H}_i x$, com $i \in \{1, \dots, m\}$, para cada uma das componentes da função objetivo. Estes modelos são posteriormente minimizados localmente, individualmente ou em conjunto, gerando novos pontos que são avaliados na verdadeira função objetivo. Uma das vantagens do DMS e do BoostDMS consiste na capacidade de aproximar toda a frente de Pareto, objetivo que esteve sempre subjacente à definição da respetiva estrutura algorítmica. Contudo, esta classe de métodos apresenta limitações quando se pretende resolver problemas de otimização com um número elevado de variáveis.

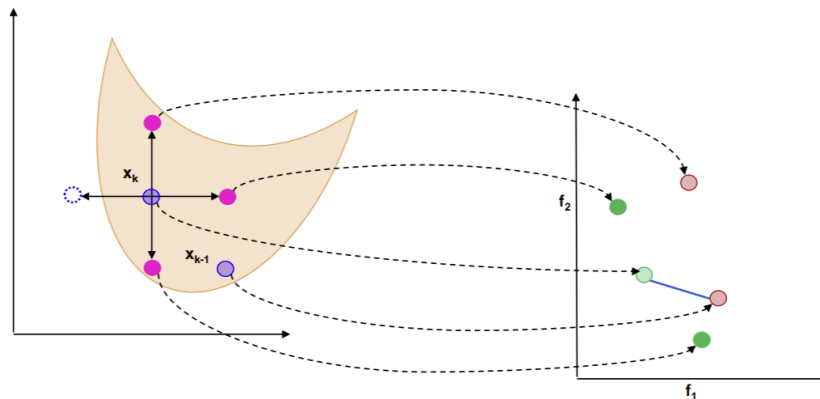


Figura 2.1: Passo de sondagem no algoritmo DMS realizado a partir do centro x_k , a resultar na geração de novos pontos não dominados (a verde escuro), num problema biobjetivo.

OTIMIZAÇÃO TOPOLÓGICA

A otimização estrutural é um processo iterativo em que, partindo de uma aproximação inicial à solução do problema, é feita uma análise do sistema para este valor e gerada uma melhor aproximação. Após sucessivas iterações, se tiver sido gerada uma aproximação que não viola os constrangimentos do problema e que satisfaz os critérios de convergência, então essa aproximação é considerada como a solução ótima e o processo de otimização termina. A área de otimização estrutural pode ser dividida em três subcategorias: 1) otimização de dimensões, na qual as variáveis de projeto são as dimensões geométricas da estrutura; 2) otimização de forma, em que as variáveis de projeto são as coordenadas dos pontos pertencentes à fronteira do domínio ocupado pela estrutura; 3) otimização de topologia. A presente dissertação incide apenas na otimização topológica.

A otimização topológica visa encontrar a distribuição ideal de material num dado domínio de projeto, que minimiza um objetivo definido, tendo em conta as condições de fronteira aplicadas sobre esse domínio, assim como os constrangimentos de projeto. Trata-se de um procedimento que distribui o material disponível numa área ou volume fixos, através da remoção gradual de pequenas porções de material nas quais as tensões aplicadas são muito baixas, sendo retirado o material que não é estritamente necessário na transmissão de esforços internos. Na otimização topológica, um problema é representado por um conjunto de variáveis booleanas que exprimem a existência ou ausência de material, sendo que o valor 1 representa a presença de material num dado ponto, enquanto 0 representa vazio.

3.1 Formulação do problema multicarga num domínio contínuo

No contexto desta dissertação, foi resolvido um problema de otimização topológica quer no domínio contínuo, quer no discreto, de modo a concluir que tipo de formulação seria mais adequada para testar algoritmos pouco utilizados nesta classe de problemas.

De modo a contextualizar a formulação em domínio contínuo, consideremos o problema de elasticidade exemplificado na Figura 3.1. Nesta abordagem é resolvido um

problema contínuo, sendo estabelecida uma relação entre o campo de tensão e a densidade relativa do material. Inicialmente, o domínio é discretizado numa malha de elementos finitos, em que toda a área de material apresenta densidade relativa uniforme igual a 1. Subsequentemente, os valores de densidade relativa em cada elemento da malha vão sendo variados entre 0 e 1, resultando eventualmente numa estrutura na qual a quantidade de material é a limitada, satisfazendo ainda os constrangimentos do projeto. Na Figura 3.1 este processo pode ser observado em termos gráficos, sendo que as zonas pretas representam a existência de material (valor de densidade relativa próximo de 1), zonas cinzentas correspondem a zonas de possível contorno da estrutura (valores entre 0 e 1) e zonas claras representam vazio de material (valores próximos de 0).

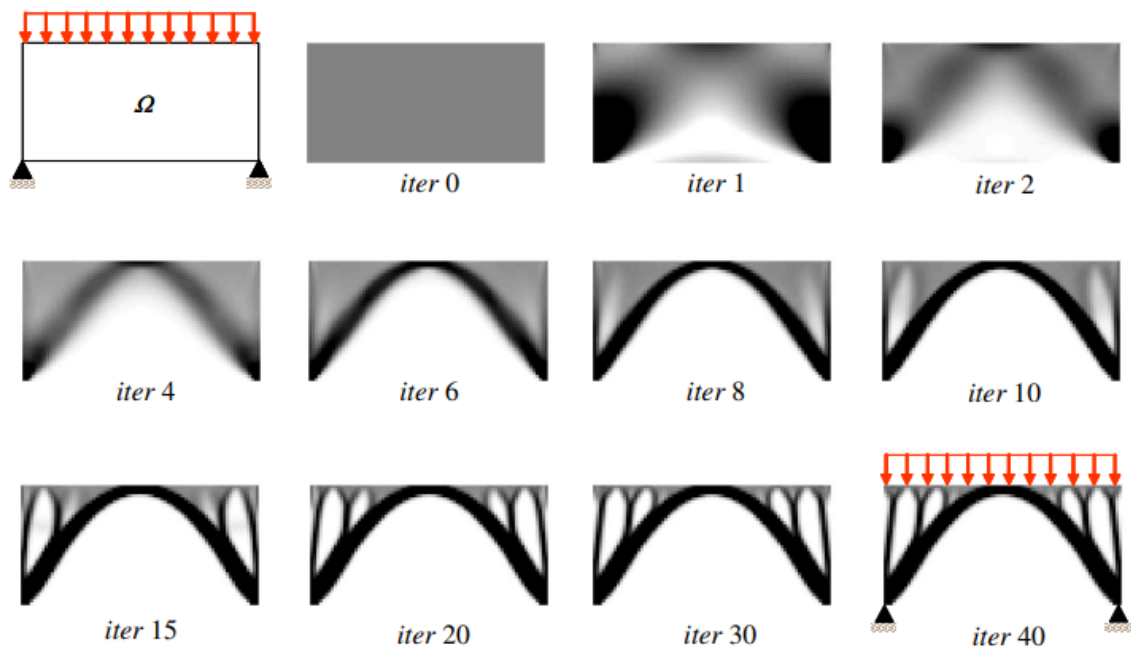


Figura 3.1: Exemplo do processo de otimização topológica num problema de elasticidade.

Passamos então à descrição do problema de otimização topológica que se procurou resolver em domínio contínuo. Foi considerado um domínio de projeto retangular, Ω , impondo-se condições de deslocamento em toda a aresta correspondente à extremidade esquerda do domínio (encastramento local). Considerou-se a aplicação de duas forças, F_1 e F_2 , na extremidade oposta. Estas forças não são aplicadas simultaneamente mas alternadamente, em condições de serviço. Estamos então perante um problema multicarga que, segundo [4], pode ser interpretado como um problema multiobjetivo. A Figura 3.2 esquematiza o problema descrito.

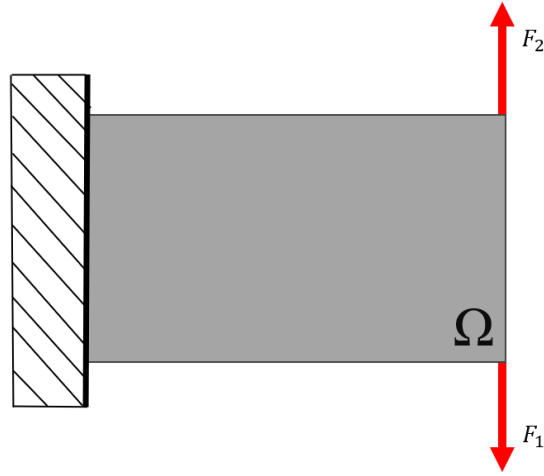


Figura 3.2: Domínio de projeto Ω e representação do esquema de forças considerado.

De forma simplificada, para o exemplo considerado e na abordagem contínua, o processo de resolução pode ser descrito nos seguintes passos. O domínio Ω é discretizado em elementos finitos, utilizando uma malha perfeitamente regular, com grelha quadrada. Como já foi mencionado, ao resolver um problema de otimização topológica, o objetivo é obter a melhor distribuição do material no domínio considerado, satisfazendo os constrangimentos existentes. Neste caso, procura-se a distribuição ótima de material capaz de suportar os dois cenários de carga, F_1 e F_2 , e que simultaneamente apresenta o menor valor de flexibilidade (equivalente ao maior valor de rigidez).

O problema pode então ser formulado como um problema biobjetivo com restrições:

$$\begin{aligned} \min_{\rho} \quad & (\phi_1, \phi_2) \\ \text{s.a} \quad & V \leq V^* \\ & \mathbf{F}_i = \mathbf{K} \cdot \mathbf{u}_i, i = 1, 2, \end{aligned} \quad (3.1)$$

onde $V \leq V^*$ representa uma restrição de volume máximo, que indica que não se pode preencher mais do que uma certa percentagem de Ω com material. Já ρ representa a variável de projeto, correspondente à densidade relativa de cada um dos elementos da malha. Esta variável de projeto define o problema no domínio contínuo, sendo que, como já foi mencionado, um valor de 1 representa presença de material num dado elemento finito (ou "pixel") do domínio Ω discretizado (ou "pixelizado"), enquanto 0 representa vazio. Num problema resolvido num domínio contínuo, o número de variáveis de projeto será igual ao número de elementos considerados na malha de elementos finitos.

Cada componente da função objetivo, ϕ_i , corresponde à energia de deformação elástica, resultante da ação de cada uma das forças, dada por:

$$\phi_i = \mathbf{u}_i^T \cdot \mathbf{K} \cdot \mathbf{u}_i \quad i = 1, 2,$$

onde \mathbf{u}_i corresponde ao vetor de deslocamento resultante da aplicação da força \mathbf{F}_i e \mathbf{K} representa a matriz de rigidez global associada ao domínio. O campo de deslocamentos

referente a cada uma das forças é a solução da equação de equilíbrio do problema de elasticidade, que na forma discretizada se escreve como $\mathbf{F}_i = \mathbf{K} \cdot \mathbf{u}_i$.

Utilizando o método da soma ponderada, o problema biobjetivo pode ser transformado num problema uniobjetivo, com pesos w e $1-w$, e $w \geq 0$:

$$\begin{aligned} \min_{\rho} \quad & w\phi_1 + (1-w)\phi_2 \\ \text{s.a} \quad & V \leq V^* \\ & \mathbf{F}_i = \mathbf{K} \cdot \mathbf{u}_i, i = 1, 2 \end{aligned} \tag{3.2}$$

Aqui w representa o peso atribuído a um caso de carga ou à *compliance* associada, sendo o peso $(1-w)$ atribuído ao outro caso de carga, como mencionado no Subcapítulo 2.1. No domínio contínuo, a matriz de rigidez é calculada com base na densidade relativa, ρ_e , de cada um dos elementos, através de:

$$\mathbf{K}_e = \rho_e^p \cdot \mathbf{K}_e^o, \tag{3.3}$$

onde p é um parâmetro de penalização e \mathbf{K}_e^o representa a matriz de rigidez de um elemento com $\rho = 1$. A matriz de rigidez global, \mathbf{K} , é obtida assemblingando as contribuições de todas as matrizes de rigidez, \mathbf{K}_e , dos elementos da malha.

3.2 Formulação do problema multicarga num domínio discreto

Considerando agora o caso de um domínio discreto, contrariamente ao que acontece no contínuo, a variável de densidade relativa pode apenas assumir valores binários, 0 ou 1. Tal implica a inexistência de zonas “cinzentas”, mencionadas no Subcapítulo 3.1, uma vez que estas correspondem a valores intermédios de densidade relativa. A Figura 3.3 representa um exemplo de um problema de otimização topológica formulado num domínio discreto, que será posteriormente analisado em detalhe no decurso desta dissertação.

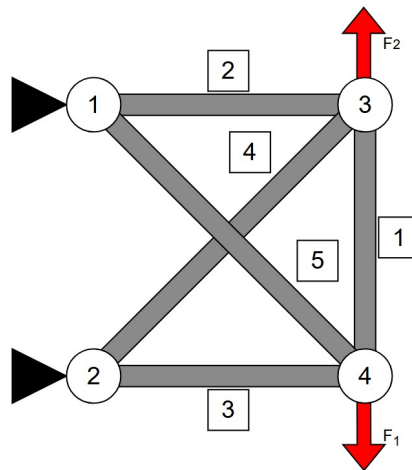


Figura 3.3: Problema de otimização topológica formulado num domínio discreto.

3.2. FORMULAÇÃO DO PROBLEMA MULTICARGA NUM DOMÍNIO DISCRETO

Trata-se do problema já descrito no Subcapítulo 3.1, embora com uma formulação distinta, expressa da seguinte forma:

$$\begin{aligned} \min_{A_e} \quad & (\phi_1, \phi_2) \\ \text{s.a} \quad & V \leq V^* \\ & \mathbf{F}_i = \mathbf{K} \cdot \mathbf{u}_i, i = 1, 2, \end{aligned} \tag{3.4}$$

mudando apenas as variáveis de projeto. Aqui, em vez da variável de projeto representar a densidade relativa individual de cada um dos elementos da malha, agora denota a área da secção transversal individual de cada um dos elementos barra. Assim, o número de variáveis de projeto é constante e igual a 5, traduzindo-se numa redução significativa da dimensão do problema, por comparação com a formulação em domínio contínuo. O cálculo da matriz de rigidez associada a cada um dos elementos barra, \mathbf{K}_e , é agora feito da seguinte forma:

$$\mathbf{K}_e = \frac{A_e \cdot E}{L_e} \cdot \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}, \tag{3.5}$$

onde A_e corresponde à variável de projeto, representando a área individual de cada elemento barra, L_e corresponde ao comprimento de cada elemento barra e E corresponde ao módulo de elasticidade longitudinal (ou módulo de Young).

SIMULAÇÃO NUMÉRICA

4.1 Problemas acadêmicos

Como já foi mencionado, esta dissertação tem como objetivo a análise do desempenho numérico e limitações associados a diferentes algoritmos quando aplicados à resolução de um problema de otimização topológica biobjetivo. Numa primeira fase, foi resolvido então um conjunto de problemas biobjetivo acadêmicos, com o objetivo de comparar os algoritmos DMS e BoostDMS, analisando as suas potenciais limitações em problemas de otimização topológica. O objetivo será a aproximação da totalidade da frente de Pareto de cada problema.

No Subcapítulo 2.2 referiu-se que uma das limitações dos algoritmos DMS e BoostDMS consiste na dificuldade de tratar problemas com um grande número de variáveis, que frequentemente estão associados à otimização topológica. O presente capítulo dedica-se a este estudo. Para tal, consideraram-se versões biobjetivo de problemas diferenciáveis e que permitem uma dimensão variável, descritos em [12–15], num total de nove. Os problemas selecionados foram então os seguintes: DTLZ2 [13]; DTLZ3 [13]; DTLZ4 [13]; DTLZ6 [13]; Jin1 [15]; BK1 [12]; MOP2 [12]; ZDT2 [14]; ZLT1 [12].

Os algoritmos DMS e BoostDMS foram utilizados na resolução dos referidos problemas, com dimensões 10, 20, 40 e 60. Foram considerados todos os valores de defeito dos dois algoritmos, sendo o critério de paragem baseado num valor de comprimento de passo mínimo para todos os pontos da lista ou num número máximo de cálculos para a função objetivo ($\alpha = 10^{-3}$ e 20000 cálculos, respetivamente).

Neste subcapítulo será usado como referência o problema DTLZ2, podendo os resultados para os restantes problemas ser encontrados no Anexo A. A equação (4.1) traduz a formulação matemática correspondente ao problema DTLZ2, para uma função objetivo com duas componentes ($m=2$).

$$\begin{aligned}
 \min_{\mathbf{x}} \quad & F(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x})), \text{ com} \\
 & f_1(\mathbf{x}) = (1 + g) \cos\left(x_1 \frac{\pi}{2}\right), \\
 & f_2(\mathbf{x}) = (1 + g) \sin\left(x_1 \frac{\pi}{2}\right), \\
 \text{s.a.} \quad & 0 \leq x_i \leq 1, \text{ com } i = 1, 2, \dots, n, \text{ e} \\
 & g = \sum_{i=2}^n (x_i - 0.5)^2
 \end{aligned} \tag{4.1}$$

Os resultados obtidos com os algoritmos DMS e BoostDMS para a versão biobjetivo do problema DTLZ2 encontram-se na Figura 4.1.

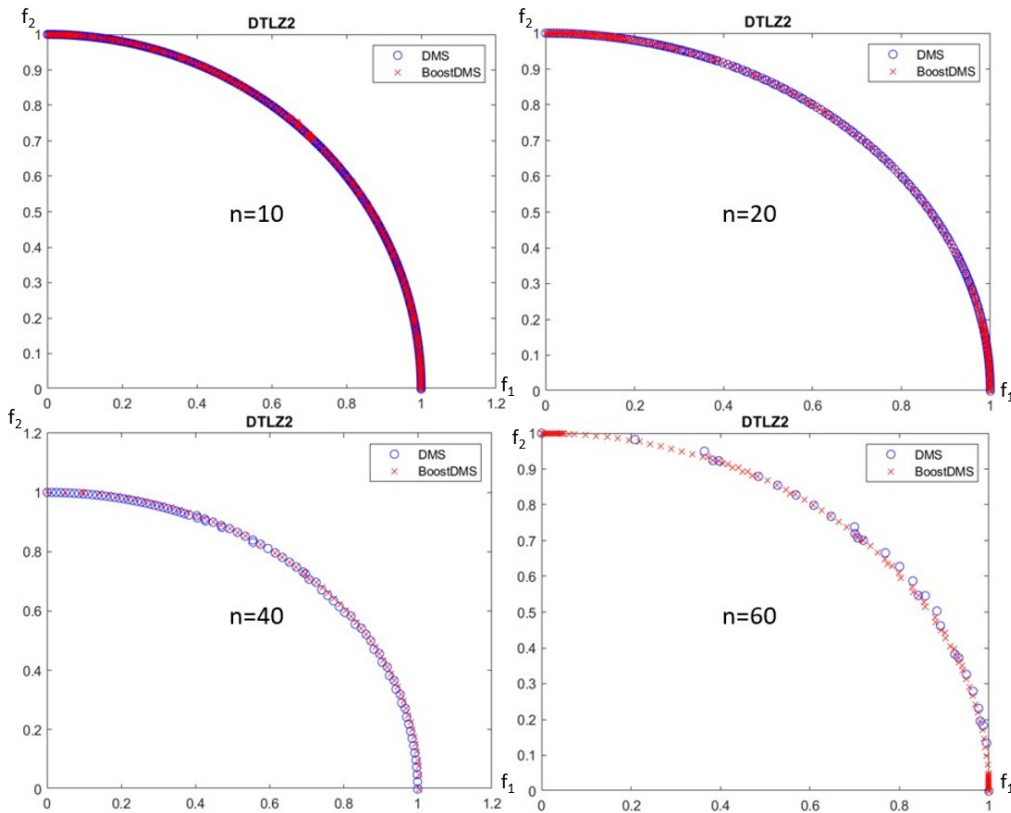


Figura 4.1: Aproximações da frente de Pareto obtidas pelos algoritmos DMS e BoostDMS, para o problema DTLZ2, com $n = 10, 20, 40, 60$.

