

UNIVERSIDADE FEDERAL DE ALAGOAS  
CAMPUS A.C. SIMÕES  
CENTRO DE TECNOLOGIA  
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA CIVIL

PEDRO HENRIQUE RIOS SILVEIRA

**MODELAGEM E OTIMIZAÇÃO DE COMPÓSITOS  
TERMO-ATIVADOS**

Maceió/AL  
Julho de 2019

PEDRO HENRIQUE RIOS SILVEIRA

## MODELAGEM E OTIMIZAÇÃO DE COMPÓSITOS TERMO-ATIVADOS

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Engenharia Civil, do Centro de Tecnologia, da Universidade Federal de Alagoas, como requisito parcial para obtenção do título de Mestre em Engenharia Civil.

Orientador: Prof. Dr. Eduardo Nobre Lages  
Coorientador: Prof. Dr. Adeildo Soares Ramos Júnior

Maceió/AL  
Julho de 2019

**Catálogo na fonte**  
**Universidade Federal de Alagoas**  
**Biblioteca Central**

Bibliotecário: Marcelino de Carvalho

S587m Silveira, Pedro Henrique Rios.  
Modelagem e otimização de compósitos termo-ativados / Pedro Henrique Rios Silveira. - 2019.  
95 f. : il. color.

Orientador: Eduardo Nobre Lages.  
Coorientador: Adeildo Soares Ramos Júnior.  
Dissertação (Mestrado em Engenharia Civil) – Universidade Federal de Alagoas. Centro de Tecnologia. Maceió, 2019.

Bibliografia: f. 94-95.

1. MATLAB (Programa de computador). 2. Otimização estrutural. I. Título.

CDU: 620.17:004.421



Universidade Federal de Alagoas – UFAL  
Unidade Acadêmica Centro de Tecnologia – CTEC  
Programa de Pós-Graduação em Engenharia Civil – PPGEC



## MODELAGEM E OTIMIZAÇÃO DE COMPÓSITOS TERMO-ATIVADOS

PEDRO HENRIQUE RIOS SILVEIRA

Dissertação submetida à banca examinadora do Programa de Pós-Graduação em Engenharia Civil da Universidade Federal de Alagoas e aprovada no dia 12 do mês de julho do ano de 2019.

Banca Examinadora:

**Prof. Dr. Eduardo Nobre Lages**  
(Orientador – PPGEC/UFAL)

**Prof. Dr. Adeildo Soares Ramos Júnior**  
(Coorientador - PPGEC/UFAL)

**Prof. Dr. Márcio André Araújo Cavalcante**  
(Avaliador Interno – PPGEC/UFAL)

**Prof. Dr. Anderson Pereira**  
(Avaliador Externo – PUC-RJ)

*Dedico este trabalho a todos que colaboraram com a minha formação enquanto ser humano, pesquisador e engenheiro.*

## AGRADECIMENTOS

Felizmente, tenho muitos a quem agradecer; tantos que não caberiam neste espaço se citados um a um. Dentre os muitos, no âmbito pessoal, escolho meus pais, pela criação privilegiada e apoio incondicional que tive durante toda a minha formação e vida; meu orientador, pela disposição em ajudar e conhecimento e paciência ímpares que sempre disponibilizou para me atender; os professores que marcaram minha vida acadêmica e me deram as ferramentas necessárias para a profissão; e minha companheira de jornada, pelo apoio, suporte, carinho e ajuda, sem as quais a caminhada teria sido ainda mais difícil.

No aspecto técnico, não poderia deixar de citar meu coorientador, dotado de um conhecimento invejável e uma capacidade analítica ímpar; minha banca de qualificação, pelas valorosas sugestões e palavras de elogio; o corpo técnico do PPGEC e do LCCV, que viabilizaram a empreitada; e todos os órgãos de apoio e fomento que sustentam a ciência brasileira contra as muitas agressões que ela sofre.

Aos demais, que não foram diretamente citados, mas que sabem que foram importantes para mim por quaisquer motivos que sejam, sinceramente, meu muito obrigado. Viver é uma aventura muito mais feliz quando se tem pessoas queridas e, por sorte, eu as tenho na figura de amigos e parentes com quem divido momentos que levarei para sempre comigo.

*Segue teu destino,  
Rega as tuas plantas,  
Ama tuas rosas.  
O resto é sombra  
De árvores alheias.*

*Odes - Ricardo Reis*

## RESUMO

O avanço da tecnologia de impressão 3D tem gerado uma revolução em termos de manufatura, permitindo a impressão de peças com geometrias complexas que podem ser definidas com base em critérios técnicos antes impraticáveis de serem executados. Dentro deste aspecto, uma aplicação que tem surgido é a de compósitos termo-ativados, nos quais a presença de um ou mais materiais com efeito de memória de forma permite a deformação programada do compósito após a impressão, contanto que o mesmo seja previamente tratados por um conjunto de estímulos termo-mecânicos. A capacidade dessa classe de compósitos de evoluir sua geometria ao longo do tempo motiva o interesse de se controlar precisamente a geometria prescrita para a configuração deformada. Neste contexto, este trabalho desenvolve formulações matemáticas, estratégias numéricas e uma base computacional para a otimização topológica da geometria da região de inclusão em compósitos termo-ativados, visando permitir a impressão e tratamento de peças que se deformem para configurações prescritas através do fenômeno de memória de forma. O documento tem como objetivo permitir a definição da geometria ótima para a impressão da região de inclusão dos compósitos em questão, viabilizando seu uso em aplicações industriais diversas. Sob esta base computacional, um exemplo de verificação é discutido, mostrando excelente concordância entre uma solução analítica obtida para um problema simplificado e a solução numérica usando o modelo formulado. Também são discutidos vários exemplos ilustrativos, nos quais o efeito dos parâmetros do modelo na resposta do algoritmo de otimização são apresentados e pontuados formalmente, igualmente com resultados correspondentes aos esperados em termos de coerência física. O sucesso da formulação simplificada apresentada, da base computacional e das estratégias desenvolvidas podem ser um excelente ponto de partida sob o qual novas considerações e melhores aproximações possam ser feitas em trabalhos futuros. Tais expansões, embasadas pelo arcabouço teórico aqui firmado, permitirão a aplicação do problema proposto em contextos que permeiam diversas indústrias.

**Palavras-chave:** Memória de forma; Impressão 3D e 4D; MATLAB; Otimização Topológica, Algoritmo de Levenberg-Marquardt.

## ABSTRACT

Advancements in 3D printing technology are generating a revolution in terms of manufacture, allowing the printing of objects in complex geometries that can be generated through the use of technical criteria that were previously impractical to execute. On that aspect, one application that has gained popularity is of thermo-activated composites, on which the presence of one or more materials with shape-memory properties allows the programmed deformation of the composites after printing, as long as they are previously trained with a specific set of thermo-mechanical stimuli. The capacity of a class of polymers to deform over time motivates the interest in controlling precisely the geometry prescribed for their deformed configuration. Within that context, this work details the development of mathematical equations, numerical strategies and a computational base for the topology optimization of the inclusion region in thermo-activated composites, aiming to allow the printing and treatment of pieces that deform to prescribed configurations based on the shape-memory effect that some polymers present. It intends to allow the optimum design of polymeric composite pieces in order to make them viable for applications on several industries. On top of the computational base, an analytical example is constructed to check the developed formulations and shows good agreement between the exact solution and numerical results based on such equations. Illustrative examples are also presented where several parameters are varied and the effect they have on the optimization algorithm are discussed. This also shows results that are considered good and physically coherent. The success of the simplified mathematical formulation and its associated computational implementation and numerical strategies can serve as a base upon which new assumptions and better approximations can be made in future works. Those expansions, supported by the theoretical base here presented, can allow the application of the proposed problem within the context of several industries.

**Keywords:** Memory Shape Polymers; 3D and 4D Printing; MATLAB; Topology Optimization, Levenberg-Marquardt Algorithm.

## LISTA DE FIGURAS

1.1	Variação de calor em um compósito ativo. a) desenho esquemático, dimensões do modelo e seção transversal b) vista superior da forma original do objeto c-f) configurações deformadas para diferentes temperaturas. ....	11
1.2	Compósito com matriz transparente, mostrando a geometria da inclusão ativa e sua configuração deformada após programação .....	12
2.1	Aparência de uma cadeia polimérica linear em meio líquido .....	18
2.2	Pontes de enxofre (azul) conectando cadeias de borracha (preto) em uma rede .....	19
2.3	Processo de deformação de uma rede polimérica por ação mecânica .....	19
2.4	Ilustração do treinamento de um polímero com memória de forma .....	21
2.5	Estrutura polimérica ilustrando o efeito de dobras sequenciais.....	21
3.1	Representação esquemática do modelo constitutivo de Zener .....	28
3.2	Representação de uma barra de treliça e seus graus de liberdade nos sistemas locais (representados pelo índice linha) e globais.....	33
3.3	Figura ilustrativa do processo de discretização .....	40
3.4	Efeito do refinamento da malha na discretização de uma região de inclusão circular .....	41
3.5	Curvas de atualização do parâmetro de amortecimento do algoritmo de Levenberg-Marquardt .....	75
4.1	Representação esquemática do sistema estrutural analítico tratado .....	78
4.2	Gráfico da função objetivo analítica .....	80
4.3	Gráfico da função objetivo analítica em nova escala .....	81
4.4	Sistema análogo ao analítico para o modelo em treliça .....	82
4.5	Modelo de viga tratado nos exemplos ilustrativos .....	85
4.6	Unidade quadrangular da malha utilizada .....	86
4.7	Discretização e informações do exemplo base .....	88
4.8	Configuração indeformada em e em escala do exemplo base.....	88
4.9	Configuração indeformada e em camadas do exemplo base.....	89
4.10	Configuração deformada do exemplo base .....	89
4.11	Gráficos de convergência do exemplo base .....	90

4.12	Configuração deformada com novo ponto inicial .....	91
4.13	Gráficos de convergência com novo ponto inicial .....	92
4.14	Configuração indeformada com novo filtro .....	93
4.15	Configuração deformada com novo filtro .....	93
4.16	Gráficos de convergência com novo filtro .....	94
4.17	Configuração indeformada com novo critério de parada .....	95
4.18	Configuração deformada com novo critério de parada .....	95
4.19	Gráficos de convergência com novo critério de parada .....	96
4.20	Discretização e valores iniciais com nova malha .....	97
4.21	Configuração indeformada e em escala com nova malha .....	97
4.22	Configuração deformada com nova malha .....	98
4.23	Gráficos de convergência com nova malha .....	98
4.24	Configuração indeformada em nova geometria prescrita .....	100
4.25	Configuração deformada em nova geometria prescrita .....	100
4.26	Gráficos de convergência em nova geometria prescrita .....	101
4.27	Configuração deformada com nova deformação de treinamento .....	102
4.28	Gráficos de convergência com nova deformação de treinamento .....	102
4.29	Discretização e informações iniciais com nova geometria .....	103
4.30	Configuração indeformada e em escala com nova geometria .....	104
4.31	Configuração indeformada e em camadas com nova geometria .....	104
4.32	Configuração deformada com nova geometria .....	105
4.33	Gráficos de convergência com nova geometria .....	105
4.34	Configuração deformada no ponto inicial .....	107
4.35	Gráficos de convergência .....	108
4.36	Configuração indeformada encontrada no ponto de mínimo .....	108
4.37	Comparação entre áreas prescritas e encontradas no ponto final .....	109
4.38	Autovalores da matriz Hessiana no ponto final .....	109
4.39	Valores da função objetivo ao longo da direção do autovetor associado ao menor autovalor no ponto final .....	110
4.40	Comparação entre áreas prescritas e encontradas no ponto final para uma malha mais refinada .....	111
4.41	Comparação entre áreas prescritas e encontradas no ponto final para uma malha mais refinada e com mais elementos de inclusão .....	112
4.42	Autovalores da matriz Hessiana no ponto final para uma malha mais refi- nada e com mais elementos de inclusão .....	112
4.43	Gráficos de convergência para uma malha mais refinada e com mais ele- mentos de inclusão .....	113

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	<b>10</b>
<b>2</b>	<b>REVISÃO BIBLIOGRÁFICA</b>	<b>15</b>
2.1	Impressão 3D e 4D .....	15
2.2	Polímeros, Memória de Forma e Programação .....	16
2.3	Materiais Compósitos .....	22
2.4	Otimização .....	24
<b>3</b>	<b>FORMULAÇÃO MATEMÁTICA E MODELO DE DISCRETIZAÇÃO</b>	<b>27</b>
3.1	Modelo Constitutivo .....	27
3.2	Elemento de Treliça e Método da Rigidez Direta .....	32
3.3	Modelo de Discretização .....	39
3.4	Treinamento e Forma Memorizada.....	42
3.5	Formalização da Otimização .....	44
3.5.1	Problema de Otimização .....	44
3.5.2	Estratégia de Filtragem .....	46
3.5.3	Vetor Gradiente.....	47
3.5.4	Hessiana .....	55
3.6	Algoritmo de Otimização .....	67
3.6.1	Método do Gradiente .....	68
3.6.2	Método de Newton .....	68
3.6.3	Método de Gauss-Newton .....	69
3.6.4	Algoritmo de Levenberg-Marquardt .....	70
<b>4</b>	<b>RESULTADOS E DISCUSSÕES</b>	<b>77</b>
4.1	Exemplo de Verificação.....	77
4.1.1	Definição do Exemplo e Função Objetivo .....	77
4.1.2	Verificação da Formulação .....	82
4.1.3	Considerações Adicionais.....	84
4.2	Exemplos Ilustrativos.....	85
4.2.1	Exemplo Base .....	85
4.2.2	Efeito do Ponto Inicial .....	91

4.2.3	Efeito do Filtro .....	93
4.2.4	Efeito do Critério de Parada .....	94
4.2.5	Efeito da Malha.....	96
4.2.6	Efeito da Curva Prescrita .....	99
4.2.7	Efeito da Deformação de Treinamento .....	101
4.2.8	Efeito da Geometria .....	102
4.2.9	O Ponto de Mínimo .....	105
4.2.10	Considerações Adicionais.....	113
<b>5</b>	<b>CONSIDERAÇÕES FINAIS</b>	<b>115</b>
5.1	Principais Contribuições .....	115
5.2	Sugestões para Trabalhos Futuros .....	116

## **REFERÊNCIAS**

## 1 INTRODUÇÃO

*"O maior desafio para qualquer pensador é iniciar o problema de tal forma que sua solução seja possível"*

– Bertrand Russell

O avanço da tecnologia de impressão 3D tem possibilitado a aplicação de polímeros termo-ativados em precisões cada vez maiores e com elevada eficiência (WU et al., 2016; GE; QI; DUNN, 2013). Uma das consequências desse avanço é o crescente número de trabalhos de pesquisa e desenvolvimento relacionados à impressão 3D, bem como a crescente variabilidade de aplicações nas quais se faz presente. Dentro dos muitos campos que têm nascido da utilização da impressão 3D, destaca-se aqui o dos polímeros com memória de forma ou SMPs (do inglês *Shape Memory Polymers*), cuja lista de aplicações é citada por autores como Wu et al. (2016) e, extensivamente, por Meng e Li (2013), entre muitos outros.

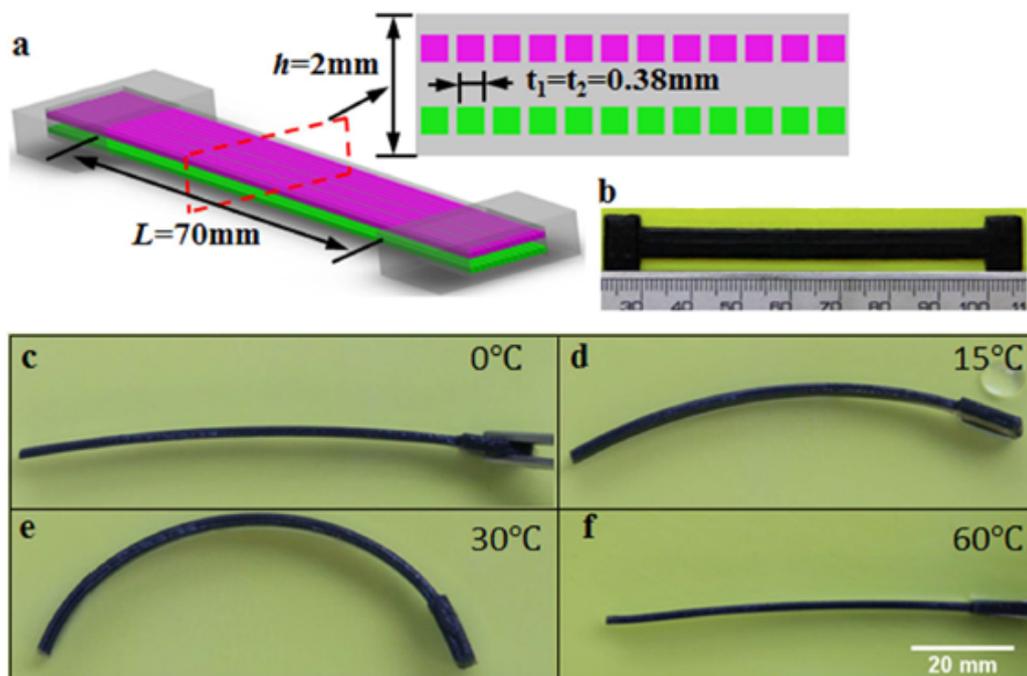
Devido à alta aplicabilidade da tecnologia, diversos estudos foram recentemente publicados e têm sido conduzidos visando entender os mecanismos que governam o comportamento dos SMPs, bem como para modelar numericamente as complexas interações que existem nesses materiais, que, de uma maneira geral, envolvem processos multifísicos não lineares e ainda não totalmente compreendidos (LIU et al., 2006; MENG; LI, 2013; MAUTE et al., 2015). Dentre esses estudos, destaca-se a procura por modelos constitutivos que consigam simular confiavelmente as relações termo-mecânicas que governam o fenômeno de memória de forma.

Entre as aplicações dos SMPs, destaca-se, neste trabalho, a dos compósitos ativos, que são uma subcategoria dos materiais compósitos na qual, dentro do contexto desta dissertação, pelo menos um dos constituintes apresenta memória de forma. Compósitos feitos dessa maneira permitem, por exemplo, a criação de superfícies curvas a partir de estruturas planas através da aplicação de estímulos externos, tendo sido os materiais previamente 'programados' para tanto. As inclusões de material com memória de forma geram estados de tensão na matriz (portanto o nome 'ativos') quando os mesmos tentam recuperar a configuração 'memorizada'. Esse estado de tensão, por sua vez, pode provocar grandes deformações (WU et al., 2016; MAUTE et al., 2015; GE; QI; DUNN, 2013).

No caso dos materiais compósitos ativos produzidos via impressão 3D, a matriz, de natureza polimérica, é incrustada com pelo menos um polímero portador de memória de forma, possivelmente por uma gama destes, podendo a própria matriz também ter memória de forma. Neste caso, as incrustações podem ser sequencialmente ativadas para recuperar a configuração 'permanente' a partir da configuração memorizada. Controlando o tipo, a fração volumétrica e a distribuição dessas incrustações, obtêm-se diferentes formas e transições temporárias e permanentes. Tal aplicação tem sido referenciada como impressão 4D (WU et al., 2016; MAUTE et al., 2015; GE; QI; DUNN, 2013).

Uma das estratégias para o uso da impressão 4D de particular interesse para este trabalho é a geração automática de configurações curvas através de estruturas impressas como planas. Wu et al. (2016) e Ge, Qi e Dunn (2013), entre outros, mostram exemplos dessa aplicação. Uma delas é apresentada na Fig. 1.1, na qual as cores rosa e verde indicam duas inclusões poliméricas com efeito de memória de forma em diferentes temperaturas, enquanto cinza refere-se à matriz polimérica

**Figura 1.1 – Variação de calor em um compósito ativo. a) desenho esquemático, dimensões do modelo e seção transversal b) vista superior da forma original do objeto c-f) configurações deformadas para diferentes temperaturas.**

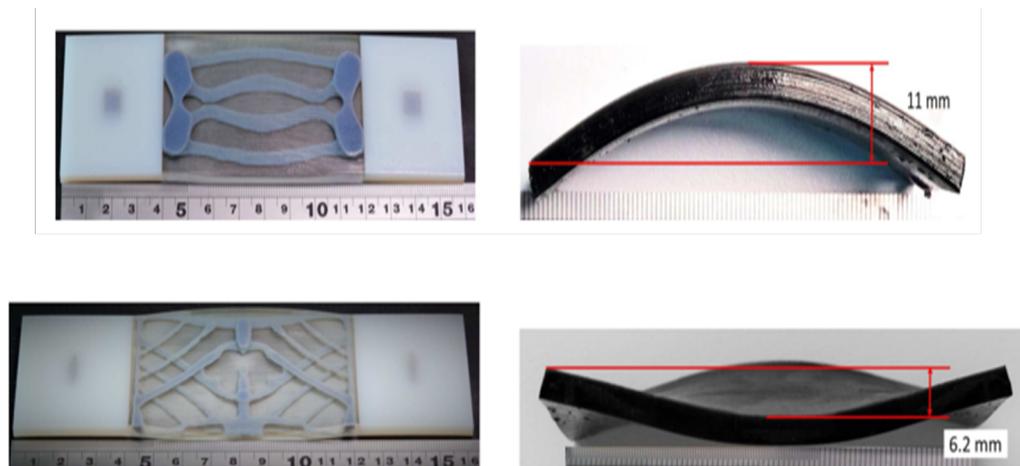


Fonte: Wu et al. (2016)

Igualmente, pode-se gerar tais configurações a partir de inclusões com geometrias irregulares, como apresentado por Maute et al. (2015) e ilustrado na Fig. 1.2.

Uma vez provada essa capacidade dos compósitos ativos de gerarem configurações espaciais curvas através da mobilização da propriedade de memória de forma de polímeros

**Figura 1.2 – Compósito com matriz transparente, mostrando a geometria da inclusão ativa e sua configuração deformada após programação**



Fonte: Maute et al. (2015)

termo-ativados, mesmo em configurações de inclusão não-triviais, bem como considerando as possíveis aplicações diretas em manufatura, uma consequência lógica é a procura por disposições ótimas dessas inclusões dentro da matriz objetivando alcançar critérios de projeto definidos (MAUTE et al., 2015), como, por exemplo, atingir, dentro de uma tolerância pré-definida, uma certa configuração prescrita. Contudo, a análise termo-mecânica acoplada desses compósitos é complexa e a definição de distribuições ótimas de inclusão caracteriza um problema inverso não trivial (MAUTE et al., 2015; MOON; RAO; CHESTER, 2016).

Dentro deste contexto, esta dissertação tem por objetivo criar um modelo de matemático e uma base computacional para a simulação destes materiais. Sob esta base computacional, formulações matemáticas e estratégias numéricas são criadas para viabilizar o projeto ótimo de peças em impressoras 3D para fins de pesquisa e desenvolvimento. Com este passo inicial, objetiva-se permitir que trabalhos futuros tenham uma fundação sólida por onde começar a expandir as formulações discutidas e tratar problemas cada vez mais complexos e realistas, até, eventualmente, chegar-se ao ponto de poder usar tais polímeros dentro do contexto das aplicações industriais apresentadas.

A dissertação é estruturada da seguinte maneira: inicia-se por uma revisão bibliográfica, objetivando embasar os desenvolvimentos seguintes. A mesma contempla os tópicos de impressão 3D e 4D, a tecnologia na qual o trabalho se baseia e que promete revolucionar a manufatura; o comportamento mecânico de polímeros, em especial os dotados do fenômeno de memória de forma, utilizado para a ativação dos compósitos estudados, bem como as maneiras usadas para treiná-los; e, por fim, tópicos selecionados acerca de compósitos e otimização, nos quais alguns detalhes importantes são pontuados.

---

No capítulo seguinte, mais extenso e central ao documento, desenvolvem-se as formulações matemáticas utilizadas para gerar os resultados discutidos, sendo, as mesmas, bem como os conceitos associados a elas, a contribuição principal do autor ao tema. As equações abordadas estendem-se em vários âmbitos. Inicia-se pela apresentação modelos constitutivos viscoelásticos que aproximam o comportamento de polímeros com memória de forma, bem como sua simplificação quasi-estática, utilizada pelo autor. Em seguida, um elemento unidimensional de treliça é adotado para aproximar tais polímeros e o Método da Rigidez Direta, usado para resolver os sistemas estruturais do trabalho, criados por malhas destas barras, é pontuado. Na sequência, explica-se como o modelo contínuo é aproximado pela malha de barras e como essa aproximação é tratada, fazendo a ponte entre o discreto e o contínuo. Logo então, explica-se como os efeitos das etapas de treinamento e liberação são tratados no modelo discreto em termos matemáticos e conceituais, bem como suas consequências em termos de equacionamento. A penúltima parte do capítulo, por sua vez, é a mais extensa e mais importante do mesmo. Nela, propõem-se um formato para a função objetivo e o problema de otimização associado, descreve uma estratégia de filtragem para a geração de topologias mais limpas, bem como suas consequências para a solução do problema, e constrói passo a passo a derivação dos vetores gradiente e matriz Hessiana da função objetivo proposta. O final do capítulo é reservado para a discussão de alguns algoritmos de otimização e os motivos dos mesmos serem ineficientes para a aplicação em questão, chegando, por fim, ao algoritmo efetivamente utilizado.

O último capítulo do núcleo da dissertação é o de resultados e discussões. Nele, inicia-se por definir um exemplo com duas variáveis de projeto, para o qual uma solução analítica é possível. Tal solução analítica é usada para que se observe o formato do gráfico do problema simplificado, bem como as implicações que o mesmo tem para a resposta do algoritmo nos problemas maiores. Na sequência, um exemplo análogo ao analítico é simulado usando-se as estratégias e equações descritas no capítulo anterior. A boa concordância entre as respostas obtidas serve de verificação para a formulação proposta, no que se segue para a seção seguinte, referente aos exemplos ilustrativos. Nela, inicia-se por definir um exemplo base, que serve de referência pra os resultados seguintes. Depois, os parâmetros do modelo construído no capítulo anterior são variados um a um, e o efeito dos mesmos nas respostas fornecidas pelo algoritmo de otimização são observadas e comentadas. O último exemplo ilustrativo, por sua vez, traz consigo ponderações a respeito de erros numéricos e sua importância na caracterização do ponto candidato encontrado. O capítulo fecha com mais considerações acerca dos resultados e do problema como um todo.

Por fim, o documento se encerra com conclusões gerais a respeito de todos os tópicos apresentados anteriormente, com a pontuação das contribuições particulares feitas por este trabalho, com sugestões para expansões futuras do mesmo e com a listagem das

citações feitas em formato de referências para consulta.

## 2 REVISÃO BIBLIOGRÁFICA

*"Pesquisar é o que eu faço quando não sei o que estou fazendo"*

– Wernher von Braun

Neste capítulo, apresenta-se a revisão bibliográfica que embasou o desenvolvimento desta dissertação, cobrindo os tópicos de impressão 3D e 4D; Polímeros e Memória de Forma; Materiais Compósitos e Otimização.

### 2.1 Impressão 3D e 4D

Desde seu surgimento, a impressão 3D tem impulsionado uma revolução na manufatura de novos materiais e estrutura. Tais elementos têm encontrado aplicações nas mais diversas áreas, por exemplo a criação de sistemas microeletromecânicos, órgãos e tecidos, partes de foguetes desenvolvidos pela Agência Espacial Norte Americana (NASA), entre muitas outras, extensivamente citadas no trabalho de Meng e Li (2013), mas também de outros, como Wu et al. (2016), Liu et al. (2006), etc. O surgimento destas aplicações está intimamente relacionado com a notável melhoria na sensibilidade das impressoras 3D modernas, que permitem a aplicação de diversos materiais em escalas cada vez menores e com maior riqueza de detalhes (WU et al., 2016; MAUTE et al., 2015; GE; QI; DUNN, 2013).

Dentre a larga gama de possibilidades, a aplicação de maior interesse no desenvolvimento desta dissertação é a chamada impressão 4D, na qual a distribuição de diferentes materiais compósitos ativos permite, após a devida programação e tratamento, a mudança entre diferentes formas temporárias da estrutura impressa (WU et al., 2016; MAUTE et al., 2015; GE; QI; DUNN, 2013) perante a prescrição de algum tipo de carregamento. Possíveis usos desses materiais incluem, por exemplo, a criação de válvulas que automaticamente se abrem ou fecham em temperaturas desejadas, bem como sensores, dispositivos biomédicos, entre muitos outros, que podem ser consultados em Meng e Li (2013), Wu et al. (2016), Liu et al. (2006), entre outros.

Mais especificamente ainda, o trabalho aqui desenvolvido se limita a tratar de compósitos poliméricos, cujo treinamento e transição entre as configurações intermediárias e

permanente se dá por meio de prescrições de natureza termo-mecânica. Nestes, inclusões poliméricas com diferentes temperaturas de transição de fase são distribuídas em uma matriz de modo a criar estruturas que se deformam em mais de uma configuração temporária para diferentes prescrições. Tal fenômeno é bem reportado por autores como Wu et al. (2016), Maute et al. (2015) e Ge, Qi e Dunn (2013), entre muitos outros. Diferentes mecanismos de transição e de treinamento, desencadeadas por radiação, prescrições eletromagnéticas, reações químicas, umidade, etc., são apresentadas em Meng e Li (2013) e Pilate et al. (2016), entre outros, mas não explicitamente tratadas neste trabalho. Vale salientar, contudo, que muito do aspecto da modelagem e simulação independe da natureza da transição e da programação, o que permitiria sua extensão para outros regimes de manufatura e treinamento com pouco esforço de modelagem e formulação.

No âmbito de equipamento, o Laboratório de Computação Científica e Visualização (LCCV/UFAL) dispõe, atualmente, de duas impressoras. A primeira delas é um exemplar comercial, adquirido em 2013, e que permite a impressão de peças usando apenas o polímero Acrilonitrila Butadieno Estireno, ou ABS, como mais comumente é conhecido, abreviação derivada do nome em inglês do mesmo (*Acrylonitrile Butadiene Styrene*). A segunda impressora, posteriormente adicionada ao núcleo de impressão 3D, foi montada dentro do próprio Laboratório e é de maior interesse para este projeto, posto que a mesma permite a impressão em uma gama maior de polímeros, incluindo alguns que apresentam a propriedade de memória de forma, dentre eles o próprio ABS e o Ácido Polilático, um polímero constituído por moléculas de ácido láctico ligadas em cadeia conhecido como PLA, sigla oriunda do inglês *Poly(Lactic Acid)*, que é biodegradável.

Mais informações técnicas não são apresentadas por questão de brevidade, uma vez que o texto desta dissertação se limita a propor e formular um problemas matematicamente, não entrando no âmbito experimental da impressão das peças desenvolvidas.

## 2.2 Polímeros, Memória de Forma e Programação

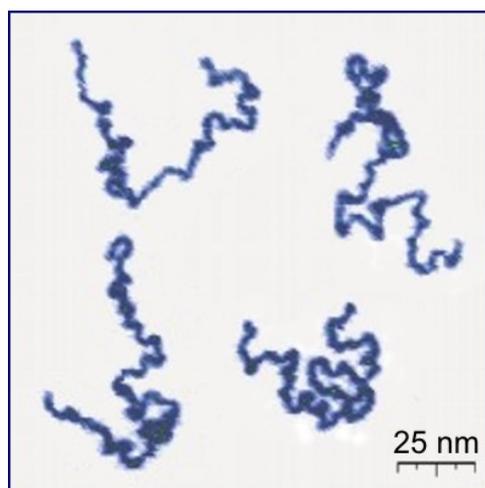
Polímeros são grandes moléculas (chamadas de macromoléculas) compostas de múltiplas subestruturas repetidamente conectadas através de ligações covalentes, estruturas essas chamadas de monômeros. Polímeros têm grande massa molecular (soma dos pesos de cada átomo constituinte das macromoléculas), bem como propriedades físicas bastante particulares associadas à relação entre essa massa elevada e menor dimensão das unidades repetidas. De maneira geral, polímeros também possuem boa resistência e podem absorver grandes deformações plásticas, têm ao menos uma transição vítrea e elasticidade com comportamento borrachoso (KUMAR; GUPTA, 2003). Dentre essas propriedades, algumas de interesse especial para a aplicação em impressão 4D são as grandes deformações elásticas absorvidas (reportadas em até 400% para alguns polímeros especiais), a facili-

dade da 'programação' da configuração temporária, o baixo peso e custo de manufatura e a biodegradabilidade de alguns polímeros (CHEN; LAGOUDAS, 2008; MOON; RAO; CHESTER, 2016), bem como sua capacidade de executar movimentos complexos entre as configurações memorizadas e a permanente (HEUCHEL et al., 2013; MOON; RAO; CHESTER, 2016), sua ductilidade e tolerância ao dano (BEHL; LENDLEIN, 2007; LI; XU, 2011), entre outras. Mais ainda, polímeros cujo efeito de memória de forma é governado por princípios termo-mecânicos podem ter suas temperaturas de transição alteradas através da modificação de sua estrutura molecular (LIU et al., 2006; BEHL; LENDLEIN, 2007), sendo essas temperaturas de transição relacionadas com o treinamento e a movimentação do polímero entre os estados permanente e memorizado. Em compensação, os mesmos são considerados por Pilate et al. (2016) como tendo resistência e rigidez insuficientes, bem como resposta mais lenta durante a recuperação de forma, para aplicações nas quais alta performance é requerida, o que pode ser agravado, em casos específicos, pelo fato de polímeros em geral terem baixa condutibilidade térmica e elétrica, além de serem quimicamente inertes. Para tais aplicações, algumas alternativas, como polímeros reforçados com nanofibras, são apresentadas pelo mesmo. Ligas metálicas, que também apresentam efeito de memória de forma, podem ser usadas em alguns desses casos, mas não são do escopo deste trabalho.

No âmbito da memória de forma, especificamente, esta pode ser entendida como a capacidade de alguns polímeros de retornar de um ou mais estados temporários 'memorizados' a um permanente pela aplicação de algum estímulo específico. Essa recuperação de forma ocorre devido à presença de transições de propriedades ou de geometria da microestrutura dos polímeros (WU et al., 2016; MOON; RAO; CHESTER, 2016; BELLIN et al., 2006; GE; QI; DUNN, 2013; CHEN; LAGOUDAS, 2008; LIU et al., 2006). De uma maneira geral, o comportamento de memória de forma dos SMPs está associado à presença de uma cadeia polimérica estável somada a algum mecanismo de transição. A estabilidade pode ter diversas origens, como, por exemplo, a presença de uma estrutura cristalina, de ligações químicas entre as cadeias, a interpenetração física das mesmas ou ainda um simples estado de emaranhamento. O mecanismo de transição, por sua vez, é o responsável pelo 'travamento' da configuração temporária e pode, também, ter diversas origens, como transições de derretimento/solidificação, vitrificação, mudanças entre fases líquidas isotrópicas e anisotrópicas, formação reversível de ligações cruzadas, etc. (MENG; LI, 2013). Neste trabalho, foca-se na transição vítrea, que corresponde à mudança do comportamento de um polímero de borrachoso para vítreo e, mais ainda, considera-se que tal mudança é desencadeada (e o treinamento é realizado) a partir de prescrições de natureza termo-mecânica. Outras naturezas de estímulos associados a SMPs podem ser consultados em Meng e Li (2013) e Pilate et al. (2016) e incluem radiação, prescrições eletromagnéticas, reações químicas, umidade, etc.

A transição vítrea é uma propriedade dos polímeros de mudar, de maneira drástica e reversível, seu comportamento termo-mecânico de um de natureza borrachosa para outro de natureza vítrea. Tal mudança acontece pela presença de cadeias de carbonos conectados covalentemente entre si através de ligações simples do tipo ' $C - C$ ', que são bastante móveis. Devido a essa mobilidade, essas cadeias ficam 'enroladas' em seu estado natural (Fig. 2.1). A estabilidade, nesse caso, é conferida pela presença de ligações cruzadas entre as cadeias, formando uma rede, seja ela de natureza física (como é o caso dos termo-plásticos) ou química (como é o caso da borracha vulcanizada, onde a ponte é feita por moléculas de enxofre, como mostrado na Fig. 2.2). Quando essa rede é mecanicamente mobilizada, as cadeias passam a ter uma orientação preferencial na direção do carregamento, o que efetivamente reduz sua entropia, que pode ser entendida como o nível de desordem do sistema. Uma vez que o carregamento é liberado, o sistema tende a retornar a uma maior entropia, segundo a termo-dinâmica, o que faz com que a rede polimérica abandone sua direção preferencial (e portanto sua deformação), como ilustrado na Fig.2.3. Esse comportamento é estudado pela teoria da elasticidade borrachosa, que dispõe de formulações para regimes de grandes deformações que envolvem parâmetros como o tamanho médio das cadeias poliméricas, a temperatura, etc. (LIU et al., 2006; WU et al., 2016; BELLIN et al., 2006; HEUCHEL et al., 2013). Para este trabalho, considera-se inicialmente um comportamento viscoelástico linear, que é suficiente para o regime de pequenas deformações e baixas variações de temperatura considerado a princípio.

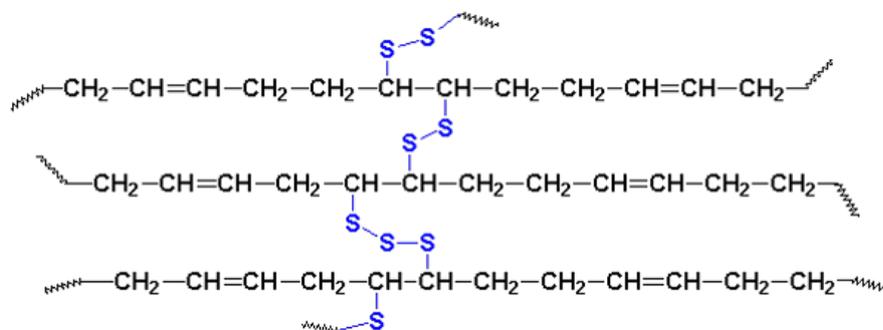
**Figura 2.1 – Aparência de uma cadeia polimérica linear em meio líquido**



**Fonte: Roiter e Minko (2005)**

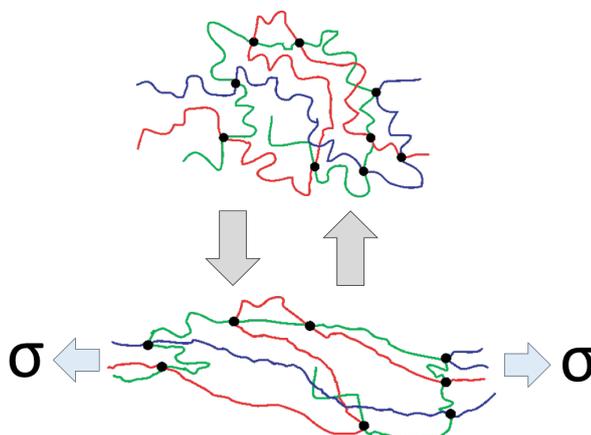
A transição vítrea fica então como o mecanismo reversível de travamento e ocorre quando uma temperatura muito baixa atua sobre a rede polimérica. Com essa redução da temperatura, as cadeias começam a formar estruturas cristalinas por meio de ligações físicas, efetivamente 'congelando' sua movimentação, sendo este o mecanismo que armazena

Figura 2.2 – Pontes de enxofre (azul) conectando cadeias de borracha (preto) em uma rede



Fonte: Dissemination of IT for the Promotion of Materials Science (2004)

Figura 2.3 – Processo de deformação de uma rede polimérica por ação mecânica



Fonte: Polymer Science Learning Center (2002)

a configuração temporária programada. Dessa forma, ao invés de responder por variação de entropia, uma vez que tal movimentação fica impedida, esforços mecânicos passam a mobilizar as ligações covalentes em si, o que acarreta em um significativo ganho de rigidez, que pode ser de algumas ordens de magnitude. Nesta caso, a resposta em termos de tensão e deformação do sistema é governada pela energia interna dessas ligações e, para pequenas deformações, pode ser considerada elástica linear (TOBUSHI et al., 2001; MOON; RAO; CHESTER, 2016; LIU et al., 2006; DAINI; LIU; GALL, 2006).

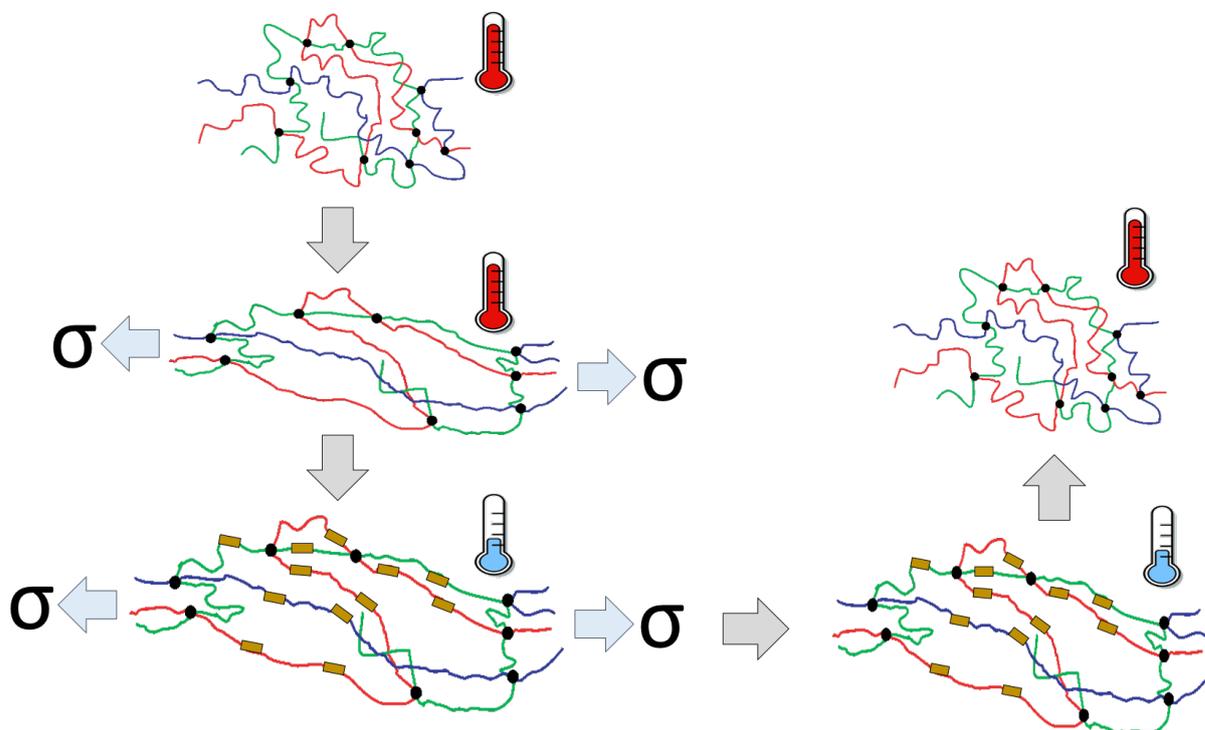
Essa configuração, com a memória de forma sendo obtida através de uma transição vítrea e um polímero borrachoso, é o tipo mais comumente empregado, sendo o caso, por exemplo, dos materiais utilizados pelas impressoras 3D do LCCV/UFAL. Outro tipo de transição comum é a de derretimento, quando a rede polimérica perde a estabilidade devido ao derretimento das ligações cruzadas ou das próprias cadeias em si. Os materiais também podem ter diferentes temperaturas de transição vítrea, com diferentes partes da

rede cristalizando em diferentes temperaturas. Um polímero pode, portanto, ter duas ou mais transições e armazenar uma ou mais configurações temporárias. Tais polímeros são ditos como tendo múltipla memória de forma (BELLIN et al., 2006; MOON; RAO; CHESTER, 2016).

