

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

cuja solução obtida usando o comando QUAPRO do SCILAB é:

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

onde α é obtido minimizando-se $\psi(\alpha) = f(\alpha) + \mu_1 g_1(\alpha)$ e:

$$\mu_1 = \max\left(\lambda_1, \frac{1}{2}(|\lambda_1| + \mu_1^{\text{velho}})\right) = 14,31 \Rightarrow \alpha = 0,5 \Rightarrow \mathbf{x}^{(2)} = (9,30; 8,86)^T;$$

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

portanto foi feito um bom progresso em direção à solução ótima que vale $\mathbf{x}^* = (9,46; 9,46)^T$.

4.6 Algoritmos de Busca Probabilísticos

Os algoritmos baseados em técnicas de programação matemática e métodos sequenciais vistos nas seções anteriores, tem como principal desvantagem o fato de não garantirem que foi obtido um mínimo global (a menos que o problema seja convexo) e em geral ficam “presos” em mínimos locais. Uma tentativa de se atingir o mínimo global consiste em alterar o valor inicial.

Outra desvantagem é que não permitem lidar com variáveis discretas. Nesse caso o espaço de solução é disjunto e desconexo e as informações de derivadas são inúteis ou não definida. Além disso, um espaço de variáveis discretas introduz múltiplos mínimos locais (várias combinações de variáveis), mesmo que o problema tenham um mínimo global no caso de variáveis contínuas.

Uma forma de lidar com ambos os problemas é utilizar algoritmos que realizam buscas randômicas que percorrem todo o espaço de solução. Na engenharia, esses algoritmos tem sido bastante aplicados, entre outras áreas, na área de engenharia de produção para resolver problemas de organização da manufatura, que em geral envolvem variáveis discretas.

Existem vários algoritmos probabilísticos na literatura, no entanto dois que recentemente emergiram como ferramentas apropriadas para solucionar problemas de otimização de engenharia em que se deseja encontrar o mínimo global são o “Simulated Annealing” e o Algoritmos Genéticos. Ambos foram inspirados em fenômenos naturais, são baseados em processos de seleção randômica guiados por decisões probabilísticas, permitem lidar com variáveis discretas e necessitam apenas do valor da função objetivo (não são necessárias derivadas, etc..). Esses algoritmos serão descritos a seguir.

No entanto, devemos ter cuidado com os algoritmos probabilísticos. O aluno deve estar se perguntando qual a vantagem de utilizar os algoritmos de programação matemática e métodos sequenciais em relação aos algoritmos probabilísticos?? Basta converter um problema contínuo em um problema discreto e utilizar um algoritmo probabilístico para encontrar o mínimo global

do problema. Na verdade não é tão simples assim. Primeiro, que o custo computacional aumenta consideravelmente com o número de variáveis de projeto, em relação aos algoritmos de programação matemática, deteriorando a eficiência da solução, o que é crítico em problemas que desejamos resolver vários problemas de otimização. Segundo, que embora esses algoritmos tenham mais chance de obter o mínimo global, não há como provar que realmente encontraram-no, o que significa que também há o risco de obterem apenas mínimos locais. Finalmente, um problema menos óbvio é que muitos problemas quando definidos no espaço contínuo apresentam várias soluções (ou não apresentam solução) devido ao problema dos múltiplos mínimos locais que surgem no espaço de solução discreto, ou seja, a solução é oscilante entre os vários mínimos locais, o que não acontece no caso contínuo. Uma tentativa de se resolver isso é introduzir técnicas que limitem o espaço de solução, no entanto, essas técnicas podem acabar por eliminar o mínimo global do espaço de solução (o que afinal motivou a utilização do algoritmo probabilístico). Isso ocorre por exemplo com o problema de otimização topológica que será visto adiante, onde os algoritmos probabilísticos não obtiveram sucesso na sua solução.

Assim é importante deixar claro que no caso de problemas discretos os algoritmos probabilísticos são indiscutivelmente úteis. Já no caso de problemas contínuos, há várias publicações na literatura comparando a utilização de algoritmos genéticos com algoritmos baseados em programação matemática para solução de problemas de otimização em engenharia mecânica, no entanto, o que se observa é que o algoritmo adotado dependerá de cada problema em específico. Dessa forma antes de adotar o uso de um algoritmo probabilístico na solução de um problema de variáveis contínuas, o aluno deve ser criterioso e verificar na literatura qual a tendência de algoritmo usado e a natureza do problema.

4.6.1 “Simulated Annealing”

O desenvolvimento do “Simulated Annealing” (ou recozimento simulado) é baseado num fenômeno de mecânica estatística relacionado com o equilíbrio de um grande número de átomos em sólidos e líquidos numa certa temperatura. Esse fenômeno é observado na solidificação de metais, onde um número de estados sólidos com diferentes estruturas atômicas internas que correspondem a diferentes níveis de energia que podem ser atingidos dependendo da taxa de resfriamento. Assim se o resfriamento é rápido, o estado sólido resultante tende a ter uma pequena margem de estabilidade porque os átomos irão assumir posições na estrutura cristalina de forma a atingir um nível de energia que corresponde a um mínimo local. Para atingir um nível de energia mais estável (correspondente a um mínimo global), o processo de recozimento é usado onde o metal é reaquecido até uma alta temperatura e resfriado lentamente, dando tempo suficiente aos átomos para encontrar posições que minimizem a energia potencial.

O algoritmo computacional que simula o processo de recozimento simulado é denominado “Metropolis”. Dada uma temperatura T , o algoritmo perturba a posição do átomo randomicamente e calcula a variação de energia do sistema ΔE . Se $\Delta E < 0$, a nova configuração de átomos é aceita. Por outro lado se $\Delta E \geq 0$, a nova configuração de átomos (estado) é aceita ou rejeitada baseada na seguinte função probabilística que calcula a probabilidade de aceitação:

$$P(\Delta E) = e^{\left(\frac{-\Delta E}{k_B T}\right)}$$

onde k_b é a constante de Boltzmann. Note que se T é alto $P(\Delta E)$ é próximo de um. Por outro lado se T é próximo de zero a probabilidade de aceitação é muito pequena.

A decisão de aceitação ou rejeição é obtida escolhendo-se de forma randômica um número ρ em $(0,1)$ e comparando-o com $P(\Delta E)$. Se $\rho < P(\Delta E)$ então a nova configuração é aceita. Se $\rho > P(\Delta E)$ então a nova configuração é rejeitada. Assim a cada temperatura um conjunto de estruturas atômicas seria gerada pela perturbação randômica da posição até que o estado de energia estável ou o chamado “equilíbrio térmico” seja atingido. 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.

Na aplicação do recozimento simulado na solução de problemas matemáticos de otimização é feita uma analogia. Os estados de energia são equivalentes ao valor da função objetivo, as configurações dos átomos na matriz do metal são equivalentes às variáveis de projeto do problema de otimização, e a temperatura T consiste num parâmetro que controla a convergência. O método utiliza apenas o valor da função objetivo. O desempenho do método é afetado pela escolha da temperatura inicial T_0 , e como atualizá-la, bem como pelo número de iterações (combinações de variáveis de projeto) necessárias para atingir o equilíbrio térmico antes de reduzir T . Esses parâmetros são denominados “cronograma de resfriamento” ou “cooling schedule”.

Se um valor baixo de T_0 é usado, o algoritmo tem baixa probabilidade de atingir o mínimo global (probabilidade de aceitar o movimento inicial é baixa). O valor inicial de T_0 deve ser alto o suficiente para permitir que todo movimento do espaço de projeto seja aceitável de forma que quase uma busca randômica seja realizada. No entanto um T_0 muito alto aumenta o tempo computacional. Dessa forma a escolha de T_0 é dada por:

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

O fluxograma do método é mostrado na figura abaixo:

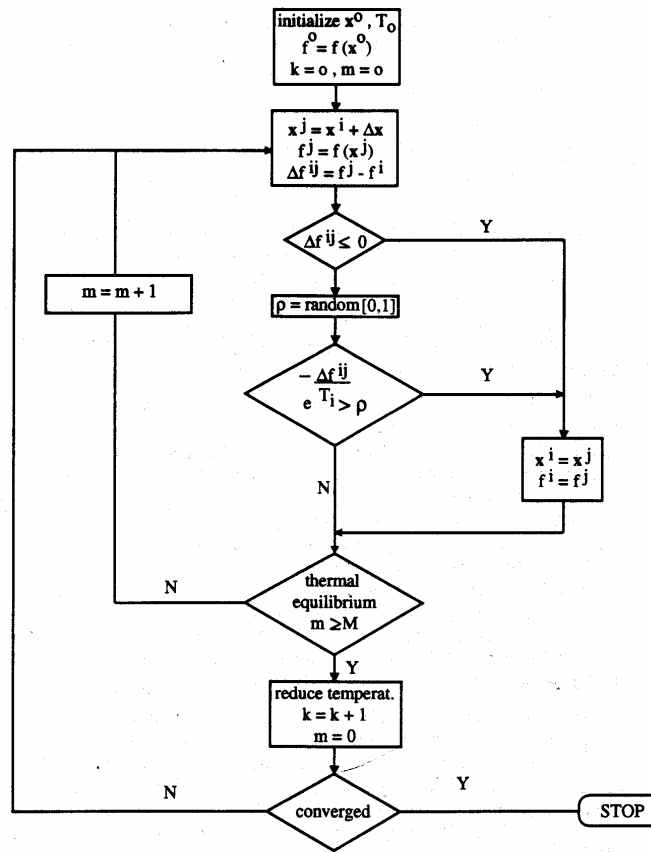


Fig. 4.6.1 - Fluxograma do método “Simulated Annealing”

Com relação à variação da temperatura, várias regras são propostas. Uma regra muito usada é a taxa de resfriamento constante dada por:

Também é possível dividir o intervalo $[0, T_0]$ em um número fixo de passos K e usar:

$$T_{k+1} = \alpha T_k \quad k = 0, 1, 2, \dots, K \quad 0,5 \leq \alpha \leq 0,95$$

$$T_k = \frac{K - k}{K} T_0 \quad k = 0, 1, 2, \dots, K$$

Em geral K varia de 5 à 20.

4.6.2 Algoritmos Genéticos

Algoritmos Genéticos usam técnicas derivadas da biologia e baseiam-se no princípio da teoria de Darwin relacionada com a sobrevivência do mais resistente. Ao longo das gerações, características que são úteis para sobrevivência são passadas adiante para os indivíduos sucessores, porque indivíduos que as possuem tem mais chance de sobreviver. Essas características são armazenadas na codificação dos cromossomos. Os mecanismos da genética são baseados em operações que permitem a troca randômica de

informações entre os cromossomos dos pais reprodutores. Entre essas operações temos: reprodução, cruzamento, mutação ocasional e inversão do código cromossômico.

Algoritmos Genéticos simulam os mecanismos da genética natural para solução de problemas matemáticos de otimização multivariável. O código cromossômico é representado por uma “palavra”. Dessa forma no problema matemático a representação da combinação de variáveis de projeto deve ser feita em termos de palavras de bits que “emulam” os cromossomos. Por exemplo:

$$\{x_1, x_2, x_3, x_4\} = \{6, 5, 3, 11\} = \underbrace{01101011}_{x_1} \underbrace{111011}_{x_2} \underbrace{11011}_{x_3} \underbrace{11011}_{x_4}$$

Nos algoritmos genéticos as operações envolvem trocas randômicas das localizações de números numa palavra. O desempenho da “palavra” ou código do cromossomo (combinação de genes) é avaliado calculando-se somente o valor da função objetivo para aquela combinação.

Algoritmos genéticos são ideais para variáveis discretas ou inteiras, Entretanto, no caso de variáveis contínuas é necessário um grande número de bits para representação das variáveis dependendo da precisão desejada. Assim por exemplo, o 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:

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

Ao contrário dos algoritmos baseados em programação matemática, algoritmos genéticos trabalham com população de palavras (ou códigos de cromossomos) e não com um único ponto do espaço de projeto. Por essa razão algoritmos genéticos tem mais chance de obter o ótimo global, uma vez que o algoritmo mantém os pontos de mínimo local e global numa “cesta” durante o processo de busca ao invés de ir de um ponto para outro correndo o risco de ficar num ponto de mínimo local. O fato de trabalhar com uma população de pontos sugere uma implementação em computadores paralelos com grandes vantagens. O resultado do algoritmo genético é uma população de bons projetos (“palavras”) ao invés de um único ponto como solução, o que pode ser bastante útil para o projetista.

A seguir é descrito um exemplo de seqüência de operações de um algoritmo genético:

1. A dimensão da população é escolhida e o valor das variáveis em cada “palavra” é decidido randomicamente associando 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 múltiplas cópias dos indivíduos mais resistentes.
3. Cruzamento: “palavras” da nova população são agrupadas randomicamente em pares para o cruzamento. Um inteiro randômico k é selecionado entre 1 e $L-1$,

onde L é o comprimento da palavra, e então duas novas palavras são geradas trocando-se os 0s e 1s que vem após a posição k no primeiro cromossomo “pai” com as localizações correspondentes no segundo cromossomo “pai”. Por exemplo, se $L=9$ e $k=5$ então:

pai 1: 01101||0111

pai 2: 01001||0001

geram

palavra gerada 1: 01101||0001

palavra gerada 2: 01001||0111

O cruzamento ilustrado é denominado cruzamento de “um ponto”, pois somente um ponto de divisão é considerado. Outras possibilidades são o cruzamento de “dois pontos” e o “multi-ponto”.

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

5. SENSIBILIDADE EM OTIMIZAÇÃO

Como visto anteriormente todos os métodos (com exceção dos métodos de ordem zero) exigem o cálculo das derivadas primeiras (ou gradientes) da função objetivo ou restrições que são também denominadas *sensibilidades*. Dessa forma a sensibilidade deve ser calculada de forma precisa, caso contrário o método caminhará numa direção errada, não chegando na solução ótima.

