

1



## Otimização Topológica – Aula 3

Prof. Dr. Emílio Carlos Nelli Silva  
 Dr. Renato Picelli  
 Escola Politécnica da USP  
 Departamento de Engenharia Mecatrônica e Sistemas  
 Mecânicos



Emilio Silva e Renato Picelli

2

## Otimização Usando Métodos Numéricos

**Algoritmos de Otimização para Otimização Topológica**

- Programação Linear Seqüencial (PLS)
- Programação Quadrática Seqüencial (PQS)
- MMA – Method of Moving Asymptotes
- Convex Linearization method (CONLIN)

→ genéricos

**Métodos Probabilísticos (Não funcionam para OT)**

- Algoritmos Genéticos
- “Simulated Annealing”

Emilio Silva e Renato Picelli

1

## Programação Quadrática Seqüencial (PQS) 3

Seja o problema de otimização:

$$\begin{aligned} & \text{Min } f(\mathbf{x}) \\ & \mathbf{x} \\ & \text{tal que } g_j(\mathbf{x}) \geq 0, \quad j=1, \dots, n_g \end{aligned}$$

A cada iteração a direção de movimento é obtida resolvendo o subproblema de programação quadrática (PQ):

$$\begin{aligned} & \text{Min } \phi(\mathbf{s}) = f(\mathbf{x}^{(i)}) + \mathbf{s}^T \nabla f(\mathbf{x}^{(i)}) + (1/2) \mathbf{s}^T \mathbf{A}(\mathbf{x}^{(i)}, \lambda^{(i)}) \mathbf{s} \\ & \mathbf{s} \\ & \text{tal que } g_j(\mathbf{x}^{(i)}) + \mathbf{s}^T \nabla g_j(\mathbf{x}^{(i)}) \geq 0, \quad j=1, \dots, n_g \end{aligned}$$

Aproximação positiva-definida da Matriz Hessiana da função Lagrangeana

Cuja solução fornece  $\mathbf{s}$  e  $\lambda_{i+1} \Rightarrow \mathbf{x}^{(i+1)} = \mathbf{x}^{(i)} + \alpha \mathbf{s}$  onde  $\alpha$  é obtido minimizando:

$$\psi(\alpha) = f(\mathbf{x}) + \sum_{j=1}^{n_g} \mu_j \left| \min(0, g_j(\mathbf{x})) \right| \quad \text{onde: } \mu_j^{(i)} = \max \left[ \left| \lambda_j^{(i)} \right|, \frac{1}{2} \left( \mu_j^{(i-1)} + \left| \lambda_j^{(i-1)} \right| \right) \right]$$

Emilio Silva e Renato Picelli

## Programação Quadrática Seqüencial (PQS) 4

A matriz  $\mathbf{A}$  deve ser positiva-definida. É inicializada com a matriz identidade e atualizada com aproximação do tipo BFGS:

$$\mathbf{A}_{\text{novo}} = \mathbf{A} - \frac{\mathbf{A} \Delta \mathbf{x} \Delta \mathbf{x}^T \mathbf{A}}{\Delta \mathbf{x}^T \mathbf{A} \Delta \mathbf{x}} + \frac{\Delta l \Delta l^T}{\Delta \mathbf{x}^T \Delta \mathbf{x}}$$

$$\text{onde: } \Delta \mathbf{x} = \mathbf{x}_{i+1} - \mathbf{x}_i \quad \text{e} \quad \Delta l = \nabla_{\mathbf{x}} L(\mathbf{x}_{i+1}, \lambda_i) - \nabla_{\mathbf{x}} L(\mathbf{x}_i, \lambda_i)$$

Onde  $L$  é a função Lagrangeana e  $\nabla_{\mathbf{x}}$  denota o gradiente de  $L$  em relação a  $\mathbf{x}$ . Para garantir que  $\mathbf{A}$  seja positiva-definida  $\Delta l$  é modificado se  $\Delta \mathbf{x}^T \Delta l \leq 0,2 \Delta \mathbf{x}^T \mathbf{A} \Delta \mathbf{x}$  e substituído por:

$$\Delta l' = \theta \Delta l + (1 - \theta) \mathbf{A} \Delta \mathbf{x} \quad \text{onde: } \theta = \frac{0,8 \Delta \mathbf{x}^T \mathbf{A} \Delta \mathbf{x}}{\Delta \mathbf{x}^T \mathbf{A} \Delta \mathbf{x} - \Delta \mathbf{x}^T \Delta l}$$

Emilio Silva e Renato Picelli

## Exemplo

5

$$\begin{aligned} \text{Min}_{x_1, x_2} \quad & f(\mathbf{x}) = 3x_1 + \sqrt{3}x_2 \\ \text{tal que} \quad & g_1 = 3 - \frac{18}{x_1} - \frac{6\sqrt{3}}{x_2} \geq 0 \\ & g_2 = x_1 - 5,73 \geq 0 \\ & g_3 = x_2 - 7,17 \geq 0 \end{aligned}$$

Ponto de Partida:

$$x_1 = 11,61; x_2 = 7,17 \Rightarrow g_1 = 0; g_3 = 0; f = 47,25 \Rightarrow$$

$$\Rightarrow \nabla f = \begin{Bmatrix} 3 \\ \sqrt{3} \end{Bmatrix}; \quad \nabla g_1 = \begin{Bmatrix} 0,1335 \\ 0,2021 \end{Bmatrix}; \quad \nabla g_2 = \begin{Bmatrix} 1 \\ 0 \end{Bmatrix};$$

$$\nabla g_3 = \begin{Bmatrix} 0 \\ 1 \end{Bmatrix} \quad \text{Iniciando com } \mathbf{A} = \mathbf{I} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$\mathbf{s}$  é obtido resolvendo:

$$\begin{aligned} \text{Min}_{s_1, s_2} \quad & \phi(\mathbf{s}) = 47,25 + 3s_1 + \sqrt{3}s_2 + 0,5s_1^2 + 0,5s_2^2 \\ \text{tal que} \quad & g_1 = 0,1335s_1 + 0,2021s_2 \geq 0 \\ & g_2 = 5,88 + s_1 \geq 0 \\ & g_3 = s_2 \geq 0 \end{aligned}$$

Solução usando o comando **QUAPRO** do SCILAB:

$$\lambda_2 = \lambda_3 = 0; \lambda_1 = 12,8; s_1 = -1,29; s_2 = 0,855 \Rightarrow$$

$$\Rightarrow \mathbf{x}_1 = \begin{Bmatrix} 11,61 \\ 7,17 \end{Bmatrix} + \alpha \begin{Bmatrix} -1,29 \\ 0,855 \end{Bmatrix};$$

onde  $\alpha$  é obtido minimizando  $\psi(\alpha)$ . Na primeira iteração  $\mu_1 = |\lambda_1|$ :

$$\psi = 3(11,61 - 1,29\alpha) + \sqrt{3}(7,17 + 0,855\alpha) +$$

$$+ 12,8 \left| 3 - \frac{18}{11,61 - 1,29\alpha} - \frac{6\sqrt{3}}{7,17 + 0,855\alpha} \right| \Rightarrow$$

$$\Rightarrow \alpha = 2,2; \quad \mathbf{x}^{(1)} = (8,77; 9,05)^T; \quad f(\mathbf{x}^{(1)}) = 41,98$$

$$g_1(\mathbf{x}^{(1)}) = -0,201 \quad \text{Atualizando } \mathbf{A}:$$

$$L = 3x_1 + \sqrt{3}x_2 - 12,8 \left( 3 - \frac{18}{x_1} + \frac{6\sqrt{3}}{x_2} \right) \Rightarrow$$

$$\Rightarrow \nabla_x L = \left( 3 - 230,4/x_1^2, \sqrt{3} - 133/x_2^2 \right)^T \Rightarrow$$

$$\Rightarrow \Delta \mathbf{x} = \mathbf{x}^{(1)} - \mathbf{x}^{(0)} = \begin{Bmatrix} -2,84 \\ 1,88 \end{Bmatrix};$$