Como dito anteriormente, além de necessitar de uma cadeia polimérica estável somada a algum mecanismo de transição, o fenômeno de memória de forma depende da 'programação' do material, que pode ser feita de diferentes maneiras, a exemplo de prescrições termo-mecânicas, como adotado neste trabalho. Dentro desse âmbito, uma maneira de programar um SMP pode ser observada na Fig. 2.4. Na primeira etapa da mesma, indicada no canto superior direito, o material apresenta estrutura molecular amorfa, com distribuição randômica das ligações e sendo cada uma das três cadeias indicadas representadas por uma cor (vermelho, azul e verde). Neste caso, os pontos pretos representam as ligações cruzadas entre as cadeias, o que confere estabilidade às mesmas, criando uma rede. Na segunda etapa, abaixo, o material é tracionado (como indicado pela presença da tensão  $\sigma$ ). Na sequência, a temperatura é reduzida, como indicado pelo termômetro no canto superior direito da figura, o que produz o efeito de cristalização discutido, no qual as cadeias interagem fisicamente uma com a outra, conectando-se. A cristalização é indicada pela presença de retângulos dourados e, para grandes deformações, tem direção preferencial igual à do carregamento, uma vez que assim também se alinham as cadeias. Na etapa seguinte, à direita, a tensão aplicada é liberada. Apesar disso, contudo, a cristalização 'armazena' deformação e oferece resistência à tendência natural das cadeias de voltar ao estado desordenado de menor entropia. Portanto, nesta etapa, pode-se perceber que uma pequena parcela da deformação aplicada é recuperada, mas a maior parte não. Essa é a configuração 'memorizada' pelo polímero. Por fim, na etapa seguinte, acima, a temperatura é novamente elevada, os cristais formados se desmancham e as cadeias recuperam sua estrutura randômica, com o material voltando à configuração indeformada original, denominada de permanente.

Outras formas de programar o material existem. Li e Xu (2011), por exemplo, discutem a programação por compressão à frio (com temperatura abaixo da de transição vítrea), na qual se tira partido do comportamento viscoelástico do polímero no estado vítreo. Mao et al. (2015), por sua vez, usam uma rotina de programação em que estruturas tridimensionais nas quais os SMPs formam as 'dobras' são deformadas de modo a tornarem-se planas. Em seguida, uma redução da temperatura fixa as estruturas na configuração planificada. Um aquecimento posterior faz com que a estrutura se dobre novamente. Fazendo com que as dobras tenham temperaturas de transição vítrea prescritas (lembrando que essa é uma propriedade que pode ser controlada pela microestrutura dos polímeros), bem como prescrevendo-se a temperatura do meio de imersão, estruturas sequencialmente autodobráveis podem ser criadas. A Fig. 2.5 mostra a evolução no tempo

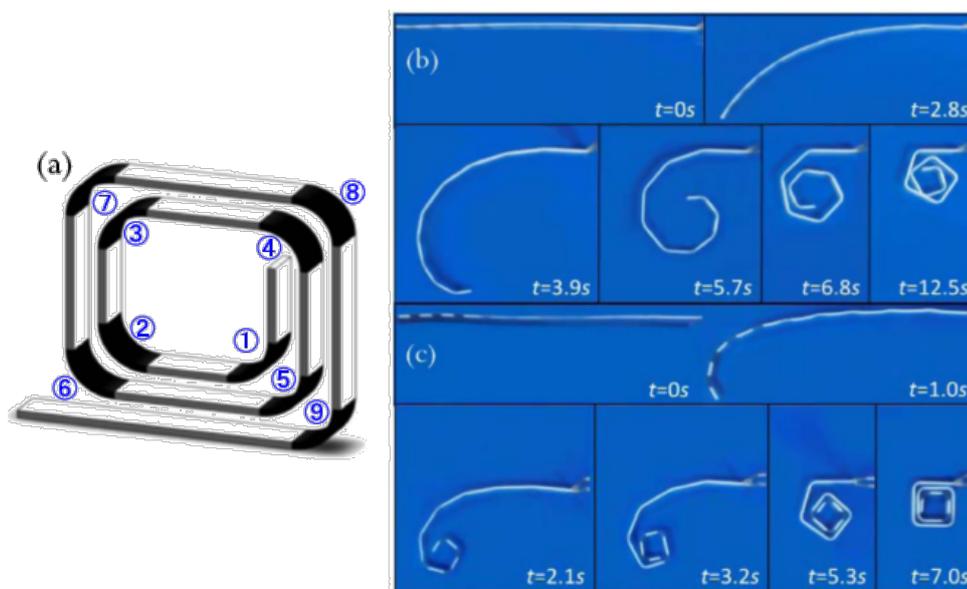
Figura 2.4 – Ilustração do treinamento de um polímero com memória de forma



Fonte: Adaptada de Polymer Science Learning Center (2002)

de duas dessas estruturas, conforme apresentadas por Mao et al. (2015).

Figura 2.5 – Estrutura polimérica ilustrando o efeito de dobras sequenciais



Fonte: Mao et al. (2015)

É importante notar que a orientação preferencial dos cristais formados é a mesma da deformação aplicada. Por conta disso, os polímeros passam a responder de forma ani-

sotrópica após a cristalização, o que dificulta sua modelagem (MOON; RAO; CHESTER, 2016). Tal consideração não será feita neste trabalho, que assume que as deformações aplicadas são suficientemente pequenas para que a cristalização aconteça com as cadeias em direções majoritariamente randômicas, tendo a direção preferencial dessa distribuição pouco efeito na resposta multidirecional do polímero, o que pode não ser verdade para todas as aplicações.

### 2.3 Materiais Compósitos

Compósitos, classicamente, podem ser definidos como materiais estruturais compostos por dois ou mais constituintes que são combinados em nível macroscópico e não são solúveis um no outro. Um conjunto de constituintes é normalmente conhecido como fase reforçante (ou 'inclusão'), que, por sua vez, está inserido em um meio normalmente contínuo, conhecido como matriz. A fase reforçante pode ser composta de fibras, partículas ou flocos. De uma maneira geral, usam-se materiais cujas propriedades físicas sejam significativamente diferentes, de modo que o resultado final se comporte diferentemente dos constituintes em caráter individual. Dessa forma, compósitos são normalmente utilizados como alternativas que apresentem melhores propriedades mecânicas, como rigidez, ductilidade, resistência, peso, etc. em comparação com os materiais base isolados. Um exemplo antigo do uso de compósitos é a adição de fibras de palha para reforçar tijolos de argila. Neste caso, as fibras servem para 'costurar' fissuras na matriz, que por sua vez se trata de um material frágil, especialmente em tração, efetivamente aumentando sua resistência a esse tipo de esforço (e, conseqüentemente, a esforços de flexão e cisalhamento), bem como sua ductilidade (KAW, 2006).

Especificamente no contexto deste trabalho, trata-se de compósitos ativos, que são aqueles em que a presença das inclusões impõe um campo de tensões (e, conseqüentemente, de deformações) na matriz em que se inserem. Um exemplo disso é o caso de vigas de concreto protendido, nas quais a armadura ativa pré-tracionada impõe um esforço de compressão e fletor na matriz, normalmente para combater a flexão oposta e seu esforço de tração associado. Esse campo de tensões é gerado pela aplicação de um estímulo térmico de aquecimento, que provoca a transição da matriz ou de pelo menos um dos polímeros usados nas inclusões entre as fases anteriormente descritas. Essa transição de fase compele os polímeros a recuperar a configuração permanente, enquanto aqueles que continuam 'congelados' mantêm a configuração temporária sem nenhum campo de tensão interno associado. A diferença entre o comportamento diferencial dos materiais cristalizados e não-cristalizados gera um desbalanço interno no material compósito, que por sua vez se traduz em deformações possivelmente complexas, que podem ser controladas pela fração e geometria das inclusões utilizadas, a geometria da matriz, e pelos materiais utilizados nas mesmas, inclusive com propriedades afetadas pela microestrutura. Mais ainda, o

regime de treinamento e de estímulo aplicados, entre outras variáveis, afeta largamente o comportamento de memória de forma desses compósitos (MAUTE et al., 2015; WU et al., 2016).

Seja pela ação da matriz ou de uma ou mais das inclusões tentando recuperar a configuração permanente, o importante é que as tensões envolvidas permitem a mudança de forma da matriz polimérica de maneira controlável, por exemplo permitindo a criação de estruturas curvas a partir de chapas planas (WU et al., 2016; MAUTE et al., 2015). Quando ambos (matriz e inclusões) fazem a transição para o estado borrachoso, a configuração original (neste caso a chapa plana) é recuperada. O inverso também é válido, podendo a configuração plana ser a temporária e a curva a permanente, como em Mao et al. (2015). As possibilidades são muitas neste aspecto.

Um tipo particularmente útil desses compósitos são os chamados de 'digitais'. Neles, a impressão de diferentes polímeros em posições precisas, sendo esses ainda misturados e curados em diferentes velocidades, permite a criação de materiais cujas propriedades podem ser facilmente modificadas para atender critérios de projeto. Mais ainda, explorando-se a variação espacial da distribuição desses polímeros, tem-se maior controle sobre a forma assumida pelos mesmos após a ativação térmica (WU et al., 2016; MAUTE et al., 2015; GE; QI; DUNN, 2013).

Uma consequência natural dessa flexibilidade na distribuição é a procura por formas ótimas de estruturar os compósitos impressos visando atingir configurações e propriedades desejadas da melhor maneira possível dado um conjunto de restrições. Tal procedimento, na qual se busca a configuração inicial ótima para a um comportamento final prescrito, trata-se de um problema inverso complexo que precisa ser estudado (MAUTE et al., 2015), especialmente porque o comportamento mecânico de compósitos ativos com memória de forma é um problema termo-mecânico não-linear e acoplado, em que grandes deformações e anisotropias atuam (MAUTE et al., 2015; WU et al., 2016; GE; QI; DUNN, 2013). Neste trabalho, em particular, empregam-se técnicas de otimização, especialmente de otimização topológica, para tal fim, semelhantemente ao que foi feito por Maute et al. (2015), que buscou uma distribuição de material com memória de forma que o permitisse atingir as configurações deformadas desejadas.

Mais ainda, o trabalho é puramente matemático e formulacional, buscando criar fundações sob as quais análises mais complexas e aplicações industriais possam ser desenvolvidas. Desta forma, o mesmo tratará, apenas, de problemas bidimensionais e elásticos lineares, onde se considera um regime de pequenos deslocamentos e deformações. Apesar disso, como ficará claro mais adiante, o problema é complexo, mesmo em sua forma mais pura, o que limitou seu desenvolvimento além do caráter exploratório deste documento.

## 2.4 Otimização

Segundo Christensen e Klarbring (2008), o termo 'otimização' pode ser entendido como o ato de prover o melhor produto possível dentro do conceito do projeto e respeitando as restrições de funcionalidade impostas. Costumeiramente, essa otimização era feita com base no conhecimento prévio e no método de tentativa e erro, ou seja, de maneira manual, ainda que a simulação de sistemas mecânicos na indústria moderna seja feita majoritariamente via métodos computacionais, tal como o método dos elementos finitos. A otimização no contexto matemático, por outro lado, funciona de maneira bastante diferente. Nele, o problema de otimização é definido formalmente, no qual se pontua de maneira explícita o que significa um 'melhor produto', por exemplo em termos de uma função matemática sobre a qual se utiliza um algoritmo de otimização, ficando as restrições de funcionalidade e de conceito definidas como limites nas variáveis de projeto que alimentam a função a ser otimizada, bem como outras funções de estado que devem respeitar limites impostos (funções de restrição).

Para a definição mais precisa do problema matemático de otimização, Christensen e Klarbring (2008) pontuam as seguintes nomenclaturas, que serão usadas por este autor de agora em diante:

- Função objetivo ( $f(\vec{x})$ ): É a função usada para 'classificar' os diferentes projetos de acordo com o quão bons eles são e costumeiramente retorna valores menores quanto melhores forem os projetos.
- Variáveis de projeto ( $\vec{x}$ ): É o vetor que engloba os parâmetros de projeto que podem ser ajustados durante o processo de otimização e que normalmente têm limites superiores e/ou inferiores definidos.
- Variáveis de estado ( $g_i(\vec{x})$ ): Para os valores dos parâmetros dados em  $\vec{x}$ ,  $g_i(\vec{x})$  é uma função que representa respostas específicas do projeto em certos âmbitos, respostas essas que, de uma maneira geral, também precisam atender critérios definidos.

Dessa forma, o problema de otimização fica matematicamente definido como

$$\begin{aligned}
 & \underset{\vec{x}}{\text{Min}} && f(\vec{x}) \\
 \text{s.t} & && g_{1i}(\vec{x}) - \bar{g}_{1i} = 0 \\
 & && g_{2i}(\vec{x}) - \bar{g}_{2i} \leq 0 \\
 & && \vec{x}_{inf} \leq \vec{x} \leq \vec{x}_{sup}
 \end{aligned} \tag{2.1}$$

no qual 's.t' funciona como uma abreviatura de 'sujeito às restrições', sendo as mesmas

fornechas pelas igualdades e desigualdades que seguem. Nesse âmbito,  $g_{2i}(\vec{x}) - \bar{g}_{2i} \leq 0$  representam restrições de desigualdade, nas quais alguns parâmetros do projeto (por exemplo, para o caso estrutural, considerado nesta dissertação, seu peso, rigidez, volume, etc.) precisam ser limitados, no caso superiormente. Se o parâmetro real precisa de um limite inferior, e não superior, pode-se usar o negativo do mesmo na restrição, juntamente com o negativo de  $\bar{g}_{2i}$ , o valor prescrito que não pode ser violado, correspondente. Já  $\vec{x}_{inf} \leq \vec{x} \leq \vec{x}_{sup}$  define uma restrição chamada de 'lateral', na qual as variáveis de projeto precisam ser limitadas a um intervalo (portanto lateralmente restringidas). Em alguns casos específicos, esses limites inferiores e superiores podem ser  $-\infty$  ou  $+\infty$ , respectivamente. Mais ainda, considera-se a comparação entre vetores apresentada como uma comparação entre cada entrada de um vetor com a entrada correspondente nos vetores limitantes, e não uma comparação entre as normas dos vetores. Por fim,  $g_{1i}(\vec{x}) - \bar{g}_{1i} = 0$  são as chamadas 'restrições de igualdade'. Uma restrição de igualdade comumente presente nos problemas estruturais é a restrição de igualdade de equilíbrio, definida como

$$[K_g(\vec{x})]\{u\} - \{F_g(\vec{x})\} = 0 \quad (2.2)$$

na qual  $[K_g(\vec{x})]$  é a matriz global de rigidez da estrutura, de uma maneira geral dependente de  $\vec{x}$ ,  $\{u\}$  seu vetor de deslocamentos globais e  $\{F_g(\vec{x})\}$  o vetor de forças globais, também, de uma maneira geral, dependente de  $\vec{x}$ . Vale salientar que o índice ' $i$ ' presente tanto nas restrições de igualdade quanto desigualdade significa que essas restrições podem ser múltiplas. Mais especificamente, podem existir tantas restrições de igualdade quanto houverem variáveis de projeto e tantas restrições de desigualdade quanto necessárias, contanto que as mesmas definam uma região viável dentro do domínio das variáveis de projeto.

Mais ainda, Christensen e Klarbring (2008) citam três tipos de problemas de otimização no contexto estrutural, bem como exemplos dos mesmos. São eles:

- Otimização de tamanho: no qual se otimiza as dimensões de uma estrutura, ou partes dela, por exemplo as áreas da seção transversal de barras de uma treliça.
- Otimização de forma: no qual se pode modificar a geometria do contorno do problema, contudo sem modificar sua conectividade, por exemplo quando o contorno é definido por algum tipo de equação e são otimizados os parâmetros que definem essa equação. Nesse tipo, novos contornos não podem ser criados.
- Otimização topológica: o tipo mais geral de otimização estrutural. No caso discreto, o mesmo permite, por exemplo, que barras de uma treliça tenham não só suas áreas modificadas, como inclusive feitas nulas, de modo que as mesmas 'somem' da estrutura, efetivamente modificando a conectividade. No caso contínuo, similarmente,

porções de uma chapa podem ter sua espessura reduzida até um valor nulo, portanto criando 'furos' que atuam como novos contornos no problema.

Christensen e Klarbring (2008) pontuam que, por estas definições, a otimização topológica e de tamanho são intimamente correlacionadas em termos conceituais, sendo a segunda um caso particular da primeira, apenas com a restrição adicional de que os valores nulos que eliminam porções da estrutura não são permitidos. Já a otimização de forma e topológica são, conceitualmente, radicalmente diferentes, no sentido de que a segunda permite a criação de novos contornos e modificação de conectividades, enquanto a primeira não o permite.

Neste trabalho, o tipo de otimização utilizada é topológica, uma vez que, como será discutido no capítulo seguinte, barras podem ser retiradas da topologia da estrutura, modificando os contornos da região de inclusão.

### 3 FORMULAÇÃO MATEMÁTICA E MODELO DE DISCRETIZAÇÃO

*"E somente sobre essa base agora firme e granítica de ignorância é que até aqui se pôde edificar a ciência; a vontade de saber sobre a base de uma vontade muito mais forte, a vontade de ignorar, de incerteza, de inverdade, não como seu oposto, mas como seu melhoramento!"*

– Além do Bem e do Mal - Friedrich Nietzsche

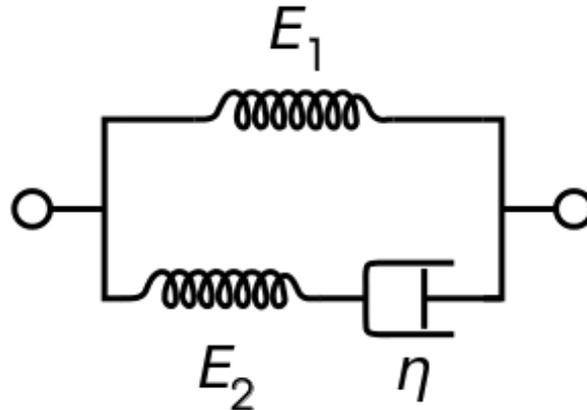
Neste capítulo, apresentam-se as formulações matemática e modelos de discretização que definem o problema tratado, bem como as hipóteses simplificadoras associadas aos mesmos. Apresentam-se um modelo constitutivo quasi-estático, a construção de um elemento discreto simplificado de barra de treliça, a solução do sistema estrutural via o Método da Rigidez Direta, a formulação usada para simular o método de treinamento do compósito, o modelo de otimização, a formulação do gradiente da função objetivo, sua Hessiana e, por fim, o algoritmo escolhido para a determinação da geometria de inclusão.

#### 3.1 Modelo Constitutivo

Para a representação do efeito de memória de forma em polímeros, diversos modelos constitutivos foram propostos e/ou discutidos por diferentes autores. Daini, Liu e Gall (2006) e Heuchel et al. (2013), por exemplo, empregam um modelo constitutivo viscoelástico baseado no de sólido-padrão, também conhecido como modelo de Zener. O modelo de Zener, em seu formato original, consiste em ligar um elemento de Maxwell em paralelo com uma mola, conforme indicado na Fig. 3.1.

Dentro deste modelo, para a consideração do efeito de memória de forma, em conformidade com a teoria discutida anteriormente, quando a temperatura e a mobilidade das cadeias são altas e a resposta do sistema é entrópica ( $T \gg T_G$ , sendo  $T_G$  a temperatura de transição de vitrificação), o amortecedor do modelo de Maxwell é considerado como tendo viscosidade bastante reduzida, virtualmente nula. Dessa forma, para carregamentos graduais (que não tenham caráter de impacto), a deformação no elemento de Maxwell é absorvida pelo amortecedor, pouco mobilizando a mola, e a rigidez do sistema passa a ser predominantemente dada pelo trecho composto pela mola em paralelo, chamado de trecho

**Figura 3.1 – Representação esquemática do modelo constitutivo de Zener**



Fonte: Pekaje - English Wikipedia

de equilíbrio. Fica claro, portanto, que o valor da rigidez dessa mola corresponde à resposta entrópica do modelo. Por sua vez, para temperaturas significativamente abaixo da de transição vítrea ( $T \ll T_G$ ), considera-se que as cadeias estão cristalizadas, o que se traduz, em termos do modelo, em um aumento da viscosidade de várias ordens de magnitude. Portanto, para as escalas de tempo tratadas, é como se o amortecedor fosse praticamente imóvel (tem viscosidade virtualmente infinita), o que faz com que todo o carregamento aplicado mobilize as duas molas com praticamente a mesma deformação. Nesse arranjo, a rigidez das ligações fica englobada na mola do trecho de não equilíbrio correspondente ao modelo de Maxwell. As diferenças entre os dois modelos consistem principalmente no tratamento do trecho de equilíbrio, que Daini, Liu e Gall (2006) consideram como não-linear, cujo valor é encontrado tendo como base a medida de entropia das cadeias. Heuchel et al. (2013), por outro lado, tratam a mola como elástica linear, limitando seu modelo a regimes de pequenas deformações. Modelos ainda mais complexos foram desenvolvidos por outros autores, como é o caso de Li e Xu (2011), que incorporam, além dos efeitos viscoelásticos, considerações viscoplásticas em seu modelo constitutivo, usado para simular o efeito de programação em polímeros com memória de forma por compressão à frio, em oposição ao método mais comumente empregado de tracionar o material em estado borrachoso. Nesses e em outros casos, contudo, o coeficiente de amortecimento do trecho de Maxwell é considerado função diretamente da temperatura, portanto  $\eta = \eta(T)$ , ou mesmo de uma variável de cristalização, que por sua vez é dependente da temperatura, portanto  $\eta = \eta(\phi) = \eta(\phi(T))$ .

Outros autores, por sua vez, discutem modelos constitutivos puramente termo-elásticos, sem efeitos viscosos ou considerações temporais. É o caso de Liu et al. (2006), que tratam um modelo termo-elástico-linear, e Chen e Lagoudas (2008), que consideram um modelo generalizado, podendo incluir não-linearidades. Para essa metodologia de tra-

tamento dos polímeros, a variável de cristalização  $\phi$ , também dependente da temperatura, é usada para definir as frações volumétricas de material em estado borrachoso e vítreo e o comportamento do polímero como um todo é aproximado de maneira quasi-estática como a da mistura entre as fases, analogamente a um material compósito. Diferentes formatos para a relação entre a cristalização e a temperatura foram propostos por diferentes autores. Uma revisão bibliográfica de algumas dessas equações pode ser consultada em Kazakevi e Steeb (2012), no qual os autores afirmam ainda que muitas dessas equações possuem parâmetros cujos significados físicos ainda não foram diretamente definidos, constituindo apenas parâmetros de ajuste de modelo. Os mesmos também comentam sobre a ausência de uma equação definitiva que aproxime efeitos de cristalização em diferentes polímeros, que podem ter curvas significativamente diferentes uns dos outros, bem como diferentes influências das pré-deformações aplicadas no efeito de cristalização, o que complica ainda mais a modelagem. Neste trabalho, levando-se em consideração as diversas metodologias propostas, adota-se o formato simplificado, condizente com alguns dos apresentados em Kazakevi e Steeb (2012) e com uma função matemática chamada de 'curva logística', de

$$\phi = \phi(T) = \frac{1}{1 + e^{C_\phi(T-T_{ref})}} \quad (3.1)$$

na qual  $T_{ref}$  é uma temperatura de referência para uma cristalização de 50% e  $C_\phi$  é o parâmetro de ajuste do modelo, que pode variar drasticamente a depender de informações como pré-deformação, taxa de aplicação do incremento de temperatura, tipo de polímero, propriedades das cadeias, etc., sendo  $e$  o número de Euler.

Com a variável de cristalização, para um volume de controle  $V$ , definem-se as frações volumétricas como

$$V_g = \phi V; \quad V_r = (1 - \phi)V; \quad V_g + V_r = V \quad (3.2)$$

na qual  $g$  refere-se à porção em estado vítreo, sendo  $r$  a porção em estado borrachoso.

Mais ainda, Liu et al. (2006) e Chen e Lagoudas (2008) consideram que as tensões na fase vítrea e borrachosa podem ser tomadas iguais. Dessa forma, tem-se que

$$[\sigma] = [\sigma_g] = [\sigma_r] \quad (3.3)$$

na qual  $[\sigma]$  é o tensor de tensão atuante no volume de referência. Neste trabalho, considera-se apenas um modelo unidimensional, de maneira que os tensores tornam-se grandezas escalares relativas à tensão normal na direção de referência, ou seja

$$\sigma = \sigma_g = \sigma_r \quad (3.4)$$

Por consequência dessa consideração à respeito das tensões, tem-se que o escalar da deformação linear na direção e no volume de referência pode ser escrito como

$$\epsilon = \phi \epsilon_g + (1 - \phi) \epsilon_r \quad (3.5)$$

Para um regime elástico linear, a deformação pode ser explicitada em função de suas parcelas, aqui limitadas a

$$\epsilon = \epsilon^E + \epsilon^T + \epsilon^i \quad (3.6)$$

na qual o índice  $T$  representa uma deformação de natureza térmica e  $E$  de natureza elástica, sendo  $i$  a deformação inelástica.

Decompondo as parcelas individualmente, a deformação na fase borrachosa, para uma aproximação elástica-linear, pode ser desmembrada simplificada como

$$\epsilon_r = \epsilon_r^E + \epsilon_r^T \quad (3.7)$$

Já a deformação na fase vítrea, também elástica-linear, pode ser decomposta como

$$\epsilon_g = \epsilon_g^E + \epsilon_g^T + \epsilon_g^\phi \quad (3.8)$$

na qual o índice  $\phi$  refere-se à parcela de deformação 'congelada' durante a cristalização, que age como deformação inelástica.

Durante a cristalização, a deformação elástica na fase borrachosa é convertida em deformação congelada na fase vítrea. Portanto, pode-se escrever

$$\delta \epsilon_g^\phi = \delta \phi \epsilon_r^E \quad (3.9)$$

sendo  $\delta \phi$  a variação de cristalização. Analogamente, para o caso do descongelamento, quando a variação de cristalização é negativa, a deformação armazenada na fase vítrea é convertida em deformação elástica na fase borrachosa novamente. A mesma, portanto, é reduzida segundo a expressão.

$$\delta\epsilon_g^\phi = \delta\phi\epsilon_g^\phi \quad (3.10)$$

Para ambos os casos, a relação entre a variação de deformação elástica na fase borrachosa e congelada na fase vítrea pode ser expressa por

$$\delta\epsilon_g^\phi = -\delta\epsilon_r^E \quad (3.11)$$

Já a deformação térmica pode ser aproximada linearmente como

$$\epsilon^T = \alpha\Delta T \quad (3.12)$$

na qual  $\alpha$  é o coeficiente de dilatação térmica linearizado e  $\Delta T$  a variação de temperatura com relação à configuração indeformada de referência. Tal coeficiente é dito linearizado pois, num caso real, o mesmo depende da temperatura do polímero e do seu estado de deformação e cristalização de maneiras complexas.

A relação constitutiva, escrita em termos da parcela elástica da deformação, fica apresentada como

$$\sigma = E_g\epsilon_g^E = E_r\epsilon_r^E \quad (3.13)$$

nas quais  $E$  é o módulo de elasticidade linear do volume de referência. Essas equações podem ser reorganizadas como

$$\epsilon_g^E = \frac{\sigma}{E_g}; \quad \epsilon_r^E = \frac{\sigma}{E_r} \quad (3.14)$$

Multiplicando-se os dois termos da equação à esquerda por  $\phi$  e os da direita por  $1 - \phi$ , tem-se que

$$\phi\epsilon_g^E = \phi\frac{\sigma}{E_g}; \quad (1 - \phi)\epsilon_r^E = (1 - \phi)\frac{\sigma}{E_r} \quad (3.15)$$

a qual, substituídos os dois termos à esquerda presentes na Eq. 3.5 pelos equivalentes à direita, porém se considerando apenas as parcelas elásticas, e não a deformação total, resulta em

$$\epsilon^E = \phi\frac{\sigma}{E_g} + (1 - \phi)\frac{\sigma}{E_r} \quad (3.16)$$

a qual, por sua vez, pode ser reorganizada como

$$\frac{\epsilon^E}{\sigma} = \frac{\phi}{E_g} + \frac{1-\phi}{E_r} \quad (3.17)$$

Por fim, elevando-se ambos os lados da igualdade por  $-1$ , bem como fazendo  $\sigma/\epsilon^E = E$ , tem-se

$$E = \frac{1}{\frac{\phi}{E_g} + \frac{1-\phi}{E_r}} \quad (3.18)$$

a qual representa o limite inferior do módulo de elasticidade equivalente do volume de controle em função da variável de cristalização (também conhecido como Modelo de Reuss). Tem-se, portanto, que

$$E = E(\phi) = E(\phi(T)) \quad (3.19)$$

Na sequência, essas relações constitutivas são usadas para desenvolver um elemento de treliça. O elemento é então usado, através do Método da Rigidez Direta, para simular o efeito de memória de forma para compósitos poliméricos com inclusões em geometrias complexas através de uma aproximação discreta, baseada no método de '*Ground Structure*', uma técnica para a definição de geometrias ótimas na qual o problema a ser tratado é aproximado por um amplo conjunto de elementos conectados (conjunto este que recebe a nomenclatura de *Ground Structure*).

### 3.2 Elemento de Treliça e Método da Rigidez Direta

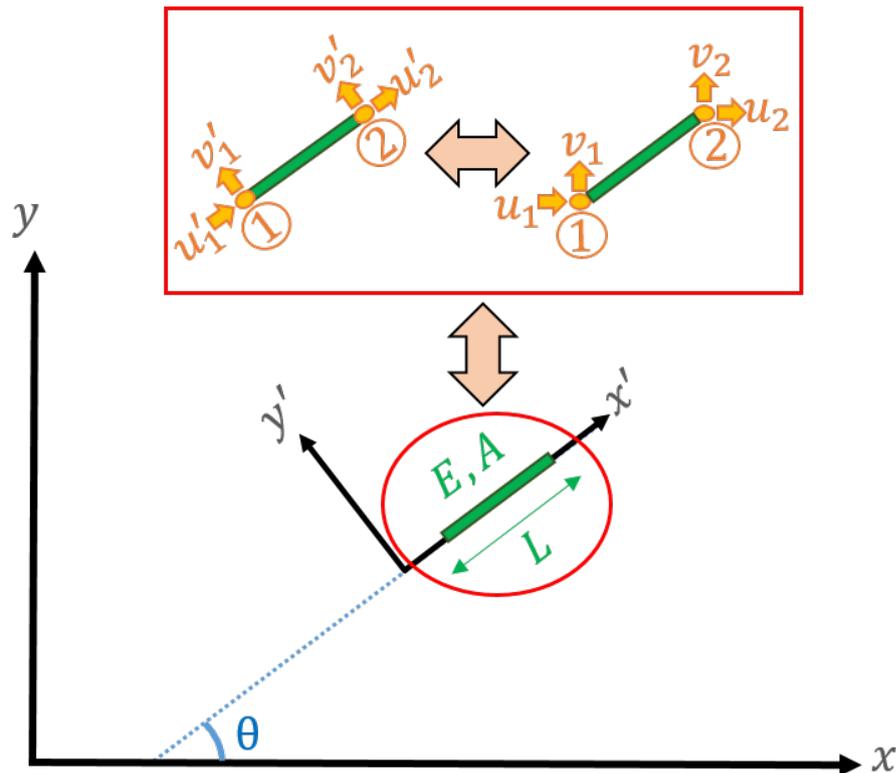
Para um elemento tradicional de treliça plana de material elástico linear e em regime de pequenos deslocamentos e pequenas deformações, a matriz de rigidez local, construída tendo como base o sistema de coordenadas locais apresentado em Fig. 3.2 pelo par de eixos  $x'$  e  $y'$ , pode ser matematicamente expressa, para o Método da Rigidez Direta, como

$$[k_l] = \frac{EA}{L} \begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (3.20)$$

na qual  $E$  é o módulo de elasticidade do material,  $A$  a área da seção transversal e  $L$  o comprimento da barra, enquanto que os graus de liberdade locais da barra são, res-

pectivamente, os deslocamentos normal e transversal (com relação ao eixo da barra, ou horizontal e vertical, com relação ao sistema global), respectivamente nos nós (1) e (2), portanto 4 graus de liberdade e uma matriz 4x4, conforme indicado na Fig. 3.2, com  $u'_1$ ,  $v'_1$ ,  $u'_2$  e  $v'_2$  representando os graus de liberdade locais em ordem.

**Figura 3.2** – Representação de uma barra de treliça e seus graus de liberdade nos sistemas locais (representados pelo índice linha) e globais



Fonte: Autor

Para uma barra  $i$  qualquer, numa orientação rotacionada de um ângulo  $\theta$  com relação ao sistema global (indicado também na Fig. 3.2 pelo par de eixos  $x$  e  $y$ ), a matriz que rotaciona as informações de deslocamentos dos graus de liberdade do sistema local para o global pode ser escrita como

$$[R]_i = \begin{bmatrix} \cos(\theta_i) & \sin(\theta_i) & 0 & 0 \\ -\sin(\theta_i) & \cos(\theta_i) & 0 & 0 \\ 0 & 0 & \cos(\theta_i) & -\sin(\theta_i) \\ 0 & 0 & \sin(\theta_i) & \cos(\theta_i) \end{bmatrix} \quad (3.21)$$

Com  $[R]_i$  definida, a matriz de rigidez de cada elemento de barra, escrita no sistema global, fica dada pela relação

$$[k_g]_i = [R]_i^T [k_l]_i [R]_i \quad (3.22)$$

Nessa matriz, os graus de liberdade passam a ser o deslocamento horizontal e vertical dos nós 1 e 2 da barra, respectivamente, indicados na Fig. 3.2 como  $u_1$ ,  $v_1$ ,  $u_2$  e  $v_2$ .

A matriz global da estrutura como um todo pode ser escrita, de uma maneira estritamente matemática, como

$$[K_g] = \sum_{i=1}^n [I_n]_i^T [k_g]_i [I_n]_i \quad (3.23)$$

na qual  $[I_n]_i$  é uma matriz topológica, definida de modo a fazer com que parcelas de rigidez das matrizes de cada barra sejam indexadas nos graus de liberdade correspondentes na matriz global, com a matriz de cada barra tendo sua própria indexação, e sendo  $n$  o número de barras a serem indexadas. Na prática, essa operação é custosa e portanto não executada. Computacionalmente, a indexação é feita termo a termo, diretamente, sem que a multiplicação matricial seja realizada, tendo a mesma apenas sentido formal e matemático. Na matriz global cheia, os graus de liberdade são, também, os deslocamentos horizontal e vertical, respectivamente, porém de cada nó da estrutura completa e não os dois nós de cada barra que a compõe. As linhas da matriz topológica são associadas aos graus de liberdade do elemento (já rotacionados para o sistema global), sendo as colunas associadas aos graus de liberdade da estrutura inteira. Se o grau de liberdade  $j$  do elemento  $i$  corresponde ao grau  $k$  da estrutura,  $I_{ni}(j, k) = 1$ ; caso contrário,  $I_{ni}(j, k) = 0$ .

Expandindo a Eq. 3.23 com a Eq. 3.22, tem-se

$$[K_g] = \sum_{i=1}^n [I_n]_i^T [R]_i^T [k_l]_i [R]_i [I_n]_i \quad (3.24)$$

ou, ainda,

$$[K_g] = \sum_{i=1}^n [A]_i^T [k_l]_i [A]_i \quad (3.25)$$

na qual  $[A]_i$  é a matriz de incidência cinemática do elemento, aqui, especificamente, escrita como  $[A]_i = [R]_i [I_n]_i$  e, portanto, com  $[A]_i^T = ([R]_i [I_n]_i)^T = [I_n]_i^T [R]_i^T$

Analogamente, o vetor de forças nodais no elemento  $i$  no sistema local ( $\{f_l\}_i$ ) pode

ser expresso no sistema global como

$$\{f_g\}_i = [R]_i^T \{f_l\}_i \quad (3.26)$$

e indexado como

$$\{F_g\} = \sum_{i=1}^n [I_n]_i^T \{f_g\}_i \quad (3.27)$$

para gerar o vetor de forças global da estrutura  $\{F_g\}$ .

Usando-se a matriz de incidência  $[A]_i$ , pode escrever ainda

$$\{F_g\} = \sum_{i=1}^n [A]_i^T \{f_l\}_i \quad (3.28)$$

Por fim, a equação de equilíbrio global do sistema estrutural fica como

$$[K_g]\{U_g\} = \{F_g\} \quad (3.29)$$

na qual  $\{U_g\}$  são os deslocamentos globais dos nós da estrutura, que podem ser explicitados como

$$\{U_g\} = [K_g]^{-1}\{F_g\} \quad (3.30)$$

na qual  $[K_g]^{-1}$  é o inverso da matriz de rigidez, também chamada de matriz de flexibilidade.

Porém, no formato apresentado para a matriz de rigidez, a mesma não possui inversa, uma vez que a vinculação da estrutura ainda não foi levada em consideração no valor dos seus elementos. Para uma estrutura não vinculada, um vetor de forças  $\{F_g\}$  não nulo estaria associado a infinitos possíveis vetores de deslocamento  $\{U_g\}$ , que seriam determinados de maneira unitária somente através de algum outro mecanismo de restrição, por exemplo a inércia em problemas dinâmicos. Para um problema estático, em que não existe mecanismo inercial, restrições diretamente aplicadas sob os deslocamentos precisam ser definidas. Tais restrições são associadas à presença de elementos de vinculação da estrutura, que por sua vez geram deslocamentos nos nós restringidos cujos valores são prescritos. A prescrição destes valores é feita através de uma rigidez artificial (aqui chamada de penalidade). Neste caso, o elemento da diagonal principal da matriz de rigidez associado ao grau de liberdade cujo valor do deslocamento é fixado fica escrito como

$$K_{g_{ii}} = K_{\alpha} \quad (3.31)$$

na qual  $i$  é o grau de liberdade fixado e  $K_{\alpha}$  é um valor grande, aqui definido como

$$K_{\alpha} = 10^5 \max \left( \text{diag}([K_g]) \right) \quad (3.32)$$

ou seja: um número (positivo) igual ao maior valor encontrado entre todos os coeficientes da diagonal da matriz de rigidez multiplicado por um coeficiente suficientemente grande, aqui considerado como  $10^5$ . Vale salientar que, até certo ponto, quanto maior o coeficiente, mais próximo será o deslocamento encontrado no grau de liberdade  $i$  do valor prescrito e, portanto, mais próximos serão os deslocamentos dependentes dos mesmo. Contudo, valores muito grandes podem gerar problemas de condicionamento na matriz de rigidez, esses associados a imprecisões durante o processo numérico de solução do sistema linear e/ou inversão da matriz. O valor de  $10^5$  foi considerado pelo autor, após algumas experimentações numéricas, como suficientemente grande para gerar respostas com boa precisão sem, contudo, causar instabilidades significativas.

Para completar a prescrição do deslocamento, modifica-se também o vetor de forças associado ao grau de liberdade  $i$ , que fica escrito como

$$F_{g_i} = K_{\alpha} \bar{U}_i \quad (3.33)$$

na qual  $\bar{U}_i$  é o valor prescrito para o deslocamento. Dessa forma, uma vez que o sistema é resolvido, tem-se, para o grau de liberdade  $i$ , a seguinte equação

$$\sum_{j=1}^{2m} K_{g_{ij}} U_{g_j} = F_{g_i} \quad (3.34)$$

na qual  $m$  é o número de nós da estrutura, sendo  $2m$  o número de graus de liberdade, para o caso de um problema bidimensional de treliça.

Se o valor de  $K_{\alpha}$  for suficientemente grande, tem-se, para a soma à esquerda, que

$$\sum_{j=1}^{2m} K_{g_{ij}} U_{g_j} \approx K_{\alpha} U_{g_i} \quad (3.35)$$

e, portanto, substituindo na Eq. 3.34, tem-se que

$$K_\alpha U_{g_i} \approx F_{g_i} \quad (3.36)$$

Por fim, usando a Eq. 3.33 para o valor de  $F_{g_i}$ , tem-se

$$K_\alpha U_{g_i} \approx K_\alpha \bar{U}_i \quad (3.37)$$

que, por sua vez, resulta em

$$U_{g_i} \approx \bar{U}_i \quad (3.38)$$

ou seja: para valores de  $K_\alpha$  suficientemente grandes, o deslocamento que se deseja preservar se aproxima do valor prescrito. Tal método foi utilizado, em oposição a outras técnicas numéricas mais robustas, para preservar a integridade das matrizes de rigidez, que são reutilizadas, como será apresentado futuramente, no cálculo dos vetores gradiente e matrizes Hessianas do problema de otimização.

