

UNIVERSIDADE DE SÃO PAULO

Escola de Engenharia de São Carlos

Aplicação de conceitos de dimensionamento de vigas no projeto de estruturas contínuas

Tarcísio Ladeia de Oliveira

**UNIVERSIDADE DE SÃO PAULO
ESCOLA DE ENGENHARIA DE SÃO CARLOS**

Tarcísio Ladeia de Oliveira

**Aplicação de conceitos de dimensionamento de vigas no
projeto de estruturas contínuas**

São Carlos

2022

Tarcísio Ladeia de Oliveira

**Aplicação de conceitos de dimensionamento de vigas no
projeto de estruturas contínuas**

Dissertação apresentada à Escola de Engenharia de São Carlos da Universidade de São Paulo, para obtenção do título de Mestre em Ciências - Programa de Pós-Graduação em Engenharia Mecânica.

Área de concentração: Projeto, Materiais e Manufatura

Orientador: Prof. Dr. Jonas de Carvalho

VERSÃO CORRIGIDA

São Carlos

2022

AUTORIZO A REPRODUÇÃO TOTAL OU PARCIAL DESTE TRABALHO,
POR QUALQUER MEIO CONVENCIONAL OU ELETRÔNICO, PARA FINS
DE ESTUDO E PESQUISA, DESDE QUE CITADA A FONTE.

Ficha catalográfica elaborada pela Biblioteca Prof. Dr. Sérgio Rodrigues Fontes da
EESC/USP com os dados inseridos pelo(a) autor(a).

048a Oliveira, Tarcísio Ladeia de
Aplicação de conceitos de dimensionamento de
vigas no projeto de estruturas contínuas / Tarcísio
Ladeia de Oliveira; orientador Jonas de Carvalho. São
Carlos, 2022.

Dissertação (Mestrado) - Programa de
Pós-Graduação em Engenharia Mecânica e Área de
Concentração em Projeto, Materiais e Manufatura --
Escola de Engenharia de São Carlos da Universidade de
São Paulo, 2022.

1. Projeto mecânico. 2. Estruturas. 3.
Otimização topológica. 4. Automação de projeto. 5.
Método dos elementos finitos. I. Título.

FOLHA DE JULGAMENTO

Candidato: Engenheiro **TARCÍSIO LADEIA DE OLIVEIRA**.

Título da dissertação: "Aplicação de conceitos de dimensionamento de vigas no projeto de estruturas contínuas".

Data da defesa: 16/03/2022.

Comissão Julgadora

Resultado

Prof. Titular **Jonas de Carvalho**

Aprovado

(Orientador)

(Escola de Engenharia de São Carlos – EESC/USP)

Prof. Dr. **Mariano Eduardo Moreno**

Aprovado

(Universidade Federal de São Carlos/UFSCar)

Prof. Dr. **Gustavo Assis da Silva**

Aprovado

(Universidade Federal de Pelotas/UFPel)

Coordenador do Programa de Pós-Graduação em Engenharia Mecânica:

Prof. Associado **Adriano Almeida Gonçalves Siqueira**

Presidente da Comissão de Pós-Graduação:

Prof. Titular **Murilo Araujo Romero**

Ao meu velho amigo Raul.

AGRADECIMENTOS

Agradeço ao meu orientador, o professor Jonas de Carvalho, pela dedicação e paciência, visto que com frequência sou um pouco independente demais.

Agradeço também aos professores João Batista de Paiva e Ernesto Massaroppi Junior por terem me fornecido durante a graduação uma parte importante dos conceitos utilizados nesta pesquisa, e à comunidade de desenvolvimento de jogos digitais por ter fornecido a outra parte.

Agradeço à minha família e meus amigos pelo apoio, especialmente dado à ocorrência da pandemia durante a maior parte do Mestrado.

E, por fim, agradeço ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) pela bolsa concedida (processo 131344/2020-0), o que garantiu meu ingresso e continuidade no programa.

*“A perfeição é alcançada,
não quando não há mais o que adicionar,
mas quando não há mais o que retirar.”
Antoine de Saint-Exupéry.*

RESUMO

OLIVEIRA, T.L. **Aplicação de conceitos de dimensionamento de vigas no projeto de estruturas contínuas**. 2022. 122p. Dissertação (Mestrado) - Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2022.

Com o desenvolvimento de ferramentas computacionais para o projeto e análise de estruturas, o uso de modelagem por vigas como método de projeto e análise passou a ser limitado ao desenvolvimento de estruturas treliçadas, também com auxílio computacional. Porém, nesses casos comumente se assume uma seção transversal constante ao longo de uma viga ou para todas as vigas, embora as modelagens matemáticas de vigas admitam seções transversais variáveis. Esta pesquisa analisa a viabilidade de utilizar modelagens computacionais de vigas para auxiliar e automatizar o projeto de estruturas contínuas, admitindo a possibilidade de vigas de seção transversal variável para possibilitar isso, com ênfase em estruturas bidimensionais. O algoritmo foi primeiramente testado de forma discretizada em uma implementação em Python, na qual apresentou resultados promissores. O algoritmo foi posteriormente refinado e implementado em um ambiente contínuo em C++, com o auxílio de geometria sólida construtiva. Também foi avaliado o uso de otimização topológica para otimizar a distribuição de material das geometrias resultantes, assim como a possibilidade de reduzir o tempo de construção na fabricação por manufatura aditiva. Os resultados mostram que esse algoritmo é capaz de criar automaticamente estruturas capazes de resistir aos esforços de projeto, especialmente quando em conjunto com métodos de otimização estrutural. Porém, as diferenças entre os critérios de tensão utilizados para cada etapa não deixam claro qual seria a melhor forma de selecioná-los.

Palavras-chave: Projeto mecânico. Estruturas. Otimização topológica. Automação de projeto. Método dos elementos finitos.

ABSTRACT

OLIVEIRA, T.L. **Applying beam sizing concepts to the design of continuum structures..** 2022. 122p. Dissertação (Mestrado) - Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2022.

Due to the development of computational tools for structural analysis and design, the use of structural modeling by beam segments as a design and analysis method became limited to the development of trussed structures, also aided by computational methods. However, in those cases a constant cross-section is commonly used throughout a beam or set of beams, even though mathematical beam models admit variable cross-sections. This research analyses the viability of using computational beam models as a way to assist and automate the design of continuum structures, admitting the possibility of beams having variable cross-sections as a way to enable that, focusing on two-dimensional structures. The algorithm was first tested in a discretized environment by means of a Python implementation, in which it displayed promising results. Afterwards, the algorithm was refined and implemented in a continuum environment in C++, with the aid of constructive solid geometry. The use of topology optimization was also evaluated as a way to optimize material distribution in the resulting geometries, as well as the possibility of reducing build time in additive manufacturing. The results show that the algorithm is capable of automatically generating structures that are capable of resisting design loads, especially when followed by structural optimization methods. However, due to the differences between the stress criteria used for those two stages, it's not clear what would be the best way to define them.

Keywords: Mechanical design. Structures. Topology optimization. Design automation. Finite element method.

LISTA DE FIGURAS

- Figura 1 – Diagrama de blocos do funcionamento do GCMMA. 48
- Figura 2 – Representação esquemática do algoritmo a ser implementado: (a) Criação da viga inicial; (b) Modificação da seção transversal da viga de acordo com os resultados da análise de elementos finitos; (c) Otimização topológica do resultado. 57
- Figura 3 – Problemas resolvidos pelos Casos 1 a 4: (a) Caso 1; (b) Caso 2; (c) Caso 3; (d) Caso 4. 64
- Figura 4 – Resultados do Caso 1. A distribuição de tensão vai de maior nível de tensão von Mises (em preto) para menor nível (em branco): (a) Viga resultante (distribuição de tensão); (b) Topologia inicial; (c) Topologia final; (d) Topologia final (distribuição de tensão). 66
- Figura 5 – Comparação das topologias resultantes com e sem pré-dimensionamento no Caso 1: (a) Com pré-dimensionamento. A tensão von Mises máxima foi de 30,7 MPa. A flexibilidade foi de 5,412 N · mm; (b) Sem pré-dimensionamento. A tensão von Mises máxima foi de 18,7 MPa. A flexibilidade foi de 4,172 N · mm. 67
- Figura 6 – Análise de elementos finitos realizada sobre modelo CAD aproximado da topologia resultante do Caso 1: (a) Tensão von Mises (máximo: 54,622 MPa, mínimo: 0,354 MPa); (b) Tensão normal (eixo X) (máximo: 15,168 MPa, mínimo: -15,285 MPa); (c) Tensão normal (eixo Y) (máximo: 52,408 MPa, mínimo: -48,209 MPa); (d) Tensão de cisalhamento (plano XY) (máximo: 0,381 MPa, mínimo: -17,585 MPa). 67
- Figura 7 – Caso 1 com diferentes intensidades de carga: (a) 15 kN. A tensão von Mises máxima foi de 30,7 MPa. A flexibilidade foi de 5,412 N · mm; (b) 12 kN. A tensão von Mises máxima foi de 32,5 MPa. A flexibilidade foi de 5,119 N · mm; (c) 9 kN. A tensão von Mises máxima foi de 30,5 MPa. A flexibilidade foi 5,155 N · mm; (d) 6 kN. A tensão von Mises máxima foi de 32,7 MPa. A flexibilidade foi de 5,602 N · mm. 68
- Figura 8 – Resultados do Caso 2. A distribuição de tensão vai de maior nível de tensão von Mises (em preto) para menor nível (em branco): (a) Viga resultante (distribuição de tensão); (b) Topologia inicial; (c) Topologia final; (d) Topologia final (distribuição de tensão). 69

Figura 9 – Análise de elementos finitos realizada sobre modelo CAD aproximado da topologia resultante do Caso 2: (a) Tensão von Mises (máximo: 96,219 MPa, mínimo: 0,488 MPa); (b) Tensão normal (eixo X) (máximo: 18,235 MPa, mínimo: -15,747 MPa); (c) Tensão normal (eixo Y) (máximo: 93,507 MPa, mínimo: -88,592 MPa); (d) Tensão de cisalhamento (plano XY) (máximo: 30,331 MPa, mínimo: -34,322 MPa).	69
Figura 10 – Comparação das topologias resultantes com e sem pré-dimensionamento no Caso 2: (a) Com pré-dimensionamento. A tensão von Mises máxima foi de 43,4 MPa. A flexibilidade foi de 10,733 N · mm; (b) Sem pré-dimensionamento. A tensão von Mises máxima foi de 31,9 MPa. A flexibilidade foi de 8,294 N · mm.	70
Figura 11 – Caso 2 com diferentes intensidades de carga: (a) 50 kN. A tensão von Mises máxima foi de 43,4 MPa. A flexibilidade foi de 10,733 N · mm; (b) 40 kN. A tensão von Mises máxima foi de 34,6 MPa. A flexibilidade foi de 8,282 N · mm; (c) 30 kN. A tensão von Mises máxima foi de 30,7 MPa. A flexibilidade foi de 7,531 N · mm; (d) 20 kN. A tensão von Mises máxima foi de 33,9 MPa. A flexibilidade foi de 7,844 N · mm.	70
Figura 12 – Resultados do Caso 3. A distribuição de tensão vai de maior nível de tensão von Mises (em preto) para menor nível (em branco): (a) Viga resultante (distribuição de tensão); (b) Topologia inicial; (c) Topologia final; (d) Topologia final (distribuição de tensão).	71
Figura 13 – Análise de elementos finitos realizada sobre modelo CAD aproximado da topologia resultante do Caso 3: (a) Tensão von Mises (máximo: 347,74 MPa, mínimo: 0,001 MPa); (b) Tensão normal (eixo X) (máximo: 53,918 MPa, mínimo: -71,814 MPa); (c) Tensão normal (eixo Y) (máximo: 338,79 MPa, mínimo: -334,88 MPa); (d) Tensão de cisalhamento (plano XY) (máximo: 3,519 MPa, mínimo: -111,95 MPa).	72
Figura 14 – Análise de elementos finitos realizada sobre modelo CAD aproximado da topologia presente na Figura 12a. A análise também utilizou o método linear de elementos finitos. É possível notar a ausência de concentração de tensão significativa na região de interseção das duas vigas diagonais.	72
Figura 15 – Resultados do Caso 4. A distribuição de tensão vai de maior nível de tensão von Mises (em vermelho) para menor nível (em azul): (a) Viga resultante (distribuição de tensão); (b) Topologia inicial; (c) Topologia final; (d) Topologia final (distribuição de tensão).	74

Figura 16 – Comparação das topologias resultantes com e sem pré-dimensionamento no Caso 4: (a) Com pré-dimensionamento. A tensão von Mises máxima foi de 176,1 MPa. A flexibilidade foi de 261,659 N · mm; (b) Sem pré-dimensionamento. A tensão von Mises máxima foi de 160,4 MPa. A flexibilidade foi de 168,892 N · mm.	74
Figura 17 – Diagrama de blocos do funcionamento da implementação em ambiente contínuo. O fluxo do programa, assim como outros parâmetros, é definido pelo arquivo de entrada.	76
Figura 18 – Representação esquemática das possíveis interfaces entre vigas. As linhas contínuas representam o contorno final, as tracejadas delimitam a região comum entre as vigas, e as linhas traço-ponto representam a linha central das vigas: (a) Interseção; (b) Conexão; (c) Sobreposição.	77
Figura 19 – Esquematização do processo de reposicionamento da viga redimensionada.	81
Figura 20 – Representação esquemática de um grafo de visibilidade.	82
Figura 21 – Problemas resolvidos pelos Casos 1 a 3: (a) Caso 1; (b) Caso 2; (c) Caso 3.	89
Figura 22 – Análise de elementos finitos das vigas redimensionadas para os Casos 1.1 e 1.2. Tensão equivalente de von Mises (MPa): (a) Caso 1.1, tensão máxima de 13,9 MPa; (b) Caso 1.2, tensão máxima de 15,6 MPa.	91
Figura 23 – Análise de elementos finitos das vigas redimensionadas para o Caso 2. Tensão equivalente de von Mises (MPa). Tensão máxima de 15,7 MPa.	92
Figura 24 – Análise de elementos finitos das vigas redimensionadas para os Casos 3.1 e 3.2. Tensão equivalente de von Mises (MPa): (a) Caso 3.1, tensão máxima de 32,2 MPa; (b) Caso 3.2, tensão máxima de 17,0 MPa.	92
Figura 25 – Vigas geradas pelo algoritmo de <i>pathfinding</i> e utilizadas para o pré-dimensionamento dos Casos 3.1 e 3.2: (a) Caso 3.1; (b) Caso 3.2.	93
Figura 26 – Resultados de otimização topológica para o Caso 1.1 com carga de intensidade 2000 N. Representação dos níveis de tensão de von Mises, em MPa: (a) Otimização topológica pura, com volume final de 264.635 mm ³ (8,3% do volume inicial); (b) Utilizando pré-dimensionamento, com volume final de 614.733 mm ³ (22,8% do volume inicial).	94
Figura 27 – Caso 1 utilizando pré-dimensionamento e tensão de von Mises máxima de 10,0 MPa. Representação dos níveis de tensão de von Mises, em MPa: (a) Flexão com carga de 2000 N, tensão máxima de 12,7 MPa; (b) Compressão com carga de 5000 N, tensão máxima de 12,4 MPa (elemento de tamanho 3 mm).	95

Figura 28 – Resultados da otimização topológica para o Caso 2. Representação dos níveis de tensão de von Mises, em MPa (a) Otimização topológica pura, com volume final de 593.822 mm^3 (18,6% do volume inicial). (b) Utilizando pré-dimensionamento, com volume final de 623.840 mm^3 (19,5% do volume inicial).	96
Figura 29 – Resultados de otimização topológica para o Caso 3.1. Representação dos níveis de tensão de von Mises, em MPa: (a) Otimização topológica pura, com volume final de 384.313 mm^3 (16,0% do volume inicial); (b) Utilizando pré-dimensionamento, com volume final de 611.341 mm^3 (22,7% do volume inicial).	97
Figura 30 – Resultados de otimização topológica para o Caso 3.1 utilizando coeficiente de sobredimensionamento. Representação dos níveis de tensão de von Mises, em MPa: (a) Coeficiente de 1,15, com volume final de 507.666 mm^3 (21,2% do volume inicial); (b) Coeficiente de 1,70, com volume final de 405.546 mm^3 (16,9% do volume inicial).	98
Figura 31 – Resultados de otimização topológica para o Caso 3.2 utilizando coeficiente de sobredimensionamento. Representação dos níveis de tensão de von Mises, em MPa. Volume final de 597.486 mm^3 (24,9% do volume inicial).	98

LISTA DE TABELAS

Tabela 1 – Parâmetros de material comuns a todos os testes.	65
Tabela 2 – Parâmetros de otimização comuns a todos os testes.	65
Tabela 3 – Parâmetros específicos de cada teste.	65
Tabela 4 – Comparação do tempo de resolução para o Caso 1 com e sem pré-dimensionamento.	68
Tabela 5 – Comparação do tempo de resolução para o Caso 2 com e sem pré-dimensionamento.	69
Tabela 6 – Comparação do tempo de resolução para o Caso 4 com e sem pré-dimensionamento.	73
Tabela 7 – Parâmetros comuns aos quatro testes analisados por elementos finitos. O material é uma aproximação do PLA extrudado no processo de fabricação por filamento fundido.	90
Tabela 8 – Parâmetros específicos de cada teste.	90
Tabela 9 – Parâmetros para a comparação de tempo de resolução. O Caso 1.2 não é utilizado para essa análise.	90
Tabela 10 – Resultados dos Casos 1.1, 2, 3.1 e 3.2, com e sem pré-dimensionamento, em termos de tempo de resolução e volume final.	99
Tabela 11 – Comparação do tempo de resolução e volume final para os Casos 1.1, 2, 3.1 e 3.2, com e sem pré-dimensionamento.	99

LISTA DE SÍMBOLOS

$M(x)$	Momento de flexão
$F_c(x)$	Força cortante
$f_c(x)$	Força cortante distribuída
$E(x)$	Módulo de elasticidade
$I(x)$	Segundo momento de área (momento de inércia de área)
$w(x)$	Deformação vertical
G	Módulo de cisalhamento
A	Área da seção transversal
κ	Coefficiente de cisalhamento de Timoshenko
φ	Ângulo de curvatura da viga
Ω	Domínio da topologia
Γ	Contorno do domínio
Γ_T	Região do contorno sobre a qual são aplicadas cargas de contato
b	Campo de cargas volumétricas
t	Campo de cargas de contato
u	Campo de deslocamento
$l(u)$	Trabalho externo
v	Campo de deslocamento virtual
$a(u, v)$	Trabalho interno virtual para uma condição de equilíbrio com campo de deslocamento u e campo de deslocamento virtual v
$E_{ijkl}(x)$	Tensor constitutivo
$\varepsilon_{ij}(u)$	Tensor de deformação para um campo de deslocamento u
E_{ad}	Conjunto de tensores constitutivos admissíveis
U	Conjunto dos campos de deslocamento admissíveis

N	Número de elementos
\mathbf{b}_e	Vetor elementar de cargas volumétricas
\mathbf{t}_e	Vetor elementar de cargas de contato
\mathbf{u}_e	Vetor elementar de deslocamentos
$\Phi_e(x)$	Função de forma elementar
$u_e(x)$	Aproximação do campo de deslocamentos elementar, utilizando função de forma $\Phi_e(x)$
Γ_e	Região do contorno externo presente no elemento
\mathbf{f}_e	Vetor elementar de cargas
\mathbf{f}	Vetor global de cargas
\mathbf{u}_e	Vetor elementar de deslocamentos
\mathbf{u}	Vetor global de deslocamentos
$c(\mathbf{u})$	Trabalho externo calculado pelo método dos elementos finitos para um vetor global de deslocamentos \mathbf{u}
\mathbf{L}_e	Matriz de posicionamento elementar
$\boldsymbol{\varepsilon}$	Vetor global de deformação
$\boldsymbol{\varepsilon}_e$	Vetor elementar de deformação
∇_S	Operador diferencial para o cálculo das deformações
\mathbf{B}_e	Matriz gradiente elementar
\mathbf{v}_e	Vetor elementar de deslocamentos virtuais
\mathbf{v}	Vetor global de deslocamentos virtuais
\mathbf{K}_e	Matriz elementar de rigidez
\mathbf{K}	Matriz global de rigidez
$\rho(x)$	Campo de densidade
p	Potência de penalização
E_{ijkl}^0	Tensor constitutivo original do elemento
ρ_e	Densidade elementar

\mathbf{K}_e^0	Matriz elementar de rigidez original do elemento
ν	Coefficiente de Poisson
ρ_{min}	Densidade elementar mínima
\mathbf{K}_e^{min}	Matriz elementar de rigidez mínima
\hat{H}_{ki}	Fator de ponderação do filtro de densidade linear
r_{min}	Raio do filtro de densidade
d_i	Variável de projeto associada a um elemento i
$\bar{\mathbf{u}}$	Vetor global de deslocamentos virtuais para o método adjunto
\bar{V}	Volume máximo
\mathcal{L}	Função de Lagrange
Λ	Multiplicador de Lagrange para a restrição não linear (no caso, restrição de volume)
λ^+	Multiplicador de Lagrange para a restrição superior da densidade
λ^-	Multiplicador de Lagrange para a restrição inferior da densidade
μ	Número da iteração
ζ	Passo máximo
η	Parâmetro de ajuste
B^μ	Energia relativa de deformação
\mathbf{x}	Vetor de variáveis de otimização
n	Número de variáveis de otimização
m	Número de restrições
$f_0(\mathbf{d})$	Função objetivo
$f_i(\mathbf{d})$	Função de restrição (para $i > 0$)
\hat{f}_i	Valor máximo imposto à função de restrição $f_i(\mathbf{x})$
d_j^{min}	Valor mínimo de d_j
d_j^{max}	Valor máximo de d_j

p_{ij}^μ	Coeficiente superior da aproximação convexa
q_{ij}^μ	Coeficiente inferior da aproximação convexa
$f_i^\mu(\mathbf{d})$	Estimativa da função f_i para a iteração μ
r_i^μ	Diferença entre $f_i(\mathbf{d}^\mu)$ e a aproximação convexa no ponto \mathbf{d}^μ
L_j^μ	Assíntota inferior
U_j^μ	Assíntota superior
α_j	Valor mínimo de d_j para o subproblema convexo
β_j	Valor máximo de d_j para o subproblema convexo
χ	Número da iteração interna
A_i, C_i, D_i	Parâmetros de otimização
\mathbf{y}, z	Variáveis artificiais de otimização
$\frac{\partial f_i}{\partial d_j}^+$	$\max \left\{ \frac{\partial f_i}{\partial d_j}, 0 \right\}$
$\frac{\partial f_i}{\partial d_j}^-$	$\max \left\{ -\frac{\partial f_i}{\partial d_j}, 0 \right\}$
$\xi_i^{\mu, \chi}$	Parâmetro de atualização da estimativa da próxima iteração
γ_j^μ	Parâmetro de atualização das assíntotas
σ_e	Tensão de von Mises no elemento e
$(\sigma_e)_{max}$	Tensão de von Mises máxima da estrutura
RR_i	Razão de rejeição da iteração i
ER	Taxa de evolução
IR	Razão de inclusão
$\phi(\mathbf{x})$	Função que delimita o contorno da geometria implicitamente
C	Constante que define a geometria
$D \setminus \Omega$	Região de espaço vazio
f	Funcional que mapeia a geometria para um modelo mecânico
$H(\phi)$	Função de Heaviside aplicada a ϕ
$R(\Omega, \mathbf{u}(\Omega))$	Função resposta

N_i	Função de forma do contorno
s_i	Parâmetro de otimização do contorno
d_S	Gradiente de forma
d_τ	Gradiente topológico
$B(r)$	Esfera de raio r
v_e	Volume do elemento
σ_{max}^{VM}	Tensão de von Mises máxima de projeto
σ_g	Medida de tensão global máxima
P	Potência da norma-p
$g(\boldsymbol{\rho})$	Função restrição
\mathbf{V}	Matriz de cálculo da tensão de von Mises
$\boldsymbol{\lambda}$	Vetor global de deslocamentos virtuais para o método adjunto para restrições de tensão global
σ_{max}	Tensão normal máxima
τ_{max}	Tensão de cisalhamento máxima
σ	Tensão normal
F_n	Força normal
A	Área da seção transversal
τ	Tensão de cisalhamento
F_c	Força cortante
Q	Primeiro momento de área (momento estático)
I	Segundo momento de área (momento de inércia de área)
t	Espessura
σ_f	Tensão de flexão
M_f	Momento de flexão
W_f	Módulo de resistência à flexão

h	Altura da seção transversal
h_i	Altura inicial da seção transversal
$(h_i)_e$	Altura inicial da seção transversal para o elemento e
$(h_f)_e$	Altura final da seção transversal para o elemento e
σ_{fea}	Tensão normal calculada pelo método dos elementos finitos
τ_{fea}	Tensão de cisalhamento calculada pelo método dos elementos finitos
$(\sigma_x)_e$	Tensão normal na direção x para o elemento e
$(\sigma_y)_e$	Tensão normal na direção y para o elemento e
$(\tau_{xy})_e$	Tensão de cisalhamento no plano xy para o elemento e
N_e	Altura final da seção transversal em número de elementos para o elemento e analisado
$(\sigma_{vm})_{max}$	Tensão de von Mises máxima para a estrutura
$(\sigma_x)_{max}$	Tensão normal máxima para a estrutura na direção x
$(\sigma_y)_{max}$	Tensão normal máxima para a estrutura na direção y
$(\tau_y)_{max}$	Tensão de cisalhamento máxima para a estrutura no plano xy
X	Ponto no espaço
$S_F^x(\mathbf{X})$	Componente de tensão para uma força P na direção x
$S_F^y(\mathbf{X})$	Componente de tensão para uma força P na direção y
$S_F^z(\mathbf{X})$	Componente de tensão para uma força P na direção z
$F_x(\mathbf{X})$	Força na direção x
$F_y(\mathbf{X})$	Força na direção y
$F_z(\mathbf{X})$	Força na direção z
$K_F^x(\mathbf{X})$	Coefficiente de cálculo da componente de tensão para uma força P na direção x
$K_F^y(\mathbf{X})$	Coefficiente de cálculo da componente de tensão para uma força P na direção y
$K_F^z(\mathbf{X})$	Coefficiente de cálculo da componente de tensão para uma força P na direção z

$S_M^x(\mathbf{X})$	Componente de tensão para um momento M na direção x
$S_M^y(\mathbf{X})$	Componente de tensão para um momento M na direção y
$S_M^z(\mathbf{X})$	Componente de tensão para um momento M na direção z
$M_x(\mathbf{X})$	Momento na direção x
$M_y(\mathbf{X})$	Momento na direção y
$M_z(\mathbf{X})$	Momento na direção z
$K_M^x(\mathbf{X})$	Coefficiente de cálculo da componente de tensão para um momento M na direção x
$K_M^y(\mathbf{X})$	Coefficiente de cálculo da componente de tensão para um momento M na direção y
$K_M^z(\mathbf{X})$	Coefficiente de cálculo da componente de tensão para um momento M na direção z
$\mathbf{F}(\mathbf{X})$	Vetor força no ponto \mathbf{X}
$\mathbf{M}(\mathbf{X})$	Vetor momento no ponto \mathbf{X}
\mathbf{r}	Vetor distância
$\sigma_f(\mathbf{X})$	Tensão de flexão no ponto \mathbf{X}
$\sigma_n(\mathbf{X})$	Tensão normal no ponto \mathbf{X}
$\mathbf{n}(\mathbf{X})$	Vetor normal à seção transversal no ponto \mathbf{X}
$\tau(\mathbf{X})$	Tensão de cisalhamento no ponto \mathbf{X}
$h_f(\mathbf{X})$	Estimativa da altura final da seção transversal com base no esforço de flexão no ponto \mathbf{X}
$h_n(\mathbf{X})$	Estimativa da altura final da seção transversal com base no esforço normal no ponto \mathbf{X}
$h_c(\mathbf{X})$	Estimativa da altura final da seção transversal com base no esforço cisalhante no ponto \mathbf{X}
$h_i(\mathbf{X})$	Altura inicial da seção transversal no ponto \mathbf{X}
$h(\mathbf{X})$	Altura final da seção transversal no ponto \mathbf{X}
c^μ	Coefficiente de normalização da tensão máxima na iteração μ

α^μ	Coefficiente de regulação da variação do coeficiente de normalização da tensão máxima na iteração μ
θ_{max}	Ângulo máximo de rotação da função de extrusão em revolução do OpenCascade
s	Passo máximo do algoritmo de <i>pathfinding</i>
V^μ	Volume total da geometria na iteração μ
V_{max}	Volume máximo da geometria

SUMÁRIO

1	INTRODUÇÃO	33
1.1	Contextualização	33
1.2	Justificativa	33
1.3	Objetivos	34
1.4	Organização da dissertação	34
2	REVISÃO BIBLIOGRÁFICA	37
2.1	Automação de projeto	37
2.2	Modelagem de vigas	37
2.3	Otimização topológica	39
2.3.1	Conceito	39
2.3.2	SIMP	41
2.3.3	Cálculo das sensibilidades	43
2.3.4	Programação matemática	44
2.3.4.1	Método do Critério de Ótimo	44
2.3.4.2	Método das Assíntotas Móveis	45
2.3.4.3	MMA Globalmente Convergente	46
2.3.5	Outros métodos de otimização topológica	49
2.3.5.1	Visão geral	49
2.3.5.2	ESO	50
2.3.5.3	BESO	50
2.3.5.4	Level-set	51
2.3.6	Restrições de tensão	52
2.3.7	Outros objetivos	55
3	IMPLEMENTAÇÃO EM AMBIENTE DISCRETIZADO	57
3.1	Considerações iniciais	57
3.2	Base teórica	58
3.3	Programa base	59
3.4	Implementação do algoritmo	59
3.4.1	Modificações do programa base	59
3.4.2	Simplificações adotadas	60
3.4.3	<i>Pathfinding</i>	63
3.5	Validação	63
3.6	Estudos de caso	63
3.7	Resultados	66

3.7.1	Caso 1	66
3.7.2	Caso 2	68
3.7.3	Caso 3	71
3.7.4	Caso 4	73
4	IMPLEMENTAÇÃO EM AMBIENTE CONTÍNUO	75
4.1	Considerações iniciais	75
4.2	Abordagens consideradas	77
4.2.1	Questões a serem superadas	77
4.2.2	Modelagem por elementos de vigas	78
4.2.3	Pontos centrais	78
4.2.4	Abordagem elementar	78
4.3	Abordagem utilizada	79
4.4	Grafos de visibilidade	81
4.5	Otimização topológica	82
4.6	Detalhes da implementação	84
4.6.1	Tamanho do elemento	84
4.6.2	Geração da geometria	84
4.6.2.1	Máximo ângulo de curvatura	84
4.6.2.2	Geometria sólida construtiva no OpenCascade	85
4.6.3	Implementação do GCMMA	86
4.6.3.1	Visão geral	86
4.6.3.2	NLOpt	86
4.6.3.3	GCMMA Solver	87
4.6.4	Tensões máximas de projeto	88
4.7	Estudos de caso	88
4.8	Resultados e discussão	91
4.8.1	Análise de elementos finitos	91
4.8.2	Otimização topológica	93
4.8.3	Tempo de otimização	98
5	CONCLUSÕES	101
5.1	Conclusões gerais	101
5.2	Conclusões sobre a implementação em ambiente discretizado	101
5.3	Conclusões sobre a implementação em ambiente contínuo	102
5.4	Trabalhos futuros	102
	REFERÊNCIAS	105

	APÊNDICES	115
	APÊNDICE A – O PROGRAMA TOPY	117
A.1	Visão geral	117
A.2	Ferramentas utilizadas	117
A.2.1	Python 2	117
A.2.2	PySparse	118
A.2.3	NumPy	118
A.2.4	SymPy	118
A.2.5	Outras bibliotecas	118
A.3	Implementação	119
A.3.1	Capacidades e limitações	119
A.3.2	Elementos	119
A.3.3	Análise de elementos finitos	120
A.3.4	Análise de sensibilidades	120
A.3.5	Restrição do espaço de solução	120
A.3.6	Atualização da matriz de densidades	120
A.3.7	Resultados	121

1 INTRODUÇÃO

1.1 Contextualização

No âmbito da otimização estrutural, as otimizações topológica, de forma e dimensional são os principais métodos, sendo comumente utilizadas em conjunto em diferentes etapas do processo de desenvolvimento da estrutura. A otimização topológica é frequentemente utilizada durante a fase de conceptualização, pois facilmente obtém topologias otimizadas simplesmente com a definição de um domínio e condições de contorno (Saitou *et al.*, 2005). As otimizações de forma e dimensional tendem a ser utilizadas nos estágios finais de desenvolvimento da geometria, por realizarem uma otimização mais fina da mesma (Gao *et al.*, 2020; Plocher; Panesar, 2019).

Porém, o processo de dimensionamento seria, teoricamente, melhor posicionado como anterior à otimização topológica, pois apresenta a capacidade de criar uma topologia simples e já adaptada a uma tensão de projeto. Esse processo comumente gera uma topologia com certo nível de sobredimensionamento, particularmente quanto à relação entre massa e resistência, o que poderia ser melhorado posteriormente pela otimização topológica e refinado pela otimização de forma. Essa sequência normalmente não é empregada, pois os métodos atuais de dimensionamento costumam ser manuais, iterativos e restritos a geometrias simples e padronizadas, como elementos de máquinas, envolvendo processos de dimensionamento de peças individuais ou repetidas edições de modelos CAD e análises de elementos finitos (Budynas; Nisbett, 2015).

Portanto, um processo de dimensionamento que fosse tão simples e intuitivo quanto o de otimização topológica, ainda que obtivesse resultados mais grosseiros, poderia aumentar a qualidade e rapidez do processo de desenvolvimento de produtos, estruturas e geometrias em geral. Esse processo poderia em seguida ser otimizado topologicamente para reduzir a massa da estrutura.

1.2 Justificativa

Na manufatura aditiva, os principais fatores responsáveis pelo custo de produção são o custo do material e o tempo de fabricação. O custo do material tende a apresentar maior peso, sendo dependente do volume, estratégia de preenchimento e geração de estruturas de suporte, enquanto o tempo de fabricação limita a taxa de produção, isto é, a quantidade de produtos que podem ser produzidos em determinado espaço de tempo, além de influenciar o consumo de energia elétrica. O tempo de fabricação é mais difícil de ser estimado com precisão que o uso de material, mas é comumente influenciado por fatores como complexidade da geometria, estratégia de preenchimento, geração de estruturas de suporte,

espessura da camada, área da base, área de superfície, entre outros fatores (Pradel *et al.*, 2017; Zhang *et al.*, 2015; Teitelbaum, 2009).