Como é possível verificar, a diferença entre as duas variantes do DMS não é significativa para dimensões pequenas (10 ou 20 variáveis). Contudo, a vantagem de utilizar o BoostDMS torna-se mais evidente à medida que a dimensão do problema aumenta. O BoostDMS é consistentemente capaz de detetar um número muito maior de pontos na aproximação da frente de Pareto do que o DMS. Para além disto, a distribuição de pontos obtida pelo BoostDMS é muito mais homogénea. No caso do DMS, parece haver uma tendência para a formação de *clusters* de pontos, quando a dimensão do problema é elevada. Isto é particularmente visível no caso de $n = 60$. Análises análogas podem ser efetuadas para a generalidade dos restantes problemas (ver Anexo A).

Conclui-se então que a definição de um passo de procura baseado na minimização de

modelos polinomiais quadráticos, que distingue o BoostDMS do DMS, é particularmente relevante para problemas de dimensão elevada, potencialmente tornando este algoritmo mais interessante no que respeita à resolução de problemas de otimização topológica, que tendem a apresentar estas características. Contudo, é de salientar que a adição deste passo de procura torna esta variante do código bastante mais dispendiosa em termos de tempo computacional, resultado dos sistemas de equações lineares associados ao cálculo dos coeficientes dos modelos.

Os resultados obtidos pelos dois algoritmos no conjunto dos nove problemas foram ainda analisados em conjunto, utilizando perfis de desempenho [16] e quatro métricas comumente consideradas pela comunidade de otimização multiobjetivo. Num perfil de desempenho, quanto mais elevada é a curva correspondente a um algoritmo melhor, é o desempenho numérico que lhe está associado. É ainda particularmente importante a análise para o valor $\tau = 1$, que representa a eficiência do algoritmo, e a análise para valores elevados de τ , que corresponde à sua robustez ou eficácia.

As quatro métricas consideradas foram a pureza [17], que mede a percentagem de pontos gerada pelo algoritmo na frente de Pareto; o hipervolume [18], que mede o volume da região dominada pela aproximação da frente de Pareto calculada pelo algoritmo, considerando um ponto de referência dominado por todos os pontos gerados por todos os algoritmos para o problema em análise; e as métricas de dispersão Γ e Δ [8]. A métrica Γ mede o maior espaço vazio entre dois pontos consecutivos na frente de Pareto. A métrica Δ mede a uniformidade destes espaços vazios. As Figuras 4.2-4.5, representam os perfis de desempenho obtidos para cada uma das métricas consideradas, para o conjunto dos nove problemas, considerando 10, 20, 40 e 60 variáveis, respetivamente. Nestes perfis o eixo das abcissas, τ , representa um fator de desempenho relativo ao melhor desempenho observado entre os algoritmos para cada problema. São considerados apenas valores de τ superiores ou iguais a 1, sendo que $\tau = 1$ corresponde ao melhor desempenho para os problemas considerados. As ordenadas, ρ , indicam a percentagem de problemas para os quais o desempenho do algoritmo está dentro de um fator τ do melhor desempenho, uma vez que se trata de uma percentagem, o valor de ρ varia entre 0 e 1.

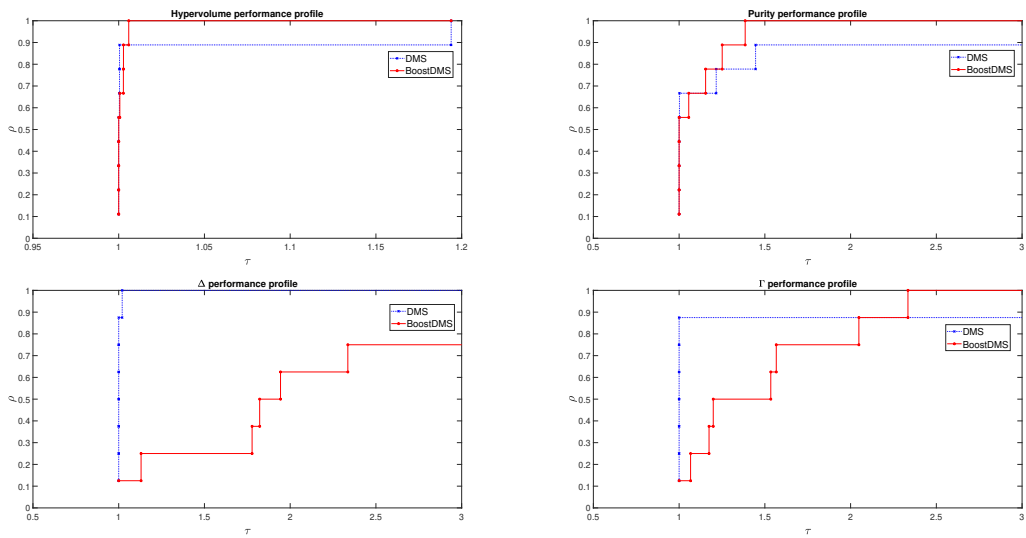


Figura 4.2: Perfis de desempenho para a coleção de problemas com dimensão $n = 10$.

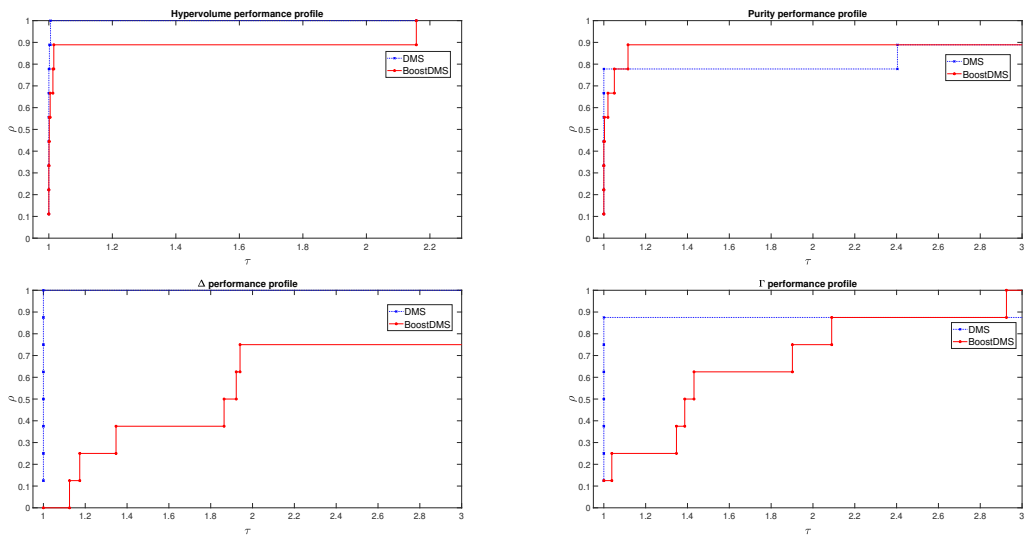


Figura 4.3: Perfis de desempenho para a coleção de problemas com dimensão $n = 20$.

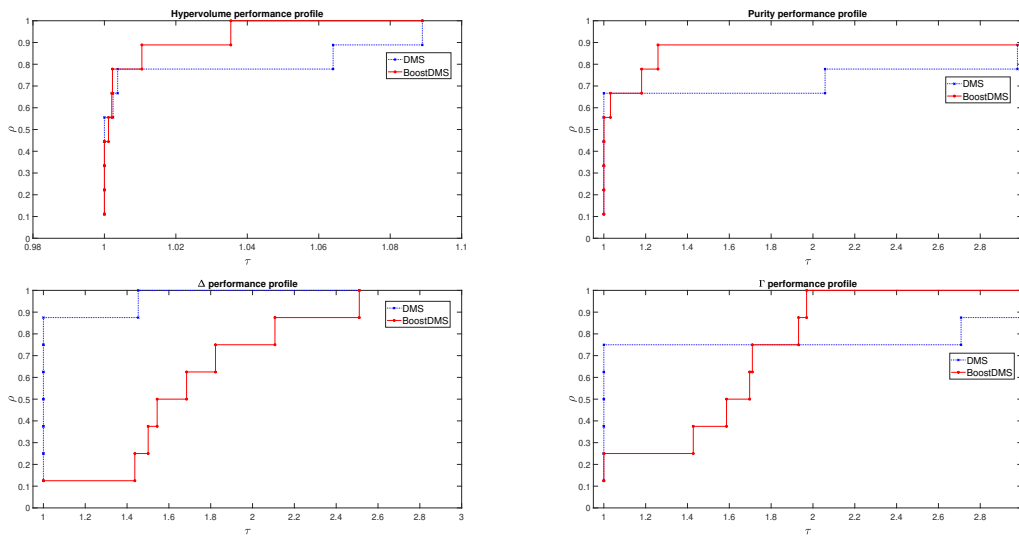


Figura 4.4: Perfis de desempenho para a coleção de problemas com dimensão $n = 40$.

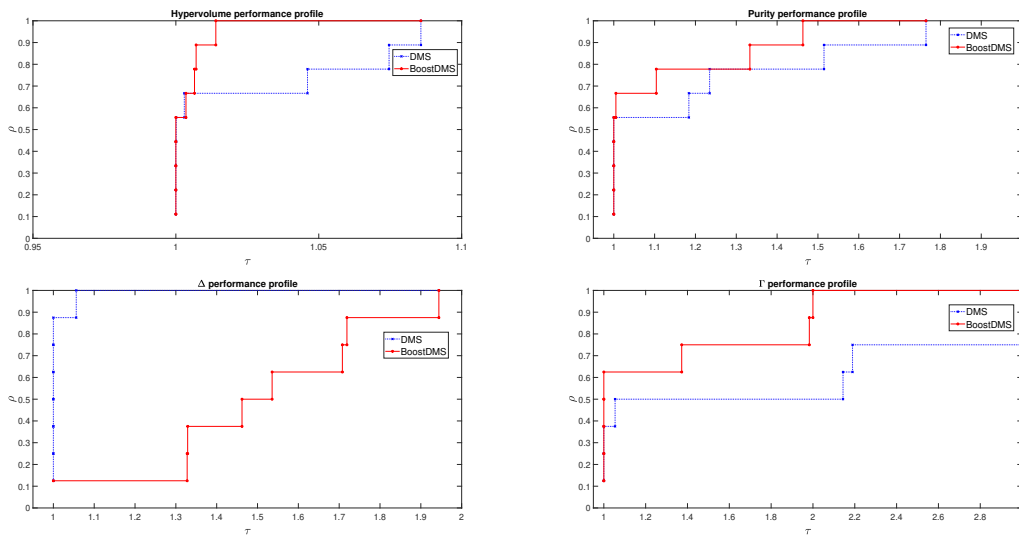


Figura 4.5: Perfis de desempenho para a coleção de problemas com dimensão $n = 60$.

Como é possível observar pelas figuras acima, conclui-se que o BoostDMS, em geral, apresenta um melhor desempenho do que o DMS para as métricas consideradas. A exceção ocorre na dispersão, independentemente do número de variáveis, e no hipervolume para $n = 20$.

4.2 Análise no contexto de otimização estrutural

4.2.1 Resolução de um problema de otimização topológica no contínuo

O primeiro problema resolvido foi então o problema formulado no Subcapítulo 3.1. Numa primeira fase do estudo do problema de otimização no domínio contínuo, procurou-se analisar diferentes resultados obtidos pelo OC. A bibliografia existente tende a abordar este tipo de problemas recorrendo a métodos de reformulação matemática, como aqueles mencionados no Capítulo 2. Os trabalhos realizados por Li et al. [19] e Sato et al. [20] utilizam o método da soma ponderada para transformar o problema multiobjetivo em uniobjetivo. Ambos os trabalhos mostraram dificuldades em obter pontos em certas zonas da fronteira de Pareto, o que justifica a exploração de outras alternativas para a resolução do problema biobjetivo. Os resultados esperados para este problema deverão ser semelhantes aos observados na Figura 4.6, onde se ilustra as diferentes distribuições de material em função do peso atribuído a cada uma das componentes da função objetivo.

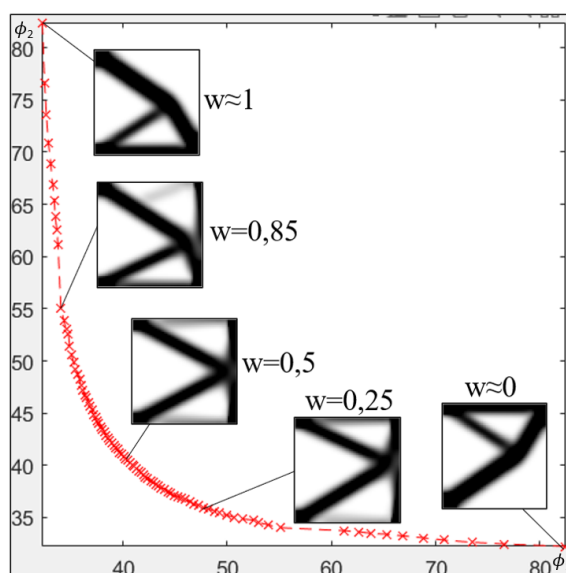


Figura 4.6: Frente de Pareto esperada para um domínio de dimensões 60x60, obtida pelo OC.

A primeira estrutura será ideal quando está aplicada apenas F_1 e a última ótima apenas para a aplicação de F_2 . Consequentemente, espera-se que quando w assume valores intermédios ($0 < w < 1$) a estrutura ótima seja uma solução com características de ambos os casos de carga. Na verdade, neste tipo de problema não se permite aos pesos atingirem os valores extremos de 1 e 0, uma vez que tal corresponde a uma das *compliances* ser teoricamente infinita (na prática, com um valor comparativamente muito elevado). Como se pode observar, este problema é simétrico, uma vez que F_1 e F_2 são consideradas iguais em módulo. Como tal, espera-se uma frente de Pareto simétrica em relação à reta $\phi_1 = \phi_2$, como se observa na Figura 4.6.

Procedeu-se à resolução deste problema de otimização topológica utilizando os códigos

educacionais *top88* e *top99* [21, 22], códigos que recorrem à utilização de filtros, ou de sensibilidades, ou de densidades [4].

A sensibilidade consiste na derivada parcial da função objetivo com respeito à variável de projeto. Num caso como o presente, em que a densidade corresponde à variável de projeto, a sensibilidade mede então o efeito local de mudanças na densidade sobre o processo de otimização. A filtragem de sensibilidades previne variações bruscas entre as sensibilidades dos vários elementos da malha, evitando certas soluções correspondentes a padrões artificiais, como soluções de *checkerboard*, vistas tipicamente neste tipo de problema, garantido distribuições mais físicas e robustas. O procedimento passa por, num dado elemento da malha, em vez de se utilizar diretamente a sensibilidade calculada, utiliza-se um valor filtrado, calculado com uma média ponderada das sensibilidades dos elementos vizinhos. O filtro de densidades atua do mesmo modo. No entanto, em vez de atuar sobre os valores de sensibilidade, como o nome indica, este filtro atua diretamente sobre os valores de densidade de cada elemento, alterando-o com base nas densidades dos valores vizinhos.

Ambos os códigos recorrem ao *Optimality Criteria Method* (OC). O *top88* é um código comparativamente muito mais eficiente (rápido) e que para além do filtro de sensibilidades também implementa o filtro de densidades. O problema foi primeiramente resolvido considerando uma dimensão de malha de 10×10 , o que equivale a 100 variáveis de projeto, e depois considerando uma dimensão de malha de 30×30 , ou seja considerando 900 variáveis. Os resultados obtidos podem ser observados na Figura 4.7 e na Tabela 4.1.

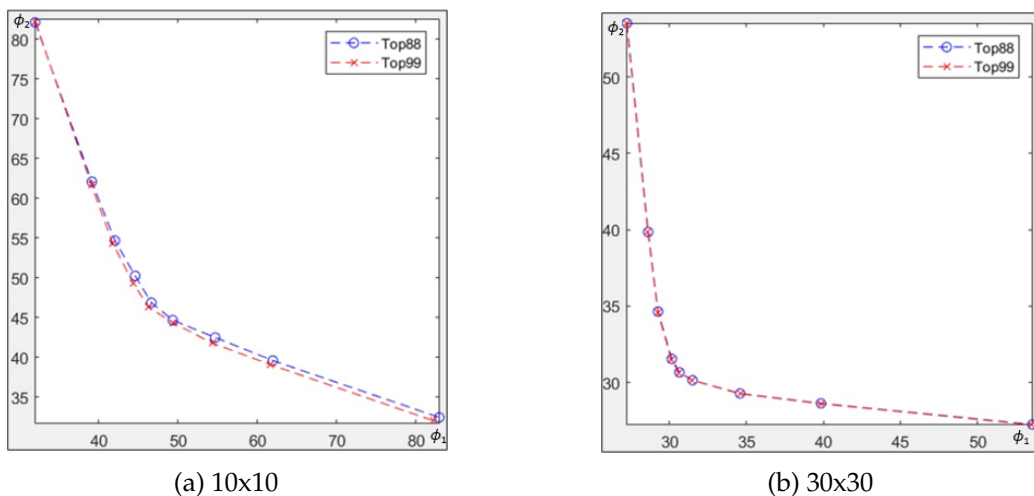


Figura 4.7: Aproximações da frente de Pareto obtidas com o *top88* e o *top99* para malhas de diferentes dimensões.

