

ALBERTO TOSHIO YAMAGUCHI

PROJETO DE SUSPENSÃO PARA FEMTO SLIDER POR OTIMIZAÇÃO
TOPOLÓGICA

MONOGRAFIA

SÃO PAULO

2008

ALBERTO TOSHIO YAMAGUCHI

PROJETO DE SUSPENSÃO PARA FEMTO SLIDER POR OTIMIZAÇÃO
TOPOLÓGICA

ÁREA DE CONCENTRAÇÃO:
ENGENHARIA MECATRÔNICA

ORIENTADOR:
PROF. DR. EMÍLIO CARLOS NELLI
SILVA

SÃO PAULO
2008

Palavras chave: otimização topológica, HDD, MEF, MITC4

RESUMO

Neste trabalho, é estudada a otimização topológica aplicada no projeto de suspensões para sistemas de alta capacidade de armazenagem magnético de dados. Com o desenvolvimento de *hard disk drives* (HDD) de alta capacidade de armazenagem e rápida varredura de dados, torna-se necessário que este se mova mais rápido e com maior precisão de movimento. Assim, mesmo pequenas vibrações induzidas pelo movimento de posicionamento nas trilhas podem gerar erros de leitura. O presente trabalho pretende projetar a suspensão do femto *slider*, visando aumentar a rigidez a vibrações, com o menor volume possível, de acordo com os parâmetros de funcionamento do HDD. Serão utilizados o método dos elementos finitos para análise e a técnica de otimização topológica para encontrar a topologia ideal. Posteriormente, será fabricado um protótipo do modelo projetado que será testado para se comparar as características reais com as características de projeto.

Key words: topology optimization, HDD, FEM, MITC4

ABSTRACT

In this work, it will be studied the application of topology optimization to the design of suspensions for high-density magnetic recording systems. With the development of high data storage capacity and fast data streaming hard disk drives (HDD), it is required from these systems to move faster with greater positioning accuracy. In this way, even small mechanical vibrations induced by the positioning movement along the tracks may lead to errors in data reading. The present work intends to create a design of a suspension for femto slider, rising stiffness against vibrations with the smaller possible volume, according to the functioning parameters of the HDD. The finite element method will be used for analysis and the topology optimization technique will be applied to find the ideal topology. Later, a prototype will be fabricated and tested, comparing the real characteristics with the model characteristics.

SUMÁRIO

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

LISTA DE SÍMBOLOS

1	INTRODUÇÃO	1
2	OBJETIVO E JUSTIFICATIVA	6
3	FUNDAMENTOS TEÓRICOS	7
3.1	Método dos elementos finitos (MEF)	7
3.1.1	Elemento de placa.....	7
3.1.2	Matriz de rigidez de membrana.....	13
3.1.3	Matriz de Massa	14
3.2	Método de Otimização Topológica	15
3.2.1	Modelo de material.....	17
3.2.2	Escala de cinza	20
3.2.3	Instabilidade tabuleiro	22
3.2.4	Dependência de malha	23
3.3	Usinagem fotoquímica	26
4	Formulação do problema.....	27
4.1	Otimização de rigidez	27
4.2	Otimização de autovalores	28
4.3	Função objetivo	30
4.4	Análise de sensibilidades.....	30
5	Implementação numérica	32
5.1	Implementação do MEF	32
5.2	Programação Linear Seqüencial (PLS)	32

5.3	Limites móveis	34
5.4	Instabilidades numéricas	36
5.4.1	Falta de simetria em matrizes	36
5.4.2	Unidades de medida	36
5.5	Restrições de massa	37
5.6	Critérios de convergência	37
5.7	Cálculo de autovalores	38
5.8	Simetria da otimização.....	39
5.9	Funcionamento do programa.....	40
6	Resultados	42
6.1	Domínio e condições de contorno	42
6.2	Validação dos limites móveis.....	42
6.3	Validação do filtro	45
6.4	Comparação com o algoritmo de 99 linhas.....	47
6.5	Análise da suspensão de um HDD atual.	48
6.6	Projeto da suspensão	49
6.6.1	Geometria inicial.....	50
6.6.2	Análise de modelos	51
6.7	Ensaio do protótipo	59
7	Conclusão	63
8	Referências	64
APÊNDICE A - Validação do MEF		
APÊNDICE B - Código de Matlab do programa		
APÊNDICE C - Desenho de fabricação		

LISTA DE FIGURAS

Figura 1.1 - Componentes básicos de um HDD.....	1
Figura 1.2 - Evolução da capacidade de armazenagem através dos anos....	2
Figura 1.3 - Comparação entre um pico <i>slider</i> e um femto slider.....	3
Figura 1.4 - Oscilações principais	4
Figura 3.1 - Graus de liberdade do elemento MITC4.....	8
Figura 3.2 - Pontos para interpolação do cisalhamento transversal	12
Figura 3.3 - Exemplos de otimização aplicada a projetos de estruturas.	16
Figura 3.4 - Modelo de material baseado no método de homogeneização.	18
Figura 3.5 - Representação de modos localizados de placa em vibração ...	19
Figura 3.6 - Gráfico da razão ρ/E , comparando os modelos SIMP e RAMP.	20
Figura 3.7 - Resultado de OT com escala de cinza	21
Figura 3.8 - Gráfico comparativo das penalizações.	21
Figura 3.9 - OT de viga engastada para três casos de penalização.	22
Figura 3.10 - Topologia apresentando instabilidade tabuleiro.	23
Figura 3.11 - Exemplo de dependência de malha.....	24
Figura 3.12 - Representação do filtro espacial	25
Figura 4.1 - Frequências naturais ao longo de um processo de OT.	28
Figura 5.1 - Função a ser linearizada	34
Figura 5.2 - Função do Matlab que utiliza o PLS.	34
Figura 5.3 - Fluxograma de funcionamento do programa.	41
Figura 6.1 - Domínio de trabalho.	42
Figura 6.2 - Placa com momento unitário aplicado em extremidade livre....	43
Figura 6.3 - Comparação entre limites fixos e limites calculados com base na derivada segunda.	43
Figura 6.4 - Placa com carga unitária no sentido de z negativo aplicado em extremidade livre.....	44
Figura 6.5 - Comparação entre limites fixos e limites calculados com base na derivada segunda para o teste com carga vertical.....	44
Figura 6.6 - Placa com carga unitária no sentido de y negativo aplicada no canto inferior esquerdo da placa.....	45
Figura 6.7 - Otimização de rigidez a torção sem a utilização de filtro.....	46

Figura 6.8 - Otimização de rigidez a torção com a utilização de filtro espacial sobre as sensibilidades.....	47
Figura 6.9 - Resultado do 99 linhas – 94 iterações.....	47
Figura 6.10 - Resultado do algoritmo implementado – 72 iterações.....	48
Figura 6.11 - Foto da suspensão do HDD de 80Gb.....	49
Figura 6.12 - Modelo de MEF da suspensão real.	49
Figura 6.13 - Principais cargas que serão utilizadas.....	50
Figura 6.14 - Resultado de teste inicial.....	50
Figura 6.15 - Otimização de rigidez a flexão na 20 ^a iteração.....	51
Figura 6.16 - Modelo de MEF para peça de maior rigidez a flexão.....	52
Figura 6.17 - Otimização 1.....	53
Figura 6.18 - Resultados das otimizações 2 e 3 respectivamente.....	54
Figura 6.19 - Resultados da otimização 3 variando-se o filtro aplicado.....	54
Figura 6.20 - Otimização 4.....	55
Figura 6.21 - Otimização 5.....	56
Figura 6.22 - Otimização 6.....	57
Figura 6.23 - Modelo final.	58
Figura 6.24 - Desenho esquemático da montagem da suspensão.....	60
Figura 6.25 - Montagem para teste do protótipo.....	60
Figura 6.26 - Amostragem das frequências x amplitude de oscilação.....	61
Figura 6.27 - Modelo aproximado do ensaio.....	62

LISTA DE TABELAS

Tabela 6.1 - Propriedades do aço inoxidável consideradas para a OT.....	42
Tabela 6.2 - Resultados para a suspensão real.	49
Tabela 6.3 - Resultados para peça de maior rigidez a flexão.	52
Tabela 6.4 - Dados para otimização 1.	52
Tabela 6.5 - Características do modelo da otimização 1.	53
Tabela 6.6 Dados para otimização 1.	53
Tabela 6.7 - Dados para otimização 4.	55
Tabela 6.8 - Características do modelo da otimização 4.	55
Tabela 6.9 - Dados para otimização 5.	56
Tabela 6.10 - Características do modelo da otimização 5.	57
Tabela 6.11 - Dados para otimização 6.	57
Tabela 6.12 - Características do modelo da otimização 6.	58
Tabela 6.13 - Características do modelo final.....	58
Tabela 6.14 - Valores das propriedades do material do protótipo.....	59
Tabela 6.15 - Comparação entre o projeto e o simulação das condições do ensaio do protótipo	62

LISTA DE ABREVIATURAS E SIGLAS

CAMD	Continuos Approximation of Material Distribution
DFE	Domínio Fixo Estendido
FEMTO	10^{-15}
gl	Número de graus de liberdade
HDD	Hard Disk Drive
MAC	Modal Assurance Criterion
MEF	Método dos elementos finitos
MITC	Mixed interpolation of tensorial components
ODD	Optical Disk Drive
OT	Otimização Topológica
PCM	Photochemical Machining
RAMP	Rational Approximation of Material Properties
SIMP	Simple Isotropic Material with Penalization

LISTA DE SÍMBOLOS

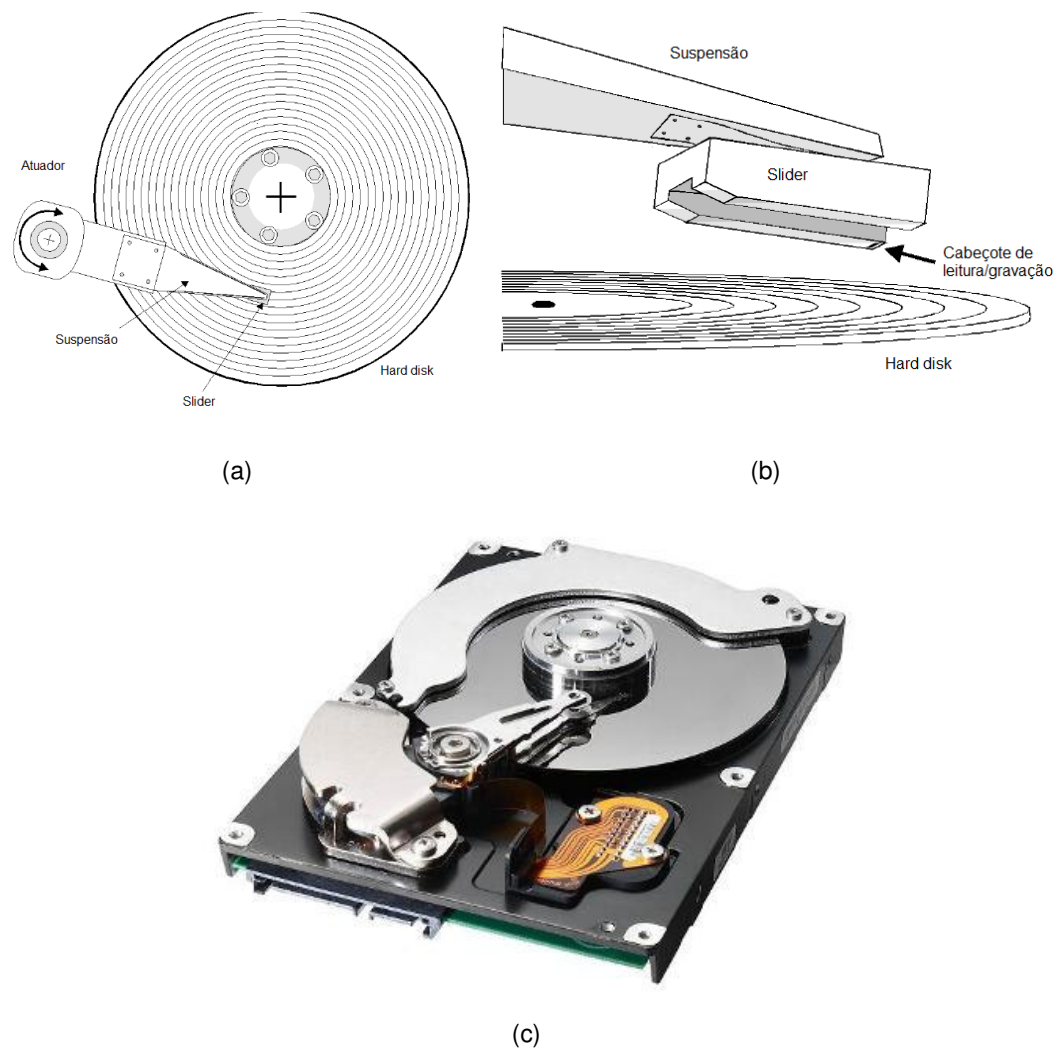
A	Matriz para cálculo da matriz de massa
B_i	Matriz de interpolação deformação-deslocamento
C_i	Matriz tensão-deformação
E	Módulo de elasticidade
f_o	Função objetivo
H	Matriz de interpolação
h	Espessura do elemento
h_i	Função de interpolação
J	Matriz Jacobiana
K	Matriz de rigidez
M	Matriz de massa
n	Número de nós
p	Coeficiente de penalização para o modelo SIMP
q	Coeficiente de penalização para o modelo RAMP
r	Coordenada natural r (relativo ao eixo x)
s	Coordenada natural s (relativo ao eixo y)
t	Coordenada natural t (relativo ao eixo z)
u_n	Vetor de deslocamentos nodais
V	Volume
w	Deslocamento no eixo z
w_i	Coeficiente de peso
x	Coordenada x
y	Coordenada y
z	Coordenada z
α	Matriz dos coeficientes da interpolação do cisalhamento transversal do MITC4
β_x	Rotação do plano deformado em relação plano xz
β_y	Rotação do plano deformado em relação plano yz
γ	Deformação de cisalhamento transversal
γ_x	Pseudo-densidade
ε	Deformação axial

τ	Tensão
Λ	Frequência de ressonância média
λ	Autovalor
λ_0	Valor de frequência mínima para ajuste da função objetivo
ρ	Densidade
θ_i	Ângulo de rotação em relação ao eixo i
ν	Coeficiente de Poisson
Φ	Autovetor
ω	Frequência de ressonância
Ω	Domínio Fixo Estendido

1 INTRODUÇÃO

A assustadora revolução na capacidade de armazenagem de *hard disk drives* (HDD) ao longo dos últimos 40 anos tem sido possível graças a miniaturização de cada componente do sistema. Esta tendência continua com a introdução do ‘femto’ *slider* que representa um grande avanço na tecnologia de armazenagem magnético de dados. [1]

Para não haver confusões com a nomenclatura, analisemos os componentes de um HDD.



Podemos verificar na Figura 1.1a que o atuador possibilita um movimento rotativo da suspensão, esta, por sua vez, transmite o movimento ao slider, visto que a sustentação do mesmo é realizada pelo filme de ar gerado durante a rotação do disco. O cabeçote de leitura/gravação é preso ao slider, como pode ser visto na Figura 1.1b.

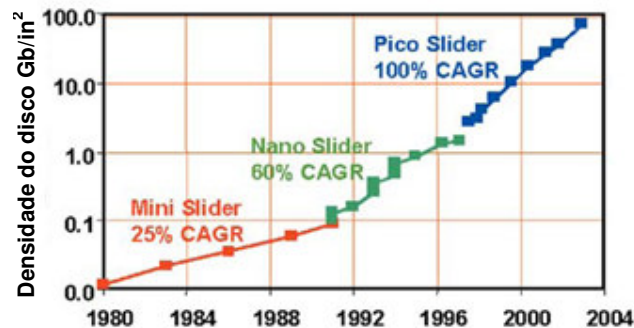


Figura 1.2 - Evolução da capacidade de armazenagem através dos anos.

A Figura 1.2 apresenta a evolução da capacidade de armazenagem ao longo dos anos. Esse aumento na densidade magnética dos discos necessita de um espaçamento muito pequeno e constante entre o cabeçote e o disco. Para se ter uma idéia, a distância entre os dois é de 20 a 60 nm, que é o mesmo valor da rugosidade superficial dos discos para os HDDs de hoje. A configuração ideal para a interface *slider*-disco deve ter um *slider* tal que 99,999% do seu peso seja sustentado pela força do ar gerada abaixo do mesmo pela rotação do disco. [2]

Um dos fatores mais importantes para o aumento da capacidade e miniaturização dos HDDs é a redução em 3 vezes do tamanho e em 30 vezes da massa do *slider* (ao longo de 1980 a 2002). Concomitantemente, a suspensão também passou por uma grande redução de tamanho. A Figura 1.3 mostra uma comparação entre um pico *slider* e um femto slider.[1]



Figura 1.3 - Comparação entre um pico *slider* e um femto slider, e suas respectivas suspensões.

O primeiro HDD a utilizar o femto *slider* surgiu em 2003, o modelo Hitachi's Travelstar 7K60, com rotação de 7200 rpm e 60Gb de capacidade de armazenagem. A redução de tamanho e massa do *slider*, a sustentação a ar e o novo design da suspensão resultaram em um grande aumento de desempenho, redução do consumo de energia e aumento da capacidade de armazenagem. Atualmente, encontramos HDDs de 750Gb disponíveis para computadores domésticos.

A redução dos componentes e da distância entre o cabeçote e o disco, e a maior densidade magnética do disco originaram a necessidade de se aumentar a precisão ao longo da operação, tendo que se controlar parâmetros tais como a vibração e flexão. No projeto de suspensões para HDDs, basicamente visa-se aumentar as frequências de ressonância e a rigidez a oscilação lateral, e, ao mesmo tempo, diminuir a rigidez a oscilações verticais e torções. A Figura 1.4 ilustra esses movimentos. Para isto, esses devem ser otimizados, visto que não é mais possível utilizar somente a intuição para se encontrar a geometria ideal. É neste ponto que se utiliza a otimização topológica.

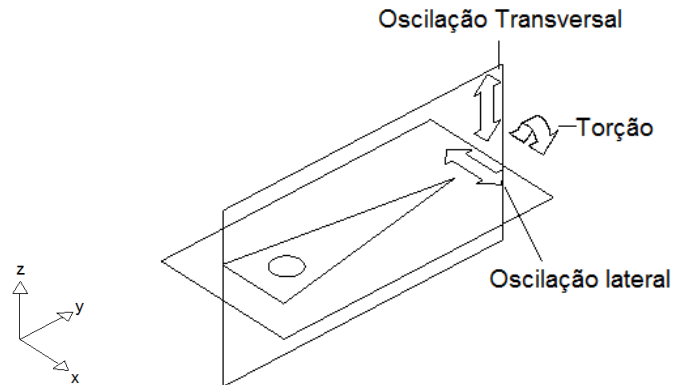


Figura 1.4 - Oscilações principais

A otimização topológica (OT) é um método computacional que permite projetar uma topologia ótima segundo um determinado critério, como exemplo temos maximizar as frequências de ressonância, minimizando a massa. O método distribui o material ao longo da peça, visando maximizar ou minimizar a função objetivo.

Em 1980, a otimização topológica surgiu no meio acadêmico dos EUA e Europa com o artigo "*Generating Optimal Topologies in Structural Design Using a Homogenization Method*" pelos autores Martin P. Bendsoe e Noboru Kikuchi. Na década de 90, a OT passou a ser largamente utilizada nas indústrias automobilística e aeronáutica dos EUA, Japão e Europa para o projeto de peças otimizadas. Recentemente, o método tem se expandido para outras áreas da engenharia no meio acadêmico como o projeto de mecanismos flexíveis, atuadores piezelétricos, motores eletromagnéticos, etc. Através da OT, o processo de projeto torna-se mais genérico e diminui a necessidade da experiência específica de alguns engenheiros. Contudo, a presença do engenheiro na análise antes e depois de se encontrar a topologia ideal é necessária [3].

O projeto de suspensão para femto slider utilizando OT já vem sendo alvo de estudo de outros profissionais, estudos mostram que as características da suspensão podem ser muito melhoradas por meio da OT [10]. Trabalhos envolvendo a programação por meio do Matlab também encontraram resultados muito satisfatórios nessa área [8].

Atualmente, também se utiliza a OT no projeto de suspensões de *optical disk drives* (ODD) que ao invés de magnetizarem um disco, utilizam a luz para sensibilizar o mesmo. Esses projetos visam aumentar a rigidez da suspensão, da mesma forma que nos HDDs, e, ao mesmo tempo, manter a rigidez da extremidade que sustenta o cabeçote em uma determinada faixa, para que seja possível movimentá-la verticalmente, possibilitando focar a luz [11,12].

2 OBJETIVO E JUSTIFICATIVA

O objetivo deste trabalho de formatura é a implementação de um algoritmo de otimização topológica aplicado no projeto de suspensão para cabeçotes de leitura/gravação de *hard disk drives* (HDD). O projeto visa aumentar as frequências de ressonância e diminuir a rigidez da suspensão em alguns graus de liberdade, aumentando a rigidez de outros graus. O algoritmo será codificado no Matlab que proporciona uma vasta biblioteca de funções e permite uma manipulação mais simplificada de matrizes. Em seguida, será fabricado um protótipo para medição das frequências de ressonância do mesmo.

A motivação deste trabalho é o domínio da otimização topológica aplicada a projetos, abordando conceitos de método de elementos finitos aplicado a placas. A peça a ser projetada também possui grande importância no mercado de HDDs, pois cada vez mais é necessário o projeto otimizado dessas suspensões devido à alta densidade de dados nos discos, alta velocidade de varredura e alta precisão de posicionamento.

3 FUNDAMENTOS TEÓRICOS

Nesta seção, serão descritos brevemente os principais conceitos teóricos utilizados na formulação e análise da otimização da suspensão do *slider* do HD.

3.1 Método dos elementos finitos (MEF)

O método dos elementos finitos (MEF) é uma técnica amplamente utilizada na solução de problemas físicos tais como transferência de calor, escoamento de fluidos e análise de estruturas e sólidos. Contudo trata-se de um procedimento numérico que pode ser utilizado em praticamente qualquer campo da Engenharia [5].

O termo *elemento finito* foi introduzido pela primeira vez em 1960 por Clough no artigo “The finite element method in plane stress analysis” [6]. A principal razão do amplo uso deste método em 1960 foi a possibilidade de se utilizar computadores para se resolver o grande número de iterações requeridas pelo MEF.