Em geral, a quantidade de material utilizado é reduzida por estratégias de preenchimento e otimização topológica (Sabiston; Kim, 2020), e o tempo de fabricação, por otimização da orientação da peça (Angelo; Stefano; Guardiani, 2020; Matos; Rocha; Costa, 2020), mas existem também outros métodos, como algoritmos genéticos (Oh *et al.*, 2020), fatiamento adaptativo da peça (Chen; Xu; Zhang, 2021), redes neurais (Giri *et al.*, 2021), entre outros. Isso pode, por exemplo, evitar ou diminuir a formação de estruturas de suporte e diminuir as dimensões ao longo do eixo vertical.

Embora a otimização topológica seja efetiva em reduzir o volume de uma peça, para problemas que envolvem resistir a forças e momentos as geometrias resultantes comumente mantêm grande parte do contorno da geometria original, portanto não há redução dimensional significativa em conjunto com a redução de volume. Em termos de garantir que a peça resista às cargas, isso é benéfico, pois contribui para aumentar sua rigidez. Consequentemente, a geometria apresenta maior superfície e requer que a ferramenta usada pelo método de manufatura aditiva percorra maiores distâncias (especialmente no perímetro e área de superfície, que são depositados mais lentamente (Pradel *et al.*, 2017)), resultando em maior tempo de fabricação.

Existem métodos de otimização estrutural que ocasionam em redução dimensional, como a otimização de forma. Porém, ela também é uma abordagem iterativa, o que aumenta o tempo necessário para projetar uma peça ou produto. Portanto, um método mais rápido que resultasse em redução dimensional e da mesma forma apresentasse volume mínimo resultaria em maior redução do tempo de fabricação e maior velocidade de desenvolvimento do projeto, além de aumentar o número de peças possíveis de serem fabricadas em um mesmo lote, aumentando a eficiência do processo.

1.3 Objetivos

- **Objetivo principal:** desenvolver e implementar um algoritmo capaz de obter topologias bidimensionais mais adequadas aos carregamentos e restrições de tensão definidos por projeto utilizando métodos de dimensionamento de vigas, com posterior otimização por métodos de otimização estrutural, como a topológica.
- **Objetivo secundário:** implementar o algoritmo desenvolvido para casos tridimensionais.

1.4 Organização da dissertação

O Capítulo 2 apresenta uma revisão dos principais conceitos utilizados no projeto, em especial otimização topológica por SIMP. O Capítulo 3 apresenta o algoritmo desenvol-

vido assim como uma primeira implementação desenvolvida para analisar seus resultados, baseada em um ambiente discretizado e com algumas simplificações. O Capítulo 4 apresenta uma implementação em um ambiente contínuo, sem a presença de tais simplificações. Por fim, o Capítulo 5 apresenta as principais conclusões do projeto e sugestões de trabalhos futuros.

2 REVISÃO BIBLIOGRÁFICA

2.1 Automação de projeto

Automação de projeto pode ser definida como um ramo da engenharia que trata do desenvolvimento de técnicas, algoritmos e tecnologias que automatizem total ou parcialmente o processo de projetagem. O termo não apresenta uma definição exata na literatura. A *Design Automation Conference* (DAC) organizada pela ASME, por exemplo, utiliza o termo de forma mais abrangente, incluindo também o projeto de sistemas automatizados (Ragsdell, 1980; New... , 2015). A área também aparece na literatura sob o nome de “Projeto Automatizado por Computador” (CAutoD - *Computer-Automated Design*) (Moharrami; Grierson, 1993), em alusão à sua relação com a área de Projeto Assistido por Computador (CAD).

A automação de projeto está mais desenvolvida na área de engenharia eletrônica, formando o ramo de automação de projeto eletrônico, um dos principais responsáveis pelo aumento da complexidade de circuitos integrados e, portanto, do aumento da velocidade de processamento dos dispositivos digitais que acompanhou a miniaturização dos transistores (Lavagno *et al.*, 2016).

Porém, a área de automação de projeto também apresenta espaço na engenharia mecânica, como evidencia a DAC. A otimização topológica apresenta papel importante nessa área, mas outros métodos de automação também estão sendo pesquisados e desenvolvidos, como métodos utilizando redes generativas antagônicas (Oh *et al.*, 2018; Shu *et al.*, 2019), aprendizado de máquina (Sharpe *et al.*, 2019; Camburn *et al.*, 2019), *design* generativo (Oh *et al.*, 2019) e inteligência computacional (Chen; Li, 2018), além do método de crescimento (*Growth Method*) utilizado para a otimização de estruturas treliçadas (Querin *et al.*, 2017). Métodos de automação de projeto já apresentam aplicações em manufatura aditiva (Wiberg; Persson; Ölvander, 2019), design de produtos (Kusiak; Salustri, 2007), desenvolvimento de dedos, garras e mãos de robôs (Honarpardaz *et al.*, 2017), desenvolvimento de robôs médicos e mecanismos (Sun *et al.*, 2019), entre outras áreas.

2.2 Modelagem de vigas

Modelos de vigas são modelos que visam descrever matematicamente o comportamento de barras, vigas, eixos e elementos similares. O modelo mais comumente utilizado é o modelo linear de Euler-Bernoulli, que descreve a relação entre cargas verticais distribuídas e a deformação vertical da viga através de uma equação diferencial. Assumindo pequenas deformações, ausência de deformação da seção transversal, seção transversal simétrica em

relação ao plano XY , eixo central alinhado ao eixo X , deslocamento no eixo Y e vigas finas (Goodno; Gere, 2018):

$$\frac{d^2}{dx^2} \left(E(x)I(x) \frac{d^2 w(x)}{dx^2} \right) = f_c(x), \quad (1)$$

em que $E(x)$ é o módulo de elasticidade, $I(x)$ o segundo momento de área, $w(x)$ o deslocamento e $f_c(x)$ a carga distribuída. A partir dela, é possível obter também o momento $M(x)$ em cada ponto e a força cortante $F_c(x)$:

$$\frac{d}{dx} \left(E(x)I(x) \frac{d^2 w(x)}{dx^2} \right) = F_c(x), \quad (2)$$

$$E(x)I(x) \frac{d^2 w(x)}{dx^2} = M(x). \quad (3)$$

Embora apresente essas limitações, o modelo é comumente usado em conjunto com outras técnicas para analisar vigas com formatos mais complexos, com cargas em diferentes números e tipos, sobredeterminadas, dinâmicas e sob grandes deformações.

Para o caso de vigas de maior espessura há também o modelo de Timoshenko, que considera os efeitos do cisalhamento (isto é, a deformação da seção transversal) e da inércia rotacional (Graff, 1975). Para o caso estático, sem presumir seção transversal constante, ela é apresentada como:

$$\begin{aligned} \frac{d}{dx} \left[GA\kappa \left(\frac{dw}{dx} - \varphi \right) \right] &= -f_c(x) \\ GA\kappa \left(\frac{dw}{dx} - \varphi \right) + \frac{d}{dx} \left(EI \frac{d\varphi}{dx} \right) &= 0, \end{aligned} \quad (4)$$

onde G é o módulo de cisalhamento, A a área da seção transversal, φ o ângulo de curvatura da viga, e κ o coeficiente de cisalhamento de Timoshenko. Isolando φ na primeira equação, podemos então simplificá-la para:

$$\frac{d}{dx} \left(\frac{F_c(x)}{GA\kappa} \right) + \frac{d^2 w}{dx^2} = \frac{M(x)}{EI}, \quad (5)$$

que evidencia sua relação com o modelo de Euler-Bernoulli. De mesma forma, ela apresenta limitações que são compensadas com o uso de extensões e modificações do modelo em conjunto com outras ferramentas, como métodos de discretização do espaço.

2.3 Otimização topológica

2.3.1 Conceito

Bendsøe e Sigmund (2013) definem a otimização topológica de estruturas sólidas como a “determinação de características como número, posição e formato dos furos e conectividade do domínio”. Ela, portanto, difere da otimização de forma, a qual Bendsøe e Sigmund (2013) definem como “encontrar o contorno ótimo do domínio”, e do dimensionamento, no qual o domínio e as variáveis de estado são conhecidas e constantes, variando-se somente as variáveis de projeto (como espessura ou diâmetro, por exemplo).

Partindo de um domínio de projeto fixo e contínuo, isto é, uma topologia bi- ou tridimensional contínua, representada genericamente por Ω , a otimização topológica pode ser descrita como um problema de distribuição de material. Várias abordagens podem ser usadas para definir a distribuição de material, conforme os objetivos de projeto, sendo a mais simples a de minimização de flexibilidade (isto é, maximização de rigidez).

Para uma estrutura que ocupa uma região Ω no espaço e apresenta contorno Γ , Bendsøe e Kikuchi (1988) definem o problema de minimização de flexibilidade para estruturas contínuas no regime elástico como a minimização do trabalho das cargas atuantes (de campo b , aplicadas sobre todo o domínio, e de superfície t , aplicadas sobre uma porção do contorno $\Gamma_T \in \Gamma$) sobre a estrutura para um campo de deslocamento u (isto é, minimização do trabalho externo $l(u)$, utilizado como medida de flexibilidade):

$$l(u) = \int_{\Omega} bud\Omega + \int_{\Gamma_T} tud\Gamma, \quad (6)$$

tal que o trabalho externo virtual para um campo de deslocamento v ($l(v)$) seja igual ao trabalho virtual das tensões em equilíbrio em relação a v (isto é, ao trabalho interno virtual $a(u, v)$),

$$a(u, v) = \int_{\Omega} E_{ijkl}(x)\varepsilon_{ij}(u)\varepsilon_{kl}(v)d\Omega, \quad (7)$$

em que E é o tensor constitutivo e ε o tensor deformação linearizado:

$$\varepsilon_{ij}(u) = \frac{1}{2} \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right), \quad (8)$$

utilizando a notação de Einstein para tensores. O problema de otimização é então definido como:

$$\begin{aligned} \min_{u \in U} \quad & l(u) \\ \text{sujeito a} \quad & a(u, v) = l(v), \quad \forall v \in U \\ & E \in \mathbf{E}_{ad} \end{aligned} \quad (9)$$

em que \mathbf{E}_{ad} é o conjunto de tensores constitutivos admissíveis e U é o conjunto dos campos de deslocamento admissíveis.

Visto que as estruturas podem apresentar geometrias complexas, o problema é comumente abordado através de algum método de discretização do domínio, como o

método dos elementos finitos. Para o método dos elementos finitos com N elementos, o trabalho externo pode ser escrito como:

$$l(u) = \int_{\Omega} b u d\Omega + \int_{\Gamma_T} t u d\Gamma = \sum_{e=1}^N \int_{\Omega_e} \mathbf{b}_e^T \Phi_e \mathbf{u}_e d\Omega + \sum_{e=1}^N \int_{\Gamma_e} \mathbf{t}_e^T \Phi_e \mathbf{u}_e d\Gamma \quad (10)$$

tal que $u_e(x) = \Phi_e(x) \mathbf{u}_e$ é a aproximação do campo de deslocamentos do elemento, em que Φ_e é a função de forma do elemento e e Γ_e a porção do contorno externo pertencente ao contorno do elemento, $\Gamma_T = \bigcup_{e=1}^N \Gamma_e$, e, analogamente, Ω_e é a região do domínio ocupada pelo elemento e . A função de forma Φ_e representa de forma matricial o contorno de um elemento, sendo cada porção do contorno delimitada por um vértice, denominado “nó”. Os campos u , b e t são transformados em vetores elementares cujos valores são os valores das funções de campo nos nós de cada elemento, para cada grau de liberdade.

Por isso, as integrais resultam em:

$$\begin{aligned} l(u) = c(\mathbf{u}) &= \sum_{e=1}^N \mathbf{f}_e^T \mathbf{u}_e = \mathbf{f}^T \mathbf{u} \\ \mathbf{f}_e &= \int_{\Omega_e} \Phi_e^T \mathbf{b}_e d\Omega + \int_{\Gamma_e} \Phi_e^T \mathbf{t}_e d\Gamma \end{aligned} \quad (11)$$

tal que $\mathbf{f}_e = \mathbf{L}_e^T \mathbf{f}$ é o vetor de forças elementar, \mathbf{f} , o global, $\mathbf{u}_e = \mathbf{L}_e \mathbf{u}$ o vetor de deslocamentos local e \mathbf{u} , o global, em que \mathbf{L}_e é a matriz que relaciona os resultados elementares aos globais.

Para o trabalho interno, reduz-se o tensor de deformação de um tensor de segunda ordem a um vetor $\boldsymbol{\varepsilon} = \nabla_S \mathbf{u}(x)$, tal que $\boldsymbol{\varepsilon}_e = \nabla_S \mathbf{u}_e(x)$ e ∇_S é o operador ∇ (gradiente) reduzido a uma forma vetorial. Para o estado plano de tensões, por exemplo, consiste em (Huebner *et al.*, 2001):

$$\nabla_S = \begin{bmatrix} \frac{\partial}{\partial x} & 0 \\ 0 & \frac{\partial}{\partial y} \\ \frac{\partial}{\partial y} & \frac{\partial}{\partial x} \end{bmatrix} \quad (12)$$

Portanto, $\boldsymbol{\varepsilon}_e = \nabla_S \Phi_e \mathbf{u}_e = \mathbf{B}_e \mathbf{u}_e$.

De forma semelhante, o tensor constitutivo é reduzido de um tensor de quarta ordem a uma matriz quadrada \mathbf{C}_e para todo elemento e . Disso, pode-se obter o trabalho interno de forma discretizada:

$$\begin{aligned}
a(u, v) &= \int_{\Omega} E_{ijkl}(x) \varepsilon_{ij}(u) \varepsilon_{kl}(v) d\Omega \\
&= \sum_{e=1}^N \mathbf{u}_e^T \left(\int_{\Omega_e} \mathbf{B}_e^T \mathbf{C}_e \mathbf{B}_e d\Omega \right) \mathbf{v}_e \\
&= \sum_{e=1}^N \mathbf{u}^T \mathbf{L}_e^T \left(\int_{\Omega_e} \mathbf{B}_e^T \mathbf{C}_e \mathbf{B}_e d\Omega \right) \mathbf{L}_e \mathbf{v} = \mathbf{u}^T \mathbf{K} \mathbf{v}
\end{aligned} \tag{13}$$

tal que $\mathbf{K} = \sum_{e=1}^N \mathbf{L}_e^T \mathbf{K}_e \mathbf{L}_e$ com $\mathbf{K}_e = \int_{\Omega_e} \mathbf{B}_e^T \mathbf{C}_e \mathbf{B}_e d\Omega$.

Com isso,

$$a(u, v) = l(v) \Rightarrow \mathbf{u}^T \mathbf{K} \mathbf{v} = \mathbf{f}^T \mathbf{v} \Rightarrow \mathbf{K} \mathbf{u} = \mathbf{f} \tag{14}$$

Portanto, o problema pode ser reescrito como:

$$\begin{aligned}
&\min_{\mathbf{u} \in U} \quad \mathbf{f}^T \mathbf{u} \\
&\text{sujeito a} \quad \mathbf{K}(E) \mathbf{u} = \mathbf{f} \\
&\quad \quad \quad E \in \mathbf{E}_{ad}
\end{aligned} \tag{15}$$

2.3.2 SIMP

O método SIMP (abreviação de *Solid Isotropic Material with Penalization*) foi um dos primeiros métodos a serem desenvolvidos e ainda é amplamente utilizado devido à facilidade de implementação (Deaton; Grandhi, 2014), além de, por ser mais antigo, ser mais testado e confiável.

O método consiste em atrelar ao campo tensorial de rigidez um campo de densidade contínuo $0 \leq \rho(x) \leq 1$ a ser utilizado como variável de projeto, penalizado por uma potência p (Bendsøe, 1989):

$$\begin{aligned}
E_{ijkl}(x) &= \rho(x)^p E_{ijkl}^0, \quad p > 1, \\
0 &\leq \rho(x) \leq 1, \quad x \in \Omega,
\end{aligned} \tag{16}$$

em que E_{ijkl}^0 é o tensor constitutivo original do elemento. Na forma discreta é necessário definir um valor mínimo para a densidade, ρ_{min} (comumente $\rho_{min} = 10^{-3}$), para evitar que a matriz de rigidez global seja singular:

$$\begin{aligned}
\mathbf{K}_e &= \rho_e^p \mathbf{K}_e^0, \quad p > 1, \\
0 &< \rho_{min} \leq \rho_e \leq 1, \quad e = 1, \dots, N,
\end{aligned} \tag{17}$$

em que \mathbf{K}_e^0 é a matriz elementar de rigidez original do elemento.

A constante p é utilizada para a penalização dos valores de densidade. Para $\rho(x) < 1$, $\rho^p(x) < \rho(x)$ (para $p > 1$), o que torna a rigidez do ponto muito menor do que o esperado para seu volume ($\rho(x)v(x)$), incentivando a diminuição da densidade em alguns pontos e o

aumento dela em outros, o que ocasiona na formação de uma distribuição de densidades que alterna entre 1 e 0. O valor $p = 3$ é comumente usado, sendo o valor mínimo que possibilita a obtenção de topologias “preto e branco”. Porém, segundo Bendsøe e Sigmund (1999), o valor mínimo de p depende do coeficiente de Poisson do material (ν):

$$\begin{aligned} p &\geq \max \left\{ \frac{2}{1-\nu}, \frac{4}{1+\nu} \right\} && \text{(em 2D)} \\ p &\geq \max \left\{ 15 \frac{1-\nu}{7-5\nu}, \frac{3}{2} \frac{1-\nu}{1-2\nu} \right\} && \text{(em 3D)} \end{aligned} \quad (18)$$

restrição que permite que densidades intermediárias apresentem sentido físico (como compósitos formados por espaços vazios e preenchidos).

Ao ser implementado, é comum a adoção de um $\rho_{min} = 10^{-3}$ para evitar que a matriz \mathbf{K} torne-se singular, o que impediria a resolução do sistema de equações lineares, redefinindo-se, portanto, os limites de $\rho(x)$ para $\rho_{min} \leq \rho(x) \leq 1$. Também é possível remover os elementos do problema ao invés de utilizar uma densidade mínima, mas isso introduz algumas complexidades computacionais e instabilidades no processo de otimização (Bruns; Tortorelli, 2003).

Também é possível utilizar uma matriz rigidez mínima K_{min}^e (Sigmund, 2007):

$$\mathbf{K}_e = \mathbf{K}_e^{min} + \rho_e^p (\mathbf{K}_e^0 - \mathbf{K}_e^{min}) \quad (19)$$

em que \mathbf{K}_e^{min} pode ser definido, por exemplo, por $\mathbf{K}_e^{min} = 10^{-9} \mathbf{K}_e^0$, resultando em (Silva; Beck; Sigmund, 2019a):

$$\mathbf{K}_e = [10^{-9} + \rho_e^p (1 - 10^{-9})] \mathbf{K}_e^0 \quad (20)$$

Para evitar problemas como a obtenção de resultados consideravelmente diferentes com a refinação da malha e a criação de padrões de “tabuleiro de xadrez”, em que elementos com material se alternam com elementos sem material, é preciso utilizar algum método ou combinação de métodos de restrição do espaço de solução. Alguns métodos comumente usados são: filtragem de sensibilidade, filtragem de densidade, controle de perímetro e restrição de gradiente, que conseguem solucionar ambos os problemas. Há também métodos específicos para o problema do tabuleiro de xadrez, como métodos de remoção direta desse padrão em grupos de elementos vizinhos e o método NoHinge (Bendsøe; Sigmund, 2013).

O método de filtragem de sensibilidade é eficiente para problemas simples e fácil de implementar, sendo um dos primeiros métodos a serem desenvolvidos para abordar o problema de criação de tabuleiros de xadrez (Sigmund; Petersson, 1998). Ele consiste em “homogeneizar” o campo de sensibilidades, tornando a sensibilidade de cada elemento igual à média das sensibilidades dos elementos dentro de um raio r_{min} , ponderadas pelo fator \hat{H}_{ki} (Bendsøe; Sigmund, 2013):

$$\widehat{\frac{\partial c}{\partial \rho_k}} = \frac{1}{\rho_k \sum_{j=1}^N \widehat{H}_{kj}} \sum_{i=1}^N \widehat{H}_{ki} \rho_i \frac{\partial c}{\partial \rho_i} \quad (21)$$

$$\widehat{H}_{ki} = r_{min} - \text{dist}(k, i), \quad \{i \in N | \text{dist}(k, i) \leq r_{min}\}$$

O método é bem-sucedido em resolver o problema em casos simples, mas não é matematicamente consistente nem apresenta uma interpretação física bem definida, o que pode torná-lo instável em aplicações mais complexas, embora tenha gerado resultados satisfatórios em um grande número de aplicações (Bendsøe; Sigmund, 2013).

A filtragem de densidades é um método mais estável, melhor definido fisicamente e gera resultados menos dependentes da malha utilizada. Ela consiste em tornar o campo de densidades menos descontínuo, de forma análoga à filtragem de sensibilidades (Bruns; Tortorelli, 2001; Bourdin, 2001; Bendsøe; Sigmund, 2013):

$$\rho_k = \frac{\sum_{i=1}^N \widehat{H}_{ki} d_i}{\sum_{j=1}^N \widehat{H}_{kj}} \quad (22)$$

$$\widehat{H}_{ki} = r_{min} - \text{dist}(k, i), \quad \{i \in N | \text{dist}(k, i) \leq r_{min}\}$$

Em que $\mathbf{d} = d_1, d_2, \dots, d_n$ são as variáveis de projeto, isto é, as variáveis utilizadas pelo método de programação matemática para realizar a otimização da função objetivo.

2.3.3 Cálculo das sensibilidades

Quando o método de programação matemática utilizado requer o cálculo do gradiente da função objetivo ou das restrições, é preciso utilizar algum método que facilite esse cálculo considerando os requisitos do problema. No caso de minimização de flexibilidade, tem-se:

$$c = \mathbf{f}^T \mathbf{u}, \quad (23)$$

tal que

$$\mathbf{K}(\boldsymbol{\rho}) \mathbf{u} = \mathbf{f}. \quad (24)$$

O gradiente de c é, portanto, calculado por:

$$\frac{\partial c}{\partial \rho_e} = \mathbf{f}^T \frac{\partial \mathbf{u}}{\partial \rho_e}. \quad (25)$$

Neste caso, \mathbf{u} é função de $\boldsymbol{\rho}$ visto que $\mathbf{K}(\boldsymbol{\rho})$ o é. Como \mathbf{u} é definido implicitamente, seu gradiente pode ser obtido, por exemplo, derivando-se a Equação 24:

$$\frac{\partial \mathbf{K} \mathbf{u}}{\partial \rho_e} = \frac{\partial \mathbf{f}}{\partial \rho_e}, \quad (26)$$

o que resulta em

$$\frac{\partial \mathbf{K}}{\partial \rho_e} \mathbf{u} + \mathbf{K} \frac{\partial \mathbf{u}}{\partial \rho_e} = \mathbf{0}, \quad (27)$$

e, portanto, o gradiente de \mathbf{u} poderia ser calculado implicitamente por:

$$\mathbf{K} \frac{\partial \mathbf{u}}{\partial \rho_e} = -\frac{\partial \mathbf{K}}{\partial \rho_e} \mathbf{u}, \quad (28)$$

o que implicaria em resolver um problema de elementos finitos para cada elemento, um processo que seria muito intensivo computacionalmente.

Um dos métodos usados para contornar isso é o *método adjunto*, que consiste em reescrever a função objetivo como (Bendsøe; Sigmund, 2013):

$$c = \mathbf{f}^T \mathbf{u} - \bar{\mathbf{u}}^T (\mathbf{K} \mathbf{u} - \mathbf{f}), \quad (29)$$

tal que $\bar{\mathbf{u}}$ é um vetor constante de deslocamentos virtuais. É importante notar que $\mathbf{K} \mathbf{u} - \mathbf{f} = \mathbf{0}$, portanto, na prática, a Equação 29 é igual à Equação 23. O cálculo do gradiente, porém, é diferente:

$$\begin{aligned} \frac{\partial c}{\partial \rho_e} &= \mathbf{f}^T \frac{\partial \mathbf{u}}{\partial \rho_e} - \bar{\mathbf{u}}^T \left(\frac{\partial \mathbf{K}}{\partial \rho_e} \mathbf{u} + \mathbf{K} \frac{\partial \mathbf{u}}{\partial \rho_e} - \mathbf{f} \right) \\ &= (\mathbf{f}^T - \bar{\mathbf{u}}^T \mathbf{K}) \frac{\partial \mathbf{u}}{\partial \rho_e} - \bar{\mathbf{u}}^T \frac{\partial \mathbf{K}}{\partial \rho_e} \mathbf{u}. \end{aligned} \quad (30)$$

O objetivo então é obter $\bar{\mathbf{u}}$ tal que $\bar{\mathbf{u}}^T \mathbf{K}(\boldsymbol{\rho}) - \mathbf{f}^T = 0$, eliminando $\frac{\partial \mathbf{u}}{\partial \rho_e}$ da função. Neste caso, como a equação é equivalente a $\mathbf{K} \mathbf{u} = \mathbf{f}$, então $\bar{\mathbf{u}} = \mathbf{u}$. Portanto:

$$\frac{\partial c}{\partial \rho_e} = -\mathbf{u}^T \frac{\partial \mathbf{K}(\boldsymbol{\rho})}{\partial \rho_e} \mathbf{u} \quad (31)$$

Para atualizar o campo de densidades, é preciso utilizar algum método de otimização não linear. O método do critério de ótimo (*Optimality Criteria - OC*) é um dos mais simples, enquanto o método das assíntotas móveis (*Method of Moving Asymptotes - MMA*) apresenta melhor desempenho para problemas mais complexos (Bendsøe; Sigmund, 2013).

2.3.4 Programação matemática

2.3.4.1 Método do Critério de Ótimo

O método do critério de ótimo é uma abordagem baseada nos multiplicadores de Lagrange. Considerando as restrições impostas pelo problema de minimização de flexibilidade com restrição de volume, com volume máximo \bar{V} , $\int_{\Omega} \rho(x) d\Omega \leq \bar{V}$ e $\rho_{min} \leq \rho(x) \leq 1$, tem-se que (Bendsøe; Sigmund, 2013):

$$\begin{aligned} \mathcal{L} &= l(\mathbf{u}) - [a_E(\mathbf{u}, \bar{\mathbf{u}}) - l(\bar{\mathbf{u}})] + \Lambda \left(\int_{\Omega} \rho(x) d\Omega - \bar{V} \right) \\ &\quad + \int_{\Omega} \lambda^+(x) (\rho(x) - 1) d\Omega + \int_{\Omega} \lambda^-(x) (\rho_{min} - \rho(x)) d\Omega \end{aligned} \quad (32)$$

onde Λ , λ^+ e λ^- são multiplicadores de Lagrange, tal que $\lambda^+ \geq 0$, $\lambda^- \geq 0$, $\lambda^+(x)(\rho(x)-1) = 0$ e $\lambda^-(x)(\rho_{min} - \rho(x)) = 0$. No caso de minimização de flexibilidade, tem-se que $u = \bar{u}$, como presente na Subseção 2.3.3. Derivando a função pela densidade e igualando-a a zero, obtém-se:

$$\frac{\partial E_{ijkl}}{\partial \rho} \varepsilon_{ij}(u) \varepsilon_{kl}(u) = \Lambda + \lambda^+ - \lambda^-. \quad (33)$$

Porém, devido à definição de λ^+ e λ^- , tem-se que $\lambda^+ = 0$ para $\rho(x) < 1$ e $\lambda^- = 0$ para $\rho(x) > \rho_{min}$. Portanto, para $\rho_{min} < \rho(x) < 1$:

$$\frac{\partial E_{ijkl}}{\partial \rho} \varepsilon_{ij}(u) \varepsilon_{kl}(u) = p\rho(x)^{p-1} E_{ijkl}^0 \varepsilon_{ij}(u) \varepsilon_{kl}(u) = \Lambda. \quad (34)$$

Isto é, a energia de deformação deve ser constante em densidades intermediárias. Porém, como áreas que apresentam alta energia de deformação também apresentam rigidez muito baixa, é utilizado um método de atualização das densidades baseado em um passo fixo:

$$\rho_{\mu+1} = \begin{cases} \max\{(1 - \zeta)\rho^\mu, \rho_{min}\} & \text{se } \rho^\mu (B^\mu)^\eta \leq \max\{(1 - \zeta)\rho^\mu, \rho_{min}\}, \\ \min\{(1 + \zeta)\rho^\mu, 1\} & \text{se } \min\{(1 + \zeta)\rho^\mu, 1\} \leq \rho^\mu (B^\mu)^\eta, \\ \rho^\mu (B^\mu)^\eta & \text{caso contrário} \end{cases} \quad (35)$$

em que μ é a iteração, ζ é o passo máximo (comumente $\zeta = 0,2$), η o parâmetro de ajuste (comumente $\eta = 0,5$) e B^μ é definido como:

$$B^\mu = (\Lambda^\mu)^{-1} p\rho(x)^{p-1} E_{ijkl}^0 \varepsilon_{ij}(u) \varepsilon_{kl}(u), \quad (36)$$

o que significa que material é retirado de regiões que apresentam maior energia de deformação ($B^\mu > 1$) e adicionado em regiões que apresentam menor energia ($B^\mu < 1$) em relação a Λ^μ , que é proporcional à energia de deformação média nas regiões que apresentam densidades intermediárias (Bendsøe; Sigmund, 2013). O Λ^μ pode ser calculado a cada iteração, por exemplo, por um método de bissecção (Sigmund, 2001).

Este caso apresenta a derivação clássica do método, em que se assume que as densidades são atualizadas diretamente, por exemplo, devido ao uso de filtragem de sensibilidade. No caso do uso de filtragem de densidade, seria necessário atualizar as variáveis de otimização \mathbf{d} .

2.3.4.2 Método das Assíntotas Móveis

O MMA é um método desenvolvido com base nos requisitos de problemas de otimização estrutural e consiste em aproximar o problema por subproblemas separáveis e convexos, de forma análoga à abordagem da programação linear (subproblemas lineares) e programação quadrática (subproblemas quadráticos) (Svanberg, 1987; Bendsøe; Sigmund, 2013).

O MMA busca resolver um problema de otimização com o seguinte formato:

$$\begin{aligned} \min \quad & f_0(\mathbf{d}), \mathbf{d} \in \mathbb{R}^n \\ \text{sujeito a} \quad & f_i(\mathbf{d}) \leq \hat{f}_i, \quad i = 1, \dots, m \\ & d_j^{\min} \leq d_j \leq d_j^{\max}, \quad j = 1, \dots, n \end{aligned} \quad (37)$$

em que f_0 é a função objetivo, f_i , as restrições, sendo \hat{f}_i seus respectivos limites superiores, e d_j^{\min} e d_j^{\max} são os limites das variáveis $d_j \in \mathbf{d}$. Para solucionar o problema, o método aproxima as funções $f_i(\mathbf{d})$ em torno de um ponto \mathbf{d}^μ por somatórios de funções convexas:

$$f_i^\mu(\mathbf{d}) \approx r_i^\mu + \sum_{j=1}^n \left(\frac{p_{ij}^\mu}{U_j^\mu - d_j} + \frac{q_{ij}^\mu}{d_j - L_j^\mu} \right) \quad (38)$$

em que as variáveis $U_i, L_i > 0$ são assíntotas verticais atualizadas a cada iteração μ (portanto, *móveis*), $f_i^\mu(\mathbf{d})$ é a estimativa da função f_i para a iteração μ e:

$$\begin{cases} p_{ij}^\mu = (U_i^\mu - d_i) \frac{\partial f_i}{\partial d_j}(\mathbf{d}^\mu) \text{ e } q_{ij}^\mu = 0 & \text{se } \frac{\partial f_i}{\partial d_j}(\mathbf{d}^\mu) > 0 \\ p_{ij}^\mu = 0 \text{ e } q_{ij}^\mu = -(L_i^\mu - d_i) \frac{\partial f_i}{\partial d_j}(\mathbf{d}^\mu) & \text{se } \frac{\partial f_i}{\partial d_j}(\mathbf{d}^\mu) < 0 \end{cases} \quad (39)$$

$$r_i^\mu = f_i(\mathbf{d}^\mu) - \sum_{i=1}^n \left(\frac{p_{ij}^\mu}{U_i^\mu - d_i} + \frac{q_{ij}^\mu}{d_i - L_i^\mu} \right) \quad (40)$$

tal que $L_j^\mu < d_j < U_j^\mu$.

Essa formulação cria um subproblema de otimização com o seguinte formato:

$$\begin{aligned} \min \quad & r_0^\mu + \sum_{i=1}^n \left(\frac{p_{ij}^\mu}{U_i^\mu - d_i} + \frac{q_{ij}^\mu}{d_i - L_i^\mu} \right), \mathbf{d} \in \mathbb{R}^n \\ \text{sujeito a} \quad & r_i^\mu + \sum_{i=1}^n \left(\frac{p_{ij}^\mu}{U_i^\mu - d_i} + \frac{q_{ij}^\mu}{d_i - L_i^\mu} \right) \leq \hat{f}_i, \quad i = 1, \dots, m \\ & \alpha_j \leq d_j \leq \beta_j, \quad j = 1, \dots, n \end{aligned} \quad (41)$$

tal que $L_j^\mu < d_j^{\min} \leq \alpha_j \leq d_j \leq \beta_j \leq d_j^{\max} < U_j^\mu$, problema que pode ser resolvido por métodos de otimização convexa. As assíntotas L_j e U_j , assim como os limites do subproblema (α_j e β_j), devem ser atualizadas a cada iteração conforme as especificidades das funções envolvidas. Por exemplo, a presença de instabilidades na convergência implica no uso de limites menores, enquanto baixas velocidades de convergência, no uso de limites mais amplos (Svanberg, 1987).

2.3.4.3 MMA Globalmente Convergente

Embora a formulação inicial do MMA apresentasse bons resultados em suas aplicações, ela não era globalmente convergente (ou seja, não convergia para um mínimo a partir de qualquer solução possível, isto é, de qualquer ponto inicial que se adéque às restrições impostas) e falhava quando usada com certos problemas (Svanberg, 2002). Por isso foi

desenvolvido o GCMMA, MMA Globalmente Convergente. O GCMMA busca resolver o seguinte problema:

$$\begin{aligned} \min \quad & f_0(\mathbf{d}) + A_0 z + \sum_{i=1}^m \left(C_i y_i + \frac{1}{2} D_i y_i^2 \right), \quad \mathbf{d} \in \mathbb{R}^n \\ \text{sujeito a} \quad & f_i(\mathbf{d}) - A_i z - y_i \leq 0, \quad i = 1, \dots, m \\ & d_j^{\min} \leq d_j \leq d_j^{\max}, \quad j = 1, \dots, n \end{aligned} \quad (42)$$

em que $\mathbf{y} = y_1, \dots, y_m$ e z são “variáveis artificiais de otimização”, utilizadas para transformar esse problema geral em problemas mais específicos, sendo elas utilizadas apenas pelo subproblema, em conjunto com os parâmetros $A_0, A_1, \dots, A_m, C_1, \dots, C_m$ e D_1, \dots, D_m , definidos pelo usuário de acordo com o problema de otimização a ser resolvido, tal que $A_0 > 0, A_i \geq 0, C_i \geq 0, D_i \geq 0, C_i + D_i \geq 0$ e $A_i C_i > A_0$ caso $A_i > 0$.