Emilio Silva e Renato Picelli

## Exemplo

6

$$\Delta l = \nabla_x L(\mathbf{x}^{(1)}) - \nabla_x L(\mathbf{x}^{(0)}) = \begin{Bmatrix} -1,31 \\ 0,963 \end{Bmatrix} \Rightarrow$$

$$\Rightarrow \text{Como } \mathbf{A} = \mathbf{I} \Rightarrow \Delta \mathbf{x}^T \mathbf{A} \Delta \mathbf{x} = 11,6;$$

$$\Delta \mathbf{x}^T \Delta l = 5,53 > 0, 2\Delta \mathbf{x}^T \mathbf{A} \Delta \mathbf{x} \Rightarrow$$

$$\Rightarrow \mathbf{A}_{\text{nov}} = \mathbf{I} - \frac{\Delta \mathbf{x} \Delta \mathbf{x}^T}{\Delta \mathbf{x}^T \Delta \mathbf{x}} + \frac{\Delta l \Delta l^T}{\Delta \mathbf{x}^T \Delta \mathbf{x}} = \begin{bmatrix} 0,453 & 0,352 \\ 0,352 & 0,775 \end{bmatrix}$$

$$\begin{aligned} \text{Min}_{s_1, s_2} \quad & \phi(\mathbf{s}) = 41,98 + 3s_1 + \sqrt{3}s_2 + 0,5(0,453s_1^2 + \\ & + 0,775s_2^2 + 0,704s_1s_2) \\ \text{tal que} \quad & g_1 = -0,201 + 0,234s_1 + 0,127s_2 \geq 0 \\ & g_2 = 3,04 + s_1 \geq 0 \\ & g_3 = 1,88 + s_2 \geq 0 \end{aligned}$$

Solução usando o comando **QUAPRO** do SCILAB:

$$\lambda_2 = \lambda_3 = 0; \lambda_1 = 14,31; s_1 = 1,059; s_2 = -0,376$$

$\alpha$  é obtido minimizando:

$$\psi(\alpha) = f(\alpha) + \mu_1 g_1(\alpha);$$

$$\text{onde: } \mu_1 = \max \left( \lambda_1, \frac{1}{2} (|\lambda_1| + \mu_1^{\text{velho}}) \right) = 14,31 \Rightarrow$$

$$\Rightarrow \alpha = 0,5 \Rightarrow \mathbf{x}^{(2)} = (9,30; 8,86)^T;$$

$$f(\mathbf{x}^{(2)}) = 43,25; \quad g_1(\mathbf{x}^{(2)}) = -0,108$$

$$\text{A solução ótima é: } \mathbf{x}^* = (9,46; 9,46)^T;$$

Emilio Silva e Renato Picelli

## MMA – Method of Moving Asymptotes

- Utilizado para problemas gerais de programação não lineares;
- É capaz de lidar com grande número de variáveis e restrições;
- O MMA lineariza o problema original, criando um subproblema de aproximação convexa;
- O MMA também calcula os limites inferior e superior dos valores que as variáveis de projeto podem assumir no subproblema linear. Esses limites são chamados de **Assíntotas**, que são **Móveis** por serem recalculadas a cada iteração da otimização;
- Variações: GCMMA, GCCMMA

Emilio Silva e Renato Picelli

## MMA – Method of Moving Asymptotes

Considere o problema:

Minimizar	$f_0(\mathbf{x})$	$(\mathbf{x} \in R^n)$
Tal que	$f_i(\mathbf{x}) \leq 0$	com $i = 1, \dots, m$
	$\underline{x}_j \leq x_j \leq \bar{x}_j$	para $j = 1, \dots, n$

### Utilizando o MMA

Dados de entrada a serem calculados na iteração  $k$

Função objetivo e derivada	$f_0^{(k)}(\mathbf{x}),$	$\frac{\partial f_0^{(k)}}{\partial x_j^{(k)}}$
Restrições e derivadas	$f_i^{(k)}(\mathbf{x}),$	$\frac{\partial f_i^{(k)}}{\partial x_j^{(k)}} \text{ com } i = 1, \dots, m$
Variáveis de projeto e Limites mínimo e máximo	$x_j^{(k)},$	$\underline{x}_j^{(k)}, \quad \bar{x}_j^{(k)}$

Emilio Silva e Renato Picelli

## MMA – Method of Moving Asymptotes

### Solução do problema dentro do MMA

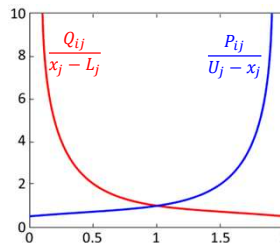
Para cada  $i = 0, 1, \dots, m$ , em cada iteração  $k$ , o subproblema convexo é escrito como:

$$f_i^{(k)}(\mathbf{x}) = r_i^{(k)} + \sum_{j=1}^n \left( \frac{P_{ij}^{(k)}}{U_j^{(k)} - x_j} + \frac{Q_{ij}^{(k)}}{x_j - L_j^{(k)}} \right)$$

onde

$$P_{ij}^{(k)} = \begin{cases} (U_j^{(k)} - x_j)^2 \partial f_i / \partial x_j, & \text{if } \partial f_i / \partial x_j > 0 \\ 0, & \text{if } \partial f_i / \partial x_j \leq 0 \end{cases}$$

$$Q_{ij}^{(k)} = \begin{cases} 0, & \text{if } \partial f_i / \partial x_j \geq 0 \\ -(x_j - L_j^{(k)})^2 \partial f_i / \partial x_j, & \text{if } \partial f_i / \partial x_j < 0 \end{cases}$$



$$r_i^{(k)} = f_i^{(k)}(\mathbf{x}) - \sum_{j=1}^n \left( \frac{P_{ij}^{(k)}}{U_j^{(k)} - x_j} + \frac{Q_{ij}^{(k)}}{x_j - L_j^{(k)}} \right) \rightarrow \text{Encontrar } x_j \text{ de modo que } r_i^{(k)} \rightarrow 0$$

↳ Resíduo

Emilio Silva e Renato Picelli

## MMA – Method of Moving Asymptotes

### Cálculo das Assíntotas Móveis

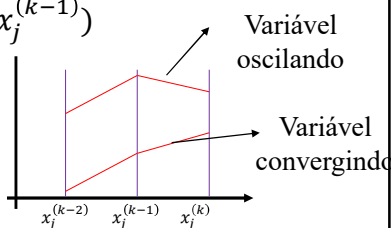
Considere  $k$  a iteração do loop de otimização.

$$\text{Para } k = 0 \text{ e } k = 1 \quad \begin{aligned} L_j^{(k)} &= x_j^{(k)} - (\bar{x}_j - \underline{x}_j) \\ U_j^{(k)} &= x_j^{(k)} + (\bar{x}_j - \underline{x}_j) \end{aligned}$$

$$\text{Para } k \geq 2 \quad \begin{aligned} L_j^{(k)} &= x_j^{(k)} - s_0 (x_j^{(k-1)} - L_j^{(k-1)}) \\ U_j^{(k)} &= x_j^{(k)} + s_0 (U_j^{(k-1)} - x_j^{(k-1)}) \end{aligned}$$

$$\text{Se } (x_j^{(k)} - x_j^{(k-1)}) \cdot (x_j^{(k-1)} - x_j^{(k-2)}) < 0 \rightarrow s_0 = 0.7$$

$$\text{Se } (x_j^{(k)} - x_j^{(k-1)}) \cdot (x_j^{(k-1)} - x_j^{(k-2)}) \geq 0 \rightarrow s_0 = 1.2$$



Emilio Silva e Renato Picelli

### Algoritmos Baseados em Programação Matemática

- não garantem que foi obtido o mínimo global (a menos que o problema seja convexo). Devem ser feitas várias tentativas alterando-se o “chute” inicial;
- há algoritmos que permitem utilizar variáveis discretas;
- Em geral, precisam de gradients.

### Algoritmos Probabilísticos

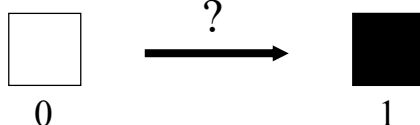
- Maior probabilidade de obter o mínimo global;
- Somente utilizam variáveis discretas;
- Não utilizam informação de gradientes;
- baseados em fenômenos observados na natureza
- processo de busca randômica guiada por decisões probabilísticas
- Imposição de restrições é difícil

variáveis discretas – espaço de solução disjunto e desconexo → introdução de múltiplos mínimos locais e dependência da discretização;

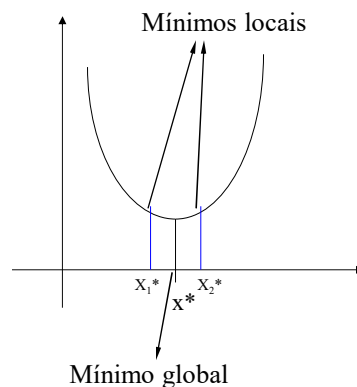
Emilio Silva e Renato Picelli

### Algoritmos com Variáveis Discretas

Como variar o material de zero a um?