Para se resolver um determinado problema físico, faz-se necessário criar um modelo matemático, utilizando um conjunto de equações diferenciais. Este modelo pode ser resolvido analiticamente por meio dos métodos de Rayleigh-Ritz, Galerkin, ou numericamente pelo método das diferenças finitas (MDF), MEF. O problema das soluções analíticas é que só podem ser utilizadas para uma pequena gama de problemas de baixa complexidade, enquanto os métodos numéricos permitem encontrar soluções aproximadas para diversos problemas de mais variada complexidade [4].

3.1.1 Elemento de placa

Neste trabalho será abordada a otimização de uma chapa de metal, na qual a espessura é muito menor do que a largura e comprimento, desta forma, não é necessária a utilização de elementos tridimensionais. Basta utilizar elementos bidimensionais que utilizam o valor da espessura, mas tratam essa dimensão de uma maneira simplificada. Dentro da família dos

elementos bidimensionais, distinguem-se dois tipos principais, os elementos de placa, utilizados neste trabalho, e os elementos de casca.

Os elementos de placa são elementos bidimensionais com movimentação no espaço tridimensional. Isso quer dizer que os nós do elemento ficam restritos dentro de um plano, contudo, podem mover-se em qualquer direção do espaço. Estes elementos são um caso especial do elemento de casca, não sendo utilizados em situações em que a superfície possua grande curvatura, como no caso de cilindros e esferas. Mesmo para os elementos de placa, existem diversas formulações para o cálculo da rigidez do elemento, dentre essas formulações utilizar-se-ão os elementos MITCn.

Os elementos MITCn (*mixed interpolation of tensorial components*) possuem uma sólida base matemática, são confiáveis e eficientes. Os MITCn são elementos quadriláteros de n nós, podendo ser divididos também em elementos de placa e elementos de casca. Neste trabalho, será utilizado o elemento de placa MITC4 (4 nós).

Elemento de placa MITC4

O MITC4 possui 3 graus de liberdade por nó: w , θ_x e θ_y . A Figura 3.1 apresenta o elemento com seus respectivos graus de liberdade.

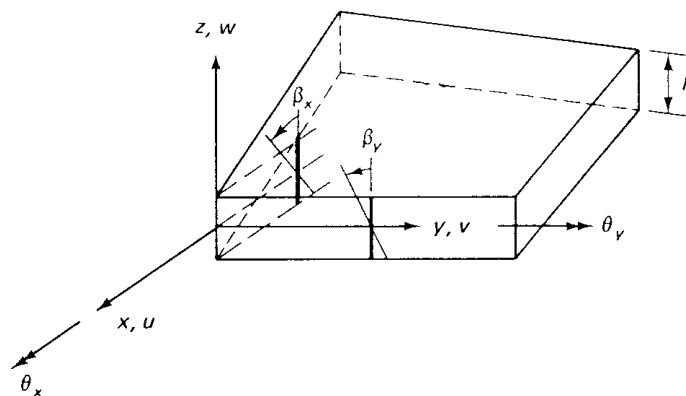


Figura 3.1 - Graus de liberdade do elemento MITC4.

As equações a seguir apresentam as deformações do elemento, sendo (3.1) as deformações por flexão e (3.3) as deformações por cisalhamento transversal:

$$\kappa = \begin{bmatrix} \frac{\partial \beta_x}{\partial x} \\ \frac{\partial \beta_y}{\partial y} \\ \frac{\partial \beta_x}{\partial y} + \frac{\partial \beta_y}{\partial x} \end{bmatrix} \quad (3.1)$$

$$\begin{bmatrix} \epsilon_{xx} \\ \epsilon_{yy} \\ \gamma_{xy} \end{bmatrix} = -z\kappa \quad (3.2)$$

$$\gamma = \begin{bmatrix} \gamma_{xz} \\ \gamma_{yz} \end{bmatrix} = \begin{bmatrix} \frac{\partial w}{\partial x} - \beta_x \\ \frac{\partial w}{\partial y} - \beta_y \end{bmatrix} \quad (3.3)$$

Algumas considerações devem ser feitas para que seja possível utilizar um elemento de placa:

- 1- A estrutura deve possuir uma espessura muito menor do que as demais dimensões;
- 2- A tensão ao longo da espessura da placa é zero;

Este elemento possui interpolações distintas para se calcular as deformações de flexão e cisalhamento transversal, contudo a matriz de rigidez final é a soma das duas matrizes que serão deduzidas a seguir. Cabe salientar que existem outras variações na formulação do MITC4 que o deixam mais estável e preciso para o cálculo de frequências de ressonância [9]. Contudo o MEF não é o escopo principal deste trabalho, assim usaremos uma formulação mais clássica.

Matriz de rigidez de flexão

Para o elemento quadrilátero de 4 nós temos as seguintes equações de interpolação para os eixos x (coordenada natural r) e y (coordenada natural s):

$$\begin{aligned}
h_1 &= \frac{1}{4}(1+r)(1+s) \\
h_2 &= \frac{1}{4}(1-r)(1+s) \\
h_3 &= \frac{1}{4}(1-r)(1-s) \\
h_4 &= \frac{1}{4}(1+r)(1-s)
\end{aligned} \tag{3.4}$$

A partir das equações de interpolação (3.4), montamos a matriz de interpolação (3.5).

$$\mathbf{H} = \begin{bmatrix} 0 & h_1 & 0 & 0 & h_2 & 0 & 0 & h_3 & 0 & 0 & h_4 & 0 \\ 0 & 0 & h_1 & 0 & 0 & h_2 & 0 & 0 & h_3 & 0 & 0 & h_4 \end{bmatrix} \tag{3.5}$$

Para obtermos a matriz de interpolação deformação-deslocamento para flexão ($\mathbf{B}_\kappa$), multiplicamos o operador $[\partial]$ (3.6) pela matriz de interpolação (3.5).

$$[\partial] = \begin{bmatrix} \frac{\partial}{\partial x} & 0 \\ 0 & \frac{\partial}{\partial y} \\ \frac{\partial}{\partial y} & \frac{\partial}{\partial x} \end{bmatrix} \tag{3.6}$$

$$\mathbf{B}_\kappa = [\partial][\mathbf{H}] \tag{3.7}$$

Lembrando que $\mathbf{B}_\kappa$ relaciona as deformações de (3.1) com os graus de liberdade w_i , θ_x^i e θ_y^i , de acordo com (3.8).

$$\begin{aligned}
\begin{bmatrix} \frac{\partial \beta_x}{\partial x} \\ \frac{\partial \beta_x}{\partial y} \end{bmatrix} &= -\frac{1}{4} \mathbf{J}^{-1} \begin{bmatrix} (1+s) & -(1+s) & -(1-s) & (1-s) \\ (1+r) & (1-r) & -(1-r) & -(1+r) \end{bmatrix} \begin{bmatrix} \theta_y^1 \\ \theta_y^2 \\ \theta_y^3 \\ \theta_y^4 \end{bmatrix} \\
\begin{bmatrix} \frac{\partial \beta_y}{\partial x} \\ \frac{\partial \beta_y}{\partial y} \end{bmatrix} &= \frac{1}{4} \mathbf{J}^{-1} \begin{bmatrix} (1+s) & -(1+s) & -(1-s) & (1-s) \\ (1+r) & (1-r) & -(1-r) & -(1+r) \end{bmatrix} \begin{bmatrix} \theta_x^1 \\ \theta_x^2 \\ \theta_x^3 \\ \theta_x^4 \end{bmatrix}
\end{aligned} \tag{3.8}$$

onde $\mathbf{J}$ é a matriz Jacobiana dada por (3.9).

$$\mathbf{J} = \begin{bmatrix} \frac{\partial x}{\partial r} & \frac{\partial y}{\partial r} \\ \frac{\partial x}{\partial s} & \frac{\partial y}{\partial s} \end{bmatrix} \tag{3.9}$$

Neste trabalho está sendo usado o estado plano de tensão mecânica (EPTM), no qual a largura e o comprimento do elemento são muito maiores que a espessura. A matriz de tensão-deformação é dada por (3.10).

$$\mathbf{C}_b = \frac{Eh^3}{12(1-\nu^2)} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{bmatrix} \quad (3.10)$$

onde:

E = Módulo de elasticidade do material;

ν = Coeficiente de Poisson.

Dessa forma a matriz de rigidez é dada por (3.11).

$$\mathbf{K}_b = \int_A \mathbf{B}_\kappa^T \mathbf{C}_b \mathbf{B}_\kappa dA \quad (3.11)$$

Contudo, estamos trabalhando com elementos isoparamétricos, de tal forma que dA é dado por (3.12).

$$dA = \det \mathbf{J} dr ds \quad (3.12)$$

Substituindo (3.12) em (3.11) e separando as integrais de dA obtemos (3.13).

$$\mathbf{K}_b = \int_{-1}^1 \int_{-1}^1 \mathbf{B}_\kappa^T \mathbf{C}_b \mathbf{B}_\kappa \det \mathbf{J} dr ds \quad (3.13)$$

Matriz de rigidez de cisalhamento transversal

Os deslocamentos transversais são definidos pelas equações de interpolação de (3.14)

$$w = \sum_{i=1}^4 h_i w_i$$

$$\begin{bmatrix} \frac{\partial w}{\partial x} \\ \frac{\partial w}{\partial y} \end{bmatrix} = -\frac{1}{4} \mathbf{J}^{-1} \begin{bmatrix} (1+s) & -(1+s) & -(1-s) & (1-s) \\ (1+r) & (1-r) & -(1-r) & -(1+r) \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ w_3 \\ w_4 \end{bmatrix} \quad (3.14)$$

As rotações β_x e β_y são dadas por (3.15).

$$\beta_x = -\sum_{i=1}^4 h_i \theta_y^i$$

$$\beta_y = \sum_{i=1}^4 h_i \theta_x^i \quad (3.15)$$

As equações de interpolação do MITC4 para o cisalhamento transversal são dadas por (3.16) e (3.17).

$$\gamma_{xz} = a_1 + b_1 y \quad (3.16)$$

$$\gamma_{yz} = a_2 + b_2 x \quad (3.17)$$

Aplicando as coordenadas naturais, extraímos a matriz dos coeficientes para cisalhamento transversal (3.18) e de deslocamento-deformação (3.19).

$$\alpha^T = [a_1 \quad b_1 \quad a_2 \quad b_2] \quad (3.18)$$

$$\mathbf{B}_\gamma = \begin{bmatrix} 1 & s & 0 & 0 \\ 0 & 0 & 1 & r \end{bmatrix} \alpha \quad (3.19)$$

Para obtermos os coeficientes de (3.18), aplicamos as relações (3.3), (3.14), (3.15) e (3.16) nos pontos A e C da Figura 3.2, obtendo as relações (3.20) e (3.21).

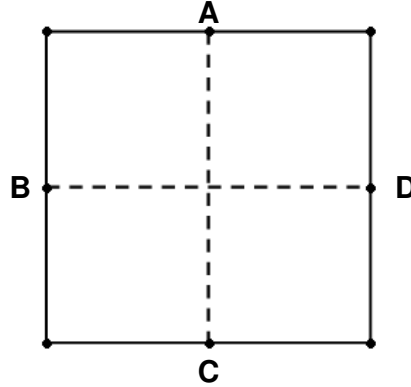


Figura 3.2 - Pontos para interpolação do cisalhamento transversal

$$\gamma_{xz}^A = a_1 + b_1 y^A = \frac{\partial w^A}{\partial x} - \beta_x^A = f^A(w_1, \theta_x^1, \theta_y^1, \dots, w_4, \theta_x^4, \theta_y^4) \quad (3.20)$$

$$\gamma_{xz}^C = a_1 + b_1 y^C = \frac{\partial w^C}{\partial x} - \beta_x^C = f^C(w_1, \theta_x^1, \theta_y^1, \dots, w_4, \theta_x^4, \theta_y^4) \quad (3.21)$$

Montando as relações (3.20) e (3.21) matricialmente, obtemos (3.22).

$$\begin{bmatrix} 1 & y^A \\ 1 & y^C \end{bmatrix} \begin{bmatrix} a_1 \\ b_1 \end{bmatrix} = [R_{2 \times 12}^{A,C}] \begin{bmatrix} w_1 \\ \theta_x^1 \\ \theta_y^1 \\ \vdots \\ w_4 \\ \theta_x^4 \\ \theta_y^4 \end{bmatrix} \quad (3.22)$$

onde $R_{2 \times 12}^{A,C}$ é uma matriz contendo os coeficientes dos graus de liberdade de f^A e f^C .

Isolando o vetor dos coeficientes a_1 e b_1 , obtemos (3.23).

$$\begin{bmatrix} a_1 \\ b_1 \end{bmatrix} = \begin{bmatrix} 1 & y^A \\ 1 & y^C \end{bmatrix}^{-1} [R_{2 \times 12}^{A,C}] \begin{bmatrix} w_1 \\ \theta_x^1 \\ \theta_y^1 \\ \vdots \\ w_4 \\ \theta_x^4 \\ \theta_y^4 \end{bmatrix} \quad (3.23)$$

Realizando o mesmo procedimento para os coeficientes a_2 e b_2 nos pontos B e D, e substituindo em (3.19), obtemos (3.24), a matriz de deslocamento-deformação B_K em função dos graus de liberdade do elemento.

$$B_Y = \begin{bmatrix} 1 & s & 0 & 0 \\ 0 & 0 & 1 & r \end{bmatrix} \begin{bmatrix} \begin{bmatrix} 1 & y^A \\ 1 & y^C \end{bmatrix}^{-1} [R_{2 \times 12}^{A,C}] \\ \begin{bmatrix} 1 & y^B \\ 1 & y^D \end{bmatrix}^{-1} [R_{2 \times 12}^{B,D}] \end{bmatrix} \quad (3.24)$$

Utilizando a matriz tensão-deformação de estado plano de tensão, para γ_{xz} e γ_{yz} , e a matriz de deslocamento-deformação, podemos deduzir a matriz de rigidez K_Y .

$$C_s = \frac{5Eh}{12(1+\nu)} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (3.25)$$

Como no caso da flexão, partimos da expressão (3.26).

$$K_s = \int_A B_Y^T C_s B_Y dA \quad (3.26)$$

Novamente efetuando as mesmas considerações do item 3.4.1.1 em relação às coordenadas naturais e o Jacobiano, obtemos (3.27). [5]

$$K_s = \int_{-1}^1 \int_{-1}^1 B_Y^T C_s B_Y \det J dr ds \quad (3.27)$$

3.1.2 Matriz de rigidez de membrana

O elemento MITC4 possui apenas os três graus de liberdade descritos nos itens anteriores, que são suficientes para calcular as frequências de ressonância relativas à oscilação transversal e à torção (Figura 1.4). Contudo, em nossa análise, existem mais dois graus de liberdade relevantes que são relativos à oscilação lateral. Estes gl podem ser desacoplados e analisados separadamente [7].

Os deslocamentos nos sentidos dos eixos x e y possuem uma componente membrana de deslocamento. Dessa forma, os dois graus de liberdade adicionais que serão tratados neste trabalho serão: u (deslocamento no sentido de x) e v (deslocamento no sentido de y). Estes graus de liberdade são importantes no aumento da rigidez à oscilação lateral. Os movimentos descritos na Figura 1.4 são os mais relevantes no que concerne a vibração, devendo ter suas frequências ressonância maximizadas para aumentar a precisão do movimento do *slider* [8].

A dedução da matriz de rigidez para u e v é bem mais simples que as demais, utilizaremos as definições dos itens anteriores para abreviar a dedução.

Iniciamos com a matriz de interpolação para u e v, utilizando as equações de interpolação e relações de deslocamento nodal (3.4):

$$\begin{aligned} u &= \sum_{i=1}^4 h_i u_i \\ v &= \sum_{i=1}^4 h_i v_i \end{aligned} \quad (3.28)$$

$$\mathbf{H} = \begin{bmatrix} h_1 & 0 & h_2 & 0 & h_3 & 0 & h_4 & 0 \\ 0 & h_1 & 0 & h_2 & 0 & h_3 & 0 & h_4 \end{bmatrix}$$

A partir de (3.28) e utilizando o operador $[\partial]$, montamos a matriz de interpolação deformação-deslocamento para membrana $\mathbf{B}_m$ (3.29).

$$\mathbf{B}_m = [\partial][\mathbf{H}] \quad (3.29)$$

Utilizamos a matriz de tensão-deformação de estado plano de tensões (3.9), para diferenciar apenas utilizaremos como nomenclatura $\mathbf{C}_m$ (mas $\mathbf{C}_m = \mathbf{C}_b$) e montamos a equação da matriz de rigidez da mesma maneira que (3.14) e (3.27), obtendo (3.30) [5].

$$\mathbf{K}_m = h \int_{-1}^1 \int_{-1}^1 \mathbf{B}_m^T \mathbf{C}_m \mathbf{B}_m \det \mathbf{J} dr ds \quad (3.30)$$

3.1.3 Matriz de Massa

A matriz de massa consistente do elemento é dada pela equação (3.31).

$$\mathbf{M} = \int_V \rho \mathbf{H}^T \mathbf{A} \mathbf{H} dV \quad (3.31)$$

onde H é a matriz de interpolação para todos os graus de liberdade da placa e A é dada pela equação (3.32). Neste trabalho serão utilizadas duas matrizes de massa, uma para o elemento MITC4, correspondendo aos graus de liberdade w , θ_x e θ_y , e outra para os efeitos de membrana, correspondendo aos graus de liberdade u e v [5].

$$\mathbf{A} = \begin{bmatrix} h & 0 & 0 \\ 0 & \frac{h^3}{12} & 0 \\ 0 & 0 & \frac{h^3}{12} \end{bmatrix} \quad (3.32)$$

3.2 Método de Otimização Topológica

O método de otimização estrutural busca encontrar um *layout* ideal para uma determinada estrutura, visando melhorar alguma característica específica, por exemplo, rigidez a deformação, rigidez a deformação, tensões internas, etc. Para tanto, faz-se necessário delinear o problema, ou seja, encontrar os parâmetros básicos tais como carregamentos, frequências de operação, fixações, materiais. Deve-se também definir as restrições de massa, volume, deflexão. Definidos estes itens básicos, o problema de otimização busca encontrar a melhor distribuição de material ao longo do domínio especificado. Existem três tipos de abordagem para o problema de otimização: otimização paramétrica, otimização de forma e otimização topológica.

A primeira consiste em, determinar uma geometria, selecionar alguns parâmetros tais como comprimento, largura, espessura, número de vigas, realizando a otimização em função destes parâmetros. Busca-se um determinado grupo de valores que melhor satisfaça as condições impostas a estrutura. Como exemplo, temos a Figura 3.3a que apresenta uma determinada configuração de treliças no início, e após a otimização, são reforçadas as mais importantes, sendo eliminadas as demais.

A otimização de forma busca encontrar o formato da estrutura, tendo como variáveis de projeto, os contornos internos e externos. Visa-se encontrar o formato de domínio que melhor atenda os requisitos

especificados. A Figura 3.3b ilustra um exemplo de otimização de forma, percebe-se a alteração das formas dos furos existentes na estrutura.

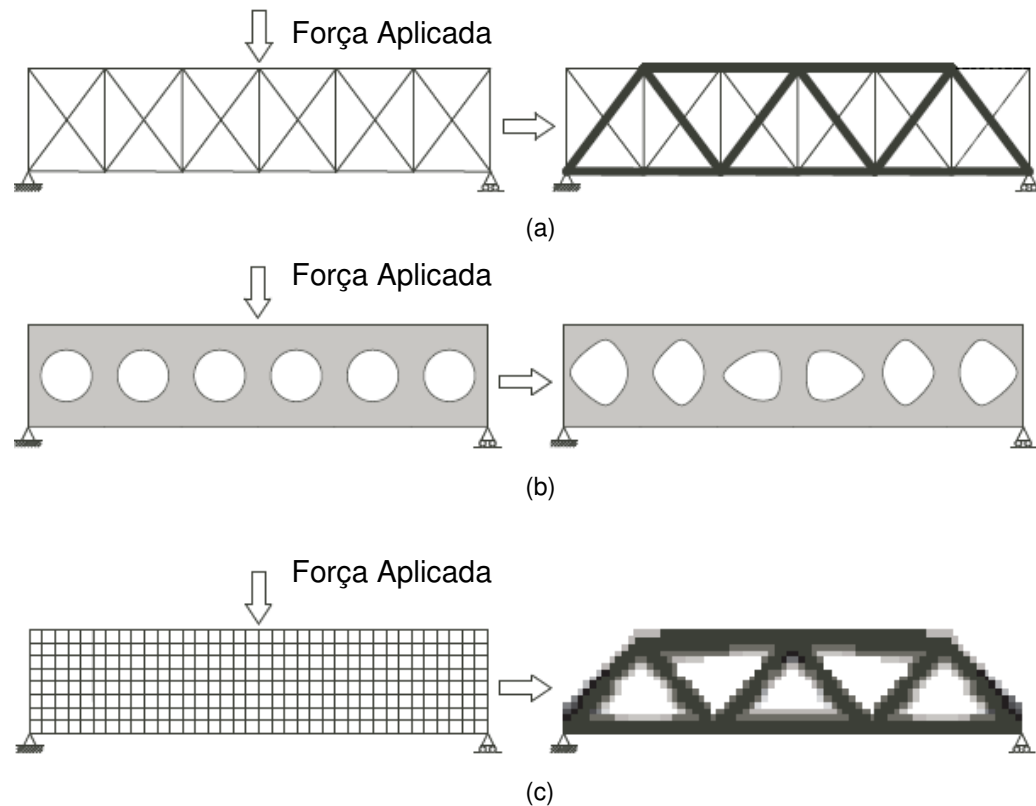


Figura 3.3 - Exemplos de otimização aplicada a projetos de estruturas. (a) Otimização paramétrica; (b) Otimização de forma; (c) Otimização topológica

Por último, temos a otimização topológica que visa distribuir o material ao longo do domínio, podendo gerar regiões vazias dentro da estrutura, definindo o número, tamanho, forma das mesmas e a conectividade entre elas. A Figura 3.3c mostra um exemplo de otimização topológica. [8, 13]

O domínio dentro do qual será feita a distribuição de material é denominado Domínio Fixo Estendido (DFE) e identificado pela letra Ω . Este é normalmente uma região retangular do espaço com dimensões determinadas pelos requisitos do projeto. No DFE são definidos os carregamentos e fixações da estrutura.

Existem basicamente duas maneiras de se abordar um problema de otimização topológica: abordagem micro, baseada no material e abordagem macro, baseada na geometria.

A abordagem micro, utilizada neste trabalho, propõe que a quantidade total de material sólido seja inferior a quantidade necessária para se cobrir todo o DFE. Dessa maneira, adota-se como configuração inicial essa quantidade distribuída igualmente ao longo de todo o DFE, gerando uma microestrutura porosa. Assim, pode-se utilizar uma malha de elementos finitos fixa com elementos de mesmas dimensões. A otimização consiste em variar a densidade de cada elemento de “0” (região vazia) a “1” (região cheia). As propriedades dentro de cada elemento são constantes.