Por exemplo, para transformar a Equação 42 no problema da Equação 37, usa-se $A_0 = 1, A_i = 0$ ($i > 0$), $D_i = 1$ e C_i grande, sendo C_i responsável por manter $f_i(\mathbf{x}) \leq 0$. Svanberg (2007) recomenda inicialmente definir $C_i = 1000$ ou 10000 e aumentá-lo caso esse valor não seja suficiente para obter soluções possíveis.

As funções envolvidas são então aproximadas da mesma forma que no MMA. As principais diferenças do GCMMA para o MMA original são a utilização de “iterações internas” e um passo adaptável.

A cada “iteração externa”, são calculados os valores das funções e seus gradientes. A partir desses resultados, calcula-se um ponto que possa ser uma solução possível. Esse ponto é usado nas funções originais e comparado às estimativas. Caso os valores obtidos sejam menores que as estimativas (isto é, *conservativos*), o método segue para a próxima iteração externa. Caso contrário, as iterações internas são usadas para melhorar essas estimativas, até obter-se um ponto conservativo. A Figura 1 ilustra esse processo.

Essas estimativas são controladas pela variável $\xi_i^{\mu, \chi}$, atualizada a cada iteração externa μ e interna χ . No início de cada iteração externa, ela é definida por:

$$\xi_i^{\mu, 0} = \frac{0.1}{n} \sum_{i=1}^n \left| \frac{\partial f_i}{\partial d_j} \right| (d_j^{\max} - d_j^{\min}), \quad i = 0, \dots, m \quad (43)$$

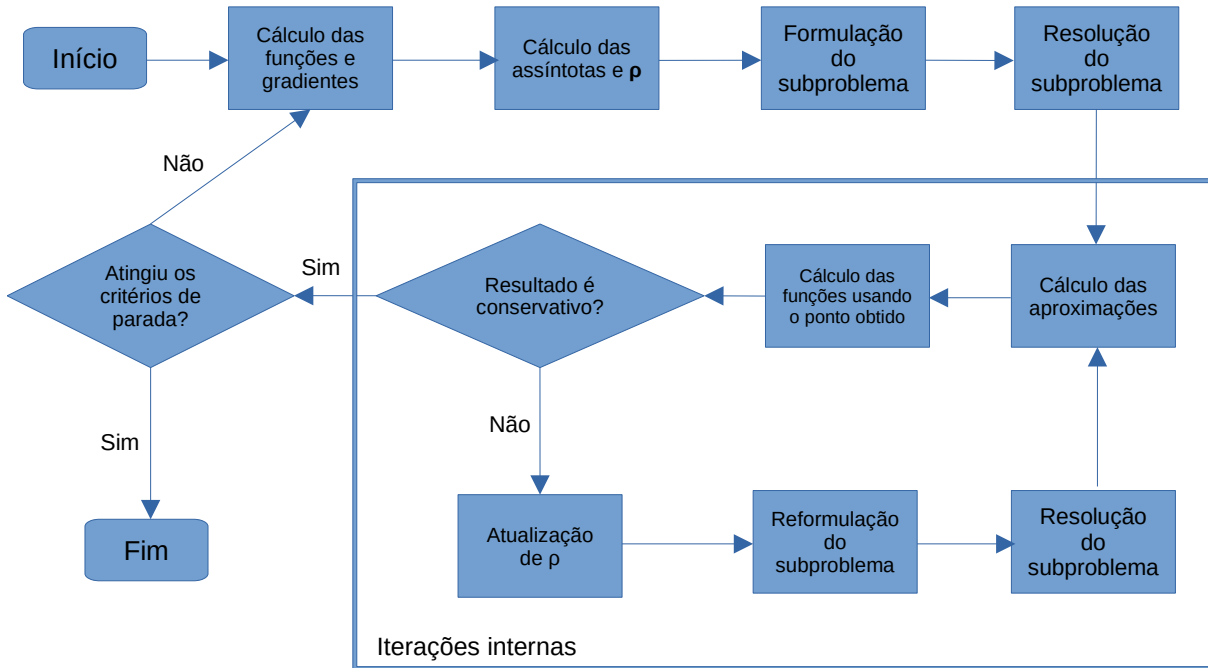
tal que $\xi_i^{\mu, \chi} \geq 10^{-6}$.

Caso a estimativa não seja conservativa, calcula-se:

$$d^\mu(\mathbf{d}) = \sum_{j=1}^n \frac{(U_j^\mu - L_j^\mu)(d_j - d_j^\mu)^2}{(U_j^\mu - d_j)(d_j - L_j)(d_j^{\max} - d_j^{\min})} \quad (44)$$

$$\delta_i^{\mu, \chi} = \frac{f_i(\mathbf{d}^{\mu, \chi}) - f_i^\mu(\mathbf{d}^{\mu, \chi})}{d^\mu(\mathbf{d}^{\mu, \chi})} \quad (45)$$

Figura 1 – Diagrama de blocos do funcionamento do GCMMA.



Fonte: Elaborado pelo autor.

e $\xi_i^{\mu,\chi}$ é atualizado por:

$$\xi_i^{\mu,\chi+1} = \begin{cases} \min\{1.1(\xi_i^{\mu,\chi} + \delta_i^{\mu,\chi}), 10\xi_i^{\mu,\chi}\} & \delta_i^{\mu,\chi} > 0 \\ \xi_i^{\mu,\chi} & \delta_i^{\mu,\chi} \leq 0 \end{cases} \quad (46)$$

Isto é, caso o resultado obtido com $\mathbf{d}^{\mu,\chi}$ seja maior que a estimativa, $\xi_i^{\mu,\chi}$ é aumentado, caso contrário, se mantém. E quando a função tende a convergir, $\xi_i^{\mu,\chi}$ tende a zero. As variáveis p_{ij} e q_{ij} no GCMMA são definidas de forma diferente à do MMA, incorporando $\xi_i^{\mu,\chi}$:

$$\begin{aligned} p_{ij}^{\mu,\chi} &= (U_i^\mu - d_i^\mu)^2 \left(1,001 \frac{\partial f_i^+}{\partial d_j}(\mathbf{d}^\mu) + 0,001 \frac{\partial f_i^-}{\partial d_j}(\mathbf{d}^\mu) + \frac{\xi_i^{\mu,\chi}}{d_j^{\max} - d_j^{\min}} \right) \\ q_{ij}^{\mu,\chi} &= (d_i^\mu - L_i^\mu)^2 \left(0,001 \frac{\partial f_i^+}{\partial d_j}(\mathbf{d}^\mu) + 1,001 \frac{\partial f_i^-}{\partial d_j}(\mathbf{d}^\mu) + \frac{\xi_i^{\mu,\chi}}{d_j^{\max} - d_j^{\min}} \right) \end{aligned} \quad (47)$$

em que:

$$\begin{aligned} \frac{\partial f_i^+}{\partial d_j} &= \max \left\{ \frac{\partial f_i}{\partial d_j}, 0 \right\} \\ \frac{\partial f_i^-}{\partial d_j} &= \max \left\{ -\frac{\partial f_i}{\partial d_j}, 0 \right\} \end{aligned} \quad (48)$$

isto é, $\xi_i^{\mu,\chi}$ é responsável por definir o quão “otimista” será a estimativa. É importante notar que os valores das sensibilidades são atualizados apenas a cada *iteração externa*, o que é vantajoso para diminuir a carga computacional durante o processo de otimização

(Svanberg, 2007), visto que, por exemplo, para otimizações topológicas baseadas em tensão, o cálculo do gradiente de tensão requer a realização de uma segunda análise de elementos finitos (Yang; Chen, 1996).

Quantos às assíntotas, nas duas primeiras iterações externas elas são atualizadas por:

$$\begin{aligned} L_j^\mu &= d_j^\mu - 0,5(d_j^{max} - d_j^{min}) \\ U_j^\mu &= d_j^\mu + 0,5(d_j^{max} - d_j^{min}) \end{aligned} \quad (49)$$

e partir da terceira iteração por:

$$\begin{aligned} L_j^\mu &= d_j^\mu - \gamma_j^\mu(d_j^{\mu-1} - L_j^{\mu-1}) \\ U_j^\mu &= d_j^\mu + \gamma_j^\mu(U_j^{\mu-1} - d_j^{\mu-1}) \end{aligned} \quad (50)$$

tal que:

$$\gamma_j^\mu = \begin{cases} 0,7 & \text{se } (d_j^\mu - d_j^{\mu-1})(d_j^{\mu-1} - d_j^{\mu-2}) < 0 \\ 1,2 & \text{se } (d_j^\mu - d_j^{\mu-1})(d_j^{\mu-1} - d_j^{\mu-2}) > 0 \\ 1,0 & \text{se } (d_j^\mu - d_j^{\mu-1})(d_j^{\mu-1} - d_j^{\mu-2}) = 0 \end{cases} \quad (51)$$

contanto que:

$$\begin{aligned} d_j^\mu - 10(d_j^{max} - d_j^{min}) &\leq L_j^\mu \leq d_j^\mu - 0,01(d_j^{max} - d_j^{min}) \\ d_j^\mu + 10(d_j^{max} - d_j^{min}) &\geq U_j^\mu \geq d_j^\mu + 0,01(d_j^{max} - d_j^{min}) \end{aligned} \quad (52)$$

E α_j^μ e β_j^μ :

$$\begin{aligned} \alpha_j^\mu &= \max\{d_j^{min}, L_j^\mu + 0,1(d_j^\mu - L_j^\mu), d_j^\mu - 0,5(d_j^{max} - d_j^{min})\} \\ \beta_j^\mu &= \min\{d_j^{max}, U_j^\mu - 0,1(U_j^\mu - d_j^\mu), d_j^\mu + 0,5(d_j^{max} - d_j^{min})\} \end{aligned} \quad (53)$$

em relação a d_j , esses limites são:

$$\begin{aligned} d_j^{min} &\leq d_j \leq d_j^{max} \\ -0,9(d_j^\mu - L_j^\mu) &\leq d_j - d_j^\mu \leq 0,9(U_j^\mu - d_j^\mu) \\ -0,5(d_j^{max} - d_j^{min}) &\leq d_j - d_j^\mu \leq 0,5(d_j^{max} - d_j^{min}) \end{aligned} \quad (54)$$

Esse método de atualização das restrições pode também ser utilizado para o MMA, em conjunto com uma forma de limitação da variação das variáveis de projeto (isto é, um passo máximo constante ou variável).

2.3.5 Outros métodos de otimização topológica

2.3.5.1 Visão geral

Outros métodos de otimização topológica também existem, os mais comuns sendo o BESO e o Level-set, mas também existem outros derivados de métodos de otimização já

existentes. Alguns já apresentam certo espaço na literatura, como no caso do método de campo de fases (*phase-field*) (Sigmund; Maute, 2013), outros apresentam menos espaço ou são mais recentes, como métodos baseados em algoritmos genéticos (Deaton; Grandhi, 2014), baseados no comportamento de seres vivos, como Colônia de Formigas, Enxames de Partículas e Sistemas Imunológicos (Munk; Vio; Steven, 2015), Recozimento Simulado (SHIM; MANOOCHEHRI, 1997), Programação Linear Inteira (Sivapuram; Picelli, 2018), *Moving Morphable Components* (MMCs) (Zhang *et al.*, 2017), *Moving Morphable Void* (MMV) (Zhang *et al.*, 2018), além de métodos baseados em redes neurais (Sosnovik; Oseledets, 2017; White *et al.*, 2019).

2.3.5.2 ESO

O método ESO (*Evolutionary Structural Optimization*) é um método mais simples que o SIMP, baseado na razão da função objetivo de cada elemento em relação ao valor máximo global da função objetivo na mesma iteração. Porém, há a desvantagem de não ser possível garantir que a solução obtida seja a melhor solução possível (Huang; Xie, 2010).

No ESO baseado em níveis de tensão, primeira implementação desenvolvida do método, a cada iteração é removido todo elemento e tal que:

$$\frac{\sigma_e}{(\sigma_e)_{max}} < RR_i, \quad (55)$$

em que σ_e denota tensão de von Mises em um elemento e , $(\sigma_e)_{max}$ a tensão de von Mises máxima da estrutura e RR_i é a razão de rejeição, que, a cada estado de equilíbrio atingido, é aumentado por uma taxa de evolução ER :

$$RR_{i+1} = RR_i + ER, \quad (56)$$

até que o estado ótimo de projeto seja atingido.

2.3.5.3 BESO

A partir do ESO, foi desenvolvido um segundo método, AESO (*Additive Evolutionary Structural Optimization*), de caráter aditivo ao invés de subtrativo. Partindo de uma estrutura mínima, suficiente para conectar a carga ao suporte, o algoritmo adicionava material com base no nível de tensionamento dos elementos já presentes (Munk; Vio; Steven, 2015; Querin; Steven; Xie, 1998). O AESO foi utilizado, em conjunto com o ESO, para desenvolver o BESO (*Bi-directional Evolutionary Structural Optimization*).

Em implementações iniciais do método BESO, a cada iteração elementos poderiam ser removidos ou adicionados conforme a razão de rejeição (RR), para elementos sólidos, ou a razão de inclusão (IR) para elementos vazios, quando comparados à razão entre a sensibilidade do elemento e a sensibilidade máxima. No caso, elementos sólidos com razão

menor que RR seriam removidos, e elementos vazios com razão maior que IR seriam adicionados (Querin; Steven; Xie, 1998; Huang; Xie, 2010).

Implementações mais recentes retornam ao uso da função de flexibilidade usada no método SIMP (Munk; Vio; Steven, 2015; Huang; Xie, 2010). As sensibilidades de flexibilidade de cada elemento são obtidas e suavizadas considerando os elementos vizinhos e a iteração anterior, para prover maior estabilidade ao algoritmo, e ordenadas de forma descendente. Elementos vazios têm suas sensibilidades definidas através desse processo de filtragem das sensibilidades. A razão de volume é utilizada como critério de parada, variando a cada iteração de acordo com uma razão evolucionária (ER).

O volume de cada iteração é utilizado para definir a quantidade final de elementos, e, portanto, o número de elementos a serem removidos ou adicionados. A partir do número de elementos sólidos e vazios e do ordenamento das sensibilidades, define-se as sensibilidades limite de adição e subtração: se forem restar apenas 725 elementos sólidos dentre os 1000 existentes, obtém-se a 725^a menor sensibilidade (Huang; Xie, 2010). Elementos sólidos com sensibilidade menor que a de subtração são removidos, e elementos vazios com sensibilidade maior que a de adição são adicionados.

Embora tenha se desenvolvido utilizando a remoção ou adição completa de material (abordagem conhecida como *hard-kill*, em que não há presença de valores intermediários, apenas alternância entre presença e ausência de um elemento), há também implementações que utilizam métodos de interpolação com variáveis contínuas, como o método SIMP (abordagem *soft-kill*, isto é, com o uso de valores intermediários para as propriedades do material e com um valor mínimo para a variável) (Huang; Xie, 2010; Xia *et al.*, 2018a). A versão *soft-kill* surgiu principalmente devido a um problema específico de otimização desenvolvido por Zhou e Rozvany (2001), no qual o ESO (e, da mesma forma, o BESO) falha em obter um resultado otimizado que mantivesse a conexão com ambos suportes em um sistema sobredeterminado (Xia *et al.*, 2018a).

2.3.5.4 Level-set

O método *Level-set* é um método também comumente aplicado para otimização topológica, sendo originalmente desenvolvido para o estudo de escoamentos multifásicos (Dijk *et al.*, 2013). Ele consiste em utilizar uma função ϕ que defina o contorno de um domínio implicitamente:

$$\begin{cases} \phi(\mathbf{X}) > C \Leftrightarrow \mathbf{X} \in \Omega \\ \phi(\mathbf{X}) = C \Leftrightarrow \mathbf{X} \in \Gamma \\ \phi(\mathbf{X}) < C \Leftrightarrow \mathbf{X} \in (D \setminus \Omega) \end{cases} \quad (57)$$

em que C é uma constante (normalmente $C = 0$), $(D \setminus \Omega)$ representa um espaço vazio e \mathbf{X} representa um ponto do domínio. A função é então discretizada e modificada para receber os parâmetros de otimização. Essa geometria é em seguida mapeada para um modelo

mecânico, utilizando, por exemplo, um funcional f :

$$\int_{\Omega} f d\Omega = \int_D f H(\phi) d\Omega \quad (58)$$

$$H(\phi) = \begin{cases} 0 & \text{para } \phi < 0 \\ 1 & \text{para } \phi \geq 0 \end{cases} \quad (59)$$

Como a função ϕ descreve implicitamente um contorno, o método torna-se essencialmente um tipo de otimização de forma, e, portanto, a topologia é atualizada com base na análise das sensibilidades da forma. Utilizando uma função resposta R que descreve as respostas do domínio e do contorno em função do deslocamento \mathbf{u} :

$$R(\Omega, \mathbf{u}(\Omega)) = \int_{\Omega} \mathbf{f}^T \mathbf{u} d\Omega + \int_{\Gamma} \mathbf{b}^T \mathbf{u} d\Gamma, \quad (60)$$

é possível obter, por exemplo, as sensibilidades de forma discretas ou parametrizadas, para $\phi(\mathbf{X}) = N_i(\mathbf{X})s_i$:

$$\frac{\partial R}{\partial s_i} = \int_{\Gamma} d_S R \frac{N_i}{\|\nabla \phi\|} d\Gamma, \quad (61)$$

em que N_i é a função de forma do contorno, s_i é um parâmetro de otimização do contorno e $d_S R$ é o gradiente de forma da função resposta R .

É possível obter também essas sensibilidades através do campo de densidades, caso ele seja usado para o modelo mecânico:

$$\frac{\partial R}{\partial s_i} = \sum_{e=1}^N \frac{\partial R}{\partial \rho_e} \frac{\partial \rho_e}{\partial s_i}. \quad (62)$$

As sensibilidades de forma são utilizadas para alterar o contorno da topologia. Para alterar a topologia, isto é, adicionar ou remover furos, é necessário o gradiente topológico $d_{\tau} R$:

$$d_{\tau} R = \lim_{r \rightarrow 0} \frac{R(\Omega \setminus B(r)) - R(\Omega)}{|B(r)|}, \quad (63)$$

em que B denota uma esfera de raio r e $|B(r)|$, seu volume. Esse gradiente informa os efeitos de adicionar ou remover uma esfera de raio r na topologia, sendo utilizado para determinar os locais mais propícios para realizar modificações topológicas.

2.3.6 Restrições de tensão

O uso de restrições de tensão ainda traz algumas dificuldades devido ao caráter local e não linear das tensões, além da ocorrência de soluções singulares durante o processo de otimização (isto é, a ocorrência de pontos mais otimizados que não são atingíveis pelo processo de otimização devido ao aumento dos níveis de tensão durante o processo de remoção de um elemento) (Le *et al.*, 2010; Cheng, 1997). Por isso, ainda não há consenso sobre como seria a melhor forma de aplicar essas restrições (Bendsøe; Sigmund, 2013).

Várias abordagens são propostas na literatura para aplicação sobre os diversos métodos de otimização topológica.

Para o método SIMP, Cheng (1997) apresenta um método baseado no relaxamento da restrição, no caso para um problema de minimização do volume de uma estrutura treliçada, mas posteriormente adaptado para estruturas contínuas (Duysinx; Bendsøe, 1998). Le *et al.* (2010) apresentam um método de normalização para restrição global de tensão (para igualá-la progressivamente com a tensão máxima), sendo esta baseada na norma-p, assim como um método de restrição por regiões (um intermediário entre local e global). Senhora *et al.* (2020), Silva *et al.* (2021) utilizam o método aumentado de Lagrange para restrições locais, baseados em Fancello e Pereira (2003) e Pereira, Fancello e Barcellos (2004). Paris *et al.* (2009) apresentam dois métodos, um baseado em restrições locais e outro em uma restrição global (este último baseado na função de Kreisselmeier-Steinhausner). Silva, Cardoso e Beck (2019), Silva, Beck e Sigmund (2019a) apresentam implementações robustas não probabilísticas para compensar incertezas e Silva, Beck e Cardoso (2018) apresentam uma abordagem probabilística. Thillaithevan, Bruce e Santer (2020) apresentam uma abordagem que mistura tensão em microescala e deformação em macroescala por sobreposição linear para a análise de microestruturas treliçadas. Meng *et al.* (2020) apresentam uma implementação para problemas termoelásticos. Além de implementações para mecanismos flexíveis (Silva; Beck; Sigmund, 2020; Leon; Gonçalves; Souza, 2020). Entre outros métodos.

Para o BESO, existe uma implementação baseada em agregação para restrição global (norma-p) (Xia *et al.*, 2018b), uma para restringir a primeira tensão principal (Chen *et al.*, 2021), uma para materiais não lineares (Xu; Han; Zhao, 2020), entre outras.

Além de existirem algoritmos para outros métodos de otimização topológica, como o level-set (Kambampati; Gray; Kim, 2020; Picelli *et al.*, 2018; Song *et al.*, 2020; Nguyen; Kim, 2020) e aprendizado profundo (Deng; To, 2020).

É comum que restrições de tensão sejam utilizadas por si, na forma de problemas de minimização de volume, visto que isso permite redução do volume em conjunto com controle dos níveis de tensão na topologia. Bendsøe e Sigmund (2013), por exemplo, definem esse problema como:

$$\begin{aligned}
 \min_{\rho} \quad & V = \sum_{e=1}^N v_e \rho_e \\
 \text{sujeito a} \quad & \mathbf{K}\mathbf{u} = \mathbf{f} \\
 & \sigma_e \leq \rho_e^p \sigma_{max}^{VM}, \quad 0 < \rho_{min} \leq \rho_e \leq 1, \quad e = 1, \dots, N
 \end{aligned} \tag{64}$$

para casos com restrições globais de tensão, usando o critério de von Mises, em que v_e é o volume do elemento e , σ_e , a tensão von Mises medida no centro do elemento, e

σ_{max}^{VM} , a tensão limite. Para uma restrição de tensão global, isso se torna:

$$\begin{aligned} \min_{\boldsymbol{\rho}} \quad & V = \sum_{e=1}^N v_e \rho_e \\ \text{sujeito a} \quad & \mathbf{K}\mathbf{u} = \mathbf{f} \\ & \sigma_g \leq \sigma_{max}^{VM}, \quad 0 < \rho_{min} \leq \rho_e \leq 1, \quad e = 1, \dots, N \end{aligned} \quad (65)$$

em que σ_g é a medida de tensão global máxima, comumente obtida através de uma função de agregamento como forma de torná-la diferenciável, possibilitando o cálculo do gradiente da restrição. Uma dessas funções é a norma-p (Le *et al.*, 2010):

$$\sigma_g = \left(\sum_{e=1}^N v_e \rho_e^{p_T P} \sigma_e^P \right)^{\frac{1}{P}}, \quad (66)$$

em que p_T é um fator de relaxamento da medida de tensão e o volume é utilizado como fator de escala, e poderia ser substituído, por exemplo, por $\frac{1}{N}$ (Holmberg; Torstenfelt; Klarbring, 2013). Conseqüentemente, o gradiente da norma-p em função da densidade é definido por:

$$\frac{\partial g}{\partial \rho_e} = \frac{\partial \sigma_g}{\partial \rho_e} - \frac{\partial}{\partial \rho_e} [\bar{\mathbf{u}}^T (\mathbf{K}\mathbf{u} - \mathbf{f})], \quad (67)$$

em que $g(\boldsymbol{\rho})$ é a função restrição, tal que:

$$\begin{aligned} \frac{\partial \sigma_g}{\partial \rho_e} &= \frac{1}{P} \left(\sum_{i=1}^N v_i \rho_i^P \sigma_i^P \right)^{\frac{1-P}{P}} \left(p_T P v_e \rho_e^{p_T P - 1} \sigma_e^P + P \sum_{i=1}^N v_i \rho_i^{p_T P} \sigma_i^{P-1} \frac{\partial \sigma_i}{\partial \rho_e} \right) \\ &= \sigma_g^{1-P} \left(p_T v_e \rho_e^{P-1} \sigma_e^P + \sum_{i=1}^N v_i \rho_i^{p_T P} \sigma_i^{P-1} \frac{\partial \sigma_i}{\partial \rho_e} \right). \end{aligned} \quad (68)$$

A derivada $\frac{\partial \sigma_i}{\partial \rho_e}$ pode ser obtida através da regra da cadeia:

$$\frac{\partial \sigma_i}{\partial \rho_e} = \frac{\partial \sigma_i}{\partial \boldsymbol{\sigma}_i} \frac{\partial \boldsymbol{\sigma}_i}{\partial \mathbf{u}_i} \frac{\partial \mathbf{u}_i}{\partial \mathbf{u}} \frac{\partial \mathbf{u}}{\partial \rho_e}. \quad (69)$$

A tensão elementar σ_i é a tensão de von Mises no centro do elemento i , e pode ser calculada utilizando o vetor de tensões no mesmo ponto $\boldsymbol{\sigma}_i$ por (Picelli *et al.*, 2018):

$$\sigma_i = \left(\boldsymbol{\sigma}_i^T \mathbf{V} \boldsymbol{\sigma}_i \right)^{\frac{1}{2}}, \quad (70)$$

em que \mathbf{V} é a matriz de cálculo da tensão equivalente de von Mises. Para o estado plano de tensões, por exemplo:

$$\mathbf{V} = \begin{bmatrix} 1 & -0,5 & 0 \\ -0,5 & 1 & 0 \\ 0 & 0 & 3 \end{bmatrix} \quad (71)$$

Portanto:

$$\frac{\partial \sigma_i}{\partial \boldsymbol{\sigma}_i} = \frac{\partial}{\partial \boldsymbol{\sigma}_i} \left[\left(\boldsymbol{\sigma}_i^T \mathbf{V} \boldsymbol{\sigma}_i \right)^{\frac{1}{2}} \right] = \frac{1}{2} \left(\boldsymbol{\sigma}_i^T \mathbf{V} \boldsymbol{\sigma}_i \right)^{-\frac{1}{2}} 2 \boldsymbol{\sigma}_i^T \mathbf{V} = \frac{\boldsymbol{\sigma}_i^T \mathbf{V}}{\sigma_i}. \quad (72)$$

O vetor de tensões no elemento é obtido por:

$$\boldsymbol{\sigma}_i = \mathbf{C}_i \mathbf{B}_i \mathbf{u}_i, \quad (73)$$

portanto,

$$\frac{\partial \boldsymbol{\sigma}_i}{\partial \mathbf{u}_i} = \frac{\partial}{\partial \mathbf{u}_i} (\mathbf{C}_i \mathbf{B}_i \mathbf{u}_i) = \mathbf{C}_i \mathbf{B}_i. \quad (74)$$

E, por fim:

$$\frac{\partial \mathbf{u}_i}{\partial \mathbf{u}} = \frac{\partial}{\partial \mathbf{u}} (\mathbf{L}_i \mathbf{u}) = \mathbf{L}_i. \quad (75)$$

Portanto,

$$\begin{aligned} \frac{\partial \sigma_i}{\partial \rho_e} &= \frac{\boldsymbol{\sigma}_i^T \mathbf{V}}{\sigma_i} \mathbf{C}_i \mathbf{B}_i \mathbf{L}_i \frac{\partial \mathbf{u}}{\partial \rho_e} \\ \sum_{i=1}^N v_i \rho_i^{p_T P} \sigma_i^{P-1} \frac{\partial \sigma_i}{\partial \rho_e} &= \sigma_g^{1-P} \sum_{i=1}^N v_i \rho_i^{p_T P} \sigma_i^{P-2} [(\mathbf{C}_i \mathbf{B}_i \mathbf{L}_i)^T \mathbf{V} \mathbf{C}_i \mathbf{B}_i \mathbf{u}_i]^T \frac{\partial \mathbf{u}}{\partial \rho_e}, \end{aligned} \quad (76)$$

e com isso,

$$\begin{aligned} \frac{\partial g}{\partial \rho_e} &= \left[\sigma_g^{1-P} \sum_{i=1}^N v_i \rho_i^{p_T P} \sigma_i^{P-2} [\mathbf{L}_i \mathbf{B}_i \mathbf{C}_i \mathbf{V} \mathbf{C}_i \mathbf{B}_i \mathbf{u}_i]^T - \bar{\mathbf{u}}^T \mathbf{K} \right] \frac{\partial \mathbf{u}}{\partial \rho_e} \\ &\quad + p_T \sigma_g^{1-P} v_e \rho_e^{p_T P-1} \sigma_e^P - \bar{\mathbf{u}}^T \frac{\partial \mathbf{K}}{\partial \rho_e} \mathbf{u}. \end{aligned} \quad (77)$$

Para simplificação da implementação, a equação pode ser reduzida a:

$$\frac{\partial g}{\partial \rho_e} = \sigma_g^{1-P} \left[p_T v_e \rho_e^{p_T P-1} \sigma_e^P - p \rho_e^{p-1} \boldsymbol{\lambda}_e^T \mathbf{K}_e^0 \mathbf{u}_e \right], \quad (78)$$

com:

$$\mathbf{K} \boldsymbol{\lambda} = \sum_{i=1}^N v_i \rho_i^{p_T P} \sigma_i^{P-2} \mathbf{L}_i \mathbf{B}_i \mathbf{C}_i \mathbf{V} \mathbf{C}_i \mathbf{B}_i \mathbf{L}_i \mathbf{u}_i, \quad (79)$$

tal que:

$$\boldsymbol{\lambda} = \frac{\bar{\mathbf{u}}}{\sigma_g^{1-P}}. \quad (80)$$

A otimização baseada em níveis de tensão é equivalente à baseada em flexibilidade quando o critério de tensão é consistente com a medida de energia elástica (Pedersen, 1998). Porém, o critério de von Mises não é consistente com essa medida em materiais compressíveis, portanto os resultados podem diferir dos obtidos por minimização de flexibilidade (Bendsøe; Sigmund, 2013).

2.3.7 Outros objetivos

Embora os objetivos mais comuns da otimização topológica sejam a minimização de flexibilidade com restrição de volume e a minimização de volume com restrição de tensão, é possível também encontrar métodos que utilizem múltiplos materiais (Zuo;

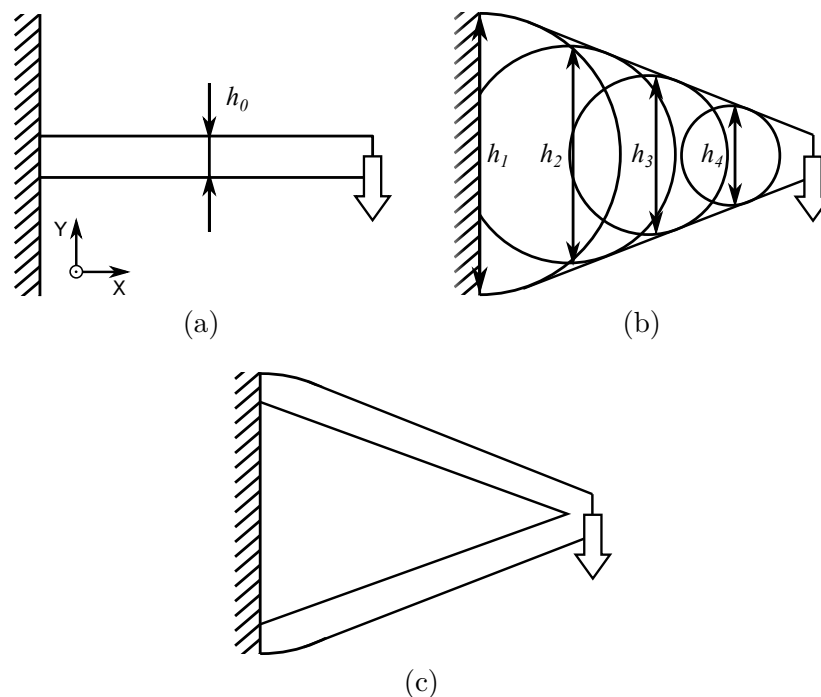
Saitou, 2017), que minimizem danos por fadiga (Holmberg; Torstenfelt; Klarbring, 2014), controlem vibrações (Zargham *et al.*, 2016), lidem com incertezas (Silva; Beck; Sigmund, 2019b; da Silva; Beck; Cardoso, 2017), otimizem a interface fluido-sólido (Alexandersen; Andreasen, 2020) e sistemas de corpos rígidos (Stoeckli; Shea, 2020), sintetizem mecanismos flexíveis (Zhu *et al.*, 2020; Zhang; Zhu, 2018), além de otimizarem topologias para tensão, deformação, flambagem, pressão, propagação de ondas, criação de suportes, entre outras, com a possibilidade de otimização multiobjetivo (Bendsøe; Sigmund, 2013; Huang; Xie, 2010).

3 IMPLEMENTAÇÃO EM AMBIENTE DISCRETIZADO

3.1 Considerações iniciais

A Figura 2 esquematiza o algoritmo implementado tanto em ambiente discretizado (neste capítulo) quanto em ambiente contínuo (Capítulo 4). Primeiramente, as condições de contorno são utilizadas para formar as vigas iniciais, conectando as forças aos suportes através de um algoritmo de *pathfinding*, de forma que estejam contidas na geometria inicial do problema (Figura 2a). Em seguida, é realizada uma análise de elementos finitos sobre a topologia formada pelas vigas e os resultados são utilizados para redimensioná-las, no caso pela criação de discos (em um ambiente bidimensional) ou esferas (em um ambiente tridimensional) a partir do centro da seção transversal (Figura 2b). Este processo de criação e dimensionamento das vigas é denominado *pré-dimensionamento*. Por fim, o resultado é otimizado topologicamente para redução de massa (Figura 2c).

Figura 2 – Representação esquemática do algoritmo a ser implementado: (a) Criação da viga inicial; (b) Modificação da seção transversal da viga de acordo com os resultados da análise de elementos finitos; (c) Otimização topológica do resultado.



Fonte: Elaborado pelo autor.