O problema de otimização topológica considerando valores discretos (zero ou um) é mal-posto, ou seja, NÃO apresenta solução. Essencialmente, o que acontece é que a utilização de valores discretos origina múltiplos mínimos locais causando instabilidades numéricas. Isso também origina uma dependência da solução em relação a discretização do problema.



Emilio Silva e Renato Picelli

## Simulated Annealing

13

Algoritmos probabilísticos mais usados {

- “Simulated Annealing”
- Algoritmos Genéticos

“Simulated Annealing” ou Recozimento Simulado

Baseado num fenômeno de mecânica estatística relacionado com o equilíbrio de um grande número de átomos em sólidos e líquidos numa certa temperatura  $\Rightarrow$  solidificação de metais ou formação de cristais

Resfriamento rápido  $\Rightarrow$  estado sólido pouco estável (átomos assumem posições de mínimos locais de energia potencial na estrutura matricial do metal). Para obter um estado mais estável de energia (mínimo global)  $\Rightarrow$  Recozimento: metal é reaquecido até altas temperaturas e resfriado lentamente (átomos tem tempo para encontrar locais de energia potencial mínima globais estáveis)

Emilio Silva e Renato Picelli

## Simulated Annealing

14

Baseado no Algoritmo “Metropolis”

Dado  $T$  (temperatura)  $\Rightarrow$  perturbação randômica da posição do átomo  $\Rightarrow$  calcula  $\Delta E$ :

- Se  $\Delta E < 0$  aceita nova configuração de átomos;
- Se  $\Delta E \geq 0$  decisão é baseada na seguinte função probabilística que calcula a probabilidade de aceitação:

$$P(\Delta E) = e^{\left(\frac{-\Delta E}{k_B T}\right)} \quad k_B - \text{constante de Boltzmann}$$

A decisão é obtida escolhendo-se de forma randômica um número  $\rho$  em  $(0,1)$  e comparando com  $P(\Delta E)$ :

- Se  $\rho < P(\Delta E)$   $\Rightarrow$  configuração é aceita
- Se  $\rho > P(\Delta E)$   $\Rightarrow$  configuração é rejeitada

Emilio Silva e Renato Picelli

## Simulated Annealing

15

Note que:

- Se  $T$  é alto  $\Rightarrow P(\Delta E)$  é próximo de 1;
- Se  $T$  é próximo de zero  $P(\Delta E)$  é muito pequeno;

Assim a cada temperatura um conjunto de estruturas atômica seria gerada pela perturbação randômica da posição até que o “equilíbrio térmico” seja atingido (estado estável). A temperatura é reduzida e as iterações são repetidas. Os passos são repetidos iterativamente enquanto a temperatura é reduzida de forma lenta até atingir o mínimo estado de energia.

Analogia com o problema matemático de otimização

Estados de energia  $\Rightarrow$  funções objetivo  
Configurações dos átomos  $\Rightarrow$  variáveis de projeto  $x$   
Temperatura  $T \Rightarrow$  parâmetro de controle de convergência  
Somente valores de função são usados

Emilio Silva e Renato Picelli

## Simulated Annealing

16

### Desempenho do Método

- Temperatura  $T_0$ ;
  - Atualização de  $T$ ;
  - Número de iterações (combinações de variáveis de projeto) necessárias para atingir o “equilíbrio térmico”, antes de reduzir  $T$ ;
- } “Cooling Schedule”

Se  $T_0$  é baixo  $\Rightarrow$  baixa probabilidade de atingir o mínimo

Escolha de  $T_0$ :

$$P(\Delta E) = 0,95 = e^{\left(\frac{-\bar{\Delta f}}{T_0}\right)} \Rightarrow T_0 = \frac{\bar{\Delta f}}{\ln(1/0,95)}$$

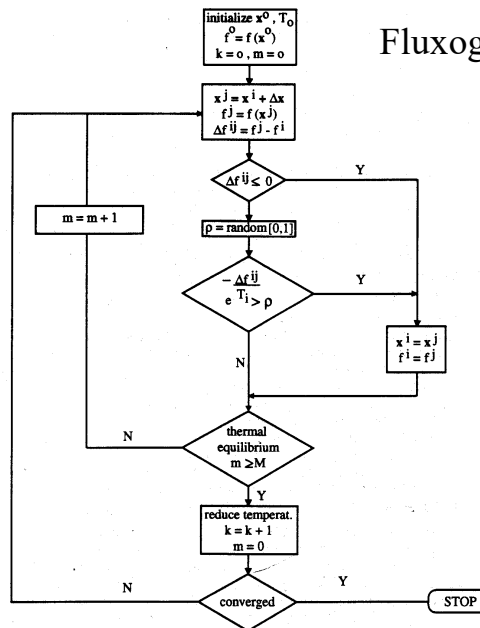
Regras de Atualização da temperatura:

- $T_{k+1} = \alpha T_k \quad k = 0,1,2,\dots,K \quad 0,5 \leq \alpha \leq 0,95$
- Dividir  $[0, T_0]$  em  $K$  passos:  $T_k = \frac{K-k}{K} T_0 \quad k = 0,1,2,\dots,K$

Emilio Silva e Renato Picelli



## Fluxograma do Método



Emilio Silva e Renato Picelli

Derivados da biologia ➡ Teoria de Darwin (sobrevivência do mais resistente. Ao longo das gerações, características que são úteis para sobrevivência são passadas adiante para os indivíduos sucessores.

Essas características são armazenadas na codificação dos cromossomos. Os mecanismos da genética para troca randômica de informações entre os cromossomos dos pais reprodutores são baseadas nas seguintes operações: reprodução, cruzamento, ocasional, mutação e inversão do código cromossômico.

Algoritmos Genéticos simulam os mecanismos da genética natural para solução de problemas de otimização. Código cromossômico é representado por uma “palavra”. As operações envolvem trocas randômicas das localizações de números numa palavra. Utiliza somente o valor da função.

Emilio Silva e Renato Picelli

Representação da combinação de variáveis de projeto palavras de bits que “emulam” os cromossomos. Por exemplo:

$$\{x_1, x_2, x_3, x_4\} = \{6, 5, 3, 11\} = \underbrace{0110}_x \underbrace{1011}_x \underbrace{1110}_x \underbrace{1011}_x$$

Ideal para variáveis discretas ou inteiras, no caso de variáveis contínuas  $\Rightarrow$  grande número de bits para representação (depende da precisão desejada). Nesse caso, número  $m$  de dígitos binários necessários para representar uma variável  $x_i$  num intervalo  $x_i^L$  e  $x_i^U$  com precisão  $x^{incr}$  vale:

$$\{x_i^L \leq x_i \leq x_i^U\} \Rightarrow 2^m \geq ((x_i^U - x_i^L) / x^{incr} + 1)$$

Trabalha com população de palavras (ou cromossomos) e não com um único ponto do espaço de projeto  $\Rightarrow$  Implementação com vantagens em computadores paralelos. O resultado do algoritmo genético é uma população de bons projetos (“palavras”).

Exemplo de seqüência de operações de um algoritmo genético:

1. Dimensão da população é escolhida e valor das variáveis em cada “palavra” é decidido randomicamente (0s e 1s para os bits).
2. Reprodução: “palavras” com um bom valor de função objetivo são copiadas para formar uma nova população (Teoria de Darwin), ou seja, é aumentada a probabilidade de sua escolha em relação ao resto da população. A nova população terá múltipla cópias dos indivíduos mais resistentes.