Já a abordagem macro não utiliza uma estrutura porosa, determinada a quantidade de material, ela adiciona e remove material ao longo da estrutura, formando buracos e modificando os contornos. A cada iteração, é necessária uma nova malha de elementos finitos e após a inserção e remoção de material, é realizada uma otimização de forma em cada contorno. [14]

3.2.1 Modelo de material

O modelo de material consiste de uma equação que define as propriedades em cada elemento do DFE, podendo variar entre “0”, ausência de material, e “1”, presença de material. A equação (3.33) apresenta o modelo do material relacionado com as propriedades do mesmo, considerando um material isotrópico.

$$C(x) = \chi(x)C_0 \quad x \in \Omega \quad (3.33)$$

onde C_0 representa a propriedade básica do material e $\chi(x)$ é uma função discreta que determina a presença ou ausência de material de acordo com a equação (3.34)

$$\chi(x) = \begin{cases} 1, & \text{presença de material} \\ 0, & \text{ausência de material} \end{cases} \quad (3.34)$$

Contudo, esta formulação discreta, ou problema 0-1, apresenta um problema de não existência de solução única, pois a adição de buracos tende sempre a melhorar os resultados, desta maneira, é, praticamente, sempre possível encontrar um conjunto novo de parâmetros que melhore a função objetivo. [15]

Para solucionar este problema, adota-se uma relaxação do problema discreto, admitindo que a função $\chi(x)$ possa assumir valores dentro do intervalo $[0,1]$, aumentando assim o número de soluções possíveis, viabilizando a solução do problema. Os estágio intermediários de $\chi(x)$ não possuem um sentido físico, sendo apenas um recurso matemático para se encontrar uma solução para o problema.

O problema da relaxação pode ser tratado através de modelos baseados na microestrutura e modelos artificiais.

Os modelos baseados na microestrutura assumem que cada elemento possui uma microestrutura cuja geometria é parametrizável. O cálculo das propriedades é efetuado por meio do método da homogeneização. A OT, utilizando o método da homogeneização, foi inicialmente aplicado por [16] e utilizava o elemento da Figura 3.4. Neste elemento, α_h e β_h são as dimensões do buraco dentro do elemento, variando seus valores entre “0”, totalmente preenchido pelo material, e “1”, vazio; θ_h é a rotação do elemento. Este método com este mesmo modelo foi examinado novamente e estendido por [17] e utilizado na OT de estruturas, visando melhorar características dinâmicas, tais como as frequências de ressonância. [18]

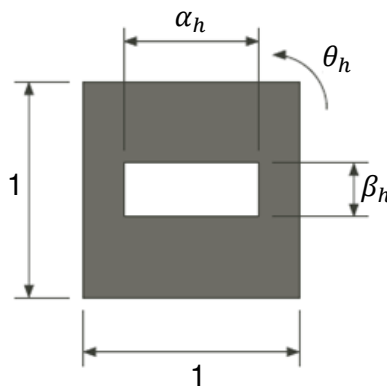


Figura 3.4 - Modelo de material baseado no método de homogeneização.

Os modelos artificiais são interpolações desenvolvidas para simplificar a programação. Estes modelos são aproximações computacionais que interpolam a densidade volumétrica, buscando solucionar o problema 0-1. Como exemplos, podemos citar o modelo SIMP (*“Simple Isotropic Material*

with Penalization”), o modelo RAMP (“Rational Approximation of Material Properties”), o modelo dos limites de Hashin-Shtrikman, o modelo dos limites de Voigt, o modelo dos limites de Reuss, entre outros. Neste trabalho, serão utilizados os modelos SIMP e RAMP, uma explicação dos demais pode ser encontrada na literatura [13,19-22].

O modelo SIMP pode ser traduzido pela equação (3.34).

$$C(x) = \gamma(x)^p C_0 \quad x \in \Omega \quad (3.34)$$

onde o expoente p é um fator de penalização para evitar densidades intermediárias a 0 e 1, pois não possuem um sentido físico e comprometem a qualidade da topologia final. O intervalo de variação de $\gamma(x)$ pode ser visto na equação (3.35), $\gamma(x)$ deve ser maior do que zero para evitar problemas de matrizes singulares que geram problemas computacionais. [19]

$$0 < \gamma_{min} < \gamma(x) < 1 \quad x \in \Omega \quad (3.35)$$

Para problemas estruturais, substituindo C pelo módulo de elasticidade e pela densidade, obtemos (3.36).

$$E(x) = \gamma(x)^p E_0, \quad \rho(x) = \gamma(x)^q \rho_0 \quad (3.36)$$

Valores típicos são $p = 3$ e $q = 1$, contudo para estes valores, na solução de problemas de autovalores, depara-se com o problema de modos localizados em áreas em que $\gamma(x)$ assume valores muito baixos. Isso ocorre, pois o valor da densidade diminui muito mais lentamente do que o módulo de elasticidade, gerando modos como o da Figura 3.5. [23]

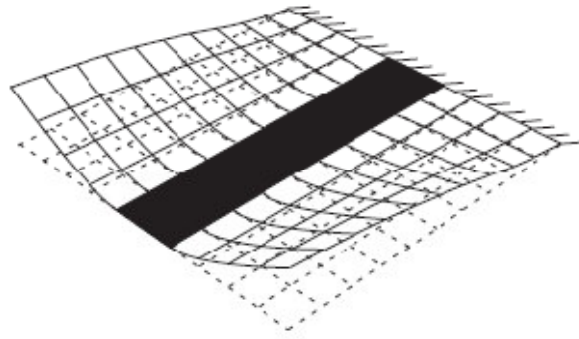


Figura 3.5 - Representação de modos localizados de uma placa em vibração

A formulação do modelo RAMP pode ser vista na equação (3.37).

$$C(x) = \frac{\gamma(x)}{1 + q(1 - \gamma(x))} C_0 \quad x \in \Omega \quad (3.37)$$

Novamente a restrição (3.35) é válida para se evitar problemas de singularidades. O RAMP é proposto para ser utilizado com o módulo de elasticidade, enquanto a densidade seria calculada por meio do SIMP. Dessa forma, os modos localizados seriam eliminados uma vez que a relação (3.38) não tenderia ao infinito em nenhum ponto do intervalo [0,1]. Valores típicos para o RAMP são $p = 3$ e $q = 1$. A Figura 3.6 mostra um gráfico comparando a razão densidade/módulo de elasticidade dos modelos SIMP e RAMP, com seus respectivos valores típicos. [13,24,25]

$$\frac{\rho(x)}{E(x)} = \frac{\gamma(x)^p \rho_0}{\frac{\gamma(x)}{1 + q(1 - \gamma(x))} E_0} = \gamma(x)^{q-1} [1 + p(1 - \gamma(x))] \frac{\rho_0}{E_0} \quad (3.38)$$

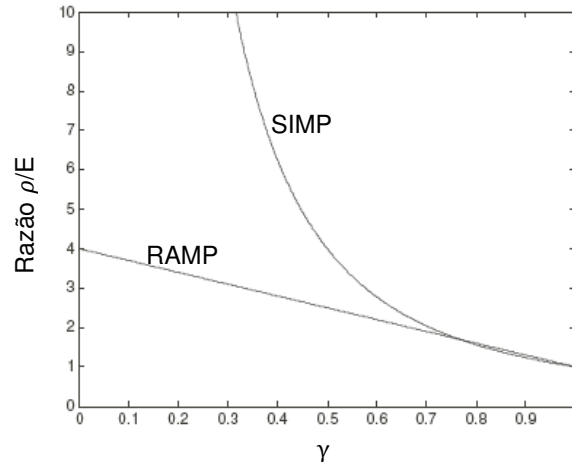


Figura 3.6 - Gráfico da razão ρ/E , comparando os modelos SIMP e RAMP.

3.2.2 Escala de cinza

A relaxação do problema de otimização gera densidades intermediárias que não possuem um sentido físico na topologia final, dificultando os trabalhos de pós-processamento da OT. A Figura 3.7 apresenta o resultado de OT na qual a escala de cinza é presente. Pode-se verificar que a geometria final para fabricação não pode ser identificada com clareza, devendo-se buscar geometrias parecidas que possam ser fabricadas.



Figura 3.7 - Resultado de OT com escala de cinza

Uma das maneiras de se diminuir a escala de cinza é a utilização de penalizações (p e q) no modelo de material, dessa forma, as regiões com densidades intermediárias tornam-se mais “frágeis”, não sendo vantajosas, fazendo com que o programa aproxime-as dos extremos 0-1.

Contudo, penalizações muito elevadas fazem com que o problema retorne ao problema discreto, voltando ao problema da inexistência de solução única. Mesmo penalizações de terceira ordem ($p = 3$) fazem com que o problema adquira um grande número de ótimos locais, terminando o programa antes de encontrar a melhor solução. O gráfico da Figura 3.8 apresenta a comparação dos resultados de uma penalização com $p = 1$ e $p = 3$. Percebe-se que o problema com $p = 3$ apresenta muitos ótimos locais. [26]

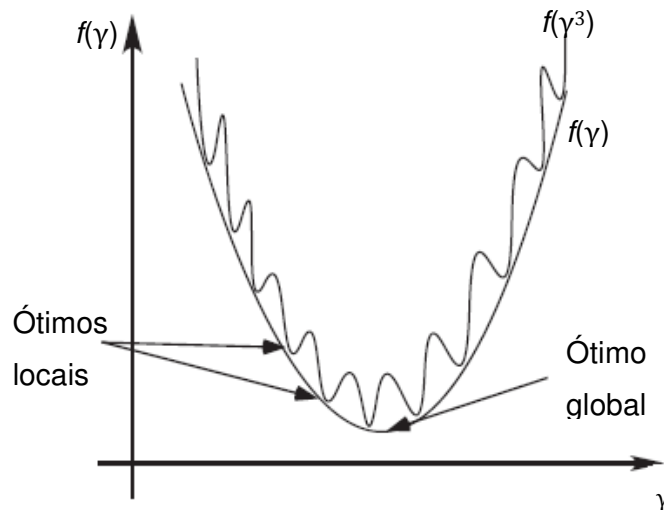


Figura 3.8 - Gráfico comparativo das penalizações e seu efeito na solução final.

A Figura 3.9 mostra a comparação de OT com três valores de p , utilizando o modelo SIMP. A rotina utilizada é a desenvolvida por Sigmund em [27]. Nota-se a diferença na escala de cinza entre as topologias, a única diferença entre os dados de entrada é o valor da penalização p .

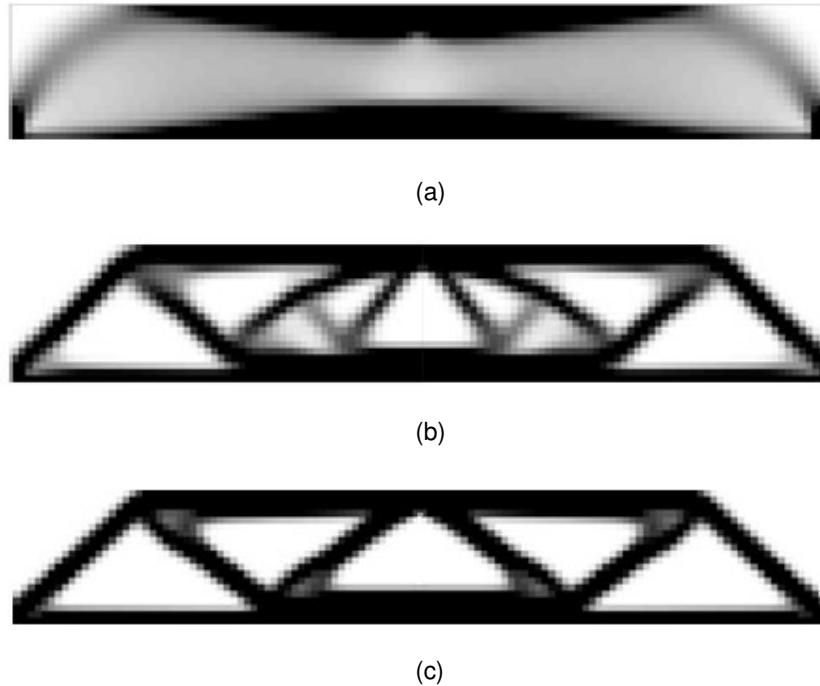


Figura 3.9 - OT de viga engastada para três casos de penalização. (a) $p=1$; (b) $p=2$; (c) $p=3$

Uma solução possível seria iniciar a OT utilizando um valor pequeno de p e, à medida que a solução se aproximasse da solução global, aumentar p para reduzir a escala de cinza. Este método é conhecido como o método da continuidade, normalmente inicia-se com o valor de $p = 1$, terminando com $p = 3$ ou $p = 4$. [13]

Contudo, apesar de alcançar bons resultados, o método da continuidade é um método heurístico e não é possível comprovar que a solução caminhe em direção ao ótimo global por mais gradual que seja a variação de p ou q , mesmo que a topologia final fique isenta de escalas de cinza. [28]

3.2.3 Instabilidade tabuleiro

A “instabilidade tabuleiro” (*Checkerboard pattern*) é um fenômeno caracterizado pela presença de elementos sem material (ilustrados de branco) cercados de elementos com material (ilustrados de preto), disposto como um tabuleiro de xadrez. Um exemplo de topologia com instabilidade tabuleiro pode ser visto na Figura 3.10.



Figura 3.10 - Topologia apresentando instabilidade tabuleiro.

Esse fenômeno ocorre devido a aproximações do MEF que fazem com que a configuração de tabuleiro seja mais rígida ao cisalhamento do que a configuração contínua para uma mesma quantidade de material.

Para solucionar o problema da instabilidade tabuleiro, os principais métodos encontrados na literatura são:

- **Aumento da ordem do elemento finito:** Significa aumentar o número de nós do elemento. Contudo, essa solução é difícil de ser implementada e possui um custo computacional elevado.
- **Uso de filtros:** Utilizando filtros nas sensibilidades das variáveis de projeto em cada iteração. São soluções computacionalmente mais eficientes, mas podem apresentar problemas de mínimos locais e dependência de malha. [15]

Também pode ser encontrado na literatura, o método de aproximação contínua da distribuição de material, conhecido por seu acrônimo em inglês CAMD (*Continuos Approximation of Material Distribution*) cujos autores defendem que está isento do problema de instabilidade tabuleiro. Esse método realiza as interpolações da pseudo-densidade ($\gamma(x)$) em cada nó, ao invés de fazê-lo por elemento. [29]

3.2.4 Dependência de malha

O problema da dependência de malha está ligado ao problema de ausência de solução única para o problema estrutural. Quanto mais refinada a malha de elementos, pode-se aumentar a quantidade de furos na estrutura, mantendo-se o mesmo volume. A Figura 3.11 ilustra a comparação de um mesmo problema para três malhas com elementos diferentes. Percebe-se que quanto mais refinada a malha, maior a quantidade de furos. [30]



(a)



(b)



(c)

Figura 3.11 - Exemplo de dependência de malha. (a) 1200 elementos; (b) 1875 elementos; (c) 3000 elementos

Para corrigir o problema de dependência de malha, pode-se aumentar o número de restrições do problema, neste caso, podem-se aplicar restrições de perímetro, gradientes globais e gradientes locais. Apesar de apresentar bons resultados, este método adiciona muitas restrições ao problema o que é difícil de ser tratado no programa. [15, 26]

Outra maneira de se corrigir o problema da dependência de malha seria implementar um filtro espacial para corrigir os valores da pseudo-densidade ($\gamma(x)$). O valor das pseudo-densidades é calculado por elemento, sendo função dos valores das pseudo-densidades dos elementos que estão dentro de uma circunferência cujo centro é o elemento em questão. Define-se um raio cujo valor é determinado pelas condições de fabricação, este é sempre o mesmo, independente do refinamento da malha. A Figura 3.12 ilustra a circunferência de raio R_{max} , tendo como centro o elemento i e como elementos vizinhos, os elementos j . R_{ij} denota a distância entre o elemento i e o elemento j .

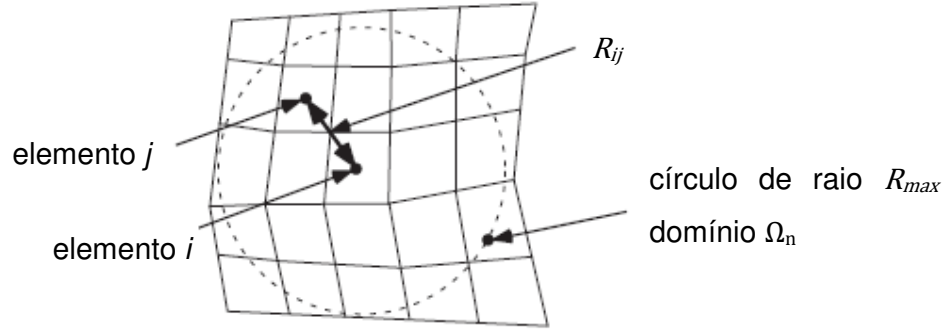


Figura 3.12 - Representação do filtro espacial

Para o cálculo do valor da pseudo-densidade, utiliza-se uma função peso definida pela equação (3.39).

$$p(\mathbf{x}_j - \mathbf{x}_i) = \begin{cases} \frac{R_{max} - R_{ij}}{R_{max}}, & \text{se } \mathbf{x}_j \in \Omega_n \\ 0, & \text{caso contrário} \end{cases} \quad (3.39)$$

onde R_{ij} é dado pela equação (3.40)

$$R_{ij} = \|\mathbf{x}_j - \mathbf{x}_i\| \quad (3.40)$$

Dessa forma, a função das pseudo-densidades é corrigida de acordo com a equação (3.41).

$$\gamma_c(\mathbf{x}_i) = \frac{\sum_{j \in \Omega_n} \gamma(\mathbf{x}_i) p(\mathbf{x}_j - \mathbf{x}_i)}{\sum_{j \in \Omega_n} p(\mathbf{x}_j - \mathbf{x}_i)} \quad (3.41)$$

onde $\gamma_c(\mathbf{x}_i)$ é o valor da pseudo-densidade corrigida. [30]

Existem também os filtros de sensibilidade cujo funcionamento é o mesmo dos filtros espaciais vistos acima, a diferença é que ao invés de ser aplicados as pseudo-densidades, esses filtros são aplicados as sensibilidades do problema. Tratam-se de filtros heurísticos de fácil implementação, baixo custo computacional e que tem produzido resultados muito bons, removendo dependência de malha e instabilidade tabuleiro.

Apesar da teoria por trás destes filtros não ser ainda totalmente conhecida e ainda ser incerto qual problema que está sendo resolvido, sua aplicação em diversos problemas tem mostrado que estes filtros produzem resultados estáveis mesmo com refinamento de malha e mantém o fundo de escala determinado pelo R_{max} escolhido. Além disso, foi constatado que este filtro ainda melhora o comportamento computacional de determinadas

otimizações, retardando a tendência do modelo SIMP de ficar “travado” em configurações 0-1. [13]

3.3 Usinagem fotoquímica

O processo de fabricação escolhido para o protótipo é a usinagem fotoquímica, conhecida também como PCM (*Photochemical Machining*). Neste processo, o material é removido normalmente de chapas de pequena espessura (até 0,0025 mm) por meio de técnicas fotográficas. As etapas do processo consistem em:

1. O desenho da peça é feito em uma escala de até 100 vezes o tamanho da peça original. Em seguida é feito um negativo do desenho, reduzindo-o para o tamanho original da peça. O negativo reduzido é chamado de *artwork*. O desenho aumentado permite que erros inerentes do desenho sejam reduzidos em 100 vezes no *artwork*.
2. A chapa de metal que será usinada é recoberta por um material fotosensível por meio de imersão, spray ou por um processo chamado de *roller coating* no qual o material é aplicado por cilindros que vão rolando por cima da chapa. Em seguida, a chapa é seca em um forno.
3. O negativo é aplicado sobre a peça, expondo-a a luz ultravioleta que endurecem as partes expostas.
4. A chapa é então imersa no reagente ou submetida a um spray do mesmo, corroendo as partes que foram expostas.
5. O material utilizado como máscara é removido e a peça é submetida a um banho para remoção de resíduos químicos.

Este processo requer especialização, contudo apresenta baixo custo ferramental e pode ser automatizado, sendo econômico para médias e grandes produções.

A usinagem fotoquímica é aplicada em peças que apresentam detalhes da ordem de micrômetros e para peças de alta fragilidade que poderiam apresentar deformações muito grandes por meio de outros processos de usinagem, como estampagem. [36]

4 Formulação do problema

O objetivo deste trabalho é gerar uma topologia ideal de uma suspensão para *femto slider*. Por ideal, entende-se que essa suspensão deva possuir baixa rigidez a deslocamentos verticais e movimentos torcionais e alta rigidez a deslocamentos no plano do disco. Além disso, é necessário elevar as frequências de ressonância, uma vez que a rotação dos HDD tem aumentado ao longo do tempo, o que permite maior velocidade de gravação e leitura.

4.1 Otimização de rigidez

A formulação para o aumento da rigidez é dada pela equação (4.1).

$$\begin{aligned} \min: f_s &= \mathbf{U}^t \mathbf{K} \mathbf{U} = \sum_{e=1}^N \gamma_c(x_e) \mathbf{u}_e^t \mathbf{k}_e \mathbf{u}_e \\ \text{tal que: } V_{min} &\leq \int_V \rho dV \leq V_{max} \\ &: \mathbf{K} \mathbf{U} = \mathbf{F} \\ &: 0 < \gamma_{min} \leq \gamma_c \leq 1 \end{aligned} \quad (4.1)$$

onde o subscrito “e” indica que o parâmetro é por elemento; $\gamma_c(x_e)$ representa a pseudo-densidade do elemento; V_{min} representa o mínimo de volume da peça e V_{max} o máximo volume, ambos são percentuais referentes a uma chapa que ocupa todo o domínio de trabalho; γ_{min} é o valor mínimo que se dá a pseudo-densidade para evitar singularidades das matrizes.

O problema de diminuição da rigidez é mais complexo, uma vez que ao apenas inverter o sinal da função objetivo do problema anterior, obter-se-ia uma estrutura com pouco material e sem uma geometria coerente. Dessa forma, torna-se necessária a adição de pelo menos mais uma condição que imponha uma rigidez mínima a estrutura. Neste caso, utiliza-se uma função multiobjetivo que é ajustada por coeficientes de peso, visando à obtenção de um ótimo de Pareto, baseado nos requisitos de projeto.