3.2 Base teórica

Como o método se baseia em conceitos de análise e dimensionamento de vigas, é necessário primeiramente definir esses conceitos e, em seguida, como eles serão aplicados para este caso. Partindo de Hibbeler (2014), assumindo um material isotrópico e tensões máximas de projeto σ_{max} e τ_{max} e considerando um elemento de viga, para os casos de cargas normais, axiais e de flexão respectivamente tem-se que:

$$\sigma = \frac{F_n}{A} \quad (81)$$

$$\tau = \frac{F_c Q}{I t} \quad (82)$$

$$\sigma_f = \frac{M_f}{W_f} \quad (83)$$

Onde τ se refere a tensões de cisalhamento, σ a tensões normais, σ_f a tensões de flexão, F_n à força normal, F_c à força de cisalhamento, M_f ao momento fletor, Q ao primeiro momento de área (momento estático), I ao segundo momento de área (momento de inércia de área), t à largura da seção transversal, A à área da seção transversal e W_f ao módulo de resistência à flexão.

Essas equações, assim como as seguintes, assumem vigas finas, isto é, com dimensões de seção transversal muito menores que seu comprimento, algo que não é necessariamente respeitado durante o processo de dimensionamento com o algoritmo desenvolvido, dependendo dos problemas utilizados, sendo, portanto, uma ferramenta heurística.

Para uma aplicação bidimensional em regime plano de tensões, com espessura constante t e elementos quadrilaterais de lado h , tem-se que:

$$A = ht \quad (84)$$

$$Q = \frac{h^2 t}{8} \quad (85)$$

$$I = \frac{h^3 t}{12} \quad (86)$$

$$W_f = \frac{h^2 t}{6} \quad (87)$$

O objetivo é modificar a dimensão h das vigas para que as tensões calculadas sejam menores que suas respectivas tensões de projeto (Figura 2), isto é, obter h em função da tensão ou das cargas envolvidas. Portanto, este é um problema de dimensionamento.

A partir disso, é possível obter uma dimensão final h_f a partir de uma dimensão inicial h_i em cada ponto, com base nas Equações 81 a 87.

3.3 Programa base

A implementação utilizou como base o programa ToPy, disponível gratuitamente no GitHub sob licença MIT (Hunter *et al.*, 2017; Hunter, 2009). O ToPy é uma implementação do algoritmo de minimização de flexibilidade utilizando o método dos elementos finitos, SIMP, OC e filtragem de sensibilidades, como descrito por Sigmund (2001). A linguagem utilizada para a implementação é o Python 2 (Millman; Aivazis, 2011), em conjunto com as bibliotecas NumPy (Harris *et al.*, 2020), SymPy (Meurer *et al.*, 2017), PySparse (Geus, 2002), entre outras. A implementação suporta minimização de flexibilidade para problemas de elasticidade linear, de condução de calor e síntese de mecanismos flexíveis, tanto em duas quanto em três dimensões, utilizando elementos quadrilaterais ou hexagonais regulares. Há suporte também para a definição de elementos ativos (que devem obrigatoriamente se manter preenchidos) e passivos (que devem se manter vazios). Uma descrição mais detalhada do programa original se encontra no Apêndice A.

O programa foi escolhido por ser uma implementação compacta, ainda que completa, da otimização topológica por SIMP, feita com base em códigos de MatLab desenvolvidos por pesquisadores da área, especialmente a versão de 99 linhas desenvolvida por Sigmund (2001) (Hunter, 2009). O fato de ser uma implementação antiga, já testada e já avaliada academicamente torna mais seguros os resultados das otimizações topológicas.

O Python, embora seja uma linguagem interpretada, vem ganhando mais espaço na academia devido à sua facilidade de aprendizado e uso, sua flexibilidade e a disponibilidade de bibliotecas voltadas para a matemática, física e engenharia (Millman; Aivazis, 2011), além de ser disponibilizado gratuitamente.

3.4 Implementação do algoritmo

3.4.1 Modificações do programa base

Como o Python 2 está atualmente obsoleto e a biblioteca PySparse não é mais mantida e está limitada ao Python 2, foi necessário atualizar o código do programa para facilitar seu uso e manutenção. No caso, o programa foi modificado para se adequar aos padrões do Python 3 e a biblioteca PySparse foi substituída por outra que realizasse suas funções, isto é, a criação de matrizes esparsas e a solução de problemas que as envolvam. Para isso, foram escolhidas as bibliotecas SciPy (Virtanen *et al.*, 2020) e MUMPS (Amestoy *et al.*, 2001; Amestoy *et al.*, 2019).

A biblioteca MUMPS é uma biblioteca especializada em resolver equações matriciais com matrizes esparsas de forma direta, utilizando também multiprocessamento para aumentar a velocidade de resolução. A biblioteca SciPy foi utilizada principalmente para armazenar e modificar matrizes esparsas. Algumas otimizações também foram feitas utilizando a biblioteca NumExpr (McLeod *et al.*, 2018).

Como o algoritmo desenvolvido por esta pesquisa não é uma modificação de algum algoritmo de otimização topológica, mas sim um complemento idealizado para agir antes do mesmo, ele foi implementado modificando o menos possível a implementação já presente, buscando garantir a validade dos resultados, além de preservar a implementação original para fins de comparação.

3.4.2 Simplificações adotadas

Idealmente, busca-se utilizar vigas contínuas com seções transversais iniciais quaisquer. Porém, o ToPy utiliza ambientes previamente discretizados com elementos equiláteros. Portanto, a implementação adotou duas simplificações para se adaptar a isso:

1. Toda seção de viga apresentará uma seção transversal com as mesmas dimensões (isto é, retangular $t \times h_i$ ou quadrada $h_i \times h_i$); e
2. Os dimensionamentos se baseiam nas tensões elementares (especificamente, a máxima tensão dentre os nós de cada elemento).

Isso limita consideravelmente as aplicações da implementação, mas seus resultados são suficientes para examinar a viabilidade do algoritmo, sendo, portanto, uma prova de conceito. Destaca-se dois problemas causados pelos dois itens, respectivamente:

1. Vigas diagonais apresentarão rigidez diferente de vigas horizontais e verticais; e
2. Não é possível distinguir entre tensão normal originada de tração/compressão e originada de flexão, portanto, por segurança, se dimensionará assumindo apenas tração/compressão, resultando em topologias sobredimensionadas no caso de estruturas que apresentem flexão. No caso tridimensional, isso também significa assumir tensões de torção como tensões de cisalhamento.

Além disso, cargas aplicadas a vigas com seção transversal maior do que um elemento podem resultar em topologias subdimensionadas caso as cargas não estejam todas alinhadas perpendicularmente aos suportes, pois o fato das vigas iniciais apresentarem maiores dimensões resulta em grandes diminuições dos níveis de tensão.

Considerando essas simplificações, denominando as tensões calculadas por elementos finitos como σ_{fea} (tensões normais) e τ_{fea} (tensões de cisalhamento), e partindo de uma dimensão inicial $(h_i)_e$ de cada elemento de viga, busca-se obter uma nova dimensão $(h_f)_e$ para obter uma seção transversal mais adequada aos carregamentos.

Portanto, de 81 e 84 tem-se que:

$$\sigma_{fea} = \frac{F_n}{(h_i)_{et}}, \quad (88)$$

o que implica em

$$F_n = (h_i)_e t \sigma_{fea}. \quad (89)$$

Aplicando-se as mesmas equações para σ_{max} e considerando F_n , resulta em:

$$\sigma_{max} = \frac{F_n}{(h_f)_e t} = \frac{(h_i)_e \sigma_{fea}}{(h_f)_e}, \quad (90)$$

portanto,

$$(h_f)_e = (h_i)_e \frac{\sigma_{fea}}{\sigma_{max}}. \quad (91)$$

E de 82, 85 e 86, para a tensão de cisalhamento, similarmente resulta em:

$$\tau_{fea} = \frac{F_c \frac{(h_i)_e^2 t}{8}}{\frac{(h_i)_e^2 t}{12}} = \frac{3F_c}{2(h_i)_e t}. \quad (92)$$

portanto,

$$F_c = \frac{2(h_i)_e t \tau_{fea}}{3}. \quad (93)$$

Para τ_{max} :

$$\tau_{max} = \frac{3F_c}{2(h_f)_e t} = \frac{3 \frac{2(h_i)_e t \tau_{fea}}{3}}{2(h_f)_e t} = \frac{(h_i)_e \tau_{fea}}{(h_f)_e}, \quad (94)$$

e a nova dimensão é calculada por:

$$(h_f)_e = (h_i)_e \frac{\tau_{fea}}{\tau_{max}}. \quad (95)$$

Portanto, para um estado plano de tensões e um elemento e :

$$(h_f)_e = (h_i)_e \max \left\{ \frac{|(\sigma_x)_e|}{\sigma_{max}}, \frac{|(\sigma_y)_e|}{\sigma_{max}}, \frac{|(\tau_{xy})_e|}{\tau_{max}} \right\} \quad (96)$$

Similarmente, para análises tridimensionais com elementos hexagonais de lado $(h_i)_e$ (portanto, $t = (h_i)_e$ em 84, 85 e 86):

$$(A_e)_f = \frac{\pi (h_f)_e^2}{4} \quad (97)$$

$$(Q_e)_f = \frac{(h_f)_e^3}{12} \quad (98)$$

$$(I_e)_f = \frac{\pi (h_f)_e^4}{64} \quad (99)$$

Portanto, para tensões normais:

$$\begin{aligned} \sigma_{fea} &= \frac{F_n}{(h_i)_e^2} \Rightarrow F_n = (h_i)_e^2 \sigma_{fea} \\ \sigma_{max} &= \frac{F_n}{\frac{\pi (h_f)_e^2}{4}} = \frac{4(h_i)_e^2 \sigma_{fea}}{\pi (h_f)_e^2} \\ &\Rightarrow (h_f)_e = (h_i)_e \sqrt{\frac{4\sigma_{fea}}{\pi \sigma_{max}}} \end{aligned} \quad (100)$$

E para tensões de cisalhamento:

$$\begin{aligned}\tau_{fea} &= \frac{F_c \frac{(h_i)_e^3}{8}}{\frac{(h_i)_e^4}{12} (h_i)_e} = \frac{3V}{2(h_i)_e^2} \Rightarrow F_c = \frac{2(h_i)_e^2 \tau_{fea}}{3} \\ \tau_{max} &= \frac{F_c \frac{(h_f)_e^3}{12}}{\frac{\pi(h_f)_e^4}{64} (h_f)_e} = \frac{16F_c}{3\pi(h_f)_e^2} = \frac{32(h_i)_e^2 \tau_{fea}}{9\pi(h_f)_e^2} \\ \Rightarrow (h_f)_e &= (h_i)_e \sqrt{\frac{32\tau_{fea}}{9\pi\tau_{max}}}\end{aligned}\quad (101)$$

Portanto, para um elemento e :

$$(h_f)_e = (h_i)_e \max \left\{ \sqrt{\frac{4 \max\{ |(\sigma_x)_e|, |(\sigma_y)_e|, |(\sigma_z)_e| \}}{\pi\sigma_{max}}}, \sqrt{\frac{32 \max\{ |(\tau_{xy})_e|, |(\tau_{xz})_e|, |(\tau_{yz})_e| \}}{9\pi\tau_{max}}} \right\} \quad (102)$$

Como a implementação se baseia em um espaço discretizado, com elementos quadriláteros ou hexagonais regularmente distribuídos, as Equações 96 e 102 podem ser reduzidas para:

$$N_e = \left\lceil \max \left\{ \frac{|(\sigma_x)_e|}{\sigma_{max}}, \frac{|(\sigma_y)_e|}{\sigma_{max}}, \frac{|(\tau_{xy})_e|}{\tau_{max}} \right\} \right\rceil \quad (103)$$

$$N_e = \left\lceil \max \left\{ \sqrt{\frac{4 \max\{ |(\sigma_x)_e|, |(\sigma_y)_e|, |(\sigma_z)_e| \}}{\pi\sigma_{max}}}, \sqrt{\frac{32 \max\{ |(\tau_{xy})_e|, |(\tau_{xz})_e|, |(\tau_{yz})_e| \}}{9\pi\tau_{max}}} \right\} \right\rceil \quad (104)$$

Em que N_e é o diâmetro do círculo ou esfera a ser formado, mas medido em número de elementos, arredondado para o próximo número par superior, para garantir que o algoritmo de preenchimento dos elementos não o arredonde para baixo durante a criação da geometria devido à discretização do espaço.

Ao gerar essa nova topologia, o espaço de projeto é então limitado a ela, resultando em um menor número de elementos a serem considerados em cada iteração. Isso permite a redução dimensional da topologia final, e também espera-se que resulte em menor tempo de otimização, visto que envolve menos elementos e o processo de pré-dimensionamento não é iterativo. Por outro lado, isso introduz uma restrição ao espaço de solução, o que significa que o resultado obtido posteriormente por otimização topológica apresentará mínimo global no máximo tão otimizado quanto o que poderia ser obtido por otimização topológica pura.

3.4.3 *Pathfinding*

Como o espaço utiliza uma malha regular, o algoritmo de *pathfinding* A^* é o mais apropriado, não somente por ser otimizado para esses casos como por ser relativamente simples de se implementar. É um algoritmo heurístico, cujo processo de tomada de decisão baseia-se em uma função escalar heurística, cujo valor em cada nó é denominado “custo”. O algoritmo como descrito por Hart, Nilsson e Raphael (1968) pode ser resumido em:

1. Obter o elemento que apresenta menor custo e removê-lo da lista de elementos;
2. Obter todos os elementos vizinhos e seus respectivos custos; e
3. Adicionar esses elementos à lista, associando-os ao elemento atual.

A sequência é repetida até que o destino seja alcançado, partindo de um elemento ou conjunto de elementos iniciais pré-determinados. O caminho final é então obtido seguindo-se o caminho inverso, partindo do elemento que atingiu o objetivo. A função custo neste caso é a menor distância a um elemento ativo (se houver um a ser alcançado) ou a um suporte, somado à distância de um elemento a outro. Nós de suporte vizinhos são considerados um único suporte.

Para ser compatível com o método dos elementos finitos, apenas os elementos presentes nos lados de cada elemento foram considerados seus vizinhos (isto é, elementos diagonais não foram considerados). Conseqüentemente, seções diagonais serão mais rígidas que seções não diagonais.

3.5 Validação

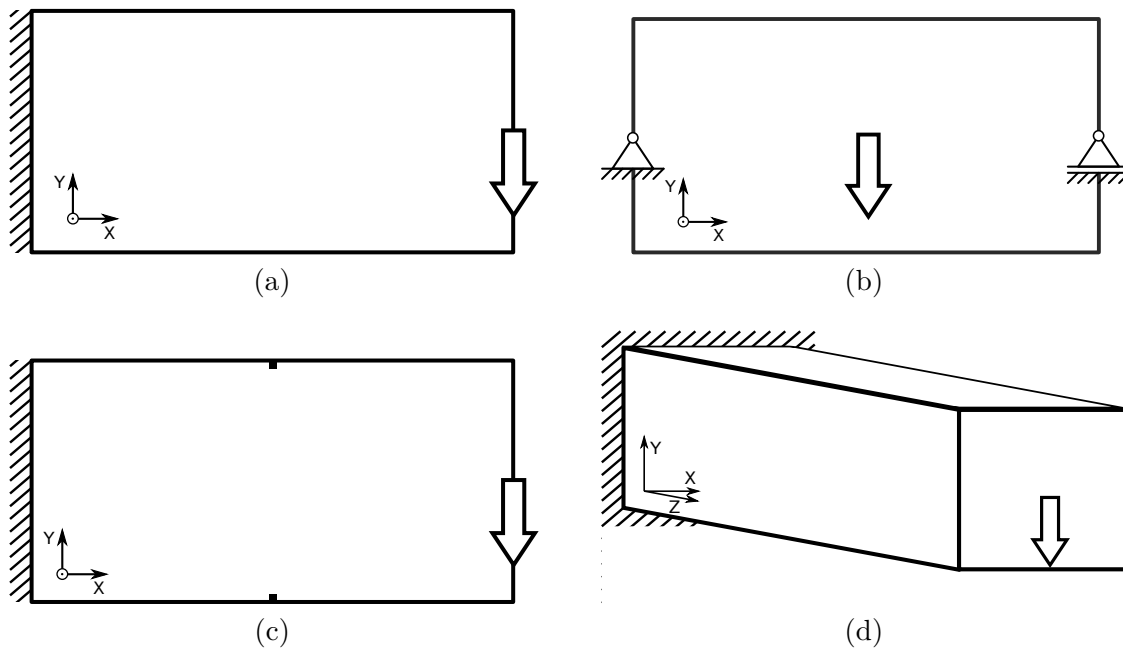
Para validar os resultados, a topologia final (resultante do pré-dimensionamento seguido pela otimização topológica) foi utilizada como base para criar uma topologia menos grosseira no software SolidEdge ST8 e foi testada no Ansys® Academic Research Mechanical, Release 18.2 (ANSYS, Inc., 2018) utilizando as mesmas condições de contorno.

3.6 Estudos de caso

O programa foi testado com dois problemas comuns na literatura (um deles tanto em 2D quanto em 3D) e um adicional, feito para explorar o uso de elementos ativos (isto é, elementos que devem se manter na geometria ao fim do processo de otimização) e sua influência na topologia final, assim como o efeito de vigas diagonais na rigidez da estrutura antes do redimensionamento. O material utilizado como base para os testes foi o AISI 1020 (MatWeb, LLC,). O material e outros parâmetros utilizados nos quatro testes estão presentes nas Tabelas 1 e 2. Parâmetros específicos para cada teste estão presentes na Tabela 3.

O Caso 1 é o problema da viga engastada (Figura 3a). O Caso 2, o ensaio de flexão em três pontos (Figura 3b). O Caso 3 é similar ao Caso 1, mas há dois elementos ativos presentes na metade da largura da estrutura, um em cada extremo do eixo Y (Figura 3c). O Caso 4 é o problema da viga engastada em 3D (Figura 3d), usado para mostrar as diferenças originadas da diferente forma de calcular $(h_f)_e$ (especialmente o formato quadrático resultante dela).

Figura 3 – Problemas resolvidos pelos Casos 1 a 4: (a) Caso 1; (b) Caso 2; (c) Caso 3; (d) Caso 4.



Fonte: Elaborado pelo autor.

As intensidades de carga foram calibradas para que as topologias preliminares resultantes pudessem ocupar o máximo de espaço possível, sem permitir que os círculos e esferas gerados tivessem seus diâmetros maiores que alguma das dimensões do espaço de projeto, e com isso permitir uma melhor comparação com seus equivalentes da otimização topológica tradicional. Os Casos 1 e 2 também foram realizados utilizando diferentes intensidades de carga para mostrar as diferentes topologias resultantes.

Embora o método de otimização topológica utilizado seja de minimização de flexibilidade, visto que é o método implementado pelo ToPy, as análises se baseiam também nos níveis de tensão obtidos ao fim do processo de otimização topológica, pois o redimensionamento é feito com base nos níveis de tensão. Como consequência, as frações de volume escolhidas para os testes (Tabela 3) são as menores frações de volume que resultam em tensões que se adequam aos limites de projeto, considerando o efeito das concentrações de tensões e a resolução da malha utilizada. Visando aprimorar essa análise, a implementação desenvolvida posteriormente (Capítulo 4), utiliza minimização de volume com restrições de tensão.

Tabela 1 – Parâmetros de material comuns a todos os testes.

Parâmetro	Valor
Módulo de elasticidade	186 GPa
Coefficiente de Poisson	0,29
Máxima tensão normal (σ_{max})	100 MPa
Máxima tensão de cisalhamento (τ_{max})	100 MPa

Fonte: Elaborado pelo autor.

Tabela 2 – Parâmetros de otimização comuns a todos os testes.

Parâmetro	Casos 2D	Casos 3D
Tolerância	0.001	0.001
p inicial	3	1
Incremento de p	-	0,1
p máximo	-	3
q inicial	2,5	1
Incremento de q	-	0,05
q máximo	-	5
Iterações com p e q constantes	-	5
η	0,5	0,4
Raio do filtro	2	1,8

Nota: O ToPy apresenta uma implementação modificada do método do critério de ótimo que permite a variação da penalização p , controle da velocidade de variação da densidade (pelo fator de amortecimento η) e a definição de um fator de filtro de escala de cinza q (ver Apêndice A).

Fonte: Elaborado pelo autor.

Tabela 3 – Parâmetros específicos de cada teste.

Parâmetro	Caso 1	Caso 2	Caso 3	Caso 4
Largura do elemento (L) (mm)	20	20	20	50
Espessura (t) (mm)	50	50	50	-
Largura (eixo X) (elementos)	200	300	200	20
Altura (eixo Y) (elementos)	150	200	150	20
Profundidade (eixo Z) (elementos)	-	-	-	50
Tipo de elemento	Q4	Q4	Q4	H8
Carga (kN)	15	50	100	250
Fração de volume	0.10	0.10	0.20	0.10

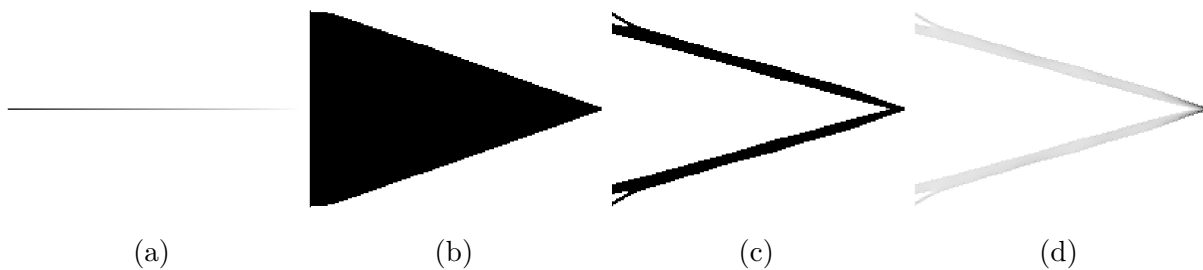
Fonte: Elaborado pelo autor.

3.7 Resultados

3.7.1 Caso 1

A Figura 4 mostra os resultados principais do Caso 1. O nível máximo de tensão von Mises medido na topologia final foi de 30,7 MPa. Porém, esse nível de tensão é causado predominantemente pela concentração de tensão no entorno do nó no qual a carga foi aplicada, como mostra a Figura 6, assim como por concentrações de tensão causadas pela qualidade da malha. Porém, a distribuição de tensão também mostra que os níveis de tensão são bem menores fora dessa região. Como $(\sigma_e)_{max} < \sigma_{max}, \tau_{max}$, portanto $(\sigma_x)_{max}, (\sigma_y)_{max} < \sigma_{max}$ e $(\tau_{xy})_{max} < \tau_{max}$, o que significa que o resultado respeitou as tensões máximas de projeto, embora a topologia resultante ainda possa ser considerada sobredimensionada. Inclusive, pela Figura 6 é possível notar que fora da região de singularidade o nível de tensão von Mises é menor que 6,4 MPa. Seria possível utilizar uma menor fração de volume para obter melhores resultados, mas isso necessitaria uma malha mais refinada, visto que a fração de volume utilizada já é reduzida (10%).

Figura 4 – Resultados do Caso 1. A distribuição de tensão vai de maior nível de tensão von Mises (em preto) para menor nível (em branco): (a) Viga resultante (distribuição de tensão); (b) Topologia inicial; (c) Topologia final; (d) Topologia final (distribuição de tensão).

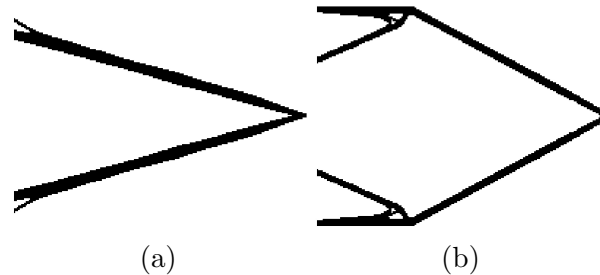


Fonte: Elaborado pelo autor.

A Figura 5 compara as topologias resultantes com e sem o pré-dimensionamento, utilizando os mesmos parâmetros de teste. Embora a topologia resultante do pré-dimensionamento ainda seja consideravelmente sobredimensionada, seria esperado que, sem isso, a topologia resultante fosse muito mais sobredimensionada, especialmente no caso de cargas de menor intensidade. Embora o volume resultante seja o mesmo, as dimensões de ambas as topologias resultantes são consideravelmente diferentes. Para melhor evidenciar essa diferença, a Figura 7 mostra os resultados do mesmo teste utilizando cargas de menores intensidades (conforme indicadas na legenda da figura).

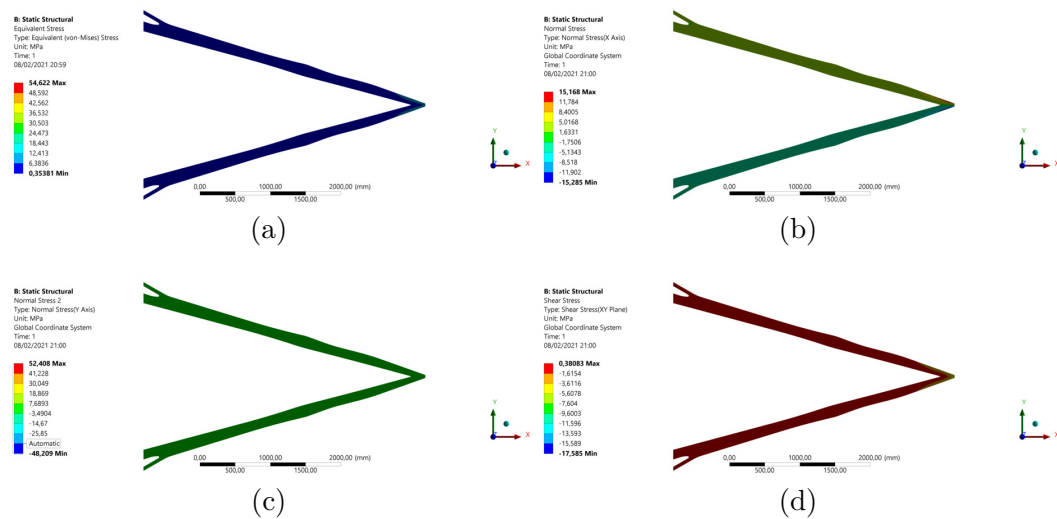
A Tabela 4 mostra uma comparação do tempo necessário para realizar o processo completo de otimização, com e sem pré-dimensionamento. Mesmo com a adição do pré-dimensionamento e buscando preencher o espaço de projeto, o tempo final de otimização foi consideravelmente menor que o da otimização pura. A redução no tempo por iteração

Figura 5 – Comparação das topologias resultantes com e sem pré-dimensionamento no Caso 1: (a) Com pré-dimensionamento. A tensão von Mises máxima foi de 30,7 MPa. A flexibilidade foi de 5,412 N · mm; (b) Sem pré-dimensionamento. A tensão von Mises máxima foi de 18,7 MPa. A flexibilidade foi de 4,172 N · mm.



Fonte: Elaborado pelo autor.

Figura 6 – Análise de elementos finitos realizada sobre modelo CAD aproximado da topologia resultante do Caso 1: (a) Tensão von Mises (máximo: 54,622 MPa, mínimo: 0,354 MPa); (b) Tensão normal (eixo X) (máximo: 15,168 MPa, mínimo: -15,285 MPa); (c) Tensão normal (eixo Y) (máximo: 52,408 MPa, mínimo: -48,209 MPa); (d) Tensão de cisalhamento (plano XY) (máximo: 0,381 MPa, mínimo: -17,585 MPa).



Fonte: Elaborado pelo autor (imagens geradas utilizando software de ANSYS Inc.).

é consequência da limitação da análise aos elementos definidos no processo de redimensionamento, significando que uma quantidade considerável de elementos (aproximadamente 50% neste caso) é retirada do processo de análise e otimização.

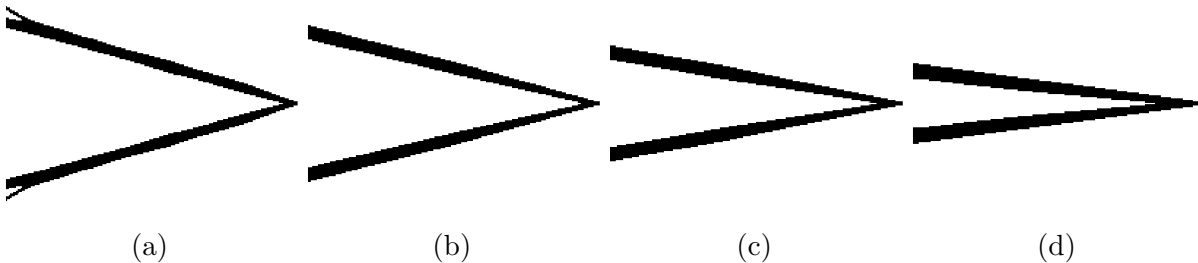
A flexibilidade resultante da otimização pura foi de 4,172 N · mm, enquanto a da otimização com pré-dimensionamento resultou em 5,412 N · mm, isto é, 29,7% maior. Porém, o resultado foi obtido em um tempo 79,5% menor. Embora a flexibilidade seja maior, o processo de pré-dimensionamento aparenta resultar em uma boa aproximação do resultado da otimização topológica pura (considerando dimensões finais semelhantes)

Tabela 4 – Comparação do tempo de resolução para o Caso 1 com e sem pré-dimensionamento.

	Total (min)	Num it.	Por it. (s)	Pré-dim. (s)
Sem	23,484	186	7,576	-
Com	4,804	66	4,167	13,244

Fonte: Elaborado pelo autor.

Figura 7 – Caso 1 com diferentes intensidades de carga: (a) 15 kN. A tensão von Mises máxima foi de 30,7 MPa. A flexibilidade foi de 5,412 N · mm; (b) 12 kN. A tensão von Mises máxima foi de 32,5 MPa. A flexibilidade foi de 5,119 N · mm; (c) 9 kN. A tensão von Mises máxima foi de 30,5 MPa. A flexibilidade foi 5,155 N · mm; (d) 6 kN. A tensão von Mises máxima foi de 32,7 MPa. A flexibilidade foi de 5,602 N · mm.



Fonte: Elaborado pelo autor.

com a diferença de envolver menor quantidade de elementos na análise, resultando em um tempo de resolução muito menor.

3.7.2 Caso 2

A Figura 8 mostra os resultados principais do Caso 2. A topologia resultante do Caso 2, de forma similar à do Caso 1, também apresentou singularidades de tensão no entorno do nó que recebeu a aplicação da carga. Mas, como os suportes também são nodais, há singularidades no entorno de seus nós (Figura 8). Similarmente, ela também resultou em tensões von Mises menores que a de ambos os níveis máximos de projeto. Essas características também podem ser notadas nas análises feitas no ANSYS (Figura 9), e o sobredimensionamento também está presente. O processo de otimização com pré-dimensionamento também é consideravelmente mais rápido do que sem ele (Tabela 5).

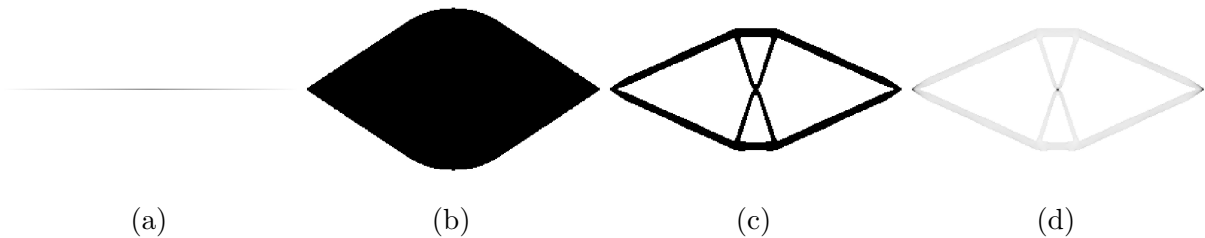
De forma semelhante ao Caso 1, a topologia resultante do processo com pré-dimensionamento apresentou flexibilidade 29,4% maior que a sem pré-dimensionamento, porém, o resultado foi obtido em um tempo 70,8% menor, significando, como no Caso 1, uma redução de aproximadamente três quartos do tempo ao custo de uma flexibilidade um quarto maior.

Tabela 5 – Comparação do tempo de resolução para o Caso 2 com e sem pré-dimensionamento.

	Total (min)	Num it.	Por it. (s)	Pré-dim. (s)
Sem	18,204	70	15,603	-
Com	5,308	38	7,933	17,025

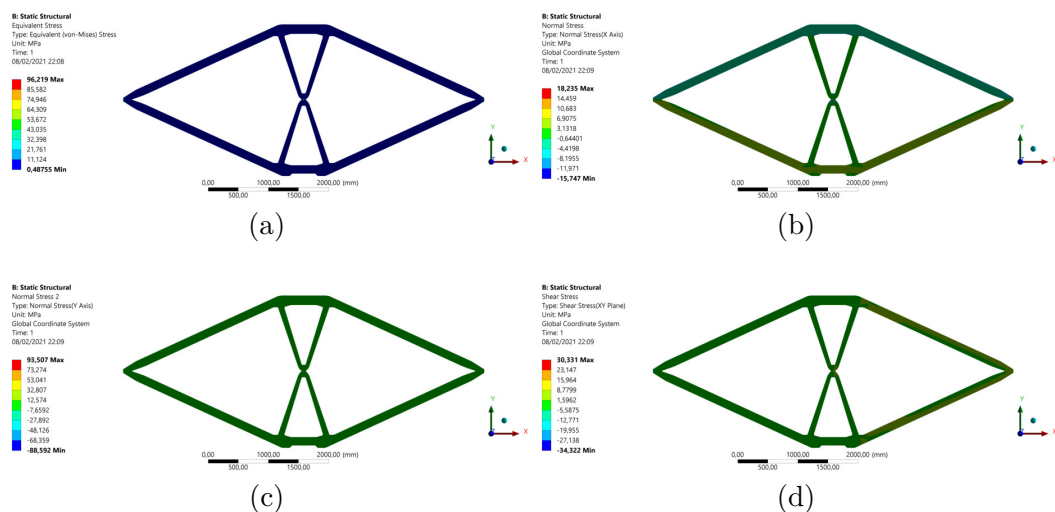
Fonte: Elaborado pelo autor.

Figura 8 – Resultados do Caso 2. A distribuição de tensão vai de maior nível de tensão von Mises (em preto) para menor nível (em branco): (a) Viga resultante (distribuição de tensão); (b) Topologia inicial; (c) Topologia final; (d) Topologia final (distribuição de tensão).