Malha de dimensões 10x10				Malha de dimensões 30x30			
top88		top99		top88		top99	
ϕ_1	ϕ_2	ϕ_1	ϕ_2	ϕ_1	ϕ_2	ϕ_1	ϕ_2
31.9772	82.1254	32.0065	82.2188	27.2442	53.5369	27.2279	53.5177
39.1990	62.0251	39.0683	61.6663	28.6258	39.8544	28.6074	39.8569
42.0471	54.6733	41.7747	54.3109	29.267	34.6454	29.2587	34.5932
44.6007	50.2412	44.3337	49.3658	30.1609	31.5527	30.1411	31.4984
46.6373	46.8693	46.2853	46.2853	30.6598	30.6687	30.642	30.642
49.3329	44.6559	49.3658	44.3337	31.5112	30.1495	31.4984	30.1411
54.6815	42.4766	54.3109	41.7747	34.5906	29.2817	34.5932	29.2587
61.9593	39.5888	61.6663	39.0683	39.8447	28.6296	39.8569	28.6074
82.9447	32.4264	82.2188	32.0065	53.5652	27.649	53.5177	27.2279

Tabela 4.1: Coordenadas dos pontos de Pareto obtidos para malhas de dimensões 10x10 e 30x30, utilizando o *top88* e o *top99*.

Nestes testes, quer o *top88* quer o *top99* foram corridos com o filtro de sensibilidades. No teste de malha de dimensões 10x10, seria expeável obter duas aproximações da frente de Pareto coincidentes. Como é possível observar, quer pelo gráfico, quer pelas colunas correspondentes da tabela de coordenadas, a aproximação da fronteira de Pareto obtida pelo *top99* é perfeitamente simétrica segundo a bisettriz o que seria de esperar para este tipo de problema. O mesmo já não acontece no caso do *top88*.

Em primeiro lugar, segundo o autor original destes códigos, sendo o *top88* e o *top99* implementações distintas, nunca serão esperados resultados exatamente iguais para um dado problema. Ainda assim, tal não explica a falta de simetria observada na frente de Pareto obtida pelo *top88*. Após terem sido realizados vários testes, apesar de não ter sido possível apurar de onde vem esta inconsistência, foi possível chegar à conclusão de que a inconsistência se torna residual com o refinamento da malha.

Na Figura 4.7 (b), com uma malha mais refinada de dimensões 30x30, a fronteira de Pareto obtida pelo *top88* aproxima-se muito mais da determinada pelo *top99* e, consequentemente, do tipo de resultado esperado para este tipo de problema. Os gráficos obtidos pelas duas variantes do código nunca serão exatamente iguais devido à diferença inerente à codificação. Ainda assim, as diferenças são consideradas desprezáveis, principalmente quando se está a trabalhar com malhas de maiores dimensões. Se o tempo computacional não for uma preocupação, é recomendado utilizar o *top99*. Se a preocupação for o tempo de cálculo, o *top88* é preferível.

Finalmente, incidindo agora apenas sobre o *top88*, decidiu-se ainda apurar as diferenças existentes ao correr o código utilizando o filtro de sensibilidades e o filtro de densidades. O estudo foi mais uma vez realizado considerando malhas de dimensões 10x10 e 30x30 e os resultados podem ser observados na Figura 4.8.

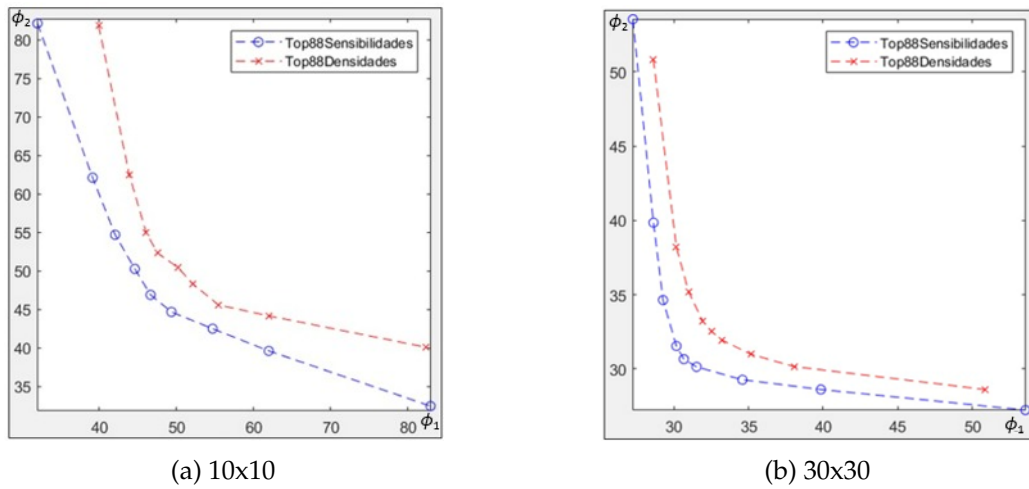


Figura 4.8: Aproximações da frente de Pareto obtidas pelo top88 para os dois tipos de filtro em malhas de diferentes dimensões.

Como se pode observar, as frentes de Pareto relativas ao filtro de sensibilidades dominam totalmente as relativas ao filtro de densidades, resultados estes que são apoiados pela bibliografia existente [23]. A aplicação do filtro de densidades resulta em valores de ϕ mais elevados, consequência deste filtro gerar mais densidades intermédias que o filtro de sensibilidades. A dominância do segundo filtro vem então do facto deste utilizar com mais frequência valores de densidade relativa iguais a 0 ou 1.

Tendo sido estudado o problema utilizando o OC, passou-se então à sua resolução também com o DMS, que não é tipicamente utilizado em otimização topológica. A escolha do DMS em detrimento do BoostDMS prende-se com os tempos computacionais associados. Contudo, uma vez que o DMS apresenta dificuldades ao resolver problemas com um elevado número de variáveis de projeto, foi tomada a decisão de se considerar uma malha de 10x10 elementos. Começou-se então por executar o OC para resolver o mesmo problema, com 100 valores de peso diferentes, resultando numa aproximação da frente de Pareto com 100 pontos. A Figura 4.9 mostra a aproximação obtida, assim como algumas das soluções correspondentes.

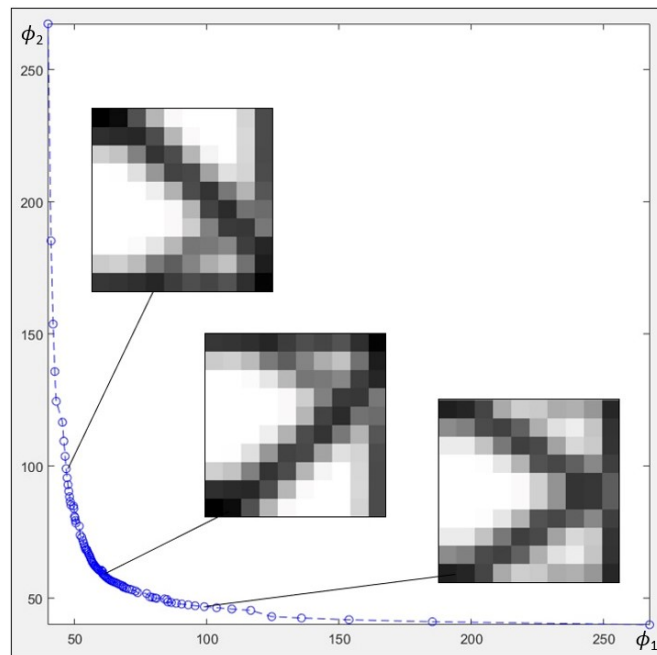


Figura 4.9: Aproximação da frente de Pareto obtida pelo OC, para uma malha de 10×10 , considerando 100 valores de peso distintos.

Os 230 segundos utilizados pelo OC para cálculo destes resultados foram utilizados como critério de paragem do DMS. A Figura 4.10 ilustra os resultados correspondentes.

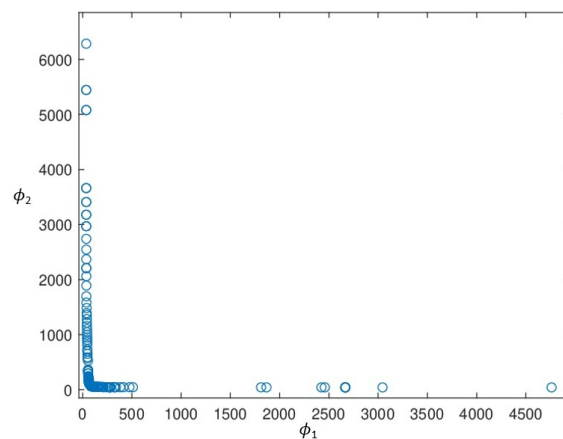


Figura 4.10: Aproximação da frente de Pareto obtida pelo DMS, permitindo os 230 segundos de tempo computacional utilizados pelo OC.

Como é possível observar, a maioria das soluções obtidas apresentam valores com uma ordem de grandeza muito superior às soluções obtidas com o *top88*. Este tipo de soluções não foram geradas pelo OC, correspondendo a valores de pesos muito próximos de 0 e 1, ou seja, a valores de compliance “infinita” no OC. Na Figura 4.11, o foco é colocado sobre a zona da frente de Pareto de maior interesse, sendo ainda possível observar algumas das estruturas correspondentes às soluções mais relevantes obtidas pelo DMS.

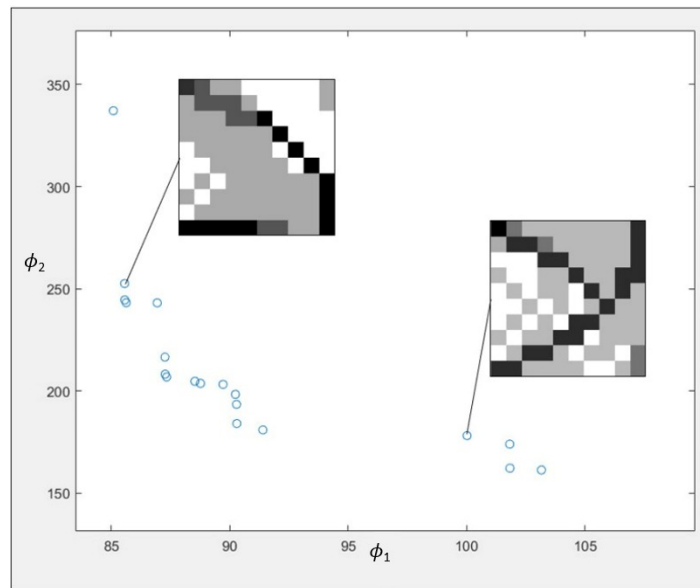


Figura 4.11: *Zoom-in* da aproximação da frente de Pareto e algumas estruturas obtidas pelo DMS para uma malha de 10x10, permitido 230 segundos de tempo computacional.

Quer a aproximação da frente de Pareto, quer as estruturas obtidas estão muito longe daquilo que se espera para este tipo de problema. Se usarmos como referência os valores da Figura 4.9, obtidos pelo OC, verifica-se que o DMS não foi capaz de obter qualquer solução nesse espaço.

Para a análise dos resultados, é ainda necessário ter em consideração o número total de iterações que o algoritmo precisaria de efetuar até atingir a convergência. Uma vez que o problema em questão terá $n = 100$ variáveis, o número de avaliações da função objetivo que o algoritmo fará será $f_{eval} = 200 \cdot Iter$, em que $Iter$ é o número de iterações. A Figura 4.12 ilustra a aproximação da frente de Pareto obtida pelo DMS, permitindo 100000 avaliações da função objetivo, correspondendo a 500 iterações, o que se traduziu num aumento significativo do tempo computacional associado. Para referência, o teste anterior correspondeu a aproximadamente 45000 avaliações da função objetivo. Foram necessários cerca de 18 minutos para concluir este teste.

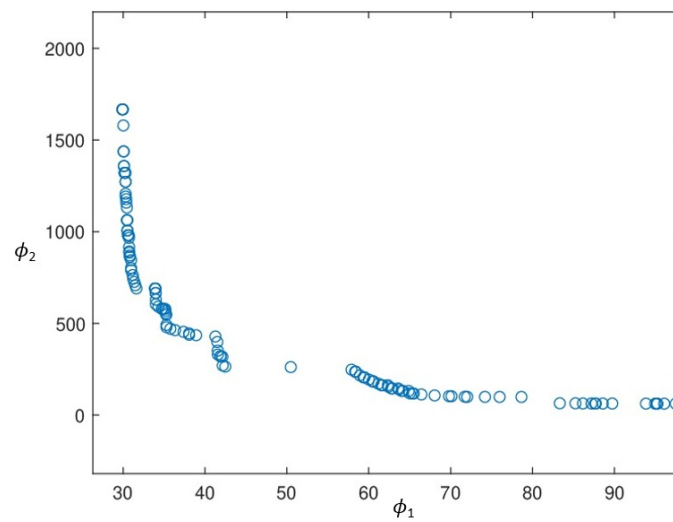


Figura 4.12: Aproximação da frente de Pareto obtida pelo DMS para uma malha de 10x10, permitindo 100000 avaliações da função objetivo.

Mais uma vez, a Figura 4.13 apresenta um *zoom-in* da frente de Pareto, desta vez sendo possível verificar a geração de pontos no espaço das soluções do OC, bem como algumas das estruturas obtidas:

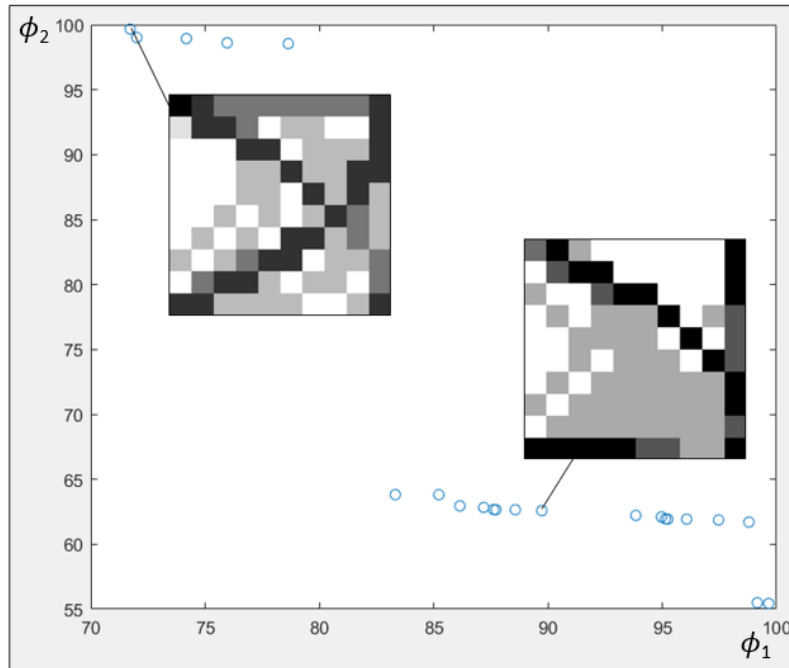


Figura 4.13: *Zoom-in* da aproximação da frente de Pareto e algumas estruturas obtidas pelo DMS para uma malha de 10x10, permitindo 100000 avaliações da função objetivo.

Verifica-se então que, permitindo um maior número de avaliações para a função objetivo, que possibilita que algumas das soluções geradas estejam mais próximas de satisfazer o critério de convergência, resulta num conjunto de soluções mais prometedoras. Aumentando gradualmente o valor de f_{eval} concluiu-se que, para a convergência ocorrer, seria necessária uma quantidade de tempo irrealista.

4.2.2 Resolução de um problema de otimização topológica no domínio discreto

A maior discrepância entre os resultados obtidos pelo OC e pelo DMS no domínio contínuo resulta do elevado número de variáveis do problema, que causa tempos computacionais proibitivos, associados à execução do DMS. Assim sendo, decidiu-se alterar o paradigma de otimização, passando ao estudo do problema no domínio discreto. Abordou-se então a otimização topológica utilizando uma “abordagem de estrutura base”, ou “*ground structure approach*” [4].

Nesta abordagem à otimização topológica começa-se por dividir o domínio de projeto num conjunto de nós. A estrutura base é depois formada conectando cada par de nós com barras, que definem os elementos estruturais a otimizar. Tipicamente, a estrutura inicial inclui como elementos todas as combinações possíveis entre nós, isto é, o número máximo

de barras possíveis. O objetivo do problema de otimização será concluir quais destes elementos serão verdadeiramente necessários para a minimização da função objetivo, de modo a encontrar a melhor distribuição de material possível, no domínio de projeto. As variáveis de projeto serão então as áreas das secções transversais das barras, o que aproxima este método de otimização topológica da otimização dimensional. Contudo, a existência prévia da *ground structure* distingue as duas abordagens, permitindo a remoção gradual de elementos de barra à medida que eles se mostram desnecessários para a minimização da função objetivo. O processo de otimização resulta tipicamente numa estrutura final com um menor número de elementos do que a estrutura base. Na otimização dimensional, a topologia é fixa e, como tal, o número de elementos nunca pode variar, podendo apenas mudar de dimensão. A Figura 4.14 ilustra três domínios com números de elementos barra ascendentes, sendo que (c) demonstra o caso em que todas as conexões entre nós estão incluídas.

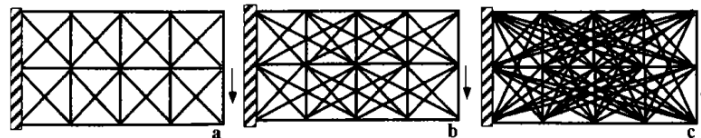


Figura 4.14: Estruturas de barras num domínio retangular com um arranjo regular de 5 por 3 nós, para transmitir uma força vertical para uma linha vertical de suportes: (a) domínio simplificado, (b) domínio com variáveis de projeto adicionais, (c) todas as conexões possíveis entre nós estão incluídas [4].

Procedeu-se então à resolução do problema de otimização topológica introduzido no Subcapítulo 3.2 utilizando este método, sendo que a estrutura de barras inicial pode ser observada na Figura 4.15, idêntica à Figura 3.3, mas agora contemplando os valores das grandezas necessárias para a resolução do problema. O problema foi resolvido em centímetros em vez de unidades SI, permitindo às variáveis de projeto estarem numa ordem de grandeza mais próxima da recomendada em [3], sendo que se considerou um módulo de Young $E = 70\text{GPa}$.

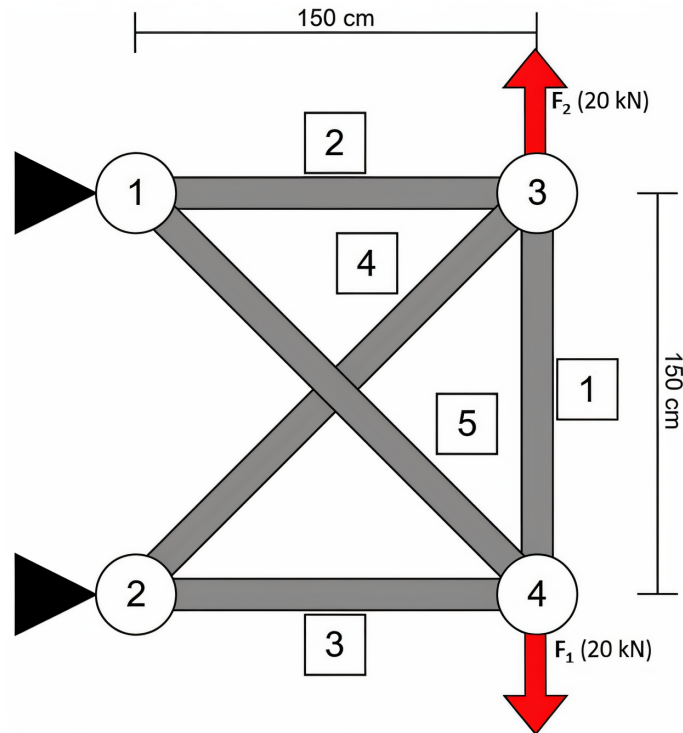


Figura 4.15: Informação associada ao problema de estrutura de barras considerado.