4.2 Otimização de autovalores

Para a otimização dos autovalores, deve-se tomar cuidado com a inversão dos modos de vibrar. Por exemplo, ao buscar uma topologia que maximize a frequência do primeiro modo de flexão, pode-se diminuir a frequência do primeiro modo de torção. O gráfico da Figura 4.1 ilustra os valores das frequências de ressonância dos três primeiros modos ao longo de um processo de otimização.

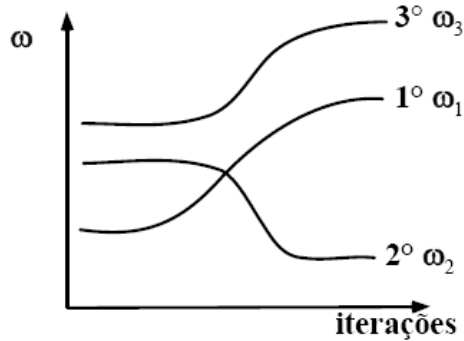


Figura 4.1 - Valores das frequências naturais ao longo de um processo de OT.

Dessa forma, devido ao problema da inversão de modos, deve-se buscar uma formulação que busque maximizar todos os modos, mantendo a menor frequência de ressonância de uma estrutura acima de um valor determinado. [3]

Para se identificar determinado modo de vibrar, utiliza-se método de MAC (*Modal Assurance Criterion*) dado pela equação (4.2).

$$MAC(\phi_a, \phi_b) = \frac{|\phi_a^t \phi_b|^2}{(\phi_a^t \phi_a)(\phi_b^t \phi_b)} \quad (4.2)$$

onde ϕ_a e ϕ_b são dois autovetores que estão sendo comparados. O valor de MAC entre 0 e 1. Quando o MAC é igual a 1, os dois autovetores representam o mesmo modo de vibrar. [31]

Uma solução para o caso da maximização das frequências de ressonância é a maximização de uma frequência de ressonância média. A equação (4.3) apresenta uma maneira de se calcular essa frequência média.

$$\Lambda = \begin{cases} \Lambda_0 + \left(\frac{1}{\eta} \sum_{i=1}^m w_i (\lambda_{ni} - \lambda_{0i})^n \right)^{1/n}, & \text{para } n = \pm 1, \pm 2, \dots; n \neq 0 \\ \Lambda_0 + \exp \left(\frac{1}{\eta} \sum_{i=1}^m w_i \ln (\lambda_{ni} - \lambda_{0i}) \right)^{1/n}, & \text{para } n = 0 \end{cases} \quad (4.3)$$

onde Λ é o valor da freqüência média, Λ_0 e η são valores arbitrários, usados para ajustar o valor da função objetivo, n e λ_{0i} são parâmetros a serem definidos de acordo com o problema de OT, λ_{ni} é o autovalor referente ao modo i , e w_i são coeficientes de peso. [18]

Outra solução possível é utilizar a formulação (4.4). Por essa formulação, a função torna-se uma função multi-objetivo.

$$\begin{aligned} \min: f_v &= \sum_{j=1}^m w_j \left(\frac{\lambda_j - \lambda_0}{\lambda_0} \right)^2 \quad \text{para } j = 1, 2, \dots, m \\ \text{tal que: } V_{min} &\leq \int_V \rho dV \leq V_{max} \\ &: (\mathbf{K} - \lambda \mathbf{M}) \mathbf{U} = 0 \\ &: 0 < \gamma_{min} \leq \gamma_c \leq 1 \end{aligned} \quad (4.4)$$

onde deseja-se aproximar o valor de λ_j ao valor de λ_0 , e w_j são coeficientes de peso. [32] O valor de λ_0 auxilia a determinar o peso de cada modo de vibrar na resposta final, é necessário que se tenha uma idéia da mínima freqüência de ressonância que se deseja obter, definindo λ_0 com um valor um pouco maior do que este mínimo.

Uma terceira solução possível é a formulação (4.5) que também é uma função multi-objetivo.

$$\begin{aligned} \min: f_v &= - \sum_{j=1}^m \frac{\lambda_j^2 - \lambda_0^2}{\lambda_0^2} \quad \text{para } j = 1, 2, \dots, m \\ \text{tal que: } V_{min} &\leq \int_V \rho dV \leq V_{max} \\ &: (\mathbf{K} - \lambda \mathbf{M}) \mathbf{U} = 0 \\ &: 0 < \gamma_{min} \leq \gamma_c \leq 1 \end{aligned} \quad (4.5)$$

onde deseja-se aumentar o valor das freqüências de ressonância. λ_0 é utilizado como um coeficiente de peso que regula a influência de cada freqüência no valor da função, quanto mais distante de λ_0 , mais relevante é o modo de vibrar na otimização.

4.3 Função objetivo

A função objetivo deve quantificar o que se quer otimizar e é função das variáveis de projeto escolhidas. Esta ainda pode ser classificada em simples ou multiobjetivo. A primeira utiliza apenas um critério de otimização, por exemplo, rigidez ou aumento da frequência de ressonância de um modo específico. Já a função multiobjetivo, utiliza mais de um critério na mesma otimização, utilizando coeficientes de peso para aumentar a relevância de um critério em relação aos outros. [33]

Neste trabalho, utilizar-se-á a função multiobjetivo, uma vez que se busca diminuir a rigidez de alguns graus de liberdade, aumentar a de outros e aumentar os valores das frequências de ressonância. A função que será utilizada é a (4.6):

$$\min: f_o = w_s \mathbf{u}_s^t \mathbf{K}_s \mathbf{u}_s - w_r \mathbf{u}_r^t \mathbf{K}_r \mathbf{u}_r - w_v \sum_{j=1}^m \frac{\lambda_j^2 - \lambda_0^2}{\lambda_0^2} \quad (4.6)$$

para $j = 1, 2, \dots, m$

onde w_r , w_s e w_v são coeficientes de peso que devem ser ajustados empiricamente, visando atender os requisitos de projeto. O subscrito “s” indica a rigidez a deslocamentos em x e y, enquanto o subscrito “r” indica a rigidez em z e as torções em x e y. Por meio dessa formulação, busca-se elevar os valores das frequências de ressonância, aumentar a rigidez a deslocamentos no plano do disco e diminuir a rigidez dos deslocamentos verticais e torcionais.

4.4 Análise de sensibilidades

Os gradientes (ou derivadas) da função objetivo e restrições são chamados de sensibilidades do problema de otimização. O cálculo desses gradientes é importante devido a necessidade da linearização da função objetivo em relação às variáveis do problema, como veremos na parte de implementação numérica da rotina de programação linear seqüencial (PLS). Diante desta necessidade, serão apresentados aqui os gradientes da função objetivo, ou sensibilidades, definidas no item anterior. O método utilizado é o método adjunto.

O gradiente da função (4.6) é dado em (4.7):

$$\frac{df_o}{d\gamma_i} = -w_r \mathbf{u}_r^t \frac{d\mathbf{K}_r}{d\gamma_i} \mathbf{u}_r + w_s \mathbf{u}_s^t \frac{d\mathbf{K}_s}{d\gamma_i} \mathbf{u}_s - w_v \sum_{j=1}^m 2 \frac{\lambda_j}{\lambda_0^2} \frac{d\lambda_j}{d\gamma_i} \quad (4.7)$$

para $j = 1, 2, \dots, m$

onde γ_i é a variável de projeto. A derivada de λ_j é dada pela equação (4.8) [32].

$$\frac{d\lambda_j}{d\gamma_i} = \frac{\boldsymbol{\phi}_j^t \left(\frac{\partial \mathbf{K}}{\partial \gamma_i} - \lambda_j \frac{\partial \mathbf{M}}{\partial \gamma_i} \right) \boldsymbol{\phi}_j}{\boldsymbol{\phi}_j^t \mathbf{M} \boldsymbol{\phi}_j} \quad \text{para } j = 1, 2, \dots, m \quad (4.8)$$

Com o intuito de simplificar alguns cálculos e deixar os vetores dos modos de vibrar com valores mais coerentes, é feita uma normalização que altera os vetores de forma a respeitar a equação (4.9).

$$\boldsymbol{\phi}_j^t \mathbf{M} \boldsymbol{\phi}_j = 1 \quad \text{para } j = 1, 2, \dots, m \quad (4.9)$$

5 Implementação numérica

5.1 Implementação do MEF

A estratégia de implementação do MEF foi dividir os graus de liberdade dos elementos em matrizes diferentes. Isso somente foi possível devido à independência dos graus de liberdades referentes a efeitos de membrana (deslocamentos no plano do disco) em relação aos graus de liberdade do MITC4. [5]

Essa separação dos graus de liberdade permite que sejam adotados critérios diferentes de otimização para esses dois graus de liberdade de maneira muito simples. Por exemplo, deseja-se maximizar a rigidez em relação aos efeitos de membrana, ao mesmo tempo em que se deseja minimizar a rigidez em relação aos graus de liberdade do MITC4. Com a separação das matrizes, o cálculo das sensibilidades torna-se muito mais simples, basta calcular sensibilidades diferentes para cada grupo de graus de liberdade. Em suma, essa separação atende perfeitamente as condições mostradas na função objetivo, tornando sua implementação mais simples.

5.2 Programação Linear Seqüencial (PLS)

O método numérico utilizado, neste trabalho, para a solução do problema de otimização é o PLS. Por este método, um problema não linear é dividido em uma série de problemas de otimização lineares que podem ser resolvidos por meio de programação linear (PL). Dentre as vantagens do PLS, pode-se citar a possibilidade de se trabalhar com um grande número de variáveis e restrições de projeto, a necessidade de se definir somente as derivadas de primeira ordem (sensibilidades).

A função objetivo e as restrições precisam ser linearizadas em relação às variáveis de projeto, da forma da equação (5.1):

$$\begin{aligned} \min: f_o &= a_1x_1 + a_2x_2 + \dots + a_nx_n \\ \text{tal que: } g_0(x) &= b_1x_1 + b_2x_2 + \dots + b_nx_n \leq k_0 \\ &: g_1(x) = c_1x_1 + c_2x_2 + \dots + c_nx_n \leq k_1 \\ &: g_n(x) = d_1x_1 + d_2x_2 + \dots + d_nx_n \leq k_n \\ &: x_{min} \leq x \leq x_{max} \end{aligned} \tag{5.1}$$

Embora as funções objetivo mostradas anteriormente não sejam lineares, podemos utilizar a série de Taylor (5.2) com n variáveis para obtermos equações linearizadas.

$$f(x) = f(x_0) + \frac{\partial f}{\partial x_i} \Big|_{x_0} (x_i - x_0) + \dots + \frac{\partial^N f}{\partial x_i^N} \Big|_{x_0} (x_i - x_0)^N + \dots \quad (5.2)$$

para $i = 1, 2, \dots, n$

onde a derivada parcial de primeira ordem de f em relação a x_i é a sensibilidade da função objetivo em relação a essa variável.

As derivadas de maior ordem não são utilizadas, pois necessita-se apenas da primeira derivada para o uso do PLS e como estes termos possuem um valor relativo muito inferior, não acrescentam erro significativo. Dessa forma, desprezando os termos de ordem maior que dois e separando constantes de variáveis, obtemos (5.3).

$$f(x) = \frac{\partial f}{\partial x_1} \Big|_{x_0} x_1 + \dots + \frac{\partial f}{\partial x_n} \Big|_{x_0} x_n + cte \quad (5.3)$$

onde a constante pode ser eliminada do processo de otimização, pois não influencia na variação da função objetivo ao longo das iterações.

Deve-se lembrar que a aproximação feita por meio da série de Taylor somente é válida para um pequeno intervalo ao redor de x_0 , então, ao utilizar o PLS, devem-se restringir os valores que cada variável pode assumir na iteração, pois se o intervalo for muito grande, a aproximação não será mais válida. A Figura 5.1 ilustra bem esse conceito.

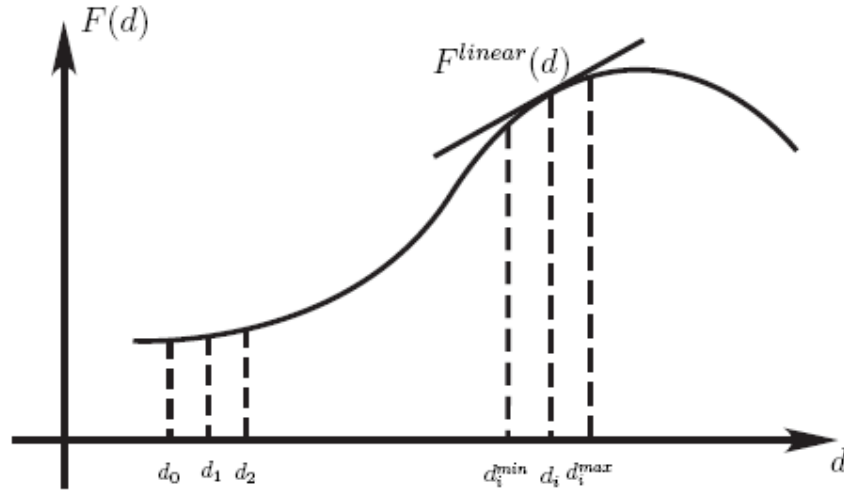


Figura 5.1 - Função a ser linearizada e sua aproximação linear ao redor de d_i .

Nota-se claramente que a função aproximada em d_i não é válida para d_1 , por exemplo. Assim, é necessário entrar com limites, dentro dos quais a aproximação é válida. Outro detalhe importante que se pode inferir da Figura 5.1 é que quanto maior a curvatura, ou seja, quanto maior o valor absoluto da segunda derivada, menor deve ser o intervalo escolhido, pois a aproximação é menos precisa. Isso faz sentido ao se analisar a função de Taylor, uma vez que o primeiro termo a ser desprezado é a segunda derivada.

O algoritmo de PLS utilizado é o Simplex e a função do Matlab que adota este método é a `linprog` cuja definição é dada pela Figura 5.2.

$$\begin{aligned} \min: & \mathbf{f}^T \mathbf{x} \quad \text{tal que: } \mathbf{Ax} \leq \mathbf{b} \\ & : \mathbf{A}_{eq} \mathbf{x} \leq \mathbf{b}_{eq} \\ & : \mathbf{lb} \leq \mathbf{x} \leq \mathbf{ub} \\ \text{linprog} & (\mathbf{f}, \mathbf{A}, \mathbf{b}, \mathbf{A}_{eq}, \mathbf{b}_{eq}, \mathbf{lb}, \mathbf{ub}, \mathbf{x}_0) \end{aligned}$$

Figura 5.2 - Função do Matlab que utiliza o PLS.

onde $\mathbf{A}$ e $\mathbf{A}_{eq}$ são matrizes e os demais itens são vetores.

5.3 Limites móveis

Neste trabalho, propõe-se calcular o valor da segunda derivada por meio do método adjunto e utilizá-lo para definir os limites de valores que

cada variável pode assumir em uma iteração. A princípio, calculam-se as derivadas segundas das funções objetivas definidas anteriormente.

A derivada de segunda ordem da função (4.6) é dada em (5.4).

$$\begin{aligned} \frac{d^2 f_o}{d\gamma_i^2} = & -w_s \mathbf{u}_s^t \frac{d^2 K_s}{d\gamma_i^2} \mathbf{u}_s + w_r \mathbf{u}_r^t \frac{d^2 K_r}{d\gamma_i^2} \mathbf{u}_r - \dots \\ & \dots - w_v \sum_{j=1}^m \left(\frac{2}{\lambda_0^2} \left(\frac{d\lambda_j}{d\gamma_i} \right)^2 + 2 \frac{\lambda_j}{\lambda_0^2} \frac{d^2 \lambda_j}{d\gamma_i^2} \right) \end{aligned} \quad (5.4)$$

onde a derivada segunda de λ_j é dada por (5.5).

$$\begin{aligned} \frac{d^2 \lambda_j}{d\gamma_i^2} = & \frac{aux(\gamma_i)(\phi_j^t \mathbf{M} \phi_j) - \left(\phi_j^t \left(\frac{\partial K}{\partial \gamma_i} - \lambda_j \frac{\partial \mathbf{M}}{\partial \gamma_i} \right) \phi_j \right) \left(\phi_j^t \frac{\partial \mathbf{M}}{\partial \gamma_i} \phi_j \right)}{(\phi_j^t \mathbf{M} \phi_j)^2} \\ aux(\gamma_i) = & \left(\phi_j^t \left(\frac{\partial^2 K}{\partial \gamma_i^2} - \frac{d\lambda_j}{d\gamma_i} \frac{\partial \mathbf{M}}{\partial \gamma_i} - \lambda_j \frac{\partial^2 \mathbf{M}}{\partial \gamma_i^2} \right) \phi_j \right) \end{aligned} \quad (5.5)$$

Como a derivada segunda está relacionada com a curvatura da curva, o importante para este trabalho é sua amplitude, portanto utilizar-se-ão apenas os valores absolutos. As variáveis de projeto podem sofrer alterações que variam entre 1-15% em relação a seus valores atuais, intervalo geralmente utilizado na literatura [34]. Desta forma, quanto maior a amplitude, mais próximo de 1% será a variação, para regiões cuja derivada segunda seja próxima de zero, o valor da alteração será mais próximo de 15%.

A implementação é feita de acordo com a equação (5.6).

$$\begin{aligned} \mathbf{lb} = & \left(0,85 + 0,14 \left(\frac{\gamma'' - \gamma''_{min}}{\gamma''_{max}} \right) \right) \gamma \\ \mathbf{ub} = & \left(1,15 - 0,14 \left(\frac{\gamma'' - \gamma''_{min}}{\gamma''_{max}} \right) \right) \gamma \end{aligned} \quad (5.6)$$

onde $\mathbf{lb}$ corresponde ao limite inferior e $\mathbf{ub}$ ao limite superior da pseudo-densidade em um iteração; γ'' é o vetor que contém todas as derivadas de segunda ordem, γ''_{min} é o menor valor de γ'' , e γ''_{max} é o maior valor; γ é o vetor das pseudo-densidades

5.4 Instabilidades numéricas

Verificou-se durante os testes do programa, uma série de instabilidades numéricas que levavam a resultados incorretos. A seguir serão descritos os principais problemas e correções adotadas.

5.4.1 Falta de simetria em matrizes

Durante o processo de montagem das matrizes de massa e rigidez, foram detectados problemas de simetria destas matrizes, fato que gerava instabilidades nos resultados da função *eigs* do matlab. Realizou-se então uma verificação da ordem do erro que gerava a falta de simetria, para tanto, utilizou-se a seguinte expressão (5.7).

$$\max (\max (\text{triu}(K) - \text{triu}(K')))) \quad (5.7)$$

onde K é a matriz que se está analisando, a função *triu* retorna a matriz triangular superior, o apóstrofe em K' indica a matriz transposta. Dessa forma, busca-se a maior diferença entre a matriz triangular superior e inferior de uma matriz. O erro encontrado era da ordem de 10^{-17} em relação à ordem da matriz. Nota-se claramente que se trata de um erro numérico que pode ser removido sem alterar as propriedades da matriz. A solução implementada está na expressão (5.8).

$$K = \text{triu}(K) + \text{triu}(K, 1)' \quad (5.8)$$

onde o termo $\text{triu}(K, 1)'$ representa a matriz triangular superior sem a diagonal principal, transposta para se tornar a matriz triangular inferior. A escolha de utilizar a matriz triangular superior ao invés da inferior é arbitrária e não influencia nos resultados.

5.4.2 Unidades de medida

Neste trabalho, o domínio de trabalho possui dimensões muito reduzidas da ordem de 10^{-3} a 10^{-5} m. Se considerarmos essas medidas em unidades SI, o problema apresentará diversos problemas de mau condicionamento numérico. Alguns problemas encontrados foram autovalores complexos e negativos, soluções do PLS alteradas de acordo com a discretização e cálculo dos deslocamentos errados. Para resolver este problema, alteraram-se todas as unidades para milímetros ao invés de

metro, solucionando assim os problemas de instabilidade numérica. Ao analisar artigos que tratam de OT, verifica-se que todos utilizam este tipo de conversão de unidades para melhorar o condicionamento numérico,. [8], [10], [35].

Portanto, as dimensões da peça serão inseridas em milímetros e as propriedades do material obedecerão à Tabela 6.1.

5.5 Restrições de massa

Como o processo de OT visa encontrar uma geometria ideal para uma menor quantidade de material, insere-se uma restrição para a “massa total” (somatória das pseudo-densidades) da peça. Determina-se a quantidade total de material que o programa pode distribuir. Neste trabalho, foi adotado um limite máximo de 50% do somatório total das pseudo-densidades de uma placa cheia caso em que todas as pseudo-densidades são iguais a 1.

Devido ao problema de minimização da rigidez, também foi necessário adotar um valor mínimo, para que a peça sempre mantivesse uma geometria coerente. Dessa forma, adotou-se como limite mínimo para a “massa total” o valor de 40%.

5.6 Critérios de convergência

A convergência do processo de otimização topológica foi feita por meio de três critérios, sendo que basta atingir o mínimo de qualquer um dos critérios para que o programa considere a convergência. Os valores mínimos foram escolhidos empiricamente, após analisar os resultados de diversos processos. Os três critérios são:

1. Maior variação da densidade de um elemento inferior a 0,001 (a variável de projeto varia de 0,0001 a 1,0);
2. Porcentagem de alteração da massa total da peça inferior a 0,3%. Realiza-se a soma das diferenças dos valores de todas as variáveis de projeto da iteração n e da iteração $n-1$, dividindo o resultado pela soma do valor de todas as variáveis da iteração n ;
3. Porcentagem de elementos que tiveram o valor da variável de projeto alterado inferior a 1%. Para evitar que variações devidas a

aproximações numéricas sejam levadas em consideração, consideram-se apenas as diferenças superiores a 0,002.

5.7 Cálculo de autovalores

O cálculo dos autovalores representa uma parte de custo computacional elevado para o programa. Dessa forma, é imperativo a busca por uma função que apresente um custo computacional o menor possível. O Matlab dispõe de duas funções principais para o cálculo de autovalores e autovetores, tratam-se das funções “eig” e “eigs”, a primeira é utilizada para matrizes cheias e a segunda para matrizes esparsas. Contudo, essa não é a única diferença entre as duas, a função “eigs” permite que sejam calculados apenas alguns modos de vibrar, utilizando a opção ‘sm’, utilizada neste trabalho, permite-se que sejam calculados somente os ‘n’ menores autovalores, onde ‘n’ também é um parâmetro da função.