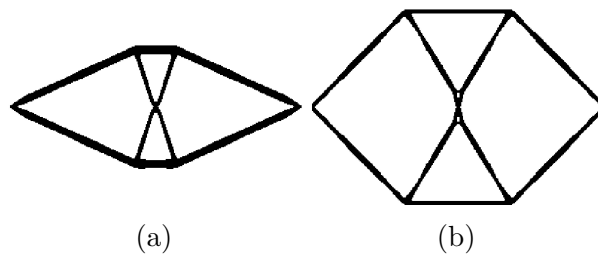
Fonte: Elaborado pelo autor.

Figura 9 – Análise de elementos finitos realizada sobre modelo CAD aproximado da topologia resultante do Caso 2: (a) Tensão von Mises (máximo: 96,219 MPa, mínimo: 0,488 MPa); (b) Tensão normal (eixo X) (máximo: 18,235 MPa, mínimo: -15,747 MPa); (c) Tensão normal (eixo Y) (máximo: 93,507 MPa, mínimo: -88,592 MPa); (d) Tensão de cisalhamento (plano XY) (máximo: 30,331 MPa, mínimo: -34,322 MPa).



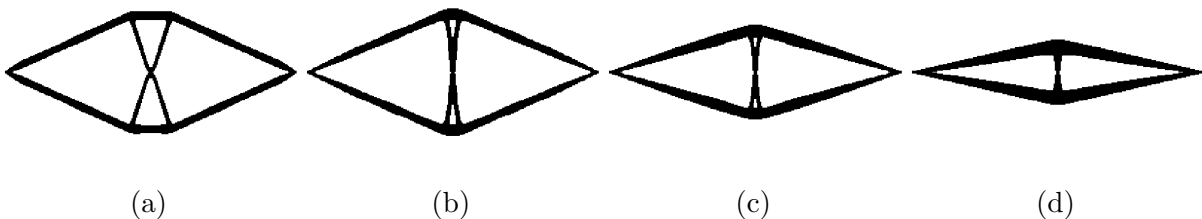
Fonte: Elaborado pelo autor (imagens geradas utilizando software de ANSYS Inc.).

Figura 10 – Comparação das topologias resultantes com e sem pré-dimensionamento no Caso 2: (a) Com pré-dimensionamento. A tensão von Mises máxima foi de 43,4 MPa. A flexibilidade foi de 10,733 N · mm; (b) Sem pré-dimensionamento. A tensão von Mises máxima foi de 31,9 MPa. A flexibilidade foi de 8,294 N · mm.



Fonte: Elaborado pelo autor.

Figura 11 – Caso 2 com diferentes intensidades de carga: (a) 50 kN. A tensão von Mises máxima foi de 43,4 MPa. A flexibilidade foi de 10,733 N · mm; (b) 40 kN. A tensão von Mises máxima foi de 34,6 MPa. A flexibilidade foi de 8,282 N · mm; (c) 30 kN. A tensão von Mises máxima foi de 30,7 MPa. A flexibilidade foi de 7,531 N · mm; (d) 20 kN. A tensão von Mises máxima foi de 33,9 MPa. A flexibilidade foi de 7,844 N · mm.



Fonte: Elaborado pelo autor.

3.7.3 Caso 3

A Figura 12 mostra os resultados principais do Caso 3. O Caso 3 se diferenciou dos anteriores por resultar em um nível de tensão von Mises máximo de 125,6 MPa e, conseqüentemente, o raciocínio usado na Subseção 3.7.1 não se aplica neste caso. Naturalmente, isso é consequência de singularidades de tensão, e as simulações realizadas no ANSYS revelam que fora das singularidades as tensões se mantêm abaixo das de projeto (Figura 13), havendo também sobredimensionamento.

É possível notar que as simulações do ANSYS não apresentam uma distribuição de tensões totalmente similar à gerada pelo ToPy, em especial quanto à ausência de concentração de tensão nos entalhes presentes na estrutura. Isso provavelmente resulta dos níveis mais elevados de tensão obtidos pela malha mais refinada utilizada na simulação (uma consequência da concentração de tensão), resultando em faixas de tensão mais largas e menos detalhadas, em conjunto com possíveis imprecisões resultantes da topologia ter sido refeita manualmente no software CAD.

Como foi dito na Subseção 3.4, havia o risco de que a adição de seções diagonais às vigas nesta implementação resultassem em seções com maior rigidez que o ideal. A Figura 14 foi gerada para analisar essa hipótese: a viga inicial criada no Caso 3 foi refeita no Solid Edge ST8 para analisar como essas seções diagonais impactariam o algoritmo. Como a figura mostra, não há concentração de tensão significativa no entorno do ponto de interseção das vigas diagonais, ao contrário do que é mostrado na Figura 12a. A análise feita no modelo CAD usou uma malha com dimensões similares, e mesmo assim resultou em uma tensão von Mises máxima quatro vezes maior, o que significa que o aumento da rigidez resultante nessa implementação é significativo, mesmo para malhas mais refinadas.

É importante notar, porém, que embora os elementos ativos tenham influenciado efetivamente o formato da viga inicial e, conseqüentemente, o formato da estrutura final, o método não conseguiu manter os elementos ativos como parte da estrutura.

Figura 12 – Resultados do Caso 3. A distribuição de tensão vai de maior nível de tensão von Mises (em preto) para menor nível (em branco): (a) Viga resultante (distribuição de tensão); (b) Topologia inicial; (c) Topologia final; (d) Topologia final (distribuição de tensão).

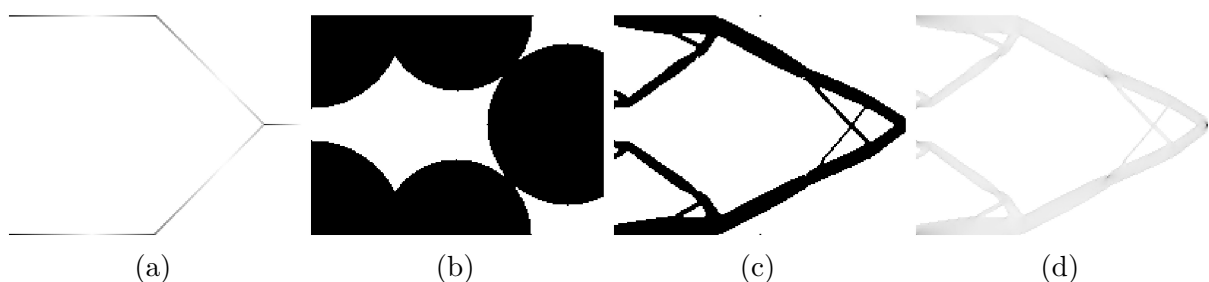
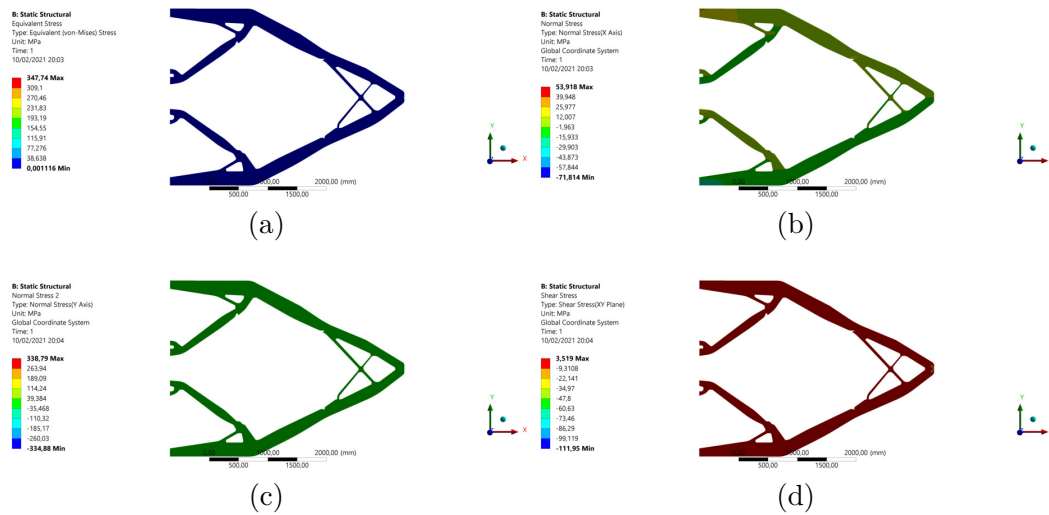
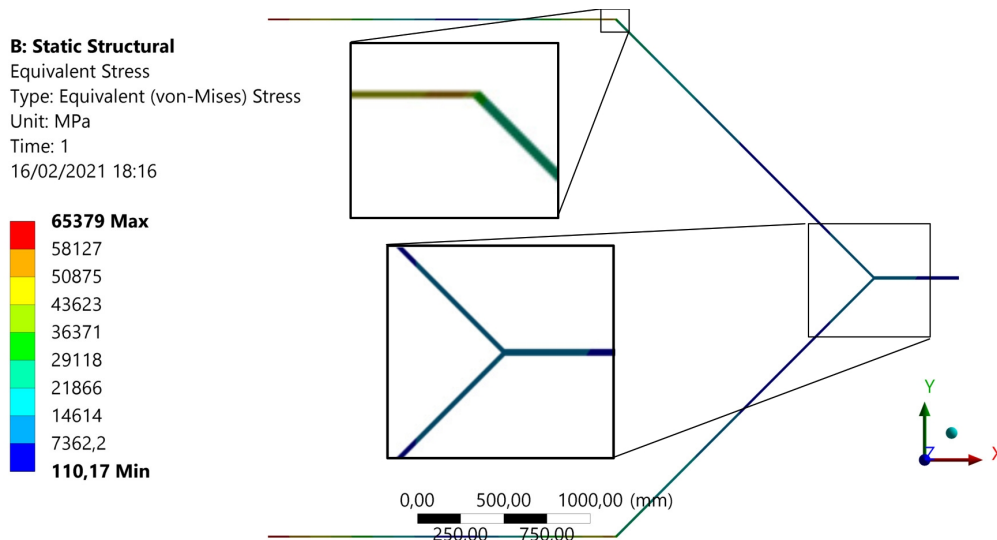


Figura 13 – Análise de elementos finitos realizada sobre modelo CAD aproximado da topologia resultante do Caso 3: (a) Tensão von Mises (máximo: 347,74 MPa, mínimo: 0,001 MPa); (b) Tensão normal (eixo X) (máximo: 53,918 MPa, mínimo: -71,814 MPa); (c) Tensão normal (eixo Y) (máximo: 338,79 MPa, mínimo: -334,88 MPa); (d) Tensão de cisalhamento (plano XY) (máximo: 3,519 MPa, mínimo: -111,95 MPa).



Fonte: Elaborado pelo autor (imagens geradas utilizando software de ANSYS Inc.).

Figura 14 – Análise de elementos finitos realizada sobre modelo CAD aproximado da topologia presente na Figura 12a. A análise também utilizou o método linear de elementos finitos. É possível notar a ausência de concentração de tensão significativa na região de interseção das duas vigas diagonais.



Fonte: Elaborado pelo autor (imagens geradas utilizando software de ANSYS Inc.).

3.7.4 Caso 4

Como esperado, o Caso 4 resultou em uma topologia com perfil quadrático (Figura 15). Isso também resultou em um preenchimento maior do espaço de projeto, mas a velocidade de otimização ainda foi similar à dos Casos 1 e 2 (Tabela 6). A viga inicial mostra uma distribuição de tensão peculiar (Figura 15a), mas isso pode ser um artefato do renderizador, visto que não impactou a geração da geometria intermediária (Figura 15b).

Tabela 6 – Comparação do tempo de resolução para o Caso 4 com e sem pré-dimensionamento.

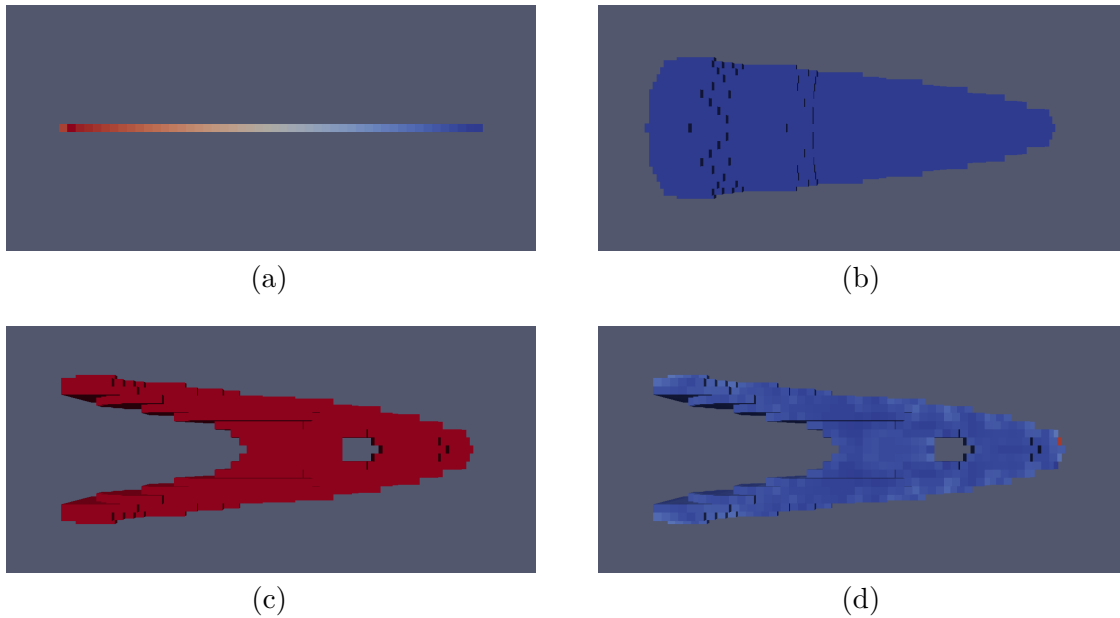
	Total (min)	Num it.	Por it. (s)	Pré-dim. (s)
Sem	47,454	46	61,896	-
Com	17,221	43	23,289	31,826

Fonte: Elaborado pelo autor.

Similarmente ao Caso 3, o máximo nível de tensão von Mises foi maior que ambas as tensões de projeto (176,1 MPa), mas a distribuição de tensão mostra que, fora da vizinhança da carga nodal, as tensões têm intensidade muito menor. Neste caso, porém, o método de otimização topológica tradicional resultou em menores níveis máximos de tensão von Mises, mas as mesmas observações feitas nos outros testes também se aplicam neste caso: a topologia apresenta maiores dimensões (ainda que apresente o mesmo volume) e também precisaria de muito mais tempo para ser obtida (especialmente quando utilizando cargas de menores intensidades), pois envolve a análise de uma maior quantidade de elementos.

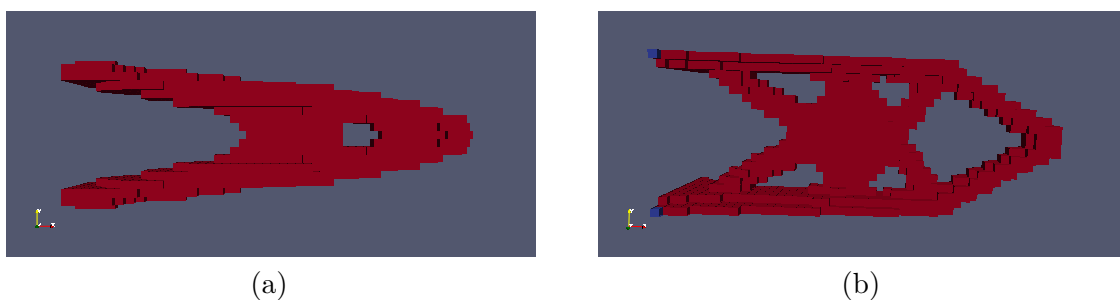
Em relação à flexibilidade, o resultado com pré-dimensionamento foi 54,9% maior, com um tempo 75,6% menor. A redução de tempo foi semelhante à dos testes 2D, porém o aumento da flexibilidade foi o dobro. Isso pode ser resultado da escolha da seção transversal: distribui-se material igualmente em torno do centro da seção transversal, ao invés de distribuí-lo considerando a direção das cargas. Pode-se notar isso no resultado da otimização topológica pura: a altura é muito maior que a espessura (sendo de 4 elementos na maior parte da extensão). Como no ambiente 2D o aumento da seção transversal sempre ocorre na direção paralela à das cargas, isso não é tão evidente.

Figura 15 – Resultados do Caso 4. A distribuição de tensão vai de maior nível de tensão von Mises (em vermelho) para menor nível (em azul): (a) Viga resultante (distribuição de tensão); (b) Topologia inicial; (c) Topologia final; (d) Topologia final (distribuição de tensão).



Fonte: Elaborado pelo autor.

Figura 16 – Comparação das topologias resultantes com e sem pré-dimensionamento no Caso 4: (a) Com pré-dimensionamento. A tensão von Mises máxima foi de 176,1 MPa. A flexibilidade foi de 261,659 N · mm; (b) Sem pré-dimensionamento. A tensão von Mises máxima foi de 160,4 MPa. A flexibilidade foi de 168,892 N · mm.



Fonte: Elaborado pelo autor.

4 IMPLEMENTAÇÃO EM AMBIENTE CONTÍNUO

4.1 Considerações iniciais

Para superar as limitações da implementação inicial, uma nova implementação foi desenvolvida, sem tomar outro programa como base. Neste caso, os cálculos partem de um meio contínuo, ao invés de um meio já discretizado, e com conhecimento da posição e área de contato das cargas aplicadas. Portanto, podemos calcular as tensões em cada ponto da linha central de cada viga, de forma geral, por:

$$\begin{aligned} S_F^x(\mathbf{X}) &= K_F^x F_x(\mathbf{X}) \\ S_F^y(\mathbf{X}) &= K_F^y F_y(\mathbf{X}) \\ S_F^z(\mathbf{X}) &= K_F^z F_z(\mathbf{X}) \end{aligned} \quad (105)$$

$$\begin{aligned} S_M^x(\mathbf{X}) &= K_M^x M_x(\mathbf{X}) \\ S_M^y(\mathbf{X}) &= K_M^y M_y(\mathbf{X}) \\ S_M^z(\mathbf{X}) &= K_M^z M_z(\mathbf{X}) \end{aligned} \quad (106)$$

em que $\mathbf{F}(\mathbf{X})$ é a força interna no ponto \mathbf{X} , $\mathbf{M}(\mathbf{X})$, o momento interno, S representa um componente de tensão e K , o coeficiente utilizado para o cálculo de cada componente. A escolha de K depende das cargas atuantes e suas direções e da seção transversal da viga inicial.

No caso de uma aplicação bidimensional, o objetivo inicial dessa implementação, isso se torna:

$$\sigma_f(\mathbf{X}) = \frac{6|\mathbf{M}(\mathbf{X})|}{th^2} \quad (107)$$

$$\sigma_n(\mathbf{X}) = \left| \frac{\mathbf{n}(\mathbf{X}) \cdot \mathbf{F}(\mathbf{X})}{th} \right| \quad (108)$$

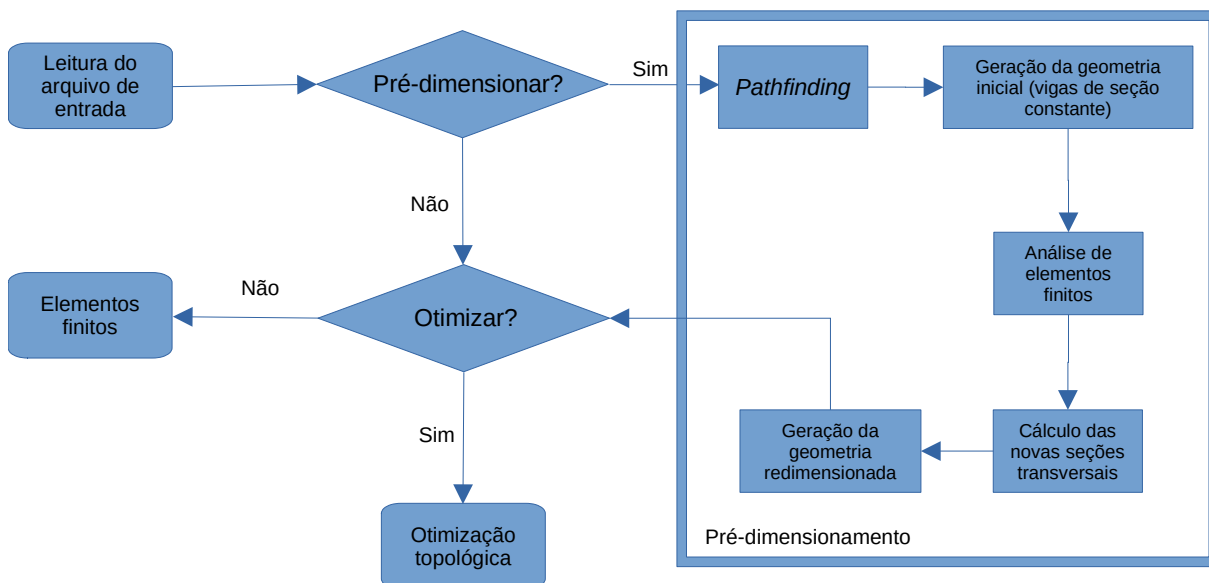
$$\tau(\mathbf{X}) = \frac{3}{2ht} |\mathbf{F}(\mathbf{X}) - [\mathbf{n}(\mathbf{X}) \cdot \mathbf{F}(\mathbf{X})] \mathbf{n}(\mathbf{X})| \quad (109)$$

em que $\sigma_f(\mathbf{X})$ é a tensão de flexão no ponto \mathbf{X} , $\sigma_n(\mathbf{X})$, a tensão normal, $\mathbf{n}(\mathbf{X})$, o vetor normal à seção transversal e $\tau(\mathbf{X})$ a tensão de cisalhamento. Consequentemente, é possível obter a nova altura $h(\mathbf{X})$ por:

$$\begin{aligned} h_f(\mathbf{X}) &= \sqrt{\frac{6|\mathbf{M}(\mathbf{X})|}{t\sigma_{max}}} \\ h_n(\mathbf{X}) &= \left| \frac{\mathbf{r}(\mathbf{X}) \cdot \mathbf{F}(\mathbf{X})}{t\sigma_{max}} \right| \\ h_c(\mathbf{X}) &= \frac{3}{2t\tau_{max}} |\mathbf{F}(\mathbf{X}) - [\mathbf{n}(\mathbf{X}) \cdot \mathbf{F}(\mathbf{X})] \mathbf{n}(\mathbf{X})| \\ h(\mathbf{X}) &= \max\{h_f, h_n, h_c, h_i\} \end{aligned} \quad (110)$$

Onde $h_i(\mathbf{X})$ é, novamente, o h original da viga no ponto \mathbf{X} , $h_f(\mathbf{X})$ é a estimativa de dimensão final com base nos esforços de flexão, $h_n(\mathbf{X})$, para os esforços normais e $h_c(\mathbf{X})$, para os de cisalhamento. Neste caso, há a vantagem de que, caso as tensões de cisalhamento e normal sejam menores que as de flexão, a topologia gerada terá dimensões e volume menores que o equivalente feito com a implementação anterior.

Figura 17 – Diagrama de blocos do funcionamento da implementação em ambiente contínuo. O fluxo do programa, assim como outros parâmetros, é definido pelo arquivo de entrada.



Fonte: Elaborado pelo autor.

A implementação foi desenvolvida em C++, utilizando a biblioteca Open CASCADE para modelagem sólida (Open Cascade,), a biblioteca LAPACK para solução de sistemas de equações lineares (Anderson *et al.*, 1999), principalmente para a implementação do método dos elementos finitos linear, a biblioteca GCMMA Solver para GCMMA (Dumas, 2018) e a biblioteca Gmsh para geração de malhas (Geuzaine; Remacle, 2009). A biblioteca de modelagem sólida foi utilizada para manipular a geometria de entrada, gerar a geometria intermediária e auxiliar na geração de malha dessa geometria para o método de elementos finitos. Para o *pathfinding*, foi utilizada a geração de grafo de visibilidade em conjunto com o A* (Berg *et al.*, 2008). Para atingir um equilíbrio entre esforço computacional e precisão dos resultados, foram utilizados os elementos GT9 (Yuqiu; Yin, 1994), elementos triangulares com graus de liberdade de rotação, apresentando precisão mais próxima a dos elementos quadráticos triangulares, mas utilizando 9 graus de liberdade por elemento ao invés de 12. A Figura 17 ilustra o funcionamento da implementação. A implementação está disponível em: <https://github.com/TarcisioLOliveira/SolidPrep>.

4.2 Abordagens consideradas

4.2.1 Questões a serem superadas

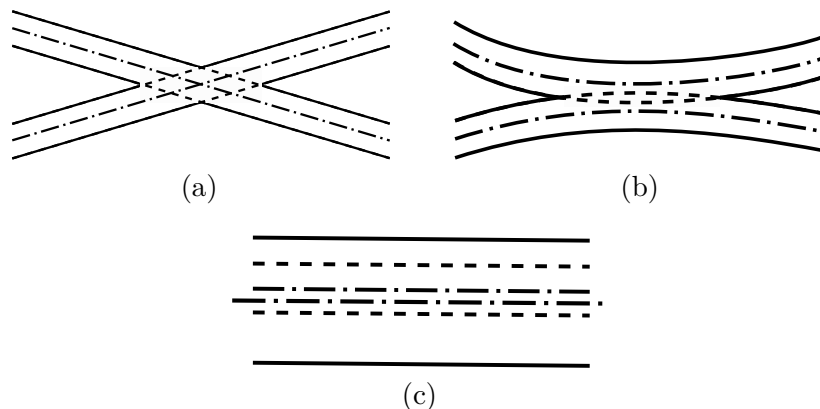
Embora realizar os cálculos e construções das topologias em um ambiente contínuo possibilita a obtenção de topologias e análises mais precisas, isso também apresenta dois desafios principais:

1. Gerar topologias complexas resultantes do ambiente contínuo ocasionadas pela união das geometrias das vigas; e
2. Realizar cálculos de vigas em um ambiente contínuo considerando essas complexidades.

Tais complexidades se referem a três categorias de interfaces físicas entre as vigas que poderiam ocorrer, dependendo do formato da geometria inicial e da disposição das condições de contorno:

1. **Interseção:** cruzamento da linha central das vigas (Figura 18a);
2. **Conexão:** cruzamento parcial, sem cruzamento da linha central (Figura 18b); e
3. **Sobreposição:** parte de uma viga é completamente sobreposta por outra viga com seção transversal de dimensões iguais ou maiores (Figura 18c).

Figura 18 – Representação esquemática das possíveis interfaces entre vigas. As linhas contínuas representam o contorno final, as tracejadas delimitam a região comum entre as vigas, e as linhas traço-ponto representam a linha central das vigas: (a) Interseção; (b) Conexão; (c) Sobreposição.



Fonte: Elaborado pelo autor.

Visando possibilitar o cálculo das vigas, algumas possíveis abordagens foram conceitualizadas antes da abordagem utilizada nesta implementação, mas que não foram adotadas por não as resolverem por completo, e serão descritas e discutidas brevemente nas próximas subseções.

4.2.2 Modelagem por elementos de vigas

A primeira abordagem idealizada consistia em modelar as vigas apenas por elementos de vigas. Esses elementos apresentam a vantagem de já serem adaptados aos cálculos de viga e automaticamente calcularem as cargas e momentos internos em cada nó.

Esse tipo de modelagem poderia ocasionar em resultados razoáveis em situações de interseção e sobreposição, embora fosse um tanto difícil fazê-lo. Visto que os elementos são unidimensionais, seria preciso investigar cada viga par a par para encontrar os casos de contato de vigas. A modelagem resultante seria, portanto, aproximada, pois implicaria na sobreposição parcial ou total dos elementos envolvidos, além de variações bruscas da seção transversal no encontro entre os pares de vigas na transição para uma situação de sobreposição.

Porém, a maior questão seria a modelagem de conexões. Como os elementos são unidimensionais, eles não apresentam suporte para o contato entre vigas que não envolva contato com a linha central da viga. Modelar essas interfaces como elementos de viga poderia resultar em erros consideráveis, visto que esses elementos estariam completamente sobrepostos aos elementos das vigas que eles conectam, e, portanto, o elemento provavelmente não modelaria corretamente essa interface.

4.2.3 Pontos centrais

Visando uma abordagem no meio contínuo, outra possibilidade analisada foi de aproveitar-se dos pontos obtidos pelo algoritmo de *pathfinding* para obter pontos de partida para delimitar seções transversais ao longo das vigas.

O método apresenta a vantagem de ser um pouco melhor adaptado ao meio contínuo. No caso de sobreposição de vigas, o método se adaptaria perfeitamente, ainda que fosse de certo modo redundante (os pontos de todas as vigas sobrepostas seriam analisados), e no caso de conexões, ele também se adaptaria ao aumento da seção transversal nessa região.

Porém, essa abordagem não conseguiria avaliar os pontos de interseção. Nesses pontos, o ideal seria utilizar as menores seções transversais da região, formadas pelos vértices resultantes da união das vigas. Como a obtenção de seções transversais é baseada nos pontos obtidos por *pathfinding*, a linha que delimita a seção transversal da região seria formada pelo vetor perpendicular ao nó. Portanto, nos pontos de interseção a seção transversal utilizada para os cálculos seria maior que a ideal.

4.2.4 Abordagem elementar

Uma abordagem elementar, isto é, baseada no cálculo por elementos finitos, seria potencialmente mais geral e melhor adaptada ao meio contínuo, e apresentaria a possibilidade de obter resultados mais otimizados. Embora para a otimização de forma isso

introduza imprecisões de cálculo e requeira refinamento da malha, o objetivo desse processo de dimensionamento é que a variação do tamanho do elemento não seja iterativa, ou seja, pelo menos heurística, e, portanto, isso não seria um problema.

Foi testada a possibilidade de tratar cada aresta de um elemento GT9 como uma seção transversal, redimensionando-a com base nas forças atuantes (simulando a análise de um ponto infinitesimal), isto é, com base em cálculos de tensões normais e de cisalhamento. O método foi implementado, mas não obteve resultados satisfatórios. Seria necessário obter uma formulação matemática mais precisa para obter melhores resultados.

4.3 Abordagem utilizada

A abordagem final escolhida é semelhante à dos pontos centrais, mas se valendo dos nós do contorno da geometria para a realização dos cálculos. A abordagem utiliza a geração de malhas unidimensionais (ou bidimensionais, em 3D) no contorno realizada pelo Gmsh durante o processo de geração da malha, sendo por isso denominada “pontos de contorno”.

Os elementos gerados são utilizados para obter o vetor normal ao contorno nesses pontos, e esse vetor é então utilizado para determinar a seção transversal a ser analisada. No âmbito bidimensional, isso significa encontrar o ponto oposto a ele no contorno. Esse ponto oposto poderia ser um ponto obtido pelo Gmsh, mas não há garantia de que haveria um ponto diretamente perpendicular, portanto é utilizado o ponto definido pela interseção entre a reta perpendicular e o contorno da geometria. Isso também implica que haverá certa redundância no processo (pois pontos de ambos lados do contorno são utilizados), mas para a implementação 2D isso contribui para aumentar a precisão do resultado, especialmente para malhas mais grosseiras. No âmbito 3D, porém, a quantidade de nós presentes no contorno torna isso proibitivo, e outra estratégia precisará ser utilizada.

Esse método apresenta todas as vantagens do método de pontos centrais, com a vantagem adicional de estar melhor adaptado para análise de interseções, visto que ele cria seções transversais a partir dos vértices gerados pelas interseções.

As reações nos apoios são calculadas pela análise de elementos finitos. Para cada seção apoiada, soma-se os vetores força para obter o vetor final da reação. No caso de estruturas estaticamente determinadas, o momento de reação foi calculado utilizando o equilíbrio de momentos externos. Para estruturas superdeterminadas, em cada apoio foi calculado o momento de reação pela soma do produto vetorial do vetor distância aos nós do suporte (em relação ao centro do mesmo) com essas mesmas forças nodais.

Esse método se baseia no cálculo integral do momento de flexão baseado na distribuição de forças internas ao longo da seção transversal, comumente usado para o

cálculo da tensão de flexão (Hibbeler, 2014):

$$\mathbf{M} = \int_A \mathbf{r} \times d\mathbf{F} = \int_A \mathbf{r} \times (\boldsymbol{\sigma} \cdot \mathbf{n}) d\mathbf{A}, \quad (111)$$

em que \mathbf{r} é a distância da linha neutra ao ponto onde é medida a força. A partir dessa equação, seria possível calcular com grande precisão o momento ao longo da seção transversal. Porém, dada a semelhança do cálculo do vetor força na equação com o cálculo do vetor de forças nodais da Equação 11, então \mathbf{F} pode ser discretizado em cada seção transversal por:

$$(\mathbf{f}_{cs})_e = \int_{A_e} \Phi_e^T (\boldsymbol{\sigma}_e \cdot \mathbf{n}_e) d\mathbf{A}_e \quad (112)$$

para cada elemento e que constitua essa seção transversal. Disso, a força distribuída ao longo da seção transversal é convertida em forças nodais concentradas. Como são forças concentradas, o cálculo da Equação 111 pode ser simplificado para:

$$\mathbf{M} = \sum_i \mathbf{r}_i \times (\mathbf{f}_{cs})_i. \quad (113)$$

para todo nó i presente na seção transversal. O método não foi amplamente testado, mas obteve resultados próximos aos analíticos no caso superdeterminado analisado, por isso foi adotado para essa pesquisa.

O cálculo dos esforços internos na seção transversal é feito de forma similar ao realizado manualmente. Antes de varrer as seções transversais da geometria, o método dos elementos finitos é utilizado para calcular os esforços nos apoios. Em seguida, para cada seção transversal o OpenCascade é utilizado para dividir a geometria naquela reta, e uma das partes é utilizada para fazer os cálculos, considerando apenas as forças externas presentes naquela parte.

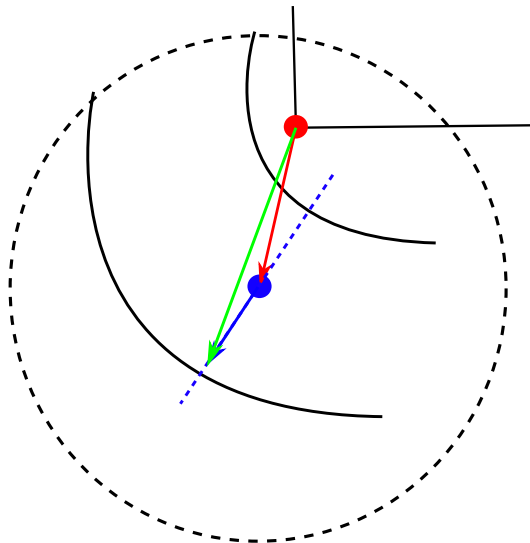
Embora esse método obtenha os resultados esperados para estruturas simples, ele não funciona para estruturas treliçadas, visto que neste caso seccionar a geometria em uma seção transversal não resulta necessariamente em duas partes distintas, portanto essa questão não foi abordada nesta pesquisa. Embora o elemento GT9 utilize rotações nodais para aumentar a precisão dos resultados de elementos finitos, essas rotações diferem da definição de rotação utilizada na mecânica do contínuo, e, portanto, não são apropriadas para o cálculo dos momentos (Yuqiu; Yin, 1994). Uma possibilidade seria utilizar o mesmo método de cálculo das reações para o cálculo dos esforços internos. Porém, esse método envolveria avaliar todos os elementos para cada seção, o que poderia ser lento. Seria necessária uma investigação mais aprofundada para analisar sua viabilidade.