Na verdade, a sensibilidade em si já nos fornece uma boa informação sobre comportamento da estrutura (ou um sistema). Ela nos permite analisar a mudança do comportamento da estrutura (ou um sistema) devido a pequenas mudanças de seus parâmetros (dimensões, propriedades de materiais, etc...) com custo menor do que realizar novas análises. Alguns pacotes comerciais de CAE apresentam um módulo somente para o cálculo de sensibilidade, assim o engenheiro já pode tomar algumas decisões preliminares de projeto.

Assim, devido às razões acima, o estudo do cálculo da sensibilidade é extremamente importante, merecendo um capítulo em todo curso de otimização.

Existem três métodos para o cálculo de sensibilidade: método de diferenças finitas (MDF), método semi-analítico e método analítico. A precisão de cálculo aumenta do primeiro para o último e a utilização de um ou outro vai depender da informação disponível sobre a expressão matemática da função objetivo.

5.1 Aproximação por Diferenças Finitas

Para analisarmos o conceito da utilização de diferenças finitas no cálculo da sensibilidade, consideremos a sensibilidade de uma função unidimensional du/dx . Utilizando-se somente dois pontos dessa função existem três métodos para calcular du/dx :

- Diferença Progressiva:

$$\frac{du}{dx} \cong \frac{u(x + \Delta x) - u(x)}{\Delta x} + O(\Delta x)$$

onde n análises adicionais são necessárias para calcular $u(x + \Delta x)$.

- Diferença Regressiva:

$$\frac{du}{dx} \cong \frac{u(x) - u(x - \Delta x)}{\Delta x} + O(\Delta x)$$

onde também n análises adicionais são necessárias para calcular $u(x - \Delta x)$.

- Diferença Central:

$$\frac{du}{dx} \cong \frac{u(x + \Delta x) - u(x - \Delta x)}{2\Delta x} + O(\Delta x^2)$$

onde $2n$ análises adicionais são necessárias (para calcular $u(x + \Delta x)$ e $u(x - \Delta x)$).

Como estamos usando aproximações para o cálculo de du/dx , é necessário realizar um estudo do erro na utilização dessas expressões. Note que a diferença central apresenta o menor erro das três (da ordem de Δx^2), mas envolve o maior custo computacional ($2n$ análises). Existem outras fórmulas que permitem calcular du/dx com uma precisão melhor, no entanto envolvem o cálculo de mais pontos também ($u(x+2\Delta x)$, $u(x+3\Delta x)$, etc.) e portanto o custo computacional aumenta.

O erro total na utilização dessas fórmulas é composto de dois tipos de erro: erro de truncamento e erro de condicionamento (ou arredondamento).

O erro de truncamento resulta da aproximação da fórmula da derivada du/dx utilizando a série de Taylor. Esse erro aumenta com o aumento de Δx . Vejamos um exemplo da sua estimativa usando uma série de Taylor progressiva:

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

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

Analogamente para a diferença central: $e_T(\Delta x) = \frac{\Delta x^2}{6} \frac{d^3u(\xi)}{dx^3}$

Já o erro de condicionamento (ou arredondamento) depende da precisão do computador, não sendo uma função contínua de Δx , diminuindo com o aumento de Δx . Esse erro se torna importante em métodos iterativos em que uma mesma operação é feita de forma repetitiva tornando esse erro significativo. Uma estimativa desse erro é dada por:

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

onde ε_u é o limite do erro absoluto no cálculo da função u .

Assim, a estimativa total do erro é:

$$e = \frac{\Delta x}{2} |s_b| + \frac{2}{\Delta x} \varepsilon_u$$

onde $|s_b|$ é o limite no cálculo da segunda derivada. A fórmula acima nos impõe um dilema na escolha do tamanho do passo Δx ("dilema do tamanho do passo"). Para Δx alto predomina o erro de truncamento e para Δx baixo predomina o erro de condicionamento. A Fig.5.1.1 ilustra esse fenômeno.

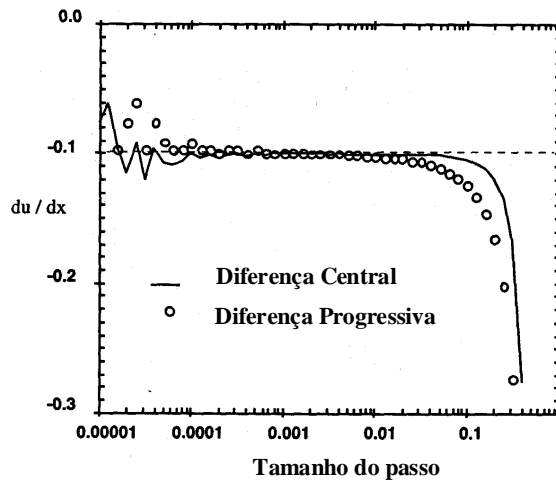


Fig.5.1.1: Efeito do passo Δx no cálculo da derivada.

A princípio utilizando a fórmula acima podemos calcular um valor ótimo para o passo que resulte em erro nulo:

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

no entanto não se consegue estimar os valores de $|S_b|$ e ε_u .

Outro efeito que deve ser levado em conta na precisão do cálculo da derivada é a magnitude da derivada. Pequenas derivadas são calculadas com menos precisão do que grandes derivadas. A magnitude da derivada pode ser estimada pela expressão abaixo denominada derivada logarítmica:

$$\frac{d_L u}{dx} = \frac{d(\log u)}{d(\log x)} = \frac{du/u}{dx/x}$$

Essa expressão fornece a variação percentual de u devido à variação percentual de x .

Se $\frac{d_L u}{dx} > 1 \Rightarrow \frac{du}{u} > \frac{dx}{x}$, ou seja, a variação relativa de u é maior do que a variação relativa de x , e portanto o valor da derivada logarítmica é alto, indicando que a derivada du/dx é fácil de calcular com precisão.

Se $\frac{d_L u}{dx} \lll 1 \Rightarrow \frac{du}{u} \lll \frac{dx}{x}$, ou seja, a variação relativa de u é menor do que a variação relativa de x , e portanto o valor da derivada logarítmica é pequeno, indicando que a derivada du/dx é difícil de calcular com precisão. No entanto, em geral, quando isso ocorre não é importante pois tem pequena influência no processo de otimização (próxima de zero).

Se $u, x \cong 0 \Rightarrow \frac{d_{Lm} u}{dx} = \frac{du/u_t}{dx/x_t}$, onde u_t, x_t são valores representativos de u e x .

Vejamos um exemplo. Considere o cálculo da sensibilidade da função objetivo do problema de otimização abaixo:

Min $f(\mathbf{x}) = u(\mathbf{x}) = x_1 + x_2$
 x_1, x_2
 tal que $g(\mathbf{x}) = \frac{1}{x_1 x_2^2} - 1 = 0$

Solução:

Temos que $u(\mathbf{x}) = \frac{1}{x_2} + x_2 \Rightarrow \frac{du}{dx_2} = -\frac{2}{x_2^3} + 1$. Essa é a solução exata para efeito de comparação.

Calculando a derivada em dois pontos, temos:

- Em $x_2 = 1 (\Delta x_2 = 0,01) \Rightarrow \frac{\Delta u}{\Delta x_2} = \frac{1/1,01^2 + 1,01 - 2}{0,01} = -0,97$, uma diferença de 3% em relação ao valor exato $\frac{du}{dx_2} = 1,0$;
- Em $x_2 = 1,29 (\Delta x_2 = 0,01) \Rightarrow \frac{\Delta u}{\Delta x_2} = \frac{1/1,3^2 + 1,3 - (1/1,29^2 + 1,29)}{0,01} = -0,0791$, uma diferença de 16% em relação ao valor exato $\frac{du}{dx_2} = 0,0683$;

Calculando as derivadas logarítmicas correspondentes temos:

- Em $x_2 = 1 \Rightarrow f = 2 \Rightarrow \frac{d_L u}{dx_2} = \frac{\Delta u}{\Delta x_2} \frac{x_2}{u} = -0,97 \times 1/2 = -0,485$;
- Em $x_2 = 1,29 \Rightarrow f = 1,891 \Rightarrow \frac{d_L u}{dx_2} = \frac{\Delta u}{\Delta x_2} \frac{x_2}{u} = 0,0791 \times 1,29/1,891 = 0,054$, que é muito pequena, indicando pouca precisão no cálculo como verificado.

5.2 Métodos Analíticos

Entre os métodos discutidos são os mais precisos, pois fornecem o valor exato da sensibilidade. Considere uma função objetivo ou restrição $f(\mathbf{u}, x)$ que dependa dos deslocamentos da estrutura \mathbf{u} e da variável de projeto x . A sensibilidade de f é dada por:

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

A expressão acima é composta de duas partes:

- uma parte explícita $\frac{\partial f}{\partial x}$ que geralmente é zero;
- uma parte implícita $\mathbf{z}^T \frac{d\mathbf{u}}{dx}$;

Para entendermos melhor o significado matemático das partes acima, consideremos o exemplo a seguir.

Seja a função dada por $f(\mathbf{u}, x) = x u_1 + u_2^2 - \ln(u_3) + x^2$, onde $\mathbf{u}^T = (u_1 \quad u_2 \quad u_3)$.

A parte explícita vale: $\frac{\partial f}{\partial x} = u_1 + 2x$.

A parte implícita é dada por:

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

Assim a derivada total vale: $\frac{df}{dx} = \frac{\partial f}{\partial x} + \mathbf{z}^T \frac{d\mathbf{u}}{dx} = u_1 + 2x + x \frac{du_1}{dx} + 2u_2 \frac{du_2}{dx} - \frac{1}{u_3} \frac{du_3}{dx}$

Note que para obter o valor final da sensibilidade de f é necessário obter a sensibilidade dos deslocamentos que dependem implicitamente das variáveis de projeto (daí o nome parte implícita). A parte explícita geralmente é nula porque raramente temos a função objetivo como função direta das variáveis de projeto.

Para obter a sensibilidade dos deslocamentos existem dois métodos clássicos: o método direto e o método adjunto que serão descritos a seguir.

5.2.1 Método Direto

Considere um sistema linear, ou seja, um sistema cuja resposta é obtida resolvendo-se uma equação linear do tipo $\mathbf{A}\mathbf{x}=\mathbf{b}$. Assim, por exemplo, no caso de um problema estrutural linear, os deslocamentos são obtidos resolvendo-se a equação:

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

onde \mathbf{u} é o vetor de deslocamentos nodais, \mathbf{K} é a matriz de rigidez e \mathbf{F} é o vetor de carregamento.

Derivando-se a equação acima em relação à variável de projeto, temos:

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

onde o sistema acima deve ser resolvido para cada variável de projeto. Calculando a parte implícita, a derivada total fica:

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

5.2.2 Método Adjunto

Considere o sistema de equação abaixo onde o vetor de carregamento é dado pelo vetor \mathbf{z} .

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

Considerando a definição de \mathbf{z} acima, esse sistema deve ser resolvido para cada restrição e para a função objetivo. Calculando a parte implícita, a derivada total fica:

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

A princípio foi obtido o mesmo resultado que com o método direto, mas do ponto de vista computacional existem diferenças. O método direto é mais eficiente quando o número de variáveis de projeto é menor do que o número de restrições. Já o método adjunto é mais eficiente quando número de variáveis de projeto é maior do que o número de restrições.

Na prática devemos considerar vários objetivos (função multiobjetivo) e portanto casos de carga. No método direto o esforço será proporcional ao número de casos de carga (ver equação acima). Já o número de restrições críticas, por outro lado, é menor do que o número de variáveis de projeto. Assim, no caso de uma função com vários objetivos (função multiobjetivo) recomenda-se usar o método adjunto.

Vejamos o exemplo a seguir em que a sensibilidade é calculada usando os dois métodos. Considere a viga mostrada na Fig.5.2.1. Deseja-se calcular a sensibilidade da restrição de deslocamento na ponta da viga (ponto 3) em relação à variável de projeto momento de inércia I_1 ($\frac{dg}{dI_1}$).

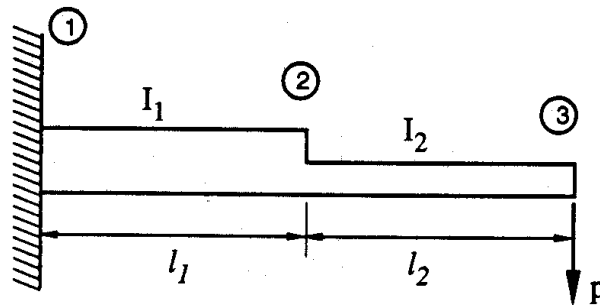


Fig.5.2.1: Viga usada no exemplo.