Foram realizados testes com as duas funções e, uma vez feitas as correções apresentadas no item 5.4, os resultados obtidos eram os mesmos. A diferença encontrada foi no tempo de processamento de cada uma. Para se ter uma idéia da diferença de desempenho, utilizando a mesma máquina para processar o algoritmo com a função “eig” e com a função “eigs”, no primeiro o tempo de processamento de 80 iterações foi de mais de 11 horas, enquanto para o segundo caso gasta-se menos de 10 minutos, considerando que no caso da função “eigs” buscaram-se apenas os 3 menores autovalores. Dessa forma, adotou-se a função “eigs” neste trabalho.

Outro fator relevante que foi levantado no cálculo das frequências de ressonância é o número de modos que são realmente necessários para a otimização topológica. Como a parte da função multiobjetivo que trata esse problema depende diretamente do valor da frequência, verificou-se que o único modo que realmente influencia nos resultados e necessita ser elevado é o referente a menor frequência de ressonância, pois esta estando acima da frequência mínima necessária, todas as demais frequências também estarão. Dessa forma, os testes executados consideraram apenas a influência da menor frequência de ressonância em cada iteração. Como será

mostrado na seção 6, não houve inversão de modos de vibrar em nenhum dos resultados finais.

O valor que será utilizado para λ_0 é 36, pois feitas as alterações do item 5.4.2, os autovalores são retornados em kHz. Dessa forma, utiliza-se 40 pois seria aproximadamente uma frequência de 1000Hz, de acordo com a expressão (5.9).

$$\lambda = \omega^2 = (2\pi f)^2 \quad (5.9)$$

Este valor foi adotado, pois a rotação de um HDD atual é da ordem de 10000 rpm, que representa aproximadamente 167Hz, dessa forma, torna-se necessário que a frequência do primeiro modo seja bem superior a 167Hz, lembrando que a função busca aproximar a frequência da peça a λ_0 , não necessariamente igualando ou ultrapassando. Verificando trabalhos previamente publicados [8], para essas condições da peça, os valores encontrados variam de 300Hz a 1000Hz, o que mostra que o valor escolhido para λ_0 é adequado.

5.8 Simetria da otimização

Durante a execução de algumas otimizações, verificou-se que muitos resultados não estavam simétricos. Apesar da simetria em relação ao eixo horizontal que passa no meio da peça existir, não se pode trabalhar apenas com metade da área de trabalho, pois perder-se-iam determinados modos de vibrar. Dessa forma, a solução adotada foi forçar a simetria ao final de cada iteração. O procedimento é simples, consiste em espelhar a densidade dos elementos da metade superior para a metade inferior. A expressão (5.10) apresenta o procedimento que foi feito no Matlab.

$$\text{dn}(\text{floor}(\text{nely}/2+1.5):\text{nely},:) = \text{flipud}(\text{dn}(1:\text{floor}(\text{nely}/2),:)); \quad (5.10)$$

A função “flipud” realiza o espelhamento da matriz superior. Esta alteração não torna a solução incoerente, pois ela auxilia modifica as sensibilidades e os limites superior e inferior, fazendo com que o “linprog” rode com as novas condições.

5.9 Funcionamento do programa

O programa de otimização implementado segue o fluxograma da **Erro! Fonte de referência não encontrada.** Os dados iniciais do material, da discretização, das dimensões são passados ao programa para poder começar a otimização. Em seguida, o programa gera valores para as variáveis de projeto. Logo no início, também são calculadas as matrizes de rigidez e massa do elemento, este processo é executado apenas uma vez, pois não há distorção do elemento, não havendo necessidade de recalculá-las em cada iteração. Definidos estes parâmetros, entra-se no módulo de MEF, no qual a matriz referente ao MITC4 é trabalhada de maneira separada da matriz de rigidez de membrana, isso é possível porque ambas são independentes. Calculados os deslocamentos e frequências por MEF, o programa calcula as sensibilidades e suas derivadas. Depois, o filtro espacial é aplicado às sensibilidades, gerando novos valores para as mesmas. Com as derivadas de segunda ordem, são calculados os intervalos de variação de cada pseudo-densidade. Então, todos esses dados são passados ao “linprog” que executa o PLS, gerando novos valores para as variáveis de projeto.

A convergência é verificada por meio da porcentagem de alteração entre a iteração $n-1$ e a iteração n , por exemplo, se a variação percentual das pseudo-densidades for menor do que 0,2%, o processo é interrompido, pois a convergência já foi alcançada, caso contrário, recomeça-se o processo.

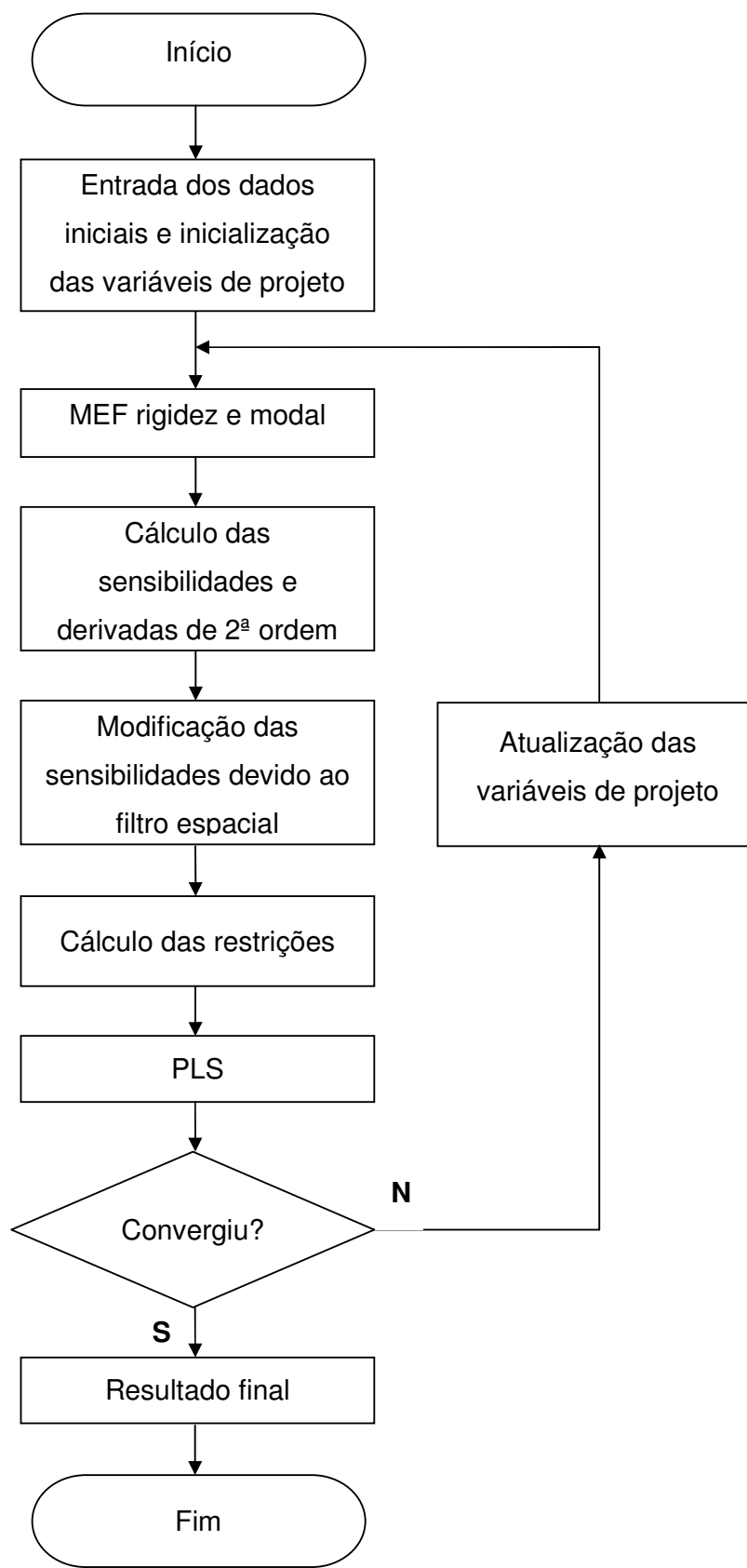


Figura 5.3 - Fluxograma de funcionamento do programa.

6 Resultados

6.1 Domínio e condições de contorno

A partir da análise da literatura ([8], [10]) sobre projeto de suspensão para *femto slider* e da suspensão de um HDD de 80 Gb (Maxtor STM380215AS) foi possível definir um domínio de trabalho e material a serem utilizados.

O material escolhido foi o aço inoxidável cujas propriedades são dadas pela Tabela 6.1 na coluna “Unid. SI”. Contudo, como foi explicado no item 5.4.2, os dados utilizados no programa serão os da coluna “Entrada”. Todos os testes e análises foram feitos utilizando-se esses dados de material.

Tabela 6.1 - Propriedades do aço inoxidável consideradas para a OT

Propriedades	Unid. SI	Entrada
E (módulo de elasticidade)	210 GPa	210 GPa
ρ (densidade)	7800 kg/m ³	7800 kg/m ³
ν (coeficiente de Poisson)	0,30	0,30

O domínio de projeto definido está representado na Figura 6.1.

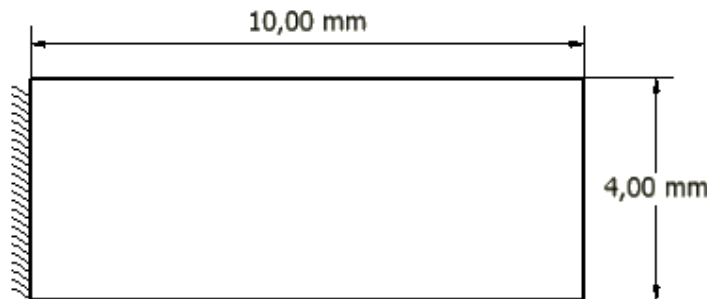


Figura 6.1 - Domínio de trabalho.

A espessura da chapa utilizada é de 0,06 mm. A parte lateral esquerda está engastada. Todos os testes serão feitos com este domínio, a não ser que seja mencionado o contrário.

6.2 Validação dos limites móveis

Foram realizados dois testes utilizando os valores da derivada segunda aplicados a problemas simples de rigidez. As condições da placa

para o primeiro teste encontram-se na Figura 6.2. Os pesos utilizados foram $w_m = 0$, $w_r = -1$, $w_v = 0$, pois objetivou-se apenas realizar uma otimização de rigidez simples para avaliar a validade dos limites móveis calculados com derivada segunda. Foi aplicado um filtro com resolução de 1,0 mm.

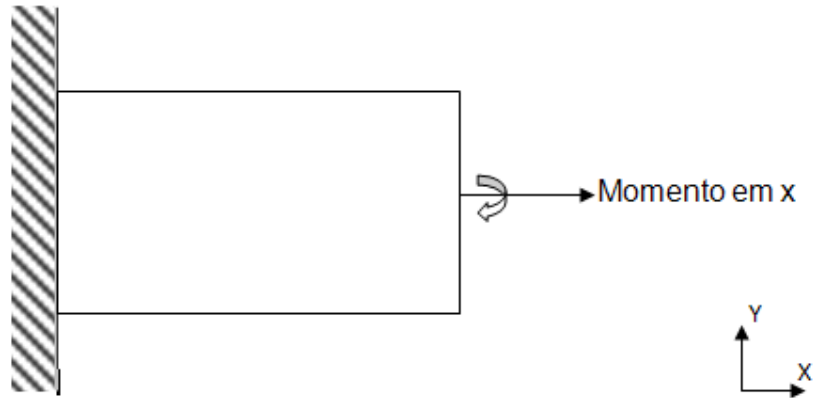


Figura 6.2 - Placa com momento unitário em x aplicado em extremidade livre.

Os resultados são apresentados na Figura 6.3.



(a)



(b)

Figura 6.3 - Comparação entre limites fixos e limites calculados com base na derivada segunda. (a) Utilizando a derivada segunda em uma malha de 32x80 elementos (48 iterações). Raio do filtro espacial de 1 mm; (b) Variação fixa em 10% em uma malha de 32x80 elementos (67 iterações). Raio do filtro espacial de 1 mm.

Nota-se que as geometrias obtidas são as mesmas. O critério de convergência é o mesmo para ambos. Contudo verifica-se que o primeiro, precisou de cerca de 30% menos iterações que o segundo, representando um ganho de desempenho considerável.

O segundo teste está ilustrado na Figura 6.4.

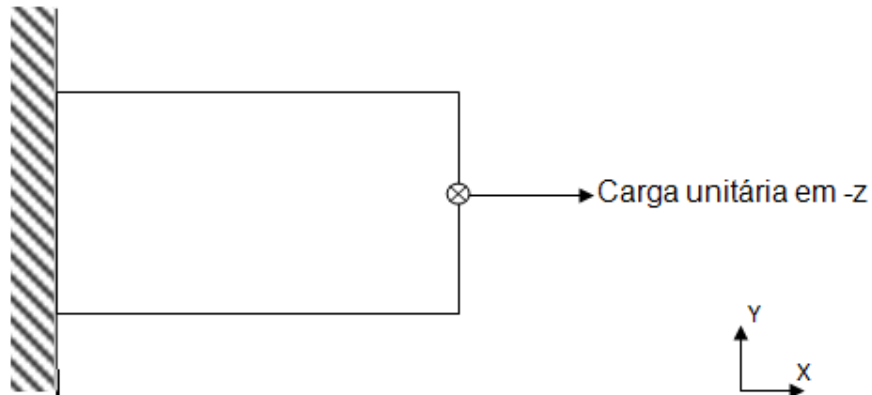


Figura 6.4 - Placa com carga unitária no sentido de z negativo aplicado em extremidade livre.

Os resultados podem ser vistos na Figura 6.5.

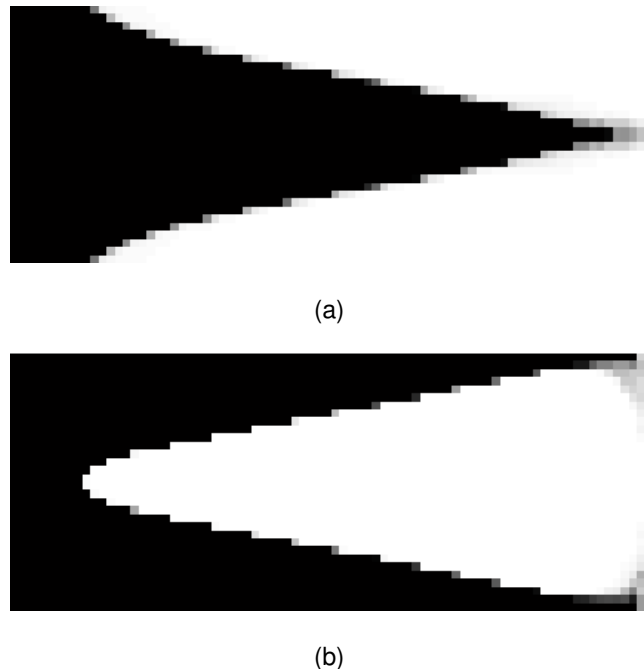


Figura 6.5 - Comparação entre limites fixos e limites calculados com base na derivada segunda para o teste com carga vertical. (a) Utilizando derivada segunda em uma malha de 32x80 elementos (38 iterações); (b) Variação fixa em 10% em uma malha de 32x80 elementos (86 iterações).

Verifica-se que no caso da variação fixa, o programa encontra um mínimo local e leva muito mais iterações para convergir. Por outro lado, com o uso da derivada segunda, a convergência é obtida com poucas iterações, definindo uma topologia coerente.

Portanto, pode-se concluir que o uso da segunda derivada melhora o desempenho e auxilia na busca do mínimo global do problema de otimização.

6.3 Validação do filtro

O filtro tem como objetivo, eliminar a instabilidade tabuleiro e a dependência de malha. Para a validação do filtro, foi realizado o mesmo teste da Figura 6.6 para dois casos: sem filtro e com filtro espacial de sensibilidades. Os pesos utilizados foram $w_m = 1$, $w_r = 0$, $w_v = 0$, pois objetivou-se apenas realizar uma otimização de rigidez simples para avaliar a validade do filtro.

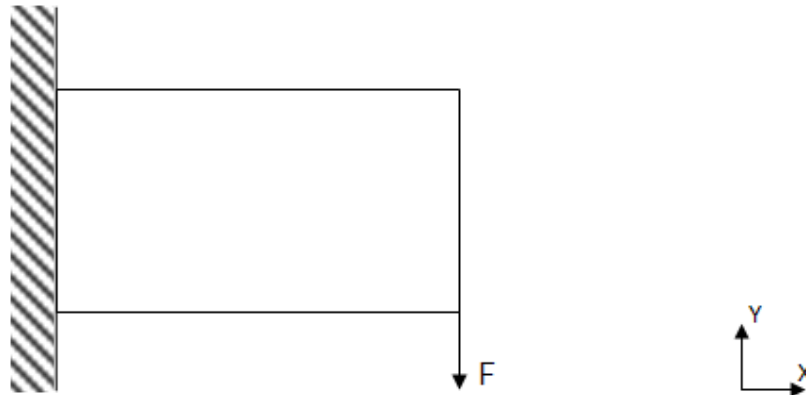
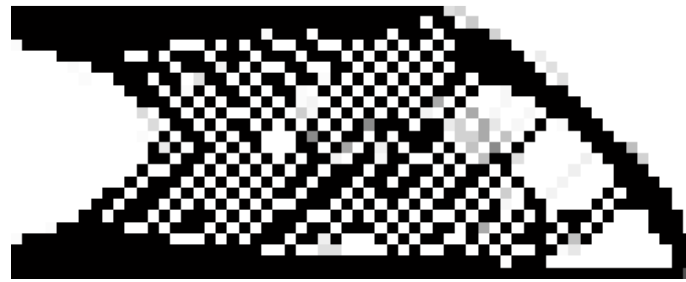


Figura 6.6 - Placa com carga unitária no sentido de y negativo aplicada no canto inferior esquerdo da placa.

Primeiramente, foi testada a otimização sem a utilização de nenhum filtro para dois casos de discretização. Os resultados são apresentados pela Figura 6.7.



(a)



(b)

Figura 6.7 - Otimização de rigidez a torção sem a utilização de filtro. (a) 1440 elementos (24x60) – 32 iterações; (b) 4000 elementos (40x100) – 46 iterações.

Observa-se a grande quantidade de regiões sob o efeito da instabilidade tabuleiro. Verifica-se também que as soluções não são exatamente as mesmas, caracterizando o problema de dependência de malha.

No teste a seguir, foi testado o filtro espacial aplicado às sensibilidades com resolução de 1 mm. Os resultados podem ser vistos na Figura 6.8.



(a)



(b)

Figura 6.8 - Otimização de rigidez a torção com a utilização de filtro espacial sobre as sensibilidades. (a) 1440 elementos (24x60) – 86 iterações; (b) 4000 elementos (40x100) – 58 iterações.

Percebe-se que as duas topologias obtidas são as mesmas, ou seja, sem dependência de malha. Os resultados também não apresentam problemas de escala de cinza, nem instabilidade tabuleiro. Esses resultados validam o filtro escolhido.

6.4 Comparação com o algoritmo de 99 linhas.

Foi realizada uma comparação dos resultados obtidos pelo algoritmo criado neste trabalho e os obtidos por um algoritmo de 99 linhas desenvolvido por Sigmund [27]. O teste realizado é o mesmo da figura 6.9 e é mencionado no paper de Sigmund. A malha utilizada é de 20x60 elementos. A função de chamada do algoritmo de 99 linhas é dado pela expressão (6.1).

$$\text{top}(60,20,0.5,3.0,1.5) \quad (6.1)$$

onde os dois primeiros parâmetros definem a malha, o terceiro parâmetro define o volume máximo de trabalho, o quarto parâmetro define a penalização do modelo de material (SIMP) e o quinto define a resolução do filtro. O resultado encontra-se na Figura 6.9.



Figura 6.9 - Resultado do 99 linhas – 94 iterações.

Os dados de entrada para o programa deste trabalho foram os mesmos. Apesar das entradas serem diferentes, o 99 linhas utiliza uma entrada baseada no número de elementos vizinhos, os filtros aplicados são os mesmos. O resultado obtido pelo programa é apresentado na Figura 6.10. Neste caso, os pesos utilizados foram $w_m = 1$, $w_r = 0$, $w_v = 0$; o domínio possui 12,0x4,0x0,06 mm e o filtro com resolução de 1,0 mm.

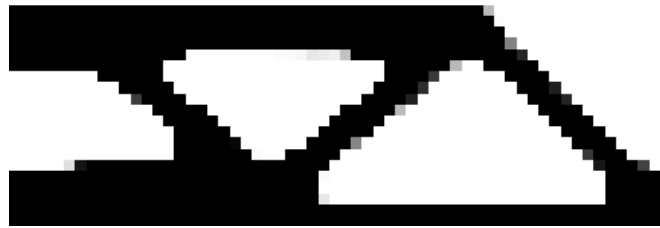


Figura 6.10 - Resultado do algoritmo implementado – 72 iterações.

Percebe-se a menor quantidade de escala de cinza e que a convergência é obtida em menos iterações no algoritmo implementado, sendo que a geometria é a mesma para os dois algoritmos.

6.5 Análise da suspensão de um HDD atual.

A análise da suspensão de um HDD de 80Gb da Maxtor modelo STM380215AS, que já utiliza o *femto slider*, foi feita para critério de comparação com os resultados obtidos pelo programa implementado. Os critérios para essa análise foram:

- Utilizar o mesmo material aplicado nas outras otimizações cujas propriedades estão descritas no item 4.6;
- Utilizar a mesma espessura de 0,06 mm;

Não houve a necessidade de se alterar o domínio, uma vez que a peça está totalmente dentro do domínio estabelecido no item 4.6. A Figura 6.11 mostra uma foto da suspensão utilizada.



Figura 6.11 - Foto da suspensão do HDD de 80Gb.

As análises foram realizadas utilizando-se o software ANSYS. A Figura 6.12 mostra o modelo criado e a Tabela 6.2 apresenta os resultados obtidos pelo software. Serão apresentados os resultados de rigidez e primeiros modos de vibrar correspondentes aos graus de liberdade da Figura 1.4, sendo estes: deslocamento vertical, torção ao redor de x e deslocamento lateral. Lembrando que o objetivo é minimizar rigidez flexão e torção e maximizar as demais características.

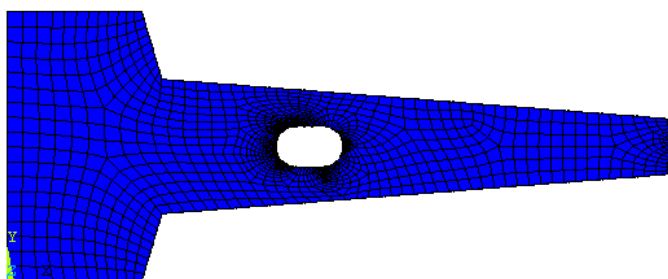


Figura 6.12 - Modelo de MEF da suspensão real.