Para respeitar a restrição geométrica (isto é, manter a geometria final dentro do espaço delimitado pela geometria inicial), após o cálculo das novas seções transversais, os pontos centrais obtidos anteriormente são deslocados para manter todo o círculo a ser formado dentro do espaço da geometria inicial.

Para todo centro de seção transversal, busca-se os pontos mais próximos presentes no contorno da geometria original que se encontrem dentro do círculo a ser formado, e esse centro é deslocado ao longo de uma reta paralela à seção transversal que cruze o mesmo para posicionar o ponto próximo obtido no contorno da seção transversal. A Figura 19 ilustra esse processo. Como esse deslocamento pode afetar a posição do centro do apoio, ele pode também afetar o cálculo dos momentos, portanto o redimensionamento é recalculado iterativamente até que a posição dos apoios se estabilize.

Na Figura 19, ponto azul é o centro da seção transversal, e o ponto vermelho, o ponto na geometria original mais próximo dele. O vetor vermelho define a distância entre os pontos, o vetor verde define a distância entre o ponto vermelho e o novo centro a ser definido, apresentando norma igual ao raio do círculo, e o vetor azul define a distância entre o novo centro e o centro atual, sendo paralelo à seção transversal. Como são conhecidas as dimensões de dois lados do triângulo e um de seus ângulos, é possível descobrir a norma do vetor azul pela lei dos cossenos, e com isso, a posição do novo centro.

Figura 19 – Esquemática do processo de reposicionamento da viga redimensionada.



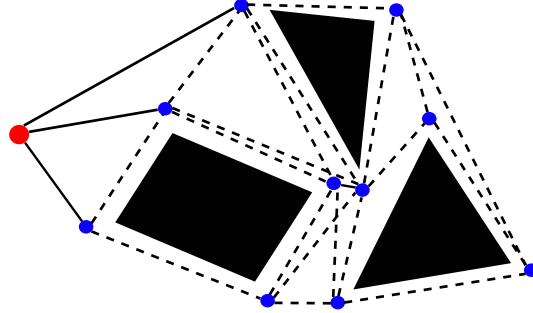
Fonte: Elaborado pelo autor.

4.4 Grafos de visibilidade

Em geometria computacional, um grafo de visibilidade é um grafo cujos nós são os vértices dos obstáculos do ambiente, conectados par a par caso a linha reta que os conecte não interseccione um obstáculo (Berg *et al.*, 2008) (Figura 20). Isto é, cada nó é conectado a todos os nós *visíveis* a partir dele. O algoritmo se baseia na definição de espaços convexos: uma linha reta que conecte quaisquer pares de pontos contidos no espaço também está inteiramente contida no espaço. Neste caso, os nós do grafo representam a transição entre duas sub-regiões convexas vizinhas da topologia.

A Figura 20 apresenta uma representação esquemática desse método. O ponto vermelho simboliza a posição atual, os pontos azuis, os nós do grafo, e os polígonos pretos, os obstáculos. As linhas contínuas representam os caminhos iniciais possíveis a partir da posição atual, e as tracejadas, possíveis caminhos a partir desses nós.

Figura 20 – Representação esquemática de um grafo de visibilidade.



Fonte: Elaborado pelo autor.

Como na implementação as vigas não apresentam dimensões desprezíveis, os nós são deslocados por uma distância predeterminada, denominada *restrição*, mais metade da largura da seção transversal, para tentar manter toda a viga contida no espaço da geometria inicial, embora no estado presente da implementação (isto é, o estado usado para a pesquisa e a obtenção dos resultados) seja possível em alguns casos que o contorno da viga interseccione o contorno da geometria inicial dependendo do valor da restrição. Para possibilitar a atuação dessa restrição da geração do caminho, para cada vértice do contorno da topologia é calculado o vetor normal a ele somando-se os vetores formados pelas arestas conectadas a esse vértice. No caso de arestas não retilíneas, no estado presente da implementação são criados oito pontos de espaçamento constante ao longo da aresta para permitir o *pathfinding* ao longo dela, porém isso pode não ser o suficiente para arestas muito longas, e essa geração de nós deve ser aprimorada no futuro.

O menor caminho entre a região de aplicação de força e um dos apoios é então obtido pelo algoritmo A*, listando a cada nó todos os nós visíveis por ele e tomando como próximo o nó que apresentar mínima distância $d = d_n + d_h$, em que d_n é a distância entre os nós e d_h é a distância euclidiana entre o nó sendo analisado e a região de apoio. Por fim, a viga é formada criando um caminho que parte do centro da região de aplicação da carga e se desloca com passo máximo constante e com rotação máxima por passo predeterminada, permitindo maior controle sobre o formato final da viga.

4.5 Otimização topológica

A parte de otimização topológica consiste em minimização de volume com restrição de tensão global, utilizando a medida de tensão global por norma-p como descrita por Le *et al.* (2010). Foi também utilizado o filtro de densidade como descrito no artigo. O

método foi escolhido principalmente por sua capacidade de modificar a geometria para eliminar pontos de concentração de tensão, além dos resultados do artigo servirem como base para avaliar a implementação do método.

Conseqüentemente, as restrições de tensão são semelhantes às Equações 66 e 78, utilizando $p_T = \frac{1}{2}$, mas com a introdução de uma variável c^μ de normalização da medida de tensão, atualizada a cada iteração, resultando em:

$$g(\boldsymbol{\rho}^\mu) = c^\mu \sigma_g^\mu = c^\mu \left[\sum_{e=1}^N v_e (\rho_e^\mu)^{p_T P} (\sigma_e^\mu)^P \right]^{\frac{1}{P}} \quad (114)$$

$$\frac{\partial g}{\partial \rho_e} = c^\mu \frac{\partial \sigma_g}{\partial \rho_e} = c^\mu (\sigma_g^\mu)^{1-P} \left[p_T v_e (\rho_e^\mu)^{p_T P-1} (\sigma_e^\mu)^P - p (\rho_e^\mu)^{p-1} (\boldsymbol{\lambda}_e^\mu)^T \mathbf{K}_e^0 \mathbf{u}_e^\mu \right] \quad (115)$$

$$\mathbf{K}^\mu \boldsymbol{\lambda}^\mu = \sum_{e=1}^N v_e (\rho_e^\mu)^{p_T P} (\sigma_e^\mu)^{P-2} \mathbf{L}_e \mathbf{B}_e \mathbf{C}_e \mathbf{V} \mathbf{C}_e \mathbf{B}_e \mathbf{u}_e^\mu \quad (116)$$

para cada iteração K . Como será utilizado filtragem de densidade, então a Equação 115 torna-se:

$$\frac{\partial g}{\partial d_i} = c^\mu (\sigma_g^\mu)^{1-P} \sum_{e=1}^N \left[p_T v_e (\rho_e^\mu)^{p_T P-1} (\sigma_e^\mu)^P - p (\rho_e^\mu)^{p-1} (\boldsymbol{\lambda}_e^\mu)^T \mathbf{K}_e^0 \mathbf{u}_e^\mu \right] \frac{\partial \rho_e}{\partial d_i} \quad (117)$$

em que:

$$\frac{\partial \rho_e}{\partial d_i} = \frac{\hat{H}_{ei}}{\sum_{j=1}^N \hat{H}_{ej}} \quad (118)$$

o que resulta em um processo similar à filtragem de sensibilidade, mas com variação dos pesos de acordo com a variável ρ_e analisada, como descrito por Bourdin (2001).

A variável c^μ é definida por:

$$c^\mu = \alpha^\mu \frac{(\sigma_e)_{max}^\mu}{\sigma_g^\mu} + (1 - \alpha^\mu) c^{K-1} \quad (119)$$

com o objetivo de aproximar a tensão obtida pela norma-p à tensão máxima por:

$$(\sigma_e)_{max}^\mu \approx c^\mu \sigma_g^\mu \quad (120)$$

com:

$$(\sigma_e)_{max}^\mu = \max\{\sigma_e\}, \quad e = 1, \dots, N \quad (121)$$

e sendo $0 < \alpha^\mu \leq 1$ um valor escolhido para o caso de c^μ oscilar a cada iteração (Le *et al.*, 2010). Para a implementação, α^μ foi definido por:

$$\alpha^\mu = \min \left\{ \frac{c^{K-1} \sigma_g^\mu}{(\sigma_e)_{max}^\mu}, \frac{(\sigma_e)_{max}^\mu}{c^{K-1} \sigma_g^\mu} \right\} \quad (122)$$

com $c^1 = \alpha^1 = 1$.

Foi observado que a suavização da tensão global proporcionada por c^μ e α^μ criam um problema de convergência: como $(\sigma_e)_{max}^\mu$ não apresenta uma relação direta com $c^\mu \sigma_g^\mu$,

é comum que ao longo do processo de otimização $(\sigma_e)_{max}^\mu$ alterne entre ser maior e menor que $c^\mu \sigma_g^\mu$, o que ocasiona oscilações nas medidas de tensão, visto que α^μ gera uma forma de “inércia”. Isso prolonga o processo de otimização, pois são necessárias algumas iterações para que $c^\mu \sigma_g^\mu$ volte a se tornar maior ou igual a $(\sigma_e)_{max}^\mu$. Portanto, para remediar isso, foi definido que quando $(\sigma_e)_{max}^\mu > c^\mu \sigma_g^\mu$, então $c^\mu = \frac{\sigma_g^\mu}{(\sigma_e)_{max}^\mu}$ e $\alpha^{K+1} = 0,5$, portanto impondo o limite $c^\mu \sigma_g^\mu \geq (\sigma_e)_{max}^\mu$, o que contribui para aumentar a velocidade de convergência.

4.6 Detalhes da implementação

4.6.1 Tamanho do elemento

Na implementação, a quantidade de elementos da malha é determinada principalmente pela propriedade “tamanho do elemento”, propriedade também presente em programas de análise por elementos finitos comerciais, como o ANSYS. No Gmsh, essa propriedade é definida por duas opções: tamanho máximo e tamanho mínimo do elemento. Para a implementação, ambas opções foram definidas com o mesmo número, que é o tamanho do elemento definido pelo usuário para a otimização. Porém, isso não significa que todos os elementos apresentem o mesmo tamanho, algo que é visível nos resultados. O Gmsh sempre usa uma malha adaptativa, resultando em tamanhos diferentes de elementos, e, portanto, em maior número de elementos para topologias mais complexas.

O Gmsh apresenta três métodos principais de geração de malha:

- **Delaunay:** método baseado na triangulação de Delaunay. Método mais rápido para geração de malhas e que resulta em malhas mais “orgânicas”, apresentando maior variação de tamanho do elemento e melhor adaptação à geometria;
- **Frontal-Delaunay:** método que resulta em malhas mais regulares, mas que também se adaptam à geometria (Rebay, 1993); e
- **MeshAdapt:** malha adaptativa utilizada caso o método escolhido não consiga produzir uma malha.

Visto que os métodos de otimização topológica costumam ser desenvolvidos e testados em malhas regulares, foi escolhido o método Frontal-Delaunay para a geração das malhas das geometrias.

4.6.2 Geração da geometria

4.6.2.1 Máximo ângulo de curvatura

O OpenCascade não apresenta por padrão alguma função que permita construir a geometria das vigas iniciais a partir dos pontos obtidos pelo algoritmo de *pathfinding*. Existe uma função parecida (`BRepPrimAPI_MakePipe`), mas que não rotaciona a seção

transversal durante o processo. Por isso, foi desenvolvida uma função que realiza essa construção utilizando duas funções de extrusão do OpenCascade: `BRepPrimAPI_MakePrism` (para seções retilíneas) e `BRepPrimAPI_MakeRevol` (para seções curvas).

A função `BRepPrimAPI_MakeRevol` é utilizada para a criação de sólidos de revolução, e, portanto apresenta uma limitação: a geometria resultante não pode apresentar auto-interseção. Consequentemente, o ângulo de curvatura definido pelo usuário para o processo de *pathfinding* deve ser, no máximo:

$$\tan \frac{\theta_{max}}{2} = \frac{s}{h_i} \quad (123)$$

em que θ_{max} é o ângulo máximo, s é o passo e h_i a altura da seção transversal.

4.6.2.2 Geometria sólida construtiva no OpenCascade

A geometria é construída utilizando geometria sólida construtiva (CSG) através da biblioteca OpenCascade. O CSG comumente é representado a partir de um conjunto de formas primitivas relacionadas por operações, mas no OpenCascade as operações booleanas estão implementadas para resultar em uma geometria bem definida, isto é, resultam em vértices, arestas, faces e sólidos. A formação desses componentes é vantajosa e necessária para o algoritmo, visto que são utilizados para a geração de malha. Porém, é necessário realizar as operações par a par, visto que adicionar todos os elementos em conjunto para a geração da geometria resulta em grande uso de memória.

Além disso, a operação booleana que se esperaria utilizar para a geração da geometria é a de união: unir, por exemplo, cada círculo gerado pelo algoritmo para gerar a geometria redimensionada. Porém, no OpenCascade, esse processo consiste em três etapas:

1. Calcular a interseção entre as duas geometrias;
2. Obter a diferença da interseção das duas geometrias; e
3. Unir as três geometrias resultantes.

Mas como apresentam duas subtrações e duas uniões, consistem no total em cinco etapas.

Testes realizados durante as fases iniciais de desenvolvimento mostraram que esse processo é lento demais para as necessidades do programa. Portanto, foi utilizado um método alternativo, mas análogo: obter a diferença das geometrias geradas com a geometria original (formando um “negativo” da geometria), e em seguida obter a diferença da geometria gerada com a original. O processo de corte consiste apenas em:

1. Calcular a interseção entre as duas geometrias;

2. Obter a diferença entre a interseção e a geometria base.

Testes realizados mostraram que esse processo é consideravelmente mais rápido, mas ainda assim tende a ser muito mais lento do que o da implementação inicial devido a essas características, especialmente considerando que o algoritmo apresenta duas etapas de geração de geometria: a primeira, para a formação das vigas iniciais (que também necessita de operações de extrusão e revolução para ser gerada), e a segunda, para a criação das vigas redimensionadas.

Portanto, a velocidade de geração dessas geometrias pode ser muito dependente da implementação, e a implementação fornecida pelo OpenCascade pode não ser a mais eficiente para o algoritmo.

4.6.3 Implementação do GCMMA

4.6.3.1 Visão geral

Ao longo do desenvolvimento da implementação, duas implementações do GCMMA foram utilizadas, sendo a do NLOpt utilizada inicialmente e a GCMMA Solver escolhida como a definitiva. Essa subseção é uma discussão sobre ambas e os motivos da escolha.

4.6.3.2 NLOpt

NLOpt é uma biblioteca de otimização não linear escrita em C e com interfaces para diversas outras linguagens. A implementação de GCMMA foi a primeira a compor a biblioteca, mas não é baseada no código desenvolvido por Svanberg (Johnson,). A interface do NLOpt é simples: define-se o objeto, critérios de parada e limites das variáveis de otimização e define-se a função objetivo e as restrições através de objetos de função. A própria biblioteca é posteriormente responsável por executar o laço de otimização.

No NLOpt, não há separação entre o cálculo da iteração externa (isto é, cálculo dos valores das funções e seus gradientes) e da iteração interna (cálculo apenas dos valores das funções), a cada iteração interna todos os valores e gradientes são recalculados, e a iteração externa apenas utiliza os últimos valores calculados para atualizar os limites e assíntotas. Isso que implica em maior esforço computacional a cada iteração, visto que, para a otimização baseada em tensão, isso resulta em duas análises de elementos finitos por iteração interna, ao invés de apenas uma.

A implementação do NLOpt é mais próxima da formulação original do MMA, não apresentando o uso dos parâmetros A_i , C_i e D_i e se limitando ao problema básico de minimização ou maximização de uma função com restrições não lineares. Por isso, se o método parte de uma solução não possível, a implementação utiliza estratégias como manter $y_i = 10^{40}$ e $\beta_i = 10^{40}$ até que um ponto possível seja atingido. Porém, isso apresenta

sucesso limitado, pois foram identificados casos em que a implementação resultou em $f_0(\mathbf{d}) \rightarrow 0$.

É importante notar, porém, que a implementação se mostra muito eficiente, embora não tenha adicionado várias vantagens do GCMMA. Por exemplo, o método dual utilizado se mostra muito rápido, e consiste em resolver separadamente para cada d_j a aproximação convexa da função objetivo. Em geral, a implementação converge aos resultados esperados, tendo sido testada para minimização de tensão e minimização de conformidade, mas nem sempre se encaixa nos requisitos dos problemas investigados nesta pesquisa.

4.6.3.3 GCMMASolver

GCMMASolver é um par de arquivos do repositório `jdumas/mma` (especificamente `GCMMASolver.h` e `GCMMASolver.cpp`) mantido por Jérémie Dumas e baseado em código desenvolvido por Niels Aage (Svanberg, 2007), utilizando método descrito por Aage e Lazarov (2013) para resolução do subproblema convexo e sendo mais próxima do método como descrito por Svanberg, apresentando apenas algumas modificações além do método de resolução do subproblema. É uma biblioteca autocontida (isto é, não depende de outras bibliotecas) que utiliza licença MIT, por isso foi integrada inteiramente ao código.

A biblioteca é baseada no uso de três funções: `OuterUpdate()`, que realiza os cálculos de cada iteração externa, `InnerUpdate()`, para as iterações internas, e `ConCheck()` para conferir se as aproximações são conservativas, sendo o laço definido externamente à biblioteca, isto é, por quem integrar a biblioteca ao próprio código.

Essa implementação se mostrou mais propícia para a realização dos testes, visto que permite reduzir a carga computacional das iterações externas e apresenta os parâmetros do GCMMA. Definir $C_i = 10^6$, reverter ao método descrito por Svanberg para o cálculo de ξ_j^μ e utilizar como limites de U_j^μ e L_j^μ :

$$\begin{aligned} d_j^\mu - 100(d_j^{max} - d_j^{min}) &\leq L_j^\mu \leq d_j^\mu - 0,001(d_j^{max} - d_j^{min}) \\ d_j^\mu + 100(d_j^{max} - d_j^{min}) &\geq U_j^\mu \geq d_j^\mu + 0,001(d_j^{max} - d_j^{min}) \end{aligned} \quad (124)$$

que no GCMMASolver original eram mais amplos, foram o suficiente para permitir que os testes conseguissem convergir a partir de frações de volume inicial tão baixas quanto 0,05, o que com o NLOpt não era possível com os parâmetros usados para os testes.

Visto que há maior flexibilidade para definição dos critérios de parada, eles foram definidos como:

$$\begin{aligned} \max\{|d_j^\mu - d_j^{\mu-1}|\} &\leq 10^{-5} \text{ ou } \frac{|V^\mu - V^{\mu-1}|}{V_{max}} \leq 10^{-6} \\ \text{contanto que } \left| \frac{(\sigma_e)_{max}^\mu}{c^\mu \sigma_g^\mu} - 1 \right| &\leq 10^{-4} \end{aligned} \quad (125)$$

em que V^μ é o volume da geometria na iteração μ e V_{max} , o máximo volume possível da geometria. Esses critérios visam impedir tanto que o processo de otimização termine

prematuramente, por exemplo, por dificuldade de convergência devido à diferença entre a tensão real e normalizada ou devido à grande amplitude de variação de volume entre iterações, quanto que ele se estenda por muito tempo ou indefinidamente devido às oscilações dos níveis de tensão quando próximo à convergência.

4.6.4 Tensões máximas de projeto

Visto que a implementação foi feita buscando minimizar as dimensões das vigas após o dimensionamento, espera-se que as tensões máximas finais da topologia sejam próximas às tensões de projeto, significando que a retirada de material ocasionaria no aumento dos níveis de tensão, violando as tensões de projeto. Portanto, deve-se investigar a possibilidade de sobredimensionar para ser possível reduzir massa, analisando também se haverá redução significativa da massa com esse processo.

Por outro lado, o processo de pré-dimensionamento se baseia nas componentes de tensão, possibilitando melhor redução das dimensões ao se basear nas componentes das cargas. A otimização topológica, porém, se baseia na tensão de von Mises máxima. Embora esse critério diminua a carga computacional, ele abre a possibilidade das componentes das tensões finais serem maiores que as de projeto. Por um lado, utilizar o maior valor de tensão de projeto seria insuficiente, visto que o critério analisa todos os componentes em conjunto. Por outro, em situações em que alguma das componentes é próxima de zero, as outras podem ultrapassar seus valores máximos. Deve-se analisar, portanto, a escolha dos critérios de tensão para cada etapa.

4.7 Estudos de caso

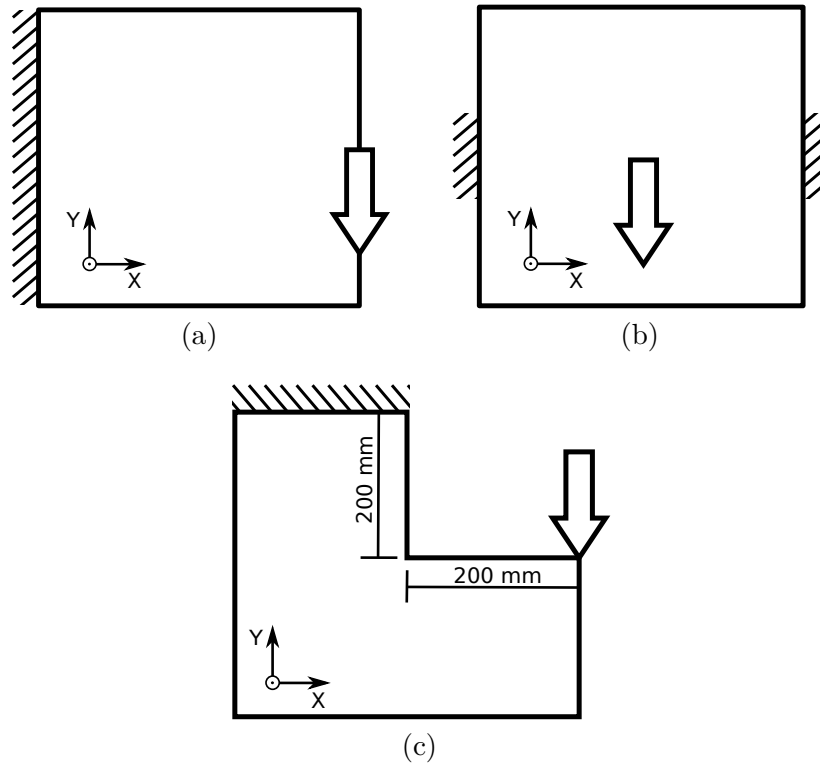
Para a avaliação da implementação e, com isso, do método desenvolvido, foram realizados três tipos de estudo: análise de elementos finitos, otimização topológica e tempo de resolução, utilizando-se problemas comuns da literatura para a análise (viga engastada, flexão em 3 pontos e viga em L) (Figura 21).

As análises de elementos finitos visam avaliar a habilidade do algoritmo em gerar estruturas que se adéquem às tensões de projeto. Neste caso, nem todos os testes apresentam dimensões finais semelhantes às obtidas por otimização topológica pura, visto que a capacidade de redução de volume é um dos critérios de comparação, especialmente para as análises seguintes. Para esta análise, foi utilizado tamanho de elemento de 3 mm.

Na subseção de otimização topológica, foi analisada a necessidade de sobredimensionamento (no caso, sobredimensionando por um coeficiente constante caso necessário), influência do tamanho dos elementos, efeito de regiões de concentração de tensão e os volumes, geometrias e níveis de tensão obtidos ao final do processo.

A análise do tempo de resolução visa comparar o tempo de resolução, o volume

Figura 21 – Problemas resolvidos pelos Casos 1 a 3: (a) Caso 1; (b) Caso 2; (c) Caso 3.



Fonte: Elaborado pelo autor.

resultante, e outros possíveis fatores que possam afetar o tempo de resolução.

A Figura 21 apresenta os problemas a serem analisados, a Tabela 7 apresenta os parâmetros comuns aos quatro testes, a Tabela 8, os parâmetros específicos de cada um, e a Tabela 9, parâmetros para comparação de tempo, que também foram utilizados para gerar algumas das geometrias analisadas na subseção de otimização topológica. Visto que o coeficiente de Poisson é baixo, foi utilizado $p = 4$ para a penalização das densidades. A máxima tensão normal e a de cisalhamento são utilizadas para o dimensionamento das vigas, enquanto a máxima tensão de von Mises é utilizada para a otimização topológica, sendo esta calculada a partir das outras duas. Diferentes tamanhos de elemento e intensidades de força foram escolhidos para balancear o tamanho da geometria com o esforço computacional necessário. Para o dimensionamento, foram utilizados elementos de tamanho 4 mm para análise das vigas iniciais, e para a otimização topológica, 5 mm.

Tabela 7 – Parâmetros comuns aos quatro testes analisados por elementos finitos. O material é uma aproximação do PLA extrudado no processo de fabricação por filamento fundido.

Propriedade	Valor
Módulo de elasticidade	2,0 GPa
Coefficiente de Poisson	0,23
Máxima tensão normal	10,0 MPa
Máxima tensão de cisalhamento	7,5 MPa
Máxima tensão equivalente de von Mises	16,4 MPa
Espessura	20 mm
Dimensões	400×400 mm ²
Raio do filtro de densidade	2x o tamanho do elemento
P (norma-p)	8
p (penalização)	4
p_T (relaxamento da tensão)	0,5

Fonte: Elaborado pelo autor, dados do material baseados em Dialami *et al.* (2021).

Tabela 8 – Parâmetros específicos de cada teste.

Propriedade	Caso 1.1	Caso 1.2	Caso 2	Caso 3.1	Caso 3.2
Volume total [10 ⁵ mm ³]	32	32	32	24	24
	Pathfinding				
Passo [mm]	2	2	2	2	2
Máxima rotação [°]	5	5	5	5	1.5
Restrição [mm]	0	0	0	10	50
	Carga				
Vetor força [10 ³ N]	(0, -2)	(0, -5)	(0, -10)	(0, -1)	(0, -1)
Área de contato [mm]	20	50	30	20	20
Área de apoio [mm]	400	400	125	200	200

Fonte: Elaborado pelo autor.

Tabela 9 – Parâmetros para a comparação de tempo de resolução. O Caso 1.2 não é utilizado para essa análise.

Parâmetro	Caso 1.1	Caso 2	Caso 3.1	Caso 3.2
Fração de volume inicial	0,1	0,05	0,1	0,1
Tamanho do elemento [mm]	5	5	5	5
Sobredimensionamento	1,0	1,0	1,7	1,0

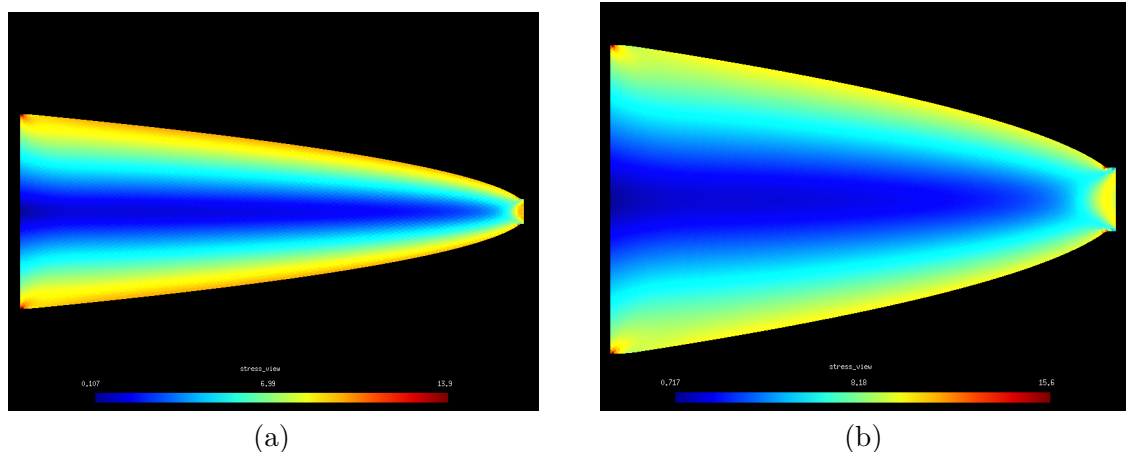
Fonte: Elaborado pelo autor.

4.8 Resultados e discussão

4.8.1 Análise de elementos finitos

A Figura 22 apresenta os resultados das análises de elementos finitos para os Casos 1.1 e 1.2. Para ambos, as tensões máximas ficaram próximas do máximo esperado (isto é, pouco acima de 10 MPa, por ação da flexão em conjunto com a diminuta carga cisalhante), porém há a ocorrência de concentração de tensão nos cantos internos e externos da geometria. Como as dimensões das seções transversais são relativamente grandes (em comparação ao comprimento das vigas), e as tensões nos cantos externos aumentam com o aumento da seção transversal, isso pode ser resultante da violação das premissas da modelagem de viga utilizada. A Figura 23, para o Caso 2, apresenta um resultado semelhante, mas com concentração de tensão no local de aplicação da carga também.

Figura 22 – Análise de elementos finitos das vigas redimensionadas para os Casos 1.1 e 1.2. Tensão equivalente de von Mises (MPa): (a) Caso 1.1, tensão máxima de 13,9 MPa; (b) Caso 1.2, tensão máxima de 15,6 MPa.

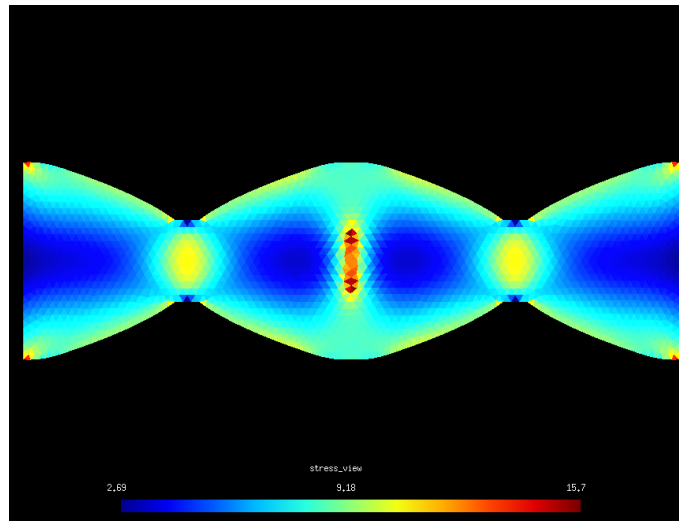


Fonte: Elaborado pelo autor.

A Figura 24 apresenta os resultados para os Casos 3.1 e 3.2. O Caso 3.1 representa a geometria obtida buscando-se uma viga com o menor comprimento possível. Isso resulta em uma viga redimensionada com um canto interno, visto que a geometria se adapta ao contorno da geometria inicial, que também apresenta esse canto interno. Como resultado, a concentração de tensão chega a 32,2 MPa nessa região. Porém, é possível também modificar os parâmetros de *pathfinding* para evitar a formação desse canto interno, o que foi feito para o Caso 3.2. Como mostra a Figura, não há presença dessa concentração de tensão, e a tensão máxima é de 17,0 MPa. Para comparação, a Figura 25 apresenta as vigas como criada pelo *pathfinding* (isto é, antes do dimensionamento).

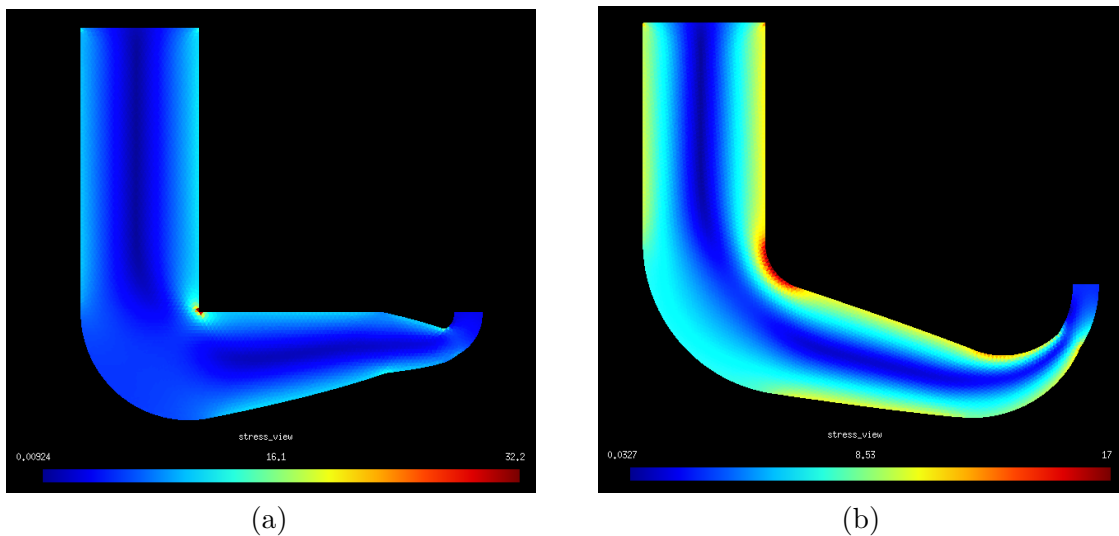
Em todos os resultados, nota-se que as tensões máximas estão muito acima da tensão máxima de flexão, e em alguns casos também da tensão máxima de von Mises. Para os Casos 1.1, 1.2 e 2, essas tensões máximas originam de regiões de concentração de

Figura 23 – Análise de elementos finitos das vigas redimensionadas para o Caso 2. Tensão equivalente de von Mises (MPa). Tensão máxima de 15,7 MPa.



Fonte: Elaborado pelo autor.

Figura 24 – Análise de elementos finitos das vigas redimensionadas para os Casos 3.1 e 3.2. Tensão equivalente de von Mises (MPa): (a) Caso 3.1, tensão máxima de 32,2 MPa; (b) Caso 3.2, tensão máxima de 17,0 MPa.