Dessa forma, para condições de vinculação que garantam iso ou hiperstaticidade da estrutura, tanto interna quanto externamente, a matriz de rigidez pode ser invertida e o sistema solucionado para um valor único do vetor  $U_g$ . Uma vez que o mesmo é determinado, pode-se fazer o caminho inverso para calcular os vetores de força e os esforços solicitantes internos do elemento. Neste caso, o deslocamento nos nós de um elemento, escritos no sistema global, podem ser expressos como

$$\{u_g\}_i = [I_n]_i \{U_g\} \quad (3.39)$$

portanto fazendo a indexação no sentido inverso (da estrutura inteira para o elemento).

Os mesmos podem ser convertidos para deslocamentos no sistema local usando a relação

$$\{u_l\}_i = [R]_i \{u_g\}_i \quad (3.40)$$

ou seja:

$$\{u_l\}_i = [A]_i \{U_g\} \quad (3.41)$$

Na sequência, a deformação axial do elemento pode ser diretamente obtida, em função dos deslocamentos locais, como

$$\epsilon_i = \frac{u'_{2i} - u'_{1i}}{L_i} \quad (3.42)$$

na qual  $u'_{1i}$  e  $u'_{2i}$  são os deslocamentos axiais do elemento no sistema local, respectivamente do nó 1 e 2. Essa equação também pode ser escrita, matricialmente, em termos do vetor de deslocamentos locais  $\{u_l\}_i$ , como

$$\epsilon_i = \frac{1}{L_i} \begin{Bmatrix} -1 & 0 & 1 & 0 \end{Bmatrix} \{u_l\}_i \quad (3.43)$$

ou, abreviadamente, como

$$\epsilon_i = \frac{1}{L_i} \{a\}^T \{u_l\}_i \quad (3.44)$$

na qual  $\{a\}^T$  é o vetor  $\{-1 \ 0 \ 1 \ 0\}$

Substituindo essa deformação no lado direito da Eq. 3.6, referente às parcelas desmembradas da deformação total, chega-se à expressão

$$\frac{1}{L_i} \{a\}^T \{u_l\}_i = \epsilon_i^E + \epsilon_i^i + \epsilon_i^T \quad (3.45)$$

Reorganizando a expressão, pode-se explicitar a deformação elástica como

$$\epsilon_i^E = \frac{1}{L_i} \{a\}^T \{u_l\}_i - \epsilon_i^i + \epsilon_i^T \quad (3.46)$$

ou, mais ainda, considerando-se que a deformação inelástica é puramente oriunda da fração congelada, como

$$\epsilon_i^E = \frac{1}{L_i} \{a\}^T \{u_l\}_i - \epsilon_i^\phi + \epsilon_i^T \quad (3.47)$$

De posse da deformação elástica, o esforço normal é calculado, usando-se também o módulo de elasticidade equivalente, como

$$N_i = E_i A_i \epsilon_i^E \quad (3.48)$$

ou, de maneira expandida, como

$$N_i = E_i A_i \left( \frac{1}{L_i} \{a\}^T \{u_l\}_i - \epsilon_i^\phi + \epsilon_i^T \right) \quad (3.49)$$

Por fim, com o esforço normal, o vetor de forças no sistema local pode ser expresso como

$$f_{li} = N_i \left\{ -1 \quad 0 \quad 1 \quad 0 \right\}^T \quad (3.50)$$

ou, ainda, de maneira expandida e matricial, como

$$\{f_l\}_i = \left[ E_i A_i \left( \frac{1}{L_i} \{a\}^T \{u_l\}_i - \epsilon_i^\phi + \epsilon_i^T \right) \right] \{a\} \quad (3.51)$$

Igualmente, essa equação pode ser escrita de maneira mais compacta, em função da matriz de rigidez local, como

$$\{f_l\}_i = [k_l]_i (\{u_l\}_i + \{u_l^T\}_i - \{u_l^\phi\}_i) \quad (3.52)$$

na qual  $\{u_l^T\}_i$  são os deslocamentos provocados pela deformação térmica, sendo  $\{u_l^\phi\}_i$  o deslocamento inelástico. Ambos os termos podem ser expandido, resultando em

$$\{f_l\}_i = [k_l]_i (\{u_l\}_i + L_i \alpha_i \Delta T_i \{a\} - L_i \Delta \phi_i \epsilon_i^\phi \{a\}) \quad (3.53)$$

Ignorando-se os efeitos térmicos e partindo-se de uma deformação cristalizada nula, cristalizando-se a deformação obtida através do deslocamento  $\{u_l\}_i$ , o vetor de forças local pode ser escrito, simplificadamente, como

$$\{f_l\}_i = [k_l]_i (\{u_l\}_i - \Delta \phi_i \{u_l\}_i) \quad (3.54)$$

ou, mais simplificadamente ainda, como

$$\{f_l\}_i = (1 - \Delta \phi_i) [k_l]_i \{u_l\}_i \quad (3.55)$$

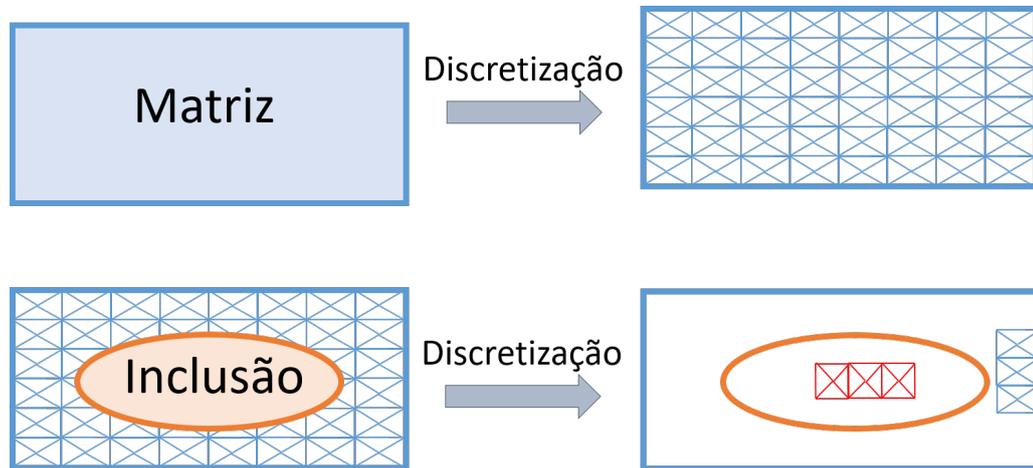
Esse formato simplificado será útil a seguir, quando o problema de otimização proposto estiver sendo discutido.

### 3.3 Modelo de Discretização

Para o tratamento do compósito polimérico em analogia a um modelo contínuo, uma malha discreta dos elementos de treliça unidimensionais apresentados anteriormente é empregada. Inicia-se por calcular, para o caso bidimensional considerado neste trabalho, a área de domínio do compósito. Na sequência, essa região é povoada de nós, os quais são

conectados por elementos de treliça, de modo a preencher todo o domínio por barras da maneira mais uniforme possível, sendo as mesmas representativas da matriz do compósito. Tal procedimento é ilustrado na parte superior da Fig. 3.3.

**Figura 3.3 – Figura ilustrativa do processo de discretização**



Fonte: Autor

Para o cálculo das áreas de cada barra da matriz, a área total da região do compósito é definida e então dividida pelo total de barras, cada qual recebendo uma fração dessa área proporcional ao seu comprimento. Tal procedimento pode ser matematicamente descrito como

$$A_i^M = A^{Tot} \frac{L_i}{L^{tot}} \quad (3.56)$$

na qual  $A_i^M$  é a área da barra de matriz  $i$ , sendo  $L_i$  seu comprimento e sendo  $A^{Tot}$  a área total da região discretizada.  $L^{Tot}$  é a soma dos comprimentos de todas as barras, portanto

$$\sum_{i=1}^n A_i^M = \frac{A^{Tot}}{L^{tot}} \sum_{i=1}^n L_i = A^{Tot} \quad (3.57)$$

na qual  $n$  é o número total de barras de matriz, ou seja:  $\sum_{i=1}^n L_i = L^{tot}$

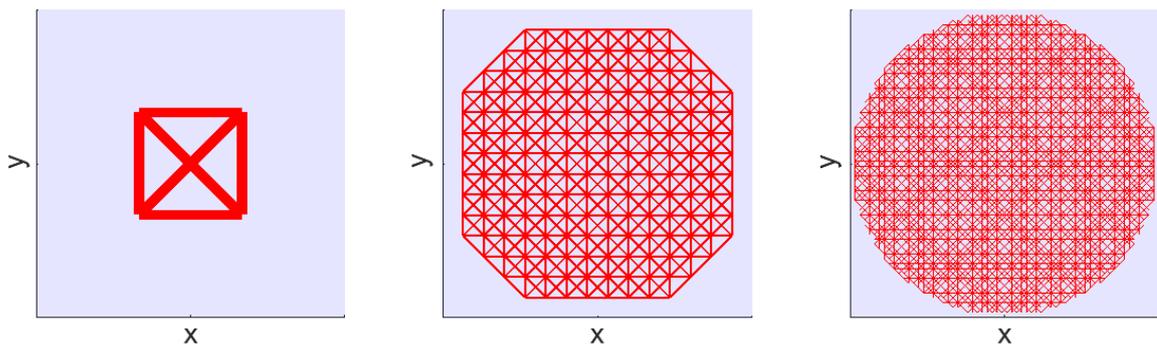
Na sequência, uma sub-região é definida no domínio, sendo a mesma aquela dentro da qual coexistem as barras anteriormente descritas, referentes à matriz, e as de inclusão. Para todas as barras de matriz cujos dois nós extremos estejam dentro da região delimitada, conecta-se aos mesmos nós extremos (portanto superposta à barra de matriz) uma barra de inclusão. Tal barra tem área da seção transversal definida como

$$A_i^I = \lambda^{vol} A_i^M \quad (3.58)$$

na qual  $A_i^I$  é a área da barra de inclusão associada à barra de matriz  $i$ , sendo  $\lambda^{vol}$  a fração volumétrica de inclusão. Tal procedimento é ilustrado na parte inferior da Fig. 3.3, com apenas algumas barras da discretização apresentadas por questão de visualização.

É importante mencionar que a região de inclusão definida não é exatamente representada pela estratégia proposta para a maioria dos casos, mas sim assintoticamente, sendo melhor a aproximação quanto mais finamente discretizada for a *Ground Structure*. Um exemplo disso é mostrado na Fig. 3.4, na qual uma região de inclusão circular é centrada numa matriz quadrada e com espaçamento uniforme em  $x$  e em  $y$  para diferentes graus de refino da malha.

**Figura 3.4 – Efeito do refinamento da malha na discretização de uma região de inclusão circular**



**Fonte: Autor**

Salienta-se ainda que quanto maior a diferença entre  $\Delta x$  e  $\Delta y$ , mais anisotrópica é a resposta da malha de barras. Para aproximar um problema isotrópico, deve-se escolher valores de  $n_x$  e  $n_y$  que minimizem essa diferença, como feito neste trabalho, conforme descrito anteriormente.

Uma consideração importante a ser feita é que essa analogia do modelo discreto com o contínuo exige parcimônia, uma vez que não leva em consideração alguns efeitos que podem ser importantes para o comportamento do compósito, como a aderência entre a matriz e a inclusão, efeito do Coeficiente de Poisson na resposta da estrutura, efeitos de dano, etc. Mais ainda, não é tratada a superposição entre elementos de matriz e inclusão, que ocupam o mesmo volume dentro do compósito, significando que, para a discretização apresentada anteriormente, existe uma incoerência volumétrica (o volume total do compósito é maior que o volume da região definida para o mesmo, por conta

da adição da inclusão ao volume da matriz). Por todos estes fatores, deve-se entender o trabalho como definindo um modelo fundamentalmente discreto, cuja associação ao caso contínuo confere apenas uma aproximação. Tal aproximação é melhor quanto mais reduzida for a fração volumétrica empregada, como foi o caso para os exemplos tratados na sequência. Ainda assim, reitera-se que o trabalho é melhor aplicado ao projeto de estruturas discretas, como discutido ao final deste documento.

### 3.4 Treinamento e Forma Memorizada

Para a etapa de treinamento do compósito, os deslocamentos prescritos são aplicados na matriz de rigidez através do método descrito anteriormente, de maneira que o sistema linear tenha solução única. Usando a Eq. 3.30 para a etapa de treinamento, tem-se

$$\{U_g^T\} = [K_g^T]^{-1}\{F_g^T\} \quad (3.59)$$

na qual o índice superior  $T$  refere-se à etapa de treinamento, sendo o vetor  $\{F_g^T\}$ , referente às ações externas, nulo, à exceção das entradas correspondentes aos deslocamentos prescritos não nulos da superfície esquerda do polímero. Como dito,  $[K_g^T]$  é a matriz de rigidez levando-se em consideração todas as barras em estado borrachoso, sendo essa a maneira mais comum de treinar compósitos poliméricos com memória de forma e a adotada neste trabalho.

De posse desses deslocamentos, calculam-se os vetores de força locais partindo-se da Eq. 3.55. Nela, substitui-se o deslocamento local  $\{u_l\}_i$  pela expressão apresentada em Eq. 3.41, resultando em,

$$\{f_l^T\}_i = (1 - \Delta\phi)[k_l^T]_i[A]_i\{U_g^T\} \quad (3.60)$$

sendo  $\Delta\phi$ , por enquanto, nulo.

De posse desses vetores locais, pode-se escrever um vetor de forças globais nodais oriundo da reação dos elementos tentando recuperar sua configuração indeformada. Tal vetor é aplicado como estado inicial de tensão da etapa de liberação da deformação prescrita no compósito, e é expresso, tomando-se como base a Eq. 3.28, como

$$\{F_g^S\} = - \sum_{i=1}^n [A]_i^T \{f_l^T\}_i \quad (3.61)$$

na qual o índice superior  $S$  refere-se aos esforços internos, sendo o sinal negativo necessário

para respeitar a convenção de se transportar esforços de uma etapa para a outra como um estado inicial de tensão. Salienta-se que o índice  $T$ , estando fora das chaves cujas siglas internas nomeiam a matriz, refere-se à transposta clássica, como é o caso de  $[A]_i^T$ , notação que vale para o restante da demonstração.

Esse vetor, por se tratar das forças de reação internas, que buscam trazer a estrutura de volta à sua configuração indeformada original, obedece à relação, obtida diretamente da sua definição, de que

$$[K_g^T]^{-1}\{F_g^T\} + [K_g^S]^{-1}\{F_g^S\} = \{0\} \quad (3.62)$$

na qual  $[K_g^S]$  é também uma matriz de rigidez referente às barras no estado borrachoso, analogamente a  $[K_g^T]$ , e com as mesmas condições de vinculação, mas não penalizada para a aplicação das deformações de treinamento, ou seja: caso liberada das deformações prescritas, a estrutura volta à sua configuração indeformada, associada ao vetor de deslocamentos globais nulos  $\{0\}$ .

Quando a inclusão é congelada, porém, os vetores de força locais são modificados, posto que há alteração na parcela  $\Delta\phi$ , previamente nula. Para um congelamento total da inclusão, mais especificamente, o termo  $(1 - \Delta\phi)$  torna-se nulo, o que por sua vez torna os vetores de força dos elementos de inclusão nulos. Essa modificação gera um 'desbalanço' dos esforços internos, também causado pelo fato de que a própria matriz de rigidez é modificada, uma vez que a rigidez dos elementos congelados aumenta drasticamente. Sendo a nova matriz de rigidez modificada penalizada chamada de  $[K_g^L]$ , onde o índice superior  $L$  refere-se à liberação da estrutura congelada, à qual está associado um vetor de forças penalizado  $\{F_g^L\}$ , tem-se que os deslocamentos finais dos nós globais do polímero após as etapas de treinamento e de liberação-congelamento podem ser escritos como

$$\{U_g^{Tot}\} = [K_g^T]^{-1}\{F_g^T\} + [K_g^L]^{-1}\{F_g^L\} \quad (3.63)$$

na qual  $\{F_g^L\}$  é o vetor de forças internas pós-congelamento, incluindo a penalização. Tal formato é oriundo do fato de que o vetor  $\{F_g^L\}$  foi escrito como estado de tensão advindo da etapa anterior, de treinamento, porém na configuração indeformada, segundo a aproximação de pequenos deslocamentos utilizada. Para 'corrigir' essa aplicação na configuração indeformada, uma vez que a recuperação acontece partindo-se do composto sob a ação da deformação de treinamento, faz-se a soma das duas parcelas, pelo princípio da superposição linear. Dessa forma,  $\{F_g^L\}$  pode ser escrito, simplificadamente, como

$$\{F_g^L\} = - \sum_{i=n_1}^{n^m} [A]_i^T \{f_l^T\}_i \quad (3.64)$$

na qual  $n^1, n^2 \dots n^m$  são os índices dos elementos de matriz, uma vez que a inclusão tem o esforço zerado.

$\{U_g^{Tot}\}$  pode ser usado, após a solução das duas etapas que o compõe, para determinar os esforços internos da estrutura após a liberação usando uma expressão análoga à anteriormente discutida em 3.60 como

$$\{f_l^L\}_i = [k_l^L]_i [A]_i \{U_g^{Tot}\} \quad (3.65)$$

Combinando as Eqs. 3.63 e 3.65, chega-se a

$$\{U_g^{Tot}\} = [K_g^T]^{-1} \{F_g^T\} + [K_g^L]^{-1} \left( - \sum_{i=n_1}^{n^m} [A]_i^T \{f_l^T\}_i \right) \quad (3.66)$$

ou, ainda,

$$\{U_g^{Tot}\} = [K_g^T]^{-1} \{F_g^T\} - [K_g^L]^{-1} \left( \sum_{i=n_1}^{n^m} [A]_i^T \{f_l^T\}_i \right) \quad (3.67)$$

como formato final para os deslocamentos totais após o ciclo termo-mecânico.

### 3.5 Formalização da Otimização

Nesta seção, define-se a otimização a ser aplicada ao projeto dos compósitos de maneira objetiva e matemática, bem como as grandezas associadas à mesma e suas propriedades. Inicia-se por definir a função objetivo e o problema de otimização. Na sequência, formulam-se o gradiente da função objetivo e sua Hessiana, e faz-se algumas considerações a respeito das propriedades e usos de ambos.

#### 3.5.1 Problema de Otimização

Para a formulação do problema de otimização, inicia-se por definir uma função objetivo que se deseja minimizar. Neste trabalho, a função objetivo a ser minimizada é definida como

$$f(A_i^I) = ||\{X(A_i^I)\} - \{\bar{X}\}||^2 \quad (3.68)$$

na qual  $A_i^I$ , as variáveis de projeto, são as áreas das barras de inclusão, sendo  $\{X\}$  as posições dos nós da discretização e  $\{\bar{X}\}$  as posições prescritas após o processo de deformação. Portanto, deseja-se modificar as áreas das barras de inclusão, efetivamente modificando a topologia da região de inclusão, uma vez que as mesmas podem ser inclusive nulas, de modo que a diferença entre a configuração deformada prescrita para o compósito e aquela encontrada após o processo de otimização, medida através do quadrado da norma da diferença entre ambas,  $\|X(A_i^I) - \{\bar{X}\}\|^2$ , seja a menor possível. Neste caso, o vetor  $\bar{X}$  tem formato

$$\{\bar{X}\} = \begin{bmatrix} \bar{x}_1 \\ \bar{y}_1 \\ \bar{x}_2 \\ \bar{y}_2 \\ \vdots \end{bmatrix} \quad (3.69)$$

na qual  $\bar{x}_1$  é a posição prescrita para o nó 1 na direção  $x$ , sendo  $\bar{y}_1$  a posição prescrita para o nó 1 na direção  $y$ , e assim analogamente para os nós 2 em diante. Dessa maneira, o vetor  $\{X(A_i^I)\}$ , ordenado em formato análogo, pode ser escrito como

$$\{X(A_i^I)\} = \{X_0\} + \{U_g^{Tot}(A_i^I)\} \quad (3.70)$$

na qual  $\{X_0\}$  é o vetor de posições originais dos nós, referentes à estrutura indeformada, e  $\{U_g\}$  é o vetor de deslocamentos globais totais, calculado após o processo de congelamento e liberação da deformação inicialmente prescrita em estado borrachoso. Portanto, a função objetivo pode ser escrita também como

$$f(A_i^I) = \|\{\{X_0\} + \{U_g^{Tot}(A_i^I)\} - \{\bar{X}\}\}\|^2 \quad (3.71)$$

ou ainda

$$f(A_i^I) = \|\{\{U_g^{Tot}(A_i^I)\} - \{\Delta X\}\}\|^2 \quad (3.72)$$

na qual  $\{\Delta X\} = \{\bar{X}\} - \{X_0\}$  é a diferença entre a posição original e a prescrita dos nós.

De posse da função objetivo, para se definir por completo o problema de otimização, resta explicitar as restrições desejadas. Neste trabalho, considera-se apenas uma restrição lateral de mínimo (ou seja: um menor valor possível para as variáveis de projeto). Com isso, o problema de otimização fica escrito como

$$\begin{aligned} \text{Min}_{\{A^I\}} \quad & \|\{U_g^{Tot}(A_i^I)\} - \{\Delta X\}\|^2 \\ \text{s.t} \quad & A_i^I \geq \bar{A}^{min} \end{aligned} \tag{3.73}$$

ou seja: nenhuma barra  $i$  pode ter área menor do que um certo valor mínimo estipulado. A escolha desse mínimo, bem como o procedimento empregado quando tal restrição é violada, é discutido na subseção seguinte.

### 3.5.2 Estratégia de Filtragem

O valor mínimo desejado para as áreas das barras de inclusão,  $\bar{A}^{min}$ , pode, à princípio, ser qualquer valor positivo (posto que valores negativos não seriam fisicamente coerentes). Porém, áreas muito pequenas podem gerar instabilidades numéricas, criando sistemas mal-condicionados, nos quais alguns valores são muito maiores que outros. Mais ainda, valores nulos de área geram indefinições em frações nas quais tanto os numeradores quanto os denominadores caminham para zero. Ainda que esses problemas sejam sanados, a presença de barras muito finas gera, para o problema em questão, topologias muito esparsas, com um grande número de elementos de inclusão espalhados por toda a matriz. Tais elementos pequenos e distribuídos são, em muitos casos, indesejáveis, uma vez que os mesmos dificultam a impressão e geram peças esteticamente desagradáveis. Com isso, tem-se a motivação de desenvolver uma estratégia de filtragem na qual barras muito pequenas são excluídas da análise, efetivamente reduzindo o número de variáveis de projeto. Tal filtragem acarreta em uma gama de vantagens: não é necessário lidar com valores nulos, há uma melhora no condicionamento dos sistemas lineares tratados e os algoritmos de otimização são acelerados, uma vez que precisam lidar com um número cada vez menor de variáveis de projeto à medida que as iterações avançam.

A estratégia de filtragem aqui adotada é simples, porém eficiente: considera-se que  $\bar{A}^{min}$  é escrito, em cada passo  $j$  do processo de otimização, como

$$\bar{A}_j^{min} = \kappa \max(\{A^I\}_j) \tag{3.74}$$

Isso significa que, para cada passo  $j$ , todas as barras menores que uma certa fração  $\kappa$  da área da maior barra são permanentemente eliminadas do problema de otimização, efetivamente reduzindo-o. Tal estratégia é análoga à uma prática comum em problemas de otimização topológica, em que um filtro semelhante é aplicado durante a otimização. A diferença é que, neste trabalho, ao invés de se aplicar o filtro uma vez a cada  $n$  iterações, com  $n$  fornecido pelo usuário, o filtro é aplicado em toda iteração. Mais ainda, uma estratégia comum em outros problemas de otimização é usar um valor de  $\kappa$  pequeno, gerando ainda uma grande quantidade de barras pequenas. Essas barras, contudo, são

eliminadas durante uma etapa posterior de pós-processamento, gerando topologias menos esparsas. Neste problema, contudo, tal estratégia mostrou-se ineficiente: uma grande quantidade de barras inicialmente finas, em iterações futuras do algoritmo, aumentam de área largamente, porém gerando apenas reduções ínfimas no valor da função objetivo. Isso faz com que topologias com erros pouco menores tenham uma quantidade muito maior de barras, às vezes com áreas maiores até do que as barras mais importantes para a redução do erro, sendo as mesmas difíceis de filtrar. Por fim, eliminando-se imediatamente barras que demoram a aumentar de área, uma vez que tem pouca interferência no erro, observa-se que o algoritmo modifica as barras que restaram de modo a reduzir o erro em iterações subsequentes, o que não seria possível em caráter de pós-processamento. Aliando-se este fato observado à propriedade de que a redução do número de variáveis de projeto acarreta em uma elevada redução, também, do custo computacional de cada iteração do algoritmo de otimização, adota-se, neste trabalho, a seguinte estratégia: ao invés de se usar um valor de  $\kappa$  pequeno e uma etapa de pós-processamento,  $\kappa$  é escolhido, em conjunto com outros parâmetros de modelo, de forma a produzir topologias 'limpas', mas sem grandes aumentos de erro em relação a topologias mais dispersas. O efeito do parâmetro  $\kappa$  é apresentado em maiores detalhes no capítulo seguinte, no qual problemas de ilustrativos são resolvidos em diferentes níveis de filtro. Neste caso, o filtro de pós-processamento mostrou-se pouco efetivo em remover barras que pouco afetam o erro, uma vez que isto é feito durante as iterações do algoritmo de otimização. O mesmo, portanto, não será utilizado.

### 3.5.3 Vetor Gradiente

Para muitos algoritmos de otimização, incluindo o utilizado neste trabalho, fornecer o gradiente da função objetivo facilita e acelera consideravelmente a definição da solução ótima, especialmente se o mesmo for calculado de maneira eficiente. Algoritmos que necessitam do gradiente, quando o mesmo não é fornecido, procuram aproximá-lo por alguma técnica numérica, como por exemplo o Método das Diferenças Finitas. O mesmo requer a avaliação da função objetivo ao menos uma vez para cada uma das variáveis de projeto, o que, para casos nos quais esta é custosa ou o número de variáveis é muito grande, pode acarretar em tempos computacionais impraticáveis. Particularmente no âmbito deste trabalho, em que a avaliação da função objetivo requer a construção e solução de 2 sistemas lineares (um para a etapa de treinamento e outro para a de liberação), processo cujo custo computacional excede  $O(N^2)$  (ou seja: tem complexidade maior que quadrática com relação ao número de graus de liberdade), repetir a avaliação para  $M$  variáveis de projeto, assumindo  $M$  como uma fração de  $N$ , resulta em uma complexidade computacional maior que  $O(N^3)$ , tornando-se, portanto, rapidamente inviável para valores de  $N$  grandes. Desta maneira, esta seção foca na definição de um gradiente computacionalmente eficiente, objetivando que o mesmo não se torne o gargalo da

implementação.

Iniciado-se essa definição, o gradiente da função objetivo aqui tratada pode ser escrito matematicamente como

$$\nabla f(A_i^I) = \frac{\partial f}{\partial A_j^I} = \frac{\partial}{\partial A_j^I} \left\| \{U_g^{Tot}\} - \{\Delta X\} \right\|^2 \quad (3.75)$$

Aplicando a derivada no quadrado da norma, chega-se a

$$\nabla f(A_i^I) = 2[J]^T \left( \{U_g^{Tot}\} - \{\Delta X\} \right) \quad (3.76)$$

na qual  $T$  denota a transposta de uma matriz ou vetor e  $[J]$  é a matriz Jacobiana de  $\{U_g^{Tot}\}$ , que pode ser escrita como

$$[J] = \frac{\partial \{U_g^{Tot}\}}{\partial \{A^I\}} \quad (3.77)$$

na qual

$$J_{ij} = \frac{\partial U_g^{Tot\ i}}{\partial A_j^I} \quad (3.78)$$

Na sequência, expande-se a derivada do termo  $\{U_g^{Tot}\}$ . Inicia-se por escrever um sistema linear global tradicional do método da rigidez direta, apresentado de maneira genérica, para uma estrutura qualquer, como

$$[K_g]\{U_g\} = \{F_g\} \quad (3.79)$$

Derivando-se ambos os lados da equação com relação a uma variável qualquer, assumindo-se que todos os termos dependem da variável em questão, tem-se, respeitando-se a ordem das derivações no âmbito de cálculo vetorial, que

$$[K_g]' \{U_g\} + [K_g]\{U_g\}' = \{F_g\}' \quad (3.80)$$

na qual o símbolo  $'$  denota a derivação com relação à variável escolhida.

Na sequência, a equação é reorganizada, visando isolar o termo  $\{U_g\}'$ , o que resulta em

$$\{U_g\}' = [K_g]^{-1} \left( \{F_g\}' - [K_g]' \{U_g\} \right) \quad (3.81)$$

Expandindo-se o termo  $\{U_g^{Tot}\}$  dentro da derivada na Eq. 3.76 como a soma dos deslocamentos das etapas de treinamento e liberação, de acordo com a análise incremental proposta, bem como substituindo-se as derivadas dos deslocamentos de cada etapa de acordo com a Eq. 3.81, particularizado-a para cada termo, chega-se ao formato

$$\begin{aligned} \nabla f(A_i^I) = 2 \left[ [K_g^T]^{-1} \left( \frac{\partial \{F_g^T\}}{\partial A_j^I} - \frac{\partial [K_g^T]}{\partial A_j^I} \{U_g^T\} \right) + \right. \\ \left. [K_g^L]^{-1} \left( \frac{\partial \{F_g^L\}}{\partial A_j^I} - \frac{\partial [K_g^L]}{\partial A_j^I} \{U_g^L\} \right) \right]^T \left( \{U_g^{Tot}\} - \{\Delta X\} \right) \quad (3.82) \end{aligned}$$

e, como  $\{F_g^T\}$  não depende de  $A_j^I$ , tem-se que

$$\begin{aligned} \nabla f(A_i^I) = 2 \left[ - [K_g^T]^{-1} \frac{\partial [K_g^T]}{\partial A_j^I} \{U_g^T\} + \right. \\ \left. [K_g^L]^{-1} \left( \frac{\partial \{F_g^L\}}{\partial A_j^I} - \frac{\partial [K_g^L]}{\partial A_j^I} \{U_g^L\} \right) \right]^T \left( \{U_g^{Tot}\} - \{\Delta X\} \right) \quad (3.83) \end{aligned}$$

Para continuar a simplificação do gradiente, investiga-se, na sequência, as derivadas das matrizes de rigidez,  $\partial[K_g^T]/\partial A_j^I$  e  $\partial[K_g^L]/\partial A_j^I$ . Para qualquer uma das duas matrizes, novamente representadas genericamente por  $[K_g]$ , tem-se, segundo a Eq. 3.25, que

$$\frac{\partial}{\partial A_j^I} [K_g] = \frac{\partial}{\partial A_j^I} \left( \sum_{i=1}^n [A]_i^T [k_l]_i [A]_i \right) \quad (3.84)$$

Como as matrizes de indexação cinemática não dependem de  $A_j^I$ , pode-se entrar com a derivada no somatório como

$$\frac{\partial}{\partial A_j^I} [K_g] = \sum_{i=1}^n [A]_i^T \frac{\partial [k_l]_i}{\partial A_j^I} [A]_i \quad (3.85)$$

Mais ainda, as matrizes locais dos elementos dependem apenas das áreas do próprio elemento, e não dos outros, portanto

$$\frac{\partial[k_l]_i}{\partial A_j^I} = \begin{cases} 0, & i \neq j \\ \frac{\partial[k_l]_j}{\partial A_j^I}, & i = j \end{cases}$$

e, expandindo-se  $[k_l]_j$ , usando-se a Eq. 3.20, tem-se que

$$\frac{\partial[k_l]_j}{\partial A_j^I} = \frac{\partial}{\partial A_j^I} \left( \frac{E_j A_j}{L_j} \begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \right) = \frac{E_j}{L_j} \begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} = \frac{1}{A_j^I} [k_l]_j \quad (3.86)$$

ou seja:

$$\frac{\partial[K_g]}{\partial A_j^I} = \frac{1}{A_j^I} [A]_j^T [k_l]_j [A]_j \quad (3.87)$$

Aplicando-se esse resultado na Eq. 3.83, chega-se a

$$\begin{aligned} \nabla f(A_i^I) = 2 \left[ - [K_g^T]^{-1} \left( \frac{1}{A_j^I} [A]_j^T [k_l^T]_j [A]_j \{U_g^T\} \right) + \right. \\ \left. [K_g^L]^{-1} \left( \frac{\partial\{F_g^L\}}{\partial A_j^I} - \frac{1}{A_j^I} [A]_j^T [k_l^L]_j [A]_j \{U_g^L\} \right) \right]^T (\{U_g^{Tot}\} - \{\Delta X\}) \quad (3.88) \end{aligned}$$

Para continuar a simplificação, usa-se a relação  $\{u_l\}_j = [A]_j \{U_g\}$ , apresentada na Eq. 3.41, resultando em

$$\begin{aligned} \nabla f(A_i^I) = 2 \left[ - [K_g^T]^{-1} \left( \frac{1}{A_j^I} [A]_j^T [k_l^T]_j \{u_l^T\}_i \right) + \right. \\ \left. [K_g^L]^{-1} \left( \frac{\partial\{F_g^L\}}{\partial A_j^I} - \frac{1}{A_j^I} [A]_j^T [k_l^L]_j \{u_l^L\}_i \right) \right]^T (\{U_g^{Tot}\} - \{\Delta X\}) \quad (3.89) \end{aligned}$$

e, mais ainda, usa-se a propriedade  $\{f_l\}_j = [k_l]_j \{u_l\}_j$ , baseada na Eq. 3.55, considerando-se que não há cristalização para o cálculo dos termos em questão (a cristalização será considerada na sequência, no termo apropriado), chegando-se a

$$\nabla f(A_j^I) = 2 \left[ [K_g^T]^{-1} \left( -\frac{1}{A_j^I} [A]_j^T \{f_l^T\}_j \right) + [K_g^L]^{-1} \left( \frac{\partial \{F_g^L\}}{\partial A_j^I} - \frac{1}{A_j^I} [A]_j^T \{f_l^T\}_j \right) \right]^T (\{U_g^{Tot}\} - \{\Delta X\}) \quad (3.90)$$

Para prosseguir com a definição do gradiente, investiga-se o termo  $\partial \{F_g^L\} / \partial A_j^I$ . Para tanto, expande-se  $\{F_g^L\}$ , conforme a Eq. 3.64, chegando-se a

$$\frac{\partial \{F_g^L\}}{\partial A_j^I} = \frac{\partial}{\partial A_j^I} \left( -\sum_{i=n_1}^{n_m} [A]_i^T \{f_l^T\}_i \right) \quad (3.91)$$

Faz-se, então, o caminho inverso ao anterior e usa-se novamente a relação  $\{f_l\}_i = [k_l]_i \{u_l\}_i$ , agora para transformar de forças para deslocamentos, resultando em

$$\frac{\partial}{\partial A_j^I} \{F_g^L\} = \frac{\partial}{\partial A_j^I} \left( -\sum_{i=n_1}^{n_m} [A]_i^T [k_l^T]_i \{u_l^T\}_i \right) \quad (3.92)$$

Também analogamente à demonstração anterior, usa-se a relação  $\{u_l\}_i = [A]_i \{U_g\}$ , outra vez no sentido contrário, para expressar os deslocamentos locais em função dos globais, resultando em

$$\frac{\partial \{F_g^L\}}{\partial A_j^I} = \frac{\partial}{\partial A_j^I} \left( -\sum_{i=n_1}^{n_m} [A]_i^T [k_l^T]_i [A]_i \{U_g^T\} \right) \quad (3.93)$$

e, por fim, usando-se a Eq. 3.25, o termo  $\sum_{i=n_1}^{n_m} [A]_i^T [k_l^T]_i [A]_i$  é compactado, chega-se em

$$\frac{\partial \{F_g^L\}}{\partial A_j^I} = -\frac{\partial}{\partial A_j^I} ([K_g^M] \{U_g^T\}) \quad (3.94)$$

na qual  $[K_g^M]$  é a matriz de rigidez considerando-se apenas as barras de matriz, uma vez que só elas entraram no somatório.

Nesse novo formato, observa-se que só  $\{U_g^T\}$  depende das áreas das barras de inclusão, e, portanto, tem-se que

$$\frac{\partial \{F_g^L\}}{\partial A_j^I} = -[K_g^M] \frac{\partial \{U_g^T\}}{\partial A_j^I} \quad (3.95)$$

Na sequência, usando-se o resultado anteriormente apresentado na Eq. 3.81, lembrando que o vetor de força da etapa de treinamento não depende das áreas das barras de inclusão, tem-se que

$$\frac{\partial\{F_g^L\}}{\partial A_j^I} = [K_g^M][K_g]^{-1} \frac{\partial[K_g^T]}{\partial A_j^I} \{U_g^T\} \quad (3.96)$$

Usando-se a relação dada na Eq. 3.87 para a derivada da matriz de rigidez, essa equação pode ser simplificada ainda mais para

$$\frac{\partial\{F_g^L\}}{\partial A_j^I} = [K_g^M][K_g]^{-1} \left( \frac{1}{A_j^I} [A]_j^T [k_l^T]_j [A]_j \right) \{U_g^T\} \quad (3.97)$$

Por fim, usando-se as relações anteriormente discutidas de maneira análoga à das outras parcelas simplificadas, o termo  $[k_l]_j [A]_j \{U_g\}$  pode ser compactado, resultando em

$$\frac{\partial\{F_g^L\}}{\partial A_j^I} = [K_g^M][K_g^T]^{-1} \left( \frac{1}{A_j^I} [A]_j^T \{f_l^T\}_j \right) \quad (3.98)$$

ou, reordenando os termos, em

$$\frac{\partial\{F_g^L\}}{\partial A_j^I} = \frac{1}{A_j^I} [K_g^M][K_g^T]^{-1} [A]_j^T \{f_l^T\}_j \quad (3.99)$$

Substituindo-se esse resultado na Eq. 3.90, chega-se a

$$\begin{aligned} \nabla f(A_i^I) = 2 \left[ - [K_g^T]^{-1} \left( \frac{1}{A_j^I} [A]_j^T \{f_l^T\}_j \right) + \right. \\ \left. [K_g^L]^{-1} \left( \frac{1}{A_j^I} [K_g^M][K_g^T]^{-1} [A]_j^T \{f_l^T\}_j - \frac{1}{A_j^I} [A]_j^T \{f_l^T\}_j \right) \right]^T \\ \left( \{U_g^{Tot}\} - \{\Delta X\} \right) \quad (3.100) \end{aligned}$$

e, portanto, cada coluna  $j$  do Jacobiano  $[J]$  pode ser escrita como

$$(J)_j = \left[ - [K_g^T]^{-1} \left( \frac{1}{A_j^I} [A]_j^T \{f_l^T\}_j \right) + [K_g^L]^{-1} \left( \frac{1}{A_j^I} [K_g^M] [K_g^T]^{-1} [A]_j^T \{f_l^T\}_j - \frac{1}{A_j^I} [A]_j^T \{f_l^T\}_j \right) \right] \quad (3.101)$$

Reordenando-se a Eq. 3.100, explicitando-se o termo  $-1/A_j^I$ , distribuindo-se o produto da parcela de liberação e modificando a sequência de alguns valores, tem-se

$$\nabla f(A_i^I) = -\frac{2}{A_j^I} \left( [K_g^T]^{-1} [A]_j^T \{f_l^T\}_j + [K_g^L]^{-1} [A]_j^T \{f_l^L\}_j - [K_g^L]^{-1} [K_g^M] [K_g^T]^{-1} [A]_j^T \{f_l^T\}_j \right)^T \left( \{U_g^{Tot}\} - \{\Delta X\} \right) \quad (3.102)$$

Simplificando-se ainda mais, bem como homogeneizando-se a notação utilizada, os vetores coluna  $[A]_j^T \{f_l^T\}_j$  e  $[A]_j^T \{f_l^L\}_j$  são agrupados lado a lado nas matrizes  $[F_l^T]$  e  $[F_l^L]$ . Analogamente, os escalares  $-2/A_j^I$  são agrupados no vetor auxiliar  $\{x\}$ , um vetor coluna. Por fim, o termo  $\{U_g^{Tot}\} - \{\Delta X\}$  fica escrito, compactamente, como  $\{y\}$ , o mesmo sendo uma função dos  $A_j^I$ , e portanto, uma função de  $\{x\}$  (ou seja:  $\{y\} = \{y(\{x\})\}$ ). Dessa forma, tem-se que o gradiente pode ser escrito, em seu novo formato como

$$\nabla f = \{x\} \odot \left[ \left( [K_g^T]^{-1} [F_l^T] + [K_g^L]^{-1} [F_l^L] - [K_g^L]^{-1} [K_g^M] [K_g^T]^{-1} [F_l^T] \right)^T \{y\} \right] \quad (3.103)$$

na qual o operador  $\odot$  é produto de Hadamard que, aplicado à duas matrizes ou vetores de mesma dimensão, resulta também numa matriz ou vetor de mesma dimensão que os de entrada e cujos termos são os produtos dos valores dos termos equivalentes nestes (ou seja:  $(a \odot b)_{ij} = (a)_{ij} * (b)_{ij}$ ).

Tal gradiente pode ser resolvido diretamente através das inversões das matrizes  $[K_g]^T$  e  $[K_g]^L$ , seguidas das multiplicações matriciais necessárias. Tal estratégia, porém, apesar de bastante simples, acarreta em grandes tempos computacionais, visto que o processo de inversão matricial é custoso. Uma alternativa que melhora consideravelmente o desempenho da implementação é resolver tais produtos entre inversas e matrizes através de algum processo de decomposição (por exemplo a decomposição de Cholesky) ou eliminação (por exemplo a eliminação de Gauss) aplicado à matriz a ser invertida, seguido pela retrosubstituição das colunas da matriz a ser multiplicada. Apesar de muito mais

eficiente que calcular a inversa de maneira direta, tal alternativa ainda acarreta em elevados custos, uma vez que é necessário fazer a retrosubstituição em  $m$  vetores, sendo  $m$  o número de variáveis de projeto, para cada inversa presente na formulação. Neste trabalho, adota-se uma terceira estratégia, esta ainda mais eficiente. Para tanto, transpõem-se ambos os lados da Eq. 3.103 e entra-se com a transposta de  $\{y\}$  nos termos entre parênteses, resultando em

$$\nabla f^T = \{x\}^T \odot \left( \{y\}^T [K_g^T]^{-1} [F_l^T] + \{y\}^T [K_g^L]^{-1} [F_l^L] - \{y\}^T [K_g^L]^{-1} [K_g^M] [K_g^T]^{-1} [F_l^T] \right) \quad (3.104)$$

Na sequência, usam-se as relações

$$\{y\}^T [K_g^T]^{-1} = \{\lambda^T\}^T; \quad \{y\}^T [K_g^L]^{-1} = \{\lambda^L\}^T \quad (3.105)$$

nas quais os vetores  $\{\lambda\}$  são variáveis auxiliares. Transpondo-se ambos os lados das duas equações, bem como observando-se que as matrizes de rigidez são simétricas, bem como suas inversas (e, portanto, ambas são iguais às suas transpostas), tem-se que os vetores  $\{\lambda\}$  podem ser explicitados através das equações

$$[K_g^T]^{-1} \{y\} = \{\lambda^T\}; \quad [K_g^L]^{-1} \{y\} = \{\lambda^L\} \quad (3.106)$$

cuja solução advém, mais uma vez, de algum processo de decomposição ou eliminação, seguido por uma retrosubstituição. A última, porém, só precisa ser realizada uma única vez, o que acarreta em um grande aumento do desempenho.