O problema em análise é semelhante ao estudado no Subcapítulo 4.2.1, considerando a mesma função objetivo e forças. Nota-se imediatamente que, apesar do que foi mencionado anteriormente acerca da *ground structure approach*, este conjunto de barras não reflete todas as combinações possíveis entre nós. Neste problema específico, a barra que liga os nós 1 e 2 terá uma área nula em qualquer solução do problema, independentemente da combinação de F_1 e F_2 utilizada, resultante dos apoios articulados existentes apenas nos nós 1 e 2.

O problema de otimização uniobjetivo correspondente a cada caso de carga, para $j=1,2$, é formulado do seguinte modo:

$$\begin{aligned} \min_A \quad & \mathbf{u}_j^T \mathbf{F}_j \\ \text{s.a} \quad & \\ & V \leq 0.375V_m \\ & t_i \geq 0, i = 1, \dots, m \end{aligned} \tag{4.2}$$

O problema de otimização terá assim um constrangimento linear, onde V corresponde ao volume atual da estrutura durante o processo de otimização, enquanto o volume máximo é V_m , sendo este o volume da estrutura inicial com 5 barras, onde $t_i = A_i l_i$ e $\sum_{i=1}^m t_i = V$. Neste caso, A_i , l_i e t_i correspondem, respetivamente, às áreas das secções transversais, comprimentos e volumes de cada uma das barras e F_j corresponde à força nodal dos graus de liberdade livres. Assim, a fração volúmica na equação (4.2) corresponde a 0.375.

Mais uma vez, o objetivo é encontrar a distribuição ótima de material, capaz de suportar os dois cenários de carga, F_1 e F_2 , e que simultaneamente corresponde ao menor valor

de flexibilidade, fazendo deste um problema biobjetivo. Utilizando o método da soma ponderada procura-se minimizar ϕ , tal que $\phi = w\phi_1 + (1 - w)\phi_2$, $\phi_j = \mathbf{u}^T \cdot \mathbf{F}_j$, $j=1, 2$. Minimizar a flexibilidade corresponde a minimizar o deslocamento vertical nos nós 4 e 3 causados pelas forças 1 e 2, respetivamente. Utilizando o teorema de Castigliano [24], chega-se à expressão que calcula o deslocamento no nó 4, sendo que R simboliza a componente vertical da reação causada pelo apoio no nó 1 e F_1 deverá ser interpretado como a componente vertical associada ao caso de carga \mathbf{F}_1 , aplicada no nó 4. Considera-se o deslocamento vertical especificamente neste nó, uma vez que, devido à natureza do problema, u_4y é a única componente do deslocamento útil no cálculo da energia de deformação elástica da estrutura:

$$u_4y = \frac{L}{E} \left[\frac{dR}{dF_1} \left(\frac{R}{A_1} + \frac{R}{A_2} + \frac{4R}{\sqrt{2}A_4} \right) + \left(1 - \frac{dR}{dF_1} \right) \left(\frac{F_1 - R}{A_3} + \frac{4(F_1 - R)}{\sqrt{2}A_5} \right) \right], \quad (4.3)$$

com

$$\frac{dR}{dF_1} = \frac{\frac{1}{A_3} + \frac{4}{\sqrt{2}A_5}}{\frac{1}{A_1} + \frac{1}{A_2} + \frac{1}{A_3} + 2\sqrt{2}\left(\frac{1}{A_4} + \frac{1}{A_5}\right)} \quad (4.4)$$

e

$$R = \left[\frac{\frac{1}{A_3} + \frac{4}{\sqrt{2}A_5}}{\frac{1}{A_1} + \frac{1}{A_2} + \frac{1}{A_3} + 2\sqrt{2}\left(\frac{1}{A_4} + \frac{1}{A_5}\right)} \right] F_1. \quad (4.5)$$

O deslocamento no nó 3 será determinado do mesmo modo, trocando apenas a barra 3 pela 2 e a barra 5 pela 4. Aqui R será consequentemente a componente vertical da reação causada pelo apoio no nó 2. uma vez que este problema é simétrico por natureza. Após aplicação do método da soma ponderada, o problema biobjetivo pode ser expresso como:

$$\begin{aligned} \min_A \quad & w\phi_1 + (1 - w)\phi_2 \\ \text{s.a} \quad & \phi_j = \mathbf{u}_j^T \mathbf{F}_j, j = 1, 2 \\ & V \leq 0.375V_m, \text{ com } V = \sum A_i l_i \\ & 0 \leq A_i \leq 600, \end{aligned} \quad (4.6)$$

A formulação (4.2) representa então as situações em que apenas um dos cenários de carga é aplicado, ou seja, em que $w = 0, 1$. A solução que normalmente corresponde ao menor valor de flexibilidade será um domínio de projeto cheio de material, isto é, a estrutura final seria igual à inicial. No entanto, o constrangimento linear de volume torna esta solução impossível. Como tal, as soluções que se esperam para cada um dos problemas uniobjetivo seriam do tipo das ilustradas na Figura 4.16, caso se permita uma fração volúmica de aproximadamente 0.4.

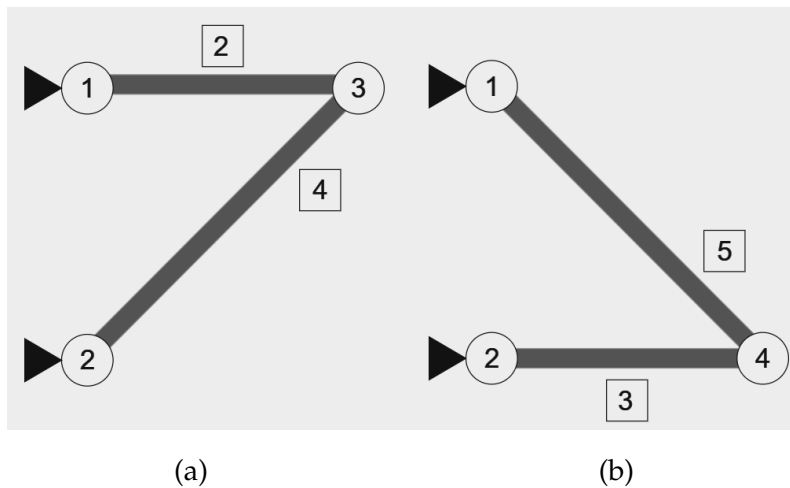


Figura 4.16: Soluções esperadas para (a) $w = 1$ e (b) $w = 0$, considerando uma fração volúmica de aproximadamente 0.4.

Este valor aproximado de 0.4 resulta da divisão entre a área que se pretende obter para os casos de carga única, igual para ambos os casos da Figura 4.16, e a área inicial da estrutura, cálculo este expresso do seguinte modo:

$$\frac{V}{V_m} = \frac{150 + \sqrt{2 \times 150^2}}{3 \times 150 + 2 \times \sqrt{2 \times 150^2}} \approx 0.414 \quad (4.7)$$

A fração volúmica final de 0.375 considerada pretende testar a validade dos processos de otimização, nomeadamente a adição de volume a barras específicas. Considerando uma fração volúmica de 0.414, o que se espera para ambos os casos extremos é uma estrutura de apenas duas barras, sendo as soluções simétricas, uma vez que as forças aplicadas também o são. Com uma fração de 0.375, as soluções correspondentes a estes casos teriam a barra diagonal com área transversal extremamente próxima do limite superior permitido e a barra horizontal com área ligeiramente menor. A Figura 4.17 representa a aproximação da frente de Pareto obtida para este problema biobjetivo pelo MMA, considerando $w \in \{0.01, 0.02, 0.03, \dots, 0.99\}$.

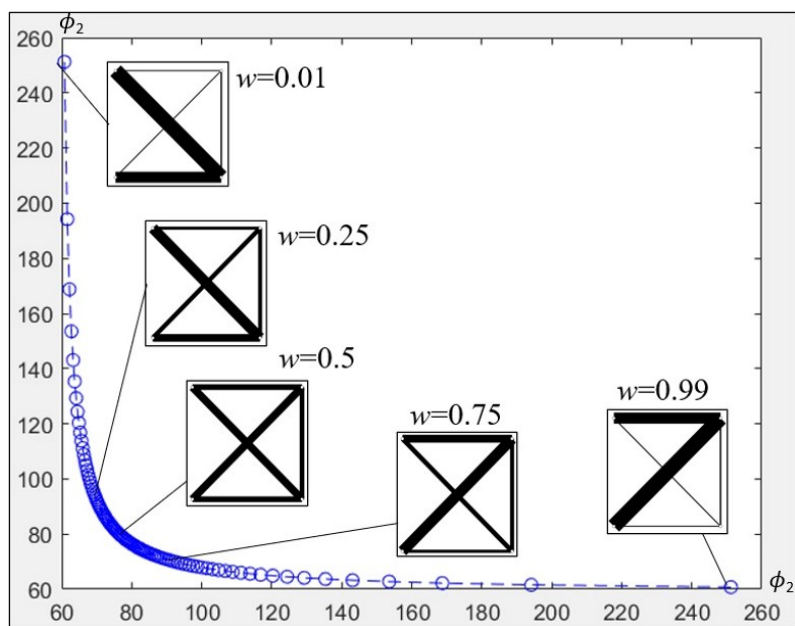


Figura 4.17: Aproximação da frente de Pareto obtida pelo MMA para a formulação em domínio discreto.

Os valores 0 e 1 não foram utilizados nos pesos da função, evitando que os valores de uma das funções objetivo tenda para o infinito. A Tabela B.1 (ver Anexo B), ilustra os valores das variáveis de projeto para cada combinação de pesos, bem como os correspondentes valores das componentes da função objetivo, permitindo uma análise detalhada. A Figura 4.18 ilustra, de modo mais simplificado, a evolução dos valores das variáveis de projeto em função do valor do peso.

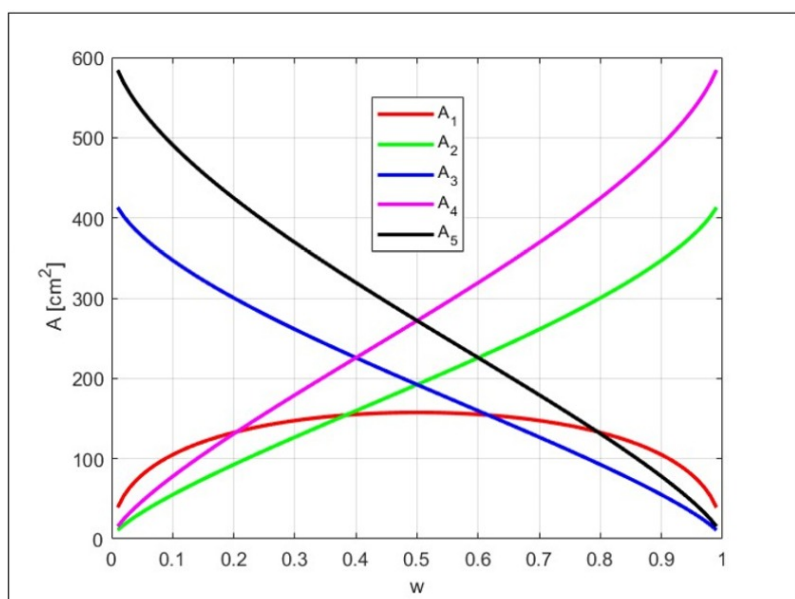


Figura 4.18: Evolução das áreas dos elementos barra em função do valor de peso.

O primeiro fator que se deve ter em conta, de modo semelhante ao que se observou

ao resolver o problema no domínio contínuo, é a simetria da aproximação da frente de Pareto obtida. Dado que as forças apresentam o mesmo valor numérico, tendo direções opostas, espera-se uma frente de Pareto que seja simétrica em relação à bissetriz dos quadrantes ímpares. Contrariamente àquilo que se observou resolvendo o problema no domínio contínuo com o *top88*, os resultados obtidos neste conjunto de testes exemplificam perfeitamente esta relação entre as situações de carga.

A evolução dos valores de área dos elementos barra, i.e, das variáveis de projeto, com a variação do peso merece igualmente análise. Quanto mais peso é colocado em F_1 , ou seja, quanto mais w se aproxima de 1, mais serão requisitadas as barras 2 e 4 e conseqüentemente, maiores serão os seus respetivos valores de área. Esta relação será inversa para as barras 3 e 5. A barra 1 atinge, como esperado, o seu valor máximo com um peso de 0.5. Uma vez que os valores de peso 0 e 1 não foram considerados, nunca estaremos perante uma situação em que apenas um cenário de carga é verdadeiramente aplicado. Esta é a razão pela qual, independentemente do valor de peso considerado, nenhuma barra atinge um valor de área de 0, visto que acabam sempre por ser pelo menos um pouco requisitadas. Do mesmo modo, não existe nenhuma situação em que uma variável de projeto atinja um valor de 600 cm^2 , correspondendo ao limite superior permitido para as mesmas. A discrepância entre as estruturas esperadas, demonstradas na Figura 4.16, e as estruturas obtidas, ilustradas na Figura 4.17, resulta deste facto.

Procedeu-se então à resolução do mesmo problema, agora utilizando o DMS. As Figuras 4.19 e 4.20 representam as aproximações da frente de Pareto obtidas correndo o DMS permitindo um máximo de 10000 e 20000 avaliações para a função objetivo. A estes testes estão associados tempos computacionais de 8 e 16 segundos, respetivamente. A abordagem baseada no MMA demorou 25 segundos de tempo computacional.

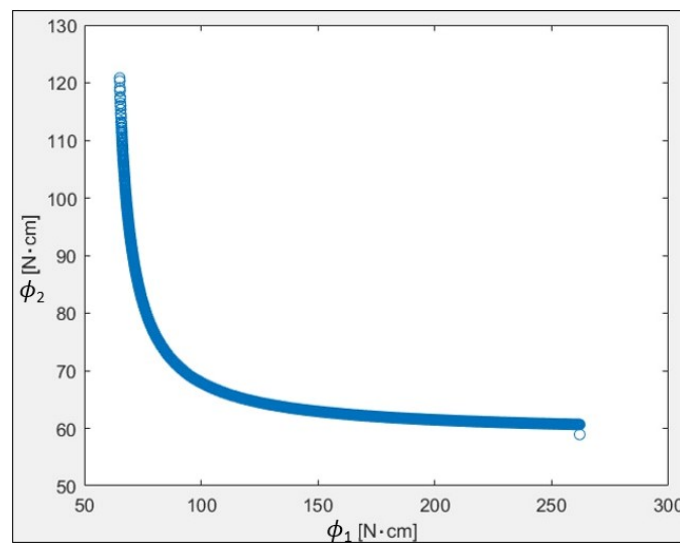


Figura 4.19: Aproximação da frente de Pareto obtida pelo DMS, permitindo 10000 avaliações da função objetivo.

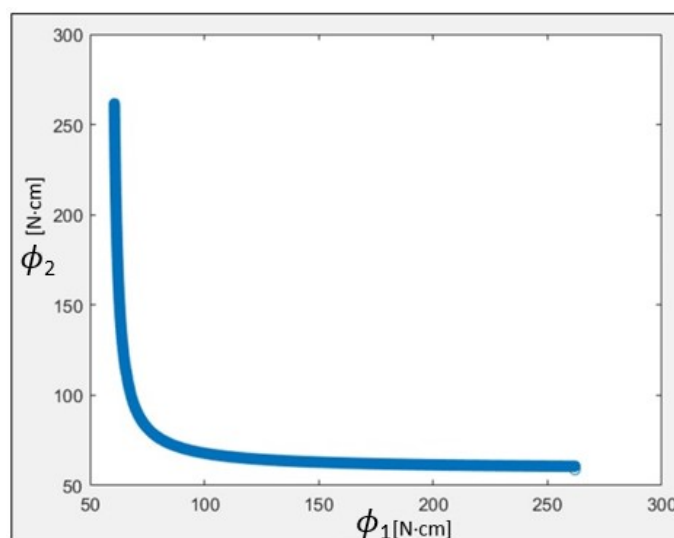


Figura 4.20: Aproximação da frente de Pareto obtida pelo DMS, permitindo 20000 avaliações da função objetivo.

Como é possível observar, a aproximação da frente de Pareto obtida pelo DMS permitindo 20000 avaliações está extremamente próxima da gerada pelo MMA, quer em termos de forma e simetria, quer em termos de qualidade de soluções. Na aproximação da frente de Pareto correspondente às 10000 avaliações nota-se um número notavelmente menor de soluções, e em particular que a aproximação da frente de Pareto obtida não é ainda simétrica como esperado para este problema. Tal resulta do critério de paragem considerado. Estes resultados tornam-se ainda mais significativos quando se considera que o DMS foi capaz de obter 2230 e 2652 pontos nestes dois testes, respetivamente, resultados claramente melhores do que os observados no estudo em domínio contínuo. É importante salientar que, no decorrer destes testes, considerou-se atingida convergência quando o comprimento de passo associado a cada ponto fosse inferior a 0.001. No entanto, o valor mínimo de comprimento de passo observado no decorrer destes testes foi muito superior a este limite, o que indica que estes resultados ainda estão longe da convergência, havendo espaço para melhoria. De modo a verificar que a natureza simétrica das soluções esperadas para este problema não foi violada, na Tabela 4.2 ilustra-se alguns dos pontos obtidos com o critério de paragem de 20000 cálculos, assim como os respetivos valores de variáveis de projeto. É importante referir que foi efetuado um outro teste, permitindo 100000 avaliações da função objetivo. A aproximação da frente de Pareto obtida é visualmente idêntica à da Figura 4.20, embora consideravelmente mais densa, tendo sido obtidos 15175 pontos. O tempo computacional associado a este teste foram aproximadamente 712 segundos, provando assim a relação não linear entre o número de avaliações da função objetivo permitidas ao algoritmo e o tempo computacional.

Tabela 4.2: Algumas soluções e valores de variáveis de projeto correspondentes obtidos pelo DMS, considerando um máximo de 100000 avaliações da função objetivo.