Fonte: Elaborado pelo autor.

tensão que são facilmente removidas manual e parcialmente fruto da modelagem (isto é, em aplicações práticas elas não apresentariam cantos perfeitamente retos, embora esses cantos reais ainda concentrem tensão). Para o Caso 3.1, porém, é necessário remover o ponto de concentração de tensão, por exemplo, através de otimização topológica ou de forma. Para o Caso 3.2, esperava-se que a curvatura utilizada fosse suficiente para reduzir as tensões a um nível aceitável, mas é possível que maiores curvaturas fossem necessárias. Porém, como as tensões são mais próximas da tensão máxima de von Mises utilizada para

Figura 25 – Vigas geradas pelo algoritmo de *pathfinding* e utilizadas para o pré-dimensionamento dos Casos 3.1 e 3.2: (a) Caso 3.1; (b) Caso 3.2.



Fonte: Elaborado pelo autor.

otimização topológica, espera-se que ele apresente melhor desempenho nessa análise que o Caso 3.1.

4.8.2 Otimização topológica

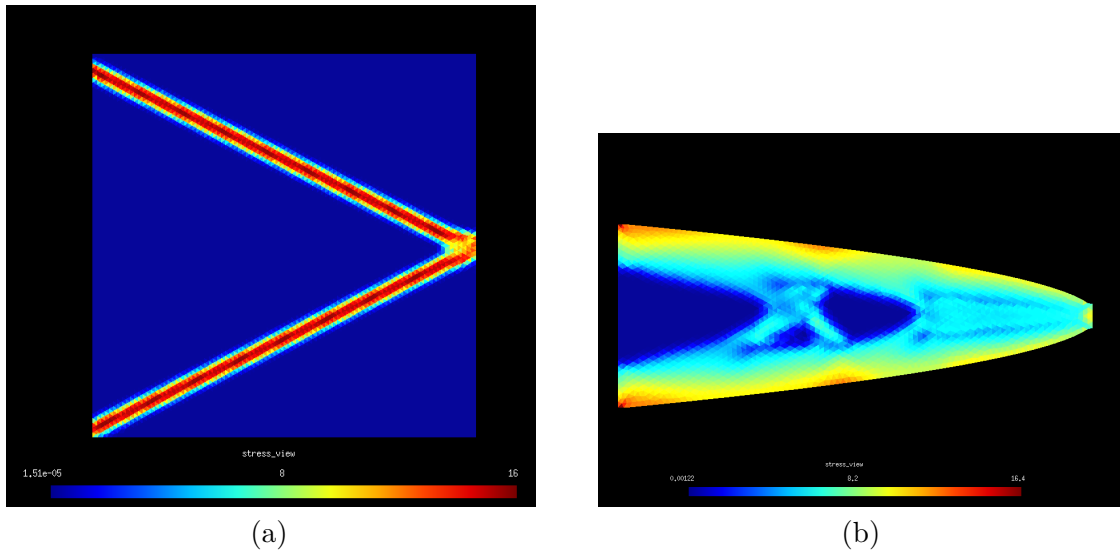
A Figura 26 apresenta os resultados obtidos de otimização topológica para o Caso 1.1. A Figura 26a apresenta o resultado para otimização topológica pura.

A Figura 26b apresenta os resultados de otimização com pré-dimensionamento. Neste caso, embora haja grande redução dimensional (a geometria se encaixa em um retângulo de $400 \times 155 \text{ mm}^2$), o volume final é consideravelmente maior (2,32 vezes maior). Considerando-se os resultados da implementação em ambiente discretizado, isso é esperado, pois, a restrição geométrica impõe também uma restrição ao espaço de solução. Porém, a redução de volume ainda é grande (77,2%). Portanto, a redução dimensional se mostra desvantajosa em relação à diminuição de volume, embora a diminuição ainda seja considerável.

Como dito anteriormente, os resultados dependem do valor escolhido para tensão de von Mises máxima, e o valor deve considerar a possibilidade dos componentes excederem as tensões máximas de projeto das componentes (normal e de cisalhamento), mas também considerar que as componentes atuam em conjunto. A Figura 27 apresenta os resultados da otimização com tensão máxima de 10 MPa.

Em ambos os casos, a tensão máxima final excedeu a tensão máxima de projeto. No teste de flexão (Figura 27a), a tensão excedeu o limite em regiões de concentração de tensão, isto é, nos cantos conectados aos apoios. No teste de compressão (27b), as tensões

Figura 26 – Resultados de otimização topológica para o Caso 1.1 com carga de intensidade 2000 N. Representação dos níveis de tensão de von Mises, em MPa: (a) Otimização topológica pura, com volume final de 264.635 mm³ (8,3% do volume inicial); (b) Utilizando pré-dimensionamento, com volume final de 614.733 mm³ (22,8% do volume inicial).



Fonte: Elaborado pelo autor.

elevadas encontram-se predominantemente próximas à região de aplicação, visto que a área de aplicação é menor que a área da seção transversal. Em ambos os casos, a maior parte da geometria está dentro das tensões de projeto, mas a retirada de material é muito baixa.

Considerando a Equação 110, espera-se que a maior seção transversal da viga pré-dimensionada apresente altura de 154,9 mm, o que resultaria em uma tensão de cisalhamento de 0,97 MPa. Em conjunto com a tensão de flexão de 10 MPa, isso resulta em uma tensão de von Mises de 10,1 MPa.

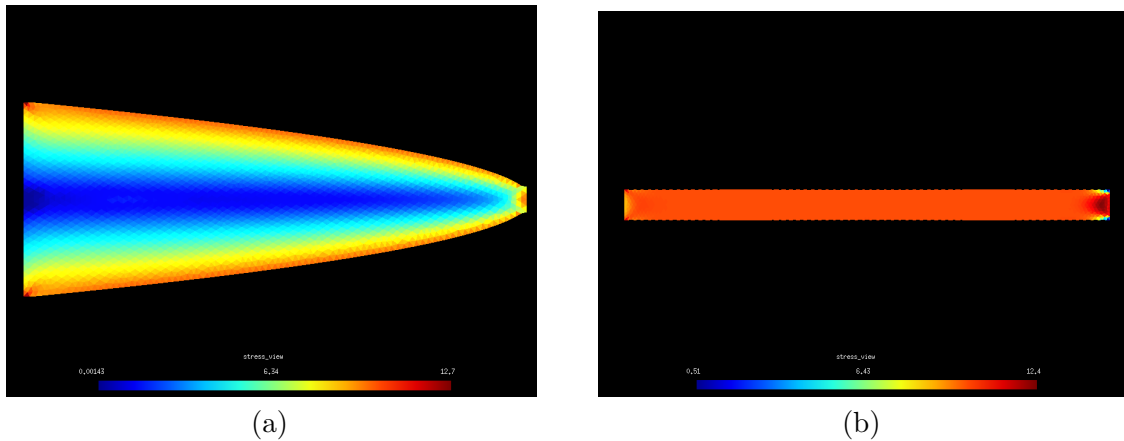
Porém, após a otimização, a viga se divide em duas. Assumindo que cada parte dessa seção transversal apresenta um terço da altura da original e com:

$$I = 2 \left(thr^2 + \frac{th^3}{12} \right) \quad (126)$$

$$Q = rth \quad (127)$$

sendo I o segundo momento de área e Q o primeiro momento de área, com $h = \frac{154,9}{3}$ mm e sendo $r = h$ a distância entre as duas partes da seção transversal, disso resulta em uma tensão de flexão de 10,4 MPa e de cisalhamento de 0,89 MPa, portanto em uma tensão de von Mises de 10,5 MPa. Como houve pouca variação da seção transversal (devido à baixa retirada de material), os níveis de tensão deveriam estar abaixo desse valor, ou pelo menos próximo. Como as tensões elevadas estão situadas predominantemente nos cantos, é mais

Figura 27 – Caso 1 utilizando pré-dimensionamento e tensão de von Mises máxima de 10,0 MPa. Representação dos níveis de tensão de von Mises, em MPa: (a) Flexão com carga de 2000 N, tensão máxima de 12,7 MPa; (b) Compressão com carga de 5000 N, tensão máxima de 12,4 MPa (elemento de tamanho 3 mm).



Fonte: Elaborado pelo autor.

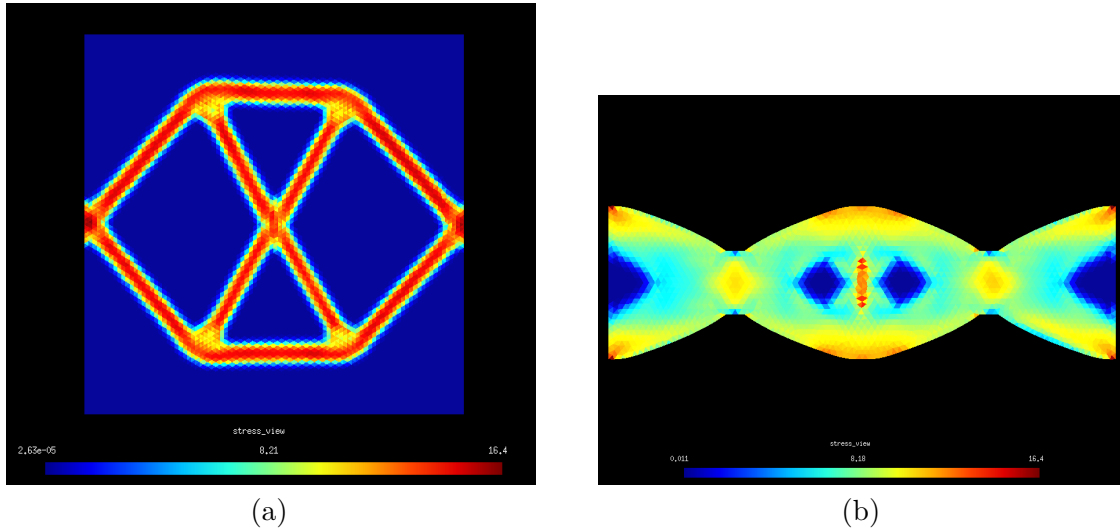
provável que sejam originadas de concentração de tensão. O fato do tensionamento no restante do contorno da topologia está mais próximo 10 MPa contribui também para essa interpretação. Porém, para maiores dimensões isso pode não se manter, possibilidade que não foi testada nesta pesquisa.

A tensão de cisalhamento, por outro lado, apresenta maior contribuição em regiões de menor seção transversal, como próximo ao ponto de aplicação de força. Como a seção transversal inicial da carga é de 20 mm, isso resulta em uma tensão de cisalhamento de 7,5 MPa, e como a tensão normal nesse ponto é próxima de zero, a tensão de von Mises total é de 13,0 MPa. Isso, porém, se mostra mais distante dos resultados obtidos por elementos finitos. Por outro lado, a tensão na região de aplicação da carga se mostra mais próxima de 10 MPa do que de 7,5 MPa, indicando que essa restrição possa ter sido violada.

Conseqüentemente, a relação entre as tensões máximas para dimensionamento e a tensão máxima de von Mises para a otimização não é clara. Por um lado, como a estrutura está sendo dimensionada como uma viga, seria esperado que ela fosse otimizada com base nas tensões usadas para o dimensionamento. Por outro lado, a estrutura gerada pode violar as premissas da modelagem de viga, tanto pelas dimensões envolvidas como pela possibilidade de interseção das geometrias. Isto é, o resultado do pré-dimensionamento é uma estrutura contínua. Portanto, o critério de von Mises seria mais adequado. O processo de geração e otimização dessas geometrias deve considerar esses dois lados para a escolha das tensões de projeto, mas a relação entre elas não é clara.

A Figura 28 apresenta os resultados para o Caso 2. Neste caso, a topologia pré-dimensionada obteve um volume final mais próximo ao da otimização topológica pura

Figura 28 – Resultados da otimização topológica para o Caso 2. Representação dos níveis de tensão de von Mises, em MPa (a) Otimização topológica pura, com volume final de 593.822 mm^3 (18,6% do volume inicial). (b) Utilizando pré-dimensionamento, com volume final de 623.840 mm^3 (19,5% do volume inicial).



Fonte: Elaborado pelo autor.

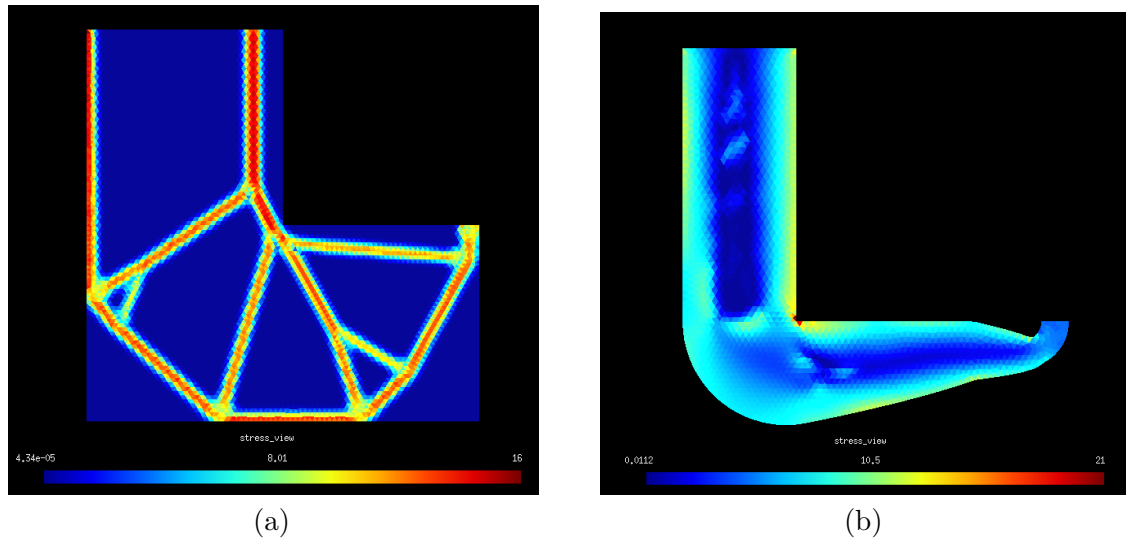
(aumento de 5,1%), e o método também conseguiu reduzir a intensidade da concentração de tensão. Por outro lado, a distribuição de tensão não é tão homogênea, e se mantém concentrada nos cantos dos apoios e na região de aplicação da carga

A Figura 29 apresenta os resultados para o Caso 3.1. Para a otimização topológica pura, o método utilizado foi capaz de eliminar a concentração de tensão e reduzir consideravelmente o volume. A Figura 29b apresenta os resultados para a otimização topológica com pré-dimensionamento. Neste caso, a otimização não foi capaz de diminuir os níveis de tensão o suficiente, devido à concentração de tensão no canto interno. Por isso, analisou-se também a possibilidade de utilizar um coeficiente de aumento da seção transversal.

A Figura 30 apresenta os resultados obtidos com a adição de um coeficiente de sobredimensionamento, feito utilizando os mesmos parâmetros usados para obter os resultados da Figura 29b (fração de volume inicial de 0,1 e elementos de 5 mm). O coeficiente de 1,15 foi suficiente para resultar em uma tensão máxima menor que a tensão de projeto, porém não foi suficiente para eliminar a região de concentração de tensão, por isso, quando malhas mais finas são usadas a tensão máxima final é maior que a tensão máxima de projeto. Para melhor controlar a concentração de tensão, foi necessário um sobredimensionamento de pelo menos 1,70.

Esses resultados implicam que, para o controle de concentração de tensão, o uso de topologias iniciais de maiores volumes é mais benéfico, ainda que resulte em maior tempo de otimização. Esses maiores volumes iniciais também tendem a resultar em menores volumes finais.

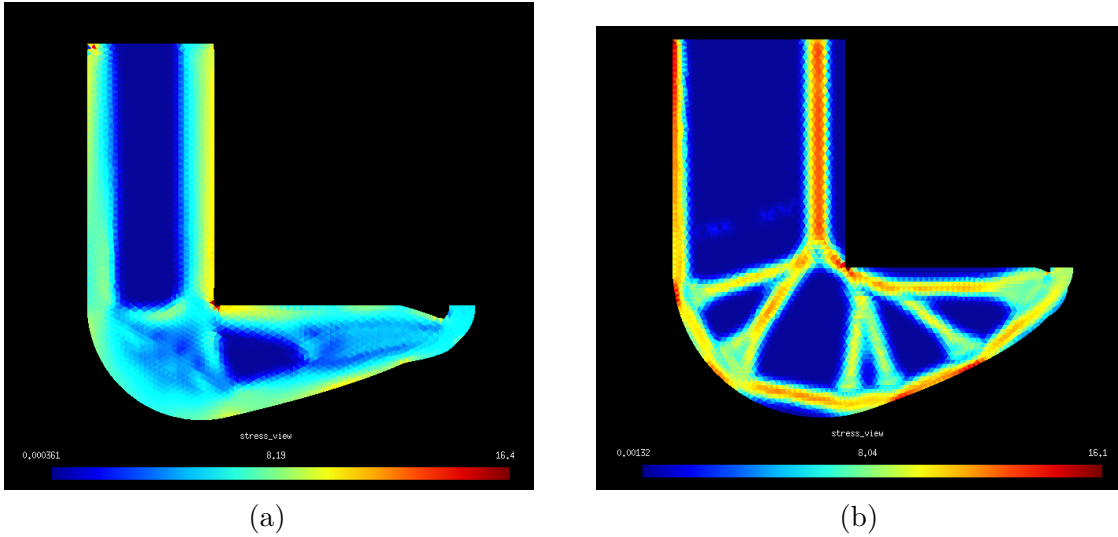
Figura 29 – Resultados de otimização topológica para o Caso 3.1. Representação dos níveis de tensão de von Mises, em MPa: (a) Otimização topológica pura, com volume final de 384.313 mm^3 (16,0% do volume inicial); (b) Utilizando pré-dimensionamento, com volume final de 611.341 mm^3 (22,7% do volume inicial).



Fonte: Elaborado pelo autor.

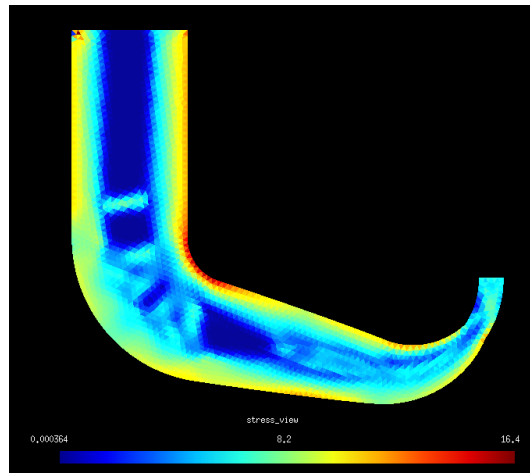
Porém, os parâmetros de *pathfinding* também podem ser ajustados para evitar a formação desse canto interno. A Figura 31 apresenta o resultado da otimização topológica para o Caso 3.2, desenvolvido para testar essa hipótese. A geometria pôde ser otimizada sem a necessidade de sobredimensionamento e o resultado se mantém para malhas mais finas.

Figura 30 – Resultados de otimização topológica para o Caso 3.1 utilizando coeficiente de sobredimensionamento. Representação dos níveis de tensão de von Mises, em MPa: (a) Coeficiente de 1,15, com volume final de 507.666 mm^3 (21,2% do volume inicial); (b) Coeficiente de 1,70, com volume final de 405.546 mm^3 (16,9% do volume inicial).



Fonte: Elaborado pelo autor.

Figura 31 – Resultados de otimização topológica para o Caso 3.2 utilizando coeficiente de sobredimensionamento. Representação dos níveis de tensão de von Mises, em MPa. Volume final de 597.486 mm^3 (24,9% do volume inicial).



Fonte: Elaborado pelo autor.

4.8.3 Tempo de otimização

A Tabela 10 compila os tempos de resolução dos quatro casos analisados, assim como o volume resultante, e a Tabela 11 apresenta uma comparação de variação de tempo de resolução e variação de volume para cada caso, comparando os resultados com e sem sobredimensionamento.

Tabela 10 – Resultados dos Casos 1.1, 2, 3.1 e 3.2, com e sem pré-dimensionamento, em termos de tempo de resolução e volume final.

Caso		Total (min)	Num it.	Por it. (s)	Pré-dim. (s)	Volume (mm³)
1.1	Sem	1123,3	470	143,4	-	264.635
	Com	96,7	448	12,9	38	614.733
2	Sem	1386,2	490	169,7	-	593.822
	Com	19,9	379	3,0	43	623.840
3.1	Sem	410,0	438	56,1	-	384.313
	Com	65,4	110	35,0	64	405.546
3.2	Sem	410,0	438	56,1	-	384.313
	Com	42,0	305	7,9	96	597.486

Fonte: Elaborado pelo autor.

Tabela 11 – Comparação do tempo de resolução e volume final para os Casos 1.1, 2, 3.1 e 3.2, com e sem pré-dimensionamento.

Caso	Variação de tempo (%)	Variação de volume (%)
1.1	-91,4	+132,3
2	-98,6	+5,1
3.1	-84,0	+6,6
3.2	-89,8	+57,1

Fonte: Elaborado pelo autor.

Em todos os casos, houve redução do tempo de resolução em aproximadamente 90%, mas o volume resultante apresentou grande variação, em um caso sendo mais de duas vezes maior que o obtido sem pré-dimensionamento (Caso 1.1) e em outro, com aumento de apenas 5,1% (Caso 2). Isto é, há uma notável troca entre tempo de resolução e volume final. Porém, a redução de tempo é muito mais pronunciada que o aumento de volume, visto que a redução de volume em relação ao volume da topologia original é de pelo menos 75% (Caso 3.2).

Nota-se também que, em geral, o número de iterações necessárias para atingir o critério de parada é menor para topologias pré-dimensionadas, especialmente no Caso 3.1. O número de nós é uma das principais causas da redução de tempo de resolução, visto que eles contribuem para o aumento do tempo de resolução do problema de elementos finitos. Como as topologias são menores, elas apresentam menor número de elementos, e, portanto, menor número de nós, resultando em menor tempo por iteração externa. O menor número de iterações também contribui para a redução do tempo total, mas como o tempo por iteração em geral é consideravelmente menor, sua influência no tempo de resolução não é tão significativa quanto seria para a otimização pura.

5 CONCLUSÕES

5.1 Conclusões gerais

Considerando os resultados apresentados, o uso de métodos de dimensionamento de vigas apresenta grande potencial para a área de projeto estrutural, especialmente para a automação de projeto, visto que se mostra bem-sucedido em gerar estruturas contínuas capazes de resistir aos esforços de projeto quando em conjunto com a otimização topológica (e, possivelmente, com outros métodos de otimização estrutural), embora seja necessário ajustar os parâmetros de dimensionamento e *pathfinding* para isso em certos casos. É preciso considerar, porém, que o uso de métodos de otimização estrutural tende a ser necessário para obter estruturas viáveis, visto que os resultados podem não se encaixar completamente dentro das premissas de modelagem de vigas.

Em ambos os métodos implementados, é notável a presença de uma “troca” entre redução de tempo de resolução e redução dimensional, e redução de volume e flexibilidade (mesmo para as estruturas resultantes de minimização de volume, é esperado que o pré-dimensionamento resulte em maior flexibilidade), embora não seja uma relação direta (isto é, é possível obter grande redução de tempo de resolução com baixo aumento de volume).

A maior vantagem seria nas fases iniciais do projeto estrutural, visto que a analogia de vigas e a maior velocidade de otimização permitem obter resultados funcionais mais rapidamente, mesmo considerando a necessidade de ajuste dos parâmetros de *pathfinding* e pré-dimensionamento.

Quanto à manufacturabilidade, a redução dimensional contribui para a diminuição da área da base e de superfície das estruturas, o que contribui para a redução do tempo de fabricação em manufatura aditiva, e pode também aumentar a quantidade de peças fabricadas por lote. Por outro lado, o aumento do uso de material pode também contribuir para o aumento dos custos. Em 3D, porém, a diminuição das dimensões pode contribuir também para a diminuição do material necessário para estruturas de suporte, o que pode reduzir ou compensar o maior uso de material. Para manufatura subtrativa, as menores dimensões significam menor quantidade de material a ser removido, o que é um dos principais fatores em questão de custo.

5.2 Conclusões sobre a implementação em ambiente discretizado

Embora o ambiente discretizado introduza limitações (no caso, concentração de tensão em contornos irregulares e modelagem imprecisa para vigas diagonais), a implementação foi suficiente para demonstrar as características do algoritmo para o problema de minimização de flexibilidade: a elevada redução do tempo de resolução ao custo de um

aumento pequeno da flexibilidade da topologia resultante (pequeno em relação à redução de tempo), mesmo considerando que as geometrias obtidas pelo pré-dimensionamento tendam a ser sobredimensionadas.

O algoritmo se mostrou eficiente na criação de uma aproximação da geometria final, aumentando a velocidade do processo de projeto e otimização sem ocasionar em grandes prejuízos à qualidade do resultado. Porém, para o âmbito tridimensional isso não se traduziu tão bem, visto que as simplificações usadas não consideram a direção de aplicação das cargas, resultando em topologias com maior perda de rigidez.

5.3 Conclusões sobre a implementação em ambiente contínuo

Para a implementação em ambiente contínuo, utilizando minimização de volume, os resultados também se mostraram promissores: houve grande redução de tempo de resolução, mas volumes finais maiores, embora a redução ainda seja considerável em comparação com o volume da geometria original.

Porém, o sucesso do pré-dimensionamento é dependente também dos parâmetros utilizados para o *pathfinding*, assim como do volume da geometria resultante do pré-dimensionamento. O próprio *pathfinding* apresenta a capacidade de remover cantos internos da geometria, e o uso do sobredimensionamento pode resultar em volumes finais menores e baixo tempo de resolução, ao custo de dimensões finais um pouco maiores.

A principal questão, porém, seria o uso dos critérios de tensão. Os critérios de dimensionamento e o de otimização são diferentes e de certa forma contraditórios, visto que um controla as componentes de tensão e é limitado a vigas, e o outro é mais abrangente e também menos estrito. Usar o critério de von Mises como principal e selecionar os outros com base nele, considerando a possibilidade de sobredimensionamento, pode ser uma solução, mas não há uma forma clara de obter os parâmetros de dimensionamento a partir dele e vice-versa.

5.4 Trabalhos futuros

Os resultados deste projeto abrem a possibilidade de novas pesquisas que visem expandi-lo ou otimizá-lo, como:

1. **Projeto de estruturas tridimensionais:** utilizar as vigas em um ambiente tridimensional para obter estruturas mais complexas;
2. **Estudo de diferentes seções transversais:** no ambiente bidimensional, as seções transversais tendem a ser retangulares. Porém, em ambientes tridimensionais há maior variedade de seções transversais possíveis, que podem resultar em menor uso de material;

3. **Estudar o uso dos critérios de tensão:** obter uma relação entre os critérios de tensão usados, ou talvez modificar o critério de tensão da otimização topológica, contribuiria para ao aumento da automatização do processo;
4. **Desenvolvimento de outros métodos de formação das vigas redimensionadas:** o uso de formas primitivas para o redimensionamento das vigas (como círculos, no caso bidimensional) torna a criação da geometria um processo mais simples de ser implementado. Porém, o número de operações booleanas necessárias o torna lento, e há a possibilidade dos círculos tornarem o contorno mais complexo que o desejado, aumentando o número de elementos necessários e criando regiões de concentração de tensão;
5. **Desenvolvimento de outros métodos de redimensionamento das vigas:** como mencionado anteriormente, o uso de um método que realizasse o redimensionamento ao nível de cada elemento individualmente, por exemplo, resultaria em uma abordagem mais geral e com resultados potencialmente mais otimizados;
6. **Utilizar uma implementação de CSG melhor adaptada ao algoritmo:** uma implementação melhor adaptada poderia aumentar a velocidade de criação da geometria;
7. **Utilização de otimização de forma:** embora o algoritmo apresentado possa ser interpretado como uma forma heurística da otimização de forma, esta também poderia ser utilizada após a etapa de otimização topológica para refinar a geometria e diminuir ainda mais seu volume e dimensões.

REFERÊNCIAS

- AAGE, N.; LAZAROV, B. Lazarov, b.s.: Parallel framework for topology optimization using the method of moving asymptotes. *structural and multidisciplinary optimization* 47, 493-505. **Structural and Multidisciplinary Optimization**, v. 47, 04 2013.
- ALEXANDERSEN, J.; ANDREASEN, C. A review of topology optimisation for fluid-based problems. **Fluids**, v. 5, 03 2020.
- AMESTOY, P.; BUTTARI, A.; L'EXCELLENT, J.-Y.; MARY, T. Performance and scalability of the block low-rank multifrontal factorization on multicore architectures. **ACM Transactions on Mathematical Software**, v. 45, p. 2:1–2:26, 2019.
- AMESTOY, P.; DUFF, I. S.; KOSTER, J.; L'EXCELLENT, J.-Y. A fully asynchronous multifrontal solver using distributed dynamic scheduling. **SIAM Journal on Matrix Analysis and Applications**, v. 23, n. 1, p. 15–41, 2001.
- ANDERSON, E.; BAI, Z.; BISCHOF, C.; BLACKFORD, S.; DEMMEL, J.; DONGARRA, J.; CROZ, J. D.; GREENBAUM, A.; HAMMARLING, S.; MCKENNEY, A.; SORENSEN, D. **LAPACK Users' Guide**. Third. Philadelphia, PA: Society for Industrial and Applied Mathematics, 1999. ISBN 0-89871-447-8 (paperback).
- ANGELO, L.; STEFANO, P. D.; GUARDIANI, E. Search for the optimal build direction in additive manufacturing technologies: A review. **Journal of Manufacturing and Materials Processing**, 07 2020.
- ANSYS, INC. **Ansys® Academic Research Mechanical, Release 18.2, Help System**. [S.l.], 2018.
- BENDSØE, M. Optimal shape design as a material distribution problem. *structural optimization* 1, 193-202. **Structural Optimization**, v. 1, p. 193–202, 01 1989.
- BENDSØE, M.; SIGMUND, O. Material interpolation schemes in topology optimization. **Archive of Applied Mechanics**, v. 69, p. 635–654, 11 1999.
- BENDSØE, M. P.; KIKUCHI, N. Generating optimal topologies in structural design using a homogenization method. **Computer Methods in Applied Mechanics and Engineering**, v. 71, n. 2, p. 197 – 224, 1988. ISSN 0045-7825. Disponível em: <http://www.sciencedirect.com/science/article/pii/0045782588900862>.
- BENDSØE, M. P.; SIGMUND, O. Book. **Topology optimization: theory, methods, and applications**. [S.l.: s.n.]: Springer Science & Business Media, 2013. ISBN 3662050862.
- BERG, M. de; CHEONG, O.; KREVELD, M. van; OVERMARS, M. **Computational geometry: algorithms and applications**. 3. ed. [S.l.: s.n.]: Springer-Verlag Berlin Heidelberg, 2008. ISBN 9783540779735,3540779736.
- BOURDIN, B. Filters in topology optimization. **International Journal for Numerical Methods in Engineering**, v. 50, n. 9, p. 2143–2158, 2001. Disponível em: <https://onlinelibrary.wiley.com/doi/abs/10.1002/nme.116>.

BRUNS, T. E.; TORTORELLI, D. A. Topology optimization of non-linear elastic structures and compliant mechanisms. **Computer Methods in Applied Mechanics and Engineering**, v. 190, n. 26, p. 3443–3459, 2001. ISSN 0045-7825. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0045782500002784>.

BRUNS, T. E.; TORTORELLI, D. A. An element removal and reintroduction strategy for the topology optimization of structures and compliant mechanisms. **International Journal for Numerical Methods in Engineering**, v. 57, n. 10, p. 1413–1430, 2003. Disponível em: <https://onlinelibrary.wiley.com/doi/abs/10.1002/nme.783>.

BUDYNAS, R. G.; NISBETT, K. **Shigley's Mechanical Engineering Design**. New York: McGraw Hill, 2015. ISBN 9789814595285.

CAMBURN, B.; HE, Y.; RAVISELVAM, S.; LUO, J.; WOOD, K. Machine learning based design concept evaluation. **Journal of Mechanical Design**, v. 142, p. 1–48, 10 2019.

CHEN, A.; CAI, K.; ZHAO, Z.-L.; ZHOU, Y.; XIA, L.; XIE, Y. Controlling the maximum first principal stress in topology optimization. **Structural and Multidisciplinary Optimization**, v. 63, 01 2021.

CHEN, Q.; XU, J.; ZHANG, S. Cylindricity and flatness optimization for mechanical parts in additive manufacturing based on tolerance adaptive slicing. **The International Journal of Advanced Manufacturing Technology**, v. 115, n. 11-12, p. 3839–3857, Jun 2021.

CHEN, Y.; LI, Y. **Computational intelligence assisted design in industrial revolution 4.0**. [*S.l.: s.n.*]: Boca Raton, FL CRC Press, Taylor & Francis Group, 2018. ISBN 9781498760669.

CHENG, G. ε -relaxed approach in structural topology optimization. **Structural Optimization**, v. 13, p. 258–266, 01 1997.

DA SILVA, G.; BECK, A.; CARDOSO, E. Topology optimization of continuum structures with stress constraints and uncertainties in loading. **International Journal for Numerical Methods in Engineering**, v. 113, 06 2017.

DEATON, J. D.; GRANDHI, R. V. A survey of structural and multidisciplinary continuum topology optimization: Post 2000. **Struct. Multidiscip. Optim.**, Springer-Verlag, Berlin, Heidelberg, v. 49, n. 1, p. 1–38, jan. 2014. ISSN 1615-147X. Disponível em: <https://doi.org/10.1007/s00158-013-0956-z>.

DENG, H.; TO, A. Topology optimization based on deep representation learning (drl) for compliance and stress-constrained design. **Computational Mechanics**, v. 66, 08 2020.

DIALAMI, N.; CHIUMENTI, M.; CERVERA, M.; ROSSI, R.; GOÑI, U. C.; DOMINGO, M. Numerical and experimental analysis of the structural performance of am components built by fused filament fabrication. **International Journal of Mechanics and Materials in Design**, v. 17, 03 2021.

DIJK, N. van; MAUTE, K.; LANGELAAR, M.; KEULEN, F. Level-set methods for structural topology optimization: A review. **Structural and Multidisciplinary Optimization**, v. 48, 09 2013.