Restrição de deslocamento na ponta: $g = c - w_3 \geq 0$

Solução:

Inicialmente, devemos obter os deslocamentos nodais da viga usando a formulação de MEF para vigas. Assim, montando e resolvendo o sistema:

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

$$\mathbf{K} = E \begin{bmatrix} 12(I_1/l_1^3 + I_2/l_2^3) & -6(I_1/l_1^2 - I_2/l_2^2) & -12I_2/l_2^3 & 6I_2/l_2^2 \\ & 4(I_1/l_1 + I_2/l_2) & -6I_2/l_2^2 & 2I_2/l_2 \\ & & sim. & 12I_2/l_2^3 \\ & & & -6I_2/l_2^2 \\ & & & 4I_2/l_2 \end{bmatrix}$$

$$\mathbf{u} = \begin{Bmatrix} w_2 \\ \theta_2 \\ w_3 \\ \theta_3 \end{Bmatrix}; \mathbf{F} = \begin{Bmatrix} 0 \\ 0 \\ p \\ 0 \end{Bmatrix}$$

obtemos o vetor de deslocamento \mathbf{u} :

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

Vejamos agora o cálculo da sensibilidade por diferentes métodos:

1º Método: Método Direto

Utilizando \mathbf{u} calculado acima:

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

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

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

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

Note que a derivada $\frac{d\mathbf{K}}{dI_1}$ apresenta zeros em todos os elementos da matriz, menos nos elementos correspondentes aos graus de liberdade do elemento 1. Isso ocorre devido à

formulação do MEF, em que a matriz global é uma montagem de submatrizes de cada elemento finito. Assim ao se calcular a derivada da matriz \mathbf{K} em relação a uma variável de projeto x_i associada a um elemento i , somente os termos da submatriz daquele elemento contribuem para a matriz $\frac{d\mathbf{K}}{dx_i}$. Essa simplificação é válida para análises

lineares em MEF e representam uma grande vantagem do ponto de vista computacional. Em análises não-lineares a matriz \mathbf{K} depende de \mathbf{u} , e a simplificação acima não ocorre. Nesse caso, a abordagem semi-analítica apresentada adiante é a mais indicada.

2º Método: Método Adjunto

Utilizando \mathbf{u} , \mathbf{z} e $\frac{d\mathbf{K}}{dI_1}$ calculados acima:

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

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

3º Método: Derivação da equação de w_3

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

Logicamente, os métodos levam ao mesmo resultado, sendo a única diferença a sua eficiência computacional. O último método foi usado somente para efeito de comparação, mostrando que os demais resultados são iguais ao valor exato, já que em geral, a equação explícita da restrição em função das variáveis de projeto não está disponível.

5.3 Métodos Semi-Analíticos

Essencialmente, seguem o mesmo procedimento de cálculo apresentado nos métodos analíticos com a diferença que a derivada da matriz rigidez $\frac{d\mathbf{K}}{dx_i}$ é calculada usando diferenças finitas, ou seja:

$$\frac{d\mathbf{K}}{dx} \cong \frac{\mathbf{K}(x + \Delta x) - \mathbf{K}(x)}{\Delta x}$$

A sensibilidade é então obtida usando os métodos direto ou adjunto descritos acima. No entanto, em algumas situações a sensibilidade calculada usando o método semi-analítico (SA) pode curiosamente apresentar um erro maior do que a calculada por diferenças finitas (DF) descrito na seção 5.1. A Fig.5.3.1 mostra o modelo de CAE de um carro usando elementos de viga.

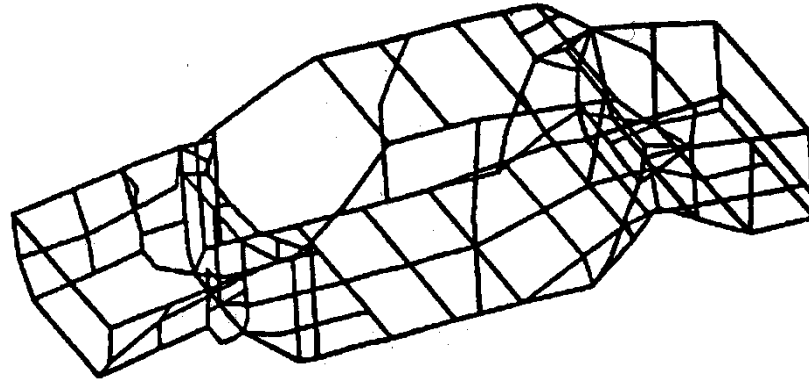


Fig.5.3.1: Modelo de MEF de um carro usando elementos de viga.

O método SA foi usado com sucesso para calcular a sensibilidade da matriz rigidez em relação a todas as áreas de seção e a maior parte das variáveis de projeto geométricas. No entanto, algumas derivadas em relação às dimensões de comprimento do carro, apresentaram sérios problemas de precisão no seu cálculo. A Fig.5.3.2 ilustra esse fenômeno.

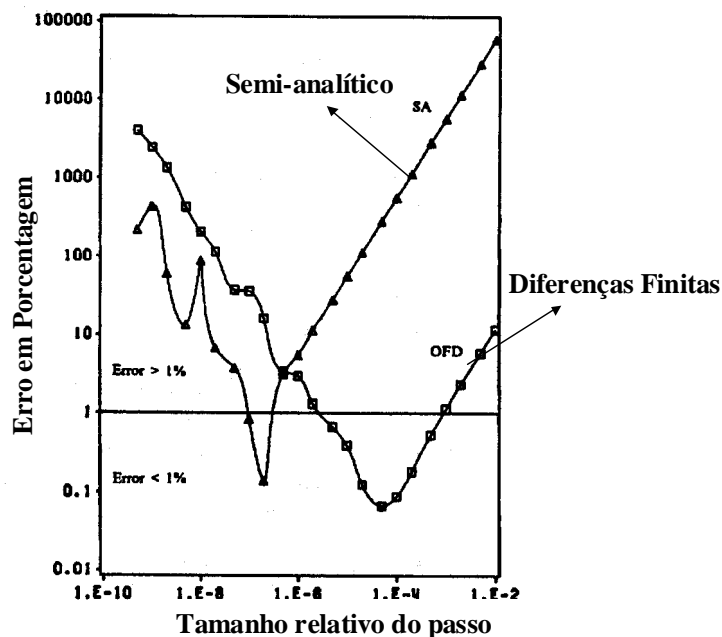


Fig.5.3.2: Erro no método semi-analítico pode ser maior do que o erro obtido com diferenças finitas.

No problema ilustrado pela Fig.5.3.2 para valores pequenos de tamanho de passo, o método SA apresenta um erro menor do que diferenças finitas (DF). A banda de valores de passos para o qual o erro é menor do que 1% é maior para a DF do que para o SA. Para valores maiores de tamanho de passo, o método SA apresenta um erro maior do que (DF). Nesse mesmo problema, para outras variáveis, não foi possível encontrar um tamanho de passo que resultasse num valor

preciso da sensibilidade! Para entender porque isso pode acontecer, consideremos o cálculo de $\frac{dg}{dI_1}$ e $\frac{dg}{dl_1}$ no exemplo anterior usando os métodos DF e SA.

Solução:

Calculando $\frac{dg}{dI_1}$ por diferenças finitas e usando o vetor \mathbf{u} obtido anteriormente, temos:

$$\frac{dg}{dI_1} \cong -\frac{w_3(I_1 + \Delta I_1) - w_3(I_1)}{\Delta I_1} \Rightarrow e_T = -\frac{\partial^2 w_3}{\partial I_1^2} \frac{\Delta I_1}{2} = -\frac{p}{3EI_1^3} (l_1^3 + 3l_1^2 l_2 + 3l_1 l_2^2) \Delta I_1 \Rightarrow \frac{e_T}{\frac{dg}{dI_1}} = -\frac{\Delta I_1}{I_1}$$

onde e_T é o erro de truncamento da série de Taylor.

Para $\frac{dg}{dl_1}$ o erro vale: $e_T = -\frac{\partial^2 w_3}{\partial l_1^2} \frac{\Delta l_1}{2} = -\frac{p}{EI_1} (l_1 + l_2) \Delta l_1 \Rightarrow \frac{e_T}{\frac{dg}{dl_1}} = -\frac{\Delta l_1}{l_1 + l_2}$

Agora, pelo método SA temos:

$$\frac{dg}{dI_1} \cong \lambda^T \frac{\mathbf{K}(I_1 + \Delta I_1) - \mathbf{K}(I_1)}{\Delta I_1} \mathbf{u} \Rightarrow e_T = \frac{\Delta I_1}{2} \lambda^T \frac{d^2 \mathbf{K}}{dI_1^2} \mathbf{u} = 0$$

onde o erro é calculado a partir da série de Taylor para a matriz \mathbf{K} . O erro é nulo porque \mathbf{K} é uma função linear de I_1 .

Mas para $\frac{dg}{dl_1}$: $e_T = \frac{\Delta l_1}{2} \lambda^T \frac{d^2 \mathbf{K}}{dl_1^2} \mathbf{u} = \frac{p \Delta l_1}{EI_1 l_1} (3l_1^2 + 7l_1 l_2 + 4l_2^2) \Rightarrow \frac{e_T}{\frac{dg}{dl_1}} = -\frac{(3l_1^2 + 7l_1 l_2 + 4l_2^2) \Delta l_1}{(l_1 + l_2)^2 l_1}$

Assim, se por exemplo: $l_1 = l_2 = l \Rightarrow e_T^{SA} = \frac{7 \Delta l}{2l}$ e $e_T^{DF} = \frac{\Delta l}{2l}$, ou seja, nesse caso o erro semi-analítico é 7 vezes maior do que o erro por diferenças finitas!!

5.4 Sensibilidade de Autovalores e Autovetores

Problemas de autovalores e autovetores estão presentes no cálculo da frequência de ressonância e cargas de flambagem. Como o cálculo da sensibilidade dessas quantidades é extremamente importante, essa seção se dedica a mostrar como calculá-los.

Para exemplificar a dedução teórica, considere a obtenção da frequência de ressonância e modos de vibrar representados pelo problema de autovalor e autovetor abaixo, onde \mathbf{K} é a matriz rigidez, \mathbf{M} é a matriz de massa e $\mu = \omega^2$ (frequência de ressonância):

$$(\mathbf{K} - \mu \mathbf{M}) \mathbf{u} = \mathbf{0}$$

que em geral é acompanhado de uma condição de normalização dos modos de vibrar:

$$\mathbf{u}^T \mathbf{M} \mathbf{u} = 1$$

Assim, como no caso estático apresentado na seção 5.2, vamos derivar as equações acima em relação a uma variável de projeto x_i .

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

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

Pré-multiplicando a primeira equação por \mathbf{u}^T , obtemos:

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

onde $\frac{d\mu}{dx_i}$ é a sensibilidade do autovalor (no caso frequência de ressonância). A sensibilidade do autovetor é obtida combinando-se as equações acima num sistema de equações mostrado abaixo, resolvendo-o:

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

A solução desse sistema permite a obtenção da sensibilidade dos autovetores e autovalores. No entanto deve-se tomar cuidado na sua solução pois a submatriz diagonal $(\mathbf{K} - \mu\mathbf{M})$ é singular.

Vejamos um exemplo. Vamos calcular a sensibilidade da primeira frequência de ressonância e modo de vibrar em relação ao valor de k no sistema mostrado na Fig. 5.4.1. Considere inicialmente o amortecimento nulo (c=0).

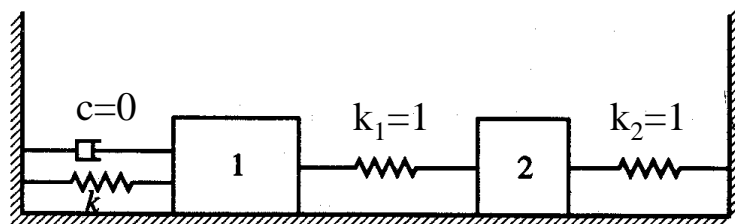


Fig.5.4.1: Sistema massa-amortecedor-mola usado para ilustrar o cálculo da sensibilidade da frequência de ressonância e modo de vibrar.

Solução:

As matrizes de massa \mathbf{M} e rigidez \mathbf{K} para esse sistema são dadas por:

$$\mathbf{K} = \begin{bmatrix} 1+k & -1 \\ -1 & 2 \end{bmatrix}; \mathbf{M} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Assim, o problema correspondente de autovalor e autovetor e sua solução vale:

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

O primeiro modo de vibrar é igual a:

$$\begin{cases} u_1 - u_2 = 0 \\ -u_1 + u_2 = 0 \end{cases} \text{ e } \mathbf{u}^T \mathbf{M} \mathbf{u} = u_1^2 + u_2^2 = 1 \Rightarrow \mathbf{u} = \frac{\sqrt{2}}{2} \begin{Bmatrix} 1 \\ 1 \end{Bmatrix}$$

As derivadas das matrizes \mathbf{K} e \mathbf{M} em relação à k são dadas por:

$$\mathbf{K}' = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}; \mathbf{M}' = \mathbf{0}$$

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

A sensibilidade do primeiro modo de vibrar é obtida montando-se e resolvendo o sistema de equações descrito acima. Assim:

$$\begin{aligned} (\mathbf{K} - \mu \mathbf{M}) &= \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}; \mathbf{M} \mathbf{u} = \frac{\sqrt{2}}{2} \begin{Bmatrix} 1 \\ 1 \end{Bmatrix}; -(\mathbf{K}' - \mu \mathbf{M}') \mathbf{u} = \frac{\sqrt{2}}{2} \begin{Bmatrix} -1 \\ 0 \end{Bmatrix}; \frac{1}{2} \mathbf{u}^T \mathbf{M}' \mathbf{u} = 0 \Rightarrow \\ \Rightarrow \begin{bmatrix} 1 & -1 & -\sqrt{2}/2 \\ -1 & 1 & -\sqrt{2}/2 \\ -\sqrt{2}/2 & -\sqrt{2}/2 & 0 \end{bmatrix} \begin{Bmatrix} u'_1 \\ u'_2 \\ \mu' \end{Bmatrix} &= \begin{Bmatrix} -\sqrt{2}/2 \\ 0 \\ 0 \end{Bmatrix} \end{aligned}$$

cuja solução vale: $\mathbf{u} = \begin{Bmatrix} u'_1 \\ u'_2 \end{Bmatrix} = \begin{Bmatrix} -\sqrt{2}/8 \\ \sqrt{2}/8 \end{Bmatrix}; \mu' = 1/2$

Deve-se tomar cuidado quando o problema apresentar multiplicidade de autovalores (ou seja, autovalores repetidos). Esse caso não será discutido aqui.