Tabela 6.2 - Resultados para a suspensão real.

Características	Valor
k_{rotx} (rigidez)	$7,5 \times 10^{-4}$ Nm/rad
k_z (rigidez)	24,8 N/m
k_y (rigidez)	23,6 kN/m
1º flexão	547 Hz
2º torção	6837 Hz
3º lateral	18681 Hz

6.6 Projeto da suspensão

Uma vez terminado o programa, iniciaram-se as otimizações para buscar a topologia ideal da suspensão para *femto slider*. A Figura 6.13

apresenta as principais cargas que serão utilizadas como condições de contorno, os valores serão alterados em cada otimização.

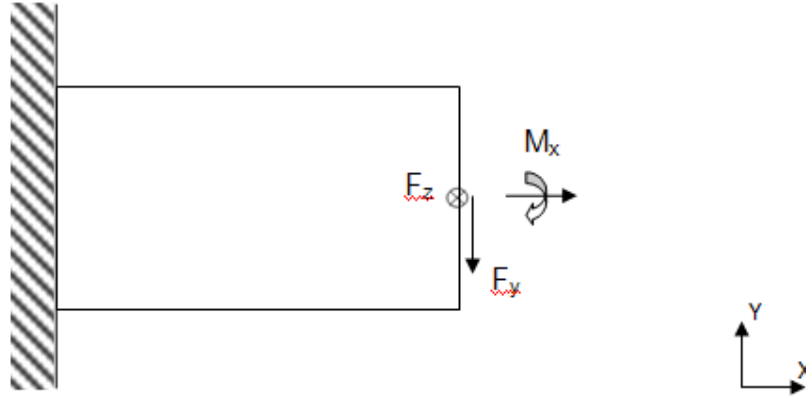


Figura 6.13 - Principais cargas que serão utilizadas.

A seguir serão mostrados os resultados obtidos com o programa.

6.6.1 Geometria inicial

Verificou-se que os problemas de aumento de freqüências de ressonância e diminuição da rigidez possuem muitos mínimos locais e apresentam muitos resultados infactíveis. Para um primeiro exemplo, adotaram-se as cargas da Figura 6.13 com valor unitário e os pesos utilizados foram: $w_m = 1$, $w_r = 1$, $w_v = 1$. O resultado pode ser visto na Figura 6.14.



Figura 6.14 - Resultado de teste inicial.

Verifica-se claramente que a figura não é factível e que os pesos precisam ser ajustados convenientemente. Neste caso, Houve predomínio completo da diminuição da rigidez. Não resultou na placa vazia, pois havia o limite mínimo de massa total.

Para evitar o problema de mínimos locais excessivos, primeiro realizou-se a convergência para uma peça de alta rigidez a flexão, bastando utilizar cargas unitárias e pesos $w_m = 0$, $w_r = -1$, $w_v = 0$. Verificou-se que a partir da 20ª iteração a peça já apresenta a geometria desejada, independente do tamanho da malha escolhida, a Figura 6.15 apresenta o resultado dessa otimização na 20ª iteração para uma malha de 80 x 32 elementos.

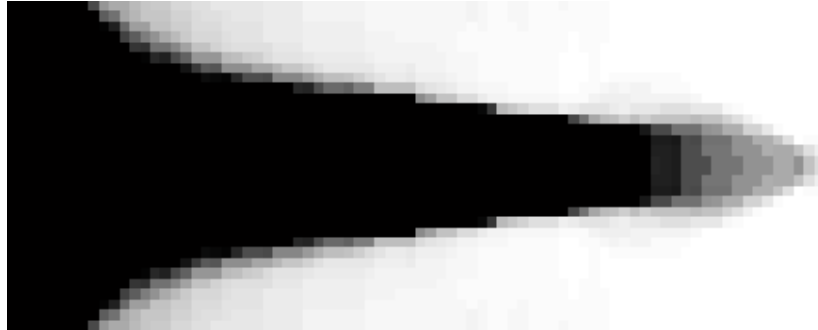


Figura 6.15 - Otimização de rigidez a flexão na 20ª iteração.

Dessa forma, a partir da 21ª iteração são acrescentados os demais critérios. Este procedimento foi adotado para todas as otimizações, pois obtém geometrias mais coerentes.

6.6.2 Análise de modelos

A seguir serão mostrados alguns resultados obtidos pelo programa implementado para a suspensão de HDD, suas respectivas análises no software de elementos finitos ANSYS e uma comparação com as características do modelo da suspensão do HDD de 80Gb.

Inicialmente, foi feita a análise do resultado da peça com maior rigidez a flexão, obtida no item 6.2. A Figura 6.16 apresenta a peça criada no ANSYS e a Tabela 6.3 apresenta os resultados.

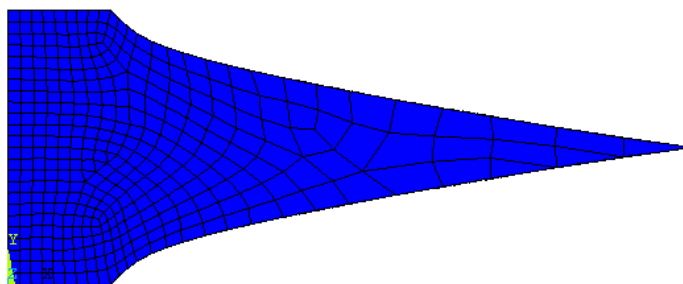


Figura 6.16 - Modelo de MEF para peça de maior rigidez a flexão.

Tabela 6.3 - Resultados para peça de maior rigidez a flexão.

Características	Valor	Variação % do modelo real
k_{rotx} (rigidez)	$4,2 \times 10^{-4}$ Nm/rad	-43,52%
k_z (rigidez)	27,7 N/m	11,69%
k_y (rigidez)	18,6 kN/m	-21,19%
1º flexão	823 Hz	50,46%
2º torção	7634 Hz	11,66%
3º lateral	33990 Hz	81,95%

A partir de agora, os ensaios serão numerados para simplificar as referências. Foram realizadas diversas otimizações com vários pesos e cargas diferentes, serão apresentados aqui somente as otimizações mais relevantes.

A Tabela 6.4 apresenta os parâmetros utilizados na otimização 1, a Figura 6.17 apresenta o resultado do programa e o modelo do ANSYS, e a Tabela 6.5 apresenta as características calculadas no ANSYS.

Tabela 6.4 - Dados para otimização 1.

Parâmetro	Valor	Parâmetro	Valor
w_m	10	F_z	0,1
w_r	1	F_y	10
w_v	30	M_x	0,1
malha	32x80	Raio do filtro	1

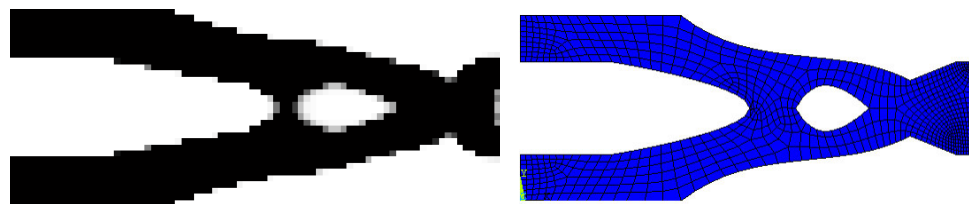


Figura 6.17 - Otimização 1 - A esquerda, resultado do programa. A direita, modelo de MEF.

Tabela 6.5 - Características do modelo da otimização 1.

Características	Valor	Variação % do modelo real
k_{rotx} (rigidez)	11×10^{-4} Nm/rad	46,80%
k_z (rigidez)	23,1 N/m	-6,77%
k_y (rigidez)	80,0 kN/m	238,98%
1º flexão	389 Hz	-28,88%
2º torção	3396 Hz	-50,33%
3º lateral	21393 Hz	14,52%

Verifica-se que as características obtidas ainda são muito aquém das desejadas, pois a peça não mostrou melhorias em relação à atual. Percebe-se que a ponta do modelo foi alargada, isso se deve a diminuição da rigidez à torção, pois o momento M_x gera um deslocamento elevado nos extremos da ponta alargada. Nota-se que ao diminuir raio do filtro, este alargamento na ponta da peça acaba desaparecendo. As otimizações 2 e 3 apresentam mais dificuldades com a diminuição da rigidez. A Tabela 6.6 apresenta os dados de entrada para as duas otimizações e a Figura 6.18 apresenta seus respectivos resultados.

Tabela 6.6 Dados para otimização 1.

Parâmetro	Otim. 1	Otim. 2	Parâmetro	Otim. 1	Otim. 2
w_m	20	10	F_z	0,1	0,1
w_r	1	1	F_y	10	10
w_v	30	20	M_x	1	0,1
malha	32x80	32x80	Raio do filtro	0,5	0,3

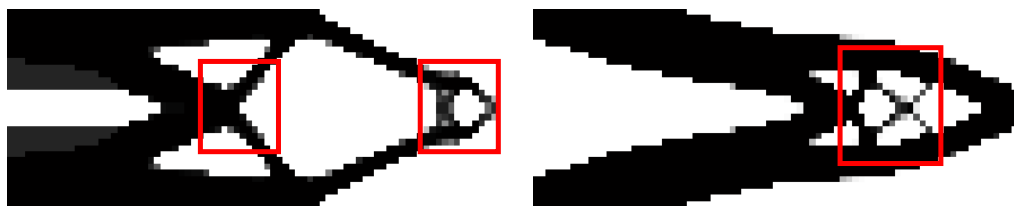


Figura 6.18 - Resultados das otimizações 2 e 3 respectivamente.

As regiões marcadas em vermelho da figura 6.20 apresentam pontos de difícil fabricação devido a dimensões muito reduzidas.

A otimização 2 apresenta claramente a convergência para a forma de treliça, significando que o critério de otimização de rigidez a efeitos de membrana se sobrepõe aos demais.

A partir da otimização 3, foram realizados testes aumentando o raio do filtro na tentativa de retirar os detalhes com dimensões reduzidas. Dessa forma, mantendo-se os parâmetros da otimização 3, alterando somente o filtro obtém-se os resultados da Figura 6.19.

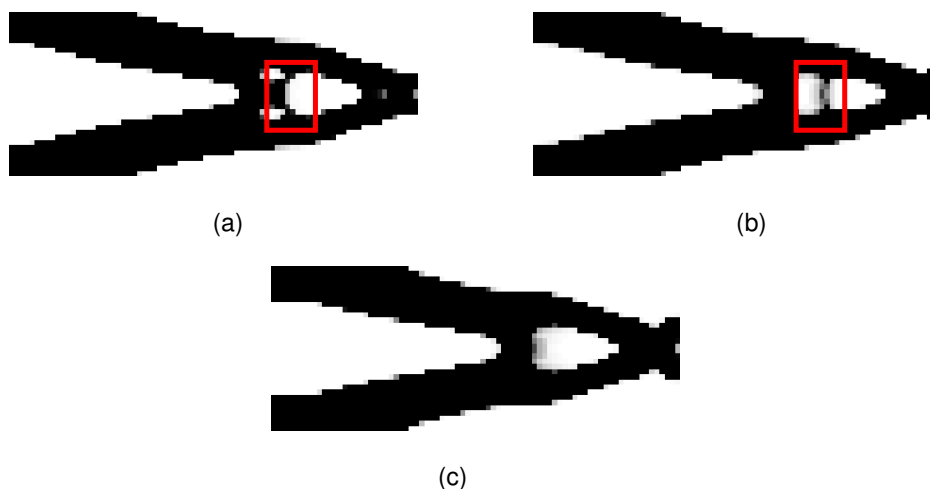


Figura 6.19 - Resultados da otimização 3 variando-se o filtro aplicado. (a) Otimização 3 - Raio do filtro = 0,5; (b) Otimização 3 - Raio do filtro = 0,6; (c) Otimização 3 - Raio do filtro = 0,7

Filtros com valor maior do que 0,7 convergem para o resultado da otimização 1.

Em seguida, foram realizadas mais otimizações com diminuição da rigidez (w_r maior que zero), contudo, os resultados obtidos não foram muito satisfatórios, apresentando novamente detalhes que complicam a fabricação da suspensão. Dessa forma, foram realizados testes aumentando a rigidez

ao invés de diminuí-la. A partir dos resultados obtidos com a variação dos parâmetros, percebeu-se que ao aumentar a influência na otimização dos efeitos de membrana, ocorre a diminuição da rigidez a flexão. Essa diminuição é decorrente da menor rigidez a flexão que a estrutura de treliça apresenta em relação à estrutura da Figura 6.16. Mantendo-se a otimização das frequências de ressonância, foi possível a obtenção de resultados muito interessantes.

A otimização 4 foi feita realizando um aumento da rigidez a flexão e a torção. A Tabela 6.7 apresenta os dados de entrada, a Figura 6.20 apresenta o resultado do programa e a modelagem no ANSYS. Os resultados das características podem ser vistos na Tabela 6.8.

Tabela 6.7 - Dados para otimização 4.

Parâmetro	Valor	Parâmetro	Valor
w_m	100	F_z	1
w_r	-1	F_y	10
w_v	10	M_x	2
malha	32x80	Raio do filtro	0,5

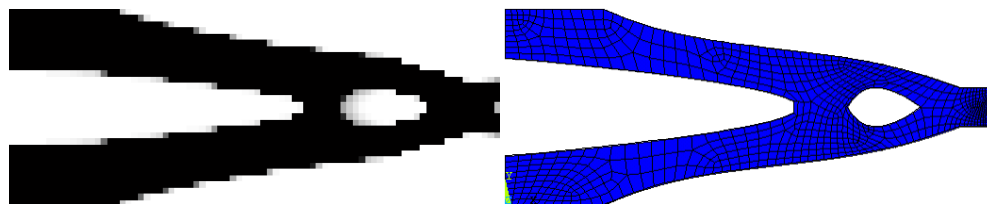


Figura 6.20 - Otimização 4 - A esquerda, resultado do programa. A direita, modelo de MEF.

Tabela 6.8 - Características do modelo da otimização 4.

Características	Valor	Variação % do modelo real
$k_{rotx}(\text{rigidez})$	$9,5 \times 10^{-4} \text{ Nm/rad}$	26,47%
$k_z(\text{rigidez})$	22,3 N/m	-9,97%
$k_y(\text{rigidez})$	71,9 kN/m	204,84%
1º flexão	457 Hz	-16,45%
2º torção	3792 Hz	-44,54%
3º lateral	22493 Hz	20,41%

As características obtidas na otimização 4 são insatisfatórias, comparadas ao modelo da suspensão atual, o ganho é muito inferior a perda de desempenho da peça.

Contudo, o resultado da otimização 4 é muito parecido com os resultados da otimização 3. Isso valida a hipótese de utilização de aumento da rigidez a flexão compensado pelo aumento da rigidez a efeitos de membrana. O critério de aumento da rigidez a flexão busca manter o formato da estrutura, enquanto os demais critérios buscam otimizar a estrutura com através de pequenas alterações. Dessa forma, as otimizações subseqüentes adotarão esta metodologia.

A seguir são apresentados os resultados da otimização 5. A Tabela 6.9 apresenta os parâmetros de entrada, a Figura 6.21 apresenta o resultado do programa e o modelo do ANSYS e a

Tabela 6.10 apresenta as características do modelo.

Tabela 6.9 - Dados para otimização 5.

Parâmetro	Valor	Parâmetro	Valor
w_m	40	F_z	1
w_r	-1	F_y	10
w_v	5	M_x	10
malha	32x80	Raio do filtro	0,5

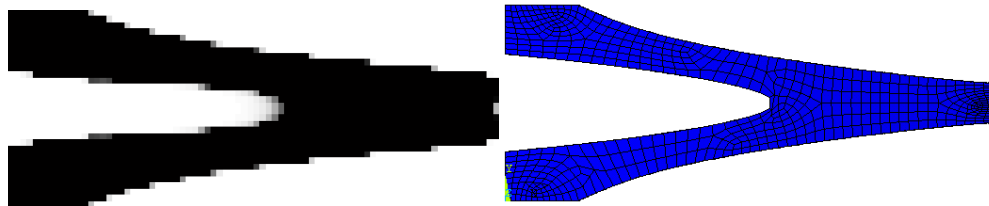


Figura 6.21 - Otimização 5 - A esquerda, resultado do programa. A direita, modelo de MEF.

Tabela 6.10 - Características do modelo da otimização 5.

Características	Valor	Variação % do modelo real
k_{rotx} (rigidez)	$7,8 \times 10^{-4}$ Nm/rad	3,80%
k_z (rigidez)	20,1 N/m	-18,95%
k_y (rigidez)	43,9 kN/m	86,02%
1º flexão	452 Hz	-17,37%
2º torção	4455 Hz	-34,84%
3º lateral	24433 Hz	30,79%

Os resultados da otimização 5 apresentam uma redução de quase 19% da rigidez a flexão, mas a diminuição das frequências de ressonância também é muito elevada. Percebe-se que houve uma predominância do critério de aumento de rigidez a torção. Isso pode ser verificado pela geometria obtida e pelo aumento do coeficiente de rigidez torcional.

A otimização 6 apresenta uma tentativa de correção dos problemas da otimização 5. Os dados de entrada estão dispostos na Tabela 6.11, os resultados do programa e do ANSYS estão na Figura 6.22 e as características do modelo estão na Tabela 6.12.

Tabela 6.11 - Dados para otimização 6.

Parâmetro	Valor	Parâmetro	Valor
w_m	40	F_z	1
w_r	-1	F_y	10
w_v	10	M_x	0,001
malha	32x80	Raio do filtro	0,3

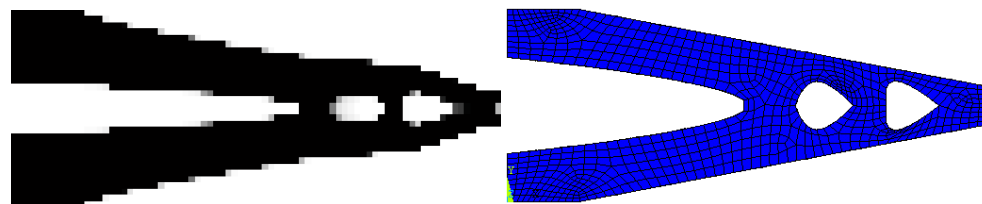


Figura 6.22 - Otimização 6 - A esquerda, resultado do programa. A direita, modelo de MEF.

Tabela 6.12 - Características do modelo da otimização 6.

Características	Valor	Variação % do modelo real
k_{rotx} (rigidez)	$8,4 \times 10^{-4}$ Nm/rad	11,87%
k_z (rigidez)	23,7 N/m	-4,44%
k_y (rigidez)	75,8 kN/m	221,19%
1º flexão	670 Hz	22,49%
2º torção	5139 Hz	-24,84%
3º lateral	35933 Hz	92,35%

Nota-se que houve o aumento da rigidez torcional, mas isso se deve também a interpretação do resultado e construção do modelo no ANSYS. O mais importante é notar que se conseguiu diminuir a rigidez a flexão e aumentar a frequência de ressonância do modelo.

A otimização 5 conseguiu a menor rigidez a flexão, enquanto a otimização 6 obteve uma relação de aumento da frequência mantendo a rigidez abaixo do modelo real. Com base nesses resultados, buscou-se mesclar os dois resultados em um novo modelo que é dado pela Figura 6.23. As características deste modelo estão apresentadas na Tabela 6.13.

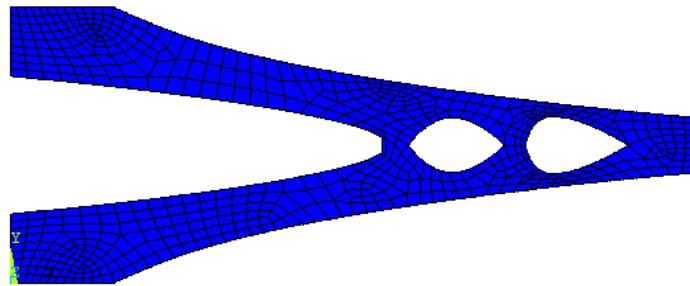


Figura 6.23 - Modelo final.

Tabela 6.13 - Características do modelo final.

Características	Valor	Variação % do modelo real
k_{rotx} (rigidez)	$6,2 \times 10^{-4}$ Nm/rad	-16,89%
k_z (rigidez)	18,8 N/m	-24,19%
k_y (rigidez)	39,1 kN/m	65,68%
1º flexão	720 Hz	31,63%
2º torção	6343 Hz	-7,23%
3º lateral	36981 Hz	97,96%

Os resultados deste modelo são muito superiores a todas as otimizações e ao modelo real. O único item em que o modelo obtido é inferior ao modelo final é a frequência de ressonância do primeiro modo de torção, mas este não impacta na solução, pois a diferença é pequena (~7%) enquanto as características mais importantes apresentaram melhoras muito significativas.

6.7 Ensaio do protótipo

Definido o modelo final, partiu-se para fabricação do protótipo que foi realizada na empresa Metalfoto localizada em Cotia-SP. Trata-se de uma empresa especializada em fotofabricação, eletroformação e usinagem a laser. [37]

O processo escolhido foi o de usinagem fotoquímica, o material utilizado foi o aço inoxidável 301, cujas propriedades são um pouco diferentes das adotadas e estão listadas na Tabela 6.14. As diferenças dos valores não alteram a escolha da solução, uma vez que o modelo escolhido é superior a peça atual comparando-se com o mesmo material. O desenho de fabricação da peça encontra-se no Anexo C.

Tabela 6.14 - Valores das propriedades do material do protótipo.

Propriedades	Valor
E (módulo de rigidez)	193 GPa
ρ (densidade)	8000 kg/m ³
ν (coeficiente de Poisson)	0,30

O teste para verificação da frequência de ressonância do primeiro modo foi realizado no laboratório de ultrassom do Departamento de Engenharia Mecatrônica da EPUSP. Neste teste, mediu-se a amplitude de oscilação do protótipo quando excitado por diversas frequências por meio de uma cerâmica piezelétrica colada a sua base. A configuração pode ser vista na Figura 6.24. A foto da montagem pode ser vista na Figura 6.25.