3. Cruzamento: “palavras” da nova população são agrupadas aleatoriamente em pares para o cruzamento. Um inteiro aleatório  $k$  é selecionado entre 1 e  $L-1$ , onde  $L$  é o comprimento da palavra, e então (por exemplo,  $L=9$  e  $k=5$ ):

pai 1 : 01101||0111      palavra gerada 1 : 01101||0001  
 pai 2 : 01001||0001      palavra gerada 2 : 01001||0111

Cruzamento de “um ponto”. Outras possibilidades: cruzamento de “dois pontos” e “multi-ponto”.

4. Mutação: seleciona uma palavra aleatoriamente e altera seu valor de 0 para 1 ou vice-versa. Evita uniformidade, ou seja, que haja muitas palavras iguais na mesma população, o que ocorre no estágio da reprodução. Caso contrário, a chance de encontrar melhores soluções seria reduzida. Efeito no desempenho do algoritmo é secundário (1 mutação em 1000 operações de bit).

## Algoritmos heurísticos não funcionam para Otimização Topológica!!

Struct Multidisc Optim (2011) 43:589–596  
 DOI 10.1007/s00158-011-0638-7

FORUM

### On the usefulness of non-gradient approaches in topology optimization

Ole Sigmund

Received: 1 February 2011 / Accepted: 11 February 2011 / Published online: 31 March 2011  
 © Springer-Verlag 2011

**Abstract** Topology optimization is a highly developed tool for structural design and is by now being extensively used in mechanical, automotive and aerospace industries throughout the world. Gradient-based topology optimization algorithms may efficiently solve fine-resolution problems with thousands and up to millions of design variables using a few hundred (finite element) function evaluations (and even less than 50 in some commercial codes). Nevertheless, non-gradient topology optimization approaches that require orders of magnitude more function evaluations for extremely low resolution examples keep appearing in the literature. This forum article discusses the practical and scientific relevance of publishing papers that use immense computational resources for solving simple problems for which there already exist efficient solution techniques.

**Keywords** Topology optimization · Genetic Algorithms · Stochastic optimization · Discrete optimization

#### 1 Introduction

Since its introduction more than two decades ago the

automotive and aerospace industries throughout the world. Gradient-based topology optimization techniques for continuum problems encompass the homogenization approach (Bendsøe and Kikuchi 1988), the density (SIMP) approach (Bendsøe 1989; Zhou and Rozvany 1991; Mlejnek 1992; Sigmund 2001), the level-set approach (Wang et al. 2003; Allaire et al. 2004), the evolutionary structural optimization approach (Xie and Steven 1997), phase-field methods (Wang and Zhou 2004), topological derivatives (Stokichowski and Zochowski 1999) etc. The required number of function evaluations (finite element calculations) is similar for the different gradient-based approaches and typically lies in the interval between 50 and 1,000, depending on physical problem but relatively independent on number of design variables. For 2D problems discretizations using 1,000 elements is the minimum (e.g. used in the interactive topology optimization applet found at [www.topopt.dtu.dk](http://www.topopt.dtu.dk)) and up to 100,000 for large-scale, fine-resolution problems, whereas discretizations in 3D problems easily reach millions of elements. Despite the use of continuous design variables in most gradient-based methods, discrete final designs can usually be obtained using penalization techniques, continuation approaches or postprocessing by thresholding and lately quite systematically using projection schemes (Guest

# Sensibilidade em Otimização

23

Objetivo: Calcular as derivadas da função objetivo e restrições em relação às variáveis de projeto

Permite analisar a mudança do comportamento da estrutura devido a pequenas mudanças de seus parâmetros (dimensões, propriedades de materiais, etc...) com custo menor do que realizar novas análises.

Métodos para o cálculo de sensibilidade

- Diferenças Finitas
- Métodos Semi-analíticos
- Métodos Analíticos

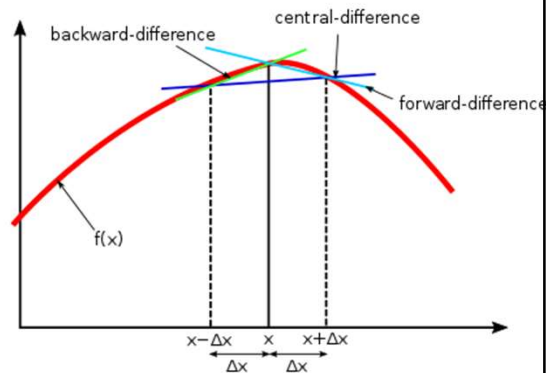
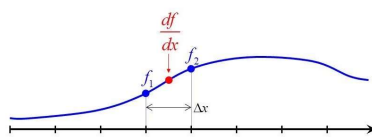
Precisão aumenta

Emilio Silva e Renato Picelli

# Método das diferenças finitas

9

$$\frac{df_{1.5}}{dx} \approx \frac{f_2 - f_1}{\Delta x}$$



Emilio Silva e Renato Picelli

# Cálculo da Sensibilidade

25

## Aproximação por Diferenças Finitas

Progressiva:	$\frac{du}{dx} \cong \frac{u(x+\Delta x) - u(x)}{\Delta x} + O(\Delta x)$	Para n variáveis de projeto: Requer n análises adicionais
Regressiva:	$\frac{du}{dx} \cong \frac{u(x) - u(x-\Delta x)}{\Delta x} + O(\Delta x)$	Requer n análises adicionais
Central:	$\frac{du}{dx} \cong \frac{u(x+\Delta x) - u(x-\Delta x)}{2\Delta x} + O(\Delta x^2)$	Requer 2n análises adicionais

↓  
Altas ordens possuem erro menor mas envolvem mais pontos (custo computacional aumenta)

Composição do erro em diferenças finitas

- Truncamento:
- Condicionamento (ou arredondamento);

Emilio Silva e Renato Picelli

# Cálculo da Sensibilidade

26

$$\begin{aligned}
 \text{---} u(x+\Delta x) &= u(x) + \Delta x \frac{du(x)}{dx} + \frac{\Delta x^2}{2} \frac{d^2u(\xi)}{dx^2} \Rightarrow \frac{du(x)}{dx} = \frac{u(x+\Delta x) - u(x)}{\Delta x} + \frac{\Delta x}{2} \frac{d^2u(\xi)}{dx^2} \Rightarrow \\
 \Rightarrow e_T(\Delta x) &= \frac{\Delta x}{2} \frac{d^2u(\xi)}{dx^2} \\
 \text{---} u(x-\Delta x) &= u(x) - \Delta x \frac{du(x)}{dx} + \frac{\Delta x^2}{2} \frac{d^2u(\xi)}{dx^2} \Rightarrow \frac{du(x)}{dx} = \frac{u(x) - u(x-\Delta x)}{\Delta x} + \frac{\Delta x}{2} \frac{d^2u(\xi)}{dx^2} \Rightarrow \\
 \Rightarrow e_T(\Delta x) &= \frac{\Delta x}{2} \frac{d^2u(\xi)}{dx^2} \\
 &= \\
 u(x+\Delta x) - u(x-\Delta x) &= 2\Delta x \frac{du(x)}{dx} + \frac{\Delta x^3}{3} \frac{d^3u(\xi)}{dx^3} \Rightarrow \frac{du(x)}{dx} = \frac{u(x+\Delta x) - u(x-\Delta x)}{2\Delta x} - \frac{\Delta x^2}{6} \frac{d^3u(\xi)}{dx^3} \Rightarrow \\
 \Rightarrow e_T(\Delta x) &= -\frac{\Delta x^2}{6} \frac{d^3u(\xi)}{dx^3}
 \end{aligned}$$

Emilio Silva e Renato Picelli

# Aproximação por Diferenças Finitas

27

## Erros em diferenças finitas

Erro de truncamento:

- erro de aproximação da fórmula da derivada;
- aumenta com o aumento de  $\Delta x$ ;

Estimativa:

Seja a Série de Taylor Progressiva:

$$u(x + \Delta x) = u(x) + \Delta x \frac{du(x)}{dx} + \frac{\Delta x^2}{2} \frac{d^2u(\xi)}{dx^2} \Rightarrow$$

$$\Rightarrow \frac{du(x)}{dx} = \frac{u(x + \Delta x) - u(x)}{\Delta x} + \frac{\Delta x}{2} \frac{d^2u(\xi)}{dx^2} \Rightarrow$$

$$\Rightarrow e_T(\Delta x) = \frac{\Delta x}{2} \frac{d^2u(\xi)}{dx^2}$$

Analogamente para a diferença central:

$$e_T(\Delta x) = \frac{\Delta x^2}{6} \frac{d^3u(\xi)}{dx^3}$$

Erro de condicionamento (ou arredondamento):