Incluindo o Amortecimento

Veamos agora o cálculo das sensibilidades considerando o amortecimento. O problema de autovalores e autovetores incluindo o amortecimento fica:

$$\mathbf{M} \bar{\mathbf{u}} + \mathbf{C} \bar{\mathbf{u}} + \mathbf{K} \bar{\mathbf{u}} = 0 \Rightarrow \bar{\mathbf{u}} = \mathbf{u} e^{\eta t} \Rightarrow (\eta^2 \mathbf{M} + \eta \mathbf{C} + \mathbf{K}) \mathbf{u} = \mathbf{0} \quad (\eta = i\omega)$$

Seguindo o mesmo procedimento anterior, a sensibilidade da frequência de ressonância é dada por:

$$\frac{d\eta}{dx_i} = - \frac{\eta^2 \mathbf{u}^T \frac{d\mathbf{M}}{dx_i} \mathbf{u} + \eta \mathbf{u}^T \frac{d\mathbf{C}}{dx_i} \mathbf{u} + \mathbf{u}^T \frac{d\mathbf{K}}{dx_i} \mathbf{u}}{2\eta \mathbf{u}^T \mathbf{M} \mathbf{u} + \mathbf{u}^T \mathbf{C} \mathbf{u}}$$

e a dos autovetores é obtido através de equação similar a anterior. Como exemplo, consideremos o mesmo problema anterior, onde agora deseja-se estimar a variação da frequência de ressonância do sistema não-amortecido pela adição de amortecimento ($c=0,2$ e $c=1,0$) e comparar com o valor exato.

Solução:

Usando a expressão acima, e sendo a variável de projeto $x_i=c$, temos:

$$\mathbf{M}' = 0; \mathbf{K}' = 0 \text{ e } \mathbf{C} = 0 \Rightarrow \frac{d(i\omega)}{dx_i} = -\frac{\eta\phi^T \frac{d\mathbf{C}}{dx_i} \phi}{2\eta\phi^T \mathbf{M} \phi} = -\frac{\phi^T \frac{d\mathbf{C}}{dx_i} \phi}{2\phi^T \mathbf{M} \phi} \text{ e } \phi = \sqrt{2}/2 \begin{Bmatrix} 1 \\ 1 \end{Bmatrix}$$

$$\mathbf{C} = \begin{bmatrix} c & 0 \\ 0 & 0 \end{bmatrix} \Rightarrow \frac{d\mathbf{C}}{dc} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}; \text{ e } \phi^T \mathbf{M} \phi = 1 \Rightarrow \frac{d\eta}{dc} = -0,5\phi^T \frac{d\mathbf{C}}{dc} \phi = -0,25$$

Assim, as estimativas das novas frequências para os diferentes valores de c, são:

$$\omega_1 = 1 \Rightarrow \eta = i \Rightarrow \eta_a = \eta|_{c=0} + \frac{d\eta}{dc} c = i - 0,25c \Rightarrow \begin{cases} c = 0,2 \Rightarrow \eta_a = i - 0,05 \\ c = 1,0 \Rightarrow \eta_a = i - 0,25 \end{cases}$$

os valores exatos são dados por:

$$\begin{bmatrix} \eta^2 + \eta c + 2 & -1 \\ -1 & \eta^2 + 2 \end{bmatrix} \begin{Bmatrix} u_1 \\ u_2 \end{Bmatrix} = 0 \Rightarrow \begin{cases} c = 0,2 \Rightarrow \eta_1 = 1,0013i - 0,05025 \\ c = 1,0 \Rightarrow \eta_1 = 1,0326i - 0,29178 \end{cases}$$

5.5 Sensibilidade da Solução Ótima

O objetivo dessa seção é estudar a sensibilidade da solução ótima em relação a um parâmetro da otimização, ou seja, o quanto a solução ótima varia alterando-se um dos parâmetros do problema.

Considere o problema de otimização abaixo.

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

cuja solução é $\mathbf{x}^*(p) \Rightarrow f^*(p) = f(\mathbf{x}^*(p), p)$, onde p é um parâmetro do problema. O objetivo é calcular $\frac{df^*}{dp}$ e $\frac{dg^*}{dp}$.

As equações que governam a solução ótima são a equação de estacionaridade:

$$\nabla f - \mathbf{N}\lambda = 0 \text{ e } n_{ij} = \frac{\partial g_j}{\partial x_i}, \quad j = 1 \dots r \text{ e } i = 1 \dots n$$

e as restrições ativas: $\mathbf{g}_a(\mathbf{x}^*, p) = 0$

Supõe-se que essas equações são satisfeitas por $\mathbf{x}^*(p)$ para todos os valores de p que não alterem as restrições ativas. Diferenciando as equações acima com relação a p, obtemos:

$$\begin{aligned} \frac{d(\nabla f - \mathbf{N}\lambda)}{dp} = 0 \Rightarrow \frac{\partial(\nabla f)}{\partial p} + \frac{d(\nabla f)}{d\mathbf{x}} \frac{d\mathbf{x}^*}{dp} - \left(\frac{\partial \mathbf{N}}{\partial p} + \frac{d\mathbf{N}}{d\mathbf{x}} \frac{d\mathbf{x}^*}{dp} \right) \lambda + \mathbf{N} \frac{d\lambda}{dp} = 0 \Rightarrow \\ (\mathbf{A} - \mathbf{Z}) \frac{d\mathbf{x}^*}{dp} - \mathbf{N} \frac{d\lambda}{dp} + \frac{\partial(\nabla f)}{\partial p} - \left(\frac{\partial \mathbf{N}}{\partial p} \right) \lambda = 0 \end{aligned}$$

onde: $a_{ij} = \frac{\partial^2 f}{\partial x_i \partial x_j}$; $z_{kl} = \sum_{j=1}^r \frac{\partial^2 g_j}{\partial x_k \partial x_l} \lambda_j$ e \mathbf{A} , portanto é a matriz Hessiana do sistema.

Diferenciando $\mathbf{g}_a(\mathbf{x}^*, p) = 0$ obtemos:

$$\frac{d\mathbf{g}_a(\mathbf{x}^*, p)}{dp} = 0 \Rightarrow \frac{\partial \mathbf{g}_a}{\partial p} + \left(\frac{d\mathbf{g}_a}{d\mathbf{x}} \right)^T \frac{d\mathbf{x}^*}{dp} = 0 \Rightarrow \mathbf{N}^T \frac{d\mathbf{x}^*}{dp} + \frac{\partial \mathbf{g}_a}{\partial p} = 0$$

Assim resolvendo o sistema de equações acima, obtemos $\frac{d\mathbf{x}^*}{dp}$ e $\frac{d\lambda}{dp}$.

Assim a derivada $\frac{df^*}{dp}$ é dada por:

$$\begin{aligned} \frac{df}{dp} &= \frac{\partial f}{\partial p} + \sum_{i=1}^n \frac{\partial f}{\partial x_i} \frac{dx_i^*}{dp} = \frac{\partial f}{\partial p} + (\nabla f)^T \frac{d\mathbf{x}^*}{dp}; \text{ mas } (\nabla f)^T = \lambda^T \mathbf{N}^T \Rightarrow \\ &\Rightarrow \frac{df}{dp} = \frac{\partial f}{\partial p} + \lambda^T \mathbf{N}^T \frac{d\mathbf{x}^*}{dp} \Rightarrow \frac{df}{dp} = \frac{\partial f}{\partial p} - \lambda^T \frac{\partial \mathbf{g}_a}{\partial p} \end{aligned}$$

Por exemplo, se: $g_j(\mathbf{x}) = G_j(\mathbf{x}) - p \geq 0 \Rightarrow \frac{\partial g_j}{\partial p} = -1 \Rightarrow \frac{df}{dp} = \lambda_j$

Portanto, pode-se dizer que λ_j é o preço que se paga em termos de aumento de f por tornar mais difícil de satisfazer g_j . Por isso, $\lambda_j > 0$ na solução ótima (KKT), caso contrário ($\lambda_j < 0$) a função f seria reduzida (otimizada) quando a restrição se torna mais difícil de ser satisfeita (um absurdo). Portanto, λ_j representa o “preço” das restrições.

Vejamos um exemplo. Considere o problema abaixo:

$$\begin{aligned} \text{Min } & f(\mathbf{x}) = x_1 + x_2 + x_3 \\ & x_1, x_2 \\ \text{tal que } & g_1 = p - x_1^2 - x_2^2 \geq 0 \\ & g_2 = x_3 - 4 \geq 0 \\ & g_3 = x_2 + p \geq 0 \end{aligned}$$

Para $p=8$ a solução vale $\mathbf{x}^* = (-2, -2, 4)^T \Rightarrow f = 0$; $g_1 = 0$; $g_2 = 0$; $\lambda^T = (0, 25, 1)$, assim temos duas restrições ativas na solução ótima. O objetivo é encontrar $\frac{d\mathbf{x}^*}{dp}$ e $\frac{d\lambda}{dp}$.

Solução:

Montando o sistema de equações acima temos:

$$\frac{\partial f}{\partial p} = 0 \text{ e } \frac{\partial \mathbf{g}_a}{\partial p} = \begin{Bmatrix} 1 \\ 0 \end{Bmatrix} \Rightarrow \frac{df}{dp} = -\lambda^T \frac{\partial \mathbf{g}_a}{\partial p} = -0,25;$$

Além disso: $\mathbf{A} = \mathbf{0}$; $\nabla f = (1,1,1)^T \Rightarrow \frac{\partial \nabla f}{\partial p} = \mathbf{0} \Rightarrow$

$$\Rightarrow \mathbf{N} = \begin{bmatrix} -2x_1 & 0 \\ -2x_2 & 0 \\ 0 & 1 \end{bmatrix} \Rightarrow \frac{\partial \mathbf{N}}{\partial p} = \mathbf{0}; \quad \frac{\partial^2 g_1}{\partial x_1^2} = \frac{\partial^2 g_1}{\partial x_2^2} = -2; \quad \frac{\partial^2 g_2}{\partial x_1^2} = \frac{\partial^2 g_2}{\partial x_2^2} = \frac{\partial^2 g_1}{\partial x_1 \partial x_2} = \frac{\partial^2 g_2}{\partial x_1 \partial x_2} = 0 \Rightarrow$$

$$\Rightarrow z_{11} = -2\lambda_1 = -0,5; \quad z_{22} = -2\lambda_1 = -0,5; \quad \mathbf{Z} = \begin{bmatrix} -0,5 & 0 & 0 \\ 0 & -0,5 & 0 \\ 0 & 0 & 0 \end{bmatrix}; \quad \mathbf{N} = \begin{bmatrix} 4 & 0 \\ 4 & 0 \\ 0 & 1 \end{bmatrix} \Rightarrow$$

$$\Rightarrow (\mathbf{A} - \mathbf{Z}) \frac{d\mathbf{x}^*}{dp} - \mathbf{N} \frac{d\lambda}{dp} + \frac{\partial(\nabla f)}{\partial p} - \left(\frac{\partial \mathbf{N}}{\partial p} \right) \lambda = 0 \Rightarrow \begin{cases} 0,5\dot{x}_1 - 4\dot{\lambda}_1 = 0 \\ 0,5\dot{x}_2 - 4\dot{\lambda}_1 = 0 \\ \dot{\lambda}_2 = 0 \end{cases}$$

e:

$$\mathbf{N}^T \frac{d\mathbf{x}^*}{dp} + \frac{\partial \mathbf{g}_a}{\partial p} = 0 \Rightarrow \begin{cases} 4\dot{x}_1 + 4\dot{x}_2 + 1 = 0 \\ \dot{x}_3 = 0 \end{cases}$$

A solução desse sistema de equação é: $\dot{x}_1 = \dot{x}_2 = -0,125$; $\dot{x}_3 = 0$; $\dot{\lambda}_1 = -0,0156$; $\dot{\lambda}_2 = 0$

Variando p de 8 para 9, o novo valor de f é: $x_1 = x_2 = -2,121$; $x_3 = 4$; $f = -0,242$;

Realizando-se uma extrapolação linear baseada nos valores de derivadas calculadas acima, obtemos:

$$(\Delta p = 1) \Rightarrow x_1 = x_2 = -2,125; \quad x_3 = 4; \quad f = -0,25;$$

6. PROBLEMAS CLÁSSICOS

Nesse capítulo continua-se à discussão iniciada no capítulo 3, sobre a formulação e características de alguns problemas clássicos em otimização de sistemas mecânicos.