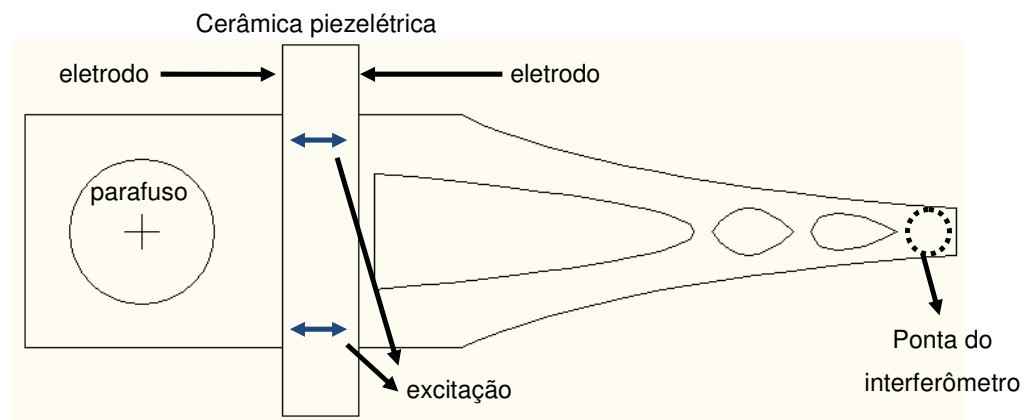


Figura 6.24 - Desenho esquemático da montagem da suspensão.

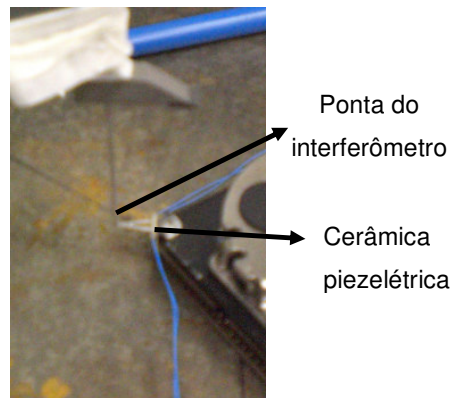


Figura 6.25 - Montagem para teste do protótipo

Os resultados das amplitudes para um intervalo de frequência de 0 a 700Hz pode ser visto na Figura 6.26.

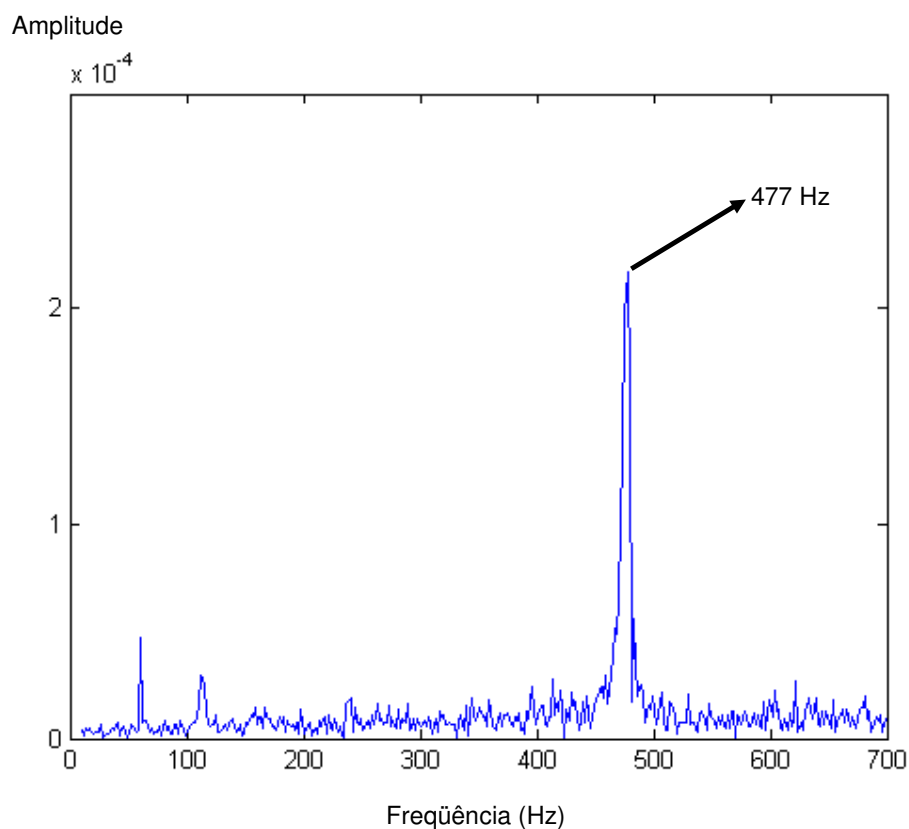


Figura 6.26 - Amostragem das frequências x amplitude de oscilação.

De acordo com os resultados, a primeira frequência de ressonância medida foi de 477 Hz, frequência bem abaixo da frequência calculada. Contudo, deve-se lembrar que as condições do experimento não refletem exatamente as condições de projeto e de funcionamento da suspensão do HDD. Deve-se atentar também ao fato da frequência de operação de um HDD de 10.000 rpm ser de 167 Hz aproximadamente, frequência bem abaixo do mínimo encontrado.

Realizou-se uma nova simulação com condições mais próximas das condições do teste. O resultado pode ser visto na Figura 6.27. A Tabela 6.15 apresenta uma comparação das características de projeto com as características do protótipo considerando as condições do ensaio.

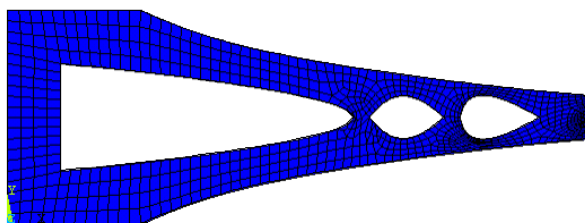


Figura 6.27 - Modelo aproximado do ensaio.

Tabela 6.15 - Comparação entre o projeto e o simulação das condições do ensaio do protótipo

Características	Projetado	Ensaiado
k_{rotx} (rigidez)	$6,2 \times 10^{-4}$ Nm/rad	$5,3 \times 10^{-4}$ Nm/rad
k_z (rigidez)	18,8 N/m	14,6 N/m
k_y (rigidez)	39,1 kN/m	33,1 kN/m
1º flexão	720 Hz	598 Hz
2º torção	6343 Hz	4791 Hz
3º lateral	36981 Hz	30100 Hz

O resultado para a frequência do primeiro modo foi de 598 Hz. Ainda está distante do modelo testado, mas isso se deve principalmente as condições em que foi realizado o teste, a influência do piezolétrico preso a suspensão e a fixação que não é exatamente igual a do HDD. Contudo, os resultados provam que o protótipo apresenta as características necessárias para operar em HDDs de alta rotação.

7 Conclusão

Os testes realizados com o programa de otimização topológica criado foram muito satisfatórios. Os principais problemas conhecidos da OT foram resolvidos, utilizando-se métodos consolidados por outros especialistas da área. O método criado para a regulação dos limites móveis apresentou resultados coerentes, sendo mais eficiente que outros métodos.

O modelo criado para a suspensão apresenta características de rigidez e ressonância teóricas superiores ao modelo atual, atingindo um aumento de cerca de 30% na frequência de ressonância do primeiro modo e uma redução da rigidez a flexão de 25%.

O protótipo fabricado apresenta uma frequência de ressonância menor do que a modelada, contudo isso se deve as condições de teste que não puderam simular fielmente as condições de um HDD.

Para trabalhos futuros, sugere-se a melhoria do programa com a utilização das técnicas de projeção para tratamento das pseudo-densidades, inserção de restrições de manufatura no projeto de OT, testes mais fidedignos ao funcionamento do HDD.

8 Referências

- [1] - Best, J., Iihara, Y., Simmons, R., Bolasna, S., Kotla, J., Tobari, A., Dorius, L., Matsumoto, T., Tsuchida, H., "The Femto Slider in Hitachi Hard Disk Drives", Hitachi Global Storage Technology, 2007, EUA.
- [2] - <http://talkelab.ucsd.edu/index/disk.html> Acesso em 29/04/2008
- [3] - Silva, E.C.N., Apostila de Mecânica Computacional – PMR-2420, Escola Politécnica da Universidade de São Paulo, São Paulo.
- [4] - Barkanov, E., "Introduction to the Finite Element Method", Institute of Materials and Structures – Riga Technical University, 2001, Riga.
- [5] - Bathe, K. J., "Finite Element Procedures", Prentice Hall, ed. 3, p165-470, EUA, 1996.
- [6] - Clough R. W. "The finite element method in plane stress analysis", Proc. American Society of Civil Engineers (2ª Conference on Electronic Computation, Pittsburg, Pennsylvania), Vol. 23, P. 345-378, 1960
- [7] - Macneal, R. H., "Finite Elements: Their Design and Performance", Marcel Dekker, 1994, EUA, p369-413
- [8] - Zhu, D., Wang, A., Jiang, T., "Topology design to improve HDD suspension dynamic characteristics", Journal of The International Society for Structural and Multidisciplinary Optimization, v31, p497-503, 2006
- [9] - Thompson, L. L., "On optimal stabilized MITC4 plate bending elements for accurate frequency response analysis", Computer and Structures, v81, p995-1008, 2003
- [10] - Yu, S. K., Liu B., "Optimal Suspension Design for Femto Sliders", IEEE Transactions on Magnetics, v39, p2423-2425, 2003
- [11] - Yoo, J., Lee, C. Y., "Topology optimization of a swing arm type actuator using the response surface method", Microsyst Technol, v13, p21-31, 2007
- [12] - Park, C., Yoo, J., Park, N. C., Park, Y. P., Shimano, T., Nakamura, S., "Design of swing arm type actuators considering thermal and dynamic characteristics", Microsyst Technol, v13, p777-796, 2007
- [13] - Bendsøe, M. P.; Sigmund, O., "Topology Optimization: Theory, Methods and applications", New York, EUA: Springer, 2003.
- [14] - Eschenauer, H.; Olhof, N., "Topology optimization of continuum structures", Applied Mechanic Review, v8, p42-51, 2001.

- [15] - Sigmund, O., Petersson, J., "Numerical instabilities in topology optimization: A survey on procedures dealing with checkerboards, mesh-dependencies and local minima", *Structural Optimization*, Springer Verlag, v16, p68-75, 1998.
- [16] - Bendsøe, M. P., Kikuchi, N., "Generating optimal topologies in structural design using a homogenization method", *Computer Methods In Applied Mechanics And Engineering*, Elsevier Science Sa Lausanne, v71, p197-224, 1988.
- [17] - Suzuki, K., Kikuchi, N., "A homogenization method for shape and topology optimization", *Computer Methods In Applied Mechanics And Engineering*, Elsevier Science Sa Lausanne, v93, p291-318, 1991.
- [18] - Ma, Z. D.; Kikuchi, N.; CHENG, H. C., "Topological design for vibrating Structures", *Computer Methods In Applied Mechanics And Engineering*, Elsevier Science Sa Lausanne, v121, p259-280, 1995.
- [19] - Bendøe, M. P., Sigmund, O., "Material interpolation schemes in topology optimization", *Archive Of Applied Mechanics*, Springer Verlag, v69, p635-654, 1999.
- [20] - Swan, C. C., Arora, J. S., "Topology design of material layout in structured composites of high stiffness and Strength", *Struct. Optim.*, v13 p45-59, 1997.
- [21] - Swan, C. C., Kosaka, I., "Voigt-Reuss topology optimization for structures with linear elastic material behaviours", *Int. J. Numer Methods Eng.*, v40, p3033-3057, 1997.
- [22] - Swan, C. C., Kosaka, I., "Voigt-Reuss topology optimization for structures with nonlinear material behaviors", *Int. J. Numer Methods Eng.*, v40, p3785-3814, 1997.
- [23] - Pedersen, N. L., "Maximization of eigenvalues using topology optimization", *Structural And Multidisciplinary Optimization*, Springer-Verlag, v20, p2-11, 2000.
- [24] - Stolpe, M.; Svanberg, K., "An alternative interpolation scheme for minimum compliance topology optimization", *Structural and Multidisciplinary Optimization*, v22, p116-124, 2001.
- [25] - Hansen, L. V., "Topology optimization of free vibrations of fiber laser packages", *Structural And Multidisciplinary Optimization*, Springer, v29, p341-348, 2005.
- [26] - Kiyono, C. Y., "Método de Otimização Topológica aplicado ao Projeto de Sonotrodos para Transdutores Piezelétricos", *Dissertação (Mestrado) – Escola Politécnica da Universidade de São Paulo*, 2008.

- [27] - Sigmund, O., "A 99 line topology optimization code written in Matlab", Structural And Multidisciplinary Optimization, Springer Verlag, v21, p120–127, 2001.
- [28] - Stolpe, M.; Svanberg, K., "An alternative interpolation scheme for minimum compliance topology optimization", Structural and Multidisciplinary Optimization, v21, p140-151, 2001.
- [29] - Matsui, K., Terada, K., "Continuous approximation of material distribution for topology optimization", International Journal For Numerical Methods In Engineering, John Wiley & Sons Ltd, v59, p1925–1944, 2004.
- [30] - Guest, J. K., Prevost, J. H.; Belytschko, T., "Achieving minimum length scale in topology optimization using nodal design variables and projection functions", International Journal For Numerical Methods In Engineering, v 61, p238-254, 2004.
- [31] - Kim, T. S., Kim, Y. Y., "Mac-based mode-tracking in structural topology optimization", Computers & Structures, v74, p375-383, 2000.
- [32] - Maeda, Y., Nishiwaki, S., Izui, K., Yoshimura, M., Matsui, K., Terada, K., "Structural topology optimization of vibrating structures with specified eigenfrequencies and eigenmode shapes", International Journal for Numerical Methods in Engineering, v67, p597-628, 2006.
- [33] - Silva, E.C.N., Apostila de otimização aplicada ao projeto de sistemas mecânicos – PMR-5215, Escola Politécnica da Universidade de São Paulo, São Paulo.
- [34] - Haftka, R. T., Gurdal, Z., Kamat, M. P., "Elements of Structural Optimization", Kluwer Academic Publishers, 1990.
- [35] - Ma, Z. D., Kikuchi, N., and Cheng, H.C., "Topological Design for Vibrating Structures", Computer Methods in Applied Mechanics and Engineering, V121, p259-280, 1995.
- [36] - Kalpakjian, S.; Schmid, S. R., "Manufacturing Engineering and Technology", New Jersey, EUA: Prentice Hall, 4ª edição, 2001.
- [37] - www.metalfoto.com.br – Acesso em 20/11/2008.

APÊNDICE A - Validação do MEF

O elemento implementado segue a formulação do MITC4 descrito na literatura [5]. Para validação do elemento, foram feitos testes simples, comparando os resultados com os obtidos pelo software ADINA. A implementação foi feita utilizando-se o software Matlab que simplifica o tratamento de matrizes.

Análise estática

A primeira validação envolve a análise estática de uma placa submetida a uma carga constante.

As propriedades do material utilizado nos testes estão presentes na Tabela 6.1, propriedades do aço inoxidável.

Inicialmente, considerou-se um elemento quadrilátero de dimensões 2x2m. Os testes foram realizados com três espessuras diferentes: 0,05m, 0,01m e 0,001m. Foi aplicada uma carga unitária ($|F| = 1N$) de acordo com a Figura A.1.

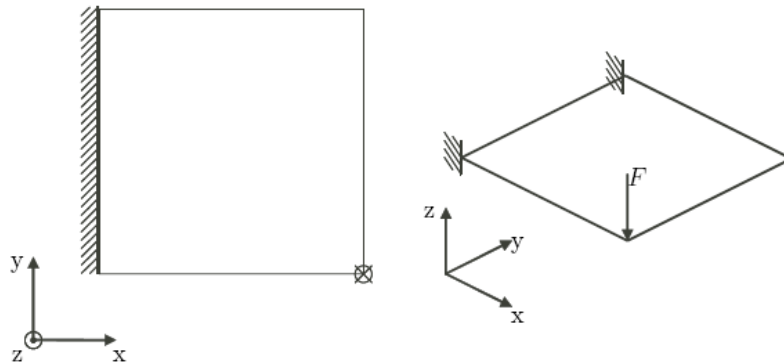


Figura A.1 - Configuração para teste de elemento

A Tabela A.1 apresenta os resultados dos deslocamentos w , θ_x e θ_y . Verifica-se que os valores estão muito próximos dos valores obtidos pelo software ADINA.

Tabela A.1 - Resultados para testes com elemento 2x2

	Adina	Implementado
h = 0,05 m		
w	$-6,68956 \times 10^{-7}$	$-6,68946 \times 10^{-7}$
θ_x	$1,33343 \times 10^{-7}$	$-1,32680 \times 10^{-7}$
θ_y	$6,68246 \times 10^{-7}$	$-6,68649 \times 10^{-7}$

h = 0,01 m		
w	$-8,35521 \times 10^{-5}$	$-8,35521 \times 10^{-5}$
θ_x	$1,66387 \times 10^{-5}$	$1,66354 \times 10^{-5}$
θ_y	$8,35485 \times 10^{-5}$	$8,35505 \times 10^{-5}$
h = 0,001 m		
w	$-8,35493 \times 10^{-2}$	$-8,35493 \times 10^{-2}$
θ_x	$1,66375 \times 10^{-2}$	$1,66375 \times 10^{-2}$
θ_y	$8,35493 \times 10^{-2}$	$8,35493 \times 10^{-2}$

Além da configuração do elemento quadrilátero regular, testou-se também a configuração do elemento distorcido em relação à proporção dos lados. A Figura A.2 esquematiza o elemento. O carregamento é o mesmo do teste anterior. A Tabela A.2 apresenta as espessuras utilizadas e os resultados obtidos.

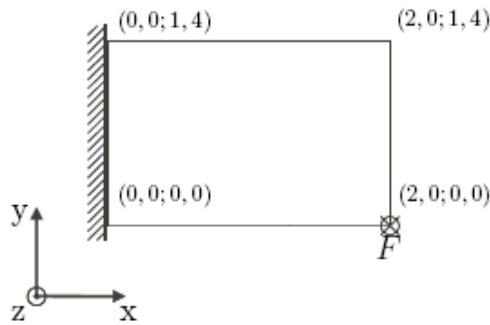


Figura A.2 - Elemento quadrilátero com a proporção dos lados distorcida.

Tabela A.2 - Resultados para testes com elemento 2x1,4

	Adina	Implementado
	h = 0,1 m	
w	$-1,01462 \times 10^{-7}$	$-1,01413 \times 10^{-7}$
θ_x	$1,91921 \times 10^{-8}$	$1,87405 \times 10^{-8}$
θ_y	$1,00976 \times 10^{-7}$	$1,01201 \times 10^{-7}$
	h = 0,01 m	
w	$-1,01097 \times 10^{-4}$	$-1,01096 \times 10^{-4}$
θ_x	$1,89593 \times 10^{-5}$	$1,89548 \times 10^{-5}$
θ_y	$1,01092 \times 10^{-4}$	$1,01094 \times 10^{-4}$
	h = 0,001 m	
w	$-1,01093 \times 10^{-1}$	$-1,01093 \times 10^{-1}$
θ_x	$1,89570 \times 10^{-2}$	$1,89569 \times 10^{-2}$
θ_y	$1,01093 \times 10^{-1}$	$1,01093 \times 10^{-1}$

Em ambos os casos, verifica-se que os deslocamentos encontrados são muito próximos aos do software ADINA, o que valida a implementação do elemento.

Análise modal

Além dos testes de deformação, foram feitos testes para se avaliar as frequências de ressonância. A configuração está esquematizada na Figura A.3. Os deslocamentos u_i são utilizados para identificar o modo de vibrar, utilizando-os no cálculo do MAC. Foi utilizada uma estrutura composta de 2 elementos de dimensão 1x1. A estrutura final tem 2m de comprimento e 1m de largura.

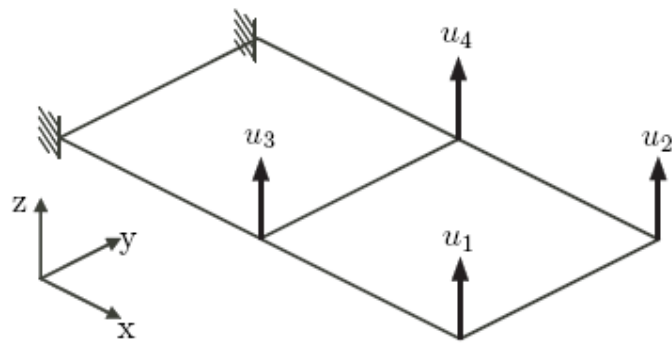
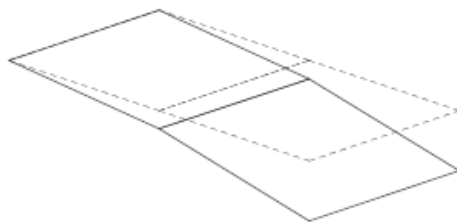
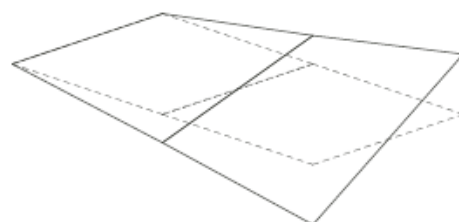


Figura A.3 - Estrutura utilizada para testes de ressonância.

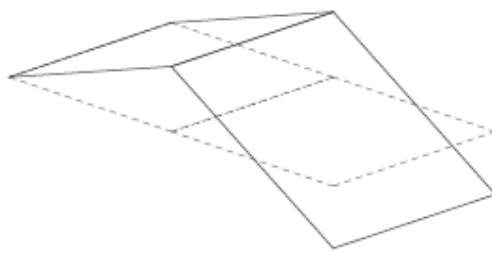
Foram avaliados os três primeiros modos de vibrar que estão esquematizados na Figura A.4.



(a)



(b)



(c)

Figura A.4 - Modos de vibrar da estrutura de teste com dois elementos. (a) 1º modo de vibrar; (b) 2º modo de vibrar; (c) 3º modo de vibrar.

Os resultados obtidos pelo software ADINA e o código implementado podem ser vistos nas Tabela A.3 e Tabela A.4. Foram testadas estruturas com duas espessuras diferentes (0,01 m e 0,001 m).

Nota-se que os valores das frequências ficaram muito próximas das obtidas pelo ADINA. Verifica-se também que a razão entre os deslocamentos do ADINA e os do código implementado é a mesma, o que garante que os modos comparados são os mesmos. O número de MAC foi calculado para mostrar que os modos encontrados pelo software ADINA e o código implementado são exatamente os mesmos.