Usando-se esses vetores auxiliares, tem-se que a Eq. 3.104 fica escrita como

$$\nabla f^T = \{x\}^T \odot \left( \{\lambda^T\}^T [F_l^T] + \{\lambda^L\}^T [F_l^L] - \{\lambda^L\}^T [K_g^M] [K_g^T]^{-1} [F_l^T] \right) \quad (3.107)$$

Fica claro, neste formato, que as retrosubstituições foram substituídas por multiplicações entre os vetores  $\{\lambda\}$  e as matrizes  $[F_l]$ . Tal substituição acelera enormemente o cálculo (por volta de 3 ordens de magnitude para alguns exemplos básicos testados durante a implementação, feita no ambiente computação científica MATLAB). Portanto, a inversa da matriz  $[K_g^T]$ , presente no termo final da equação, também é reescrita em termos de um vetor auxiliar. Para tanto, faz-se

$$\{\lambda^L\}^T [K_g^M] [K_g^T]^{-1} = \{\lambda^3\}^T \quad (3.108)$$

Mais uma vez aplicando-se a transposta e considerando-se que as matrizes de rigidez são simétricas, bem como suas inversas, pode-se calcular  $\{\lambda^3\}$  pela equação

$$[K_g^T]^{-1} [K_g^M] \{\lambda^L\} = \{\lambda^3\} \quad (3.109)$$

que pode ser resolvida através de uma outra eliminação/decomposição seguida de retro-substituição, considerando-se que o produto  $[K_g^M] \{\lambda^L\}$  é um vetor conhecido. Substituindo-se  $\{\lambda^3\}^T$  na Eq. 3.107 chega-se, finalmente, à expressão

$$\nabla f^T = \{x\}^T \odot \left( \{\lambda^T\}^T [F_l^T] + \{\lambda^L\}^T [F_l^L] - \{\lambda^3\}^T [F_l^T] \right) \quad (3.110)$$

ou, aplicando-se a transposta em ambos os lados da equação novamente

$$\nabla f = \{x\} \odot \left( [F_l^T]^T \{\lambda^T\} + [F_l^L]^T \{\lambda^L\} - [F_l^T]^T \{\lambda^3\} \right) \quad (3.111)$$

Por fim, considera-se possibilidade de que apenas alguns nós tenham suas posições prescritas, em oposição a todos, ou mesmo que seja prescrito o valor de uma coordenada, mas não da outra, para um ou mais dos nós. Neste caso, inicialmente, para cada coordenada  $i$  não mapeada, basta fazer  $\{U_g^{Tot}\}_i - \{\Delta X\}_i = 0$ , ou seja:  $\{\Delta X\}_i = \{U_g^{Tot}\}_i$ , o que significa que, em cada passo da otimização, o valor prescrito nas coordenadas não mapeadas é considerado exatamente igual ao obtido, independentemente de qual seja, o que resulta numa contribuição nula da coordenada do nó no cálculo do erro.

#### 3.5.4 Hessiana

Outra grandeza de particular interesse em problemas de otimização é a matriz Hessiana da função objetivo com relação às variáveis de projeto. Tal interesse advém do fato de que algumas propriedades da matriz Hessiana servem para a caracterização do problema em questão, bem como da solução encontrada para o mesmo. Em particular, se a Hessiana for positiva-definida (ou seja: atenda à propriedade  $\{x\}^T [H] \{x\} > 0$ , na qual  $\{x\}$  é um vetor qualquer e  $[H]$  é a Hessiana) em todo o domínio das variáveis de projeto, a função objetivo é convexa. No formato proposto para o problema de otimização, no qual só há restrições laterais, isso significaria, caso se confirmasse, que o problema como um todo seria convexo, e que, portanto, qualquer ponto extremo encontrado seria automaticamente um ponto de mínimo. Um problema convexo também é mais simples de ser resolvido e permite a utilização de algoritmos específicos para o mesmo, frequentemente

mais velozes que algoritmos para problemas genéricos. Mesmo que o problema não seja convexo, contudo, a matriz Hessiana ser positiva-definida num ponto extremo encontrado garante ao menos que o ponto em questão é um mínimo local, ainda que não se possa afirmar que tal mínimo local é um mínimo global do problema apenas com essa informação. A última propriedade pode ser provada fazendo-se a expansão da função objetivo por séries de Taylor até o termo quadrático em torno de um ponto  $\{x\}$  candidato a mínimo. Neste caso, tem-se que

$$f(\{x\} + \{\delta x\}) \approx f + \{\delta x\}^T \{g\} + \{\delta x\}^T [H] \{\delta x\} \quad (3.112)$$

na qual  $f$  é o valor da função objetivo em  $x$ , sendo  $\{g\}$  seu gradiente e  $[H]$  sua Hessiana neste ponto, com  $\{\delta x\}$  pequeno.

Se  $\{x\}$  é um ponto extremo de  $f$ , então, por definição, tem-se que  $g$  (e, consequentemente,  $\{\delta x\}^T \{g\}$ ) é nulo. Logo, a Eq. 3.112 é reescrita como

$$\delta f \approx \{\delta x\}^T [H] \{\delta x\} \quad (3.113)$$

na qual  $\delta f = f(\{x\} + \{\delta x\}) - f$  representa a variação em  $f$  com o incremento  $\{\delta x\}$ .

Para  $[H]$  positiva-definida, tem-se que  $\{\delta x\}^T [H] \{\delta x\} > 0$  para qualquer incremento  $\{\delta x\}$  não nulo. Reduzindo-se o tamanho do incremento  $\{\delta x\}$  o suficiente para que os termos de ordem superior possam ser ignorados, isso significa que, na vizinhança do ponto  $\{x\}$ , a função objetivo só aumenta. O mesmo é, portanto, um ponto de mínimo. Mais ainda, sendo  $[H]$  positiva-definida para qualquer  $\{x\}$ , esse mínimo é necessariamente um mínimo global de  $f$ , bem como seu único ponto extremo.

Analogamente, para  $[H]$  negativa-definida em um ponto de extremo, o inverso é válido e pode-se afirmar que tal ponto é um máximo local de  $f$ , uma vez que todas as direções ao redor de  $\{x\}$  aumentam o valor da mesma. Por fim, sendo  $[H]$  indefinida, seguindo o mesmo raciocínio, pode-se afirmar que  $\{x\}$  é um ponto de sela, uma vez que, ao redor do mesmo, certas direções fazem a função objetivo aumentar de valor, enquanto outras a fazem reduzir.

É importante ressaltar que o teste da Hessiana é inconclusivo quando a mesma é semi-definida (seja positiva ou negativa). Ser semi-definida significa que, para certas direções, o termo  $\{\delta x\}^T [H] \{\delta x\}$  é nulo. Neste caso, os termos de ordem superior não podem ser desprezados, mesmo quando  $\{\delta x\}$  é muito pequeno, o que significa que a expansão em segunda ordem é insuficiente para caracterizar o ponto.

Uma vez que a importância da mesma foi apresentada, para iniciar a derivação da

Hessiana, lembra-se que o gradiente da função objetivo pode ser escrito, em uma notação mais enxuta, para abreviar as equações de grande dimensão que seguem, como

$$D_j f = 2D_j \{U_g^{Tot}\}^T \{y\} \quad (3.114)$$

na qual o operador  $D_j$  denota a derivada  $\partial/\partial A_j^I$

Derivando-se mais uma vez a função objetivo, tem-se

$$D_{jk} f = D_k \left( 2D_j \{U_g^{Tot}\}^T \{y\} \right) \quad (3.115)$$

na qual, analogamente, o operador  $D_k$  denota a derivada  $\partial/\partial A_k^I$

Entrando-se com a derivada e aplicando-se a regra da cadeia, tem-se que

$$D_{jk} f = 2D_k \{y\}^T D_j \{U_g^{Tot}\} + 2D_{jk} \{U_g^{Tot}\} \{y\} \quad (3.116)$$

Usando-se a relação  $D_k \{y\}^T = D_k \{U_g^{Tot}\}^T$ , anteriormente apresentada, tem-se a equação

$$D_{jk} f = 2D_k \{U_g^{Tot}\}^T D_j \{U_g^{Tot}\} + 2D_{jk} \{U_g^{Tot}\} \{y\} \quad (3.117)$$

Por agora, foca-se no termo  $D_{jk} \{U_g^{Tot}\}$ , uma vez que os outros são conhecidos da demonstração do gradiente. Para explicitar o valor do mesmo, toma-se a relação apresentada na Eq. 3.80 e deriva-se ambos os lados novamente, resultando em

$$D_k \left( D_j [K_g] \{U_g\} + [K_g] D_j \{U_g\} \right) = D_k \left( D_j \{F_g\} \right) \quad (3.118)$$

Expandindo o termo à esquerda, tem-se

$$D_{jk} [K_g] \{U_g\} + D_j [K_g] D_k \{U_g\} + D_k [K_g] D_j \{U_g\} + [K_g] D_{jk} \{U_g\} = D_{jk} \{F_g\} \quad (3.119)$$

Simplificando e isolando  $D_{jk} \{U_g\}$ , tem-se

$$D_{jk} \{U_g\} = [K_g]^{-1} \left( D_{jk} \{F_g\} - D_{jk} [K_g] \{U_g\} - D_j [K_g] D_k \{U_g\} - D_k [K_g] D_j \{U_g\} \right) \quad (3.120)$$

Particularizando este resultado para as etapas de treinamento e liberação, tem-se que

$$\begin{aligned}
D_{jk}\{U_g^{Tot}\} = & [K_g^T]^{-1} \left( D_{jk}\{F_g^T\} - D_{jk}[K_g^T]\{U_g^T\} \right. \\
& \left. - D_j[K_g^T]D_k\{U_g^T\} - D_k[K_g^T]D_j\{U_g^T\} \right) + [K_g^L]^{-1} \left( D_{jk}\{F_g^L\} \right. \\
& \left. - D_{jk}[K_g^L]\{U_g^L\} - D_j[K_g^L]D_k\{U_g^L\} - D_k[K_g^L]D_j\{U_g^L\} \right) \quad (3.121)
\end{aligned}$$

Uma vez que os vetores de força da etapa de treinamento não dependem das áreas das barras de inclusão, tanto suas primeiras quanto segundas derivadas são nulas. Mais ainda, tanto as matrizes de liberação quanto de treinamento, em cada uma de suas entradas, carregam uma dependência ou linear ou não existente com relação à área de qualquer barra, o que torna suas segundas derivadas também nulas. Com isso, a tem-se que

$$\begin{aligned}
D_{jk}\{U_g^{Tot}\} = & [K_g^T]^{-1} \left( - D_j[K_g^T]D_k\{U_g^T\} - D_k[K_g^T]D_j\{U_g^T\} \right) \\
& + [K_g^L]^{-1} \left( D_{jk}\{F_g^L\} - D_j[K_g^L]D_k\{U_g^L\} - D_k[K_g^L]D_j\{U_g^L\} \right) \quad (3.122)
\end{aligned}$$

Na sequência, usando-se o valor explicitado na Eq. 3.81 para as derivadas dos deslocamentos, tanto da etapa de liberação quanto de treinamento, tem-se que

$$\begin{aligned}
D_{jk}\{U_g^{Tot}\} = & [K_g^T]^{-1} \left[ - D_j[K_g^T][K_g^T]^{-1} \left( D_k\{F_g^T\} - D_k[K_g^T]\{U_g^T\} \right) \right. \\
& \left. - D_k[K_g^T][K_g^T]^{-1} \left( D_j\{F_g^T\} - D_j[K_g^T]\{U_g^T\} \right) \right] \\
& + [K_g^L]^{-1} \left[ D_{jk}\{F_g^L\} - D_j[K_g^L][K_g^L]^{-1} \left( D_k\{F_g^L\} - D_k[K_g^L]\{U_g^L\} \right) \right. \\
& \left. - D_k[K_g^L][K_g^L]^{-1} \left( D_j\{F_g^L\} - D_j[K_g^L]\{U_g^L\} \right) \right] \quad (3.123)
\end{aligned}$$

Mais uma vez, sabendo que as derivadas das forças da etapa de treinamento são nulas, tal expressão simplifica para

$$\begin{aligned}
D_{jk}\{U_g^{Tot}\} &= [K_g^T]^{-1} \left[ D_j[K_g^T][K_g^T]^{-1} D_k[K_g^T]\{U_g^T\} \right. \\
&\quad \left. + D_k[K_g^T][K_g^T]^{-1} D_j[K_g^T]\{U_g^T\} \right] + [K_g^L]^{-1} \left[ D_{jk}\{F_g^L\} \right. \\
&\quad \left. - D_j[K_g^L][K_g^L]^{-1} \left( D_k\{F_g^L\} - D_k[K_g^L]\{U_g^L\} \right) \right. \\
&\quad \left. - D_k[K_g^L][K_g^L]^{-1} \left( D_j\{F_g^L\} - D_j[K_g^L]\{U_g^L\} \right) \right] \quad (3.124)
\end{aligned}$$

Os resultados para os termos  $D[K_g^T]$ ,  $D[K_g^L]$  e  $D\{F_g^L\}$  são conhecidos da demonstração anterior, referente ao gradiente. Resta, portanto, investigar o termo  $D_{jk}\{F_g^L\}$ . Usando o resultado apresentado na Eq. 3.96, tem-se que

$$D_{jk}\{F_g^L\} = D_k \left( D_j\{F_g^L\} \right) = D_k \left( [K_g^M][K_g^T]^{-1} D_j[K_g^T]\{U_g^T\} \right) \quad (3.125)$$

Usando a regra da cadeia, tem-se que

$$\begin{aligned}
D_{jk}\{F_g^L\} &= D_k[K_g^M][K_g^T]^{-1} D_j[K_g^T]\{U_g^T\} + [K_g^M] D_k[K_g]^{-1} D_j[K_g^T]\{U_g^T\} \\
&\quad + [K_g^M][K_g^T]^{-1} D_{jk}[K_g^T]\{U_g^T\} + [K_g^M][K_g^T]^{-1} D_j[K_g^T] D_k\{U_g^T\} \quad (3.126)
\end{aligned}$$

Uma vez que  $[K_g^M]$  não depende das áreas de inclusão e que  $D_{jk}[K_g] = 0$ , como discutido anteriormente, tem-se, na sequência, que

$$D_{jk}\{F_g^L\} = [K_g^M] D_k[K_g^T]^{-1} D_j[K_g^T]\{U_g^T\} + [K_g^M][K_g^T]^{-1} D_j[K_g^T] D_k\{U_g^T\} \quad (3.127)$$

Dentre os termos presentes nessa equação, o único que não se conhece explicitamente é a derivada  $D[K_g^T]^{-1}$  (tanto com relação a  $k$  quanto a  $j$ ). Para explicitá-lo, parte-se da propriedade

$$[K_g^T]^{-1}[K_g^T] = [I] \quad (3.128)$$

na qual  $[I]$  é a matriz identidade. Derivando-se ambos os lados da equação com o operador  $D_k$ , respeitando-se a ordem apropriada da regra da cadeia no produto matricial e observando que  $[I]$  não depende das áreas das barras, tem-se que

$$D_k[K_g^T]^{-1}[K_g^T] + [K_g^T]^{-1}D_k[K_g^T] = [0] \quad (3.129)$$

no qual  $[0]$  é uma matriz nula. Reordenando essa expressão, tem-se que

$$D_k[K_g^T]^{-1}[K_g^T] = -[K_g^T]^{-1}D_k[K_g^T] \quad (3.130)$$

Multiplicando ambos os lados pela inversa de  $[K_g^T]$ , chega-se a

$$D_k[K_g^T]^{-1}[K_g^T][K_g^T]^{-1} = -[K_g^T]^{-1}D_k[K_g^T][K_g^T]^{-1} \quad (3.131)$$

e, por fim, como  $[K_g^T][K_g^T]^{-1} = [I]$  e  $D_k[K_g^T]^{-1}[I] = D_k[K_g^T]^{-1}$ , obtém-se

$$D_k[K_g^T]^{-1} = -[K_g^T]^{-1}D_k[K_g^T][K_g^T]^{-1} \quad (3.132)$$

Aplicando esse resultando na Eq. 3.127, tem-se que

$$\begin{aligned} D_{jk}\{F_g^L\} &= -[K_g^M][K_g^T]^{-1}D_k[K_g^T][K_g^T]^{-1}D_j[K_g^T]\{U_g^T\} \\ &\quad + [K_g^M][K_g^T]^{-1}D_j[K_g^T]D_k\{U_g^T\} \end{aligned} \quad (3.133)$$

Substituindo esse resultado na Eq. 3.124, por sua vez, encontra-se

$$\begin{aligned} D_{jk}\{U_g^{Tot}\} &= [K_g^T]^{-1} \left[ D_j[K_g^T][K_g^T]^{-1}D_k[K_g^T]\{U_g^T\} \right. \\ &\quad \left. + D_k[K_g^T][K_g^T]^{-1}D_j[K_g^T]\{U_g^T\} \right] \\ &+ [K_g^L]^{-1} \left[ -[K_g^M][K_g^T]^{-1}D_k[K_g^T][K_g^T]^{-1}D_j[K_g^T]\{U_g^T\} \right. \\ &\quad \left. + [K_g^M][K_g^T]^{-1}D_j[K_g^T]D_k\{U_g^T\} \right. \\ &\quad \left. - D_j[K_g^L][K_g^L]^{-1} \left( D_k\{F_g^L\} - D_k[K_g^L]\{U_g^L\} \right) \right. \\ &\quad \left. - D_k[K_g^L][K_g^L]^{-1} \left( D_j\{F_g^L\} - D_j[K_g^L]\{U_g^L\} \right) \right] \end{aligned} \quad (3.134)$$

Distribuindo-se os termos entre parênteses, a expressão fica escrita como

$$\begin{aligned}
D_{jk}\{U_g^{Tot}\} = & [K_g^T]^{-1} \left( D_j[K_g^T][K_g^T]^{-1} D_k[K_g^T]\{U_g^T\} \right. \\
& \left. + D_k[K_g^T][K_g^T]^{-1} D_j[K_g^T]\{U_g^T\} \right) + \\
& [K_g^L]^{-1} \left( - [K_g^M][K_g^T]^{-1} D_k[K_g^T][K_g^T]^{-1} D_j[K_g^T]\{U_g^T\} \right. \\
& + [K_g^M][K_g^T]^{-1} D_j[K_g^T] D_k\{U_g^T\} - D_j[K_g^L][K_g^L]^{-1} D_k\{F_g^L\} \\
& + D_j[K_g^L][K_g^L]^{-1} D_k[K_g^L]\{U_g^L\} - D_k[K_g^L][K_g^L]^{-1} D_j\{F_g^L\} \\
& \left. + D_k[K_g^L][K_g^L]^{-1} D_j[K_g^L]\{U_g^L\} \right) \quad (3.135)
\end{aligned}$$

Agora que todos os termos são conhecidos, resgata-se, na sequência, as propriedades apresentadas na demonstração do gradiente, direta ou indiretamente, de que

$$D_j[K_g]\{U_g\} = \frac{1}{A_j^I} [A]_j^T \{f_l\}_j \quad (3.136a)$$

$$D_j\{F_g^L\} = \frac{1}{A_j^I} [K_g^M][K_g^T]^{-1} [A]_j^T \{f_l^T\}_j \quad (3.136b)$$

$$D_j\{U_g\} = -\frac{1}{A_j^I} [K_g]^{-1} [A]_j^T \{f_l\}_j \quad (3.136c)$$

$$D_j[K_g] = \frac{1}{A_j^I} [A]_j^T [k_l]_j [A]_j \quad (3.136d)$$

bem como seus equivalentes para as derivadas no indexador  $k$ . Aplicando esses resultados à Eq. 3.135, tem-se que

$$\begin{aligned}
D_{jk}\{U_g^{Tot}\} = & [K_g^T]^{-1} \left[ + \left( \frac{1}{A_j^I} [A]_j^T [k_l^T]_j [A]_j \right) [K_g^T]^{-1} \left( \frac{1}{A_k^I} [A]_k^T \{f_l^T\}_k \right) \right. \\
& \left. + \left( \frac{1}{A_k^I} [A]_k^T [k_l^T]_k [A]_k \right) [K_g^T]^{-1} \left( \frac{1}{A_j^I} [A]_j^T \{f_l^T\}_j \right) \right] \\
& + [K_g^L]^{-1} \left[ - [K_g^M] [K_g^T]^{-1} \left( \frac{1}{A_k^I} [A]_k^T [k_l^T]_k [A]_k \right) [K_g^T]^{-1} \left( \frac{1}{A_j^I} [A]_j^T \{f_l^T\}_j \right) \right. \\
& + [K_g^M] [K_g^T]^{-1} \left( \frac{1}{A_j^I} [A]_j^T [k_l^T]_j [A]_j \right) \left( - \frac{1}{A_k^I} [K_g^T]^{-1} [A]_k^T \{f_l^T\}_k \right) \\
& - \left( \frac{1}{A_j^I} [A]_j^T [k_l^L]_j [A]_j \right) [K_g^L]^{-1} \left( \frac{1}{A_k^I} [K_g^M] [K_g^T]^{-1} [A]_k^T \{f_l^T\}_k \right) \\
& + \left( \frac{1}{A_j^I} [A]_j^T [k_l^L]_j [A]_j \right) [K_g^L]^{-1} \left( \frac{1}{A_k^I} [A]_k^T \{f_l^L\}_k \right) \\
& - \left( \frac{1}{A_k^I} [A]_k^T [k_l^L]_k [A]_k \right) [K_g^L]^{-1} \left( \frac{1}{A_j^I} [K_g^M] [K_g^T]^{-1} [A]_j^T \{f_l^T\}_j \right) \\
& \left. + \left( \frac{1}{A_k^I} [A]_k^T [k_l^L]_k [A]_k \right) [K_g^L]^{-1} \left( \frac{1}{A_j^I} [A]_j^T \{f_l^L\}_j \right) \right] \quad (3.137)
\end{aligned}$$

Isolando os termos  $(1/A_k^I)(1/A_j^I)$  que se repetem ao longo das parcelas, obtém-se

$$\begin{aligned}
D_{jk}\{U_g^{Tot}\} = & \frac{2}{A_k^I A_j^I} \left[ [K_g^T]^{-1} \left( [A]_j^T [k_l^T]_j [A]_j [K_g^T]^{-1} [A]_k^T \{f_l^T\}_k \right) \right. \\
& \left. + [A]_k^T [k_l^T]_k [A]_k [K_g^T]^{-1} [A]_j^T \{f_l^T\}_j \right) \\
& + [K_g^L]^{-1} \left( - [K_g^M] [K_g^T]^{-1} [A]_k^T [k_l^T]_k [A]_k [K_g^T]^{-1} [A]_j^T \{f_l^T\}_j \right. \\
& - [K_g^M] [K_g^T]^{-1} [A]_j^T [k_l^T]_j [A]_j [K_g^T]^{-1} [A]_k^T \{f_l^T\}_k \\
& - [A]_j^T [k_l^L]_j [A]_j [K_g^L]^{-1} [K_g^M] [K_g^T]^{-1} [A]_k^T \{f_l^T\}_k \\
& + [A]_j^T [k_l^L]_j [A]_j [K_g^L]^{-1} [A]_k^T \{f_l^L\}_k \\
& - [A]_k^T [k_l^L]_k [A]_k [K_g^L]^{-1} [K_g^M] [K_g^T]^{-1} [A]_j^T \{f_l^T\}_j \\
& \left. + [A]_k^T [k_l^L]_k [A]_k [K_g^L]^{-1} [A]_j^T \{f_l^L\}_j \right) \right] \quad (3.138)
\end{aligned}$$

Por fim, distribuindo as multiplicações e reordenando os termos, chega-se a

$$\begin{aligned}
D_{jk}\{U_g^{Tot}\} = \frac{2}{A_k^T A_j^T} & \left( [K_g^T]^{-1} [A]_j^T [k_l^T]_j [A]_j [K_g^T]^{-1} [A]_k^T \{f_l^T\}_k \right. \\
& + [K_g^T]^{-1} [A]_k^T [k_l^T]_k [A]_k [K_g^T]^{-1} [A]_j^T \{f_l^T\}_j \\
& + [K_g^L]^{-1} [A]_j^T [k_l^L]_j [A]_j [K_g^L]^{-1} [A]_k^T \{f_l^L\}_k \\
& + [K_g^L]^{-1} [A]_k^T [k_l^L]_k [A]_k [K_g^L]^{-1} [A]_j^T \{f_l^L\}_j \\
& - [K_g^L]^{-1} [K_g^M] [K_g^T]^{-1} [A]_j^T [k_l^T]_j [A]_j [K_g^T]^{-1} [A]_k^T \{f_l^T\}_k \\
& - [K_g^L]^{-1} [K_g^M] [K_g^T]^{-1} [A]_k^T [k_l^T]_k [A]_k [K_g^T]^{-1} [A]_j^T \{f_l^T\}_j \\
& - [K_g^L]^{-1} [A]_j^T [k_l^L]_j [A]_j [K_g^L]^{-1} [K_g^M] [K_g^T]^{-1} [A]_k^T \{f_l^T\}_k \\
& \left. - [K_g^L]^{-1} [A]_k^T [k_l^L]_k [A]_k [K_g^L]^{-1} [K_g^M] [K_g^T]^{-1} [A]_j^T \{f_l^T\}_j \right) \quad (3.139)
\end{aligned}$$

Usando os lambdas calculados anteriormente, pode-se simplificar a expressão anterior como

$$\begin{aligned}
D_{jk}\{U_g^{Tot}\} = \frac{2}{A_k^T A_j^T} & \left( \{\lambda^T\}^T [A]_j^T [k_l^T]_j [A]_j [K_g^T]^{-1} [F_l^T]_k \right. \\
& + \{\lambda^T\}^T [A]_k^T [k_l^T]_k [A]_k [K_g^T]^{-1} [F_l^T]_j \\
& + \{\lambda^L\}^T [A]_j^T [k_l^L]_j [A]_j [K_g^L]^{-1} [F_l^L]_k \\
& + \{\lambda^L\}^T [A]_k^T [k_l^L]_k [A]_k [K_g^L]^{-1} [F_l^L]_j \\
& - \{\lambda^3\}^T [A]_j^T [k_l^T]_j [A]_j [K_g^T]^{-1} [F_l^T]_k \\
& - \{\lambda^3\}^T [A]_k^T [k_l^T]_k [A]_k [K_g^T]^{-1} [F_l^T]_j \\
& - \{\lambda^L\}^T [A]_j^T [k_l^L]_j [A]_j [K_g^L]^{-1} [K_g^M] [K_g^T]^{-1} [F_l^T]_k \\
& \left. - \{\lambda^L\}^T [A]_k^T [k_l^L]_k [A]_k [K_g^L]^{-1} [K_g^M] [K_g^T]^{-1} [F_l^T]_j \right) \quad (3.140)
\end{aligned}$$

Reordenando a expressão com base nos coeficientes, tem-se

$$\begin{aligned}
D_{jk}\{U_g^{Tot}\} = \frac{2}{A_k^T A_j^T} & \left( \{\lambda^T\}^T [A]_j^T [k_l^T]_j [A]_j [K_g^T]^{-1} [F_l^T]_k \right. \\
& + \{\lambda^L\}^T [A]_j^T [k_l^L]_j [A]_j [K_g^L]^{-1} [F_l^L]_k \\
& - \{\lambda^3\}^T [A]_j^T [k_l^T]_j [A]_j [K_g^T]^{-1} [F_l^T]_k \\
& - \{\lambda^L\}^T [A]_j^T [k_l^L]_j [A]_j [K_g^L]^{-1} [K_g^M] [K_g^T]^{-1} [F_l^T]_k \\
& + \{\lambda^T\}^T [A]_k^T [k_l^T]_k [A]_k [K_g^T]^{-1} [F_l^T]_j \\
& + \{\lambda^L\}^T [A]_k^T [k_l^L]_k [A]_k [K_g^L]^{-1} [F_l^L]_j \\
& - \{\lambda^3\}^T [A]_k^T [k_l^T]_k [A]_k [K_g^T]^{-1} [F_l^T]_j \\
& \left. - \{\lambda^L\}^T [A]_k^T [k_l^L]_k [A]_k [K_g^L]^{-1} [K_g^M] [K_g^T]^{-1} [F_l^T]_j \right) \quad (3.141)
\end{aligned}$$

Na sequência, para simplificar os cálculos e as expressões, escrevem-se as relações

$$[\Lambda^T] = [K_g^T]^{-1} [F_l^T] \quad (3.142a)$$

$$[\Lambda^L] = [K_g^L]^{-1} [F_l^L] \quad (3.142b)$$

$$[\Lambda^3] = [K_g^L]^{-1} [K_g^M] [K_g^T]^{-1} [F_l^T] \quad (3.142c)$$

$$[A]_j^T [k_l^T]_j [A]_j = [I_n]_j^T [R]_j^T [k_l^T]_j [R]_j [I_n]_j = [I_n]_j^T [k_g^T]_j [I_n]_j \quad (3.142d)$$

resultando em

$$\begin{aligned}
D_{jk}\{U_g^{Tot}\} = \frac{2}{A_k^T A_j^T} & \left( \{\lambda^T\}^T [I_n]_j^T [k_g^T]_j [I_n]_j [\Lambda^T]_k \right. \\
& + \{\lambda^L\}^T [I_n]_j^T [k_g^L]_j [I_n]_j [\Lambda^L]_k - \{\lambda^3\}^T [I_n]_j^T [k_g^T]_j [I_n]_j [\Lambda^T]_k \\
& - \{\lambda^L\}^T [I_n]_j^T [k_g^L]_j [I_n]_j [\Lambda^3]_k + \{\lambda^T\}^T [I_n]_k^T [k_g^T]_k [I_n]_k [\Lambda^T]_j \\
& + \{\lambda^L\}^T [I_n]_k^T [k_g^L]_k [I_n]_k [\Lambda^L]_j - \{\lambda^3\}^T [I_n]_k^T [k_g^T]_k [I_n]_k [\Lambda^T]_j \\
& \left. - \{\lambda^L\}^T [I_n]_k^T [k_g^L]_k [I_n]_k [\Lambda^3]_j \right) \quad (3.143)
\end{aligned}$$

chegando-se, finalmente, a um formato final para o  $D_{jk}\{U_g^{Tot}\}$ .

Na sequência, por questão de abreviação, escreve-se:

$$\begin{aligned}
(H_2)_{jk} &= D_{jk}\{U_g^{Tot}\}\{y\} = \\
&\frac{2}{A_k^T A_j^T} \left( \{\lambda^T\}^T [I_n]_j^T [k_g^T]_j [I_n]_j [\Lambda^T]_k + \{\lambda^L\}^T [I_n]_j^T [k_g^L]_j [I_n]_j [\Lambda^L]_k \right. \\
&\quad - \{\lambda^3\}^T [I_n]_j^T [k_g^T]_j [I_n]_j [\Lambda^T]_k - \{\lambda^L\}^T [I_n]_j^T [k_g^L]_j [I_n]_j [\Lambda^3]_k \\
&\quad + \{\lambda^T\}^T [I_n]_k^T [k_g^T]_k [I_n]_k [\Lambda^T]_j + \{\lambda^L\}^T [I_n]_k^T [k_g^L]_k [I_n]_k [\Lambda^L]_j \\
&\quad \left. - \{\lambda^3\}^T [I_n]_k^T [k_g^T]_k [I_n]_k [\Lambda^T]_j - \{\lambda^L\}^T [I_n]_k^T [k_g^L]_k [I_n]_k [\Lambda^3]_j \right) \{y\} \quad (3.144)
\end{aligned}$$

e, ainda,

$$(X)_{jk} = \frac{2}{A_k^T A_j^T} \quad (3.145)$$

logo

$$D_{jk}f = 2D_k\{U_g^{Tot}\}^T D_j\{U_g^{Tot}\} + (X)_{jk}(H_2)_{jk} \quad (3.146)$$

ou, ainda,

$$[H] = 2[J]^T[J] + [X] \odot [H_2] \quad (3.147)$$

na qual  $[H]$  é, finalmente, a Hessiana desejada, sendo  $[J]$  a matriz Jacobiana apresentada na demonstração do gradiente e  $\odot$  o produto de Hadamard. Ainda que, neste formato, todos os termos sejam conhecidos, faz-se ainda mais uma simplificação na Hessiana. Para tanto, o valor da Jacobiana é resgatado da Eq. 3.101 e reescrito com os termos de área isolados, resultando em

$$\begin{aligned}
D_j\{U_g^{Tot}\} &= -\frac{1}{A_j^T} \left( [K_g^T]^{-1} [A]_j^T \{f_l^T\}_j + [K_g^L]^{-1} [A]_j^T \{f_l^L\}_j \right. \\
&\quad \left. - [K_g^L]^{-1} [K_g^M] [K_g^T]^{-1} [A]_j^T \{f_l^T\}_j \right) \quad (3.148)
\end{aligned}$$

Com isso, o termo  $2D_k\{U_g^{Tot}\}^T D_j\{U_g^{Tot}\}$  pode ser escrito como

$$\begin{aligned}
2D_k\{U_g^{Tot}\}^T D_j\{U_g^{Tot}\} = & \\
& - \frac{2}{A_j^T A_k^T} \left( [K_g^T]^{-1} [A]_k^T \{f_l^T\}_k + [K_g^L]^{-1} [A]_k^T \{f_l^L\}_k \right. \\
& - [K_g^L]^{-1} [K_g^M] [K_g^T]^{-1} [A]_k^T \{f_l^T\}_k \left. \right)^T \left( [K_g^T]^{-1} [A]_j^T \{f_l^T\}_j \right. \\
& \left. + [K_g^L]^{-1} [A]_j^T \{f_l^L\}_j - [K_g^L]^{-1} [K_g^M] [K_g^T]^{-1} [A]_j^T \{f_l^T\}_j \right) \quad (3.149)
\end{aligned}$$

Usando a matriz  $[X]$  apresentada a pouco, bem como agrupando os termos  $[A]_i^T \{f_l\}_i$  nas matrizes  $[F]$  descritas na demonstração do gradiente, pode-se escrever, em formato matricial

$$\begin{aligned}
2D\{U_g^{Tot}\}^T D\{U_g^{Tot}\} = & \\
& - [X] \odot \left( [K_g^T]^{-1} [F_l^T] + [K_g^L]^{-1} [f_l^L] - [K_g^L]^{-1} [K_g^M] [K_g^T]^{-1} [F_l^T] \right)^T \\
& \left( [K_g^T]^{-1} [F_l^T] + [K_g^L]^{-1} [f_l^L] - [K_g^L]^{-1} [K_g^M] [K_g^T]^{-1} [F_l^T] \right) \quad (3.150)
\end{aligned}$$

Por questão de abreviação, escreve-se

$$\begin{aligned}
[H_2] = - \left( [K_g^T]^{-1} [F_l^T] + [K_g^L]^{-1} [f_l^L] - [K_g^L]^{-1} [K_g^M] [K_g^T]^{-1} [F_l^T] \right)^T \\
\left( [K_g^T]^{-1} [F_l^T] + [K_g^L]^{-1} [f_l^L] - [K_g^L]^{-1} [K_g^M] [K_g^T]^{-1} [F_l^T] \right) \quad (3.151)
\end{aligned}$$

e então a Hessiana, em seu formato final, é explicitada como

$$[H] = [X] \odot \left( [H_1] + [H_2] \right) \quad (3.152)$$

com  $[H_1]$  representando os termos cruzadas, cujas derivadas são de primeira ordem, e  $[H_2]$  representando os termos quadráticos, muito mais custosos de calcular.

Novamente, pode-se desejar mapear apenas alguns dos nós da malha, e não todos. Assim como no gradiente, a correção para tanto é direta. Em  $[H_2]$ , basta fazer com que o termo em  $\{y\}$  associado ao nó seja nulo, igual ao caso do gradiente. Para o termo  $[H_1]$ , por sua vez, a correção pode ser feita escrevendo-se o mesmo como

$$[H_2] = -[h_2]^T [h_2] \quad (3.153)$$

na qual

$$[h_2] = [K_g^T]^{-1}[F_l^T] + [K_g^L]^{-1}[f_l^L] - [K_g^L]^{-1}[K_g^M][K_g^T]^{-1}[F_l^T] \quad (3.154)$$

e fazendo com que as linhas de  $[h_2]$  correspondentes aos nós não mapeados sejam nulas.

Dada essa formulação, o processo de solução consiste, portanto, em tomar termos das colunas das matrizes  $[\Lambda]$ , termos estes selecionados pelas matrizes de indexação  $[I_n]$ ; multiplicar o vetor resultante pelas matrizes de rigidez dos elementos no sistema global  $[k_g]$ ; indexar novamente o vetor resultante usando a transposta da matriz de indexação e fazer a multiplicação (equivalente a um produto escalar) com os vetores  $\{\lambda\}$  para cada termo não simétrico da matriz  $[H_1]$ , sendo os termos simétricos calculados diretamente (usando a simetria). Tal cálculo é computacionalmente custoso, requerendo dois *loops*, um para cada um dos índices ( $i$  e  $j$ ), bem como usa série de operações matriciais em cada passo desses *loops*, sendo esse o gargalo da implementação. De fato, quando os tempos computacionais são comparados, o cálculo da matriz  $[H_1]$  pode ser assumido insignificante em comparação com  $[H_2]$ , especialmente levando-se em consideração que os termos da mesma podem ser reaproveitados do gradiente. Contudo, para o problema em questão, tal procedimento ainda é ordens de magnitude mais eficiente que calcular a Hessiana através de diferenças finitas sob o gradiente da função objetivo para cada coluna da Hessiana (e ainda outras ordens de magnitude mais eficiente do que usar diferenças finitas duas vezes por termo da Hessiana sob a função objetivo). Por fim, vale salientar que parte do custo excessivo da Hessiana é causado pela natureza interpretada do *MATLAB*, que é largamente mais eficiente para códigos vetorizados e que usem operações matriciais do que na solução de problemas de álgebra linear de maneira indicial, como é o caso para os termos quadráticos. Em outras linguagens, particularmente as compiladas, a diferença entre o custo dos dois termos pode ser significativamente menor.

### 3.6 Algoritmo de Otimização

Para a solução do problema de otimização proposto, vários algoritmos descritos na literatura são testados. Os mesmos são apresentados dentro das subseções a seguir.

### 3.6.1 Método do Gradiente

Inicialmente, por se tratar de uma alternativa extremamente simples e de primeira ordem, investiga-se o Método do Gradiente, que tem como equação governante

$$\{\Delta A^I\} = -\alpha\{g\} \quad (3.155)$$

na qual  $\{\Delta A^I\}$  é o vetor de incrementos das áreas das barras de inclusão, sendo  $\{g\}$  o gradiente, direção de maior crescimento da função (e sendo  $-\{g\}$ , portanto, a direção de maior decréscimo) e  $\alpha$  uma variável de controle do tamanho do passo.

Várias estratégias existem na literatura para a determinação do valor de  $\alpha$ , que, se escolhido apropriadamente, garante a convergência do método, que é bastante estável. Uma estratégia possível é a de busca linear, na qual diversos valores de  $\alpha$  são testados e escolhe-se um que garanta ou simplesmente uma redução da função objetivo ou o atendimento simultâneo de outros critérios, como de curvatura, criados de modo a evitar passos muito pequenos ou muito grandes (ARORA, 2011; NOCEDAL; WRIGHT, 2006). Tal estratégia mostra-se, para o problema proposta, bastante ineficiente, uma vez que exige avaliar a função objetivo, composta com base na solução de dois sistemas estruturais que podem ter grande número de elementos e graus de liberdade, um elevado número de vezes. Mais ainda, como usa apenas informações associadas ao gradiente, o mesmo apresenta convergência extremamente lenta, justificada, acredita o autor, pela presença de vales estreitos na função objetivo, que são percorridos lentamente e de maneira oscilatória. Mesmo quando o problema tratado é simples, o método do gradiente converge, na melhor das hipóteses, linearmente, o que acarreta em um elevado número de iterações. Outras estratégias, que envolvem usar o gradiente para aproximar informações de ordem mais alta, também são testados com pouco sucesso. Tais estratégias, bem como mais considerações a respeito do Método do Gradiente, podem ser obtidas em Arora (2011) e Nocedal e Wright (2006).

### 3.6.2 Método de Newton

Na sequência, busca-se acelerar a convergência investigando um método de segunda ordem, nominalmente o Método de Newton. Nele, a atualização do passo da iteração é dado resolvendo-se a equação

$$[H]\{\Delta A^I\} = -\{g\} \quad (3.156)$$

na qual  $[H]$  é a matriz Hessiana anteriormente descrita em sua totalidade.

Tal método, contudo, apresenta três grandes problemas: o primeiro é que, devido

ao termo que contém as derivadas de segunda ordem, o cálculo da matriz Hessiana em sua forma completa é extremamente custoso. Para alguns dos casos analisados, por exemplo, o mesmo consome mais de 90% do esforço computacional total, sendo, portanto, mais caro que a solução dos sistemas lineares de treinamento e liberação, o cálculo do gradiente, o cálculo da contribuição das derivadas cruzadas e a solução do sistema linear do passo somadas, e ainda elevando o tempo de solução em uma ordem de grandeza. O segundo problema é que, para certos casos e em certas iterações, o sistema linear apresentado é mal-condicionado ou a matriz Hessiana é não-inversível, necessitando de um termo de amortecimento que garanta solução. Por fim, o algoritmo é instável, de forma que, para ter garantia de convergência, o mesmo necessita de uma variável de controle do tamanho do passo, tal que sua equação fica escrita como

$$\{\Delta A^I\} = -\gamma[H]\setminus\{g\} \quad (3.157)$$

na qual  $\gamma$  precisa ser escolhido de forma a garantir convergência e  $[H]\setminus\{g\}$  representa a solução do sistema linear  $[H]\{\Delta A^I\} = \{g\}$  por algum método que não necessite da formação da inversa explícita de  $[H]$ , que é uma operação custosa.