6.1 Minimização da Flexibilidade

A flexibilidade (C) é dada por $\mathbf{u}^T \mathbf{F}$, onde \mathbf{u} e \mathbf{F} são o vetor deslocamento e força, respectivamente. A minimização da flexibilidade está diretamente relacionada com a maximização da rigidez na direção dada por \mathbf{F} . Trata-se de uma função objetivo clássica em otimização estrutural, e foi demonstrado matematicamente que esse problema é convexo. Em geral é acompanhada de uma restrição global de volume da estrutura (V), e eventualmente restrições locais de tensão mecânica. Assim, a sua formulação é do tipo:

$$\begin{aligned} & \text{Min}_{\mathbf{x}, \mathbf{u}} \quad \mathbf{u}^T \mathbf{F} \\ & \text{tal que} \quad \mathbf{K}\mathbf{u} = \mathbf{F} \text{ (e condições de contorno)} \\ & \quad \quad \mathbf{x} - \mathbf{x}_{\min} \geq 0 \\ & \quad \quad V - V_{\max} \leq 0 \end{aligned}$$

onde \mathbf{x} são as variáveis de projeto. Considerando a equação de equilíbrio $\mathbf{K}\mathbf{u} = \mathbf{F}$, temos:

$$\mathbf{u}^T \mathbf{F} = \mathbf{u}^T \mathbf{K}\mathbf{u}$$

e portanto a sua formulação simplificada fica:

$$\begin{aligned} & \text{Min}_{\mathbf{x}} \quad \mathbf{u}^T \mathbf{K}\mathbf{u} \\ & \text{tal que} \quad \mathbf{x} - \mathbf{x}_{\min} \geq 0 \\ & \quad \quad V - V_{\max} \geq 0 \end{aligned}$$

Um resultado interessante de ser apresentado é o cálculo da sensibilidade da flexibilidade. Derivando-se a fórmula da flexibilidade em relação à variável x_i temos:

$$\begin{aligned} C = \mathbf{u}^T \mathbf{K}\mathbf{u} & \Rightarrow \frac{dC}{dx_i} = \frac{d\mathbf{u}^T}{dx_i} \mathbf{K}\mathbf{u} + \mathbf{u}^T \frac{d\mathbf{K}}{dx_i} \mathbf{u} + \mathbf{u}^T \mathbf{K} \frac{d\mathbf{u}}{dx_i}; \text{ mas: } \mathbf{K}\mathbf{u} = \mathbf{F} \Rightarrow \frac{d(\mathbf{K}\mathbf{u})}{dx_i} = \\ & = \frac{d\mathbf{F}}{dx_i} \Rightarrow \frac{d\mathbf{K}}{dx_i} \mathbf{u} + \mathbf{K} \frac{d\mathbf{u}}{dx_i} = \frac{d\mathbf{F}}{dx_i} = \mathbf{0} \Rightarrow \mathbf{K} \frac{d\mathbf{u}}{dx_i} = -\frac{d\mathbf{K}}{dx_i} \mathbf{u} \Rightarrow \frac{d\mathbf{u}}{dx_i} = -\mathbf{K}^{-1} \frac{d\mathbf{K}}{dx_i} \mathbf{u} \Rightarrow \\ & \Rightarrow \frac{dC}{dx_i} = -\mathbf{u}^T \frac{d\mathbf{K}}{dx_i} (\mathbf{K}^{-1})^T \mathbf{K}\mathbf{u} + \mathbf{u}^T \frac{d\mathbf{K}}{dx_i} \mathbf{u} - \mathbf{u}^T \mathbf{K} \mathbf{K}^{-1} \frac{d\mathbf{K}}{dx_i} \mathbf{u} \Rightarrow \frac{dC}{dx_i} = -\mathbf{u}^T \frac{d\mathbf{K}}{dx_i} \mathbf{u} \end{aligned}$$

Resulta portanto numa expressão matemática simples. Devido a simplicidade de seus gradientes e sua convexidade, trata-se de um problema fácil de ser implementado.

6.2 Maximização da Freqüência de Ressonância

A sua formulação é do tipo:

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

ou

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

Ao contrário da minimização da flexibilidade, esse problema não é convexo. A razão disso é devido a um problema que surge denominado inversão de modos, que será descrito a seguir

6.2.1 Inversão de Modos

Suponha que desejamos maximizar a primeira freqüência de ressonância de uma viga bi-engastada usando como variáveis de projeto as áreas das seções ao longo da viga. Ao final da otimização, verifica-se que o valor da freqüência de ressonância aumentou, no entanto, realizando uma análise modal completa da viga, percebe-se que a freqüência em questão não é mais a primeira freqüência, mas a segunda, ou a terceira, etc. O gráfico da Fig.6.5.1 ilustra o que acontece durante a otimização:

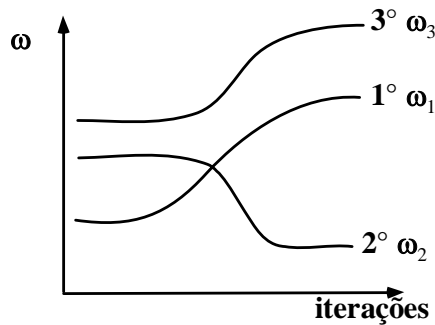


Fig.6.5.1: Inversão de modos.

Ou seja, durante a otimização houve uma inversão de modos de vibrar que ocorreu pelo fato de estarmos alterando a geometria da viga. A Fig.6.5.2 ilustra o caso em que o modo de vibrar que era inicialmente o primeiro (\mathbf{u}_1) apresenta uma freqüência de ressonância maior do que o modo de vibrar que era inicialmente o segundo (\mathbf{u}_2). Note

que isso aconteceu porque, devido a distribuição das áreas, a nova geometria apresenta dois pontos de "estrangulamento" que estão associados à configuração do segundo modo de vibrar inicial (u_2) reduzindo consideravelmente a sua rigidez, causando portanto a inversão das freqüências e modos. Ainda na mesma figura, aumentando o valor das áreas nesses pontos de "estrangulamento" chegamos num ponto em que ambos os modos apresentam a mesma configuração e a mesma freqüência de ressonância.

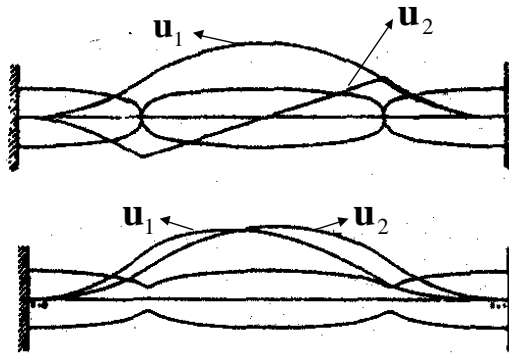


Fig.6.5.2: Inversão de modos de uma viga bi-engastada.

É interessante comentar que esse problema não foi notado pelos pesquisadores da área quando a resolução analítica dos primeiros problemas de maximização de freqüência de ressonância foram publicados no começo da década de 70. Somente quando foi utilizado o MEF para uma análise mais detalhada dos modos de vibrar dos resultados é que se percebeu o problema.

Possíveis soluções para tentar contornar esse problema exigem a alteração da formulação da função objetivo. Alguns exemplos de formulações propostas estão ilustrados abaixo:

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

ou

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

onde ω_0 é uma freqüência especificada. As formulações acima tentam contornar o problema de inversão de modos, no entanto o problema de otimização em si continua

sendo não-convexo, ou seja, podemos ter mais do que uma configuração estrutural que resulte na maximização da frequência. Em geral, obtém-se sucesso para a maximização da primeira frequência de ressonância.

A formulação similar a apresentada na seção 3.3 para a solução de problemas Min-Max também é usada:

$$\begin{array}{l}
 \text{Max } \beta \\
 \mathbf{x} \\
 \text{tal que } (\mathbf{K} - \omega^2 \mathbf{M})\mathbf{u} = \mathbf{0} \text{ e } \mathbf{u}^T \mathbf{M} \mathbf{u} = 1 \\
 \omega \geq \beta \\
 \mathbf{x} - \mathbf{x}_{\min} \geq 0 \\
 \mathbf{V} - \mathbf{V}_{\max} \leq 0
 \end{array}$$

Um outro problema interessante envolvendo frequência de ressonância é o projeto de uma estrutura com frequências de ressonância especificadas ω_{01} , ω_{02} , ω_{03} , etc. A sua formulação é dada por:

$$\begin{array}{l}
 \text{Min } (\omega_1^2 - \omega_{01}^2) + (\omega_2^2 - \omega_{02}^2) + (\omega_3^2 - \omega_{03}^2) + \dots \\
 \mathbf{x} \\
 \text{tal que } (\mathbf{K} - \omega^2 \mathbf{M})\mathbf{u} = \mathbf{0} \\
 \mathbf{u}^T \mathbf{M} \mathbf{u} = 1 \\
 \mathbf{V} - \mathbf{V}_{\max} \leq 0
 \end{array}$$

No entanto, trata-se de um problema altamente não-convexo.

7. ASPECTOS PRÁTICOS DA OTIMIZAÇÃO EM ENGENHARIA

Nesse capítulo são discutidos alguns tópicos relacionados com a aplicação prática da otimização em problemas de grande porte de engenharia.

7.1 Aproximações

As técnicas de aproximação são usadas em geral quando lidamos com problemas de otimização que apresentam um alto custo computacional no cálculo da função objetivo e derivadas ou quando a função objetivo não é dada por uma expressão matemática ou solução de um problema numérico, mas se conhece apenas o seu valor experimental para algumas combinações de variáveis de projeto.

Quando um problema apresenta um alto custo computacional a utilização de métodos apresentados nos capítulos 3 e 4 (com exceção do 4.4 e 4.5) é desestimulada. As técnicas de aproximação tentam reduzir esse custo computacional aproximando o problema de otimização por séries de Taylor em cada iteração. Métodos mais sofisticados baseados nas técnicas de aproximação já foram apresentados nas seções 4.4 e 4.5 (PLS e PQS, respectivamente). Nessa seção as aproximações são discutidas de uma forma mais abrangente.

Essencialmente, as técnicas de aproximação realizam a otimização aproximando a função objetivo e restrições por funções aproximadoras do tipo série de Taylor, ou seja:

$$F(\mathbf{x}) \approx F(\mathbf{x}^0) + \nabla F(\mathbf{x}^0) \cdot \delta \mathbf{x} + \frac{1}{2} \delta \mathbf{x}^T \mathbf{H}(\mathbf{x}^0) \delta \mathbf{x} + \dots$$

onde $\nabla F(\mathbf{x})$ indica gradiente, $\mathbf{H}(\mathbf{x})$ é a matriz Hessiana e $\delta \mathbf{x} = \mathbf{x} - \mathbf{x}^0$. A aproximação é atualizada a cada iteração usando o \mathbf{x} corrente e o processo é repetido até que ocorra a convergência. Entre as principais diferenças em relação aos métodos PLS e PQS vistos na seções 4.4 e 45, temos:

- $\nabla F(\mathbf{x})$ e $\mathbf{H}(\mathbf{x})$ não são em geral atualizados em todas as iterações, mas apenas quando não ocorre mais progresso na otimização;
- $\nabla F(\mathbf{x})$ e $\mathbf{H}(\mathbf{x})$ são calculados resolvendo um sistema de equações lineares montado com informações do projeto e não usando os métodos apresentados anteriormente.

Os dois conceitos acima reduzem o custo computacional da análise, já que os gradientes e eventualmente até a matriz Hessiana não precisam ser avaliados em toda iteração, além de permitir o cálculo dessas informações através da solução de um sistema linear.

Veamos como $\nabla F(\mathbf{x})$ e $\mathbf{H}(\mathbf{x})$ podem ser calculados. Suponha que possuímos k tipos de projetos além do projeto inicial \mathbf{x}^0 : $\mathbf{x}^1, \mathbf{x}^2, \dots, \mathbf{x}^k$, cada um com n variáveis de projeto, representando uma configuração de projeto diferente e com valores de função objetivo F iguais a: F^1, F^2, \dots, F^k , respectivamente. Considerando a expansão da série de Taylor acima usando somente termos de primeira e segunda ordem, temos:

$$F(\mathbf{x}) \approx F(\mathbf{x}^0) + \nabla F(\mathbf{x}^0) \cdot \delta \mathbf{x} + \frac{1}{2} \delta \mathbf{x}^T \mathbf{H}(\mathbf{x}^0) \delta \mathbf{x} \Rightarrow \Delta F = \left. \frac{\partial F}{\partial x_1} \right|_{x_0} \delta x_1 + \left. \frac{\partial F}{\partial x_2} \right|_{x_0} \delta x_2 + \dots +$$

$$+ \left. \frac{\partial F}{\partial x_n} \right|_{x_0} \delta x_n + \frac{1}{2} (H_{11} \delta x_1^2 + H_{22} \delta x_2^2 + \dots + H_{nn} \delta x_n^2) + H_{12} \delta x_1 \delta x_2 + \dots + H_{1n} \delta x_1 \delta x_n +$$

$$+ H_{23} \delta x_2 \delta x_3 + \dots + H_{n-1,n} \delta x_{n-1} \delta x_n \quad \text{onde } \delta x_i = x_i - x_i^0; \Delta F = F - F^0; i = 1, n$$

Substituindo $\mathbf{x}^1, \mathbf{x}^2, \dots, \mathbf{x}^k$ na equação da série de Taylor acima teremos k equações e como incógnitas as componentes de $\nabla F(\mathbf{x})$ e $\mathbf{H}(\mathbf{x})$, ou seja:

$$\left. \frac{\partial F}{\partial x_1} \right|_{x_0}, \left. \frac{\partial F}{\partial x_2} \right|_{x_0}, \dots, \left. \frac{\partial F}{\partial x_n} \right|_{x_0} \text{ e } H_{11}, H_{12}, \dots, H_{nn}, \text{ totalizando } L = n + n(n+1)/2 \text{ incógnitas.}$$

Se $k=L$ a solução é obtida de forma direta. Se $k>L$, teremos mais equações do que incógnitas, e nesse caso a solução é obtida utilizando-se o método dos mínimos quadrados como descrito na seção 4.3.2.2. Se $k<L$, teremos mais incógnitas do que equações e somente algumas incógnitas serão calculadas. Obtido os gradientes e as componentes da matriz Hessiana é montado um subproblema de programação linear com as séries de Taylor da função objetivo e restrições que é resolvido.

Vejam os exemplos para ilustrar o método. Considere a solução do problema de otimização abaixo:

$$\begin{aligned} \text{Min } F(\mathbf{x}) &= x_1 + x_2 \\ \text{tal que } g &= \frac{1}{x_1} + \frac{1}{x_2} - 2 \leq 0 \\ 0,1 \leq x_i &\leq 5 \quad i = 1, 2 \end{aligned}$$

A tabela 7.1.1 abaixo descreve a informação sobre projetos (combinações de variáveis de projeto) anteriores que são conhecidos.