- depende do computador;
- não é uma função contínua;
- diminui com o aumento de  $\Delta x$ ;
- importante em métodos iterativos;

Estimativa:

$$e_c(\Delta x) = \frac{2}{\Delta x} \varepsilon_u$$

Limite no erro absoluto para o cálculo da função u

Emilio Silva e Renato Picelli

# Aproximação por Diferenças Finitas

28

Estimativa do erro total:  $e = \frac{\Delta x}{2} \left| \frac{d^2u}{dx^2} \right| + \frac{2}{\Delta x} \varepsilon_u$

Limite no cálculo da segunda derivada

Dilema no tamanho do passo  $\Delta x$

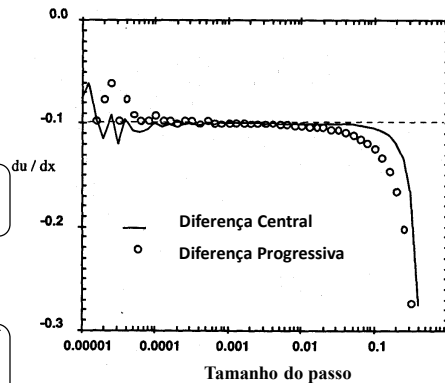
$\Delta x$  grande

predomina erro de truncamento

$\Delta x$  pequeno

predomina erro de condicionamento

$$e = 0 \Rightarrow \Delta x_{opt} = 2 \sqrt{\frac{\varepsilon_u}{|S_b|}}$$



Emilio Silva e Renato Picelli

## Exemplo

29

$$\begin{array}{l} \text{Min}_{x_1, x_2} \quad f(\mathbf{x}) = x_1 + x_2 \\ \text{tal que} \quad g(\mathbf{x}) = \frac{1}{x_1 x_2} - 1 = 0 \end{array}$$

$$\text{Assim: } f(\mathbf{x}) = \frac{1}{x_2} + x_2 \Rightarrow \frac{df}{dx_2} = -\frac{2}{x_2^3} + 1$$

Calculando a derivada em dois pontos

temos:

$$\begin{aligned} \text{Em } x_2 = 1 \quad (\Delta x_2 = 0,01) \Rightarrow \frac{\Delta f}{\Delta x_2} &= \frac{1/1,01^2 + 1,01 - 2}{0,01} = \\ &= -0,97 \end{aligned}$$

$$3\% \text{ em relação ao valor exato } \frac{df}{dx_2} = -1,0$$

$$\begin{aligned} \text{Em } x_2 = 1,29 \quad (\Delta x_2 = 0,01) \Rightarrow \frac{\Delta f}{\Delta x_2} &= \\ &= \frac{1/1,3^2 + 1,3 - (1/1,29^2 + 1,29)}{0,01} = -0,0791 \end{aligned}$$

$$16\% \text{ em relação ao valor exato } \frac{df}{dx_2} = -0,0683$$

Derivadas logarítmicas:

$$\begin{aligned} \text{Em } x_2 = 1 \Rightarrow f = 2 \Rightarrow \frac{d_L f}{dx_2} &= \frac{\Delta f}{\Delta x_2} \frac{x_2}{f} = \\ &= -0,97 \times 1/2 = -0,485 \end{aligned}$$

$$\begin{aligned} \text{Em } x_2 = 1,29 \Rightarrow f = 1,891 \Rightarrow \frac{d_L f}{dx_2} &= \\ &= \frac{\Delta f}{\Delta x_2} \frac{x_2}{f} = -0,0791 \times 1,29/1,891 = \\ &= -0,054 \end{aligned}$$

Muito pequena, indicando pouca precisão no cálculo, como verificado.

Emilio Silva e Renato Picelli

## Métodos Analíticos

30

Seja a equação de equilíbrio:  $\mathbf{K}\mathbf{u} = \mathbf{F}$  onde:  $\mathbf{u}$  - vetor de deslocamentos nodais;  $\mathbf{K}$  - matriz de rigidez;  $\mathbf{F}$  - vetor de carga; Considerando uma função  $f(\mathbf{u}, x)$  que dependa de  $\mathbf{u}$  e  $x$  (variável de projeto), temos que a sensibilidade de  $f$  é dada por:

$$\frac{df}{dx} = \frac{\partial f}{\partial x} + \mathbf{z}^T \frac{d\mathbf{u}}{dx} \quad \text{onde: } \mathbf{z} = \nabla f \left( z_i = \frac{\partial f}{\partial u_i} \right)$$

Parte explícita  
(geralmente zero)

Parte implícita

Exemplo: Seja  $\mathbf{u}^T = (u_1 \quad u_2 \quad u_3)$  e

$$f(\mathbf{u}, x) = xu_1 + u_2^2 - \ln(u_3) + x^2 \Rightarrow \frac{\partial f}{\partial x} = u_1 + 2x; \quad \mathbf{z} = \begin{Bmatrix} \partial f / \partial u_1 \\ \partial f / \partial u_2 \\ \partial f / \partial u_3 \end{Bmatrix} = \begin{Bmatrix} x \\ 2u_2 \\ -1/u_3 \end{Bmatrix}; \quad \frac{d\mathbf{u}}{dx} = \begin{Bmatrix} du_1 / dx \\ du_2 / dx \\ du_3 / dx \end{Bmatrix} \Rightarrow$$

$$\Rightarrow \frac{df}{dx} = \frac{\partial f}{\partial x} + \mathbf{z}^T \frac{d\mathbf{u}}{dx} = u_1 + 2x + x \frac{du_1}{dx} + 2u_2 \frac{du_2}{dx} - \frac{1}{u_3} \frac{du_3}{dx}$$

Emilio Silva e Renato Picelli

# Métodos Analíticos

31

Para obter  $\frac{du}{dx}$  existem dois métodos clássicos:

## Método Direto

Resolvido para cada variável de projeto

$$\mathbf{K} \frac{d\mathbf{u}}{dx} = \frac{d\mathbf{F}}{dx} - \frac{d\mathbf{K}}{dx} \mathbf{u} \Rightarrow \frac{d\mathbf{u}}{dx} = \mathbf{K}^{-1} \left( \frac{d\mathbf{F}}{dx} - \frac{d\mathbf{K}}{dx} \mathbf{u} \right) \Rightarrow$$

$$\Rightarrow \mathbf{z}^T \frac{d\mathbf{u}}{dx} = \mathbf{z}^T \mathbf{K}^{-1} \left( \frac{d\mathbf{F}}{dx} - \frac{d\mathbf{K}}{dx} \mathbf{u} \right) \Rightarrow \frac{df}{dx} = \frac{\partial f}{\partial \mathbf{x}} + \mathbf{z}^T \mathbf{K}^{-1} \left( \frac{d\mathbf{F}}{dx} - \frac{d\mathbf{K}}{dx} \mathbf{u} \right)$$

Mais eficiente quando: n° variáveis de projeto < n° de restrições

## Método Adjunto

Resolvido para função objetivo e cada restrição

$$\mathbf{K}\lambda = \mathbf{z} \Rightarrow \lambda = \mathbf{K}^{-1}\mathbf{z} \Rightarrow \lambda^T = \mathbf{z}^T \mathbf{K}^{-1} \quad (\mathbf{K}^{-1} = (\mathbf{K}^{-1})^T)$$

$$\frac{df}{dx} = \frac{\partial f}{\partial \mathbf{x}} + \mathbf{z}^T \frac{d\mathbf{u}}{dx} = \frac{\partial f}{\partial \mathbf{x}} + \mathbf{z}^T \mathbf{K}^{-1} \left( \frac{d\mathbf{F}}{dx} - \frac{d\mathbf{K}}{dx} \mathbf{u} \right) \Rightarrow \frac{df}{dx} = \frac{\partial f}{\partial \mathbf{x}} + \lambda^T \left( \frac{d\mathbf{F}}{dx} - \frac{d\mathbf{K}}{dx} \mathbf{u} \right)$$