Mesmo sanados todos estes problemas, contudo, o Método de Newton ainda tem outra grande limitação: para que  $[H]\setminus\{g\}$  defina uma direção de decréscimo de  $f$ , a Hessiana tem que ser positiva-definida para todos os pontos em que for calculada. Avaliações da Hessiana para casos de análise mostram, contudo, que a mesma é, para a maior parte dos casos, indefinida, e, portanto, o Método não tem garantia de convergência, mesmo com o ajuste do parâmetro  $\gamma$ . As considerações feitas a respeito do Método de Newton, bem como outras informações adicionais, podem ser encontradas em Arora (2011) e Nocedal e Wright (2006).

### 3.6.3 Método de Gauss-Newton

Outra alternativa, que busca contornar o problema do elevado custo computacional e de parte da instabilidade do Método de Newton, é o método de Gauss-Newton, sendo o mesmo o primeiro método específico para problemas de mínimos quadrados não-lineares testado. Nele, ao invés de usar a a matriz Hessiana em seu formato completo, usa-se apenas os termos cruzados, ignorando-se os termos quadráticos. Dessa forma, a equação governante do Método de Gauss-Newton pode ser escrita como

$$\{\Delta A^I\} = -\gamma(2[J]^T[J])\setminus\{g\} \quad (3.158)$$

na qual  $[J]$  é a matriz Jacobiana das derivadas dos termos do vetor de erro  $y_i$  em função das áreas  $A_j^I$ , tal como anteriormente descrito, e  $\setminus$  representa a solução do sistema linear por

algum método apropriado que não envolva inversão direta.  $\gamma$  é, assim como anteriormente, um parâmetro que regula o tamanho do passo e pode ser estimado de diversas maneiras, como por busca linear (NOCEDAL; WRIGHT, 2006).

Igualmente, usando-se as matrizes auxiliares anteriormente escritas, pode-se formular o problema como

$$\{\Delta A^T\} = -([X] \odot [H_1]) \setminus \{g\} \quad (3.159)$$

sendo  $\odot$  o produto de Hadamard e  $[X]$  e  $[H_1]$  as matrizes descritas na demonstração da Hessiana, notadamente com  $[H_1]$  representando o produto  $[J]^T[J]$  separado da divisão de cada elemento pelo múltiplo das áreas das barras associadas ao seu par de indexadores.

A vantagem deste método é que, ao contrário da Hessiana cheia, quando  $[J]$  tem ranking cheio, o produto  $[J]^T[J]$  é positivo definido e  $-(2[J]^T[J]) \setminus \{g\}$  é uma direção de decréscimo da função, contanto que o gradiente não seja nulo. Mais ainda, se o algoritmo convergir, tal convergência acontece para um ponto de mínimo. Apesar disso, contudo, a convergência do método não é garantida nem sequer nos casos em que o Método de Newton converge, por exemplo quando o ponto inicial é suficientemente próximo da solução, sendo, neste aspecto, pior que o mesmo. Isto acontece porque Gauss-Newton assume que os termos quadráticos podem ser desprezados, ou seja, que  $[H_1]_{ij} \gg [H_2]_{ij}$ , o que só é verdade em funções onde a não linearidade é baixa e há pouca dependência entre uma variável e outra, ou quando os resíduos (dados pelos termos do vetor  $\{y\}$ ) são pequenos, o que não ocorre em vários casos do problema tratado. Mais ainda, assim como em Newton, o produto  $[J]^T[J]$  pode gerar uma matriz mal-condicionada e de difícil inversão numérica, notadamente quando  $[J]$  é deficitária em termos de ranking, ou próxima disto. Nestes casos,  $-(2[J]^T[J]) \setminus \{g\}$  não é, necessariamente, uma direção de minimização, e o algoritmo não converge, mesmo com o ajuste do parâmetro  $\gamma$  (NOCEDAL; WRIGHT, 2006). Portanto, parte-se para a quarta e última estratégia testada.

#### 3.6.4 Algoritmo de Levenberg-Marquardt

Uma alternativa comumente usada para problemas de mínimos quadrados não-lineares, especialmente os que apresentam dificuldades de convergência, é o Algoritmo de Levenberg-Marquardt, que faz uma interpolação entre os métodos do gradiente e de Gauss-Newton, podendo tirar proveito das vantagens de ambos, notadamente a estabilidade e garantia de convergência do Método do Gradiente e a velocidade de convergência do Método de Gauss. A equação governante do método pode ser escrita, para o problema em questão, como

$$(2[J]^T[J] + \lambda[I])\{\Delta A^I\} = \{g\} \quad (3.160)$$

que é resolvida para  $\{\Delta A^I\}$  simbolicamente como

$$\{\Delta A^I\} = -(2[J]^T[J] + \lambda[I]) \setminus \{g\} \quad (3.161)$$

equação essa análoga à do método de Gauss-Newton, a exceção do termo  $\lambda[I]$ , no qual  $\lambda$  é o parâmetro de interpolação e  $[I]$  é a matriz identidade, sendo  $\setminus$ , mais uma vez, uma solução para o sistema linear que não exija inversão da matriz. Neste caso, o passo é controlado pelo tamanho do parâmetro  $\lambda$ , ao invés de  $\gamma$ , como em Gauss-Newton.

Já a interpolação funciona da seguinte maneira: para valores de  $\lambda$  suficientemente grandes, tem-se que

$$(2[J]^T[J] + \lambda[I])^{-1} \approx (\lambda[I])^{-1} = \frac{1}{\lambda}[I] \quad (3.162)$$

Aplicando esta equação à anterior, tem-se que

$$\{\Delta A^I\} \approx -\frac{1}{\lambda}[I]\{g\} = -\frac{1}{\lambda}\{g\} \quad (3.163)$$

na qual o termo  $1/\lambda$  é análogo ao  $\alpha$  de controle de passo apresentado no Método do Gradiente, significando que o método de Levenberg-Marquardt se aproxima do mesmo quando  $\lambda$  cresce.

De maneira inversa, quando  $\lambda$  é suficientemente pequeno, chega-se a

$$2[J]^T[J] + \lambda[I] \approx 2[J]^T[J] \quad (3.164)$$

logo

$$\{\Delta A^I\} = -(2[J]^T[J] + \lambda[I]) \setminus \{g\} \approx -([J]^T[J]) \setminus \{g\} \quad (3.165)$$

ou seja: o método de Levenberg-Marquardt decai para o Método de Gauss-Newton.

A vantagem de Levenberg-Marquardt com relação aos métodos que interpola é que  $\lambda[I]$  é um termo 'regularizador' de  $[J]^T[J]$ . Isso significa que, para  $\lambda$  suficientemente grande,  $2[J]^T[J] + \lambda[I]$  é positiva definida, numericamente inversível e  $-(2[J]^T[J] + \lambda[I]) \setminus \{g\}$  é uma direção de minimização da função objetivo, mesmo quando  $[J]$  é mal condicionada. Portanto, para pontos distantes do mínimo, com grandes resíduos ou com  $[J]$

problemático, para os quais o Método de Gauss não garantiria convergência, Levenberg-Marquardt pode aproximar-se mais do Método do Gradiente, 'amortecendo' a iteração com o aumento do parâmetro  $\lambda$ , e buscar estabilidade. Ao contrário, quando a aproximação da Hessiana feita pelo termo  $2[J]^T[J]$  é boa e a matriz é bem-condicionada, pode-se 'relaxar' o parâmetro  $\lambda$ , efetivamente aumentando-se o tamanho do passo do algoritmo e aproximando o mesmo de uma iteração de Gauss-Newton, aproveitando-se da convergência acelerada do mesmo. Outra vantagem é que Levenberg-Marquardt, assim como Gauss-Newton, não necessita das segundas derivadas da matriz Hessiana, o que acelera o cálculo profundamente (ainda que cada iteração seja significativamente mais custosa que no Método do Gradiente).

Fica, portanto, bastante clara a importância do parâmetro  $\lambda$ , para o qual uma estratégia de atualização apropriada precisa ser definida. Para tanto, é interessante pensar no algoritmo de Levenberg-Marquardt como pertencente à classe dos métodos de 'Região de Confiança'. Para os mesmos, inicia-se por definir uma equação do tipo

$$\rho_k = \frac{f(\{A^I\}_k) - f(\{A^I\}_k + \{\Delta A^I\}_k)}{m_k(\{0\}) - m_k(\{\Delta A^I\}_k)} \quad (3.166)$$

na qual  $\rho_k$  uma razão entre a mudança no valor da função objetivo efetivamente medido e o valor esperado para uma aproximação quadrática da mesma. Para tanto, a função  $m_k(\{\Delta A^I\})$  é definida através de uma expansão de Taylor como sendo

$$m_k(d) = f_k + \{d\}_k^T \{g\}_k + \frac{1}{2} \{d\}_k^T [H]_k \{d\}_k \quad (3.167)$$

na qual, por questão de compacidade da notação, tem que  $\{A^I\}_k = d$ ,  $f(\{A^I\}_k) = f_k$ ,  $\{g(\{A^I\}_k)\} = \{g\}_k$  e  $[H(\{A^I\}_k)] = [H]_k$  e, por consequência,  $m_k(\{0\}) = f_k$ . Assim como anteriormente,  $\{g\}$  é o gradiente e  $[H]$  a Hessiana em sua forma completa.

Dessa maneira, a Eq. 3.166 fica escrita, também de maneira compacta, usando-se a Eq. 3.167, como

$$\rho_k = \frac{f_k - f_{k+1}}{-\{g\}_k^T \{d\}_k - \frac{1}{2} \{d\}_k^T [H]_k \{d\}_k} \quad (3.168)$$

Contudo, como descrito anteriormente, o Método de Levenberg-Marquardt substitui a matriz Hessiana por sua aproximação em Gauss-Newton, que, para o problema tratado, tem formato  $[H]_k \approx 2[J]_k^T[J]_k$ . Fazendo-se essa substituição na Eq. 3.168, tem-se que

$$\rho_k = \frac{f_k - f_{k+1}}{-\{g\}_k^T \{d\}_k - \frac{1}{2} \{d\}_k^T \left( 2[J]_k^T [J]_k \right) \{d\}_k} \quad (3.169)$$

ou seja:

$$\rho_k = \frac{f_k - f_{k+1}}{-\{g\}_k^T \{d\} - \{d\}^T [J]^T [J] \{d\}} \quad (3.170)$$

Na sequência, isolando-se o termo repetido  $-\{d\}^T$ , obtêm-se

$$\rho_k = \frac{f_k - f_{k+1}}{-\{d\}^T (\{g\}_k + [J]^T [J] \{d\})} \quad (3.171)$$

Para prosseguir com a demonstração, resgata-se a expressão descrita na Eq. 3.160 para o sistema linear do algoritmo e distribui-se os termos entre parênteses na mesma, chegando-se a

$$2[J]^T [J] \{\Delta A^I\} + \lambda [I] \{\Delta A^I\} = \{g\} \quad (3.172)$$

Em seguida, fazendo  $\lambda [I] \{\Delta A^I\} = \{\Delta A^I\}$  e  $\{\Delta A^I\} = \{d\}$ , continuando com a abreviação de anteriormente, encontra-se

$$2[J]^T [J] \{d\} + \lambda \{d\} = -\{g\} \quad (3.173)$$

Por fim, isolando-se o termo  $[J]^T [J] \{d\}$ , chega-se a

$$[J]^T [J] \{d\} = -\frac{1}{2} (\{g\} + \lambda \{d\}) \quad (3.174)$$

Substituído-se esse resultado na Eq. 3.171, obtêm-se

$$\rho_k = \frac{f_k - f_{k+1}}{-\{d\}^T \left[ \{g\}_k - \frac{1}{2} (\{g\} + \lambda \{d\}) \right]} \quad (3.175)$$

que, por fim, após simplificação e reordenação, resulta em

$$\rho_k = \frac{f_k - f_{k+1}}{\frac{1}{2} \{d\}^T (\lambda \{d\} - \{g\})} \quad (3.176)$$

Na sequência, inicia-se por lembrar que o termo  $\rho_k$  representa, como descrito an-

teriormente, a relação entre a mudança prevista para o valor da função objetivo, com base na aproximação quadrática feita, e o valor efetivamente observado. Dessa forma, uma vez que o termo  $\lambda_k \{d\}^T \{d\}$  é positivo, por consequência de que  $\{d\}^T \{d\} = |\{d\}|^2$  e que  $\lambda_k$  é positivo, bem como que, para que haja um decréscimo da função objetivo, o produto escalar  $\{d\}^T(-\{g\})$  também deve ser positivo, uma vez que o incremento  $\{d\}$  precisa estar próximo da direção de decréscimo dada por  $-\{g\}$ , tem-se que a variável livre do parâmetro  $\rho_k$  é exatamente o fator  $\lambda_k$ , e, mais ainda, quanto maior o mesmo, menor é o denominador da fração. Tendo isso em mente, pode-se entender porque uma estratégia comum de atualização do parâmetro  $\lambda$  consiste em aplicar as seguintes regras:

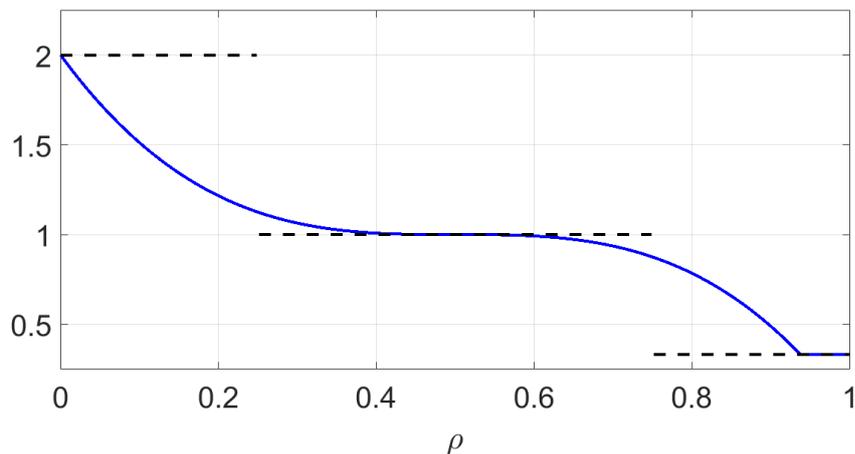
$$\begin{aligned} \text{Se } \rho < \rho_1, \quad \lambda &:= \lambda\beta \\ \text{Se } \rho > \rho_2, \quad \lambda &:= \lambda/\gamma \\ \text{Se } \rho > 0, \quad \{x\} &:= \{x\} + \{d\} \end{aligned} \tag{3.177}$$

ou seja: se a mudança na variável  $\rho$  é menor do que um valor esperado definido (ou seja: a mudança real no valor da função objetivo é menor do que uma fração definida da mudança esperada para a mesma), isso significa que a aproximação feita é ruim e o passo precisa ser restringido, o que é feito com o aumento da variável  $\lambda$ , que, como discutido, tem correlação com o tamanho do passo. Também, como se pontuou, aumentar o valor de  $\lambda$  reduz o denominador, portanto aproximando-o do valor do numerador encontrado menor que o previsto, indicando que o passo dado está distante demais da região em torno do ponto para a qual a aproximação feita é considerada boa, denominada de 'Região de Confiança', daí o nome do método. Do contrário, se o valor de  $\rho$  é maior do que um máximo previsto, isso significa que o passo está dentro da região de confiança correspondente a uma aproximação quadrática de boa qualidade, e, portanto, pode ser aumentado, o que é feito diretamente pela redução do valor do parâmetro  $\lambda$ . Por fim, sendo  $\rho > 0$ , isso significa que a função objetivo teve seu valor reduzido e, portanto, pode-se aceitar o passo e atualizar as variáveis de projeto, sendo o mesmo rejeitado caso contrário. Fica claro, portanto, que  $0 < \rho_1 < \rho_2 < 1$  e que  $\beta, \gamma > 1$ , sendo, comumente,  $\beta < \gamma$ . Por sorte, o método é pouco sensível a mudanças nestes parâmetros, sendo os valores comumente utilizados  $\rho_1 = 0,25$ ,  $\rho_2 = 0,75$ ,  $\beta = 2$  e  $\gamma = 3$  bons para a maior parte dos casos. Contudo, pequenas variações no método tradicional apresentam ganhos de desempenho tangíveis, mesmo que diminutos. Uma delas é utilizar uma estratégia de atualização contínua para o parâmetro  $\rho$ , de modo a evitar os saltos discretos que acontecem na estratégia tradicional e permitir o ajuste de maneira mais suave. Uma delas tem formato

$$\begin{aligned}
&\text{Se } \rho > 0; \\
&\quad \{x\} := \{x\}; \\
&\quad \lambda := \lambda \max\left(\frac{1}{\gamma}, 1 - (\beta - 1)(2\rho - 1)^n\right); \quad \nu = \beta; \\
&\text{Senão, } \quad \lambda := \lambda\nu; \quad \nu := 2\nu;
\end{aligned} \tag{3.178}$$

Nesta variação, caso  $\rho > 0$ ,  $\lambda_{k+1}/\lambda_k$  pode variar de maneira contínua e suave entre a redução máxima de  $1/3$  e o aumento máximo de  $2$ , sendo não suave apenas nas proximidades de  $\rho = 1$ , onde o valor é truncado como sendo  $1/3$ . Mais ainda, para repetidas iterações onde não haja convergência (ou seja:  $f_{k+1} > f_k$ ), o parâmetro  $\nu$  acelera o aumento do amortecimento, de modo a trazer o passo mais rapidamente para dentro da região de confiança. Nesta versão,  $\nu$  é inicializado com o valor de  $\beta$  e  $n$  é um inteiro ímpar. Um gráfico comparativo entre as duas abordagens é apresentado na Fig. 3.5, na qual a curva contínua em azul representa a estratégia contínua e as linhas pontilhadas pretas a proposta discreta clássica, sendo o eixo horizontal o parâmetro  $\rho$  e o eixo vertical a relação  $\lambda_{k+1}/\lambda_k$ . Para a geração da curva contínua, assim como para os exemplos do capítulo seguinte, usa-se  $n = 3$  e os mesmos valores de  $\beta$  e  $\gamma$  anteriormente apresentados, respectivamente de  $2$  e  $3$ .

**Figura 3.5 – Curvas de atualização do parâmetro de amortecimento do algoritmo de Levenberg-Marquardt**



Fonte: Autor

Neste trabalho, o valor inicial de  $\lambda$  é considerado como sendo de

$$\rho_0 = \max(\text{abs}(\text{diag}([J]^T[J]))) \tag{3.179}$$

ou seja: como tendo valor equivalente ao maior termo (em módulo) dentre aqueles da

diagonal de  $[J]^T[J]$

A estratégia contínua apresentada, seu comparativo com a estratégia discreta e a justificativa dos valores escolhidos para o parâmetro são tiradas de Nielsen (1999), que apresenta ainda o equacionamento da 'Região de Confiança' replicado. Outras informações a respeito do algoritmo e de suas propriedades de convergência podem ser encontrados em Nocedal e Wright (2006), que também ilustra outros métodos de otimização não-testados nesta dissertação, incluindo esta e outras abordagens via 'Região de Confiança'.

## 4 RESULTADOS E DISCUSSÕES

*"Um especialista é uma pessoa que cometeu todos os erros que podem ser cometidos em um campo de atuação bastante específico"*

– Niels Bohr

Neste capítulo, apresentam-se os resultados obtidos com a implementação computacional das equações anteriormente discutidas, bem como discussões acerca destes. Inicia-se com um exemplo analítico de verificação, para o qual soluções analíticas são obtidas e comparadas com as respostas numéricas para diversas das grandezas anteriormente formuladas. Por fim, exemplos ilustrativos são contrastados, de modo a analisar o efeito de diversos parâmetros de modelo na resposta da formulação.

### 4.1 Exemplo de Verificação

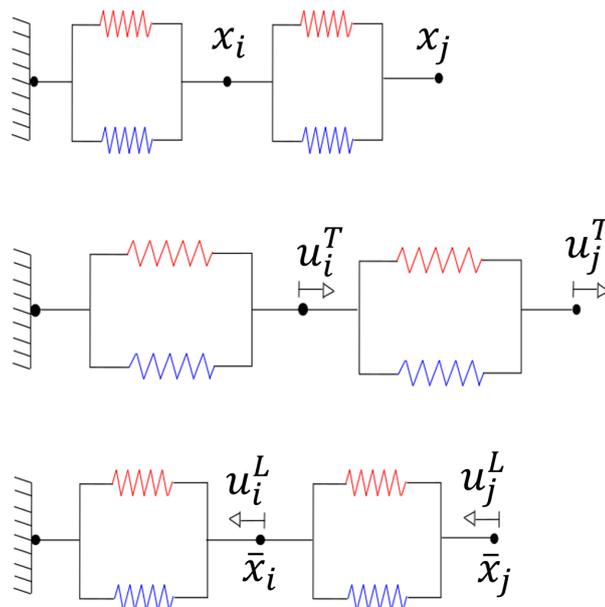
Nesta seção, define-se um exemplo analítico base. O mesmo é utilizado para a verificação da formulação anteriormente apresentada no âmbito do problema estrutural, do Método da Rigidez Direta, das formulações do vetor gradiente e da matriz Hessiana, bem como da implementação desenvolvida. Considerações acerca da natureza do problema de otimização e dos custos computacionais associados à formulação também são apresentadas.

#### 4.1.1 Definição do Exemplo e Função Objetivo

Como exemplo de verificação, com objetivo de contrastar as equações desenvolvidas com resultados analíticos, inicia-se por definir o sistema estrutural apresentado na Fig. 4.1. Nele, dois conjuntos de molas em paralelo são ligadas em sequência. As molas superiores em vermelho representam elementos de inclusão, sendo as molas inferiores em azul os elementos de matriz, uma vez que, nos sistemas treliçados propostos, as barras de matriz ficam superpostas às de inclusão (e, portanto, também ligadas aos mesmos nós que elas em paralelo). O sistema apresenta apenas graus de liberdade horizontais, associados aos deslocamentos dos três nós indicados na figura, com o nó da esquerda iniciando na origem do sistema e os dois nós seguintes tendo posições iniciais de, respectivamente,  $x_i$  e  $x_j$ , conforme a primeira fileira da figura. As molas de inclusão possuem dois valores de

rigidez, um associado ao estado borrachoso e outro ao estado vítreo. Já as barras de matriz apresentam rigidez única. O nó esquerdo é engastado (e, portanto, tem deslocamento prescrito nulo). Ao nó da direita, por sua vez, é prescrito um deslocamento durante a etapa de treinamento com a inclusão no estado borrachoso. Esse deslocamento, simbolizado por  $u_j^T$ , provoca, por sua vez, um deslocamento  $u_i^T$  no nó central, conforme a segunda fileira. Na sequência, a inclusão é congelada, o que modifica sua rigidez para o valor vítreo e converte a deformação elástica na mesma em plástica. Por fim, a prescrição de deslocamento é aliviada na extremidade livre, sendo substituída por uma prescrição de força nula. O resultado da soma de ambas essas etapas é indicado na terceira e última fileira, na qual os deslocamentos  $u_i^L$  e  $u_j^L$  referem-se à passagem da configuração de treinamento para a configuração liberada.  $\bar{x}_i$  e  $\bar{x}_j$  são as posições que se deseja prescrever para os dois nós à direita após o processo de otimização, também como indicado na figura.

Figura 4.1 – Representação esquemática do sistema estrutural analítico tratado



Fonte: Autor

Para a solução deste problema, o Método da Rigidez Direta é implementado no pacote de matemática simbólica Maple. O mesmo se trata de uma versão análoga à discutida no capítulo anterior, porém apenas com graus de liberdade horizontais, como descrito. As rigidezes das molas são escritas como se fossem associadas a barras segundo as expressões

$$k^M = \frac{E_m A_m}{L}; k_{r_i}^I = \frac{E_r A_i}{L}; k_{g_i}^I = \frac{E_g A_i}{L}; k_{r_j}^I = \frac{E_r A_j}{L}; k_{g_j}^I = \frac{E_g A_j}{L} \quad (4.1)$$

nas quais  $k^M$  é a rigidez, de mesmo valor, associada às duas molas de matriz, sendo  $k_{r_i}^I$ ,  $k_{g_i}^I$ ,  $k_{r_j}^I$  e  $k_{r_j}^I$ , respectivamente, as rigidezes da mola de inclusão à esquerda em estado borrachoso e vítreo, seguidas pelas rigidez em estado borracho e vítreo da mola de inclusão à direita. Com relação aos lados direitos das equações,  $L$  é um comprimento único associado à todas as barras representas por molas, sendo  $E_m$  o módulo de elasticidade linear da matriz e  $E_r$  e  $E_g$ , respectivamente, as rigidezes em estado borrachoso e vítreo associadas às duas barras de inclusão. Por fim,  $A_m$  é a área das duas barras de matriz, sendo  $A_i$  e  $A_j$  as áreas, respectivamente, da barras da esquerda e direita.

Faz-se essa analogia entre barras e molas para que se possa confrontar diretamente os resultados obtidos pela implementação numérica das equações desenvolvidas com respostas analíticas para o sistema de molas. Para tanto, o sistema estrutural descrito é resolvido analiticamente pelo Maple para valores quaisquer dos parâmetros descritos usando-se o Método da Rigidez Direta, tal como apresentado no Capítulo 2, mas convertido para o formato apropriado para os graus de liberdade adotados. Vale salientar que o coeficiente grande adotado para os nós fixados é deixado, também, como um variável analítica. Subsequentemente, será tirado o limite das soluções obtidas com o mesmo tendendo ao infinito, para que se obtenha resultados exatos para as grandezas medidas. Uma vez que tais deslocamentos são obtidos analiticamente, os mesmos podem ser derivados também de maneira exata, o que, para deslocamentos prescritos também em formato de variáveis, permite a criação de soluções analíticas exatas para a função objetivo, o gradiente, a matriz Jacobiana e a matriz Hessiana tomando-se o limite do deslocamento quando a rigidez artificial tende a infinito, como discutido.

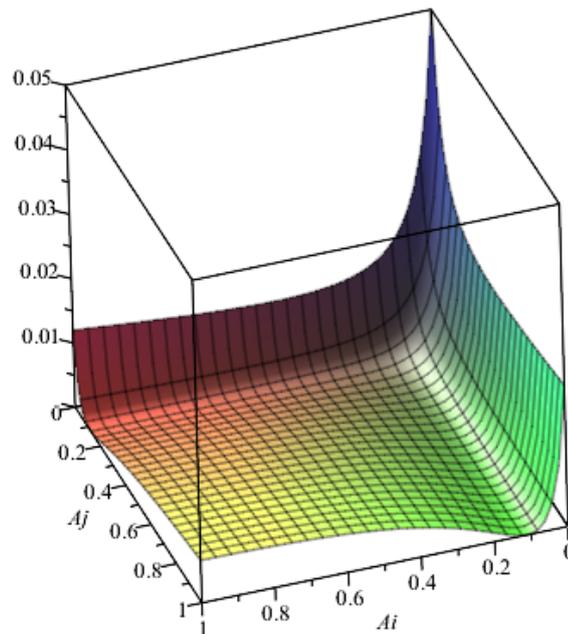
Contudo, mesmo para um sistema simples como este, as soluções encontradas são grandes e de difícil compreensão. Para facilitar o entendimento e permitir a comparação com uma solução numérica, valores são atribuídos às variáveis deixadas abertas. Para as áreas das barras de matriz, os módulos de elasticidade das barras de matriz e de inclusão em estado borrachoso e os comprimentos de todas, adotam-se valores unitários. Aos nós da direita e do meio, prescrevem-se deslocamentos de, respectivamente, 0,2 e 0,1, sendo o esticção dado na etapa de treinamento de valor 0,3. O módulo de elasticidade das barras de inclusão em estado vítreo é considerado como tendo valor de 10, enquanto que as áreas das mesmas são deixadas em aberto. Neste caso específico, a função objetivo tem como resposta analítica

$$\begin{aligned}
f = \frac{1}{200(10A_j + 1)^2(10A_i + 1)^2(A_i + A_j + 2)^2} & (4000A_i^4A_j^2 - 4000A_i^3A_j^3 \\
& + 10000A_i^2A_j^4 - 400A_i^4A_j + 2800A_i^3A_j^2 + 17200A_i^2A_j^3 \\
& - 1600A_iA_j^4 + 100A_i^4 - 3040A_i^3A_j + 129000A_i^2A_j^2 - 43600A_iA_j^3 \\
& + 100A_j^4 + 60A_i^3 - 560A_i^2A_j - 2240A_iA_j^2 + 180A_j^3 + 85A_i^2 - \\
& 458A_iA_j + 157A_j^2 + 12A_i + 36A_j + 4) \quad (4.2)
\end{aligned}$$

Pode-se observar, portanto, que a função objetivo é do tipo racional multi-variável (ou seja: uma fração na qual tanto o numerador quanto o denominador são polinômios em múltiplas variáveis, no caso  $A_i$  e  $A_j$ ). Dessa forma, suas derivadas e, por consequência, a Hessiana, o gradiente e a Jacobiana, também são funções racionais que não serão apresentadas por questão de brevidade.

Para facilitar a compreensão, um gráfico da função objetivo é apresentado na Fig. 4.2, no qual o eixo vertical representa o valor da mesma, sendo o par de eixos horizontais as áreas das duas barras de inclusão. Nele, as áreas das barras são plotadas com valores no intervalo entre 0 e 1.

**Figura 4.2 – Gráfico da função objetivo analítica**

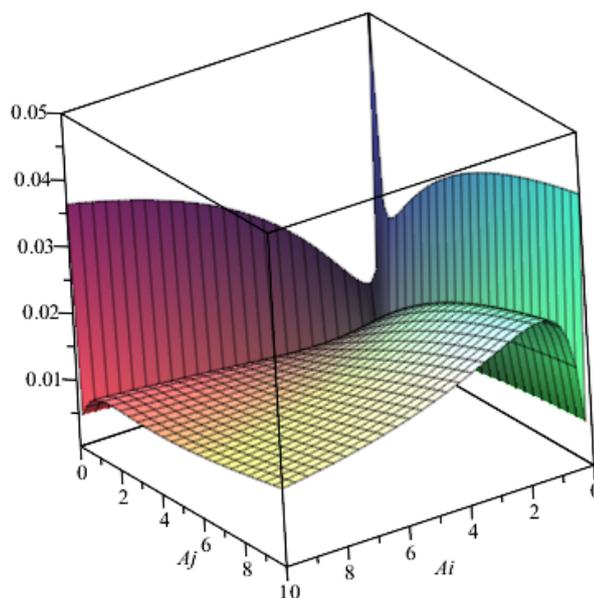


**Fonte: Autor**

Para barras de inclusão de área nula, a função objetivo deve ter como valor a soma dos quadrados dos deslocamentos prescritos, uma vez que sua configuração defor-

mada após o treinamento e liberação será a própria configuração indeformada, posto que, fisicamente falando, o efeito de memória de forma não se manifesta pra barras de inclusão inexistentes. Neste caso, em que o vetor de deslocamentos é nulo, tem-se que  $f = \bar{u}_i^2 + \bar{u}_j^2 = (0,1)^2 + (0,2)^2 = 0,05$ , o que de fato se observa no gráfico e com perfeita exatidão na verificação do valor analítico de  $f$  no ponto  $(0,0)$ . Outro ponto notável a ser observado é a presença de um vale estreito e em formato de 'v' curvo nas proximidades das retas  $x = 0,1$  e  $y = 0,1$ . Para ilustrar melhor a presença do mesmo, uma vez que a proximidade da Fig. 4.2 mascara sua verdadeira forma, uma versão do gráfico com as áreas das barras podendo atingir um limite superior de 10, e não 1, é apresentada na Fig. 4.3

**Figura 4.3 – Gráfico da função objetivo analítica em nova escala**



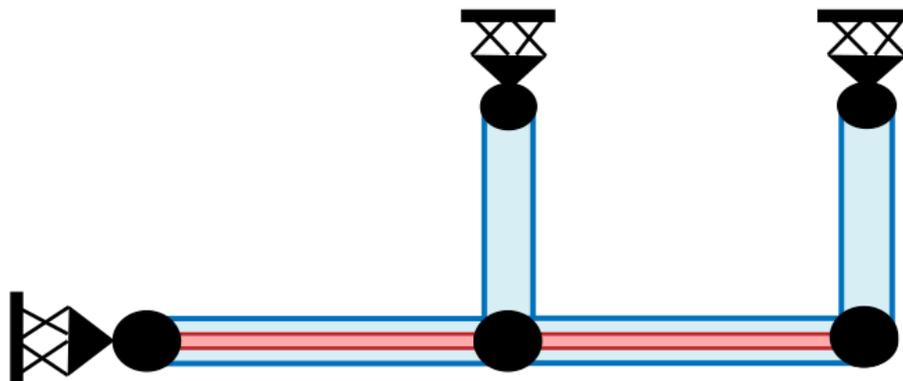
**Fonte: Autor**

Tal vale estreito, agora muito mais pronunciado, se presente nos problemas ilustrativos apresentados na sequência em versões multi-dimensionais, pode justificar a lenta convergência do Método do Gradiente (e, em menos escala, do Algoritmo de Levenberg-Marquardt), que, para percorrer o vale até o ponto de mínimo, precisa progredir em passos pequenos e oscilatórios. Mais ainda, pode-se observar que mesmo este problema simples é claramente não convexo, apresentando regiões côncavas, quase planas e convexas, bem como, possivelmente, pontos de sela. Isso justifica, por exemplo, o fato do Método de Newton não ser capaz de solucionar o problema. Mais ainda, nas proximidades do vale, percebe-se uma rápida mudança no gradiente e na curvatura da superfície, o que pode ajudar a explicar os problemas de condicionamento das matrizes tratadas, usadas para aproximar a curva quadraticamente.

## 4.1.2 Verificação da Formulação

Para a verificação da formulação matemática construída anteriormente, inicia-se por definir um ponto arbitrário para as variáveis de projeto (áreas das barras de inclusão) analisadas. O ponto escolhido foi  $(0,15,0,3)$ . Na sequência, um análogo ao sistema estrutural descrito pelas molas anteriores é construído para a versão em treliça implementada no código. O sistema análogo adotado é apresentado na Fig. 4.4, na qual os três nós inferiores representam os três nós do sistema em molas. Ligando esses três nós estão quatro barras organizadas em pares, uma superposta à outra. Desse par, tal como na mola, uma barra representa a matriz e a outra a inclusão (respectivamente em azul e por cima em vermelho). Também como anteriormente, o nó esquerdo é restringido horizontalmente. Contudo, somente com estes nós e barras, ter-se-ia um sistema estrutural hipostático, uma vez que não há rigidezes verticais associadas a nenhum dos nós inferiores, embora os graus de liberdade verticais existam. De modo a estabilizar o sistema, tornando-o hiperestático, duas barras são adicionadas aos dois nós à direita. Tais barras, posicionadas verticalmente e associadas a vinculações de segundo grau (que restringem movimentos verticais e horizontais), de acordo com a aproximação de pequenas rotações, deformações e deslocamentos feita, não modificam o comportamento do sistema na direção horizontal, mas, juntamente com uma restrição na direção vertical aplicada ao nó da esquerda, também tornando seu apoio um de segundo gênero, convertem o sistema estrutural para um do tipo hiperestático, sendo as matrizes de rigidez de ambos os estágios de carregamento inversíveis e numericamente estáveis.

**Figura 4.4 – Sistema análogo ao analítico para o modelo em treliça**



Fonte: Autor

As áreas, módulos de elasticidade e comprimentos das barras inferiores são considerados como tendo os mesmos valores que os anteriormente utilizados, de modo a compatibilizar os dois sistemas. As barras de cima, adicionadas apenas por questão de inversibilidade da matriz, poderiam ter quaisquer valores não-nulos e não muito discrepantes dos utilizados para as demais, por questão de estabilidade numérica. Para homogeneizar

o problema o máximo possível, escolhe-se os mesmos valores adotados para as duas barras inferiores de matriz.

Por fim, as soluções obtidas numericamente pela implementação usando as expressões discutidas no capítulo anterior para o vetor gradiente e a matriz Hessiana são comparados com aqueles determinados analiticamente pelo Maple. Para os resultados analíticos, tem-se as seguintes respostas para a função objetivo, o gradiente e a Hessiana:

$$f = 0,2141815910 \cdot 10^{-4} \quad (4.3a)$$

$$\{g\} = 10^{-2} \left\{ -0,1343348435 \quad -0,0131447143 \right\}^T \quad (4.3b)$$

$$[H] = \begin{bmatrix} 0,2477374390 & 0,0563192273 \\ 0,0563192273 & 0,0141031136 \end{bmatrix} \quad (4.3c)$$

Já os resultados numéricos obtidos com a implementação são, truncados para o mesmo número de casas decimais

$$f = 0,2141464497 \cdot 10^{-4} \quad (4.4a)$$

$$\{g\} = 10^{-2} \left\{ -0,1342819964 \quad -0,0131321361 \right\}^T \quad (4.4b)$$

$$[H] = \begin{bmatrix} 0,2477326669 & 0,0563194994 \\ 0,0563194994 & 0,0141025055 \end{bmatrix} \quad (4.4c)$$

Pode-se perceber excelente concordância entre as respostas do exemplo analítico e do resultado numérico baseado nas equações discutidas. Na função objetivo, incompatibilidades entre as respostas aparecem apenas no quinto algarismo, correspondendo a uma diferença de menos de 0,0165% entre a solução numérica e o valor exato, mesmo com os sistemas estruturais tratados sendo fundamentalmente diferentes, ainda que análogos, e mesmo com todos os erros numéricos associados ao processo, como as soluções numéricas dos sistemas lineares da estrutura e a aplicação da rigidez artificial nos nós fixados, para citar alguns. No gradiente, o maior erro relativo percentual encontrado é aproximadamente (e ligeiramente menor que) 0,0957%, referente ao termo  $(g)_2$ . Já na Hessiana, por fim, o maior erro é de aproximadamente 0,00431% em  $(H)_{22}$ . Tais valores são consistentes com rigidez artificial adotada, que só garantiria exatidão entre as respostas até o quinto algarismo (visto que a mesma é 5 ordens de magnitude maior que a maior rigidez medida na diagonal da matriz).

### 4.1.3 Considerações Adicionais

É importante observar que o gradiente analítico construído é relativamente eficiente de ser calculado, exigindo, como passo mais custoso, a solução de três sistemas lineares análogos aos da estrutura (uma vez que provem de suas matrizes de rigidez). Para o MATLAB e a forma com que a implementação foi construída, os custos computacionais da montagem das matrizes do problema estrutural, da atualização dos elementos em cada iteração, dos custos do próprio algoritmo de otimização, etc., são significativamente superiores ao mesmo, sendo seu impacto na implementação bastante pequeno. Já o seu cálculo via diferenças finitas, contudo, exige que todas as operações associadas à função objetivo sejam feitas um total de ao menos  $n + 1$  vezes, sendo  $n$  o número de variáveis de projeto (e portanto o tamanho do gradiente), considerando-se diferenças avançadas ou atrasadas, e um total de  $2n$  vezes para diferenças centrais, sendo, em ambos os casos, significativamente mais custosos e bastante ineficientes, especialmente para problemas maiores.

O cálculo da Hessiana em sua forma cheia, por sua vez, é muito mais impactante em termos de custo computacional, sendo responsável, em certos casos, por parcelas relevantes do tempo despendido na simulação, até mesmo aumentando a ordem de magnitude do tempo de análise em uma ou até duas unidades, para alguns dos problemas tratados. Contudo, ainda assim, considerando-se que o custo para calcular a Hessiana via diferenças finitas centrais usando o gradiente é, analogamente ao caso anterior, aproximadamente  $2n$  vezes maior que o custo de chamada da função objetivo, a Hessiana em sua forma analítica é muito mais eficientemente computada que na forma numérica.

Por fim, a Hessiana truncada nos termos cruzados (portanto sem os termos de segunda ordem) usada para o Algoritmo de Levenberg-Marquardt cai no meio das duas análises. Seu custo computacional é maior que o do cálculo do gradiente, mas também muito menor do que o da Hessiana cheia e também muito mais eficiente que seu cálculo via diferenças finitas.

Considera-se, portanto, que os resultados obtidos são suficientemente bons para garantir, com certo grau de segurança, que as equações utilizadas estão corretas, bem como que as mesmas são muito mais eficientes em termos de custo computacional que suas versões numéricas. Dessa forma, tais equações são utilizadas para gerar os resultados apresentados a seguir, que atestam ainda mais para sua validade. Em alguns destes exemplos, os tempos computacionais associados aos cálculos da função objetivo, do gradiente e da Hessiana, tanto em suas formas numéricas quanto analíticas, são apresentados em caráter informativo e para embasar as observações feitas nesta subseção.

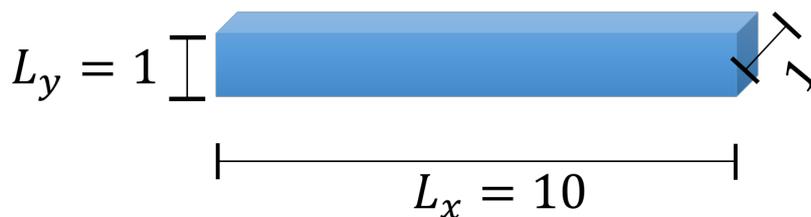
## 4.2 Exemplos Ilustrativos

Nesta seção, apresenta-se uma série de exemplos ilustrativos, de modo a explicitar o efeito dos parâmetros da formulação apresentada que ficaram em aberto na resposta do algoritmo de otimização. Cada subseção apresenta o estudo de um parâmetro, bem como discussões a respeito do mesmo. Os parâmetros consistem em variações, cumulativas ou não, feitas em cima de um exemplo base. O exemplo base, bem como cada variação feita e os resultados obtidos nelas, são apresentados nas subseções a seguir.

### 4.2.1 Exemplo Base

O problema base a ser tratado é o de uma viga de dimensões  $10 \times 1 \times 1$ , conforme ilustrado na Fig. 4.5, sendo o plano  $x-y$  discretizado e sendo a espessura da viga considerada unitária por consistência com o tratamento bidimensional apresentado.