-
- DUMAS, J. **MMA and GCMMA**. [*S.l.: s.n.*]: GitHub, 2018. <https://github.com/jdumas/mma>.
- DUYSINX, P.; BENDSØE, M. P. Topology optimization of continuum structures with local stress constraints. **International Journal for Numerical Methods in Engineering**, v. 43, n. 8, p. 1453–1478, 1998. ISSN 0029-5981.
- FANCELLO, E.; PEREIRA, J. Structural topology optimization considering material failure. **Latin American Journal of Solids and Structures**, v. 1, 11 2003.
- GAO, J.; XIAO, M.; ZHANG, Y.; GAO, L. A comprehensive review of isogeometric topology optimization: Methods, applications and prospects. **Chinese Journal of Mechanical Engineering**, v. 33, p. 87, 11 2020.
- GEUS, R. **The Jacobi-Davidson algorithm for solving large sparse symmetric eigenvalue problems with application to the design of accelerator cavities**. 2002. Tese (Doutorado) — ETH Zurich, Zürich, 2002. Diss., Technische Wissenschaften ETH Zürich, Nr. 14734, 2002. Disponível em: <https://www.research-collection.ethz.ch/handle/20.500.11850/147153>.
- GEUS, R. *et al.* **PySparse - Python Sparse Matrix Library**. 2010. Acesso em: 2 mar. 2021. Disponível em: <https://sourceforge.net/projects/pysparse/files/>.
- GEUZAINÉ, C.; REMACLE, J.-F. Gmsh: A 3-D finite element mesh generator with built-in pre- and post-processing facilities. **International Journal for Numerical Methods in Engineering**, v. 79, n. 11, p. 1309–1331, 2009. Disponível em: <https://onlinelibrary.wiley.com/doi/abs/10.1002/nme.2579>.
- GIRI, J.; SHAHANE, P.; JACHAK, S.; CHADGE, R.; GIRI, P. Optimization of FDM process parameters for dual extruder 3d printer using artificial neural network. **Materials Today: Proceedings**, v. 43, p. 3242–3249, 2021. ISSN 2214-7853. CRMSC-2021. Disponível em: <https://www.sciencedirect.com/science/article/pii/S2214785321009962>.
- GOODNO, B. J.; GERE, J. M. **Mechanics of Materials**. 9. ed. [*S.l.: s.n.*]: Cengage Learning, 2018. ISBN 9781337093347.
- GRAFF, K. F. **Wave motion in elastic solids**. [*S.l.: s.n.*]: Dover Publications, 1975. ISBN 9780814202326,0814202322,0486667456,9780486667454.
- HARRIS, C. R.; MILLMAN, K. J.; WALT, S. J. van der; GOMMERS, R.; VIRTANEN, P.; COUNAPEAU, D.; WIESER, E.; TAYLOR, J.; BERG, S.; SMITH, N. J.; KERN, R.; PICUS, M.; HOYER, S.; KERKWIJK, M. H. van; BRETT, M.; HALDANE, A.; RÍO, J. Fernández del; WIEBE, M.; PETERSON, P.; GÉRARD-MARCHANT, P.; SHEPPARD, K.; REDDY, T.; WECKESSER, W.; ABBASI, H.; GOHLKE, C.; OLIPHANT, T. E. Array programming with NumPy. **Nature**, v. 585, p. 357–362, 2020.
- Hart, P. E.; Nilsson, N. J.; Raphael, B. A formal basis for the heuristic determination of minimum cost paths. **IEEE Transactions on Systems Science and Cybernetics**, v. 4, n. 2, p. 100–107, 1968.
- HIBBELER, R. C. **Mechanics of Materials**. 9th. ed. [*S.l.: s.n.*]: Pearson Education (Prentice Hall), 2014.

HOLMBERG, E.; TORSTENFELT, B.; KLARBRING, A. Stress constrained topology optimization. **Structural and Multidisciplinary Optimization**, v. 48, n. 1, p. 33–47, Feb 2013.

HOLMBERG, E.; TORSTENFELT, B.; KLARBRING, A. Fatigue constrained topology optimization. **Structural and Multidisciplinary Optimization**, v. 50, 01 2014.

HONARPARDAZ, M.; TARKIAN, M.; ÖLVANDER, J.; FENG, X. Finger design automation for industrial robot grippers: A review. **Robotics and Autonomous Systems**, v. 87, p. 104 – 119, 2017. ISSN 0921-8890. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0921889015303171>.

HUANG, X.; XIE, M. **Evolutionary Topology Optimization of Continuum Structures: Methods and Applications**. 1. ed. Chichester, West Sussex, U.K: Wiley, 2010. ISBN 9780470746530.

HUEBNER, K. H.; DEWHIRST, D. L.; SMITH, D. E.; BYROM, T. G. **The Finite Element Method for Engineers**. 4th. ed. [*S.l.: s.n.*]: Wiley, 2001. ISBN 0471370789.

HUNTER, W. **Predominantly solid-void three-dimensional topology optimisation using open source software**. 2009. Dissertação (Mestrado) — University of Stellenbosch. Faculty of Engineering. Dept. of Mechanical and Mechatronic Engineering, 2009.

HUNTER, W. *et al.* **ToPy - Topology optimization with Python**. [*S.l.: s.n.*]: GitHub, 2017. <https://github.com/williamhunter/topy>.

JOHNSON, S. G. **The NLOpt nonlinear-optimization package**. Disponível em: <http://github.com/stevengj/nlopt>.

KAMBAMPATI, S.; GRAY, J. S.; KIM, H. A. Level set topology optimization of structures under stress and temperature constraints. **Computers & Structures**, v. 235, p. 106265, 2020. ISSN 0045-7949. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0045794920300687>.

KLERK, A. de; GROENWOLD, A. On the robustness of the Q4 membrane element. *In*: ZINGONI, A. (ed.). **Progress in Structural Engineering, Mechanics and Computation**. Cape Town, Africa: [*S.l.: s.n.*], 2004. p. 961–966.

Kusiak, A.; Salustri, F. A. Computational intelligence in product design engineering: Review and trends. **IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)**, v. 37, n. 5, p. 766–778, 2007.

LAVAGNO, L.; MARKOV, I.; MARTIN, G.; SCHEFFER, L. **Electronic Design Automation for IC Implementation, Circuit Design, and Process Technology**. [*S.l.: s.n.*]: CRC Press, 2016. ISBN 978-1-4822-5460-0.

LE, C.; NORATO, J.; BRUNS, T.; HA, C.; TORTORELLI, D. Stress-based topology optimization for continua. **Structural and Multidisciplinary Optimization**, v. 41, p. 605–620, 04 2010.

LEON, D. M. D.; GONÇALVES, J. F.; SOUZA, C. E. de. Stress-based topology optimization of compliant mechanisms design using geometrical and material nonlinearities. **Structural and Multidisciplinary Optimization**, v. 62, n. 1, p. 231–248, 2020. ISSN 1615-147X.

MATOS, M. A.; ROCHA, A. M. D.; COSTA, L. Many-objective optimization of build part orientation in additive manufacturing. **The International Journal of Advanced Manufacturing Technology**, p. 1–16, 2020.

MATWEB, LLC. **AISI 1020 Steel, normalized at 870°C (1600°F)**. [*S.l.*]. <http://www.matweb.com/search/DataSheet.aspx?MatGUID=2740581369f3421dac0d8c91294b5935&ckck=1>, accessed: 2021-02-07.

MCLEOD, R.; ALTED, F.; VALENTINO, A.; MENTEN, G. de; WIEBE, M.; CGOHLKE; BEDINI, A.; MAMREHN; TECHTONIK anatoly; ERB, S.; SHADCHIN, A.; BUNIN, A.; KOUIJ, T.; PAVLYK, O.; JELLOUL, M. B.; GARRISON, L.; HURTADO, D. M.; CAREY, C.; SARAHAN, M.; COX, D.; PLESIVČAK, Z.; BORGDORFF, J.; COURBET, C.; DICKINSON, M.; LEITAO, B.; LAAT, A. de; PITROU, A.; PORTNOY, A.; ORTEGA, A. L.; BÖHN, A. **pydata/numexpr: NumExpr v2.6.9**. Zenodo, 2018. Disponível em: <https://doi.org/10.5281/zenodo.2483274>.

MENG, Q.; XU, B.; WANG, C.; ZHAO, L. Stress constrained thermo-elastic topology optimization based on stabilizing control schemes. **Journal of Thermal Stresses**, Taylor & Francis, v. 43, n. 8, p. 1040–1068, 2020. Disponível em: <https://doi.org/10.1080/01495739.2020.1766391>.

MEURER, A.; SMITH, C.; PAPROCKI, M.; ČERTÍK, O.; KIRPICHEV, S.; ROCKLIN, M.; KUMAR, A.; IVANOV, S.; MOORE, J.; SINGH, S.; RATHNAYAKE, T.; VIG, S.; GRANGER, B.; MULLER, R.; BONAZZI, F.; GUPTA, H.; VATS, S.; JOHANSSON, F.; PEDREGOSA, F.; SCOPATZ, A. SymPy: Symbolic computing in python. **PeerJ Computer Science**, v. 3, p. e103, 01 2017.

Millman, K. J.; Aivazis, M. Python for scientists and engineers. **Computing in Science & Engineering**, v. 13, n. 2, p. 9–12, 2011.

MOHARRAMI, H.; GRIERSON, D. E. Computer-automated design of reinforced concrete frameworks. **Journal of Structural Engineering**, v. 119, n. 7, p. 2036–2058, 1993. Disponível em: <https://ascelibrary.org/doi/abs/10.1061/%28ASCE%290733-9445%281993%29119%3A7%282036%29>.

MUNK, D. J.; VIO, G. A.; STEVEN, G. P. Topology and shape optimization methods using evolutionary algorithms: a review. **Structural and Multidisciplinary Optimization**, v. 52, n. 3, p. 613–631, 2015. ISSN 1615-147X.

NEW Perspectives on Design Automation: Celebrating the 40th Anniversary of the ASME Design Automation Conference. **Journal of Mechanical Design**, v. 137, n. 5, 05 2015. ISSN 1050-0472. 050301. Disponível em: <https://doi.org/10.1115/1.4030256>.

NGUYEN, S. H.; KIM, H.-G. Stress-constrained shape and topology optimization with the level set method using trimmed hexahedral meshes. **Computer Methods in Applied Mechanics and Engineering**, v. 366, p. 113061, 2020. ISSN 0045-7825. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0045782520302450>.

OH, S.; JUNG, Y.; KIM, S.; LEE, I.; KANG, N. Deep generative design: Integration of topology optimization and generative models. **Journal of Mechanical Design**, v. 141, p. 1, 07 2019.

OH, S.; JUNG, Y.; LEE, I.; KANG, N. Design automation by integrating generative adversarial networks and topology optimization. *In: . [S.l.: s.n.]*, 2018. p. V02AT03A008.

OH, Y.; KO, H.; SPROCK, T.; BERNSTEIN, W.; KWON, S. Part decomposition and evaluation based on standard design guidelines for additive manufacturability and assemblability. **Additive Manufacturing**, 11 2020.

Open Cascade. **Open CASCADE Technology**. Acesso em: 12 mar. 2021. Disponível em: <https://www.opencascade.com/open-cascade-technology/>.

PARÍS, J.; NAVARRINA, F.; COLOMINAS, I.; CASTELEIRO, M. Topology optimization of continuum structures with local and global stress constraints. **Structural and Multidisciplinary Optimization**, v. 39, p. 419–437, 2009.

PEDERSEN, P. Some general optimal design results using anisotropic, power law nonlinear elasticity. **Structural optimization**, v. 15, p. 73–80, 1998.

PEREIRA, J.; FANCELLO, E.; BARCELLOS, C. Topology optimization of continuum structures with material failure constraints. **Structural and Multidisciplinary Optimization**, v. 26, n. 1-2, p. 50–66, Jan 2004.

PICELLI, R.; TOWNSEND, S.; BRAMPTON, C.; NORATO, J.; KIM, H. Stress-based shape and topology optimization with the level set method. **Computer Methods in Applied Mechanics and Engineering**, v. 329, p. 1 – 23, 2018. ISSN 0045-7825. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0045782517304395>.

PLOCHER, J.; PANESAR, A. Review on design and structural optimisation in additive manufacturing: Towards next-generation lightweight structures. **Materials & Design**, v. 183, p. 108164, 2019. ISSN 0264-1275. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0264127519306021>.

PRADEL, P.; ZHU, Z.; BIBB, R.; MOULTRIE, J. **Complexity is not for free: the impact of component complexity on additive manufacturing build time**. Loughborough University, 2017. Disponível em: <https://hdl.handle.net/2134/24338>.

Python Software Foundation. **Python 3.0 Release**. 2008. Acesso em: 2 mar. 2021. Disponível em: <https://www.python.org/download/releases/3.0/>.

Python Software Foundation. **Sunsetting Python 2**. 2020. Acesso em: 2 mar. 2021. Disponível em: <https://www.python.org/doc/sunset-python-2/>.

Python Software Foundation. **General Python FAQ**. 2021. Acesso em: 2 mar. 2021. Disponível em: <https://docs.python.org/3/faq/general.html>.

QUERIN, O.; STEVEN, G.; XIE, Y. Evolutionary structural optimisation (ESO) using a bidirectional algorithm. **Engineering Computations**, v. 15, p. 1031–1048, 12 1998.

QUERIN, O. M.; VICTORIA, M.; ALONSO, C.; ANSOLA, R.; MARTÍ, P. **Topology design methods for structural optimization**. London: Academic Press, 2017. ISBN 9780081009161.

RAGSDELL, K. M. Design and automation. **Journal of Mechanical Design**, v. 102, n. 3, p. 424–429, 07 1980. ISSN 0161-8458. Disponível em: <https://doi.org/10.1115/1.3254764>.

- REBAY, S. Efficient unstructured mesh generation by means of delaunay triangulation and bowyer-watson algorithm. **Journal of Computational Physics**, v. 106, n. 1, p. 125–138, 1993. ISSN 0021-9991. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0021999183710971>.
- SABISTON, G.; KIM, I. Y. 3D topology optimization for cost and time minimization in additive manufacturing. **Structural and Multidisciplinary Optimization**, v. 61, p. 1–18, 02 2020.
- SAITOU, K.; IZUI, K.; NISHIWAKI, S.; PAPALAMBROS, P. A survey of structural optimization in mechanical product development. **Journal of Computing and Information Science in Engineering**, v. 5, 09 2005.
- SENHORA, F.; NO, O. G.-L.; MENEZES, I.; PAULINO, G. Topology optimization with local stress constraints: a stress aggregation-free approach. **Structural and Multidisciplinary Optimization**, v. 62, 10 2020.
- SHARPE, C.; WIEST, T.; WANG, P.; SEEPERSAD, C. A comparative evaluation of supervised machine learning classification techniques for engineering design applications. **Journal of Mechanical Design**, v. 141, p. 1–38, 09 2019.
- SHIM, P. Y.; MANOOCHEHRI, S. Generating optimal configurations in structural design using simulated annealing. **International Journal for Numerical Methods in Engineering**, v. 40, n. 6, p. 1053–1069, 1997. Disponível em: <https://onlinelibrary.wiley.com/doi/abs/10.1002/%28SICI%291097-0207%2819970330%2940%3A6%3C1053%3A%3AAID-NME97%3E3.0.CO%3B2-I>.
- SHU, D.; CUNNINGHAM, J.; STUMP, G.; MILLER, S.; YUKISH, M.; SIMPSON, T.; TUCKER, C. 3D design using generative adversarial networks and physics-based validation. **Journal of Mechanical Design**, v. 142, p. 1–51, 11 2019.
- SIGMUND, O. A 99 line topology optimization code written in matlab. structural and multidisciplinary optimization 21, 120-127. **Structural and Multidisciplinary Optimization**, v. 21, p. 120–127, 04 2001.
- SIGMUND, O. Morphology-based black and white filters for topology optimization. **Structural and Multidisciplinary Optimization**, v. 33, n. 4-5, p. 401–424, Jan 2007.
- SIGMUND, O.; MAUTE, K. Topology optimization approaches a comparative review. **Structural and Multidisciplinary Optimization**, v. 48, n. 6, p. 1031–1055, 2013. ISSN 1615-147X.
- SIGMUND, O.; PETERSSON, J. Numerical instabilities in topology optimization: A survey on procedures dealing with checkerboards, mesh-dependencies and local minima. **Structural Optimization**, v. 16, n. 1, p. 68–75, 1998. ISSN 0934-4373.
- SILVA, G.; CARDOSO, E.; BECK, A. Non-probabilistic robust continuum topology optimization with stress constraints. **Structural and Multidisciplinary Optimization**, v. 59, 04 2019.
- SILVA, G. A.; BECK, A. T.; CARDOSO, E. L. Topology optimization of continuum structures with stress constraints and uncertainties in loading. **International Journal for Numerical Methods in Engineering**, v. 113, n. 1, p. 153–178, 2018. Disponível em: <https://onlinelibrary.wiley.com/doi/abs/10.1002/nme.5607>.

SILVA, G. A.; BECK, A. T.; SIGMUND, O. Topology optimization of compliant mechanisms considering stress constraints, manufacturing uncertainty and geometric nonlinearity. **Computer Methods in Applied Mechanics and Engineering**, v. 365, p. 31, 2020. ISSN 0045-7825.

SILVA, G. A. d.; AAGE, N.; BECK, A. T.; SIGMUND, O. Three-dimensional manufacturing tolerant topology optimization with hundreds of millions of local stress constraints. **International Journal for Numerical Methods in Engineering**, v. 122, n. 2, p. 548–578, 2021. Disponível em: <https://onlinelibrary.wiley.com/doi/abs/10.1002/nme.6548>.

SILVA, G. A. da; BECK, A. T.; SIGMUND, O. Stress-constrained topology optimization considering uniform manufacturing uncertainties. **Computer Methods in Applied Mechanics and Engineering**, v. 344, p. 512 – 537, 2019. ISSN 0045-7825. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0045782518305231>.

SILVA, G. A. da; BECK, A. T.; SIGMUND, O. Stress-constrained topology optimization considering uniform manufacturing uncertainties. **Computer Methods in Applied Mechanics and Engineering**, v. 344, p. 512 – 537, 2019. ISSN 0045-7825. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0045782518305231>.

SIVAPURAM, R.; PICELLI, R. Topology optimization of binary structures using integer linear programming. **Finite Elements in Analysis and Design**, v. 139, p. 49 – 61, 2018. ISSN 0168-874X. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0168874X17305619>.

SONG, Y.; MA, Q.; HE, Y.; ZHOU, M.; WANG, M. Stress-based shape and topology optimization with cellular level set in B-splines. **Structural and Multidisciplinary Optimization**, v. 62, 11 2020.

SOSNOVIK, I.; OSELEDETS, I. Neural networks for topology optimization. **Russian Journal of Numerical Analysis and Mathematical Modelling**, v. 34, 09 2017.

STOECKLI, F.; SHEA, K. Topology optimization of rigid-body systems considering collision avoidance. **Journal of Mechanical Design**, v. 142, p. 1–15, 01 2020.

SUN, Y.; XU, L.; YANG, J.; LUETH, T. Automatic design in matlab using pde toolbox for shape and topology optimization. *In: . [S.l.: s.n.]*, 2019.

SVANBERG, K. The method of moving asymptotes—a new method for structural optimization. **International Journal for Numerical Methods in Engineering**, v. 24, n. 2, p. 359–373, 1987. Disponível em: <https://onlinelibrary.wiley.com/doi/abs/10.1002/nme.1620240207>.

SVANBERG, K. A class of globally convergent optimization methods based on conservative convex separable approximations. **SIAM Journal on Optimization**, v. 12, p. 555–573, 01 2002.

SVANBERG, K. **MMA and GCMMA - Two Methods for Nonlinear Optimization**. [S.l.], 2007. Disponível em: <https://people.kth.se/~krille/mmagcmma.pdf>.

TEITELBAUM, G. A. **PROPOSED BUILD GUIDELINES FOR USE IN FUSED DEPOSITION MODELING TO REDUCE BUILD TIME AND MATERIAL VOLUME**. 2009. Tese (Doutorado) — Department of Mechanical Engineering. University of Maryland (College Park, Md.), 2009.

THILLAITHEVAN, D.; BRUCE, P.; SANTER, M. Stress-constrained optimization using graded lattice microstructures. **Structural and Multidisciplinary Optimization**, Oct 2020.

VIRTANEN, P.; GOMMERS, R.; OLIPHANT, T. E.; HABERLAND, M.; REDDY, T.; COURNAPEAU, D.; BUROVSKI, E.; PETERSON, P.; WECKESSER, W.; BRIGHT, J.; van der Walt, S. J.; BRETT, M.; WILSON, J.; MILLMAN, K. J.; MAYOROV, N.; NELSON, A. R. J.; JONES, E.; KERN, R.; LARSON, E.; CAREY, C. J.; POLAT, İ.; FENG, Y.; MOORE, E. W.; VanderPlas, J.; LAXALDE, D.; PERKTOLD, J.; CIMRMAN, R.; HENRIKSEN, I.; QUINTERO, E. A.; HARRIS, C. R.; ARCHIBALD, A. M.; RIBEIRO, A. H.; PEDREGOSA, F.; van Mulbregt, P.; SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. **Nature Methods**, v. 17, p. 261–272, 2020.

WHITE, D. A.; ARRIGHI, W. J.; KUDO, J.; WATTS, S. E. Multiscale topology optimization using neural network surrogate models. **Computer Methods in Applied Mechanics and Engineering**, v. 346, p. 1118 – 1135, 2019. ISSN 0045-7825. Disponível em: <http://www.sciencedirect.com/science/article/pii/S004578251830450X>.

WIBERG, A.; PERSSON, J.; ÖLVANDER, J. Design for additive manufacturing – a review of available design methods and software. **Rapid Prototyping Journal**, ahead-of-print, 07 2019.

XIA, L.; XIA, Q.; HUANG, X. D.; XIE, Y. M. Bi-directional evolutionary structural optimization on advanced structures and materials: A comprehensive review. **Archives of Computational Methods in Engineering**, v. 25, n. 2, p. 437–478, 2018. ISSN 1134-3060.

XIA, L.; ZHANG, L.; XIA, Q.; SHI, T. Stress-based topology optimization using bi-directional evolutionary structural optimization method. **Computer Methods in Applied Mechanics and Engineering**, v. 333, p. 356 – 370, 2018. ISSN 0045-7825. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0045782518300379>.

XU, B.; HAN, Y.; ZHAO, L. Bi-directional evolutionary stress-based topology optimization of material nonlinear structures. **Structural and Multidisciplinary Optimization**, Nov 2020.

YANG, R.-J.; CHEN, C.-J. Stress-based topology optimization. **Structural and Multidisciplinary Optimization**, v. 12, p. 98–105, 10 1996.

YUQIU, L.; YIN, X. Generalized conforming triangular membrane element with vertex rigid rotational freedoms. **Finite Elements in Analysis and Design**, v. 17, n. 4, p. 259–271, 1994. ISSN 0168-874X. Disponível em: <https://www.sciencedirect.com/science/article/pii/0168874X94900027>.

ZARGHAM, S.; WARD, T.; RAMLI, R.; BADRUDDIN, I. Topology optimization: a review for structural designs under vibration problems. **Structural and Multidisciplinary Optimization**, v. 53, 06 2016.

ZHANG, W.; LI, D.; YUAN, J.; SONG, J.; GUO, X. A new three-dimensional topology optimization method based on moving morphable components (MMCs). **Comput. Mech.**, Springer-Verlag, Berlin, Heidelberg, v. 59, n. 4, p. 647–665, abr. 2017. ISSN 0178-7675. Disponível em: <https://doi.org/10.1007/s00466-016-1365-0>.

ZHANG, W.; LI, D.; ZHOU, J.; DU, Z.; LI, B.; GUO, X. A moving morphable void (MMV)-based explicit approach for topology optimization considering stress constraints. **Computer Methods in Applied Mechanics and Engineering**, v. 334, p. 381 – 413, 2018. ISSN 0045-7825. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0045782518300574>.

ZHANG, X.; ZHU, B. **Topology Optimization of Compliant Mechanisms**. Singapore: Springer Singapore, 2018. ISBN 9789811304323.

ZHANG, Y.; BERNARD, A.; VALENZUELA, J. M.; KARUNAKARAN, K. Fast adaptive modeling method for build time estimation in additive manufacturing. **CIRP Journal of Manufacturing Science and Technology**, v. 10, p. 49–60, 2015. ISSN 1755-5817. Disponível em: <https://www.sciencedirect.com/science/article/pii/S1755581715000292>.

ZHOU, M.; ROZVANY, G. On the validity of ESO type methods in topology optimization. **Structural and Multidisciplinary Optimization**, v. 21, p. 80–83, 03 2001.

ZHU, B. L.; ZHANG, X. M.; ZHANG, H. C.; LIANG, J. W.; ZANG, H. Y.; LI, H.; WANG, R. X. Design of compliant mechanisms using continuum topology optimization: A review. **Mechanism and Machine Theory**, v. 143, p. 34, 2020. ISSN 0094-114X.

ZUO, W.; SAITOU, K. Multi-material topology optimization using ordered SIMP interpolation. **Structural and Multidisciplinary Optimization**, v. 55, 02 2017.

APÊNDICES

APÊNDICE A – O PROGRAMA TOPY

A.1 Visão geral

O ToPy foi desenvolvido de 2005 a 2009 (Hunter *et al.*, 2017) no Departamento de Engenharia Mecânica e Mecatrônica da Universidade de Stellenbosch, na África do Sul por Hunter (2009). O objetivo foi desenvolver uma implementação de otimização topológica que utilizasse somente ferramentas gratuitas de código aberto, como alternativa ao uso de programas e linguagens proprietários, como o MatLab, que é comumente utilizado para desenvolver implementações compactas de otimização topológica, como a de 99 linhas desenvolvida por Sigmund (2001), utilizada como base para a implementação (Hunter, 2009).

Sendo assim, o ToPy é um programa de otimização topológica baseado em SIMP que utiliza filtragem de sensibilidades e o método do critério de ótimo (OC) como base para a implementação, apresentando também algumas extensões em relação ao código utilizado como base.

A.2 Ferramentas utilizadas

A.2.1 Python 2

O Python é uma linguagem interpretada de alto nível orientada a objetos (Python Software Foundation, 2021). Foi inicialmente criado com o objetivo de ser uma linguagem de *scripting*, mas a participação da comunidade desenvolvedora e subsequentes incrementos às funcionalidades da linguagem a tornaram uma linguagem de programação de propósito geral.

A terceira versão da linguagem, conhecida como Python 3, foi lançada em 2008, sem compatibilidade total com a versão anterior, o Python 2 (Python Software Foundation, 2008), significando que programas e bibliotecas desenvolvidos para versão anterior teriam de ser adaptados para essa nova versão para funcionar plenamente. O Python 2 continuou a ser mantido, na forma de sua versão 2.7, até 1 de janeiro de 2020, quando foi declarado seu fim de vida (Python Software Foundation, 2020).

Embora a maior parte das bibliotecas e programas desenvolvidos com o Python 2 já tenham migrado à nova versão, bibliotecas e programas que deixaram de ser mantidos não conseguem mais funcionar plenamente.

A.2.2 PySparse

O PySparse é uma biblioteca de resolução de problemas matriciais que utilizam matrizes esparsas (Geus, 2002). Foi criada utilizando Python e C++, com o objetivo de facilitar a resolução de equações de Maxwell com o algoritmo de Jacobi-Davidson. O Python foi utilizado como interface para facilitar o uso da biblioteca (em especial para sua aplicação na biblioteca de elementos finitos PyFemax), e a linguagem C++, para que a implementação tivesse maior desempenho.

O PySparse foi desenvolvido em Python 2 e não apresenta novas versões desde 2010 (Geus *et al.*, 2010), não havendo sido atualizado para o Python 3.

No ToPy, o PySparse foi utilizado para armazenar a matriz rigidez (\mathbf{K}) e realizar os cálculos de elementos finitos.

A.2.3 NumPy

O NumPy é uma biblioteca de cálculos matriciais feita em Python e C, com o objetivo de facilitar seu uso (graças à interface em Python) e ao mesmo tempo oferecer um código de alto desempenho, sendo aplicado em diversas áreas acadêmicas, como na física, química e engenharia (Harris *et al.*, 2020). Além de oferecer uma interface intuitiva para criar e utilizar vetores e matrizes n-dimensionais, também é utilizada como base para outras bibliotecas com aplicações mais específicas, como a SciPy (computação científica), Pandas (estatística), Astropy (astronomia), Biopython (biologia), entre outras.

O NumPy foi utilizado para armazenar e realizar operações sobre matrizes e vetores densos, como a matriz de densidades e os vetores força e deslocamento.

A.2.4 SymPy

O SymPy é uma biblioteca de computação simbólica, utilizada para obter computacionalmente resultados analíticos de equações com variáveis incógnitas, incluindo integração e derivação, sendo baseada no NumPy (Meurer *et al.*, 2017).

No ToPy, foi utilizado para calcular a matriz rigidez elementar de cada elemento.

A.2.5 Outras bibliotecas

Para a visualização dos resultados, o ToPy utiliza a biblioteca Matplotlib para problemas bidimensionais e PyVTK para problemas tridimensionais.

Também são usadas as bibliotecas Typing, para documentação do código, e Pathlib, para melhor lidar com caminhos de diretórios e arquivos no sistema.

A.3 Implementação

A.3.1 Capacidades e limitações

O ToPy é capaz de resolver problemas lineares de minimização de flexibilidade, síntese de mecanismos e condução de calor, tanto em duas quanto em três dimensões, utilizando topologias iniciais retangulares (em 2D) ou em formato de prisma retangular (em 3D), mas sem suporte a otimização multiobjetivo. Utiliza apenas restrição de volume, com dois critérios de parada possíveis (iteração máxima ou critério de convergência). As características do material e dos elementos são sempre tratadas como unitárias. Em duas dimensões, assume-se regime plano de tensões.

O programa não apresenta interface gráfica, os problemas e seus parâmetros são descritos inteiramente em arquivos (um por problema). As condições de contorno são inseridas por nó. Elementos ativos (que devem ser mantidos na topologia) e passivos (que não devem estar presentes) também podem ser especificados.

A.3.2 Elementos

Apenas elementos de lados de dimensões iguais são utilizados. Há oito elementos diferentes, cinco bidimensionais e três tridimensionais:

1. **Q4**: elemento linear elástico simples com formato quadrilátero regular;
2. **Q4T**: elemento linear para problemas de condução de calor com formato quadrilátero regular;
3. **Q4bar**: adaptação do Q4 para obter melhor precisão em problemas de flexão (Klerk; A.Groenwold, 2004);
4. **Q5B**: elemento quadrado linear elástico com equilíbrio penalizado (Hunter, 2009, pp. 24-26);
5. **Q4a5B**: aperfeiçoamento do elemento Q4bar (Hunter, 2009, pp. 26-28);
6. **H8**: elemento linear elástico simples com formato hexagonal regular;
7. **H18B**: versão tridimensional do elemento Q4a5B;
8. **H8T**: elemento linear para problemas de condução de calor com formato hexagonal regular.

Os elementos são criados utilizando o SymPy: as funções de forma são criadas simbolicamente e submetidas a derivações para formar a matriz de interpolação de deformação elementar (**B**). A matriz linear elástica (**C**) é criada de forma similar. As matrizes são utilizadas então para obter a matriz rigidez (**K**), que utiliza integração simbólica para

obter seu resultado, e os símbolos são então substituídos pelos seus respectivos valores. Essa integração é realizada somente na primeira vez que o programa é executado, sendo então salva como arquivo.

A.3.3 Análise de elementos finitos

O PySparse é utilizado para criar e armazenar a matriz esparsa \mathbf{K} , e também para resolver a equação matricial do método. Em problemas bidimensionais, a equação é resolvida de forma direta por fatorização LU. Para problemas tridimensionais, é resolvida de forma iterativa por gradiente conjugado pré-condicionado (PCG), utilizando sobre-relaxamento simétrico sucessivo (SSOR) como pré-condicionador.

Os problemas bidimensionais são resolvidos por um método direto por ser mais exato e o número de graus de liberdade ser relativamente pequeno (8 por elemento). Para problemas tridimensionais, o número de graus de liberdade é consideravelmente maior (24 por elemento), significando que para problemas com grande número de elementos o método direto requereria uma grande quantidade de memória, por isso utiliza-se uma abordagem iterativa.

A.3.4 Análise de sensibilidades

As sensibilidades são calculadas conforme descrito por Sigmund (2001), com adição de cálculos específicos para condução de calor e síntese de mecanismos. A implementação utiliza funções de matemática vetorial e tensorial do NumPy para obter resultados mais rapidamente.

A.3.5 Restrição do espaço de solução

Foi utilizada uma filtragem de sensibilidades de forma iterativa, como implementado por Sigmund (2001).

Também foi introduzido um “filtro de escala de cinza” (GSF), uma modificação do OC baseada em otimização sequencial aproximada (SAO) que consiste em adicionar um fator exponencial q à atualização das densidades, resultando em:

$$\rho_{\mu+1} = \begin{cases} \max\{(1 - \zeta)\rho^\mu, \rho_{min}\} & \text{se } [\rho^\mu(B^\mu)^\eta]^q \leq \max\{(1 - \zeta)\rho^\mu, \rho_{min}\}, \\ \min\{(1 + \zeta)\rho^\mu, 1\} & \text{se } \min\{(1 + \zeta)\rho^\mu, 1\} \leq [\rho^\mu(B^\mu)^\eta]^q, \\ [\rho^\mu(B^\mu)^\eta]^q & \text{caso contrário} \end{cases} \quad (128)$$

A.3.6 Atualização da matriz de densidades

A atualização da matriz de densidades pode ser feita de forma: recíproca, exponencial e diagonal quadrática. A matriz é atualizada de forma iterativa, utilizando as

operações vetoriais do NumPy quando possível. O programa permite a especificação dos seguintes parâmetros:

1. **ETA:** fator de amortecimento (controla a velocidade de variação da densidade);
2. **APPROX:** tipo de aproximação;
3. **P_FAC:** fator exponencial p inicial (do SIMP);
4. **P_INCR:** incremento de p a cada iteração;
5. **P_CON:** número de iterações a serem feitas antes do primeiro incremento a p ;
6. **P_MAX:** valor máximo de p ;
7. **Q_FAC:** fator exponencial q inicial (filtro de escala de cinza - GSF);
8. **Q_INCR:** incremento de q a cada iteração;
9. **Q_CON:** número de iterações a serem feitas antes do primeiro incremento a q ;
10. **Q_MAX:** valor máximo de q ;

Apenas **P_FAC** e **ETA** devem ser obrigatoriamente especificados.

A.3.7 Resultados

Devido à ausência de interface gráfica, os resultados numéricos são exibidos na linha de comando e os resultados gráficos são salvos como arquivo. Para problemas bidimensionais, são salvos no formato PNG, e para problemas tridimensionais, no formato VTK.

São exibidos a cada iteração: o número da iteração, o valor da função objetivo, o volume da topologia, a máxima variação absoluta da matriz de densidades, o fator p , o fator q , o η médio e a quantidade de elementos totalmente sólidos ou vazios em relação ao volume total.



EESC • USP