ϕ_1	ϕ_2	A_1	A_2	A_3	A_4	A_5
262	51.925	0.005826	3.62E-05	462.935	8.60E-05	599.9479
78.172	78.172	152.3809	195.0812	195.0812	271.8305	271.8305
60.666	261.999	585.7934	15.19773	413.7684	10.7752	36.92268

Passou-se então a uma comparação direta entre o DMS e o MMA. Mais uma vez, para este teste foi implementado no DMS um critério de paragem com base no tempo computacional, considerando um limite de 25 segundos, idêntico ao tempo computacional requerido pelo MMA para a resolução do problema. A Figura 4.21 ilustra as aproximações das frentes de Pareto obtidas por cada um dos algoritmos.

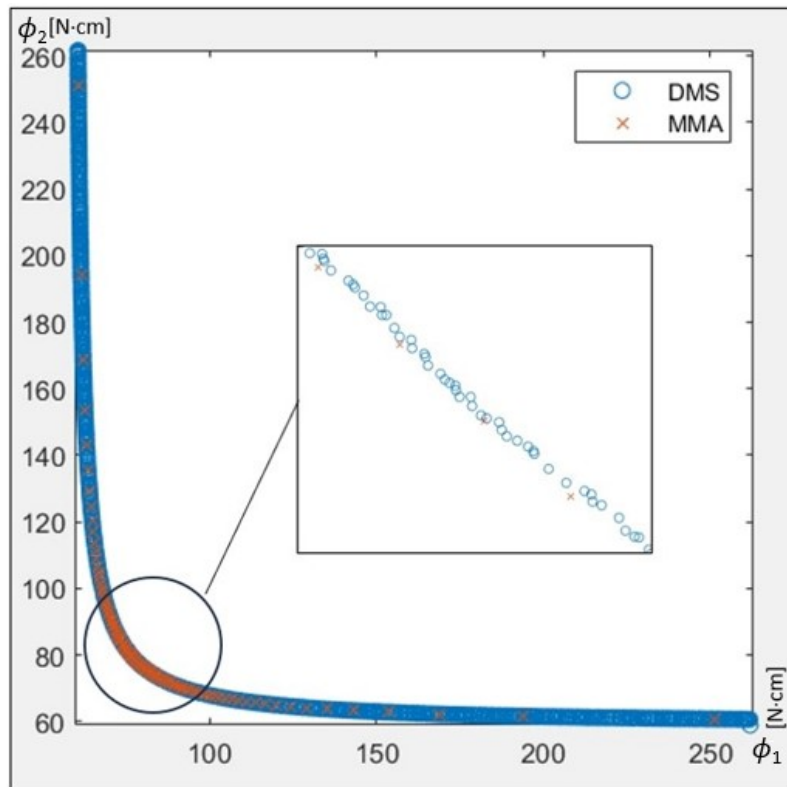


Figura 4.21: Aproximações das frentes de Pareto obtidas pelo DMS e pelo MMA, permitindo um tempo computacional máximo de 25 segundos.

É possível observar que as soluções obtidas pelo MMA dominam algumas das soluções obtidas pelo DMS. No entanto, é importante notar que o DMS foi capaz de gerar 454 soluções no mesmo tempo computacional em que o MMA gera 100. Combinado com o facto de que as soluções obtidas pelo DMS estão longe de convergir, tal evidencia a viabilidade do DMS na resolução de problemas de otimização topológica em domínio discreto. Comparando os resultados obtidos com as formulações nos dois domínios, a hipótese de que, perante este tipo de problema, o número de variáveis é o maior obstáculo do DMS fica bem estabelecida.

Finalmente, compararam-se ainda os resultados obtidos pelo DMS e pelo BoostDMS, estando os mesmos ilustrados na Figura 4.22.

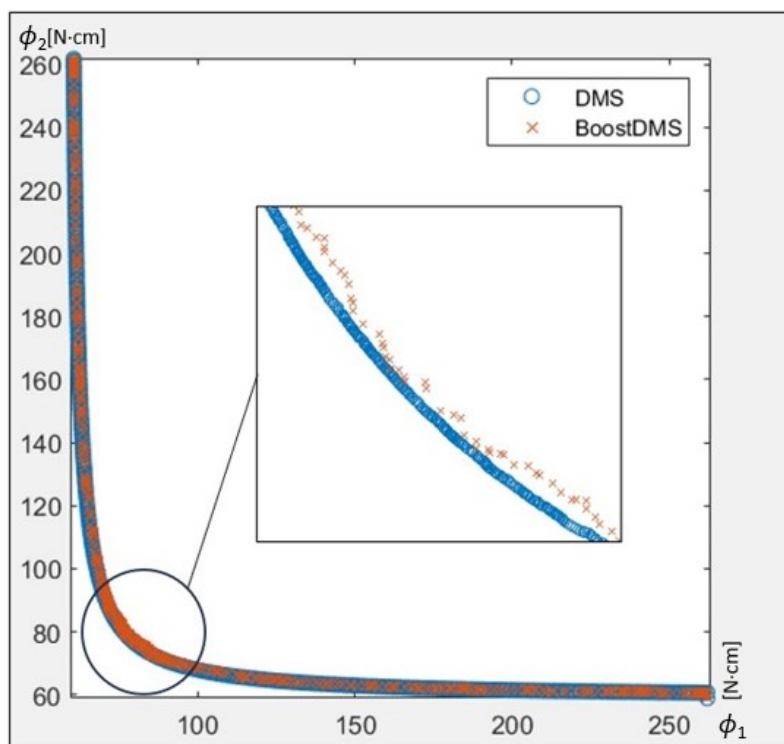


Figura 4.22: Aproximações das frentes de Pareto obtidas pelo DMS e pelo BoostDMS, permitindo um tempo computacional de 25 segundos.

Os resultados obtidos pelo DMS dominam totalmente os obtidos pelo BoostDMS. A adição do passo de procura aumenta consideravelmente o tempo computacional necessário para adicionar pontos à aproximação da frente de Pareto, como resultado da resolução dos sistemas de equações lineares que calculam os coeficientes dos modelos polinomiais quadráticos, usados neste passo.

De modo a continuar o estudo do potencial do DMS associado à resolução de problemas de otimização topológica em domínio discreto, decidiu-se considerar um problema com um maior número de variáveis. O novo problema, resolvido ainda em domínio discreto, considera uma repetição do problema anteriormente abordado, agora com um domínio de projeto mais interessante, resultante da adição de novos nós e, conseqüentemente, do aumento do número de variáveis de projeto. A Figura 4.23 ilustra o problema proposto, com as dimensões e valores numéricos do problema anterior.

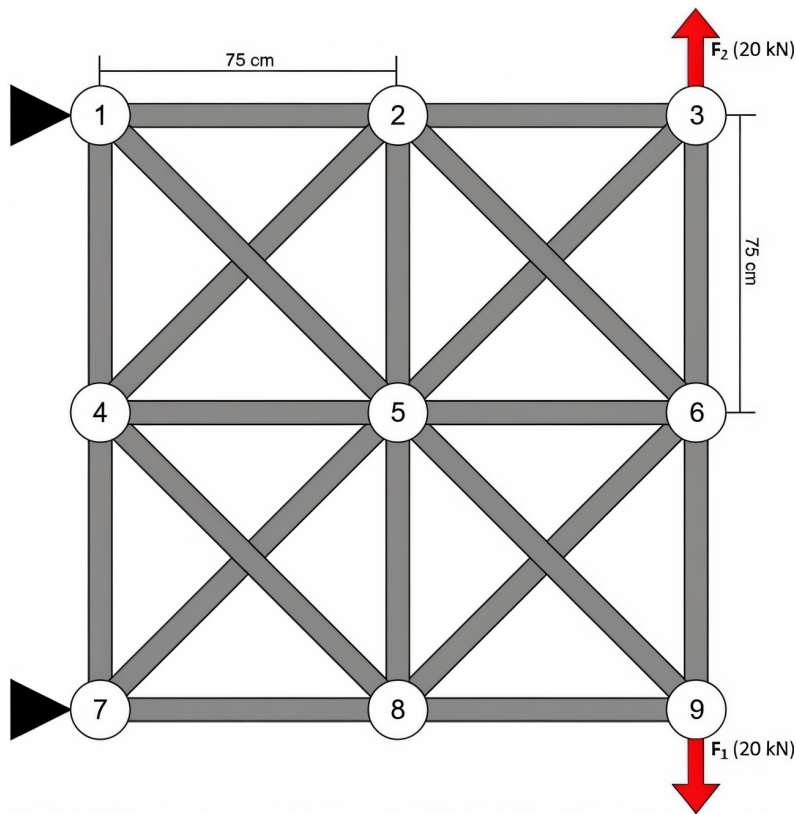


Figura 4.23: Informação associada ao problema de 20 elementos de barra considerado.

O novo problema tem o quádruplo das variáveis de projeto do problema formulado em domínio discreto original. Como tal, os resultados serão mais esclarecedores no que toca à viabilidade do uso do DMS em otimização topológica. Mais uma vez, para as situações de aplicação de carga única, pretende-se obter estruturas semelhantes às da Figura 4.16, o que consiste na presença dos elementos de barra 1-2, 2-3, 3-5, e 5-7, no caso de aplicação de F_2 . Tal traduz-se na utilização de um valor máximo de fração volúmica calculado do seguinte modo:

$$\frac{V}{V_m} = \frac{2 \times 75 + 2 \times \sqrt{2 \times 75^2}}{12 \times 75 + 8 \times \sqrt{2 \times 75^2}} \approx 0.207, \quad (4.8)$$

tendo sido então permitido um valor de 0.2. Novamente, começou-se por resolver o problema com o MMA, recorrendo ao método da soma ponderada, com o objetivo de obter uma aproximação da frente de Pareto com 100 pontos de referência. Os resultados, correspondentes a um tempo computacional de cerca de 42 segundos, encontram-se representados na Figura 4.24.

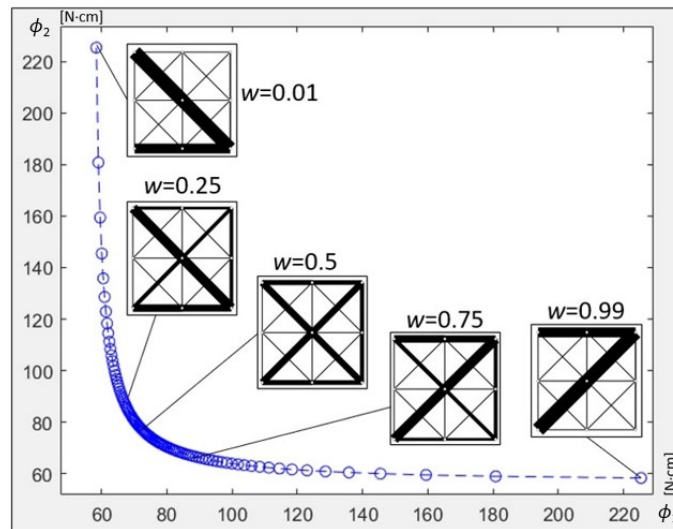


Figura 4.24: Aproximação da frente de Pareto obtida pelo MMA para o problema de 20 elementos barra.

Apesar do problema possuir um número maior de variáveis de projeto, a manutenção dos valores numéricos e da simetria do problema torna o novo exercício quase idêntico ao original em termos de solução. Tal como se esperaria duma treliça com este tipo de *design*, os elementos de barra centrais são desnecessários no suporte de qualquer um dos casos de carga considerados. Como tal, independentemente do valor de peso considerado, a solução do problema de otimização não adiciona qualquer área a estes elementos barra.

O problema foi então resolvido com o DMS, considerando como critério de paragem os 42 segundos de execução utilizados pelo MMA. Os resultados, bem como a comparação com o MMA, estão representados na Figura 4.25.

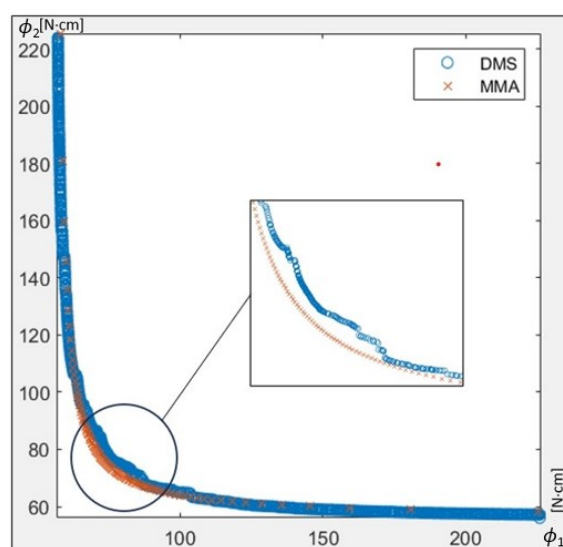


Figura 4.25: Aproximações da frente de Pareto obtidas pelo MMA e pelo DMS para o problema de 20 elementos barra, permitindo um tempo computacional máximo de 42 segundos.

De forma análoga às conclusões extraídas da resolução do problema com 5 variáveis de projeto, o DMS gera uma aproximação da frente de Pareto com a forma e simetria esperadas, sendo no entanto dominada pela curva obtida pelo MMA. Ainda assim, mais uma vez, o DMS consegue captar um conjunto de soluções mais numeroso. Ao contrário do que se observou no problema anterior, desta vez nota-se, principalmente no meio da curva, uma maior discrepância entre as soluções obtidas pelos dois algoritmos. Tal deve-se ao facto de que o tempo necessário para o MMA obter a sua aproximação final está mais longe do tempo computacional requerido pelo DMS para atingir convergência, prova do impacto que o número de variáveis de projeto tem sobre o DMS.

Outra evidência clara do impacto do aumento do número de variáveis surge correndo o BoostDMS. A Figura 4.26 prova que a introdução de 15 variáveis de projeto adicionais torna o problema complexo o suficiente para que o BoostDMS já não seja capaz de capturar uma aproximação da frente de Pareto aceitável dentro da janela de tempo considerada. Nota-se que em cada iteração, o BoostDMS resolve agora um sistema de equações lineares com $\frac{21 \times 22}{2}$ variáveis.

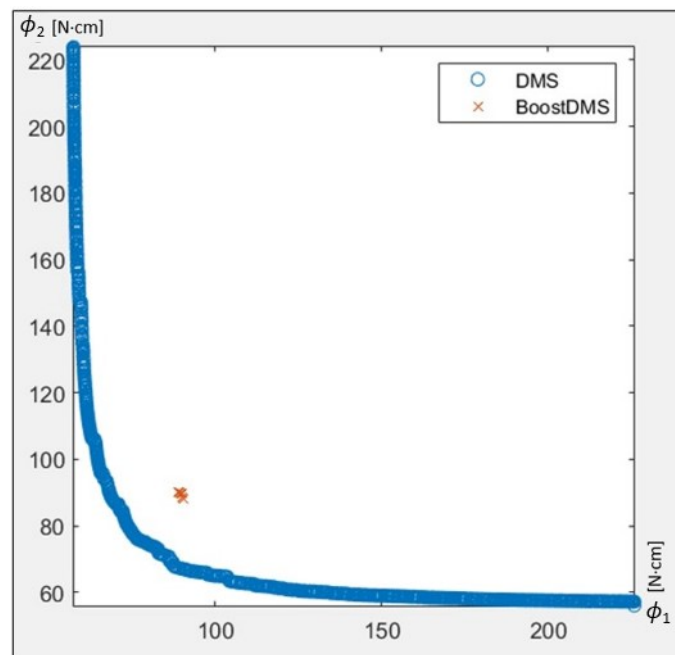


Figura 4.26: Aproximações da frente de Pareto obtidas pelo DMS e pelo BoostDMS para o problema de 20 elementos barra, permitindo um tempo computacional de 42 segundos.

Com o estudo em domínio discreto conclui-se, mais uma vez, que o número de variáveis de projeto parece ser determinante para o desempenho computacional do DMS em problemas de otimização topológica. Contudo, contrariamente àquilo que se observou no Subcapítulo 4.2.1, o facto do problema já não se resolver no domínio contínuo permite ao DMS obter não só bons resultados, mas uma frente de Pareto muito mais detalhada do que aquela captada pelo MMA. No entanto, o passo de procura adicional providenciado pelo BoostDMS demonstra ser o suficiente para fazer do tempo computacional um fator

proibitivo. Como tal, se no futuro se planear estudar mais a fundo esta variante do DMS, recomenda-se fazê-lo para problemas de menor dimensão, ou melhorar os algoritmos de resolução de sistemas de equações lineares.

Para aprofundar o estudo, finalmente resolveu-se o problema anterior permitindo todas as ligações possíveis entre nós, aumentando o número de variáveis de projeto para 28. A Figura 4.27 representa a nova situação em análise.

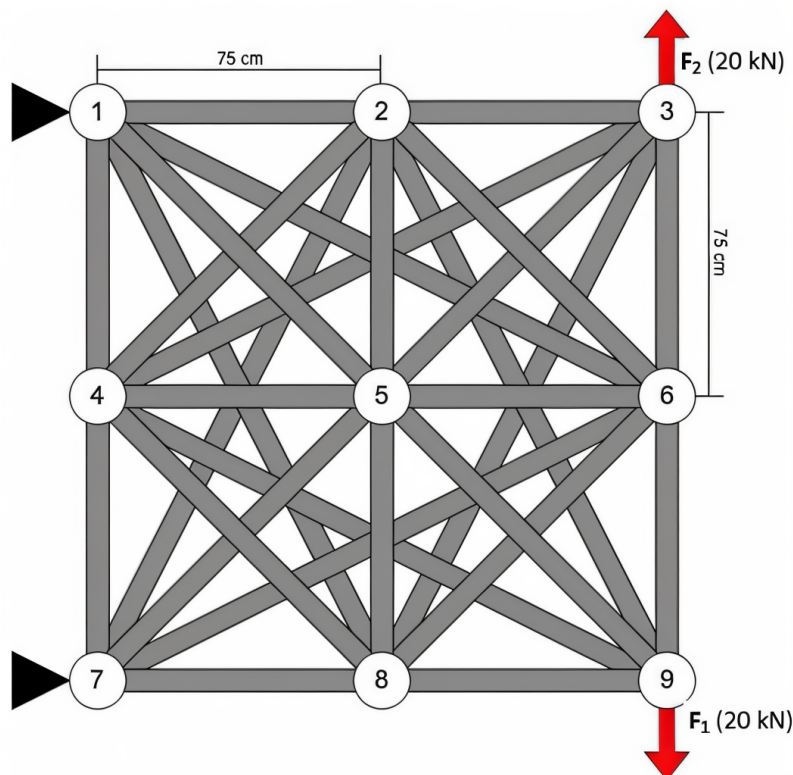


Figura 4.27: Informação associada ao problema de 28 elementos de barra considerado.