Tabela A.3 - Resultados para testes com espessura $h = 0,01$ m

	Adina	Implementado
1º modo		
$\omega_n(\text{rad/s})$	13,7668	13,7666
u_1	0,160671	-1,00000
u_2	0,160671	-0,99998
u_3	0,04892	-0,304479
u_4	0,04892	-0,304460
MAC	1,00000	
2º modo		
$\omega_n(\text{rad/s})$	63,8146	63,8207
u_1	0,246220	-0,499991
u_2	-0,246220	0,500041
u_3	0,114975	-0,233472
u_4	-0,114975	0,233429
MAC	1,00000	
3º modo		
$\omega_n(\text{rad/s})$	153,524	153,480
u_1	-0,134675	0,162659

u_2	-0,134675	0,162438
u_3	0,139937	-0,168962
u_4	0,139937	-0,168834
MAC	1,00000	

Tabela A.4 - Resultados para testes com espessura $h = 0,001$ m

	Adina	Implementado
1º modo		
$\omega_n(\text{rad/s})$	1,37670	1,37670
u_1	0,508092	-1,000000
u_2	0,508092	-1,000000
u_3	0,154693	-0,304458
u_4	0,154693	-0,304458
MAC	1,00000	
2º modo		
$\omega_n(\text{rad/s})$	6,38272	6,38272
u_1	0,778717	-0,500000
u_2	-0,778717	0,500000
u_3	0,363552	-0,233430
u_4	-0,363552	0,233429
MAC	1,00000	
3º modo		
$\omega_n(\text{rad/s})$	15,3573	15,3572
u_1	-0,425914	0,162439
u_2	-0,425914	0,162437
u_3	0,442550	-0,168783
u_4	0,442550	-0,168782
MAC	1,00000	

APÊNDICE B - Código de Matlab do programa

```
function fim = HDDSuspension(nelx,nely,f0,wv1,wr1,wm1,Resol1)

% nelx = 80;           % Number of elements in x
% nely = 32;           % Number of elements in y
% Resol = 1;           % Minimum resolution achievable by lithography process
% f0 = 40;             % Minimum desired eigenvalue
width = 4;            % Width of the working area
leng = 10;            % Length of the working area
h = 0.06;             % Thickness
volp_max = 0.5;       % Maximum percentage of material
volp_min = 0.40;      % Minimum percentage of material
nm = 1;               % Number of desired modes

% Material model variables
p = 1;                % Mass - SIMP
q = 3;                % Stiffness - RAMP
E = 210e3;            % Young's modulus
poisson = 0.3;        % Poisson's ratio
ro = 7800e-6;         % Material density

el_wid = width/nely;  % Element width
el_len = leng/nelx;   % Element length

% Element matrices calculation
[Kem, Ke, Me] = ElementMatrices(el_wid, el_len, h, E, poisson, ro);

% condição inicial para as pseudo-densidades
dn = volp_max*ones(nely, nelx);
dn_old = zeros(nely, nelx);
change = 1;
change1 = 1;
change2 = 1;
loop = 1;

% Weight coefficients
wv = 0;               % ressonancia
wr = -1;              % MITC4
wm = 0;               % Membrana

% Load definitions
F = sparse(3*(nelx+1)*(nely+1),1);
Fm = sparse(2*(nelx+1)*(nely+1),1);
Fm(2*(nelx+1)*(nely+1)-(nely)) = 10;
% F(floor(3*(nelx+1)*(nely+1)-(nely+1)*1.5),1) = 1;
F(floor(3*(nelx+1)*(nely+1)-(nely+1)*1.5),1) = 1;

% Boundary conditions
fixeddofm = 1:(nely+1)*2;
fixeddof = 1:(nely+1)*3;
alldofm = 1:2*(nelx+1)*(nely+1);
alldof = 1:3*(nelx+1)*(nely+1);
freedofm = setdiff(alldofm,fixeddofm);

% Optimization restrictions
options = optimset('Simplex', 'on');
A(1,:) = ones(1,(nely*nelx))/(nely*nelx);
```

```

A(2,:) = -ones(1,(nely*nelx))/(nely*nelx);
b(1) = volp_max;
b(2) = -volp_min;

while change > 0.001 && change1 > 0.003 && change2 > 0.01

    % Sensitivity calculation
    [sens, sens2] = Sensitivityv6(Kem,Ke,Me,Fm,F,fixeddof,alldof,...
        freedofm,fixeddofm,dn,p,q,nm,f0,wv,wr,wm);
    sens = Filter2(sens,Resol, el_len, el_wid);
    sens2 = Filter2(sens2,Resol, el_len, el_wid);

    % Matrix to vector transformation
    sens_vec = Matrix2Vector(sens);
    sens2_vec = Matrix2Vector(sens2);
    dn_vec = Matrix2Vector(dn);

    % PLS restrictions
    lb = max(dn_vec.*(sens2_vec/max(sens2_vec)*0.14+0.85),0.0001);
    ub = min(dn_vec.*(1.15-sens2_vec/max(sens2_vec)*0.14),1);
    dn_vec = linprog(sens_vec, A, b, [], [], lb, ub, dn, options);

    % Vector to matrix transformation
    dn = Vector2Matrix(dn_vec, nelx);

    % Symmetry correction
    dn(floor(nely/2+1.5):nely,:) = flipud(dn(1:floor(nely/2),:));

    % Greatest variation of a single variable
    change = max(max(abs(dn - dn_old)));

    % Percentage of variation in relation to the total volume
    change1 = sum(sum(abs(dn - dn_old)))/sum(sum(dn));

    % Percentage of changing elements
    change2 = length(find((abs(dn - dn_old))>0.005))/(nelx*nely);

    % Print results
    disp([' It.: ' sprintf('%4i',loop)...
        ' Vol.: ' sprintf('%6.3f',sum(sum(dn))/(nelx*nely)) ...
        ' ch1.: ' sprintf('%6.3f',change )...
        ' ch2.: ' sprintf('%6.3f',change1 )...
        ' ch3.: ' sprintf('%6.3f',change2 )]);

    % Plot densities
    colormap(gray); imagesc(-dn); axis equal; axis tight; axis off; grid
on;
    pause(1e-6);
    dn_old = dn;
    loop = loop+1;

    p = 1 + floor(loop/60);
    if q < 6
        q = 3 + floor(loop/25);
    else
        q = 6;
    end

    if p < 3

```

```

        p = 1 + floor(loop/25);
    else
        p = 3;
    end

    if loop == 20
        wv = wv1; % ressonancia
        wr = wr1; % MITC4
        wm = wml;
        Resol = Resol1;
        F(floor(3*(nelx+1)*(nely+1)-(nely+1)*1.5)+1,1) = 0.1;
    end
end

fim = 1;

% Sentivity calculation
function [sens,sens2]=Sensitivity(Kem,Ke,Me,Fm,F,fixeddof,alldof,...
                                freedofm,fixeddofm,dn,p,q,nm,f0,wv,wr,wm)

nely = length(dn(:,1));
nelx = length(dn(1,:));

mode = sparse(1,nm);
mode2 = sparse(1,nm);
sens = sparse(nely,nelx);
sens2 = sparse(nely,nelx);

% Coefficient for in-plane stiffness matrix
r = 3;

% Options for eigs function
opts.disp = 0;
opts.warning = 1;
opts.issym = 1;
opts.cholB = 1;
opts.isreal = 1;
opts.v0 = ones(3*(nelx+1)*(nely+1),1);

freedof = setdiff(alldof,fixeddof);

% Calculating the global matrices
[Km,K,M] = GlobalMatrices(nelx, nely, Kem, Ke, Me, dn, p, q);

% Displaments calculation
u(fixeddof,:) = 0;
um(fixeddofm,:) = 0;
u(freedof,:) = K(freedof,freedof)\F(freedof,1);
um(freedofm,:) = Km(freedofm,freedofm)\Fm(freedofm,:);

% Erasing the elements correspondig to the fixed dof
K(fixeddof,:) = sparse(length(K(fixeddof,1)),length(K(1,:)));
M(fixeddof,:) = sparse(length(M(fixeddof,1)),length(M(1,:)));
K(:,fixeddof) = sparse(length(K(1,:)),length(K(fixeddof,1)));
M(:,fixeddof) = sparse(length(M(1,:)),length(M(fixeddof,1)));

% Increasing the eigenvalues of the fixed dof because they aren't important

```

```

K(fixeddof,fixeddof) = speye(length(fixeddof))*10000000000;
M(fixeddof,fixeddof) = speye(length(fixeddof));

M = chol(M);

% Eigenvalue and eigenvector calculation
[vect, freq] = eigs(K,M,nm,'sm',opts);
% [vect, freq] = eig(full(K),full(M),'chol');

% Transforming the diagonal eigenvalue matrix in a vector
freq = diag(freq);

% Rescaling the eigenvectors such that u'*M*u = 1
for i = 1:nm
    valuei = vect(:,i)'*M*vect(:,i);
    vect(:,i) = vect(:,i)/sqrt(valuei);
end

% Sensitivity analysis of all elements
for elx = 1:nelx
    for ely = 1:nely
        n1 = (nely+1)*(elx-1)+ely;
        n2 = (nely+1)* elx +ely;
        edof=[3*n2-2; 3*n2-1; 3*n2; 3*n1-2; 3*n1-1; 3*n1;...
            3*n1+1; 3*n1+2; 3*n1+3; 3*n2+1; 3*n2+2; 3*n2+3];
        edofm=[2*n2-1; 2*n2; 2*n1-1; 2*n1; 2*n1+1; 2*n1+2; 2*n2+1; 2*n2+2];

%         Stiffnes sensitivity calculation
        dfs=wr*(-(1+q)/(1+q*(1-dn(ely,elx)))^2)*u(edof)'*Ke*u(edof);
        dfm = -wm*((dn(ely,elx)^(r-1))*r)*um(edofm)'*Kem*um(edofm);

%         Eigenvalue sensitivity
        for i = 1:nm
%             Considering only the eigenvalues smaller than the minimun
            if (freq(i)-f0) < 0 && freq(i)>0
                mode(i)=(vect(edof,i)'*((1+q)/(1+q*(1-dn(ely,elx)))^2)*...
                    Ke-freq(i)*((dn(ely,elx)^(p-1))*p)*Me)*vect(edof,i));
            else
                mode(i) = 1;
            end
        end
        w = wv*2*freq(1:nm)/f0;
        dfv = (mode*w);

%         Second derivative of stiffness objective function
        d2fs = wr*(-((1+q)*q^2*(1+q*(1-dn(ely,elx))))/...
            (1+q*(1-dn(ely,elx)))^4)*u(edof)'*Ke*u(edof);
        d2fm = -wm*((dn(ely,elx)^(r-2))*r*(r-1))*um(edofm)'*Kem*um(edofm);

%         Second derivative of eigenvalue objective function
        for i = 1:nm
            if (freq(i)-f0) < 0 && freq(i)>0
                mode2(i)=vect(edof,i)'*((1+q)*q^2*(1+q*(1-dn(ely,elx)))/...
                    (1+q*(1-dn(ely,elx)))^4)*Ke-...
                    mode(i)*((dn(ely,elx)^(p-1))*p)*Me-...
                    freq(i)*((dn(ely,elx)^(p-2))*p*(p-1))*Me-...
                    mode(i)*(dn(ely,elx)^(p-1))*p*Me)*vect(edof,i);
            else
                mode2(i)=0;
            end
        end
    end
end

```

```

end
d2fv = (mode2*w)+ 2*vw*sum(mode.^2)/f0^2;

% Multi-objective sensitivity calculation
sens(ely, elx) = dfm - dfs - dfv;

% Second derivative calculation(only the absolute value is important)
sens2(ely, elx) = abs(d2fm - d2fs - d2fv);
end
end

function [Km, K, M] = GlobalMatrices(nelx, nely, KEm, KE, ME, dn, p, q)

K = sparse(3*(nelx+1)*(nely+1),3*(nelx+1)*(nely+1));
Km = sparse(2*(nelx+1)*(nely+1),2*(nelx+1)*(nely+1));
M = sparse(3*(nelx+1)*(nely+1),3*(nelx+1)*(nely+1));

for elx = 1:nelx
    for ely = 1:nely
        n1 = (nely+1)*(elx-1)+ely;
        n2 = (nely+1)* elx +ely;
        edof = [3*n2-2; 3*n2-1; 3*n2; 3*n1-2; 3*n1-1; 3*n1; 3*n1+1; 3*n1+2;
3*n1+3; 3*n2+1; 3*n2+2; 3*n2+3];
        edofm = [2*n2-1; 2*n2; 2*n1-1; 2*n1; 2*n1+1; 2*n1+2; 2*n2+1;
2*n2+2];
        % RAMP Model
        K(edof,edof) = K(edof,edof) + (dn(ely,elx)/(1+q*(1-
dn(ely,elx)))).*KE;

        % SIMP Model
        K(edof,edof) = K(edof,edof) + (dn(ely,elx)^p)*KE;
        Km(edofm,edofm) = Km(edofm,edofm) + (dn(ely,elx)^3)*KEm;
        M(edof,edof) = M(edof,edof) + (dn(ely,elx)^p)*ME;

    end
end

% Due to numeric errors, there are some spurious diferences that remove the
% matrix simmetry. This operation will return simmetry to the matrix. Here
% we define that the lower triangular matrix is equal to the upper
% triangular matrix transposed
% This procedure enhance the accuracy of the eigenvalue calculation,
% avoiding complex results
Km = triu(Km)+triu(Km,1)';
K = triu(K)+triu(K,1)';
M = triu(M)+triu(M,1)';
function [Km, K, M] = ElementMatrices (larg_elem,comp_elem,h,E,poisson,ro)

% Coordinates of the element nodes
x_elem = [comp_elem/2; -comp_elem/2; -comp_elem/2; comp_elem/2 ];
y_elem = [larg_elem/2; larg_elem/2; -larg_elem/2; -larg_elem/2 ];

% Coordinates of the intermediate of element nodes
x_abcd = [(x_elem(1)+x_elem(2))/2; (x_elem(2)+x_elem(3))/2; ...
(x_elem(3)+x_elem(4))/2; (x_elem(4)+x_elem(1))/2];
y_abcd = [(y_elem(1)+y_elem(2))/2; (y_elem(2)+y_elem(3))/2; ...
(y_elem(3)+y_elem(4))/2; (y_elem(4)+y_elem(1))/2];

% Matrices initial declaration

```

```

Km = sparse(8,8);
K = sparse(12,12);
M = sparse(12,12);

% Constant to account for the nonuniformity of the shearing stresses
k = 5/6;

% Plane stress matrix for bending
Cb = E*(h^3)/(12*(1-poisson^2))*[1      poisson  0
                                poisson  1      0
                                0      0      ((1-poisson)/2)];

% Plane stress matrix for in-plane strain
Cm = E/(1-poisson^2)*[1      poisson  0
                     poisson  1      0
                     0      0      ((1-poisson)/2)];

% Plane stress matrix for transverse shear strain
Cs = k*E*h/(2*(1+poisson))*[1  0
                             0  1];

% Matrix for mass matrix calculation
A = [h 0      0
     0 h^3/12 0
     0 0      h^3/12]*ro;

% Shear strain auxiliary calculation
% Passing through points A, B, C and D
for nno = 1:4
    % 2x2 Gauss integration
    if nno == 1
        r = 0;
        s = 1;
    else if nno == 2
        r = -1;
        s = 0;
    else if nno == 3
        r = 0;
        s = -1;
    else
        r = 1;
        s = 0;
    end
end

% dH is the matrix which contains dH/dr(column1) and dH/ds(column2)
dH(1,1) = (1+s)/4;
dH(2,1) = -(1+s)/4;
dH(3,1) = -(1-s)/4;
dH(4,1) = (1-s)/4;
dH(1,2) = (1+r)/4;
dH(2,2) = (1-r)/4;
dH(3,2) = -(1-r)/4;
dH(4,2) = -(1+r)/4;

% Jacobian matrix
dxdr = dH(:,1)'\*x_elem;
dxds = dH(:,2)'\*x_elem;
dydr = dH(:,1)'\*y_elem;

```

```

dyds = dH(:,2)'\*y_elem;
J = [dxdr dydr
      dxds dyds];
IJ = inv(J);

% dH/dx and dH/dy calculation
dHdx(1) = IJ(1,:)*dH(1,:)\';
dHdx(2) = IJ(1,:)*dH(2,:)\';
dHdx(3) = IJ(1,:)*dH(3,:)\';
dHdx(4) = IJ(1,:)*dH(4,:)\';
dHdy(1) = IJ(2,:)*dH(1,:)\';
dHdy(2) = IJ(2,:)*dH(2,:)\';
dHdy(3) = IJ(2,:)*dH(3,:)\';
dHdy(4) = IJ(2,:)*dH(4,:)\';

beta = (1/4)*[(1+r)*(1+s) (1-r)*(1+s) (1-r)*(1-s) (1+r)*(1-s)];

if nno == 1 || nno == 3
    R(nno, :) = [dHdx(1) 0 beta(1) dHdx(2) 0 beta(2) ...
                  dHdx(3) 0 beta(3) dHdx(4) 0 beta(4)];
else
    R(nno, :) = [dHdy(1) -beta(1) 0 dHdy(2) -beta(2) 0 ...
                  dHdy(3) -beta(3) 0 dHdy(4) -beta(4) 0];
end
end

% Bs calculation auxiliary matrices
aux2 = [1 y_abcd(1)
        1 y_abcd(3)];
aux3 = [1 x_abcd(2)
        1 x_abcd(4)];
Rac = [R(1,:); R(3,:)];
Rbd = [R(2,:); R(4,:)];
aux2 = inv(aux2)*Rac;
aux3 = inv(aux3)*Rbd;

% Integrating in each node of the element
for nno = 1:4
    % 2x2 Gauss integration
    if nno == 1
        r = sqrt(3)/3;
        s = sqrt(3)/3;
    else if nno == 2
        r = -sqrt(3)/3;
        s = sqrt(3)/3;
    else if nno == 3
        r = -sqrt(3)/3;
        s = -sqrt(3)/3;
    else
        r = sqrt(3)/3;
        s = -sqrt(3)/3;
    end
end
end

% dH is the matrix which contains dH/dr(column1) and dH/ds(column2)
dH(1,1) = (1+s)/4;
dH(2,1) = -(1+s)/4;
dH(3,1) = -(1-s)/4;
dH(4,1) = (1-s)/4;

```

```

dH(1,2) = (1+r)/4;
dH(2,2) = (1-r)/4;
dH(3,2) = -(1-r)/4;
dH(4,2) = -(1+r)/4;

% Jacobian matrix
dxdr = dH(:,1)'*x_elem;
dxds = dH(:,2)'*x_elem;
dydr = dH(:,1)'*y_elem;
dyds = dH(:,2)'*y_elem;
J = [dxdr dydr
     dxds dyds];
IJ = inv(J);

%Cálculo dos dH/dx e dH/dy
dHdx(1) = IJ(1,:)*dH(1,:)' ;
dHdx(2) = IJ(1,:)*dH(2,:)' ;
dHdx(3) = IJ(1,:)*dH(3,:)' ;
dHdx(4) = IJ(1,:)*dH(4,:)' ;
dHdy(1) = IJ(2,:)*dH(1,:)' ;
dHdy(2) = IJ(2,:)*dH(2,:)' ;
dHdy(3) = IJ(2,:)*dH(3,:)' ;
dHdy(4) = IJ(2,:)*dH(4,:)' ;

% Stress-strain matrix for in-plane strain
Bm = [dHdx(1) 0          dHdx(2) 0          dHdx(3) 0          dHdx(4) 0
      0        dHdy(1) 0          dHdy(2) 0          dHdy(3) 0          dHdy(4)
      dHdy(1) dHdx(1) dHdy(2) dHdx(2) dHdy(3) dHdx(3) dHdy(4) dHdx(4)];

% Stress-strain matrix for bending
Bb = [0 0    -dHdx(1) 0 0    -dHdx(2) 0 0    -dHdx(3) 0 0    -dHdx(4)
      0 dHdy(1) 0 0 dHdy(2) 0 0 dHdy(3) 0 0 dHdy(4) 0
      0 dHdx(1) -dHdy(1) 0 dHdx(2) -dHdy(2) 0 dHdx(3) ...
      -dHdy(3) 0 dHdx(4) -dHdy(4)];

% Stress-strain matrix for transverse shear strain
Bs = [1 s 0 0; 0 0 1 r]*[aux2; aux3];

% Interpolation functions for mass matrix calculation
h1 = (1+r)*(1+s)/4;
h2 = (1-r)*(1+s)/4;
h3 = (1-r)*(1-s)/4;
h4 = (1+r)*(1-s)/4;

% Interpolation functions matrix
N = [h1 0 0 h2 0 0 h3 0 0 h4 0 0
     0 0 -h1 0 0 -h2 0 0 -h3 0 0 -h4
     0 h1 0 0 h2 0 0 h3 0 0 h4 0];

% Element stiffness and mass matrices
Km = h*Bm'*Cm*Bm*det(J)+Km;
K = Bb'*Cb*Bb*det(J)+Bs'*Cs*Bs*det(J)+K;
M = N'*A*N*det(J) + M;
end

% Due to numeric errors, there are some spurious differences that remove the
% matrix symmetry. This operation will return symmetry to the matrix. Here
% we define that the lower triangular matrix is equal to the upper
% triangular matrix transposed
% This procedure enhance the accuracy of the eigenvalue calculation,

```

```

% avoiding complex results
Km = triu(Km)+triu(Km,1)';
K = triu(K)+triu(K,1)';
M = triu(M)+triu(M,1)';

% Matrix to vector conversion
function [vetor] = Matrix2Vector(Matrix)
ncol = length(Matrix(1,:));
nrow = length(Matrix(:,1));
ini = 1;
for col = 1:ncol
    vetor(ini:col*nrow) = Matrix(:,col);
    ini = col*nrow + 1;
end

% Vector to matrix conversion
function Matrix = Vector2Matrix(vect, ncols)
nrows = length(vect)/ncols;
ini = 1;
for col = 1:ncols
    Matrix(:,col) = vect(ini:col*nrows);
    ini = col*nrows + 1;
end

% Filtro para remoção de dependencia de malha e instabilidade tabuleiro
function [sens_new] = Filter(sens_old, resol, eldx, eldy)
nely = length(sens_old(:,1)); % Number of elements in y
nelx = length(sens_old(1,:)); % Number of elements in x
sens_new = zeros(nely, nelx);
adj_x = floor(resol/eldx);
adj_y = floor(resol/eldy);

for elx = 1:nelx
    for ely = 1:nely
        wttotal=0.0;
        for viz_x = max(1,ely-adj_y):min(nely,ely+adj_y)
            for viz_y = max(1,elx-adj_x):min(nelx,elx+adj_x)
                w = max(0, (resol-(((elx-viz_y)*eldx)^2+...
                    ((ely-viz_x)*eldy)^2)^0.5))/resol);
                wttotal = wttotal+w;
                sens_new(ely,elx) = sens_new(ely,elx)+w*sens_old(viz_x,viz_y);
            end
        end
        sens_new(ely,elx) = sens_new(ely,elx)/wttotal;
    end
end
end

```

APÊNDICE C - Desenho de fabricação