Tabela 7.1.1: Projetos conhecidos.

| Projeto | x_1 | x_2 | F | g |
|---------|-------|-------|-----|-------|
| 2 | 2 | 2 | 4 | -1 |
| 1 | 0,5 | 2 | 2,5 | 0,5 |
| 0 | 1,5 | 1 | 2,5 | -0,33 |

Solução:

Utilizando os três projetos disponíveis na tabela podemos escrever $k=2$ equações:

$$\Delta F^1 = 1,5 = 0,5 \left. \frac{\partial F}{\partial x_1} \right|_{x_0} + 1,0 \left. \frac{\partial F}{\partial x_2} \right|_{x_0}$$

$$\Delta F^2 = 0,0 = -1,0 \left. \frac{\partial F}{\partial x_1} \right|_{x_0} + 1,0 \left. \frac{\partial F}{\partial x_2} \right|_{x_0}$$

onde os termos da matriz Hessiana não foram considerados. A solução é igual à: $\nabla F \approx (1 \ 1)^T$

Para as restrições podemos montar um sistema similar:

$$\Delta g^1 = -0,6667 = 0,5 \frac{\partial g}{\partial x_1} \Big|_{x_0} + 1,0 \frac{\partial g}{\partial x_2} \Big|_{x_0}$$

$$\Delta g^2 = 0,8333 = -1,0 \frac{\partial g}{\partial x_1} \Big|_{x_0} + 1,0 \frac{\partial g}{\partial x_2} \Big|_{x_0}$$

cuja solução vale: $\nabla g = (-1 \ -0,1667)^T$.

Assim, o subproblema de programação linear fica:

$$\begin{array}{l} \text{Min } F \approx 2,5 + \delta x_1 + \delta x_2 \\ \delta x \\ \text{tal que } g \approx -0,33 - \delta x_1 - 0,167 \delta x_2 \leq 0 \\ -0,2 \leq \delta x_i \leq 0,2 \quad i = 1,2 \end{array}$$

onde foram adotado limites móveis iguais à 0,2. A solução desse problema é:

$$\delta x_1 = \delta x_2 = -0,2; F \approx 2,1; g \approx -0,1 \Rightarrow \mathbf{x} = (1,3 \ 0,8)^T$$

Com essa nova informação podemos montar uma nova aproximação incluindo o termo H_{11} da matriz Hessiana. Seguindo o mesmo procedimento anterior:

$$\begin{array}{l} \text{Min } F \approx 2,1 + \delta x_1 + \delta x_2 \\ \delta x \\ \text{tal que } g \approx 0,0192 - 0,7018 \delta x_1 + \\ -1,6576 \delta x_2 + 5,964 \delta x_1^2 \leq 0 \\ -0,2 \leq \delta x_i \leq 0,2 \quad i = 1,2 \end{array}$$

Cuja solução vale: $\delta \mathbf{x} = (-0,1598 \ 0,1252)^T \Rightarrow \mathbf{x} = (1,14 \ 0,9252)^T$.

Na sétima iteração obtém-se $\mathbf{x} = (0,989 \ 1,011)^T$ enquanto a solução ótima é: $\mathbf{x}^* = (1 \ 1)^T$. A Fig.7.1.1 ilustra as várias iterações até chegar no ponto ótimo.

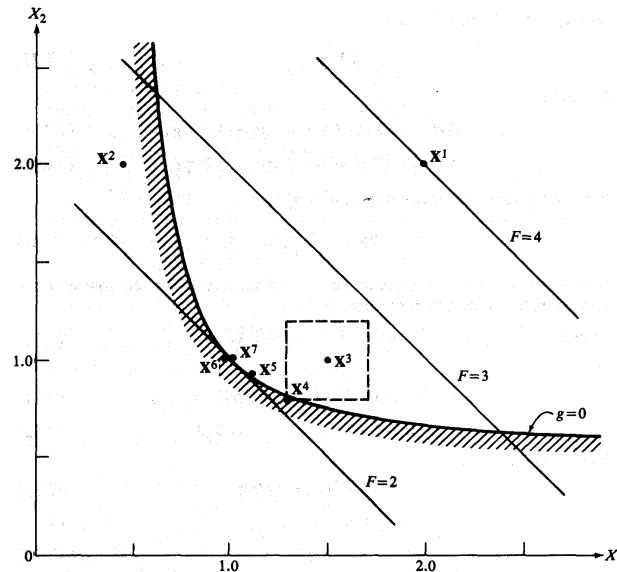


Fig.7.1.1: Otimização usando aproximação seqüencial.

Existem outros tipos de função aproximadora que são obtidas realizando-se uma mudança de variável. Por exemplo:

$$y_i = y_i(\mathbf{x}) = \frac{1}{x_i} \Rightarrow F(\mathbf{y}) = F(\mathbf{y}_0) + \sum_{i=1}^n (y_i - y_{0i}) \left(\frac{\partial g}{\partial y_i} \right)_{y_0} \Rightarrow F(\mathbf{x}) = F(\mathbf{x}_0) +$$

$$+ \sum_{i=1}^n (y_i(x_i) - y_i(x_{0i})) \left(\frac{\partial g}{\partial x_i} \cdot \frac{dy_i}{dx_i} \right)_{x_0} \Rightarrow F(\mathbf{x}) = F(\mathbf{x}_0) + \sum_{i=1}^n (x_i - x_{0i}) \frac{x_{0i}}{x_i} \left(\frac{\partial g}{\partial x_i} \right)_{x_0}$$

Essa equação de aproximação é denominada "Aproximação Recíproca". A vantagem de se fazer essa mudança de variável é preservar a linearidade das variáveis de projeto no problema de otimização. Como exemplo considere o problema de minimização do peso de uma treliça mostrado abaixo, onde as variáveis de projeto são as áreas.

$$\begin{aligned} \text{Min } W &= \rho AL \quad (\text{Linear}) \\ \text{tal que } \sigma &= \frac{P}{A} \leq \bar{\sigma} \quad (\text{n\~ao-linear}) \\ \delta &= \frac{PL}{EA} \leq \bar{\delta} \quad (\text{n\~ao-linear}) \end{aligned}$$

A princípio embora função objetivo seja linear, as duas restrições são não-lineares em relação às variáveis de projeto. Realizando-se a mudança de variável $B = \frac{1}{A}$, obtemos a nova formulação:

$$\begin{aligned} \text{Min } W &= \rho \frac{L}{B} \quad (\text{n\~ao-linear}) \\ \text{tal que } \sigma &= PB \leq \bar{\sigma} \quad (\text{Linear}) \\ \delta &= \frac{PLB}{E} \leq \bar{\delta} \quad (\text{Linear}) \end{aligned}$$

Aproximações recíprocas tem grande aplicação quando a matriz de rigidez do elemento estrutural é o produto da variável de projeto por uma matriz de constantes. Nesse caso, os deslocamentos serão, em geral, inversamente proporcionais às variáveis de projeto. Assim, se a função objetivo ou restrições forem funções lineares dos deslocamentos (e portanto funções não-lineares em relação às variáveis de projeto), a aproximação recíproca garantirá a linearidade dessas funções em relação às variáveis de projeto. A tabela 7.1.2 mostra essa situação para os tipos básicos de estruturas:

Tabela 7.1.2: Vantagem da aproximação recíproca.

| | matriz | deslocamento | Aproximação Recíproca |
|------------|---|------------------------------------|-----------------------|
| •Trelças: | $\mathbf{k} = A * \mathbf{C} \Rightarrow$ | $\delta \propto 1/A \Rightarrow$ | $y = 1/A$ |
| •Membrana: | $\mathbf{k} = t * \mathbf{C} \Rightarrow$ | $\delta \propto 1/t \Rightarrow$ | $y = 1/t$ |
| •Placas: | $\mathbf{k} = t^3 * \mathbf{C} \Rightarrow$ | $\delta \propto 1/t^3 \Rightarrow$ | $y = 1/t^3$ |
| •Pórticos: | $\mathbf{k} = f(A, I, C) \Rightarrow$ | $\delta \propto 1/I \Rightarrow$ | $y = 1/I$ |

Entretanto, essa linearidade já não é possível de se obter usando a aproximação recíproca quando há funções mais complicadas que dependem da frequência de ressonância, por exemplo. Nesse caso, a frequência é a razão entre a rigidez e a massa, e portanto a dependência em relação às variáveis de projeto é mais complicada.

Podemos dizer de uma forma geral que as aproximações são competitivas quando:

- há poucas variáveis de projeto;
- custo computacional da análise é alto;
- podem ser usados dados experimentais de projetos conhecidos;
- podem fornecer um resultado inicial para outro método.

7.2 Eliminação de Restrições

Em geral nos problemas de otimização em engenharia devem ser considerados milhares de restrições de desigualdade. Por exemplo, consideremos o caso das restrições de tensão mecânica. Como discutido na seção 3.3, o valor máximo da tensão mecânica muda de posição a medida que a estrutura é alterada, resultando num problema de otimização do tipo Min-Max. A solução é impor restrições locais de tensão mecânica ao longo do maior número possível de pontos críticos na estrutura. Isso pode resultar num grande número de restrições no problema de otimização. No entanto somente algumas serão críticas (ativas) e as demais poderiam ser eliminadas do problema.

Além disso, existem restrições que são muito difíceis de calcular como por exemplo a restrição de agitação ("fluttering") das asas de um avião.

Assim com o intuito de reduzir o custo computacional associado às restrições, duas estratégias são adotadas:

- Eliminar as restrições menos críticas baseadas num valor de corte e proceder com a otimização com as restrições mais críticas. A cada iteração (ou algumas iterações) as restrições consideradas são atualizadas;

- Eliminar as restrições difíceis de calcular, verificando ao final da otimização a sua validade. Se forem violadas incluí-las e proceder a otimização com a solução obtida.

Essas estratégias constituem a chamada técnica de eliminação de restrições. O histograma da Fig.7.2.1 ilustra o emprego dessa técnica. As restrições são separadas por regiões no modelo de MEF e conforme o valor de corte que é adotado para o valor das restrições, mais ou menos restrições são eliminadas da otimização. A cada iteração todas as restrições são verificadas e aquelas que caso foram violadas são reincluídas na otimização na iteração seguinte.

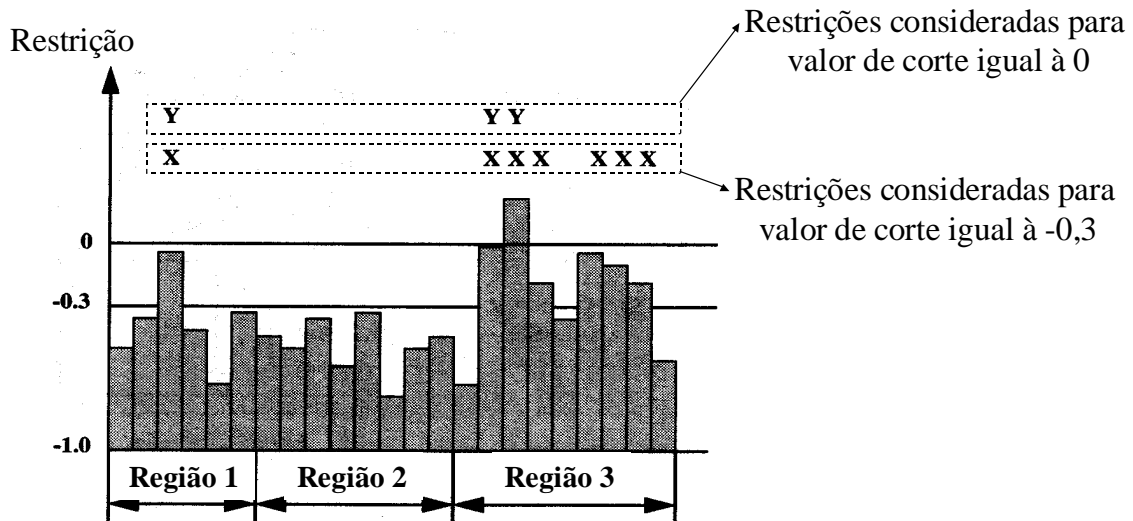


Fig.7.2.1: Histograma para a eliminação de restrições.

A Fig.7.2.2 ilustra um algoritmo para projeto de treliça que inclui o método de aproximação recíproca e a técnica de eliminação de restrições.

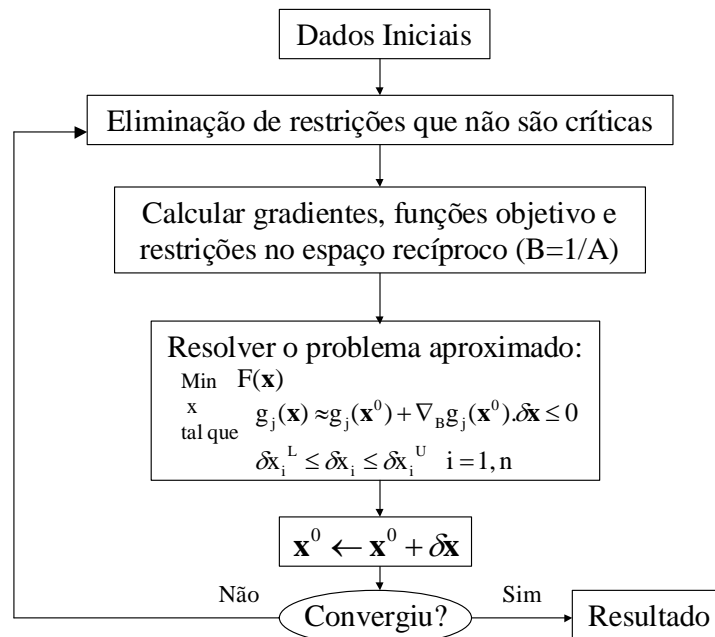


Fig.7.2.2: Algoritmo para o projeto de treliças que inclui os métodos de aproximação recíproca e eliminação de restrições.