Mais eficiente quando: n° variáveis de projeto > n° de restrições

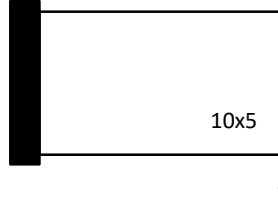
Função Multiobjetivo  $\Rightarrow$  Método Adjunto

Emilio Silva e Renato Picelli

# Método direto

15

- Derivada da função objetivo “Flexibilidade” com relação às variáveis de projeto



$$f(\mathbf{x}) = \mathbf{F}^T \mathbf{U} \Rightarrow f'(\mathbf{x}) = \frac{d(\mathbf{F}^T \mathbf{U})}{dx_i} = \frac{d(\mathbf{U}^T \mathbf{K} \mathbf{U})}{dx_i} =$$

$$\mathbf{F} = \mathbf{K} \mathbf{U} \Rightarrow \frac{d\mathbf{U}^T}{dx_i} \mathbf{K} \mathbf{U} + \mathbf{U}^T \mathbf{K} \frac{d\mathbf{U}}{dx_i} + \mathbf{U}^T \frac{d\mathbf{K}}{dx_i} \mathbf{U} =$$

$$= 2\mathbf{U}^T \mathbf{K} \frac{d\mathbf{U}}{dx_i} + \mathbf{U}^T \frac{d\mathbf{K}}{dx_i} \mathbf{U}; \text{ Encontrar } \frac{d(\mathbf{U})}{dx_i}:$$

$$\frac{d\mathbf{F}}{dx_i} = \frac{d(\mathbf{K} \mathbf{U})}{dx_i} \Rightarrow \frac{d\mathbf{F}}{dx_i} = \mathbf{K} \frac{d\mathbf{U}}{dx_i} + \frac{d\mathbf{K}}{dx_i} \mathbf{U} \Rightarrow \frac{d\mathbf{U}}{dx_i} = -\mathbf{K}^{-1} \frac{d\mathbf{K}}{dx_i} \mathbf{U}$$

Substituindo:

$$f'(\mathbf{x}) = -2\mathbf{U}^T \mathbf{K} \mathbf{K}^{-1} \frac{d\mathbf{K}}{dx_i} \mathbf{U} + \mathbf{U}^T \frac{d\mathbf{K}}{dx_i} \mathbf{U} \Rightarrow f'(\mathbf{x}) = -\mathbf{U}^T \frac{d\mathbf{K}}{dx_i} \mathbf{U}$$

Emilio Silva e Renato Picelli



## Derivada da matriz de rigidez

16

- Calcular  $\frac{d\mathbf{K}}{dx_i}$

Utilizando um **modelo de material** como auxílio:

$$\mathbf{K}(\mathbf{x}) = \mathbb{A} \rightarrow \mathbf{K}_i(x_i) \Rightarrow \mathbf{K}(\mathbf{x}) = \mathbb{A} \rightarrow x_i^p \mathbf{K}_i$$

$$\frac{d\mathbf{K}}{dx_i} = p x_i^{p-1} \mathbf{K}_i \Rightarrow f'(\mathbf{x}) = -p x_i^{p-1} \mathbf{u}_i^T \mathbf{K}_i \mathbf{u}_i$$

Emilio Silva e Renato Picelli

## Método adjunto

17

- Derivada da função Lagrangeana (estendida)

$$L(\mathbf{x}) = f(\mathbf{x}) + \overset{0}{\text{equilíbrio}} \Rightarrow \frac{dL(\mathbf{x})}{dx_i} = \frac{df(\mathbf{x})}{dx_i}$$

$$\mathbf{F} = \mathbf{K}\mathbf{U}$$

$$L(\mathbf{x}) = \mathbf{F}^T \mathbf{U} + \lambda^T (\mathbf{K}\mathbf{U} - \mathbf{F}) \Rightarrow \frac{dL(\mathbf{x})}{dx_i} = \frac{d(\mathbf{F}^T \mathbf{U})}{dx_i} + \lambda^T \frac{d(\mathbf{K}\mathbf{U} - \mathbf{F})}{dx_i} \Rightarrow$$

variável adjunta

$$\Rightarrow \frac{dL(\mathbf{x})}{dx_i} = \mathbf{F}^T \frac{d\mathbf{U}}{dx_i} + \lambda^T \left( \frac{d\mathbf{K}}{dx_i} \mathbf{U} + \mathbf{K} \frac{d\mathbf{U}}{dx_i} \right)$$

- Agrupar derivadas desconhecidas, e.g.,  $\frac{d\mathbf{U}}{dx_i}$ :

$$\frac{dL(\mathbf{x})}{dx_i} = (\mathbf{F}^T + \lambda^T \mathbf{K}) \frac{d\mathbf{U}}{dx_i} + \lambda^T \frac{d\mathbf{K}}{dx_i} \mathbf{U} \quad \text{Para cancelar } \frac{d\mathbf{U}}{dx_i} :$$

$$\mathbf{F}^T + \lambda^T \mathbf{K} = \mathbf{0} \Rightarrow \mathbf{K}\lambda = -\mathbf{F} \Rightarrow \lambda = -\mathbf{U}$$

Emilio Silva e Renato Picelli

## Método adjunto

135

•  $\lambda = -\mathbf{U}$ :  $\frac{dL(\mathbf{x})}{dx_i} = (\mathbf{F}^T - \mathbf{U}^T \mathbf{K}) \frac{d\mathbf{U}}{dx_i} - \mathbf{U}^T \frac{d\mathbf{K}}{dx_i} \mathbf{U} \Rightarrow \frac{dL(\mathbf{x})}{dx_i} = -\mathbf{U}^T \frac{d\mathbf{K}}{dx_i} \mathbf{U}$

$$\frac{dL(\mathbf{x})}{dx_i} = \frac{df(\mathbf{x})}{dx_i} \Rightarrow \frac{df(\mathbf{x})}{dx_i} = -px_i^{p-1} \mathbf{u}_i^T \mathbf{K}_i \mathbf{u}_i$$

- Geralmente

Função objetivo:  $\phi = \phi(\mathbf{U}(\mathbf{x}))$       $\mathbf{R} = \mathbf{K}\mathbf{U} - \mathbf{F} = \mathbf{0}$

Problema adjunto:  $L(\mathbf{x}) = \phi + \boldsymbol{\beta}^T \mathbf{R} = \phi$

Sensibilidade:  $\frac{dL(\mathbf{x})}{dx_i} = \frac{d\phi}{dx_i} + \boldsymbol{\beta}^T \frac{d\mathbf{R}}{dx_i} = \frac{d\phi}{dx_i} \Rightarrow \frac{d\mathbf{R}}{dx_i} = \frac{\partial \mathbf{R}}{\partial x_i} + \frac{\partial \mathbf{R}}{\partial \mathbf{U}} \frac{\partial \mathbf{U}}{\partial x_i} = \mathbf{0}$

$$\Rightarrow \frac{\partial \mathbf{U}}{\partial x_i} = - \left( \frac{\partial \mathbf{R}}{\partial \mathbf{U}} \right)^{-1} \frac{\partial \mathbf{R}}{\partial x_i} = -(\mathbf{K})^{-1} \frac{\partial \mathbf{R}}{\partial x_i};$$

Emilio Silva e Renato Picelli

## Método adjunto

19

$\frac{d\phi}{dx_i} = \frac{\partial \phi}{\partial x_i} + \frac{\partial \phi}{\partial \mathbf{U}} \frac{\partial \mathbf{U}}{\partial x_i}$  Substituindo  $\frac{\partial \mathbf{U}}{\partial x_i}$ :

$$\frac{d\phi}{dx_i} = \frac{\partial \phi}{\partial x_i} - \frac{\partial \phi}{\partial \mathbf{U}} (\mathbf{K})^{-1} \frac{\partial \mathbf{R}}{\partial x_i} = \frac{\partial \phi}{\partial x_i} + \boldsymbol{\lambda} \frac{\partial \mathbf{R}}{\partial x_i} \Rightarrow \mathbf{K}\boldsymbol{\lambda} = -\frac{\partial \phi}{\partial \mathbf{U}}$$