**Figura 4.5 – Modelo de viga tratado nos exemplos ilustrativos**

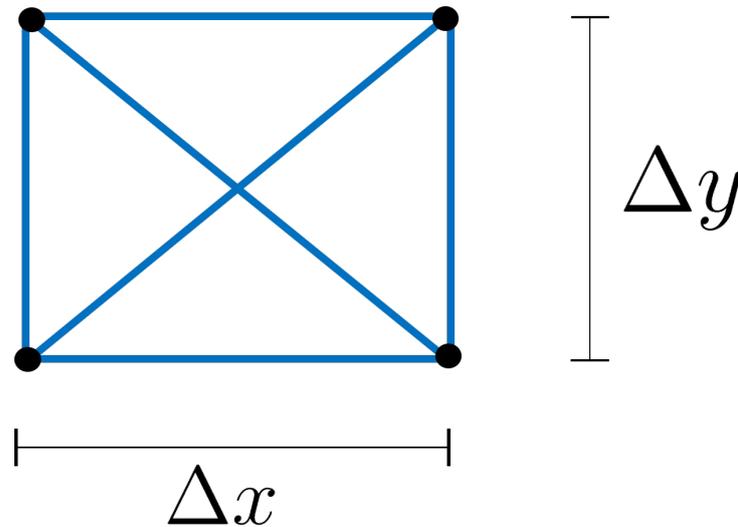


Fonte: Autor

As unidades utilizadas para as dimensões da viga, bem como para todas as demais grandezas, são consideradas arbitrárias e consistentes, e portanto não explicitadas. Para a discretização da viga no plano  $x-y$ , usa-se uma malha de 101 nós na horizontal e 11 na vertical, totalizando 1111 nós, todos espaçados uniformemente. Os nós dividem a estrutura em 1000 unidades quadradas (ou seja:  $\Delta x = \Delta y$ ), conforme apresentado na Fig. 4.6.

Cada da malha tem barras à direita, esquerda, cima, baixo e duas barras cruzadas passando por seu centro, também conforme a Fig. 4.6. Vale salientar que o ponto central de cruzamento, contudo, não representa um nó da estrutura, ao contrário dos pontos nos vértices dos quadrados. Mais ainda, a barra a direita de um quadrado é a mesma que a barra a esquerda do quadrado ao lado e assim por diante (e, analogamente, a barra de cima de um quadrado é a mesma barra que a de baixo do quadrado acima). A escolha dessa malha é justificada pela necessidade de se criar um comportamento isotrópico para a matriz. Tal comportamento não seria obtido com elementos retangulares, para os quais as barras cruzadas teriam uma direção preferencial. Para está configuração de malha, assim como outros que serão tratadas futuramente com a mesma geometria, tem-se um total de  $(n_x - 1)n_y$  barras horizontais,  $n_x - 1$  por camada,  $(n_y - 1)n_x$  barras verticais,

Figura 4.6 – Unidade quadrangular da malha utilizada



Fonte: Autor

$n_y - 1$  por camada e ainda um total de  $2(n_x - 1)(n_y - 1)$  barras diagonais. Somando horizontais, verticais e cruzadas, tem-se um total de  $4n_x n_y - 3n_x - 3n_y + 2$  barras. Os nós mapeados para as configurações prescritas são apenas os da linha média, em conformidade com as formulações de viga, que representam a mesma através da reta que passa pelo centroide das seções transversais. Tal linha média representa, para esta discretização da malha, 101 nós horizontalmente alinhados passando pelo meio da viga. A matriz tem módulo de elasticidade de valor 1. Os valores adotados para as rigidezes das barras de inclusão são, respectivamente para o estado borrachoso e vítreo, de 2 e 20. Tal aumento é em conformidade com a natureza física da cristalização dos polímeros, que normalmente elevam suas rigidezes em uma ou mais ordens de magnitude. A viga é considerada engastada em sua extremidade esquerda e esticada horizontalmente em sua extremidade direita durante a fase de treinamento. O deslocamento aplicado nessa extremidade livre é de 0,1. Como estimativa inicial, faz-se as barras de inclusão terem áreas equivalentes a 1% das áreas das barras de matrizes que as mesmas superpõe. Mais ainda, a região de inclusão é considerada como englobando todos os nós da matriz, exceto os externos. O filtro do algoritmo é considerado como sendo de 1%. Isso significa que, para cada iteração do processo de otimização, as barras cujas áreas sejam menores do que 1/10 da área da maior barra daquela iteração são eliminadas do restante do processo de otimização. Como critérios de parada para o algoritmo, encerra-se a otimização quando a mudança na função objetivo for menor que 1% em 5 iterações seguidas ou que um máximo de 100 iterações seja atingido. Outro critério de parada é que o algoritmo encerra a busca caso 5 tentativas de iteração seguidas não resultem em uma iteração viável (uma para a qual o valor da função objetivo decresça). Por fim, a configuração da linha média que se deseja prescrever é uma de natureza quadrática, onde os pontos  $(x,y)$  de cada nó da mesma tem

a posição final desejada expressa de acordo com a equação

$$\bar{x}_f(x_i, y_i) = x_i \quad (4.5a)$$

$$\bar{y}_f(x_i, y_i) = y_i + \alpha \left( \frac{x_i}{L_x} \right)^2 \quad (4.5b)$$

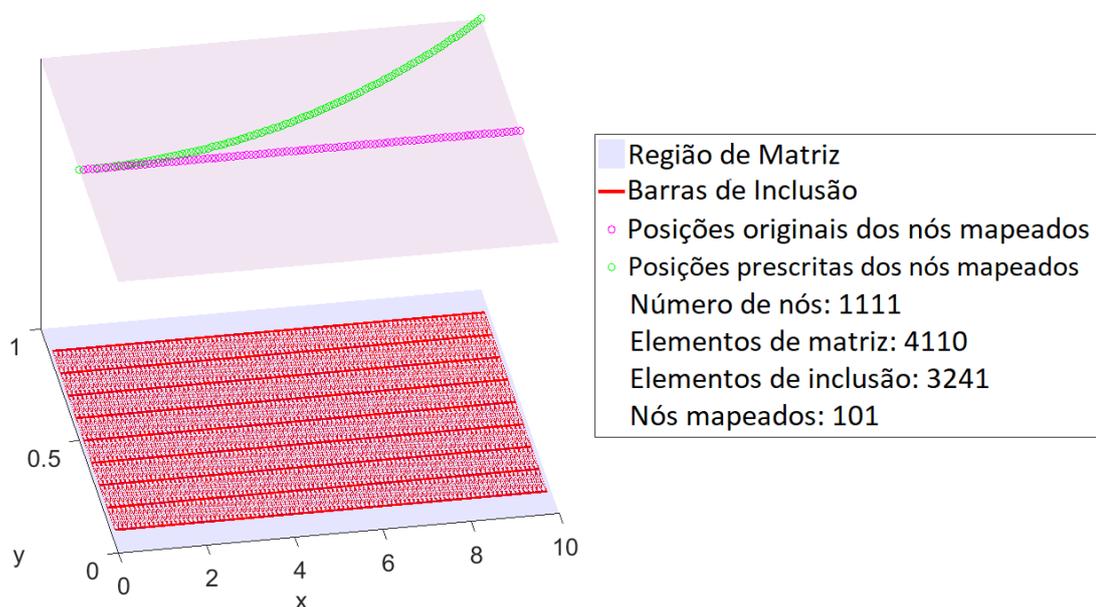
na qual  $(x_i, y_i)$  é a posição inicial do ponto (sua posição na configuração indeformada da estrutura), sendo  $(\bar{x}_f, \bar{y}_f)$  as posições finais prescritas e  $\alpha$  um parâmetro de escala que determina o quanto sobe a extremidade livre da linha média (ou seja: o ponto extremo da mesma à direita). Observa-se que apenas um deslocamento vertical é prescrito para os nós, e não um horizontal.

Neste caso, a Fig.4.7 mostra como fica a malha antes do processo de otimização, na qual a legenda à direita indica, na sequência, a região de matriz, representada contínua, em azul claro e na camada inferior, sob a qual são representadas as barras discretizada de inclusão em vermelho, sendo os nós mapeados indicados em magenta, na camada superior, e sendo a posição prescrita para os nós indicada em verde. As informações que seguem são o número de elementos de matriz (4410), de inclusão (3241), sendo as áreas dos mesmos as variáveis de projeto, e o número de nós mapeados (101). Vale salientar que neste, bem como em outros exemplos, os eixos, bem como as áreas da barras de inclusão, são representados fora de escala, por questão de visualização.

A configuração indeformada após o processo de otimização, por sua vez, é apresentada na Fig. 4.8 e objetiva ilustrar claramente a distribuição de material na peça da maneira como a mesma é impressa, antes das etapas de treinamento e liberação. Nela, os eixos horizontal e vertical são apresentados em mesma escala e a matriz em forma não discretizada é apresentada como uma região em azul, enquanto a inclusão é apresentada em sua forma discretizada em barras vermelhas. Por questões ilustrativas, a inclusão é apresentada fora de escala, de modo que se possa ver a distribuição de material claramente mesmo quando a mesma é muito pequena. Assim, a maior barra de inclusão é apresentada como tendo espessura do traço equivalente à metade do comprimento da menor barra. A matriz, por sua vez, quando apresentada, tem espessura equivalente a 1/5 desse valor, também meramente por questão ilustrativa.

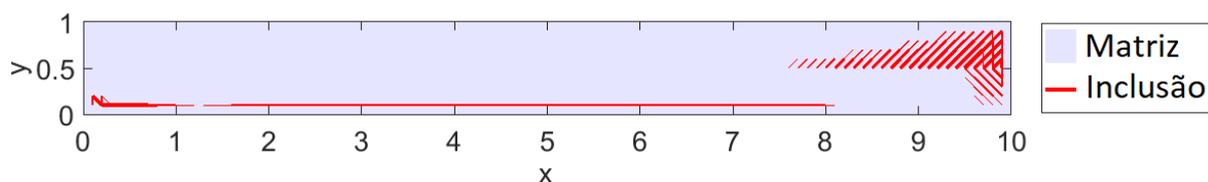
Para facilitar a visualização, bem como mostrar a comparação entre as malhas de matriz e inclusão otimizada, a Fig. 4.9 apresenta as duas discretizações em camadas distintas e fora de escala, com a dimensão vertical esticada de modo a se aproximar da horizontal. A malha de matriz é apresentada abaixo, em azul, com a de inclusão acima, em vermelho. Ambas são apresentadas na configuração indeformada.

Figura 4.7 – Discretização e informações do exemplo base



Fonte: Autor

Figura 4.8 – Configuração indeformada em e em escala do exemplo base

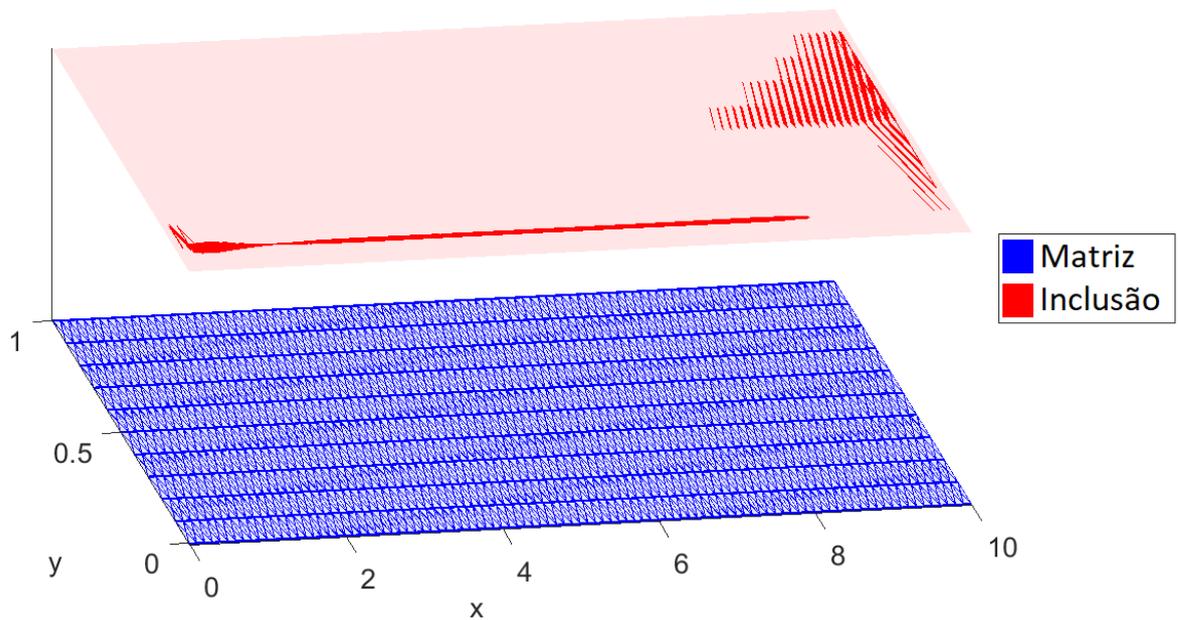


Fonte: Autor

Na sequência, apresenta-se também, na Fig. 4.10, a configuração deformada obtida, comparando-se a curva ao final do processo de otimização para a linha média da viga com a equação que foi prescrita como alvo para o otimizador. Assim como na figura anterior, os eixos são apresentados fora de escala por questão de visualização. Tal como anteriormente, a região em azul representa a matriz deformada não discretizada, enquanto as barras vermelhas são a inclusão no formato discreto. As curvas prescrita e obtida estão, respectivamente representadas, em preto e verde. A legenda  $|y|^2$  apresenta o valor final da função objetivo, sendo o quadrado da norma do vetor de diferenças ( $y$ ), de resultado 0,069087.

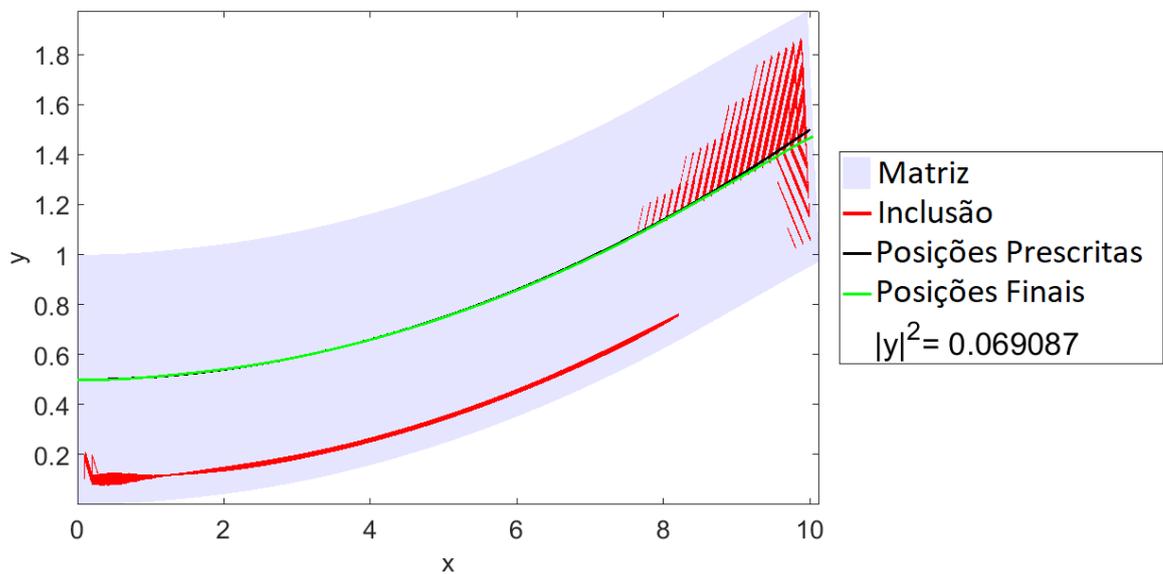
Por fim, apresenta-se o gráfico de convergência do algoritmo de otimização, sendo

Figura 4.9 – Configuração indeformada e em camadas do exemplo base



Fonte: Autor

Figura 4.10 – Configuração deformada do exemplo base

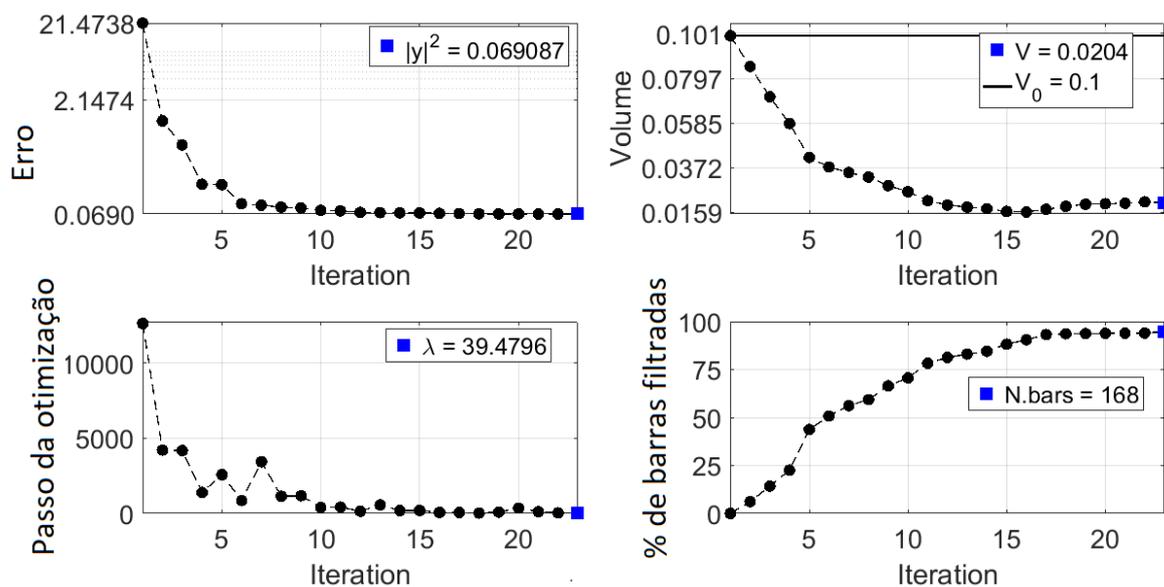


Fonte: Autor

o mesmo dividido em quatro subgráficos. No canto superior esquerdo, tem-se o erro no eixo vertical em escala logarítmica, sendo o eixo horizontal, assim como em todos

os demais plots, referente ao número de iterações realizadas. Abaixo desse subgráfico, tem-se o valor do parâmetro de amortecimento do Algoritmo de Levenberg-Marquardt, sendo o gráfico ao lado, no canto inferior direito, referente à porcentagem de barras filtradas. Por fim, o subgráfico do canto superior esquerdo compara o volume inicial de inclusão fornecido ao algoritmo de otimização ( $V_0$ ) em uma linha horizontal contínua com o valor corrente de cada iteração em linha pontilhada. Em todos os gráficos, as legendas internas representam os valores correntes das grandezas medidas em tempo real durante o processo de otimização e são, nas figuras apresentadas, por se tratarem dos últimos pontos analisados, os resultados finais das mesmas após a filtragem. Tais resultados são os pontos finais dos gráficos, em quadrados azuis. No caso do gráfico de barras filtradas, em particular, o valor apresentado na legenda é o número absoluto de barras ainda presentes, enquanto o eixo vertical mostra a porcentagem de barras filtradas.

Figura 4.11 – Gráficos de convergência do exemplo base



Fonte: Autor

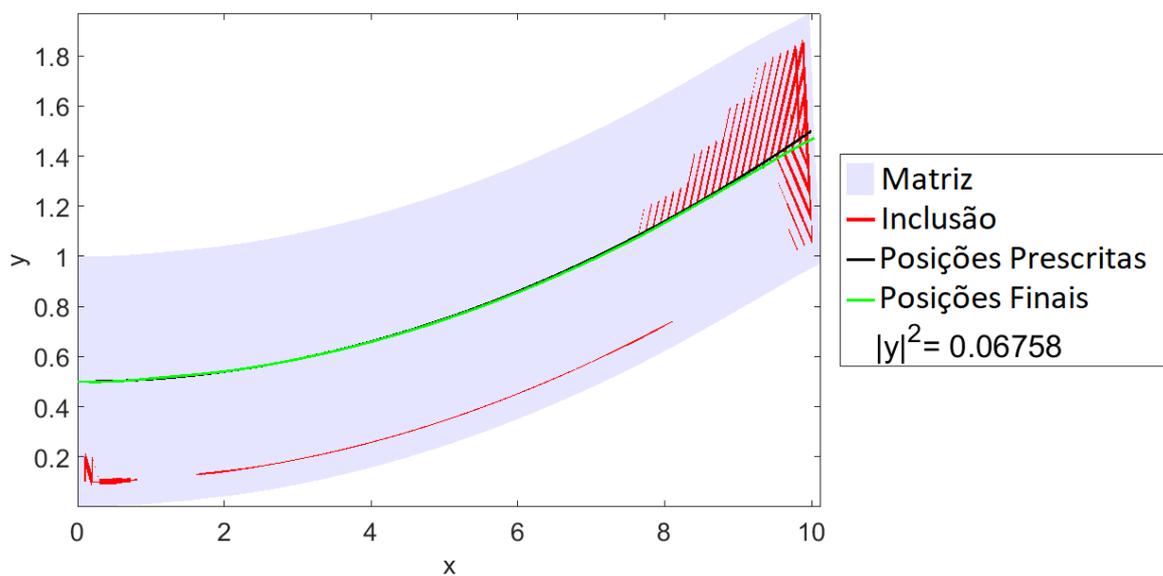
Para esta malha e parâmetros em particular, a avaliação da função objetivo leva aproximadamente 0,6 s, com o gradiente correspondendo a cerca de 0,15 s adicionais, ou cerca de 25% do custo combinado de ambos. Em comparação, sua avaliação via diferenças finitas centrais leva mais de 1 hora, um tempo impraticável, mesmo para a malha modesta empregada. O cálculo da Hessiana exata leva aproximadamente 180 s, enquanto sua versão truncada nos termos lineares (e que é efetivamente usada) tem custo de aproximadamente 8 s, ambos significativamente menores do que os necessários para a sua aproximação via diferenças finitas centrais, cujo custo é compatível (e ainda maior!) do que o cálculo do gradiente pela mesma estratégia. Vale salientar que esses valores são aproximados,

tomados com base em uma média de 5 avaliações.

#### 4.2.2 Efeito do Ponto Inicial

Nesta subseção, o efeito do ponto inicial na resposta do algoritmo é testado. Inicia-se por reduzir a fração volumétrica do 1% anterior, referente ao exemplo base, para 0,1%. Os resultados para a configuração deformada fora de escala e para os gráficos de convergência são apresentados, respectivamente, nas Figs. 4.12 e 4.13.

**Figura 4.12 – Configuração deformada com novo ponto inicial**

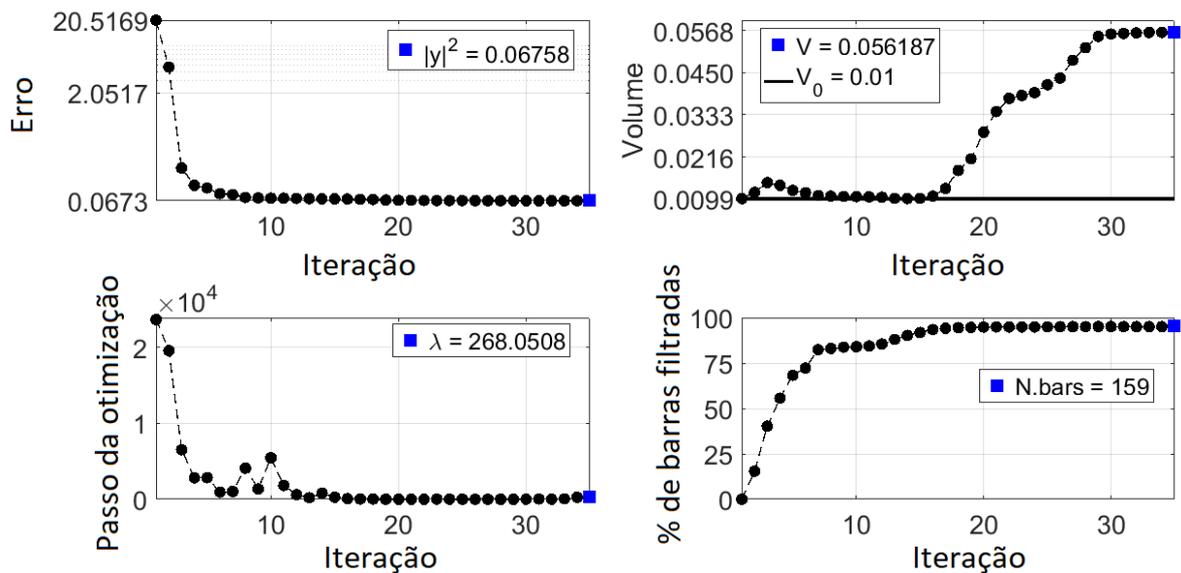


**Fonte: Autor**

Pode-se perceber que, para as tolerâncias consideradas, os resultados obtidos com os dois pontos iniciais são bastante distintos, com os valores referente à menor fração volumétrica acarretando em um erro menor e um menor número de barras, porém necessitando de mais iterações e de um volume de inclusão mais de duas vezes e meia maior. Uma conclusão importante que deve ser tirada desse resultado é que, ao menos para o algoritmo, estratégia de filtragem e critério de parada empregados neste trabalho, a resposta do algoritmo carrega importante dependência do ponto inicial adotado. As possíveis justificativas para esse resultado são muitas, incluindo a presença de regiões de difícil navegação para o algoritmo, a existência de múltiplos mínimos locais (ou pontos de sela), uma vez que o problema tratado é não convexo, o fato de que a eliminação de barras muda o problema tratado, etc.

Em ambos os casos (o desta subseção e o do exemplo base), a investigação da matriz Hessiana completa no ponto de solução indica a presença de autovalores tanto positivos

Figura 4.13 – Gráficos de convergência com novo ponto inicial



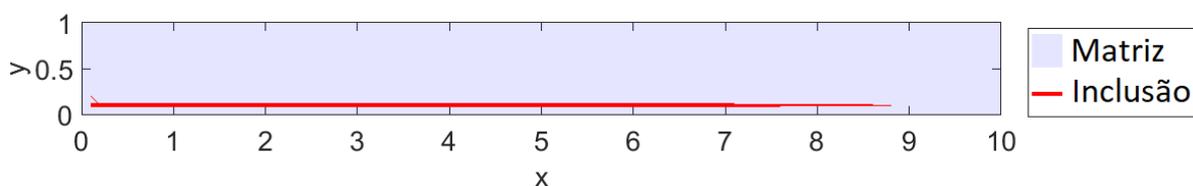
Fonte: Autor

quanto negativos. Mais que isso, os vetores gradientes obtidos têm valores muito pequenos se comparados com os dos primeiros passos do algoritmo, porém não nulos. Somadas essas observações ao fato de que respostas encontradas, apesar de bastante diferentes em termos de áreas e distribuições, têm geometrias e erros muito semelhantes, a hipótese mais razoável parece ser a de que o algoritmo encontra-se, de fato, navegando uma região dentro de um vale que é praticamente plana e dentro da qual o valor da função objetivo muda pouco. Essa hipótese é corroborada pelo fato dos maiores autovalores positivos serem várias ordens de magnitude maiores do que os maiores valores negativos em módulo, indicando que as direções de aumento geram impactos bem maiores no erro que as de decréscimo da função objetivo. Mais do que isso, vários autovalores são ainda várias ordens de magnitude menores do que tais valores extremos. A investigação desse fenômeno, a apresentação dos valores dos gradientes, dos autovalores da Hessiana e outros testes com o algoritmo serão feitos mais adiante no texto. Por agora, considera-se que os resultados obtidos são suficientemente bons, com erros considerados pequenos e distribuições de material fisicamente coerentes. De fato, os maiores erros encontrados no segundo caso tratado, referentes ao nó na extremidade direita da linha média, são diminutos. A posição prescrita para esse nó é de (10, 1,5), enquanto que o valor obtido é de aproximadamente 10,04, 1,4718. Em comparação com o deslocamento prescrito de 0,5 na direção vertical, o deslocamento encontrado representa um erro de menos de 6%. Outros pontos apresentam erros significativamente menores, como se pode perceber pelas figuras apresentadas, que mostram excelente concordância entre as linhas prescritas e obtidas.

## 4.2.3 Efeito do Filtro

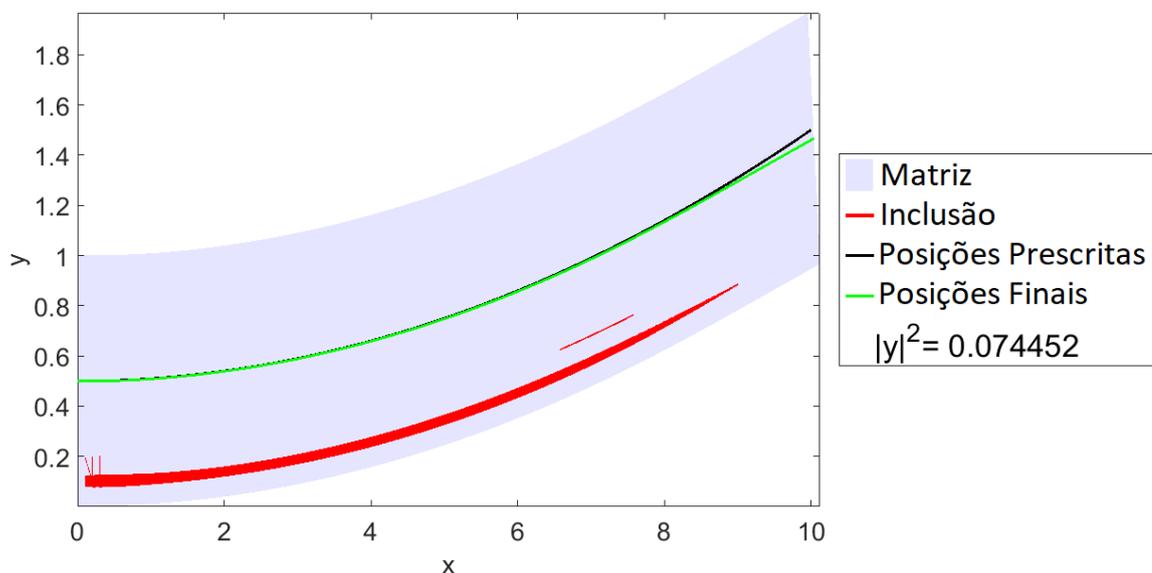
Apesar dos resultados anteriores terem sido considerados satisfatórios em termos do erro, pode-se perceber a presença de várias barras de inclusão dispersas, algumas das quais sendo bastante diminutas e tendo pouco efeito na resposta do algoritmo. De modo a corrigir esse problema, o último caso tratado, com volume inicial de inclusão de 0,1%, é resolvido novamente para um filtro de 10%. Os resultados obtidos para as configurações indeformada em escala e deformada são apresentadas, respectivamente, nas Figs. 4.14 e 4.15, bem como o gráfico de convergência, apresentado em na Fig. 4.16.

**Figura 4.14 – Configuração indeformada com novo filtro**



Fonte: Autor

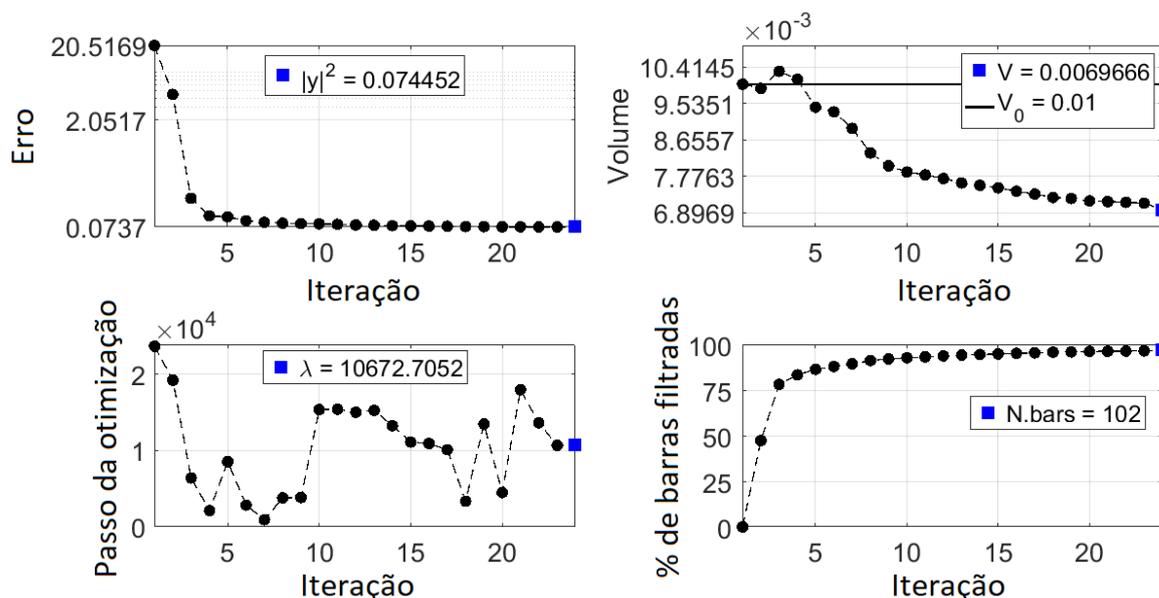
**Figura 4.15 – Configuração deformada com novo filtro**



Fonte: Autor

Em comparação com o exemplo anterior, nota-se uma enorme redução do volume de inclusão de quase 90%, um aumento do erro de pouco mais de 10% e uma redução no número total de barras, resultando em uma geometria bem mais limpa. Mais ainda, foram

Figura 4.16 – Gráficos de convergência com novo filtro



Fonte: Autor

necessárias menos iterações e mais barras foram eliminadas mais cedo durante a otimização, resultando em menores tempos computacionais, uma vez que o custo do processo de otimização depende do número de passos e o custo de cada passo depende drasticamente do número de barras, tornando-se muito mais rápido à medida que as mesmas vão sendo eliminadas. Tal efeito, contudo, foi ligeiramente mitigado pelo fato de que, em um maior número de iterações, houve a necessidade de se reverter os resultados anteriormente obtidos e aumentar o valor de  $\lambda$ , conforme discutido durante a formalização do algoritmo. Apesar disso, os tempos computacionais obtidos ainda foram consideravelmente menores.

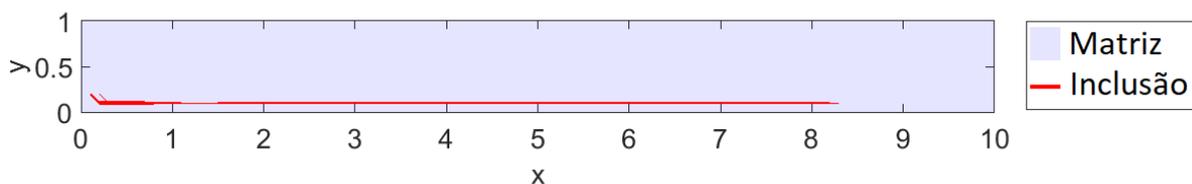
Apesar da geometria estar muito mais limpa, nota-se ainda a presença de pequenas barras, que poderiam ter sido eliminadas.

#### 4.2.4 Efeito do Critério de Parada

Observando-se o passo a passo das iterações realizadas pelo algoritmo no exemplo anterior, percebe-se que as barras finas que sobraram ao fim da otimização, em sua maioria, vinham em uma tendência de redução. Acredita-se, portanto, que o processo de otimização foi interrompido muito cedo e que, se autorizado a continuar, poderia livrar-se das mesmas. Para testar essa hipótese, bem como averiguar a possibilidade de um erro de fechamento menor, o exemplo anterior é analisado mais uma vez, porém tendo como critério de parada que o algoritmo interrompa o cálculo de novos passos apenas quando

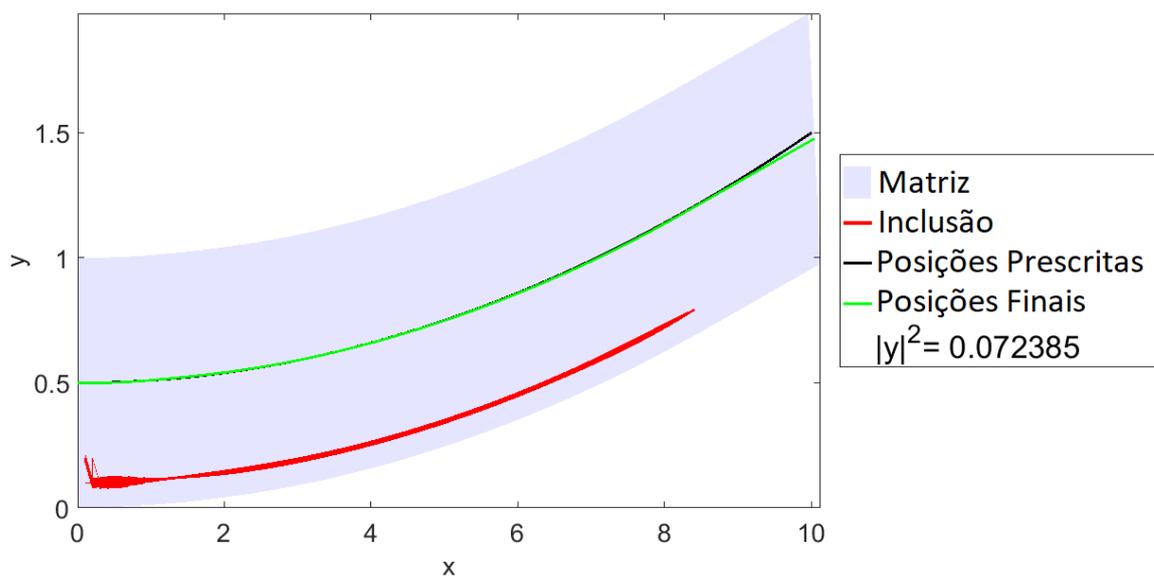
uma mudança menor do que  $10^{-6}$  seja atingida ao longo de 5 iterações. Esse critério, em contraste com o de antes, de 1%, representa uma mudança de  $10^{-4}\%$ , portanto quatro ordens de magnitude menor. Para o mesmo, os resultados em termos de configuração indeformada em escala e deformada são apresentados nas Figs. 4.17 e 4.18, bem como o gráfico de convergência na Fig. 4.19.

**Figura 4.17 – Configuração indeformada com novo critério de parada**



Fonte: Autor

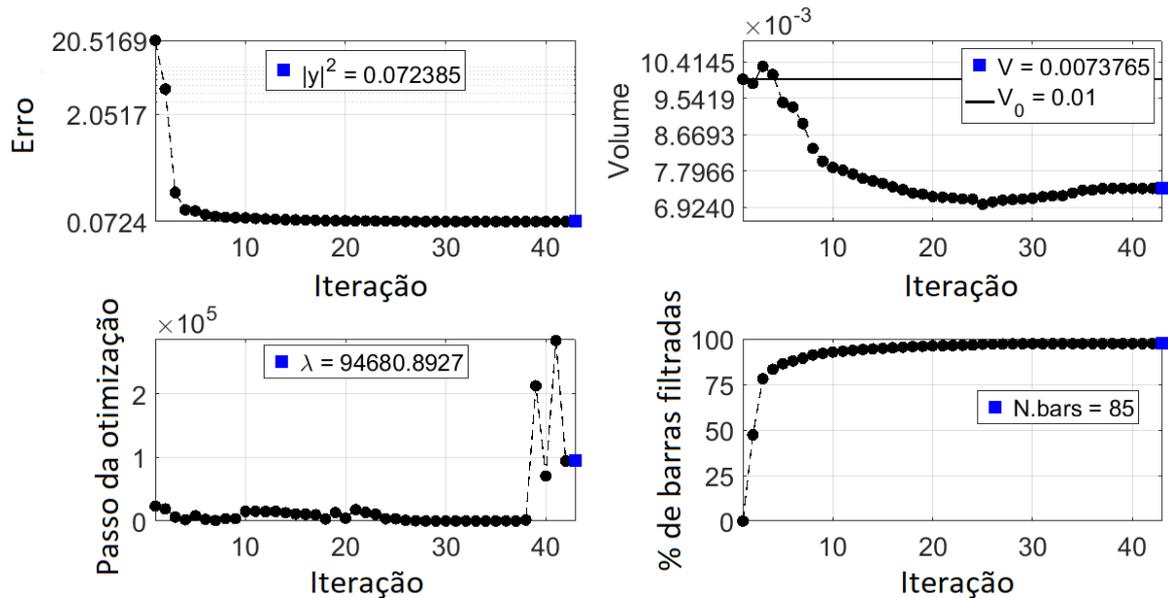
**Figura 4.18 – Configuração deformada com novo critério de parada**



Fonte: Autor

Percebe-se que, assim como esperado, uma parte das barras finas desapareceu com os passos adicionais necessários para que o novo critério de parada seja atingido. O erro encontrado também é menor, ainda que esteja mais próximo do resultado anterior do que do de menor filtro. Salienta-se que, apesar do maior número de iterações, como as barras são eliminadas mais cedo com o filtro maior, esse resultado ainda demanda menos tempo computacional do que com o filtro de 1%, mesmo com um critério de parada muito mais estrito.