7.3 Utilização de Funções Base

Para realização da otimização de forma de uma estrutura é necessário inicialmente representar a sua geometria através de funções splines ou polinômios. No entanto, muitas vezes essas curvas não representam de forma satisfatória a geometria da estrutura. Além disso, em alguns casos se dispõe da informação sobre formas estruturais ótimas para diferentes condições de projeto. Essas formas podem ter sido obtidas ao longo dos anos por métodos experimentais por exemplo e se espera que a estrutura a ser otimizada para as novas condições tenha pequenas mudanças em relação as demais formas. Nesse caso, podemos definir uma "base" como sendo constituída pela geometria dessas formas básicas e representar a geometria da estrutura a ser otimizada como uma combinação linear da geometria das formas básicas (base), onde os coeficientes da combinação linear são as variáveis de projeto.

Consideremos como exemplo clássico o problema de otimização aerodinâmica de um perfil de asa. A forma do perfil da asa pode ser obtida pela combinação linear da forma de perfis pré-existent, ou seja:

$$Y = a_1 Y^1 + a_2 Y^2 + \dots + a_n Y^n$$

onde Y^1, Y^2, \dots, Y^n contém as coordenadas dos perfis, constituindo numa base numericamente definida de perfis de asa e a_1, a_2, \dots, a_n são as variáveis de projeto. Assim o problema de otimização consiste em encontrar os coeficientes dessa combinação linear. A Fig.7.3.1 ilustra um exemplo de base de perfis de asa.

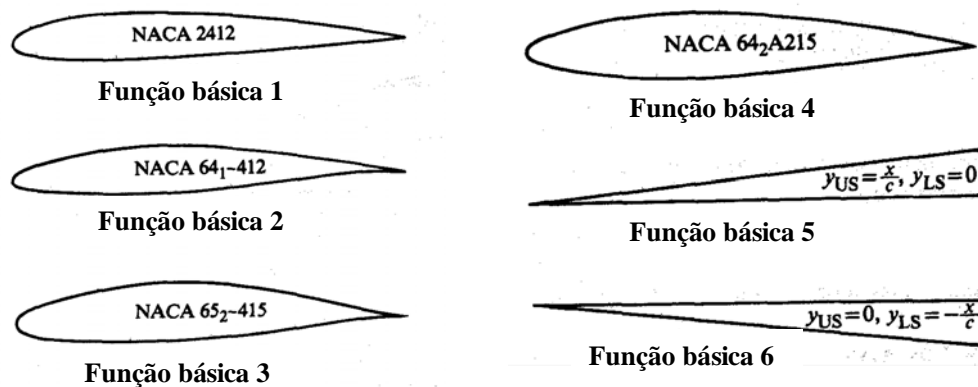


Fig.7.3.1: Base de perfis de asa.

Entre as vantagens da utilização de funções base nesse problema temos:

- as formas de perfis de asa **não** são bem representadas por polinômios (qualquer tipo) ou funções "spline" gerando problemas no condicionamento numérico e no resultado da otimização;
- reduz o número de variáveis de projeto e o custo computacional;
- permite o projetista utilizar projetos existentes de perfis de asa;
- permite alterar apenas porções do perfil;
- essa base pode ser ampliada com os novos perfis otimizados;
- muito eficiente quando usado juntamente com técnicas de aproximação;

Como exemplo consideremos o projeto de um perfil para $M=0,1$ e ângulo de ataque de 6° . A formulação do problema de otimização é descrita abaixo.

$$\begin{aligned} & \text{Max } C_L \\ & \text{tal que } |C_M| \leq 0,05 \\ & \quad |C_p| \leq 2 \text{ em } x/c = 0,01 \\ & A \geq 0,075c^2 \end{aligned}$$

onde C_L é o coeficiente de sustentação, C_M é o coeficiente de momento “pitching”, C_p é o coeficiente de pressão na superfície superior, A é a área da seção do perfil e c é o comprimento da corda da asa. Como hipótese é considerada aerodinâmica invíscida, portanto não há coeficiente de arrasto.

Solução:

Quatro resultados obtidos de diferentes formas são apresentados na Fig.7.3.2.

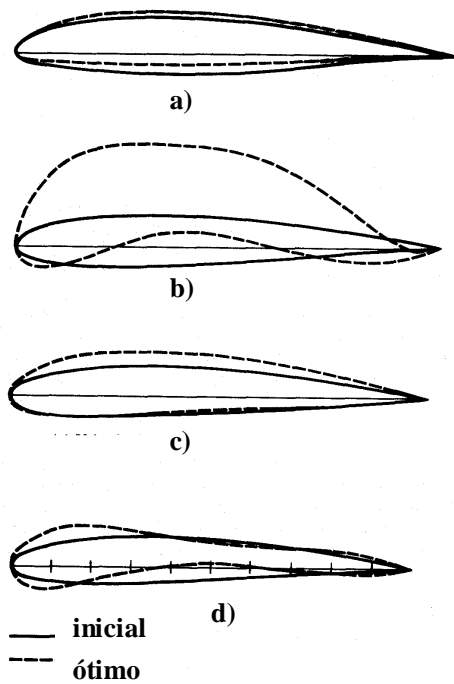


Fig.7.3.2: Resultado da otimização de perfis.

Na situação a) o perfil é representado por polinômios. No caso b) e c) o perfil foi representado usando funções base, mas no caso c) restrições geométricas extras foram impostas. No caso d) o resultado foi obtido usando o método das aproximações. A tabela 7.3.1 mostra os valores de projeto inicial e a tabela 7.3.2 mostra o resultado do projeto ótimo.

Tabela 7.3.1: Projeto Inicial.

| Projeto Inicial | | | | | |
|-----------------|-------|--------|------|------|--------|
| Fig. | C_L | C_M | A | t/c | Camber |
| a | 0,985 | -0,056 | 0,08 | - | - |
| b,c | 0,986 | -0,065 | 0,08 | 0,12 | 0,02 |
| d | 0,986 | -0,065 | 0,08 | 0,12 | 0,02 |

Tabela 7.3.2: Projeto ótimo.

| Projeto Ótimo | | | | | |
|---------------|-------|--------|-------|-------|--------|
| Fig. | C_L | C_M | A | t/c | Camber |
| a | 1,085 | -0,074 | 0,088 | - | - |
| b | 1,478 | -0,072 | 0,171 | 0,253 | 0,14 |
| c | 1,106 | -0,075 | 0,103 | 0,149 | 0,04 |
| d | 1,144 | -0,075 | 0,075 | 0,15 | 0,04 |

Esse problema ilustra um fato importante. Note que os resultados b) e d) não são realísticos. Isso ocorreu porque o algoritmo de análise não incluía o efeito da viscosidade. Assim, qualquer que seja o método de otimização utilizado é de extrema importância que o algoritmo de análise seja o mais genérico possível, devendo ser capaz de simular com precisão as formas complexas resultantes da otimização, e poder nos dizer se o resultado da otimização é viável ou não. Hipóteses simplificadoras embutidas no algoritmo de análise podem levar a resultados de otimização não realísticos.

8. INTRODUÇÃO À OTIMIZAÇÃO TOPOLÓGICA

Otimização topológica (OT) consiste num método computacional que permite projetar a topologia ótima de estruturas segundo um certo critério de custo (por exemplo, máxima rigidez e menor peso). Basicamente, o método de otimização topológica distribui o material no interior de um domínio fixo de forma a maximizar ou minimizar uma função custo especificada (por exemplo, máxima rigidez e mínimo volume de material). O material em cada ponto do domínio pode variar de ar (não há presença de material) até sólido (total presença de material) podendo assumir densidades intermediárias entre ar e sólido de acordo com um modelo de material definido.

Um algoritmo de otimização é usado para encontrar de forma iterativa a distribuição ótima de material, o que torna o processo rápido, caso contrário milhões de análises seriam necessárias para encontrar a distribuição ótima. Na aceleração do processo de busca da distribuição ótima de material, os métodos de otimização utilizam-se da informação dos gradientes (ou derivadas) da função custo em relação à quantidade de material em cada elemento. A distribuição de material é representada, por exemplo, associando um valor de densidade a cada elemento (subdomínio), obtido da discretização do domínio inicial.

Dessa forma, a OT combina essencialmente métodos de otimização com um método numérico de análise, como por exemplo, o método de elementos finitos (MEF). Outros métodos numéricos de análise podem ser usados, no entanto, devem ser genéricos o suficiente de forma a lidar com estruturas de formas complexas resultantes da otimização.

A Otimização Topológica surgiu na área acadêmica na década de 80 nos EUA e Europa com a publicação do artigo "*Generating Optimal Topologies in Structural Design Using a Homogenization Method*" pelos autores Martin P. Bendsoe e Noboru Kikuchi, publicado no *Computer Methods in Applied Mechanics and Engineering*, **71**, pp.197-224, 1988. Já na década de 90 passou a ser largamente utilizada nas indústrias automobilística e aeronáutica dos EUA, Japão e Europa para o projeto de peças mecânicas otimizadas, além de recentemente ter se expandido para outras áreas da engenharia no meio acadêmico como o projeto de mecanismos flexíveis, atuadores piezoelétricos, antenas e motores eletromagnéticos, etc. Ela torna o processo de projeto mais genérico, sistemático, otimizado, e independente da experiência específica de alguns engenheiros, fornecendo a topologia inicial, otimizada para uma certa aplicação, do dispositivo a ser construído. Obviamente, a presença do engenheiro é necessária para a obtenção do projeto final e verificação de seu desempenho para o qual foi projetado (utilizando métodos numéricos e experimentais).

8.1 Introdução

Consideremos o problema clássico de projeto de uma estrutura com a mínima flexibilidade (máxima rigidez) com o menor volume de material como mostrado na Fig.8.1.1. Suponha que a princípio não conhecemos a forma ou a topologia¹ da estrutura.

¹ O conceito de forma está relacionado com a geometria externa da estrutura enquanto o conceito de topologia engloba não somente a geometria externa, mas também a presença de furos internos na estrutura.

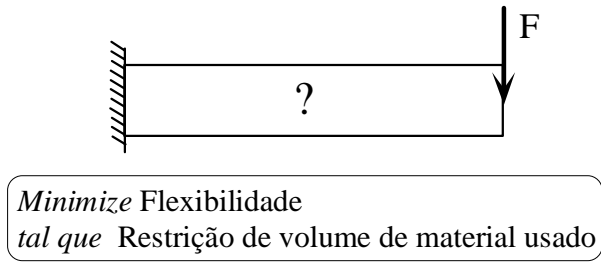


Fig.8.1.1: Problema exemplo.

Temos essencialmente três abordagens em otimização estrutural para a solução desse problema como descrito na Fig.8.1.2. Na primeira delas, é assumido uma forma pré-definida para a estrutura, por exemplo uma viga em "I", e são escolhidas como variáveis de projeto algumas medidas (ou razão entre as medidas) que caracterizam a geometria dessa estrutura como as dimensões b , t e h mostradas na figura. Então, utilizando um software de otimização determina-se o valor ótimo das variáveis b , t e h que fornece a menor flexibilidade (maior rigidez) para a restrição de material imposta. Essa abordagem é denominada otimização paramétrica, e não altera a forma da estrutura, apenas as suas dimensões. Numa segunda abordagem, os contornos externos da estrutura são parametrizados por curvas splines e os parâmetros dessas curvas constituem as variáveis de projeto. Utilizando um software de otimização determinam-se os parâmetros ótimos das curvas splines, e portanto a forma ótima, que minimiza a flexibilidade estrutural para a restrição de material imposta. Trata-se de uma abordagem mais genérica do que a anterior em que a forma externa da estrutura é alterada. Essa abordagem é chamada otimização de forma. Finalmente, a última abordagem mostrada na figura consiste em se encontrar a distribuição ótima de "furos" no interior da estrutura que minimize a flexibilidade considerando a restrição de material. As variáveis de projeto seriam, por exemplo, variáveis que indicam a distribuição de material em cada ponto do domínio. Essa abordagem é denominada otimização topológica sendo a mais genérica em relação as anteriores. A quantidade de material removido (para a mesma flexibilidade) é crescente na ordem apresentada das abordagens, sendo a otimização topológica a que resulta na maior remoção de material da estrutura.

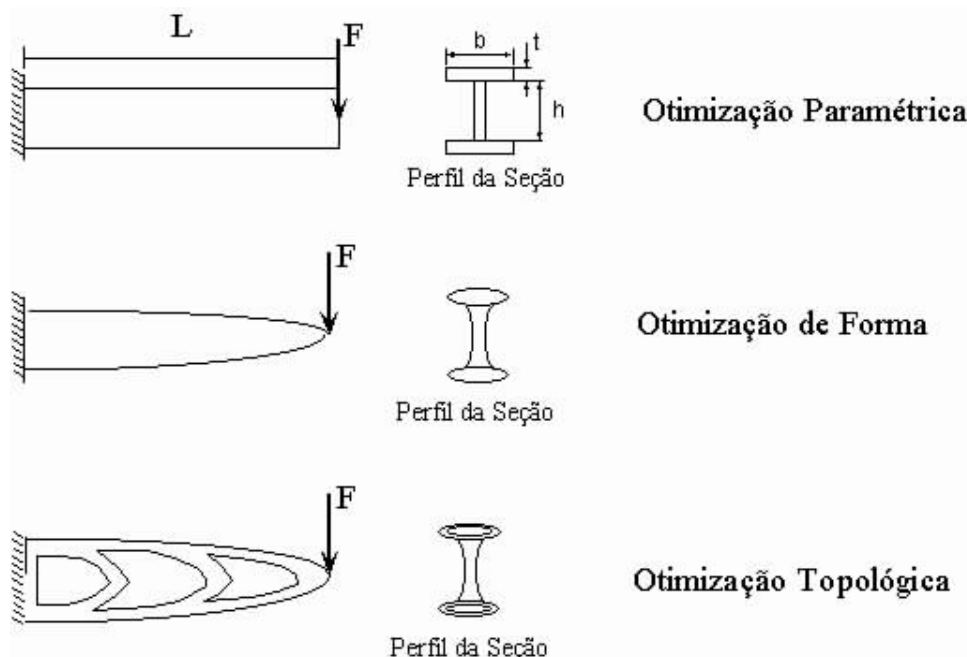


Fig.8.1.2: Abordagens para solução do problema.