Neste caso, o valor de fração volúmica foi calculado utilizando o mesmo raciocínio que no problema de 20 variáveis de projeto. Visto que apenas o valor de V_m varia, a fração volúmica torna-se então a seguinte:

$$\frac{V}{V_m} = \frac{2 \times 75 + 2 \times \sqrt{2 \times 75^2}}{12 \times 75 + 8 \times \sqrt{2 \times 75^2} + 8 \times \sqrt{75^2 + 150^2}} \approx 0.117. \quad (4.9)$$

Segue-se o conjunto de soluções obtido pelo MMA, a ser utilizado como valor de referência para comparação com o DMS. A esta aproximação da frente de Pareto, ilustrada na Figura 4.28, está associado um tempo computacional de aproximadamente 50 segundos.

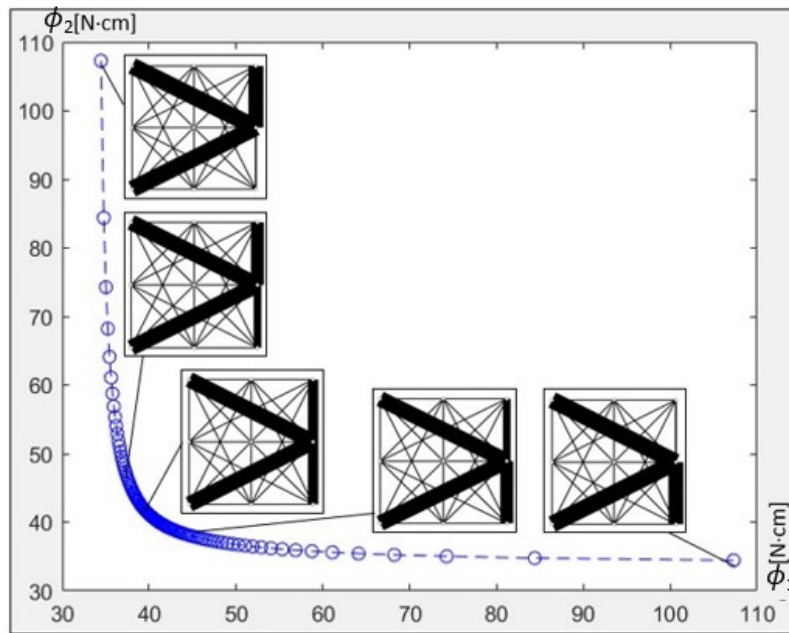


Figura 4.28: Aproximação da frente de Pareto obtida pelo MMA para o problema de 28 elementos barra.

Como se pode observar, a adição dos novos elementos barra altera substancialmente as soluções obtidas, permitindo novas estruturas que, ocupando a mesma percentagem volúmica, apresentam valores mais baixos de flexibilidade. Na Figura 4.29, adicionalmente representam-se os resultados obtidos pelo DMS, não incluindo o passo de procura com base em modelos quadráticos, utilizando os 50 segundos de tempo computacional como critério de paragem.

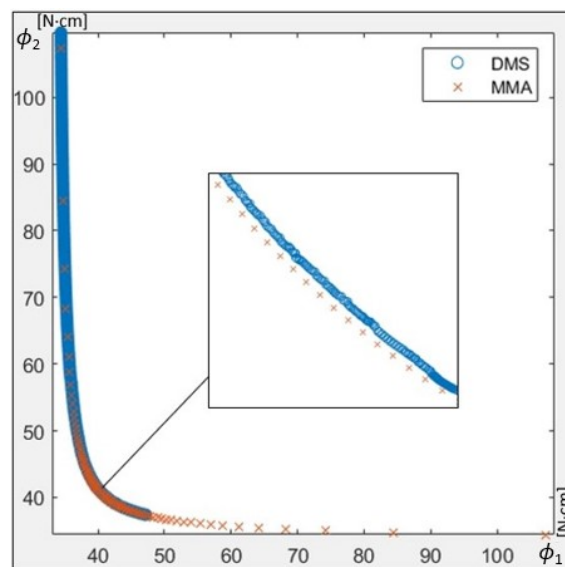


Figura 4.29: Aproximações das frentes de Pareto obtidas pelo MMA e pelo DMS para o problema de 28 elementos barra, permitindo um tempo computacional máximo de 50 segundos.

Pela primeira vez, o tempo necessário para o MMA obter resultados não é suficiente para o DMS obter uma aproximação da frente de Pareto completamente desenvolvida. Apesar das soluções obtidas pelo DMS evidenciarem alguma qualidade, o número de variáveis de projeto claramente prejudicou o desempenho deste algoritmo. No entanto, é ainda importante considerar que o DMS, no tempo considerado, foi capaz de obter 3458 pontos. Para referência, na Figura 4.30 representam-se os resultados obtidos pelo DMS, quando se altera o critério de paragem para um máximo de 100000 avaliações da função objetivo, correspondendo a aproximadamente 90 segundos de tempo computacional.

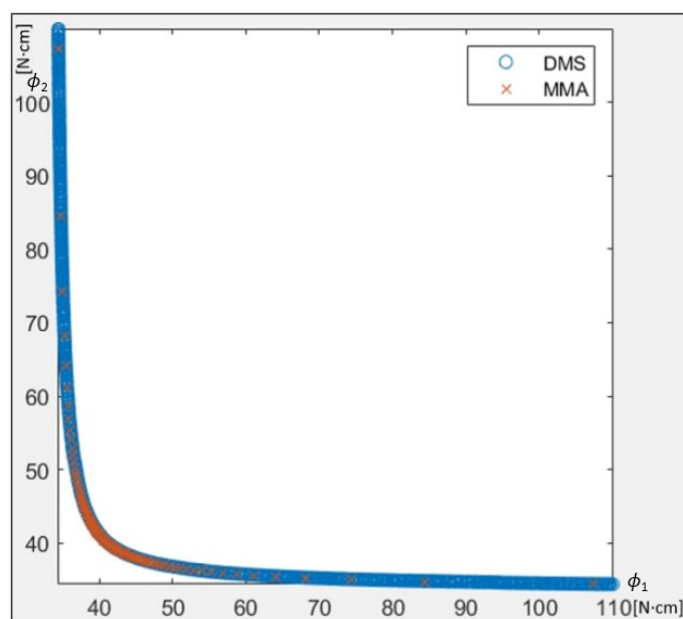


Figura 4.30: Aproximação da frente de Pareto obtida pelo DMS para o problema de 28 elementos barra, permitindo um máximo de 100000 avaliações da função objetivo e comparação com a solução obtida com o MMA.

Como se pode verificar o DMS continua capaz de obter bons resultados neste problema. No entanto, considerando um problema com 28 variáveis de projeto, atinge-se um ponto no qual uma comparação entre algoritmos com base no tempo computacional começa a favorecer fortemente o MMA, de modo semelhante àquilo que se observou durante o estudo em domínio contínuo.

CONCLUSÕES E TRABALHO FUTURO

Ao longo desta dissertação foi analisada a aplicação de métodos de otimização multi-objetivo, com e sem derivadas, no projeto de estruturas mecânicas multicarga. Foram explorados diferentes algoritmos, incluindo o Direct Multisearch (DMS) e a sua variante BoostDMS, comparando o seu desempenho numérico com métodos tradicionais baseados em gradientes, como o Optimality Criteria Method (OC) e o Method of Moving Asymptotes (MMA). A investigação envolveu a formulação e resolução de problemas em domínios contínuos e discretos, com o objetivo de avaliar a viabilidade e o desempenho destas abordagens na obtenção de soluções ótimas para otimização topológica. Os resultados permitiram compreender as vantagens e limitações de cada método, fornecendo direções para trabalhos futuros na área.

É possível concluir então que o número de variáveis de projeto é o fator que mais influencia o desempenho do DMS no que toca a este tipo de problemas, principalmente quando a comparação entre algoritmos é efetuada com base no tempo computacional. Relativamente a problemas de otimização topológica em domínio contínuo, tornou-se claro que a utilização do DMS não é competitiva face aos métodos tradicionalmente utilizados. Apesar das estruturas obtidas mostrarem potencial, quando comparadas com as soluções obtidas pelo OC, os tempos computacionais exigidos tornam impraticável a aplicação do DMS a este tipo de formulação. A dimensão dos problemas que se conseguem resolver é um fator extremamente limitante, visto que, mesmo permitindo tempos computacionais razoáveis, o DMS é incapaz de obter uma frente de Pareto comparável à do OC num problema de 100 variáveis. Esta distância entre algoritmos é exacerbada pelo facto da malha de 10x10 correspondente já ser considerada uma malha grosseira em otimização topológica.

Porém, quando o problema a resolver apresenta uma dimensão reduzida, como visto com a resolução do problema em domínio discreto, o DMS é capaz de obter aproximações da frente de Pareto bastante mais detalhadas do que o MMA, apesar da existência de alguns pontos dominados, fruto dos critérios de paragem considerados. Considerando ainda que existe uma dependência entre o tempo de execução do algoritmo e o "hardware" utilizado, podemos concluir que, nas condições adequadas, os resultados do DMS têm o potencial de

melhorar substancialmente. No que toca ao BoostDMS, a adição dos modelos polinomiais quadráticos torna o tempo computacional um fator proibitivo, sendo de facto realista utilizá-lo apenas em problemas com um número muito reduzido de variáveis de projeto, ou se, mais uma vez, o critério de comparação considerado não for o tempo de execução.

Para além dos estudos realizados nesta dissertação, seria bastante interessante explorar a aplicação destes algoritmos a problemas de minimização de massa, que são bastante relevantes no contexto da engenharia estrutural, especialmente quando se procura reduzir o peso sem comprometer a resistência e rigidez da estrutura. Este tipo de formulação poderia revelar novas perspetivas sobre a eficácia do DMS e do BoostDMS, nomeadamente em cenários onde a massa seja um objetivo crítico. Para além disto, também seria relevante testar estes métodos em problemas com mais do que dois objetivos. Expandir a análise para problemas com três ou mais objetivos permitiria avaliar de forma mais abrangente a capacidade dos algoritmos em gerar aproximações da frente de Pareto diversificadas e bem distribuídas, algo fundamental em problemas de decisão complexos com múltiplos critérios de desempenho.

Em suma, no que toca a otimização topológica e problemas multiobjetivo no geral, a utilização do DMS não deve ser menosprezada, principalmente se aquilo que o agente de decisão valorizar mais for o detalhe da frente de Pareto final. Tendo em conta os resultados obtidos ao longo da dissertação, seria de extremo interesse uma continuação da comparação entre o DMS e os algoritmos tipicamente utilizados em otimização topológica. No entanto, no futuro, esta comparação deve ser efetuada com uma versão do DMS cujos resultados tenham atingido a convergência, independentemente do tempo computacional necessário.

APÊNDICES

PROBLEMAS ACADÉMICOS

BK1 [12]

$$\begin{aligned} \min_{\mathbf{x}} \quad & (f_1(\mathbf{x}), f_2(\mathbf{x})) \\ \text{s.a.} \quad & -5 \leq x_i \leq 10, \text{ com } \mathbf{x} = (x_1, \dots, x_n) \text{ e} \\ & f_1(\mathbf{x}) = \sum_{i=1}^n x_i^2 \\ & f_2(\mathbf{x}) = \sum_{i=1}^n (x_i - 5)^2 \end{aligned}$$

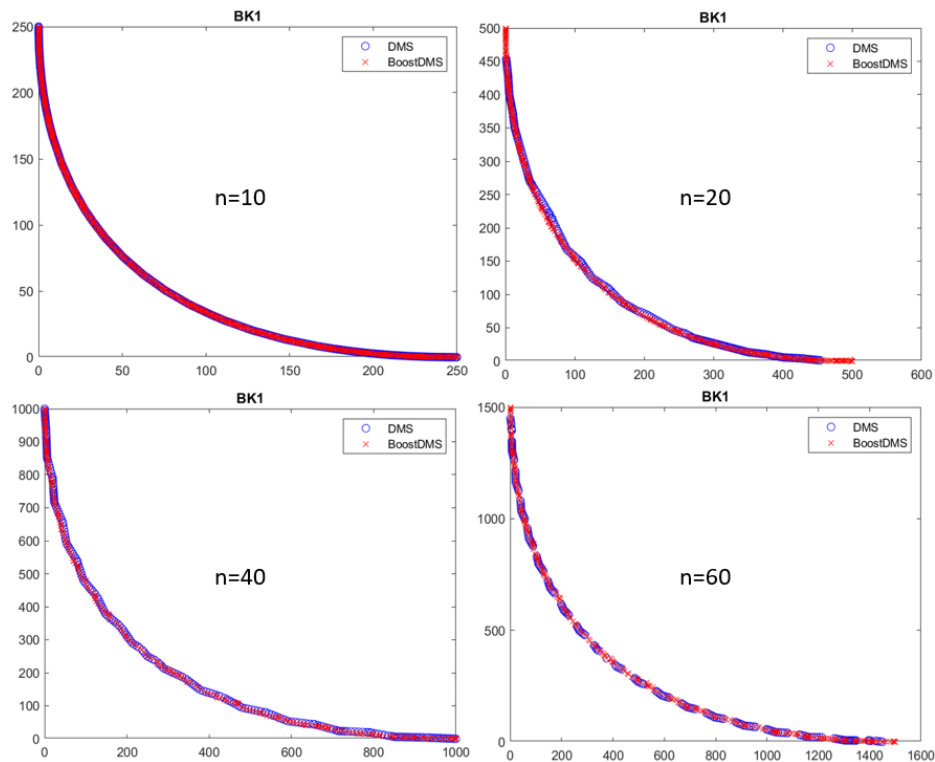


Figura A.1: Frentes de Pareto obtidas pelos algoritmos DMS e BoostDMS para o problema BK1, considerando 10, 20, 40 e 60 variáveis.

DTLZ3 [13]

$$\begin{aligned} & \min_{\mathbf{x}} (f_1(\mathbf{x}), \dots, f_m(\mathbf{x})) \\ & \text{s.a. } 0 \leq x_i \leq 1, \text{ com } \mathbf{x} = (x_1, \dots, x_n), k = n - m + 1 \text{ e} \\ & g(\mathbf{x}) = 100(k + (\sum_{i=m}^n (x_i - 0.5)^2 - \cos(20\pi(x_i - 0.5))))), \\ & f_1(\mathbf{x}) = (1 + g(\mathbf{x})) \prod_{i=1}^{m-1} \cos(0.5\pi x_i), \\ & f_j(\mathbf{x}) = (1 + g(\mathbf{x})) \sin(0.5\pi x_{m-j+1}) \prod_{i=1}^{m-j} \cos(0.5\pi x_i), 2 \leq j \leq m, j \in \mathbb{N} \end{aligned}$$

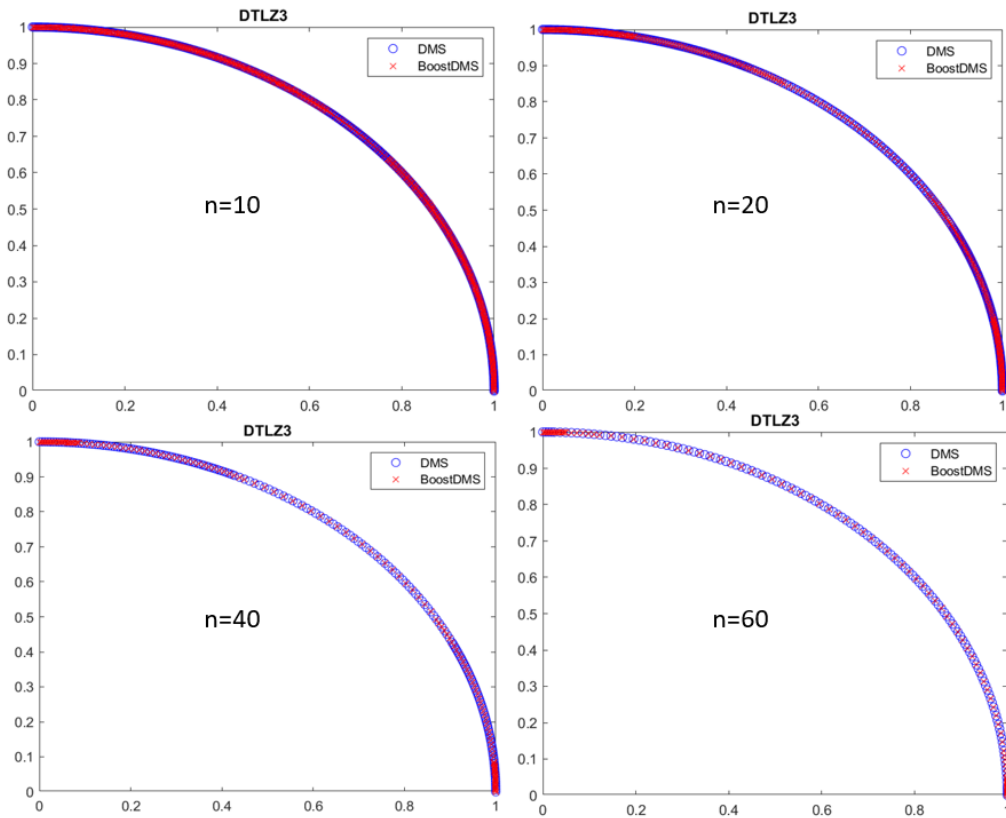


Figura A.2: Frentes de Pareto obtidas pelos algoritmos DMS e BoostDMS para o problema DTLZ3, considerando 10, 20, 40 e 60 variáveis.