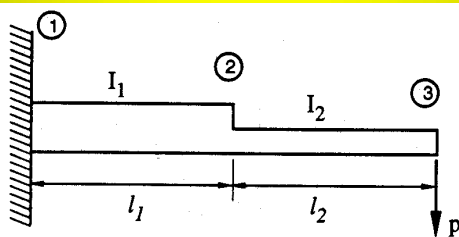
$$\frac{d\phi}{dx_i} = \frac{\partial \phi}{\partial x_i} + \boldsymbol{\lambda} \frac{\partial \mathbf{R}}{\partial x_i}$$

Por exemplo, se:  $\frac{\partial \mathbf{R}}{\partial x_i} = \frac{\partial \mathbf{K}}{\partial x_i} \mathbf{U} \Rightarrow \frac{d\phi}{dx_i} = \frac{\partial \phi}{\partial x_i} + \boldsymbol{\lambda}^T \frac{d\mathbf{K}}{dx_i} \mathbf{U}$

Emilio Silva e Renato Picelli

## Exemplo

37



Restrição de deslocamento na ponta:  $g = c - w_3 \geq 0$ ; Calcular  $\frac{dg}{dI_1}$

$$\mathbf{Ku} = \mathbf{F} \text{ onde:}$$

$$\mathbf{K} = \mathbf{E} \begin{bmatrix} 12(I_1/l_1^3 + I_2/l_2^3) & -6(I_1/l_1^2 - I_2/l_2^2) & -12I_2/l_2^3 & 6I_2/l_2^2 \\ -6(I_1/l_1^2 - I_2/l_2^2) & 4(I_1/l_1 + I_2/l_2) & -6I_2/l_2^2 & 2I_2/l_2 \\ \text{sim.} & & 12I_2/l_2^3 & -6I_2/l_2^2 \\ & & -6I_2/l_2^2 & 4I_2/l_2 \end{bmatrix}; \mathbf{u} = \begin{Bmatrix} w_2 \\ \theta_2 \\ w_3 \\ \theta_3 \end{Bmatrix}; \mathbf{F} = \begin{Bmatrix} 0 \\ 0 \\ p \\ 0 \end{Bmatrix} \Rightarrow$$

$$\Rightarrow \mathbf{u} = \begin{pmatrix} p \\ E \end{pmatrix} \begin{Bmatrix} l_1^3/3I_1 + l_1^2 l_2/2I_1 \\ l_1^2/2I_1 + l_1 l_2/I_1 \\ (l_1^3 + 3I_1^2 l_2 + 3I_1 l_2^2)/3I_1 + l_2^3/3I_2 \\ l_1^2/2I_1 + l_1 l_2/I_1 + l_2^2/I_2 \end{Bmatrix}$$

Emilio Silva e Renato Picelli

## Exemplo

38

### 1º Método: Método Direto

$$g = c - w_3 \geq 0 \Rightarrow \frac{dg}{dI_1} = \frac{\partial g}{\partial I_1} + \mathbf{z}^T \frac{d\mathbf{u}}{dI_1} = \mathbf{z}^T \frac{d\mathbf{u}}{dI_1}; \frac{d\mathbf{u}}{dI_1} = \mathbf{K}^{-1} \left( \frac{d\mathbf{F}}{dI_1} - \frac{d\mathbf{K}}{dI_1} \mathbf{u} \right); \frac{d\mathbf{F}}{dI_1} = \mathbf{0};$$

$$\frac{d\mathbf{K}}{dI_1} = \begin{pmatrix} E \\ I_1^3 \end{pmatrix} \begin{bmatrix} 12 & -6l_1 & 0 & 0 \\ -6l_1 & 4l_1^2 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \Rightarrow \frac{d\mathbf{K}}{dI_1} \mathbf{u} = \begin{pmatrix} E \\ I_1^3 \end{pmatrix} \begin{Bmatrix} 12w_2 - 6l_1\theta_2 \\ -6l_1w_2 + 4l_1^2\theta_2 \\ 0 \\ 0 \end{Bmatrix} = \begin{pmatrix} p \\ I_1 \end{pmatrix} \begin{Bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{Bmatrix} \Rightarrow$$

$$\Rightarrow \frac{d\mathbf{u}}{dI_1} = -\mathbf{K}^{-1} \frac{d\mathbf{K}}{dI_1} \mathbf{u} = -\frac{p}{EI_1^2} \begin{Bmatrix} l_1^2 l_2/2 + l_1^3/3 \\ l_1 l_2 + l_1^2/2 \\ l_1^2 l_2 + l_1 l_2^2 + l_1^3/3 \\ l_1 l_2 + l_1^2/2 \end{Bmatrix}; \mathbf{z} = \begin{Bmatrix} \partial g / \partial w_2 \\ \partial g / \partial \theta_2 \\ \partial g / \partial w_3 \\ \partial g / \partial \theta_3 \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \\ -1 \\ 0 \end{Bmatrix} \Rightarrow$$

$$\Rightarrow \frac{dg}{dI_1} = \mathbf{z}^T \frac{d\mathbf{u}}{dI_1} = \frac{p}{EI_1^2} \left( l_1^2 l_2 + l_1 l_2^2 + \frac{l_1^3}{3} \right)$$

Emilio Silva e Renato Picelli

## Exemplo

39

### 2º Método: Método Adjunto

$$g = c - w_3 \geq 0 \Rightarrow \frac{dg}{dI_1} = \frac{\partial g}{\partial I_1} + \lambda^T \left( \frac{dF}{dI_1} - \frac{dK}{dI_1} \mathbf{u} \right); \frac{dF}{dI_1} = \mathbf{0}; \mathbf{K}\lambda = \mathbf{z} \Rightarrow \lambda = \mathbf{K}^{-1}\mathbf{z} = \mathbf{K}^{-1} \begin{Bmatrix} 0 \\ 0 \\ -1 \\ 0 \end{Bmatrix} =$$

$$= -\left(\frac{1}{E}\right) \begin{Bmatrix} l_1^3/3I_1 + l_1^2 l_2/2I_1 \\ l_1^2/2I_1 + l_1 l_2/I_1 \\ (l_1^3 + 3l_1^2 l_2 + 3l_1 l_2^2)/3I_1 + l_2^3/3I_2 \\ l_1^2/2I_1 + l_1 l_2/I_1 + l_2^2/I_2 \end{Bmatrix} \Rightarrow \frac{dg}{dI_1} = -\lambda^T \frac{dK}{dI_1} \mathbf{u} = \frac{p}{EI_1^2} \left( l_1^2 l_2 + l_1 l_2^2 + \frac{l_1^3}{3} \right)$$

### 3º Método: Derivação da expressão de $w_3$

$$w_3 = \frac{p}{3EI_1} (l_1^3 + 3l_1^2 l_2 + 3l_1 l_2^2) + \frac{pl_2^3}{3EI_2} \Rightarrow \frac{dw_3}{dI_1} = \frac{p}{EI_1^2} \left( \frac{l_1^3}{3} + l_1^2 l_2 + l_1 l_2^2 \right)$$

Emilio Silva e Renato Picelli

## Sensibilidade de Autovalores e Autovetores

40

Considere o problema de autovetor e autovalor:

$$(\mathbf{K} - \mu \mathbf{M})\mathbf{u} = \mathbf{0} \quad (\mu = \omega^2) \text{ e } \mathbf{u}^T \mathbf{M} \mathbf{u} = 1$$

autovalor
autovetor

Objetivo é encontrar:  $\frac{d\mu}{dx_i}$  e  $\frac{d\mathbf{u}}{dx_i}$

Derivando as equações acima:

$$(\mathbf{K} - \mu \mathbf{M})\mathbf{u} = \mathbf{0} \Rightarrow (\mathbf{K} - \mu \mathbf{M}) \frac{d\mathbf{u}}{dx_i} - \frac{d\mu}{dx_i} \mathbf{M} \mathbf{u} = - \left( \frac{d\mathbf{K}}{dx_i} - \mu \frac{d\mathbf{M}}{dx_i} \right) \mathbf{u} \quad (1)$$

$$\mathbf{u}^T \mathbf{M} \mathbf{u} = 1 \Rightarrow \mathbf{u}^T \mathbf{M} \frac{d\mathbf{u}}{dx_i} = -\frac{1}{2} \mathbf{u}^T \frac{d\mathbf{M}}{dx_i} \mathbf{u} \quad (2)$$