Figura 4.19 – Gráficos de convergência com novo critério de parada



Fonte: Autor

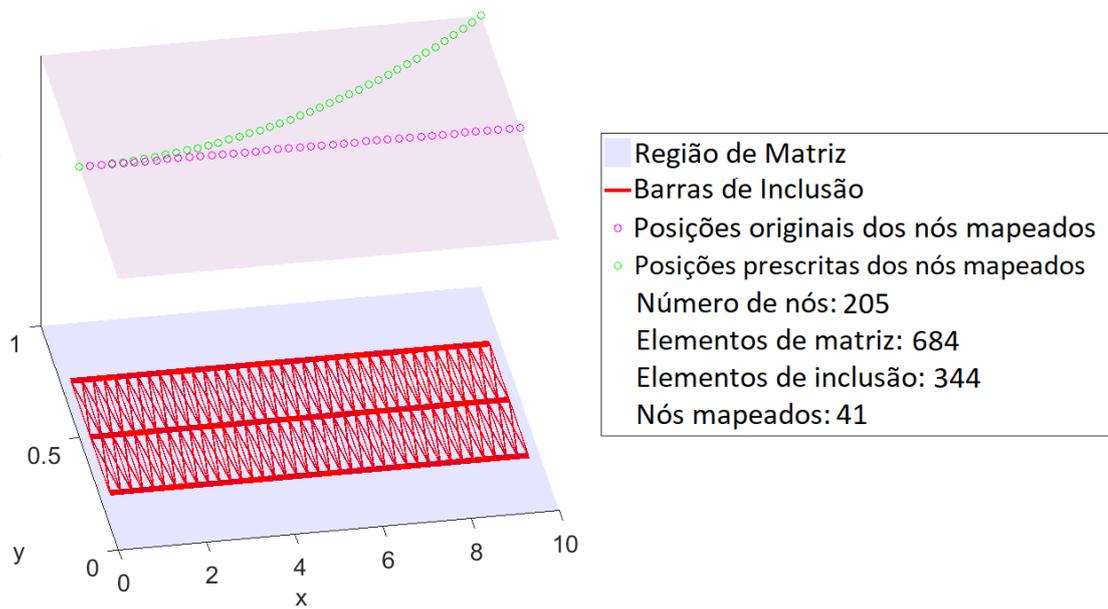
## 4.2.5 Efeito da Malha

Até então, tem-se fixado a malha tratada com um valor que é um equilíbrio entre o custo computacional e consumo de memória dos exemplos com a qualidade das soluções. De modo a ilustrar o efeito da malha na resposta, o exemplo anterior é analisado novamente, porém com 41 nós na horizontal e 5 na vertical, em contraste com os 101 nós na horizontal e 11 na vertical anteriormente utilizados. Para este caso, a malha antes do processo de otimização, bem como as informações associadas à mesma, são apresentadas na Fig. 4.20, que indica a presença, neste exemplo, de 684 barras de matriz, 344 de inclusão e 205 nós ao total, sendo 41 nós mapeados.

Os resultados obtidos para este caso são apresentados, para as configurações indeformada em camadas (com as barras de matriz explicitadas) e deformada, respectivamente, nas Figs 4.21 e 4.22

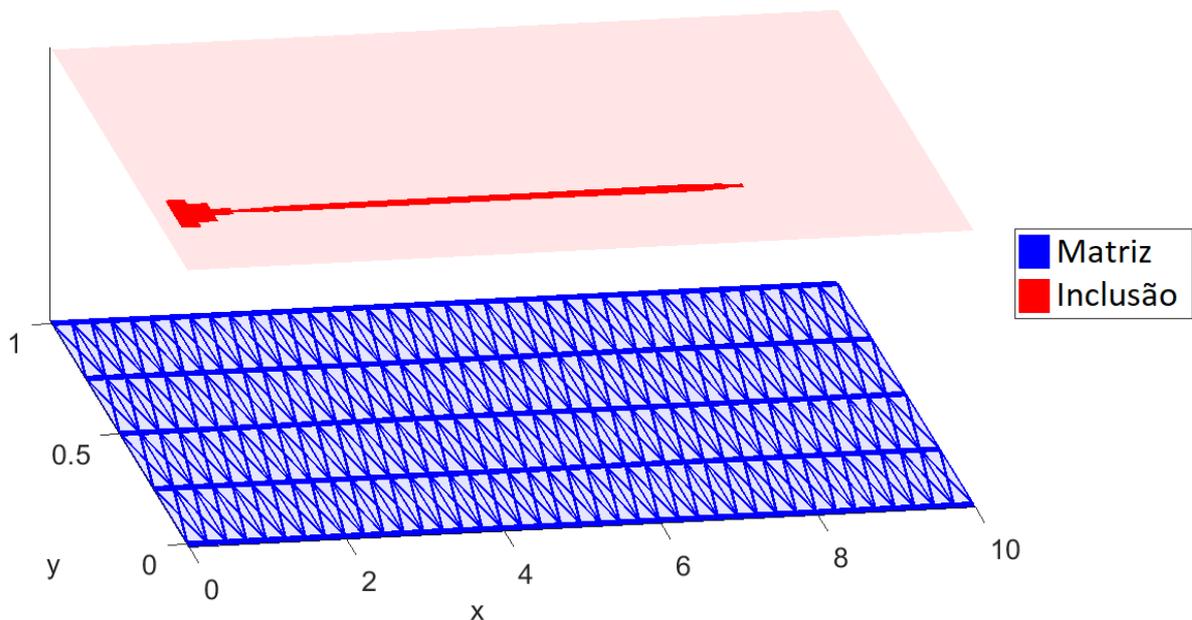
Nota-se que, mesmo com um número bem menor de nós para serem mapeados, o que resultaria, por si só, em menores valores para a função objetivo, o resultado obtido para a mesma é significativamente maior, mostrando uma perda expressiva na qualidade das respostas obtidas, como pode ser observado pelo gráfico de convergência na Fig. 4.23, que mostra que, para o passo inicial, o erro da malha mais grossa, pelo menor número de pontos, é também menor, mas que o mesmo jamais reduz até o mesmo patamar que o anterior.

Figura 4.20 – Discretização e valores iniciais com nova malha



Fonte: Autor

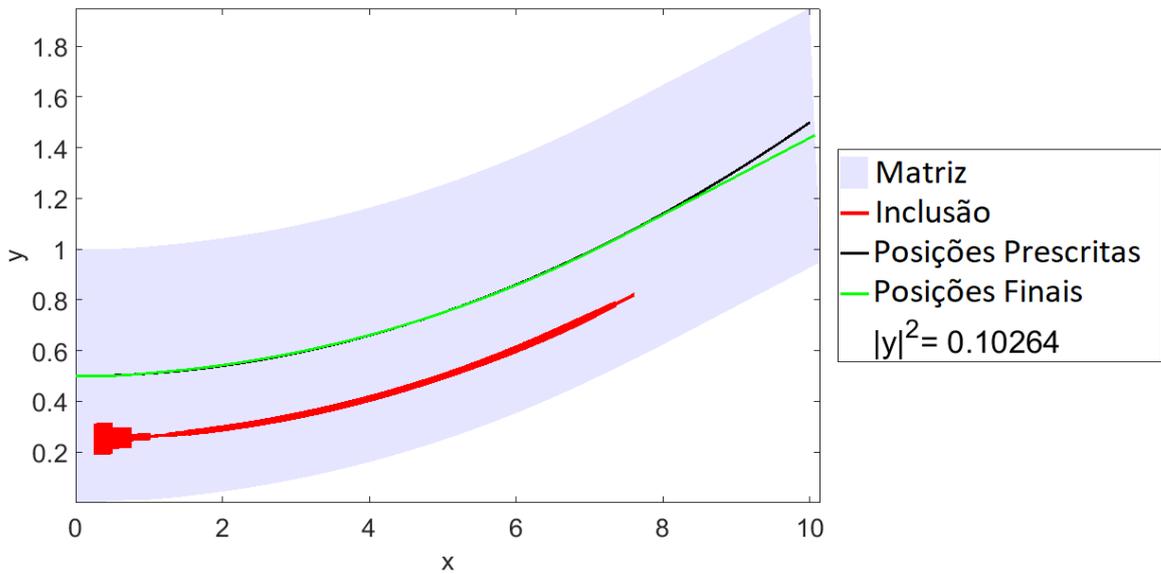
Figura 4.21 – Configuração indeformada e em escala com nova malha



Fonte: Autor

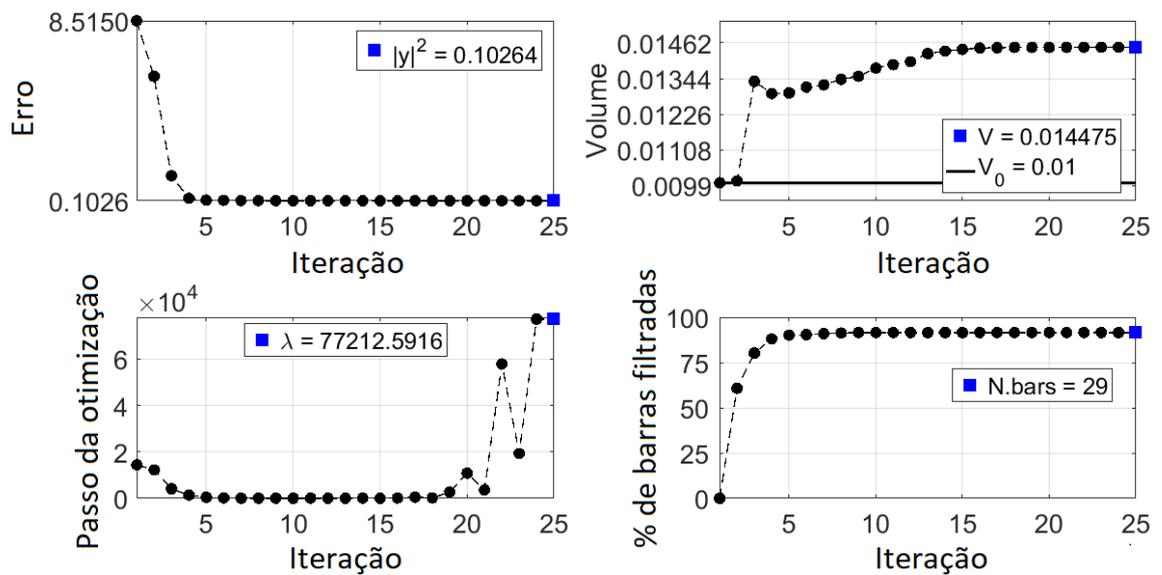
Analisando-se o gráfico de convergência, nota-se que, para esta malha mais grossa, tem-se uma convergência mais rápida e com o valor do amortecimento mais estável, com

Figura 4.22 – Configuração deformada com nova malha



Fonte: Autor

Figura 4.23 – Gráficos de convergência com nova malha



Fonte: Autor

um menor número de ajustes e de iterações revertidas. Tem-se ainda uma filtragem mais rápida e resultados que se estabilizam com menos iterações, mostrando que, por um lado, o aumento da malha melhora a qualidade das respostas, mas, por outro, onera o

algoritmo de otimização. Os tempos computacionais, neste caso, são bastante reduzidos em comparação com os anteriores.

Para esta malha reduzida, o cálculo da função objetivo leva aproximadamente 0,09 s. Já o gradiente leva aproximadamente 0,011 s adicionais, cerca de 15% do custo combinando de ambos. Em comparação, sua avaliação via diferenças finitas centrais leva aproximadamente 58 s, um tempo significativamente maior. O cálculo da Hessiana exata leva aproximadamente 2,2 s adicionais, enquanto sua versão truncada nos termos lineares tem custo de aproximadamente 0,019 s, ambos ainda significativamente menores do que os necessários para a sua aproximação via diferenças finitas centrais, com custo medido em aproximadamente 65 s. Vale salientar que esses valores são aproximados, tomados com base em uma média de 5 avaliações.

#### 4.2.6 Efeito da Curva Prescrita

Até então, tem-se prescrito uma geometria para linha média que é relativamente simples em termos de equação, constituindo um polinômio de segundo grau. De modo a se testar o efeito de geometrias mais complexas na resposta do algoritmo, prescreve-se uma curva para a linha média de equação

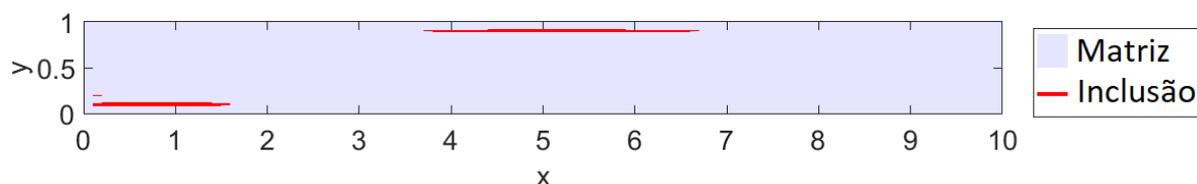
$$\bar{x}_f(x_i, y_i) = x_i \quad (4.6a)$$

$$\bar{y}_f(x_i, y_i) = y_i + \frac{\alpha}{4} \left[ 1 + \sin \left( \frac{2\pi}{L_x} x - \frac{\pi}{2} \right) \right] \quad (4.6b)$$

na qual a prescrição para o deslocamento  $\Delta y(x)$  é, agora, uma curva senoidal. Salienta-se que a mesma é construída de tal forma que, em  $x = 0$ , a derivada com relação a  $x$  é nula, respeitando-se a condição de engastamento de viga emulada na extremidade esquerda da malha. Os resultados obtidos para este caso, usando-se a malha original de 101x11 e mantendo-se todos os demais parâmetros do exemplo anterior, são, respectivamente para a configuração indeformada em escala e deformada, apresentados nas Figs. 4.24 e 4.25. O único parâmetro modificado é o de filtro, que tem efeitos muito diferentes para cada caso e, neste exemplo em particular, apresentou-se com resultados melhores para 20% do que para os 10% anteriores. Além das configurações deformada e indeformada, os gráficos de convergência são apresentados na Fig. 4.26.

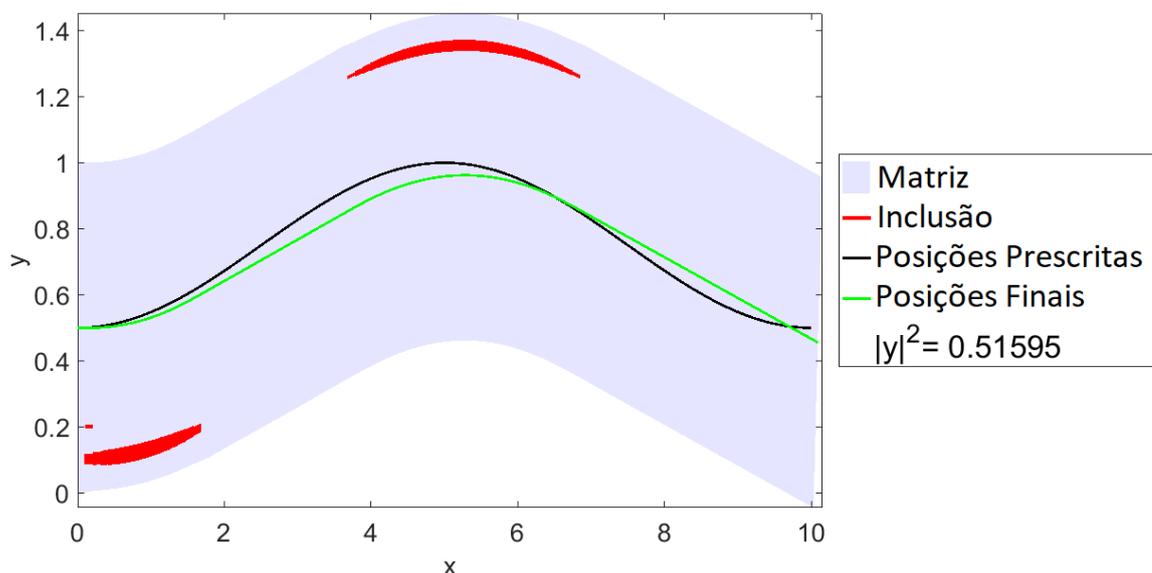
Como a configuração em questão é claramente mais complexa de ser mapeada, trocando a curva de segunda ordem por uma senoide, o erro obtido, como previsto, é maior. Porém, observa-se, ainda, uma boa concordância entre a resposta prescrita e a obtida, com a curva encontrada pelo algoritmo aproximando a geometria desejada. Ainda assim, como o esforço de flexão necessário para curvar a viga exatamente como

Figura 4.24 – Configuração indeformada em nova geometria prescrita



Fonte: Autor

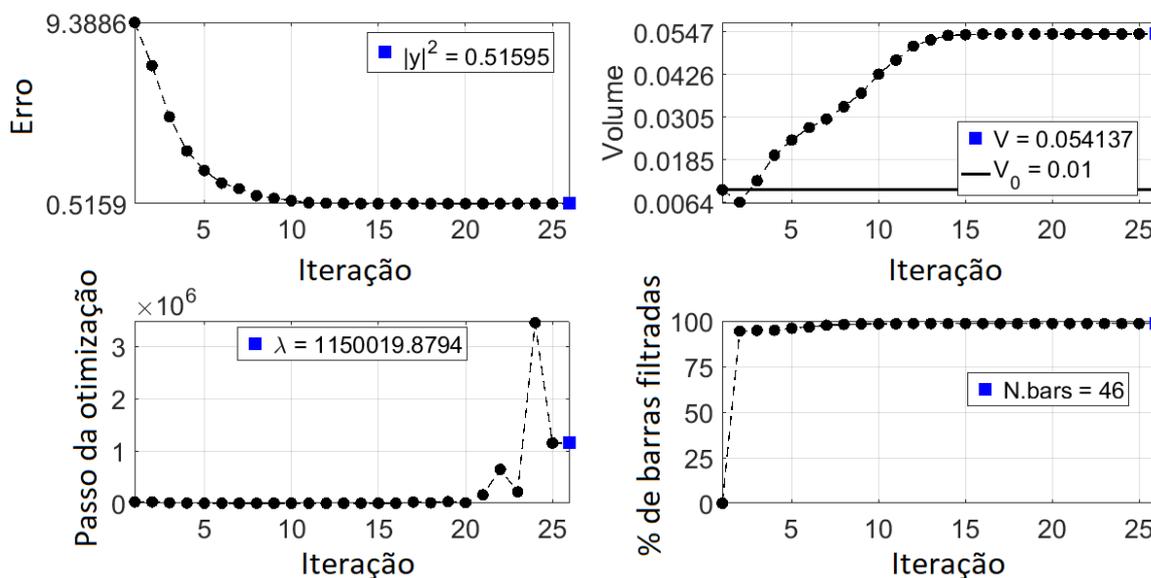
Figura 4.25 – Configuração deformada em nova geometria prescrita



Fonte: Autor

requerido é bastante grande, o algoritmo aproxima a última parte da senoide por um trecho final quase reto, no qual a posição prescrita para os pontos é, por vezes, superior à encontrada, enquanto, em outras, fica por baixo da mesma. Mais ainda, nota-se uma significativa discrepância entre a rotação nula imposta na extremidade livre à direita o valor encontrado. Assim como esperado, o algoritmo aloca material na parte inferior esquerda da viga para a subida, assim como no exemplo anterior, mas, neste caso em particular, também aloca material na parte superior central, garantindo a descida vista na curva senoide. Neste caso, assim como no anterior, os erros maiores encontram-se na porção direita do elemento. Isso pode ser explicado por conta da maior distância entre as barras de inclusão à direita e o apoio. Nesta configuração, as barras próximas ao engaste têm seu efeito propagando ao longo de uma porção maior da barra e os esforços gerados pelas mesmas têm mais influência na configuração da viga, portanto o ajuste nas mesmas

Figura 4.26 – Gráficos de convergência em nova geometria prescrita



Fonte: Autor

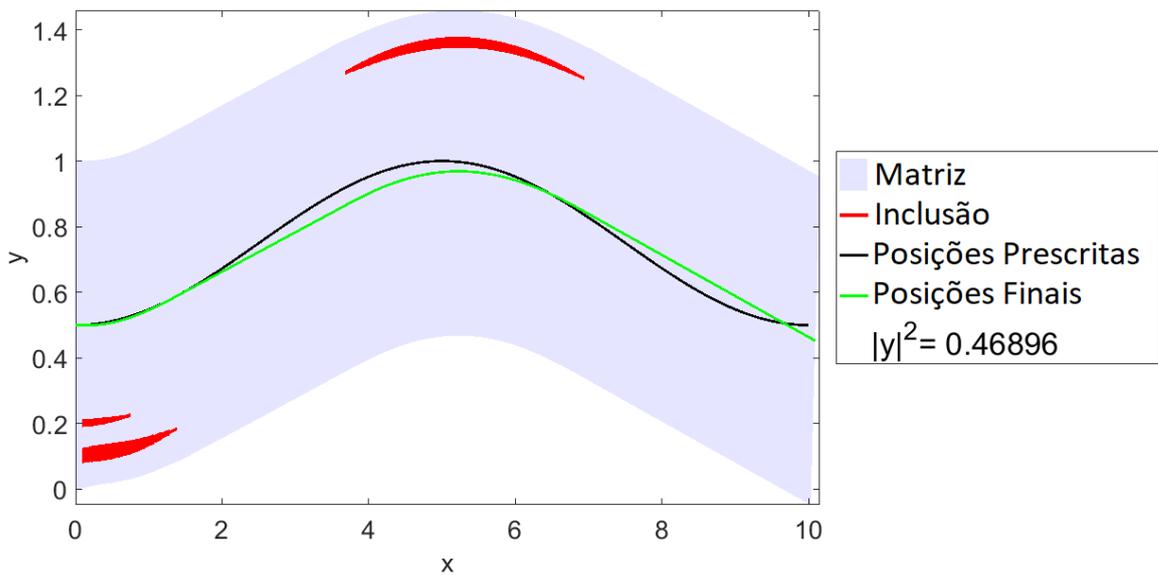
é mais efetivo em encontrar configurações melhor aproximadas.

#### 4.2.7 Efeito da Deformação de Treinamento

Em todos os exemplos anteriores, o esticção inicial dado na estrutura tem sido de 1, conforme apresentado no início do capítulo. Porém, acredita-se que uma nova deformação de treinamento de valor maior possa ajudar o algoritmo de otimização a encontrar uma configuração melhor para o exemplo anterior, uma vez que permite ao mesmo gerar mais tensões na matriz para deformá-la. Para testar esta hipótese, faz-se um esticção inicial duas vezes maior, portanto  $\bar{u}_x = 2,0$  em todos os nós da extremidade direita. Os demais parâmetros do modelo são todos mantidos iguais e os resultados encontrados neste caso são apresentados nas Figs. 4.27 e 4.28, respectivamente tratando da configuração deformada e dos gráficos de convergência.

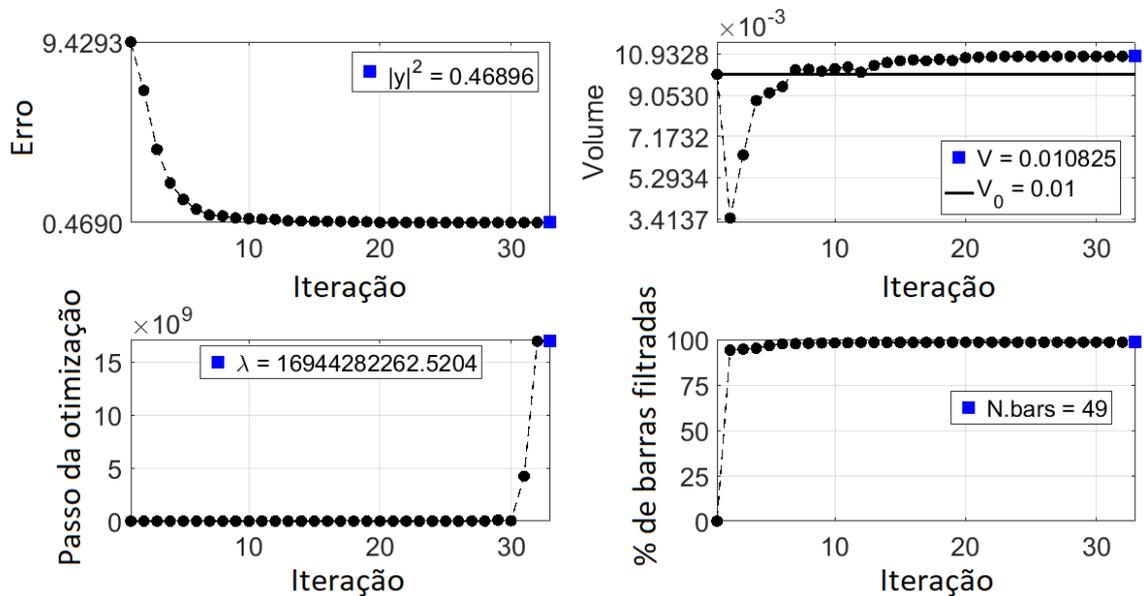
Percebe-se que, assim como previsto, um parâmetro de esticção maior permite ao algoritmo curvar a matriz mais facilmente, resultando em um erro menor, bem como a utilização de 5 vezes menos material de inclusão. Como as barras todas são menores, neste caso, do que exemplo anterior, tem-se ainda que o filtro se comporta de maneira diferente, deixando duas camadas de inclusão ao invés de uma na parte inferior esquerda. Isso reforça a afirmação feita de que o valor apropriado para o filtro depende largamente do problema tratado, com pequenas mudanças em outros parâmetros afetando a maneira como o mesmo se comporta.

Figura 4.27 – Configuração deformada com nova deformação de treinamento



Fonte: Autor

Figura 4.28 – Gráficos de convergência com nova deformação de treinamento



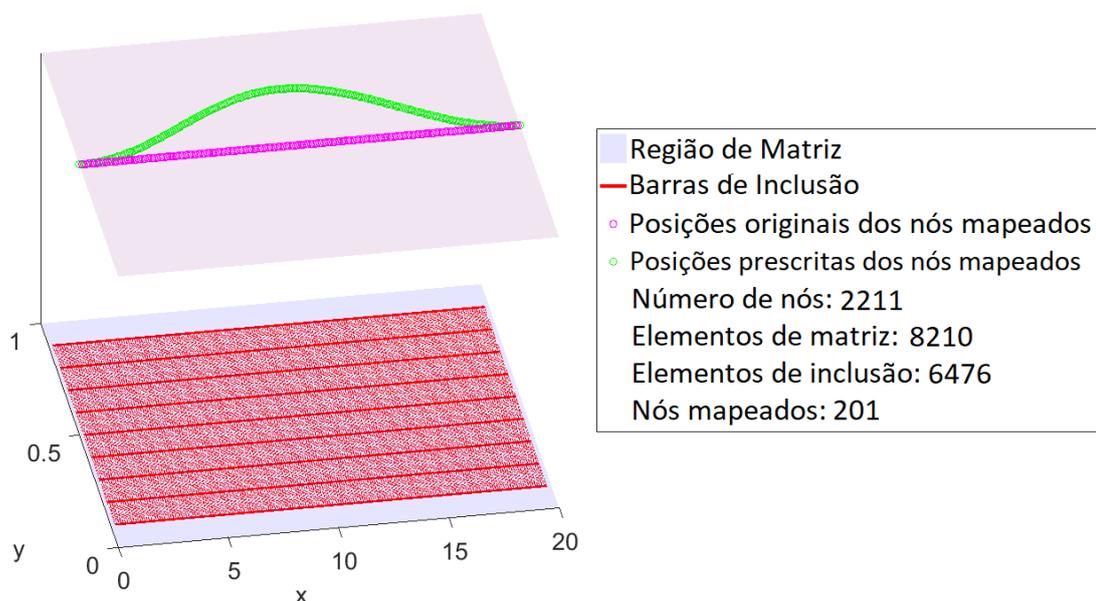
Fonte: Autor

## 4.2.8 Efeito da Geometria

Assim como o aumento do parâmetro de esticção permite ao algoritmo dobrar a matriz mais facilmente, espera-se que fazer a viga mais flexível produza o mesmo efeito.

Mais ainda, aumentar o comprimento da viga discretizada, além de garantir tal redução da rigidez, faz com que seja muito mais fácil para o algoritmo fazer a subida e descida prescritas, reduzindo as curvaturas envolvidas no problema. Para testar esse efeito, faz-se  $L_x = 20$ , ao invés do valor de 10 prescrito até então. Para evitar anisotropia, conforme discutido anteriormente, aumenta-se a discretização ao longo do eixo  $x$  de  $n_x = 101$  para  $n_x = 201$ , portanto dobrando-se o número de divisões ao longo do eixo e mantendo-se o tamanho das mesmas. Todos os demais parâmetros, inclusive o esticão anteriormente modificado, são mantidos iguais. O resultado para a malha pré-otimização e os valores iniciais associados à mesma é apresentada na Fig. 4.29, que indica a presença, neste exemplo, de 8210 barras de matriz, 6467 barras de inclusão e um total de 2211 nós, sendo 201 nós mapeados.

**Figura 4.29 – Discretização e informações iniciais com nova geometria**

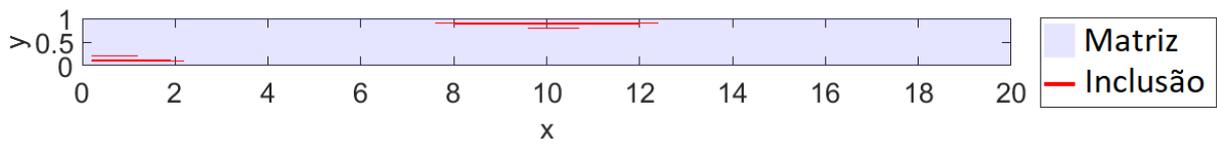


**Fonte: Autor**

Na sequência, a Fig. 4.30 apresenta a configuração indeformada em escala e com a matriz não-discretizada. Em contraste, a Fig. 4.31 apresenta a mesma configuração indeformada, porém fora de escala e em camadas, com as barras de matriz explicitadas, para que se possa ver a discretização utilizada. Já a Fig. 4.32, por sua vez, apresenta a configuração deformada, também fora de escala, com as curvas prescrita, encontrada e o erro final. Por fim, a Fig. 4.33, mostra os gráficos de convergência.

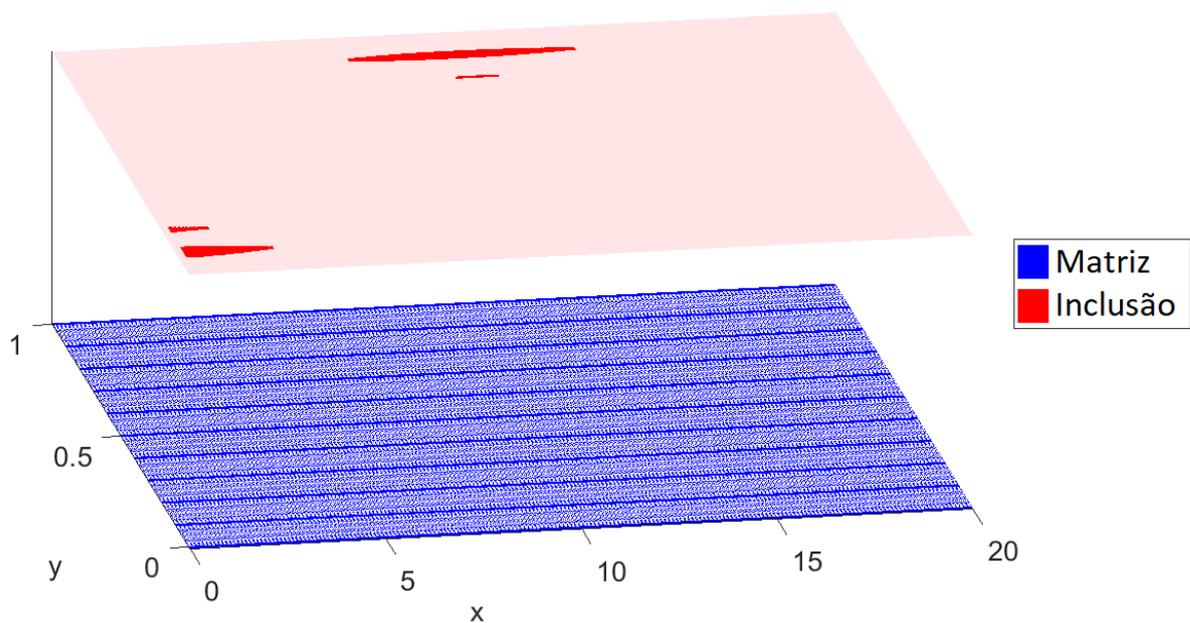
Percebe-se que, assim como esperado, o algoritmo é capaz de encontrar resultados bem melhores, com erros significativamente menores, mesmo para um conjunto de pontos mapeados duas vezes maior, um menor volume de inclusão, mesmo para uma barra mais

**Figura 4.30 – Configuração indeformada e em escala com nova geometria**



Fonte: Autor

**Figura 4.31 – Configuração indeformada e em camadas com nova geometria**

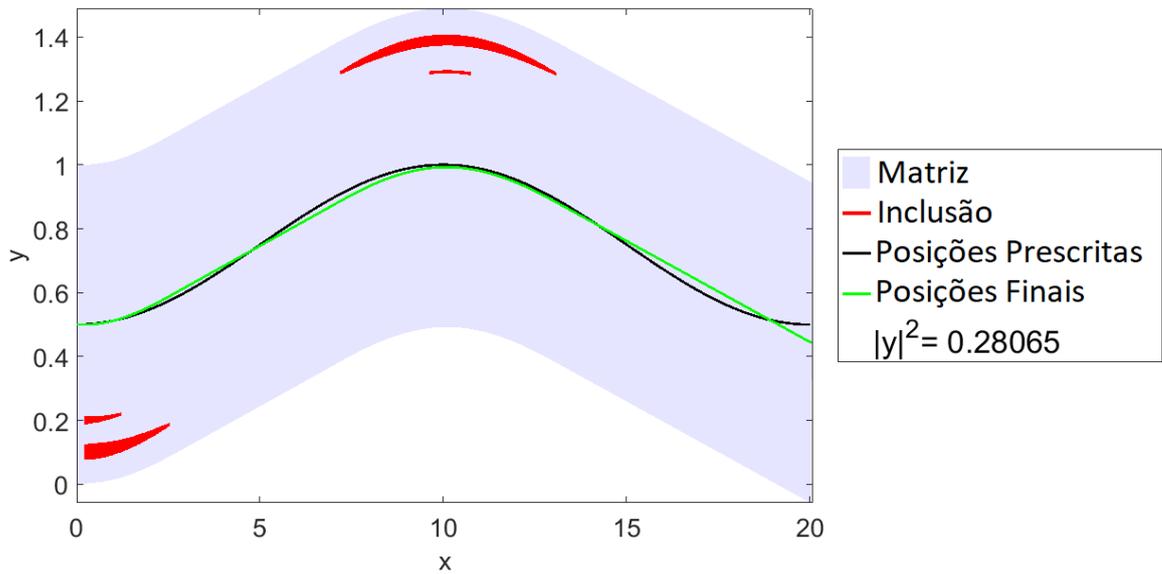


Fonte: Autor

longa, e com uma maior concordância entre as curvas prescrita e calculada. Seguindo com as vantagens, são necessárias menos iterações para encontrar este resultado, em comparação com o anterior, mesmo com o número inicial de variáveis de projeto sendo quase duas vezes maior do que no caso anterior (6476, em comparação com 3242), mostrando o grande impacto da geometria da matriz na resposta do algoritmo.

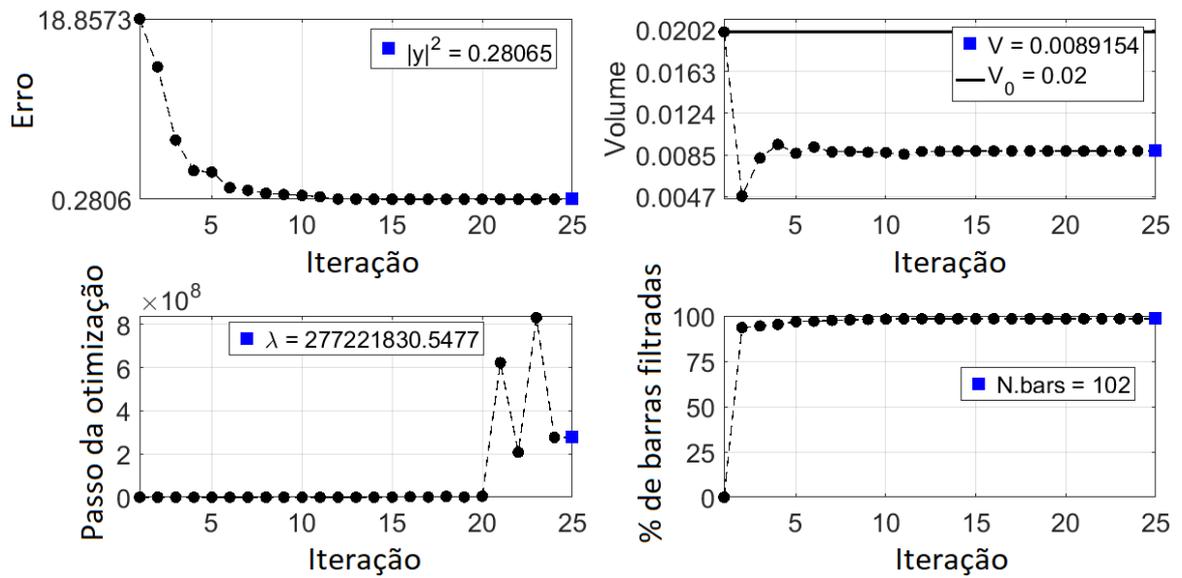
Para esta malha com mais elementos, o cálculo da função objetivo leva aproximadamente 1,3 s. Já o gradiente leva aproximadamente 0,25 s adicionais, cerca de 20% do custo combinando de ambos. O cálculo da Hessiana truncada nos termos lineares tem custo de aproximadamente 41 s adicionais. Os demais custos não foram averiguados e os valores apresentados são, novamente, a média de 5 medições.

Figura 4.32 – Configuração deformada com nova geometria



Fonte: Autor

Figura 4.33 – Gráficos de convergência com nova geometria



Fonte: Autor

4.2.9 O Ponto de Mínimo

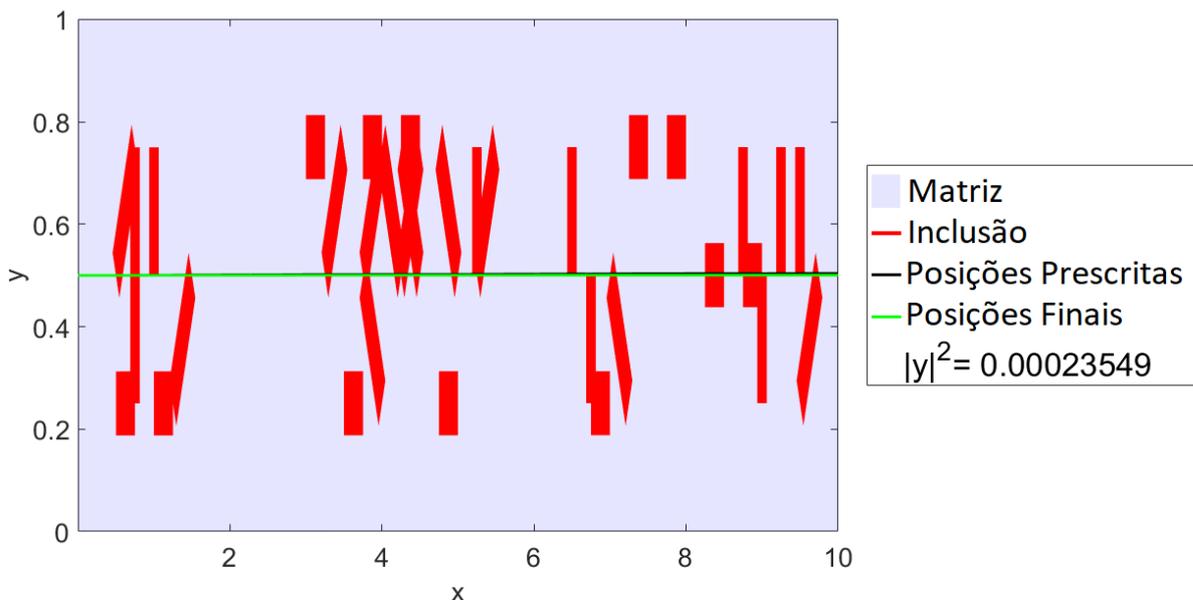
Até então, tem-se estabelecido critérios de parada para o algoritmo e tomado a resposta dada como o melhor resultado possível dentro de tais critérios. Nesta subseção,

contudo, tem-se um objetivo diferente: deseja-se descobrir se é efetivamente possível, para o problema analisado, encontrar um ponto de mínimo (dentro de uma certa tolerância numérica), ao menos para certos casos específicos. Mais ainda, deseja-se comparar a resposta obtido com valores conhecidos, tal que se possa medir a qualidade da solução encontrada. Por fim, deseja-se ainda investigar os critérios de qualificação do ponto-candidato, de maneira que se possa determinar o quão precisas são as estratégias numéricas utilizadas para tal fim. Para tanto, retoma-se a malha de 40x4 divisões utilizada anteriormente. Contudo, dos 344 elementos de inclusão nela presentes, 34 (aproximadamente 1/10 do total) são escolhidos aleatoriamente e preservados, enquanto os demais são desconsiderados. Para estes elementos, as áreas inicialmente obtidas, calculadas conforme a metodologia anteriormente apresentada, também foram modificadas de maneira aleatória, tendo novos valores atribuídos as mesmas conforme uma distribuição estatística homogênea com limites de 10% e 1000% de seus valores originais. Para estes valores aleatórios de áreas em elementos também aleatoriamente selecionados, a configuração deformada dos pontos discretos ao longo da linha média da viga foi determinada e subsequentemente aplicada como configuração prescrita para o algoritmo de otimização. Em seguida, o algoritmo de otimização foi aplicado ao problema, dado como ponto inicial de partida do mesmo as áreas homogêneas (não randomizada) dos elementos selecionados. Com isso, objetiva-se prescrever para o algoritmo de otimização uma configuração geométrica da linha média qualquer, contanto que a mesma fosse possível de ser obtida de maneira exata (e, portanto, com erro quadrático nulo), ao invés de uma geometria prescrita conhecida para o qual tal garantia não existe. Mais ainda, os valores randômicos das áreas que geraram a configuração prescrita qualquer são exatamente o ponto de mínimo do problema, de tal maneira que os resultados obtidos pelo algoritmo de otimização, partindo o mesmo das áreas homogêneas (portanto largamente diferente do ponto de mínimo), podem ser diretamente comparados com os mesmos.

De forma a permitir que o algoritmo de otimização encontre o ponto de mínimo conhecido, algumas medidas são tomadas. Primeiro, o filtro de áreas é retirado, impedindo que elementos necessários sejam permanentemente excluídos durante a trajetória percorrida pelo algoritmo. Neste caso, quando as restrições de área mínima são ativadas, tais áreas são simplesmente consideradas como sendo iguais às áreas mínimas prescritas e o algoritmo de otimização segue para a próxima iteração. Para evitar que tal procedimento seja frequentemente necessário, contudo, considera-se como área mínima o valor de  $10^{-20}$  vezes a maior área encontrada, portanto um valor muito pequeno. Mais ainda, o critério de parada foi redefinido, de tal forma que o algoritmo só encerra a busca pelo ponto de mínimo quando a mudança na função objetivo é menor que  $10^{-20}$  do seu próprio valor ao longo de 5 iterações. Para este critério, a Fig. 4.34 mostra a configuração deformada da viga no ponto inicial de áreas homogêneas, bem como a configuração prescrita para a

linha média, obtida através do resultado da distribuição de áreas aleatórias. Vale salientar que, uma vez que a distribuição de áreas e barras é aleatória, e portanto não há direção ou região preferencial para a colocação das barras de dimensões específicas, a configuração deformada é bastante próxima da indeformada, como fica evidenciado na figura, devido aos esforços fletores internos se auto-cancelarem.