DTLZ4 [13]

$$\begin{aligned}
 & \min_{\mathbf{x}} (f_1(\mathbf{y}), \dots, f_m(\mathbf{y})) \\
 & \text{s.a. } 0 \leq \mathbf{x} \leq 1, \text{ com } \mathbf{x} = (x_1, \dots, x_n) \text{ e} \\
 & \alpha = 100, \mathbf{y} = \mathbf{x}^\alpha \\
 & g(\mathbf{y}) = \sum_{i=m}^n (y_i - 0.5)^2, \\
 & f_1(\mathbf{y}) = (1 + g(\mathbf{y})) \prod_{i=1}^{m-1} \cos(0.5\pi y_i), \\
 & f_j(\mathbf{y}) = (1 + g(\mathbf{y})) \sin(0.5\pi x_{m-j+1}) \prod_{i=1}^{m-j} \cos(0.5\pi y_i), 2 \leq j \leq m, j \in \mathbb{N}
 \end{aligned}$$

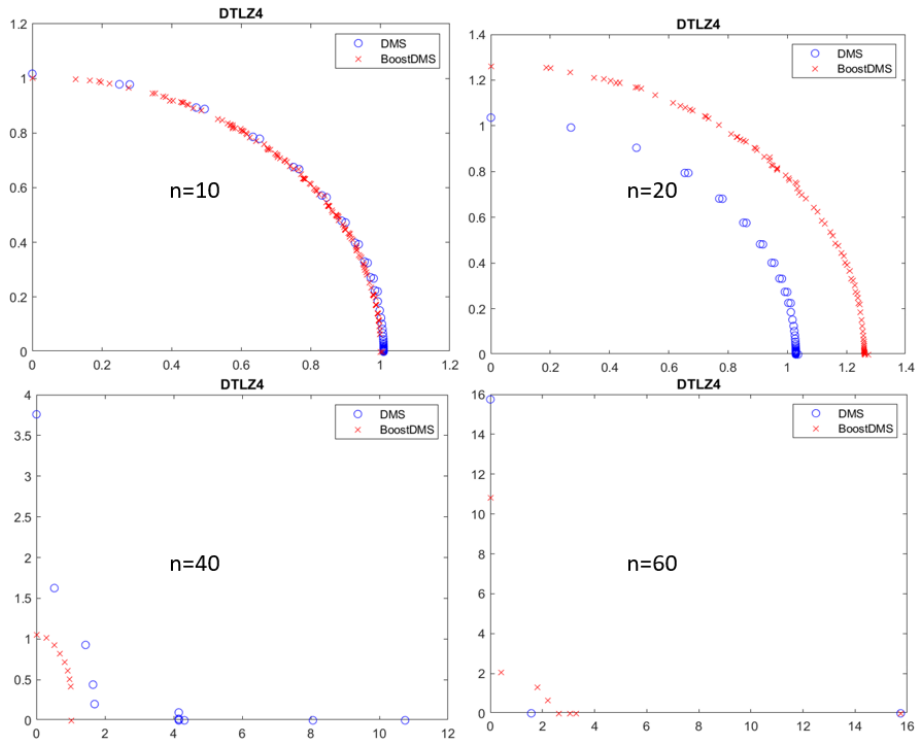


Figura A.3: Frentes de Pareto obtidas pelos algoritmos DMS e BoostDMS para o problema DTLZ4, considerando 10, 20, 40 e 60 variáveis.

DTLZ6 [13]

$$\begin{aligned} & \min_{\mathbf{x}} (f_1(\mathbf{x}), f_2(\mathbf{x})) \\ & \text{s.a. } 0 \leq \mathbf{x} \leq 1, \text{ com } \mathbf{x} = (x_1, \dots, x_n), k = n - m + 1 \text{ e} \\ & g(\mathbf{x}) = 1 + \frac{9}{k} + \sum_{i=m}^n x_i, \\ & h(\mathbf{x}) = m - \sum_{i=1}^n f_i(\mathbf{x})(1 + \sin(3\pi x_i)), \\ & f_1(\mathbf{x}) = x_1, \\ & f_2(\mathbf{x}) = (1 + g(\mathbf{x}))h(\mathbf{x}) \end{aligned}$$

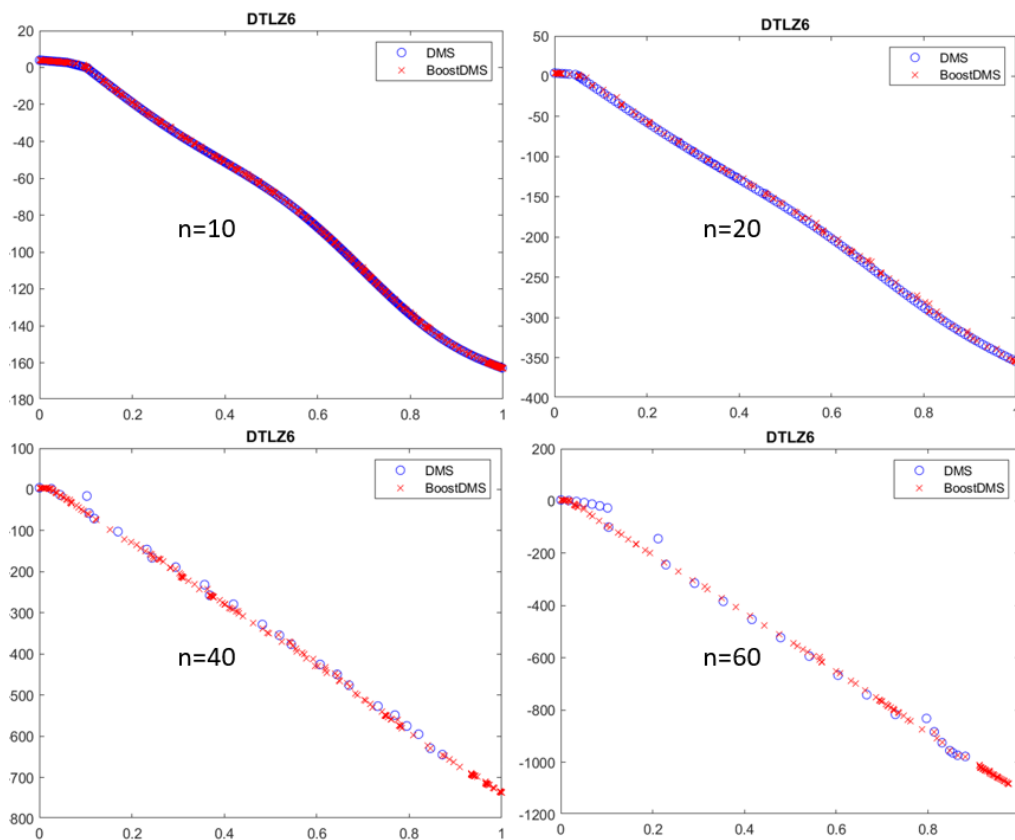


Figura A.4: Frentes de Pareto obtidas pelos algoritmos DMS e BoostDMS para o problema DTLZ6, considerando 10, 20, 40 e 60 variáveis.

Jin1 [15]

$$\begin{aligned} \min_{\mathbf{x}} \quad & (f_1(\mathbf{x}), f_2(\mathbf{x})) \\ \text{s.a.} \quad & 0 \leq x_i \leq 1, \text{ com } \mathbf{x} = (x_1, \dots, x_n) \text{ e} \\ & f_1(\mathbf{x}) = \frac{1}{n} \sum_{i=1}^n x_i^2, \\ & f_2(\mathbf{x}) = \frac{1}{n} \sum_{i=1}^n (x_i - 2)^2 \end{aligned}$$

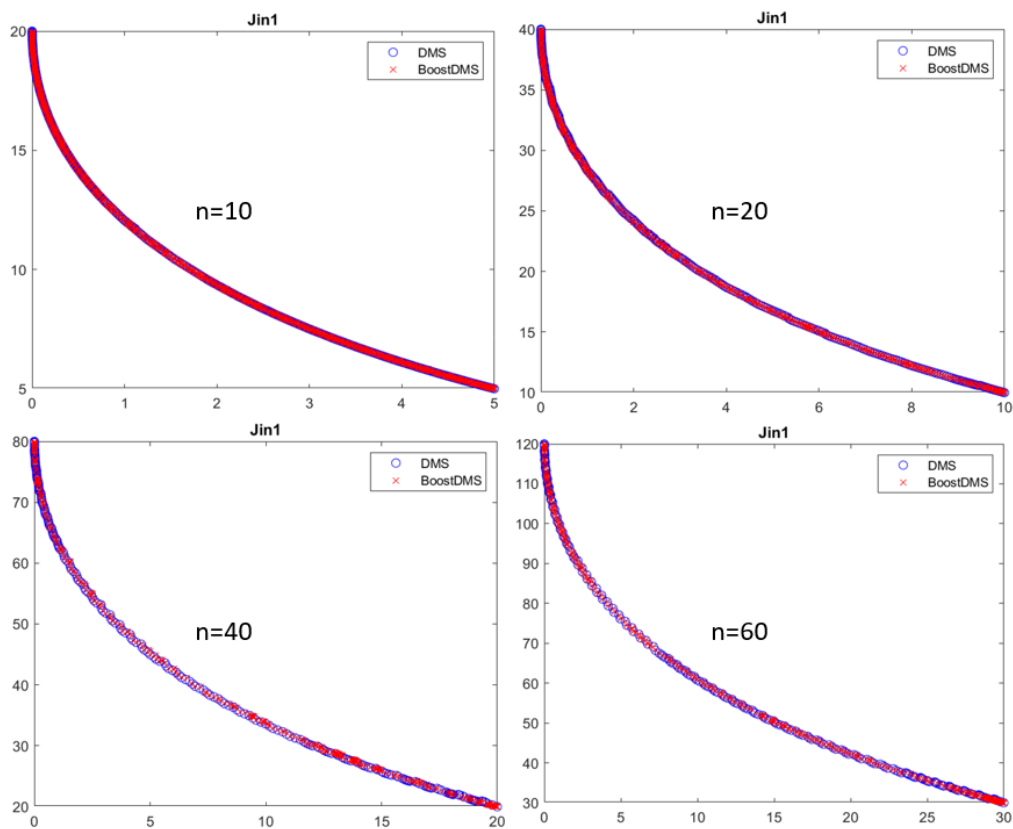


Figura A.5: Frentes de Pareto obtidas pelos algoritmos DMS e BoostDMS para o problema Jin1, considerando 10, 20, 40 e 60 variáveis.

MOP2 [12]

$$\begin{aligned} \min_{\mathbf{x}} \quad & (f_1(\mathbf{x}), f_2(\mathbf{x})) \\ \text{s.a} \quad & -4 \leq x_i \leq 4, \text{ com } \mathbf{x} = (x_1, \dots, x_n) \text{ e} \\ & f_1(\mathbf{x}) = 1 - \exp\left(-\sum_{i=1}^n (x_i - 1/\sqrt{n})^2\right), \\ & f_2(\mathbf{x}) = 1 - \exp\left(-\sum_{i=1}^n (x_i - 1/\sqrt{n})^2\right) \end{aligned}$$

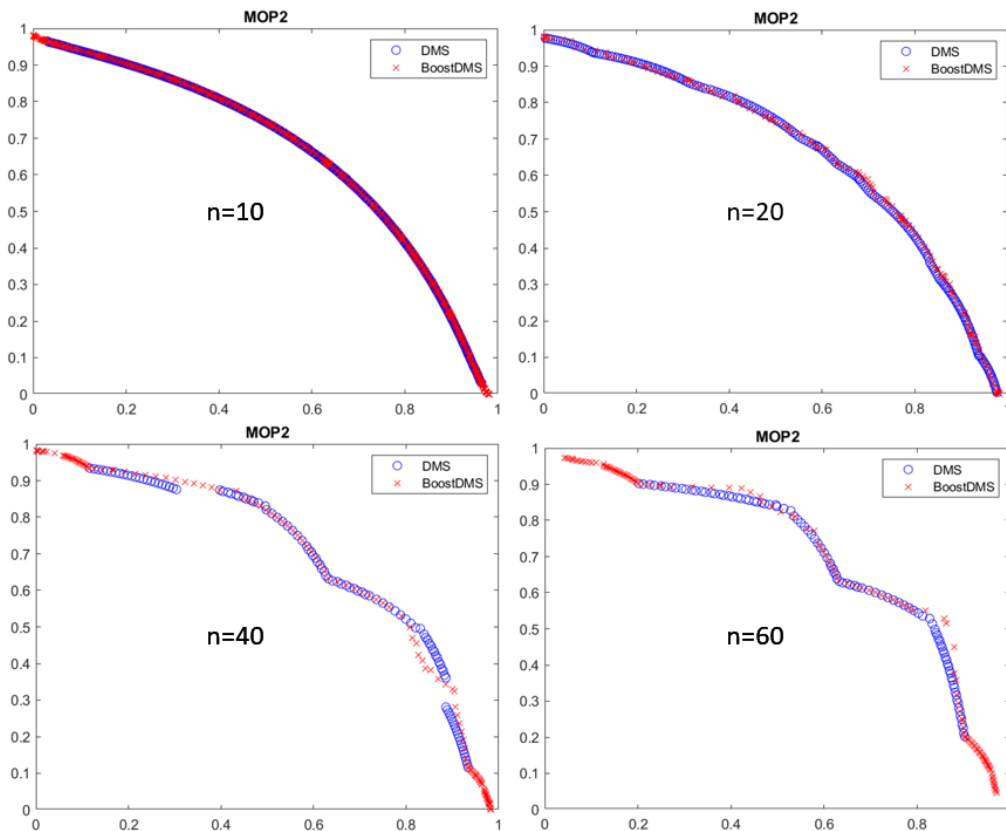


Figura A.6: Frentes de Pareto obtidas pelos algoritmos DMS e BoostDMS para o problema MOP2, considerando 10, 20, 40 e 60 variáveis.

ZDT2 [14]

$$\begin{aligned} \min_{\mathbf{x}} \quad & (f_1(\mathbf{x}), f_2(\mathbf{x})) \\ \text{s.a} \quad & 0 \leq \mathbf{x} \leq 1, \text{ com } \mathbf{x} = (x_1, \dots, x_n) \text{ e} \\ & f_1(\mathbf{x}) = x_1, \\ & f_2(\mathbf{x}) = g(\mathbf{x})h(\mathbf{x}), \\ & g(\mathbf{x}) = 1 + \frac{9}{n-1} \sum_{i=2}^n x_i, \\ & h(\mathbf{x}) = 1 - (f_1(\mathbf{x})/g(\mathbf{x}))^2 \end{aligned}$$

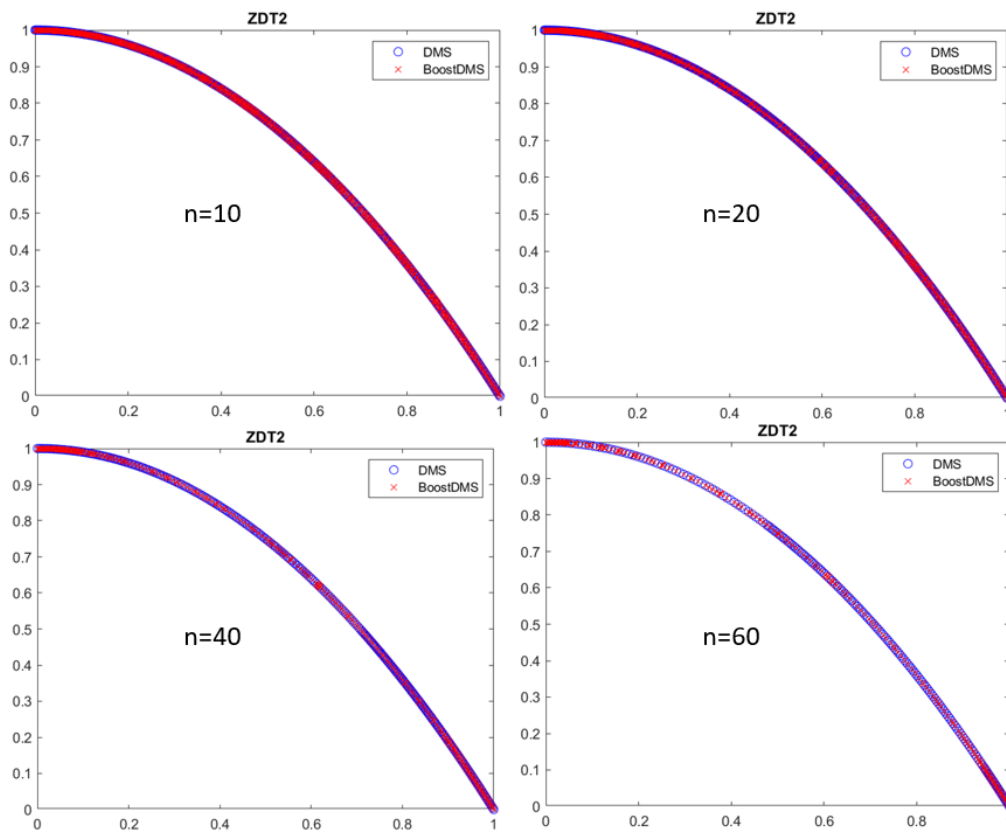


Figura A.7: Frentes de Pareto obtidas pelos algoritmos DMS e BoostDMS para o problema ZDT2, considerando 10, 20, 40 e 60 variáveis.

ZLT1 [12]

$$\begin{aligned} & \min_{\mathbf{x}} (f_1(\mathbf{x}), \dots, f_m(\mathbf{x})) \\ & \text{s.a. } -1000 \leq \mathbf{x} \leq 1000, \text{ com } \mathbf{x} = (x_1, \dots, x_n) \text{ e} \\ & f_j(\mathbf{x}) = (x_j - 1)^2 + \sum_{i=1}^{j-1} x_i^2 + \sum_{i=j+1}^n x_i^2, 1 \leq j \leq m, j \in \mathbb{N} \end{aligned}$$

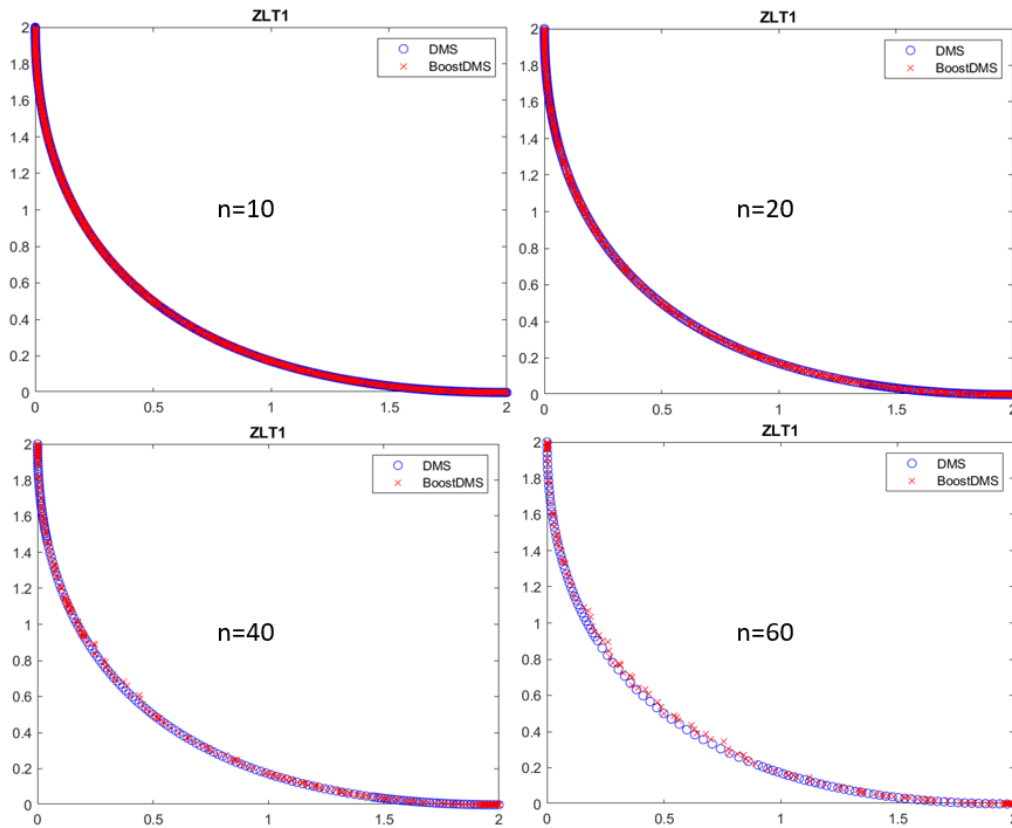


Figura A.8: Frentes de Pareto obtidas pelos algoritmos DMS e BoostDMS para o problema ZLT1, considerando 10, 20, 40 e 60 variáveis.