Pré-multiplicando (1) por  $\mathbf{u}^T$ :

$$\mathbf{u}^T (\mathbf{K} - \mu \mathbf{M}) \frac{d\mathbf{u}}{dx_i} - \frac{d\mu}{dx_i} \mathbf{u}^T \mathbf{M} \mathbf{u} = -\mathbf{u}^T \left( \frac{d\mathbf{K}}{dx_i} - \mu \frac{d\mathbf{M}}{dx_i} \right) \mathbf{u} \Rightarrow \frac{d\mu}{dx_i} = \frac{\mathbf{u}^T \left( \frac{d\mathbf{K}}{dx_i} - \mu \frac{d\mathbf{M}}{dx_i} \right) \mathbf{u}}{\mathbf{u}^T \mathbf{M} \mathbf{u}}$$

Sensibilidade de um autovalor

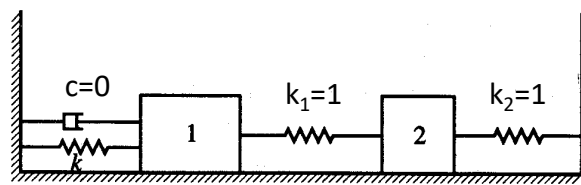
Emilio Silva e Renato Picelli

## Sensibilidade de Autovalores e Autovetores <sup>41</sup>

A sensibilidade dos autovetores é obtida combinando-se as equações (1) e (2) e resolvendo o sistema:

$$\text{Singular} \rightarrow \left\{ \begin{array}{cc} (\mathbf{K} - \mu \mathbf{M}) & -\mathbf{M}\mathbf{u} \\ -\mathbf{u}^T \mathbf{M} & 0 \end{array} \right\} \begin{Bmatrix} \frac{d\mathbf{u}}{dx_i} \\ \frac{d\mu}{dx_i} \end{Bmatrix} = \begin{Bmatrix} -\left( \frac{d\mathbf{K}}{dx_i} - \mu \frac{d\mathbf{M}}{dx_i} \right) \mathbf{u} \\ \frac{1}{2} \mathbf{u}^T \frac{d\mathbf{M}}{dx_i} \mathbf{u} \end{Bmatrix}$$

Exemplo



Calcular a sensibilidade da primeira frequência de ressonância e modo de vibrar em relação ao valor de k.

Emilio Silva e Renato Picelli

## Exemplo

42

$$\mathbf{K} = \begin{bmatrix} 1+k & -1 \\ -1 & 2 \end{bmatrix}; \mathbf{M} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \Rightarrow \begin{bmatrix} 2-\omega^2 & -1 \\ -1 & 2-\omega^2 \end{bmatrix} \begin{Bmatrix} u_1 \\ u_2 \end{Bmatrix} = 0 \Rightarrow \omega_1 = 1 \text{ e } \omega_2 = \sqrt{3}$$

Para o primeiro modo de vibrar:

$$\begin{aligned} u_1 - u_2 = 0 \\ -u_1 + u_2 = 0 \end{aligned} \text{ e } \mathbf{u}^T \mathbf{M} \mathbf{u} = u_1^2 + u_2^2 = 1 \Rightarrow \mathbf{u} = \frac{\sqrt{2}}{2} \begin{Bmatrix} 1 \\ 1 \end{Bmatrix} \text{ e } \mathbf{K}' = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}; \mathbf{M}' = \mathbf{0}$$

Assim, a sensibilidade da primeira frequência vale:  $\frac{d(\omega^2)}{dx_i} = \mathbf{u}^T \mathbf{K}' \mathbf{u} = 0,5$

Sensibilidade do primeiro modo de vibrar:

$$(\mathbf{K} - \mu \mathbf{M}) = \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}; \mathbf{M} \mathbf{u} = \frac{\sqrt{2}}{2} \begin{Bmatrix} 1 \\ 1 \end{Bmatrix}; -(\mathbf{K}' - \mu \mathbf{M}') \mathbf{u} = \frac{\sqrt{2}}{2} \begin{Bmatrix} -1 \\ 0 \end{Bmatrix}; \frac{1}{2} \mathbf{u}^T \mathbf{M}' \mathbf{u} = 0$$

$$\begin{bmatrix} 1 & -1 & -\sqrt{2}/2 \\ -1 & 1 & -\sqrt{2}/2 \\ -\sqrt{2}/2 & -\sqrt{2}/2 & 0 \end{bmatrix} \begin{Bmatrix} u'_1 \\ u'_2 \\ \mu' \end{Bmatrix} = \begin{Bmatrix} -\sqrt{2}/2 \\ 0 \\ 0 \end{Bmatrix} \Rightarrow \mathbf{u} = \begin{Bmatrix} u'_1 \\ u'_2 \end{Bmatrix} = \begin{Bmatrix} -\sqrt{2}/8 \\ \sqrt{2}/8 \end{Bmatrix}; \mu' = 1/2$$

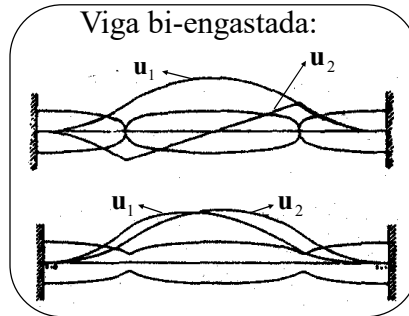
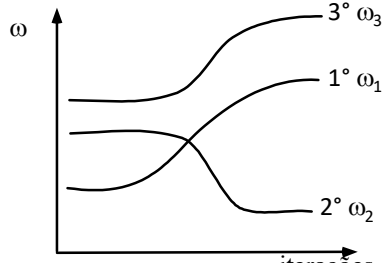
Deve se tomar cuidado quando o autovalor for repetido (possue multiplicidade)

Emilio Silva e Renato Picelli

# Inversão de modos durante a otimização

43

Exemplo:



Possíveis soluções:

$$\begin{aligned} & \text{Max } \omega_1^2 \\ & \text{tal que } (\mathbf{K} - \omega^2 \mathbf{M})\mathbf{u} = \mathbf{0} \\ & \mathbf{u}^T \mathbf{M} \mathbf{u} = 1 \\ & \omega_1 > \omega_0; \omega_2 > \omega_0; \\ & \omega_3 > \omega_0; \text{etc...} \\ & V - V_{\max} \leq 0 \end{aligned}$$

$$\begin{aligned} & \text{Max } (\omega_1^2 - \omega_0^2) + (\omega_2^2 - \omega_0^2) + (\omega_3^2 - \omega_0^2) + \dots \\ & \text{tal que } (\mathbf{K} - \omega^2 \mathbf{M})\mathbf{u} = \mathbf{0} \\ & \mathbf{u}^T \mathbf{M} \mathbf{u} = 1 \\ & V - V_{\max} \leq 0 \end{aligned}$$

Emilio Silva e Renato Picelli

# Problemas Clássicos

44

Maximização da Frequência de Ressonância

Em geral, primeira frequência

$$\begin{aligned} & \text{Max } \omega^2 \\ & \text{tal que } (\mathbf{K} - \mu \mathbf{M})\mathbf{u} = \mathbf{0} \quad (\mu = \omega^2) \quad \mathbf{u}^T \mathbf{M} \mathbf{u} = 1 \\ & \mathbf{x} - \mathbf{x}_{\min} \geq 0 \\ & V - V_{\max} \leq 0 \end{aligned}$$

Problema de autovalor e autovetor

Normalização

Problemas equivalentes:

$$\begin{aligned} & \text{Max } \beta \\ & \text{tal que } (\mathbf{K} - \mu \mathbf{M})\mathbf{u} = \mathbf{0} \text{ e } \mathbf{u}^T \mathbf{M} \mathbf{u} = 1 \\ & \mu \geq \beta \\ & \mathbf{x} - \mathbf{x}_{\min} \geq 0 \\ & V - V_{\max} \leq 0 \end{aligned}$$

$$\begin{aligned} & \text{Min } V \\ & \text{tal que } (\mathbf{K} - \mu \mathbf{M})\mathbf{u} = \mathbf{0} \text{ e } \mathbf{u}^T \mathbf{M} \mathbf{u} = 1 \\ & \mathbf{x} - \mathbf{x}_{\min} \geq 0 \\ & \mu > \mu_0 \end{aligned}$$

Emilio Silva e Renato Picelli