**Figura 4.34 – Configuração deformada no ponto inicial**



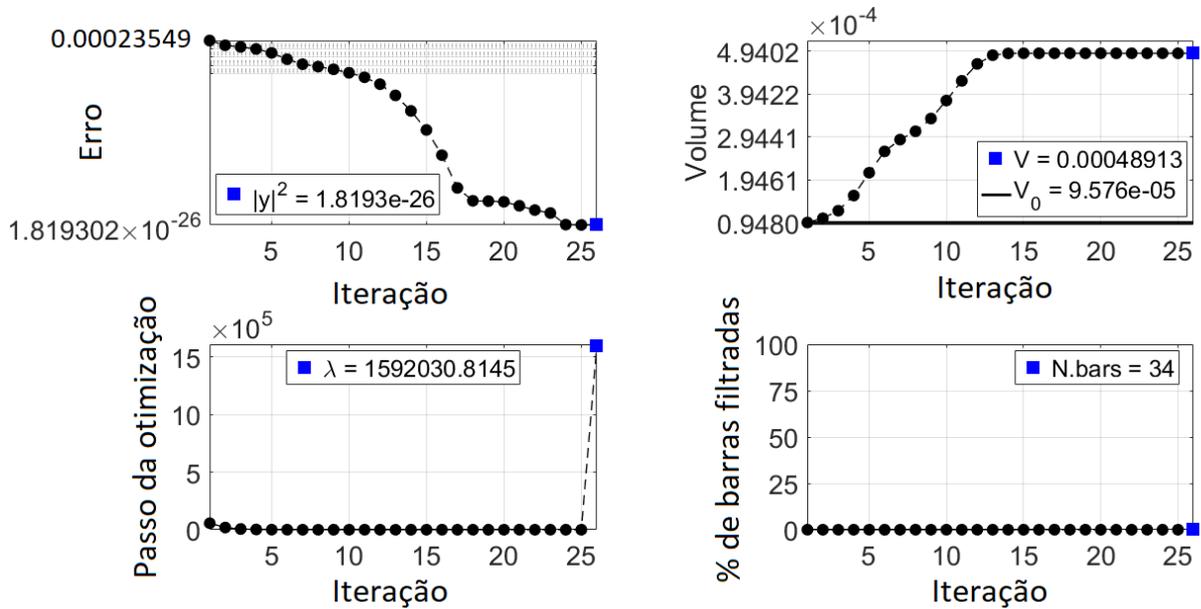
**Fonte: Autor**

Na sequência, a Fig. 4.35 mostra os gráficos de convergência ao longo das 26 iterações realizadas, que se encerraram com base no critério de parada de 5 tentativas de iteração seguidas sem redução da função objetivo, cujo valor final foi de  $1,819302 \cdot 10^{-26}$  (portante bastante pequeno). Salienta-se, mais uma vez, que o gráfico com o valor da função objetivo ao longo das iterações é apresentado em escala logarítmica.

A Fig. 4.36, por sua vez, mostra o resultado final encontrado na configuração indeformada, com a camada superior indicando as áreas dos elementos de inclusão e com a camada inferior apresentando a discretização da matriz.

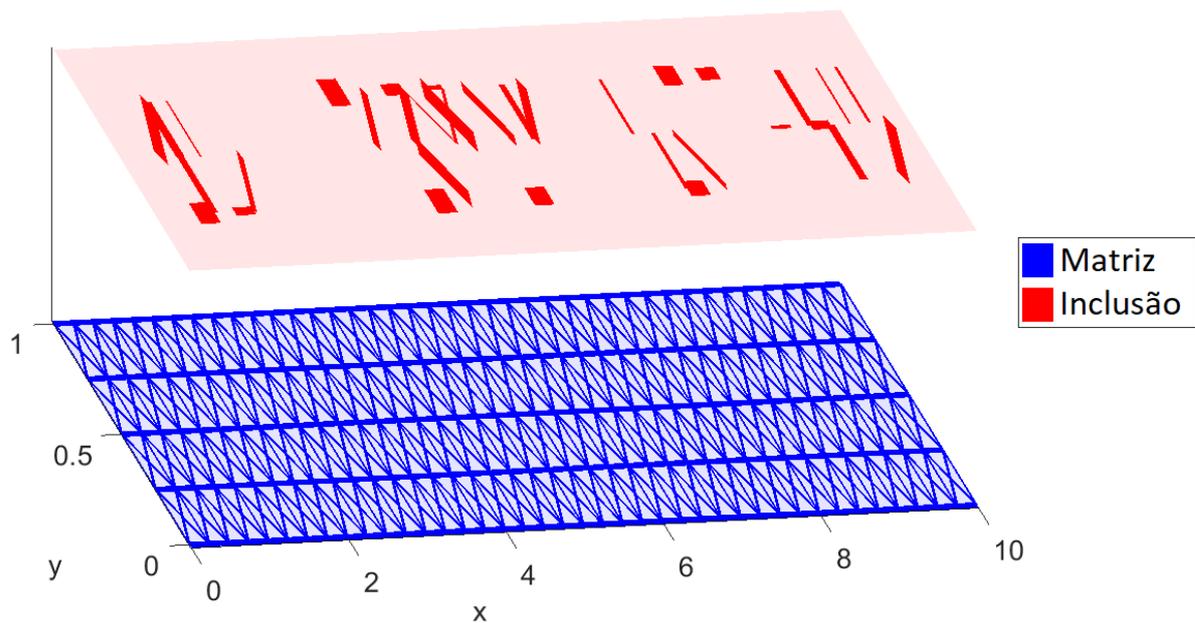
Mais ainda, para ilustrar a qualidade da solução encontrada, as áreas dos 34 elementos de inclusão são apresentadas em um gráfico de barras na Fig. 4.37, onde se compara os valores prescritos (em vermelho) com os encontrados após a otimização (em azul) em excelente concordância. Uma investigação dos autovalores da matriz Hessiana no ponto de mínimo mostra a mesma como positiva definida, conforme indicado na Fig. 4.38, como esperado. Nesta figura, os autovalores positivos são apresentados em escala logarítmica e em vermelho claro, com as barras vermelho-escuro e azul-escuro indicando,

Figura 4.35 – Gráficos de convergência



Fonte: Autor

Figura 4.36 – Configuração indeformada encontrada no ponto de mínimo

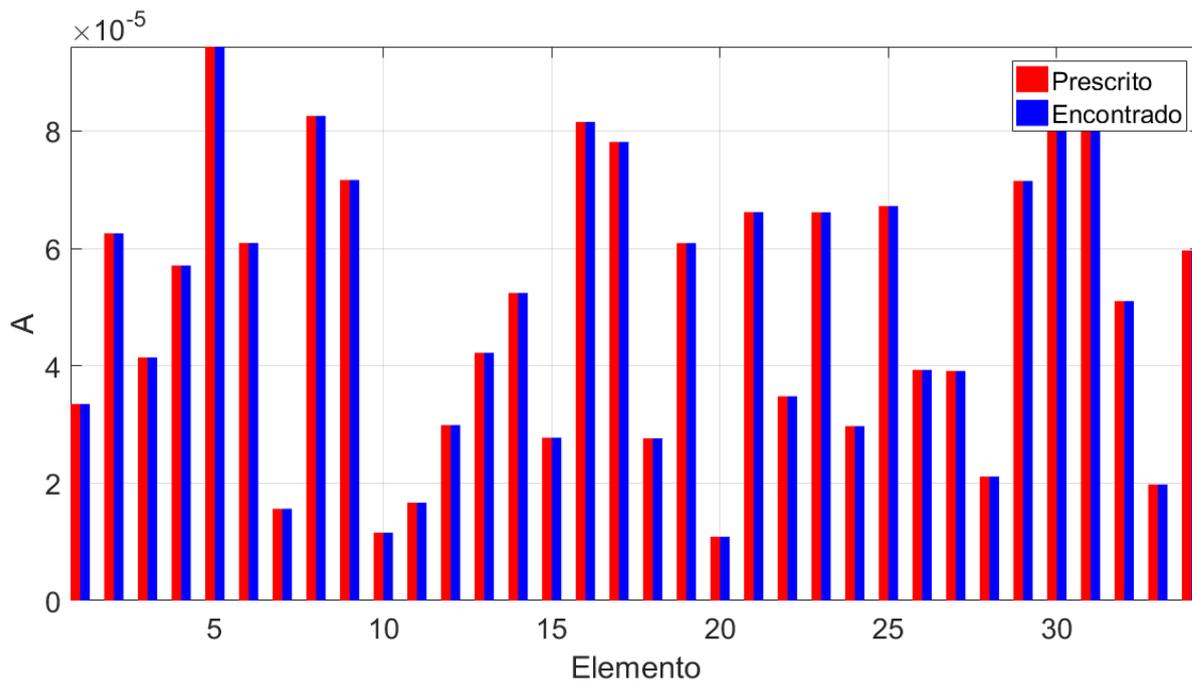


Fonte: Autor

respectivamente, o maior e o menor autovalores.

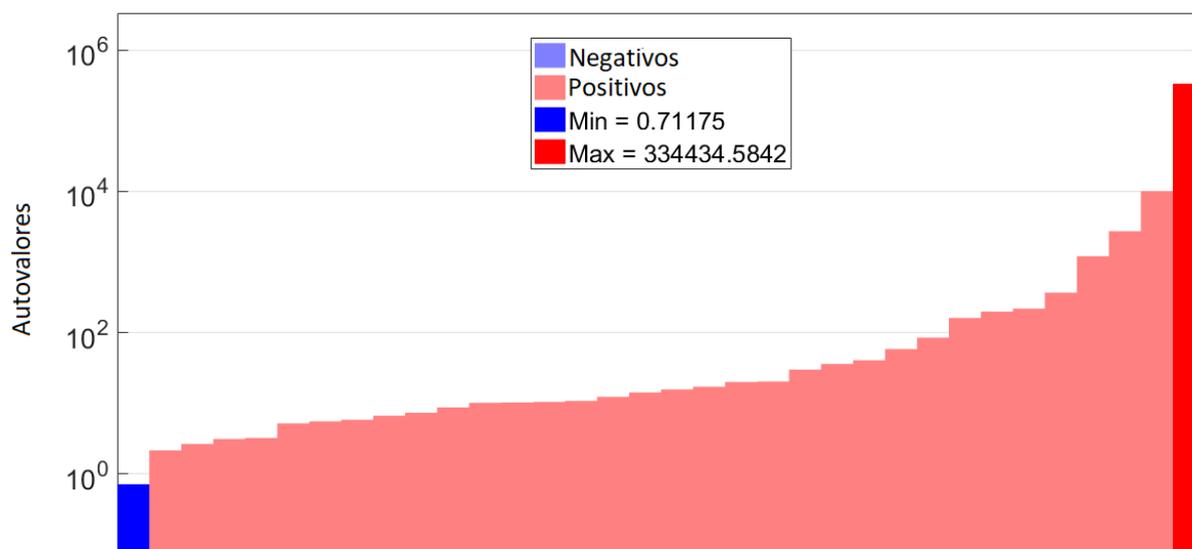
Um gráfico da função objetivo na vizinhança do ponto de mínimo (em vermelho) e

Figura 4.37 – Comparação entre áreas prescritas e encontradas no ponto final



Fonte: Autor

Figura 4.38 – Autovalores da matriz Hessiana no ponto final

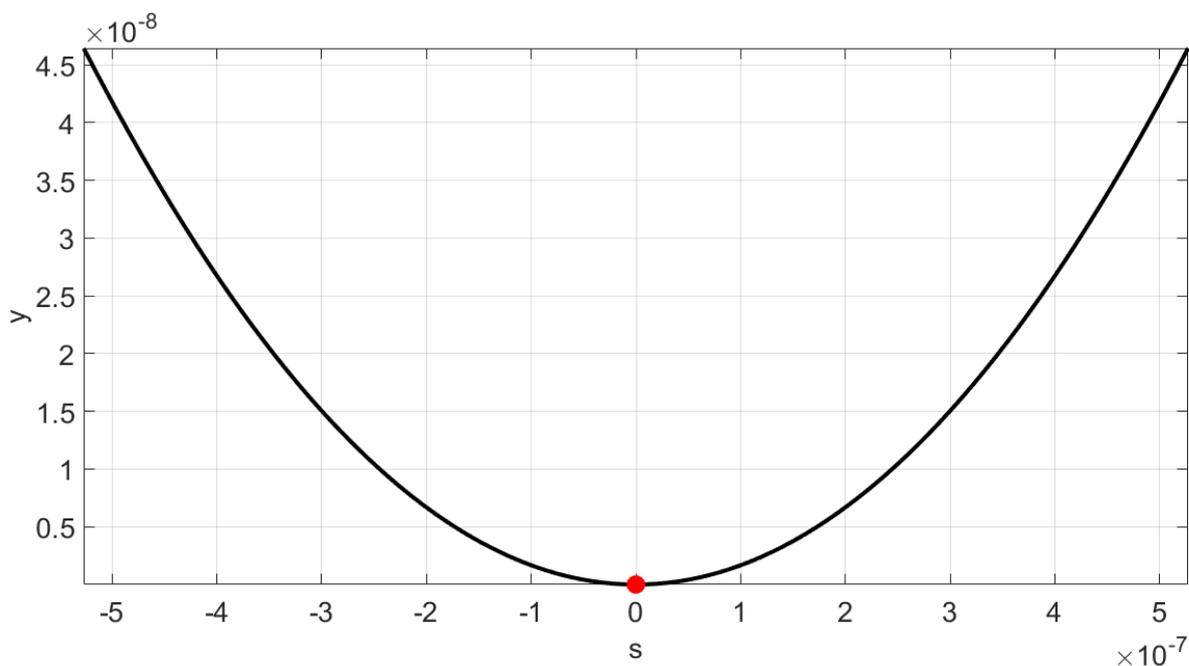


Fonte: Autor

na direção do autovetor associado ao menor autovalor, apresentada na Fig. 4.39, mostra que ambos os sentidos (positivo e negativo) do mesmo promovem crescimento da função objetivo, como esperado. Nesta figura,  $s$  representa uma parametrização ao longo da

direção definida pelo autovetor, enquanto  $y$  é o valor da função objetivo ao longo dessa direção.

**Figura 4.39 – Valores da função objetivo ao longo da direção do autovetor associado ao menor autovalor no ponto final**

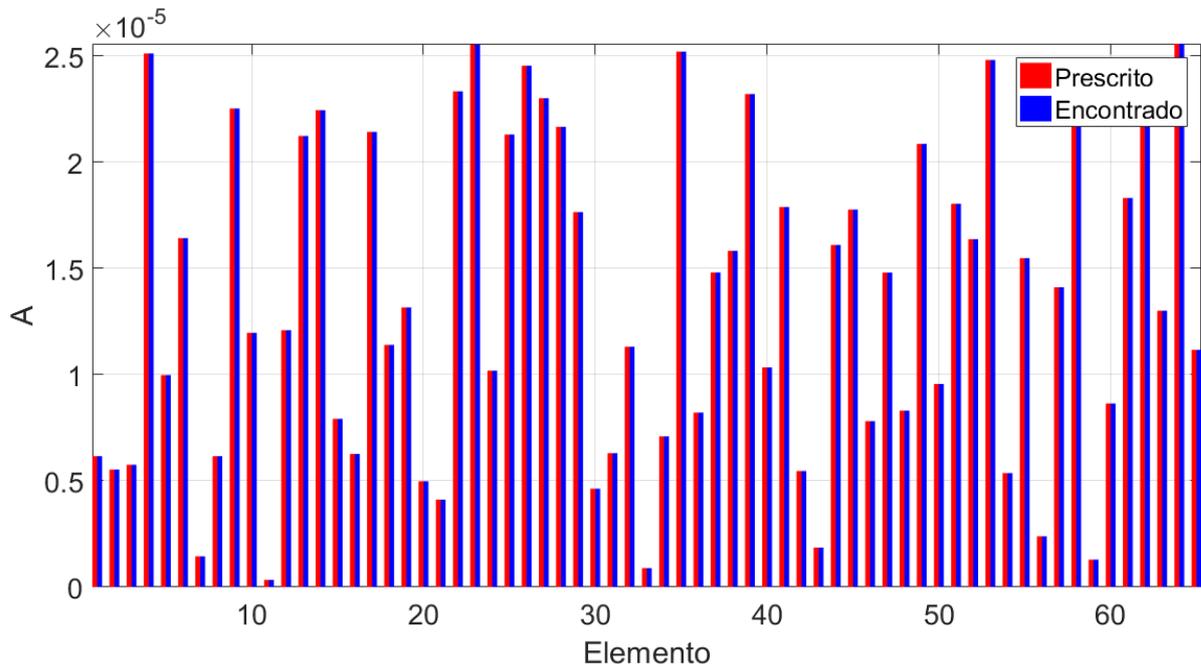


Fonte: Autor

Para confirmar tais resultados, outras 4 distribuições aleatórias de inclusões e áreas são geradas. Após um número semelhante de iterações, todas convergem para o ponto de mínimo prescrito com erros comparáveis, indicando que, ao menos para esta geometria e malha, o algoritmo é consistentemente capaz de encontrar o ponto de mínimo de maneira exata (dentro de uma certa tolerância numérica). Resultados análogos são observados para a malha de  $100 \times 10$  divisões, na qual são selecionados 65 dos 3241 elementos de inclusão originais (aproximadamente  $1/25$  do total). Para este caso, os valores prescritos e encontrados para as áreas das barras são apresentados na Fig. 4.40. Os mesmos são obtidos após 29 iterações, interrompidas pelo critério de 5 tentativas seguidas de redução da função objetivo sem sucesso.

Vale salientar, contudo, que este não é sempre o caso e os mesmos resultados não são observados para a malha de  $100 \times 10$  divisões quando um número significativamente maior de elementos são selecionados. Para 324 elementos, por exemplo, nenhum dos 5 testes realizados mostrou convergência, tendo todos sido interrompidos após o limite de 100 iterações estabelecido. Neste caso, o gráfico de comparação com as áreas encontradas, apresentado na Fig. 4.41 para os 60 primeiros elementos (conforme numerados pela im-

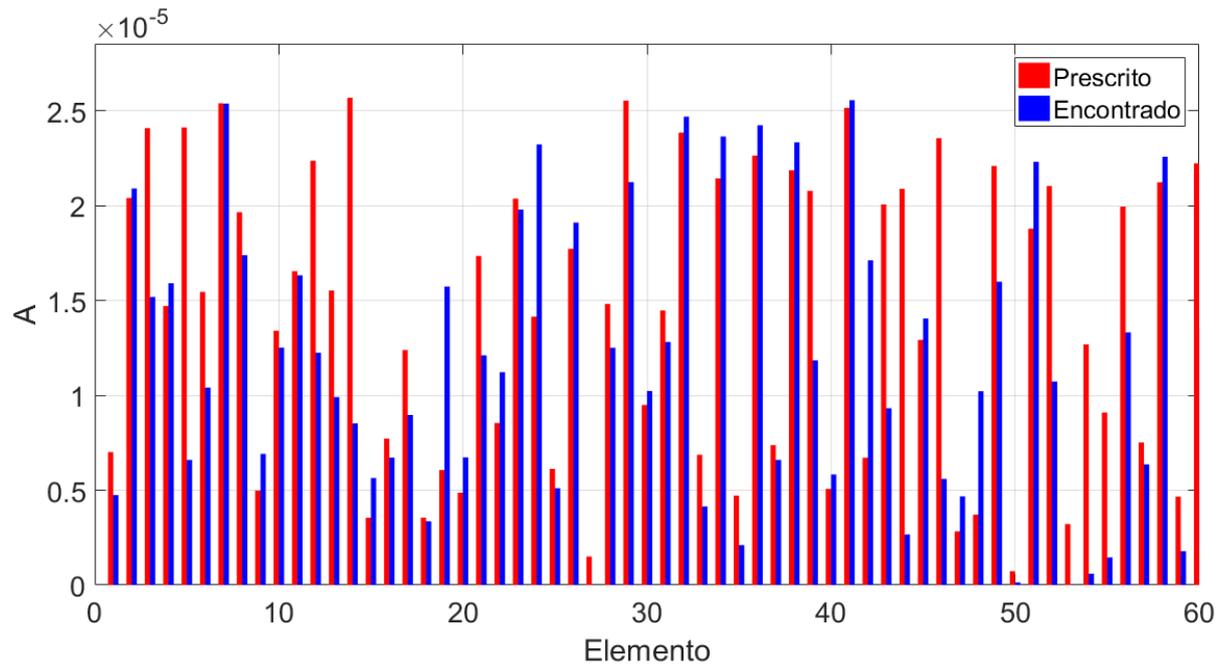
Figura 4.40 – Comparação entre áreas prescritas e encontradas no ponto final para uma malha mais refinada



Fonte: Autor

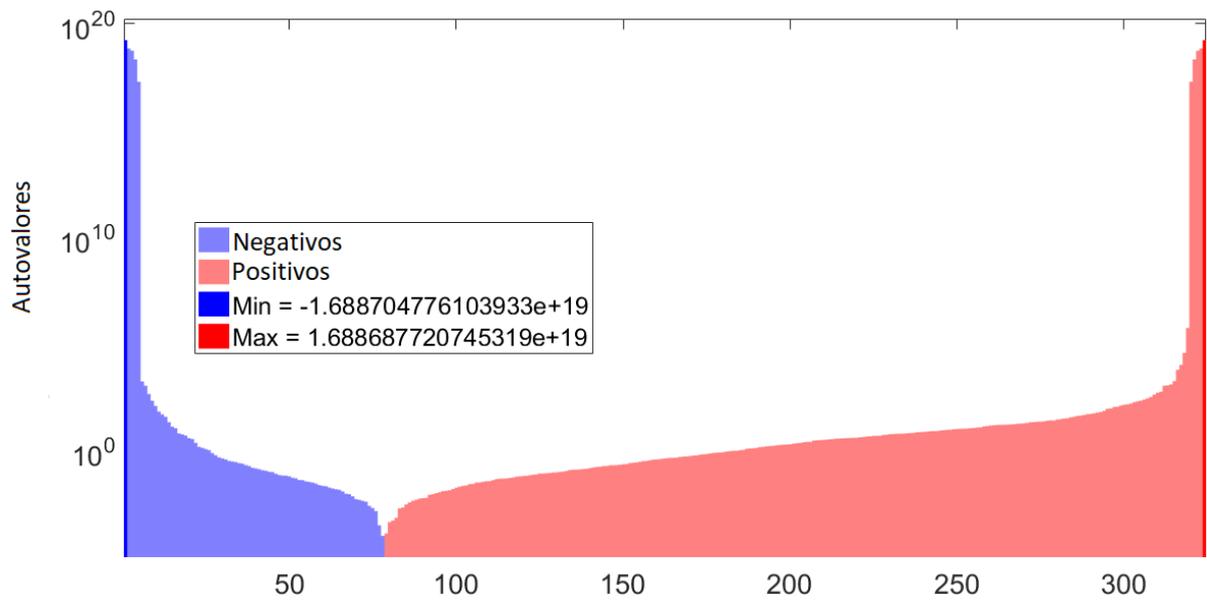
plementação), indica que, apesar de várias das mesmas terem se aproximado dos valores prescritos, algumas ainda se apresentaram largamente diferentes do esperado. Acredita-se que essas discrepâncias são provocadas pelo fato de tais barras apresentarem efeito bastante diminuto no valor da função objetivo, configurando uma região de vale bastante plana e de extrema dificuldade de navegação pelo algoritmo. Tal hipótese é embasada pelos resultados encontrados para os autovalores da matriz Hessiana apresentados na Fig. 4.42, que indicam a presença de termos negativos (indicados em azul claro e em módulo, com o mínimo em módulo em azul escuro) não encontrados nos exemplos anteriores, configurando a mesma como indefinida. A presença destes vales planos e estreitos, como anteriormente especulado e descrito, representam um desafio grande para o algoritmo de otimização, o que pode justificar a dificuldade, nestes casos, de se encontrar o mínimo, mesmo após tantas iterações. Ainda assim, observa-se uma tendência, para uma boa parte das barras, de aproximação entre os valores prescritos e os encontrados. Somando-se isto à clara redução no valor da função objetivo, apresentada nos gráficos de convergência da Fig. 4.43, considera-se que os resultados obtidos são razoáveis, especialmente levando-se em conta que tais flutuações seriam bastante pequenas se comparadas a imprecisões de medição, impressão e treinamento em peças reais.

Figura 4.41 – Comparação entre áreas prescritas e encontradas no ponto final para uma malha mais refinada e com mais elementos de inclusão



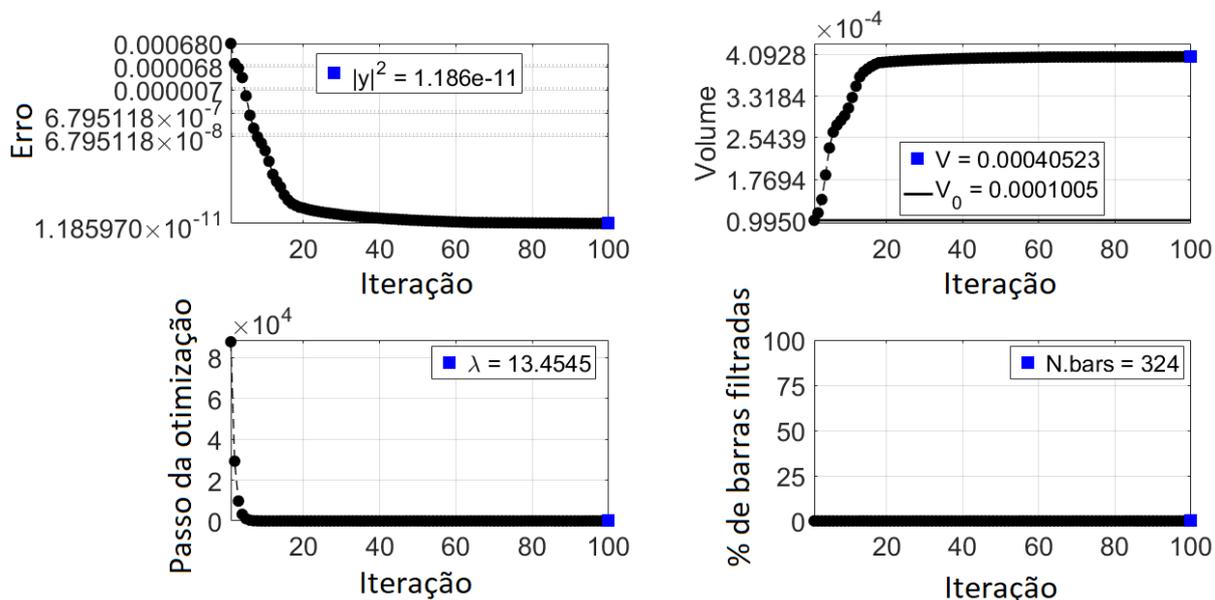
Fonte: Autor

Figura 4.42 – Autovalores da matriz Hessiana no ponto final para uma malha mais refinada e com mais elementos de inclusão



Fonte: Autor

Figura 4.43 – Gráficos de convergência para uma malha mais refinada e com mais elementos de inclusão



Fonte: Autor

#### 4.2.10 Considerações Adicionais

Pelos exemplos ilustrativos apresentados anteriormente, pode-se perceber que os parâmetros livres interferem fortemente na resposta do modelo, bem como um com o outro, e que a escolha dos mesmos depende largamente dos critérios de projeto adotados. Por exemplo, filtros menores tendem a gerar respostas com menores erros, porém com um maior volume de inclusão na forma de barras dispersas, mas ambos os fenômenos dependem largamente do efeito do filtro. Um esticção maior, por outro lado, reduz tanto o volume de inclusão quanto os erros, porém, para valores grandes do mesmo, ter-se-ia que possuir equipamentos apropriados e, a depender dos valores utilizados, uma formulação não-linear, que não é tratada neste documento. Percebe-se ainda que, como esperado, diferentes prescrições geram diferentes erros e que, até certo ponto, refinar a matriz ajuda a diminuí-los, por permitir ao algoritmo ajustes mais finos. Contudo, mesmo com malhas muito finas e tolerâncias estritas de convergência, existem limites físico impostos pela rigidez da matriz que impedem que certas configurações sejam fielmente replicadas. Considerando-se que o ponto inicial, por conta da natureza do filtro e dos vales na função objetivo, pode modificar fortemente a resposta encontrada, se não em termos de erro, ao menos no que tange a distribuição de material, recomenda-se que múltiplos pontos de partida (definidos, por exemplo, por diferentes frações de inclusão, como feito neste trabalho) sejam combinados com diferentes valores de filtragem, gerando vários resultados diferentes, e que a configuração desejada seja escolhida entre estas conforme algum crité-

---

rio de projeto objetivo, como menor erro, menor volume, menor número de barras, etc. De fato, a menos que estratégias mais sofisticadas sejam criadas futuramente, o projeto das peças, assim como as respostas geradas neste trabalho, passará por várias etapas nas quais os parâmetros serão modificados um a um ou em conjunto, até que se obtenha uma resposta considerada viável. Por fim, reitera-se que, para o caso específico tratado, as ressalvas feitas com relação à conexão entre o caso discreto e o contínuo são minimizadas devido ao baixo volume de inclusão presente nos compósitos, bem como pela natureza local e direcional das respostas, o que pode não ser verdade para outros exemplos.

## 5 CONSIDERAÇÕES FINAIS

*“É duro falhar, mas pior é nunca ter tentado o sucesso”*

– Theodore Roosevelt

Com a boa qualidade dos resultados obtidos, tanto o de verificação quanto os ilustrativos, considera-se que as estratégias e formulações desenvolvidas foram bem sucedidas em propor uma metodologia matemática e computacional para a obtenção da distribuição ótima de material de inclusão com memória de forma em compósitos poliméricos termo-ativados. Considera-se ainda que a base formulacional e numérica descritas, bem como o código computacional associado, podem servir de suporte para importantes futuros desenvolvimentos no âmbito da impressão 4D com otimização topológica. Tais desenvolvimentos, para os quais se dão algumas sugestões ao final deste capítulo, podem ter significativo impacto em importantes indústrias, conforme apresentado ao longo do documento, ajudando o desenvolvimento de diversas tecnologias associadas a materiais inteligentes, meta-materiais e técnicas avançadas de manufatura.

### 5.1 Principais Contribuições

As principais contribuições deste documento incluem, mas não se limitam à:

- Compilação de uma série de referências relevantes, em particular no âmbito de explicar físico-quimicamente o efeito de memória de forma que certos polímeros apresentam;
- Desenvolvimento de uma estratégia discreta para a aproximação dos compósitos poliméricos tratados, bem como os desenvolvimentos associados de um elemento de treliça com propriedade de memória de forma e uma maneira numérica incremental de tratar as etapas de treinamento e liberação;
- Proposição de uma formalização matemática para o problema de otimização tratado, para a qual se desenvolveram uma função objetivo e uma estratégia de filtragem, vetor gradiente e matriz Hessiana associados;
- Implementação computacional do elemento de treliça, de um *solver* numérico para o problema estrutura e de um algoritmo de otimização capazes de resolver o problema

de otimização proposto, com diversas particularizações e customizações objetivando desempenho ótimo para o código;

- Construção de um exemplo de verificação que pode ser expandido para testar derivações futuras das estratégias propostas, bem como um estudo acerca dos efeitos de alguns dos vários parâmetros livres do modelo proposto, além das ferramentas de pós-processamento associadas à visualização gráfica dessas respostas numéricas.

Acredita-se que tais contribuições possam ser de grande utilidade e relevância para trabalhos futuros que busquem expandir os desenvolvimentos aqui feitos.

## 5.2 Sugestões para Trabalhos Futuros

No âmbito de tais expansões, por virtude do caráter de base e exploratório deste trabalho, pode-se motivar, pelos resultados obtidos com o mesmo, diversas sugestões para exercícios futuros que tomem este como base. Recomenda-se, por exemplo, como passo inicial, a adição do efeito térmico nos esforços e deformações e a extensão das formulações desenvolvidas do 2D para o 3D. Tais modificações, que ainda não afetariam o regime elástico-linear e de pequenos deslocamentos, rotações e deformações adotado, já permitiriam a flexibilização da formulação proposta para a topologia de outras peças e aproximariam os resultados obtidos do comportamento real dos polímeros tratados. Ainda dentro dessas aproximações limitantes, outras duas flexibilizações poderosas que podem ser diretamente incorporadas na formulação são a possibilidade de se ter vários materiais de inclusão com diferentes temperaturas de transição, definindo mais de uma configuração intermediária memorizada. Analogamente, pode-se permitir que o algoritmo de otimização modifique, também, a região de matriz. Neste último caso, a mesma poderia conter vazios ou ainda, preferencialmente, ser impressa como uma estrutura em barras, inclusive em escalas não-macroscópicas, tendo correlação com a área de meta-materiais. Em ambos os casos, flexibilizar-se-ia a análise, permitindo regimes de deformação pós-impressão mais complexos e soluções mais próximas das geometrias prescritas. Mais ainda, para a matriz discreta, resolvem-se os problemas citados em relação à passagem do discreto para o contínuo.

Os maiores impactos em termos de flexibilização da formulação, contudo, devem vir da extensão das equações desenvolvidas e técnicas propostas para a solução de problemas física e/ou geometricamente não-lineares. O esforço teórico e computacional, neste caso, seria muito mais intenso, mas permitiria uma vasta gama de aplicações nas quais grandes deformações, deslocamentos e rotações são necessários. Mais ainda, problemas não-lineares, em vários casos, tendem a ser mais flexíveis em termos das soluções encontradas, portanto melhores concordâncias entre as configurações prescritas e obtidas podem ser advir deste caso, mesmo para situações em que as geometrias desejadas sejam

complexas.

Fugindo do âmbito teórico e computacional, a validação desta e de outras formulações futuras deve ser feita com base na comparação entre os resultados numéricos e testes experimentais conduzidos em peças reais manufaturadas em impressoras 3D físicas. Para fazer a passagem entre o modelo e o real, outros dois aspectos teóricos entrariam ainda em jogo: o refinamento da base computacional e a utilização de sistemas computacionais mais poderosos, permitindo a simulação de malhas mais finas, sendo as apresentadas neste documento muito próximas do limite físico do computador no qual foram simuladas, bem como o desenvolvimento de uma estratégia de pós-processamento para transformar as configurações discretas geradas em estruturas coerentes com a distribuição espacial real dentro do compósito e a resolução da impressora, seja para o caso contínuo, seja para estruturas discretas. Neste trabalho, conforme mencionado, ignora-se a possível superposição de duas barras de inclusão que, por virtude do processo de otimização, tenham ficado muito grandes, bem como a superposição entre matriz e inclusão no volume do compósito. Mais ainda, por se ter adotado um modelo de barra unidimensional, as regiões dos nós, nas quais barras invariavelmente estariam superpostas, são tratadas como rótulas perfeitas, o que não é caso real de impressão.

Todas essas considerações feitas somam-se ainda à necessidade de se obter valores reais para os módulos de elasticidade, temperatura de transição, cristalização, etc., para os materiais envolvidos. Tais medições, invariavelmente, carregam consigo erros e aproximações, bem como a possibilidade de se modificar os materiais usados para melhor adaptá-los a aplicações específicas. Nestes dois aspectos, respectivamente, pode-se incorporar confiabilidade para lidar com as incertezas e técnicas físico-químicas diversas para o desenvolvimento dos polímeros, entre muitos outros possíveis refinamentos.

As possibilidades são muitas e todas as sugestões feitas precisaram ser abstraídas das discussões feitas neste documento por questão de brevidade e de delimitação do escopo do trabalho, mas serviriam, em maior ou menor grau, para torná-lo ainda mais relevante, no que ficam como sugestões do autor.

## REFERÊNCIAS

- ARORA, J. S. *Introduction to Optimum Design, Third Edition*. [s.n.], 2011. 896 p. ISBN 0123813751. Disponível em: <<http://www.amazon.com/Introduction-Optimum-Design-Third-Edition/dp/0123813751>>.
- BEHL, M.; LENDLEIN, A. Shape-memory polymers are an emerging class of active polymers that. *Materials Today*, Elsevier Ltd, v. 10, n. 4, p. 20–28, 2007. ISSN 1369-7021. Disponível em: <[http://dx.doi.org/10.1016/S1369-7021\(07\)70047-0](http://dx.doi.org/10.1016/S1369-7021(07)70047-0)>.
- BELLIN, I. et al. Polymeric triple-shape materials. *Proceedings of the National Academy of Sciences of the United States of America*, v. 103, n. 48, p. 18043–18047, 2006. ISSN 0027-8424. Disponível em: <<http://www.pnas.org/cgi/doi/10.1073/pnas.0608586103>{\% }5Cnpapers3://publication/doi/10.1073/pnas.0608586}>.
- CHEN, Y. C.; LAGOUDAS, D. C. A constitutive theory for shape memory polymers. Part I. Large deformations. *Journal of the Mechanics and Physics of Solids*, v. 56, n. 5, p. 1752–1765, 2008. ISSN 00225096.
- CHRISTENSEN, P. W.; KLARBRING, A. *An Introduction to Structural Optimization*. 1. ed. [S.l.]: Springer, 2008. 214 p. ISBN 9781402086656.
- DAINI, J.; LIU, Y.; GALL, K. Finite Strain 3D Thermoviscoelastic Constitutive Model for Shape Memory Polymers. *Polymer Engineering and Science*, v. 46, n. 4, p. 486–492, 2006.
- DISSEMINATION OF IT FOR THE PROMOTION OF MATERIALS SCIENCE. *One Big Happy Molecule*. 2004. Disponível em: <<http://pslc.ws/macrog/exp/rubber/sepisode/spill.htm>>. Acesso em 01/05/2018.
- GE, Q.; QI, H. J.; DUNN, M. L. Active materials by four-dimension printing. *Applied Physics Letters*, v. 103, n. 13, 2013. ISSN 00036951.
- HEUCHEL, M. et al. Thermally induced shape-memory effects in polymers: Quantification and related modeling approaches. *Journal of Polymer Science, Part B: Polymer Physics*, v. 51, n. 8, p. 621–637, 2013. ISSN 08876266.

KAW, A. K. *Mechanics of Composite Materials*. 2. ed. Boca Raton, Florida, USA: Taylor & Francis, 2006. 457 p. ISBN 9780849313431.

KAZAKEVI, R.; STEEB, H. On the evolution law for the frozen fraction in linear theories. v. 82, n. 8, p. 1103–1115, 2012.

KUMAR, A.; GUPTA, R. K. *Fundamentals of Polymer Engineering*. 2. ed. New York: Marcel Dekker, Inc, 2003. 693 p. ISBN 0824708679.

LI, G.; XU, W. Thermomechanical behavior of thermoset shape memory polymer programmed by cold-compression: Testing and constitutive modeling. *Journal of the Mechanics and Physics of Solids*, Elsevier, v. 59, n. 6, p. 1231–1250, 2011. ISSN 00225096. Disponível em: <<http://dx.doi.org/10.1016/j.jmps.2011.03.001>>.

LIU, Y. et al. Thermomechanics of shape memory polymers: Uniaxial experiments and constitutive modeling. *International Journal of Plasticity*, v. 22, n. 2, p. 279–313, 2006. ISSN 07496419.

MAO, Y. et al. Sequential Self-Folding Structures by 3D Printed Digital Shape Memory Polymers. *Scientific Reports*, Nature Publishing Group, v. 5, p. 13616, 2015. ISSN 2045-2322. Disponível em: <<http://dx.doi.org/10.1038/srep13616>> <<http://www.nature.com/doi/10.1038/srep13>>.

MAUTE, K. et al. Level Set Topology Optimization of Printed Active Composites. *Journal of Mechanical Design*, v. 137, n. 11, p. 111402, 2015. ISSN 1050-0472. Disponível em: <<http://mechanicaldesign.asmedigitalcollection.asme.org/article.aspx?doi=10.1115/1.4030994>>.

MENG, H.; LI, G. A review of stimuli-responsive shape memory polymer composites. *Polymer*, Elsevier Ltd, v. 54, n. 9, p. 2199–2221, 2013. ISSN 00323861. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0032386113001444>>.

MOON, S.; RAO, I. J.; CHESTER, S. A. Triple Shape Memory Polymers: Constitutive Modeling and Numerical Simulation. *Journal of Applied Mechanics*, v. 83, n. 7, p. 071008, 2016. ISSN 0021-8936. Disponível em: <<http://appliedmechanics.asmedigitalcollection.asme.org/article.aspx?doi=10.1115/1.4033380>>.

NIELSEN, H. B. Damping Parameter in Marquardt ' S Method Damping Parameter in Marquardt ' S Method. *Mathematical Modelling*, n. M, 1999.

NOCEDAL, J.; WRIGHT, S. J. *Numerical Optimization*. second. New York, NY, USA: Springer, 2006.

PILATE, F. et al. Shape-memory polymers for multiple applications in the materials world. *European Polymer Journal*, Elsevier Ltd, v. 80, p. 268–294, 2016. ISSN 00143057. Disponível em: <<http://dx.doi.org/10.1016/j.eurpolymj.2016.05.004>>.

POLYMER SCIENCE LEARNING CENTER. *Theory of Rubber Conformation*. 2002. Disponível em: <<https://www.doitpoms.ac.uk/tlplib/stiffness-of-rubber/rubber-conformation.php?printable=1>>. Acesso em 01/05/2018.

TOBUSHI, H. et al. Thermomechanical constitutive model of shape memory polymer. *Mechanics of Materials*, v. 33, n. 10, p. 545–554, 2001. ISSN 01676636.

WU, J. et al. Multi-shape active composites by 3D printing of digital shape memory polymers. *Scientific reports*, Nature Publishing Group, v. 6, n. April, p. 24224, 2016. ISSN 2045-2322. Disponível em: <<http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=4829848&tool=pmcentrez&rendertype=ab>>.