OTIMIZAÇÃO TOPOLOGICA EM DOMÍNIO DISCRETO

Tabela B.1: Valores de energia de deformação elástica e áreas de elementos barra obtidos pelo MMA em função de valores de peso.

w	$f_1[\text{N}\cdot\text{cm}]$	$f_2[\text{N}\cdot\text{cm}]$	$A_1[\text{cm}^2]$	$A_2[\text{cm}^2]$	$A_3[\text{cm}^2]$	$A_4[\text{cm}^2]$	$A_5[\text{cm}^2]$
0.01	251.1966	60.76813	39.10293	11.00349	413.0942	15.56128	584.2035
0.02	194.1635	61.587	53.67853	17.47856	401.7606	24.71842	568.1753
0.03	168.7632	62.22514	64.22374	23.12531	392.5988	32.70413	555.2186
0.04	153.5422	62.77154	72.6859	28.31691	384.5865	40.04615	543.8874
0.05	143.1003	63.26053	79.81902	33.20342	377.3223	46.95673	533.6143
0.06	135.3514	63.70965	86.00954	37.86451	370.5977	53.5485	524.1043
0.07	129.2969	64.12932	91.48681	42.34902	364.2874	59.89056	515.1802
0.08	124.39	64.52629	96.40007	46.68979	358.3089	66.02933	506.7253
0.09	120.3034	64.90527	100.8528	50.9103	352.6041	71.99803	498.6575
0.1	116.8272	65.26968	104.9204	55.02815	347.1304	77.82155	490.9166
0.11	113.8198	65.6221	108.6594	59.05699	341.8553	83.51919	483.4563
0.12	111.1819	65.96457	112.1137	63.00772	336.7531	89.10638	476.2408
0.13	108.8413	66.29868	115.3183	66.88925	331.8033	94.59569	469.2408
0.14	106.7441	66.62577	118.3013	70.70898	326.9893	99.9976	462.4327
0.15	104.8493	66.94692	121.0859	74.47318	322.2969	105.321	455.7966
0.16	103.1249	67.26307	123.6913	78.18721	317.7144	110.5734	449.316
0.17	101.5456	67.575	126.1338	81.85574	313.2317	115.7615	442.9765
0.18	100.0911	67.88341	128.4272	85.48285	308.8401	120.891	436.7659
0.19	98.74485	68.1889	130.5833	89.07216	304.5321	125.9671	430.6734
0.2	97.49313	68.49202	132.6125	92.6269	300.301	130.9942	424.6897
0.21	96.32464	68.79325	134.5238	96.14997	296.1408	135.9766	418.8064
0.22	95.22984	69.09302	136.3248	99.64399	292.0464	140.9179	413.016
0.23	94.20067	69.39175	138.0226	103.1114	288.0131	145.8215	407.3121
0.24	93.23022	69.6898	139.6233	106.5543	284.0366	150.6906	401.6885
0.25	92.31258	69.98752	141.1323	109.9748	280.1131	155.5279	396.1398
0.26	91.44263	70.28524	142.5544	113.3748	276.2391	160.3362	390.661
0.27	90.61589	70.58327	143.8941	116.756	272.4113	165.1179	385.2478
0.28	89.82848	70.8819	145.1553	120.12	268.6269	169.8754	379.8958
0.29	89.07695	71.18142	146.3414	123.4684	264.8831	174.6107	374.6013
0.3	88.35827	71.48211	147.4556	126.8027	261.1775	179.3261	369.3608

w	$f_1[\text{N}\cdot\text{cm}]$	$f_2[\text{N}\cdot\text{cm}]$	$A_1[\text{cm}^2]$	$A_2[\text{cm}^2]$	$A_3[\text{cm}^2]$	$A_4[\text{cm}^2]$	$A_5[\text{cm}^2]$
0.31	87.66975	71.78423	148.5007	130.1241	257.5076	184.0233	364.1708
0.32	87.009	72.08805	149.4795	133.4341	253.8714	188.7043	359.0284
0.33	86.37386	72.39383	150.3941	136.7339	250.2668	193.3709	353.9307
0.34	85.7624	72.70183	151.2468	140.0246	246.6918	198.0247	348.875
0.35	85.17289	73.01232	152.0394	143.3074	243.1448	202.6673	343.8587
0.36	84.60375	73.32554	152.7737	146.5835	239.624	207.3004	338.8794
0.37	84.05356	73.64177	153.4512	149.8539	236.1277	211.9254	333.935
0.38	83.52103	73.96127	154.0734	153.1197	232.6546	216.5439	329.0232
0.39	83.00497	74.28432	154.6415	156.3818	229.203	221.1573	324.142
0.4	82.50429	74.61119	155.1566	159.6414	225.7718	225.767	319.2895
0.41	82.01801	74.94217	155.6198	162.8993	222.3594	230.3744	314.4637
0.42	81.54521	75.27757	156.032	166.1566	218.9648	234.9809	309.663
0.43	81.08506	75.61767	156.3939	169.4142	215.5866	239.5878	304.8855
0.44	80.63676	75.96281	156.7062	172.673	212.2236	244.1965	300.1295
0.45	80.19961	76.31331	156.9695	175.934	208.8748	248.8083	295.3936
0.46	79.77293	76.66953	157.1843	179.1982	205.5391	253.4246	290.6761
0.47	79.35609	77.03182	157.3509	182.4665	202.2152	258.0466	285.9755
0.48	78.94853	77.40056	157.4697	185.7398	198.9024	262.6757	281.2904
0.49	78.54969	77.77617	157.5409	189.019	195.5994	267.3133	276.6193
0.5	78.15906	78.15906	157.5647	192.3052	192.3052	271.9607	271.9607
0.51	77.77617	78.54969	157.5409	195.5994	189.019	276.6193	267.3133
0.52	77.40056	78.94853	157.4697	198.9024	185.7398	281.2904	262.6757
0.53	77.03182	79.35609	157.3509	202.2152	182.4665	285.9755	258.0466
0.54	76.66953	79.77293	157.1843	205.5391	179.1982	290.6761	253.4246
0.55	76.31331	80.19961	156.9695	208.8748	175.934	295.3936	248.8083
0.56	75.96281	80.63676	156.7062	212.2236	172.673	300.1295	244.1965
0.57	75.61767	81.08506	156.3939	215.5866	169.4142	304.8855	239.5878
0.58	75.27757	81.54521	156.032	218.9648	166.1566	309.663	234.9809
0.59	74.94217	82.01801	155.6198	222.3594	162.8993	314.4637	230.3744
0.6	74.61119	82.50429	155.1566	225.7718	159.6414	319.2895	225.767
0.61	74.28432	83.00497	154.6415	229.203	156.3818	324.142	221.1573
0.62	73.96127	83.52103	154.0734	232.6546	153.1197	329.0232	216.5439
0.63	73.64177	84.05356	153.4512	236.1277	149.8539	333.935	211.9254
0.64	73.32554	84.60375	152.7737	239.624	146.5835	338.8794	207.3004
0.65	73.01232	85.17289	152.0394	243.1448	143.3074	343.8587	202.6673
0.66	72.70183	85.7624	151.2468	246.6918	140.0246	348.875	198.0247
0.67	72.39383	86.37386	150.3941	250.2668	136.7339	353.9307	193.3709
0.68	72.08805	87.009	149.4795	253.8714	133.4341	359.0284	188.7043
0.69	71.78423	87.66975	148.5007	257.5076	130.1241	364.1708	184.0233
0.7	71.48211	88.35827	147.4556	261.1775	126.8027	369.3608	179.3261
0.71	71.18142	89.07695	146.3414	264.8831	123.4684	374.6013	174.6107
0.72	70.8819	89.82848	145.1553	268.6269	120.12	379.8958	169.8754
0.73	70.58327	90.61589	143.8941	272.4113	116.756	385.2478	165.1179
0.74	70.28524	91.44263	142.5544	276.2391	113.3748	390.661	160.3362
0.75	69.98752	92.31258	141.1323	280.1131	109.9748	396.1398	155.5279
0.76	69.6898	93.23022	139.6233	284.0366	106.5543	401.6885	150.6906
0.77	69.39175	94.20067	138.0226	288.0131	103.1114	407.3121	145.8215
0.78	69.09302	95.22984	136.3248	292.0464	99.64399	413.016	140.9179
0.79	68.79325	96.32464	134.5238	296.1408	96.14997	418.8064	135.9766

APÊNDICE B. OTIMIZAÇÃO TOPOLÓGICA EM DOMÍNIO DISCRETO

w	$f_1[\text{N}\cdot\text{cm}]$	$f_2[\text{N}\cdot\text{cm}]$	$A_1[\text{cm}^2]$	$A_2[\text{cm}^2]$	$A_3[\text{cm}^2]$	$A_4[\text{cm}^2]$	$A_5[\text{cm}^2]$
0.8	68.49202	97.49313	132.6125	300.301	92.6269	424.6897	130.9942
0.81	68.1889	98.74485	130.5833	304.5321	89.07216	430.6734	125.9671
0.82	67.88341	100.0911	128.4272	308.8401	85.48285	436.7659	120.891
0.83	67.575	101.5456	126.1338	313.2317	81.85574	442.9765	115.7615
0.84	67.26307	103.1249	123.6913	317.7144	78.18721	449.316	110.5734
0.85	66.94692	104.8493	121.0859	322.2969	74.47318	455.7966	105.321
0.86	66.62577	106.7441	118.3013	326.9893	70.70898	462.4327	99.9976
0.87	66.29868	108.8413	115.3183	331.8033	66.88925	469.2408	94.59569
0.88	65.96457	111.1819	112.1137	336.7531	63.00772	476.2408	89.10638
0.89	65.6221	113.8198	108.6594	341.8553	59.05699	483.4563	83.51919
0.9	65.26968	116.8272	104.9204	347.1304	55.02815	490.9166	77.82155
0.91	64.90527	120.3034	100.8528	352.6041	50.9103	498.6575	71.99803
0.92	64.52629	124.39	96.40007	358.3089	46.68979	506.7253	66.02933
0.93	64.12932	129.2969	91.48681	364.2874	42.34902	515.1802	59.89056
0.94	63.70965	135.3514	86.00954	370.5977	37.86451	524.1043	53.5485
0.95	63.26053	143.1003	79.81902	377.3223	33.20342	533.6143	46.95673
0.96	62.77154	153.5422	72.6859	384.5865	28.31691	543.8874	40.04615
0.97	62.22514	168.7632	64.22374	392.5988	23.12531	555.2186	32.70413
0.98	61.587	194.1635	53.67853	401.7606	17.47856	568.1753	24.71842
0.99	60.76813	251.1966	39.10293	413.0942	11.00349	584.2035	15.56128

BIBLIOGRAFIA

- [1] J. M. Lourenço. *The NOVAtesis L^AT_EX Template User's Manual*. NOVA University Lisbon. 2021. URL: <https://github.com/joaomlourenco/novathesis/raw/main/template.pdf> (ver p. i).
- [2] M. P. Bendsøe e N. Kikuchi. «Generating optimal topologies in structural design using a homogenization method». Em: *Comput. Methods Appl. Mech. Eng.* 71.2 (1988). DOI: [https://doi.org/10.1016/0045-7825\(88\)90086-2](https://doi.org/10.1016/0045-7825(88)90086-2) (ver pp. 2, 5).
- [3] K. Svanberg. «The method of moving asymptotes — a new method for structural optimization». Em: *Int. J. Numer. Methods Eng.* 24.2 (1987). DOI: <https://doi.org/10.1002/nme.1620240207> (ver pp. 2, 6, 28).
- [4] M. P. Bendsøe e O. Sigmund. *Topology Optimization: Theory, Methods, and Applications*. Berlin: Springer, 2003. DOI: [10.1007/978-3-662-05086-6](https://doi.org/10.1007/978-3-662-05086-6) (ver pp. 3, 10, 21, 27, 28).
- [5] R. Marler e J. Arora. «Survey of multi-objective optimization methods for engineering». Em: *Struct. Multidiscip. Optim.* (2004). DOI: [10.1007/s00158-003-0368-6](https://doi.org/10.1007/s00158-003-0368-6) (ver pp. 3–5).
- [6] J. Nocedal e S. Wright. *Numerical Optimization*. Springer Series in Operations Research and Financial Engineering. Springer New York, 2006. DOI: [10.1007/978-0-387-40065-5](https://doi.org/10.1007/978-0-387-40065-5) (ver p. 5).
- [7] R. L. Burden e J. D. Faires. *Numerical Analysis*. 9^a ed. Brooks/Cole, 2010 (ver p. 6).
- [8] A. L. Custódio, J. F. A. Madeira, A. I. F. Vaz e L. N. Vicente. «Direct Multisearch for multiobjective optimization». Em: *SIAM J. Optim.* (2011). DOI: [10.1137/10079731X](https://doi.org/10.1137/10079731X) (ver pp. 7, 17).
- [9] R. Andreani, A. L. Custódio e M. Raydan. «Using first-order information in Direct Multisearch for multiobjective optimization». Em: *Optim. Methods Software* 37 (2022). DOI: [10.1080/10556788.2021.1932536](https://doi.org/10.1080/10556788.2021.1932536) (ver p. 7).
- [10] C. Davis. «Theory of positive linear dependence». Em: *Am. J. Math.* 76.4 (1954). DOI: [10.2307/2372648](https://doi.org/10.2307/2372648) (ver p. 7).

- [11] C. P. Brás e A. L. Custódio. «On the use of polynomial models in multiobjective directional direct search». Em: *Comput. Optim. Appl.* 77 (2020). DOI: [10.1007/s10589-020-00233-8](https://doi.org/10.1007/s10589-020-00233-8) (ver p. 8).
- [12] S. Huband, P. Hingston, L. Barone e L. While. «A review of multiobjective test problems and a scalable test problem toolkit». Em: *IEEE Trans. Evol. Comput* 10.5 (2006). DOI: [10.1109/TEVC.2005.861417](https://doi.org/10.1109/TEVC.2005.861417) (ver pp. 15, 47, 52, 54).
- [13] K. Deb, L. Thiele, M. Laumanns e E. Zitzler. «Scalable multi-objective optimization test problems». Em: *Proceedings of the 2002 Congress on Evolutionary Computation. CEC'02 (Cat. No.02TH8600)*. Vol. 1. 2002. DOI: [10.1109/CEC.2002.1007032](https://doi.org/10.1109/CEC.2002.1007032) (ver pp. 15, 48–50).
- [14] E. Zitzler, K. Deb e L. Thiele. «Comparison of multiobjective evolutionary algorithms: empirical results». Em: *Evol. Comput* 8.2 (2000). DOI: [10.1162/106365600568202](https://doi.org/10.1162/106365600568202) (ver pp. 15, 53).
- [15] Y. Jin, M. Olhofer e B. Sendhoff. «Dynamic Weighted Aggregation for evolutionary multi-objective optimization: why does it work and how?» Em: *Proceedings of the 3rd Annual Conference on Genetic and Evolutionary Computation*. Morgan Kaufmann Publishers Inc., 2001 (ver pp. 15, 51).
- [16] J. J. Moré e S. M. Wild. «Benchmarking derivative-free optimization algorithms». Em: *SIAM J. Optim.* 20.1 (2009). DOI: [10.1137/080724083](https://doi.org/10.1137/080724083) (ver p. 17).
- [17] S. Bandyopadhyay, S. K. Pal e B. Aruna. «Multiobjective GAs, quantitative indices, and pattern classification». Em: *IEEE Trans. Syst. Man Cybern. - Part B: Cybern.* 34 (2004) (ver p. 17).
- [18] E. Zitzler, L. Thiele, M. Laumanns, C. M. Fonseca e V. G. da Fonseca. «Performance assessment of multiobjective optimizers: An analysis and review». Em: *IEEE T. Evolut. Comput.* 7 (2003) (ver p. 17).
- [19] Y. Li, Q. Yang, T. Chang, T. Qin e F. Wu. «Multi-load cases topological optimization by weighted sum method based on load case severity degree and ideality». Em: *Adv. Mech. Eng.* 55 (2020). DOI: [10.1177/1687814020947510](https://doi.org/10.1177/1687814020947510) (ver p. 20).
- [20] Y. Sato, K. Izui, T. Yamada e S. Nishiwaki. «Pareto frontier exploration in multiobjective topology optimization using adaptive weighting and point selection schemes». Em: *Struct. Multidiscip. Optim.* 55 (2017). DOI: [10.1007/s00158-016-1499-x](https://doi.org/10.1007/s00158-016-1499-x) (ver p. 20).
- [21] E. Andreassen, A. Clausen, M. Schevenels, B. S. Lazarov e O. Sigmund. «Efficient topology optimization in MATLAB using 88 lines of code». Em: *Struct. Multidiscip. Optim.* 43 (2011). DOI: [10.1007/s00158-010-0594-7](https://doi.org/10.1007/s00158-010-0594-7) (ver p. 21).
- [22] O. Sigmund. «A 99 line topology optimization code written in Matlab». Em: *Struct. Multidiscip. Optim.* 21.3 (2001). DOI: [10.1007/s001580050176](https://doi.org/10.1007/s001580050176) (ver p. 21).

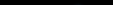
- [23] O. Sigmund e K. Maute. «Sensitivity filtering from a continuum mechanics perspective». Em: *Struct. Multidiscip. Optim.* (2012) (ver p. 23).
- [24] F. P. Beer, E. R. J. Jr., J. T. DeWolf e D. F. Mazurek. *Mechanics of Materials*. 8th. McGraw-Hill Education, 2021 (ver p. 30).



2025

Optimização multiobjetivo com e sem derivadas aplicada a estruturas mecânicas multicargas

Tiago Neves



NOVA

UNIVERSITY OF LISBON

NOVA FCT

NOVA FCSH

NOVA FEF

NOVA FEP

NOVA FEUP

NOVA FGV

NOVA FIM

NOVA FISI

NOVA FQ

NOVA FARM

NOVA FEN

NOVA FGA

NOVA FGE

NOVA FGP

NOVA FIM

NOVA FISI

NOVA FQ

NOVA FARM

NOVA FEN

NOVA FGA

NOVA FGE

NOVA FGP

NOVA FIM

NOVA FISI

NOVA FQ

NOVA FARM

NOVA FEN

NOVA FGA

NOVA FGE

NOVA FGP

NOVA FIM

NOVA FISI

NOVA FQ

NOVA FARM

NOVA FEN

NOVA FGA

NOVA FGE

NOVA FGP

NOVA FIM

NOVA FISI

NOVA FQ

NOVA FARM

NOVA FEN

NOVA FGA

NOVA FGE

NOVA FGP