O procedimento típico de projeto estrutural usando a otimização topológica, considerando o problema acima, é apresentado na Fig.8.1.3. O primeiro passo consiste em se definir o domínio no qual a estrutura pode existir. Esse domínio é limitado pelas condições de contorno da estrutura (pontos em que ela deve estar restrita) e pelos pontos de aplicação de carga. Outras limitações podem estar relacionadas com a restrição do espaço ocupado.

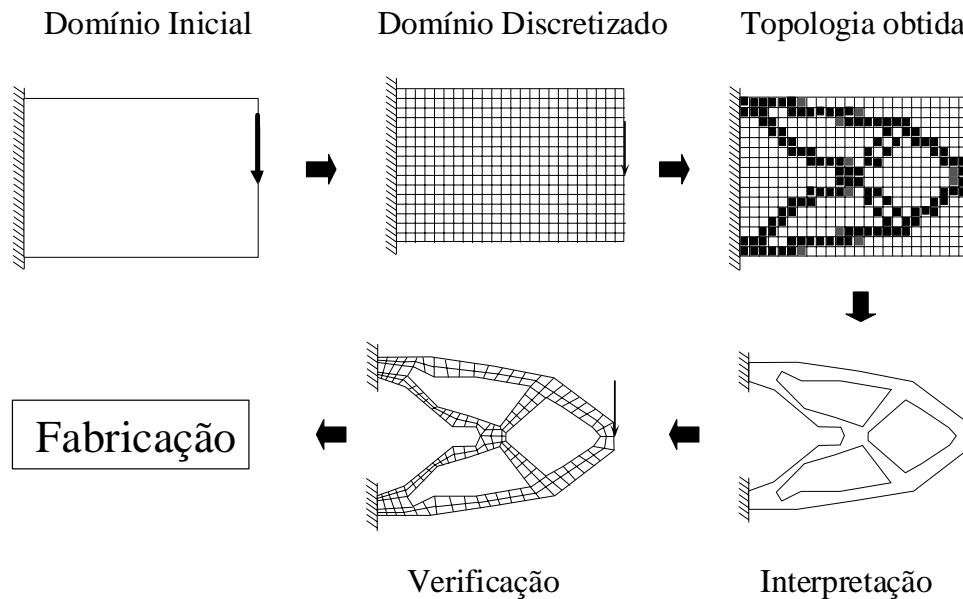


Fig.8.1.3: Procedimento típico de projeto estrutural por otimização topológica.

É importante nessa fase definir o domínio com o maior tamanho possível de forma a não limitar o domínio de trabalho da OT. Como em qualquer método de otimização, quanto mais restrições são impostas, menor é a melhora de desempenho da solução obtida. Assim, restrições de espaço de ocupação da estrutura reduzem a optimalidade da solução no caso da otimização topológica. No segundo passo o domínio é discretizado em elementos finitos e são aplicadas as condições de contorno. No terceiro passo, os dados do domínio são fornecidos ao software de otimização topológica que num processo iterativo distribui o material no domínio de forma a minimizar (ou maximizar) a função objetivo especificada, no caso, flexibilidade. O resultado obtido é do tipo mostrado na Fig.8.1.3 (item "topologia obtida") onde a cor escura indica a presença de material e a cor branca indica a ausência de material no ponto do domínio. Note que podem surgir pontos com cores intermediárias, denominados de escalas de cinza (ou "gray scale" em inglês). Esses pontos indicam a presença de materiais intermediários que não podem ser implementados na prática e sempre ocorrem, ou seja, a presença do "gray scale" é inerente a obtenção da solução ótima. Dessa forma, a imagem da estrutura obtida por OT representa um excelente ponto de partida que necessita ser interpretado para se obter o projeto final da estrutura. Essa interpretação (quarta etapa) pode ser feita usando-se métodos de processamento de imagem, ou simplesmente desenhando-se uma estrutura baseada na imagem obtida por OT. A penúltima etapa consiste em se verificar o resultado final da estrutura. Em geral os resultados gerados por OT não são intuitivos e é interessante fazer uma verificação da estrutura final usando MEF, para criarmos confiança na solução através da comprovação da optimalidade do resultado. Finalmente, a última etapa é a fabricação da estrutura. Hoje, existem várias técnicas de fabricação que permitem fabricar estruturas com formas complexas como prototipagem rápida, entre outras.

8.2 Conceitos Teóricos da Otimização Topológica

A OT é baseada em dois conceitos teóricos principais que serão descritos a seguir: domínio fixo estendido e modelo de material.

8.2.1 Domínio Fixo Estendido

O domínio estendido fixo de projeto, ilustrado na Fig.8.1.3, consiste num domínio de forma fixa limitado pelos pontos de apoio da estrutura e pontos de aplicação de carregamento, e que conterá a estrutura desconhecida. O objetivo da otimização topológica é determinar os espaços sem material (“buracos”) e a conectividade da estrutura através da remoção e adição de material nesse domínio. O problema de otimização consiste portanto em se encontrar a distribuição ótima de propriedades de materiais no domínio estendido fixo.

Na implementação numérica o domínio estendido fixo é discretizado em elementos finitos. Portanto, o modelo de elementos finitos do domínio não é alterado durante o processo de otimização, sendo alterado somente a distribuição de material nos elementos. Essa é uma grande vantagem em relação a otimização de forma, pois sendo o domínio fixo, as derivadas de qualquer função q são facilmente calculadas usando a expressão:

$$\frac{\partial}{\partial A_n} \int_{\Omega} q d\Omega = \int_{\Omega} \frac{\partial q}{\partial A_n} d\Omega$$

onde A_n é uma variável de projeto, e q é uma função. Já na otimização de forma pelo fato do domínio Ω variar as expressões para obter os gradientes se tornam muito mais complicadas.

8.2.2 Modelo de Material

O modelo de material é uma equação que define a mistura em microescala de dois ou mais materiais (um deles pode ser ar) permitindo que hajam estágios intermediários ao se passar da condição de zero material (“buraco”) a sólido em cada ponto do domínio. Isso garante a relaxação do problema de otimização evitando que se passe de ar para sólido de forma brusca, o que torna difícil o tratamento numérico do problema. É portanto um ponto chave na formulação da OT. A princípio, os estágios intermediários de materiais não têm significado físico sendo apenas decorrentes de um recurso matemático para relaxação do problema. A definição do modelo de material define o grau de relaxação do problema.

Existem vários modelos de materiais descritos na literatura que podem ser utilizados. O mais simples é o chamado método das densidades, que consiste numa equação matemática que define o valor da propriedade (\mathbf{E}) em cada ponto do domínio em função da densidade (x) do material usado no projeto (que varia de zero a um valor máximo) e a propriedade básica (\mathbf{E}^0) do material a ser distribuído, como descrito na equação abaixo.

$$\mathbf{E}_i = x_i^p \mathbf{E}^0$$

Dessa forma tem-se uma variável (e portanto um grau de liberdade) em cada ponto do domínio, representada pelo valor da densidade naquele ponto. No caso do problema discretizado, define-se uma variável por elemento finito.

Como vantagens esse método é fácil de implementar e apresenta um baixo custo computacional. No entanto, apresenta como desvantagem o fato que a sua formulação em si é empírica, pode ocorrer a formação de grande quantidade de materiais intermediários (escalas de cinza), além do resultado final depender da discretização da malha de MEF. Esses problemas são contornados com a implementação de "filtros" na otimização.

Um modelo de material mais complexo e robusto foi proposto por Bendsoe e Kikuchi e baseia-se na definição em cada ponto do domínio de uma microestrutura de material composto, como mostrado na Fig.8.2.1. Essa microestrutura consiste numa célula unitária com um buraco retangular no seu interior cujas dimensões são definidas pelas variáveis de projeto a e b e pelo ângulo θ .

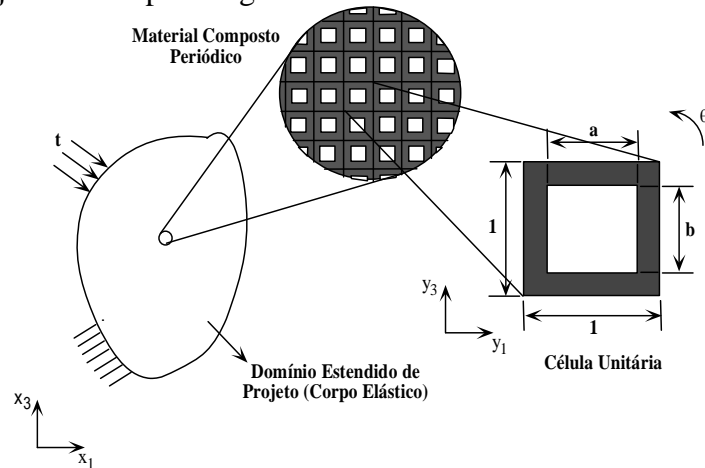


Fig.8.2.1: Modelo de material baseado em homogeneização.

Assim, em cada ponto do domínio, as propriedades no domínio são iguais às propriedades de um material composto gerado pela repetição periódica da microestrutura de dimensões a , b , e θ correspondente aquele ponto. Dessa forma variando-se os valores de a , b , e θ ao longo do domínio estendido fixo durante a otimização altera-se a distribuição das propriedades de material nesse domínio, de maneira que ao final da otimização existirão pontos com ar ($a=b=1$), pontos com sólido ($a=b=0$) e alguns pontos com materiais intermediários. Nesse sentido o problema consiste em se otimizar a distribuição de material num domínio perfurado com infinitos microfuros. As propriedades em cada ponto do domínio são obtidas através do método de homogeneização, que permite o cálculo das propriedades efetivas (c^H) de um material composto conhecida a geometria e composição de sua célula unitária.

Quando a célula unitária é rotacionada de um ângulo θ o novo tensor de propriedades efetivas (ou homogeneizadas) c^G é dado pela a equação:

$$c^G = R(\theta)^t c^H R(\theta)$$

onde R é uma matriz de rotação.

Esse método é mais difícil de implementar e apresenta um custo computacional maior, no entanto, seu modelo de material é mais realista, o resultado é independente da discretização da malha, e o resultado em geral apresenta pouca escala de cinza.

8.3 Implementação Numérica

Na implementação numérica a OT combina o método de elementos finitos (MEF) com algoritmos de otimização, em geral, programação sequencial. O fluxograma da Fig.8.3.1 ilustra um exemplo de implementação da OT.

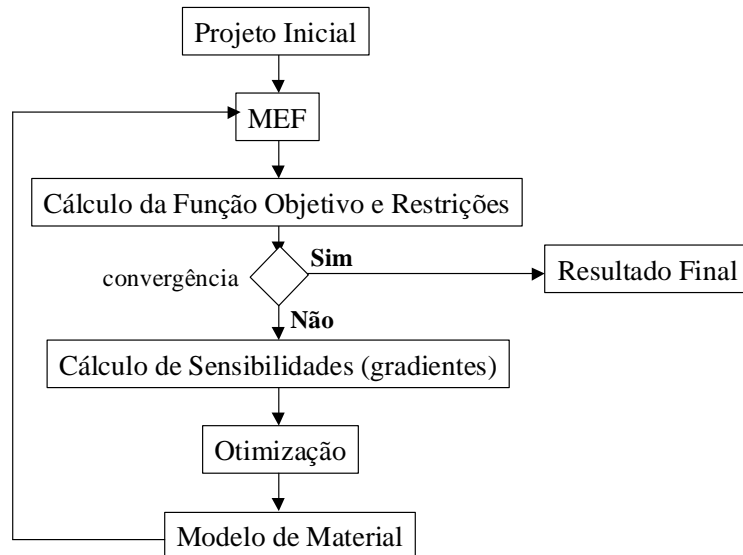


Fig.8.3.1: Fluxograma de implementação da OT.

8.4 Formulação para Treliças

Nos itens anteriores discutiu-se a teoria de OT para um meio contínuo, no entanto OT também é aplicada para projeto de estruturas discretas, como treliças. A idéia é partir de um domínio fixo estendido altamente discretizado em treliças e utilizar como variáveis de projeto as áreas dos elementos de treliça que podem variar de zero a um valor máximo. Ao final da otimização, a topologia da estrutura discreta é dada pelos elementos de treliça com área maior que um valor mínimo. Um exemplo de formulação é dado por:

$$\begin{array}{l}
 \text{Min } \mathbf{u}^T \mathbf{k} \mathbf{u} \quad (\text{flexibilidade}) \\
 \mathbf{A} \\
 \text{tal que } \text{Vol} \leq V_0 \\
 \quad \quad 0 < A_i \leq A_{\max} \quad i = 1, \dots, N
 \end{array}$$

A Fig.8.4.1 mostra alguns exemplos de domínios iniciais de projeto para problemas bidimensionais e tridimensionais.

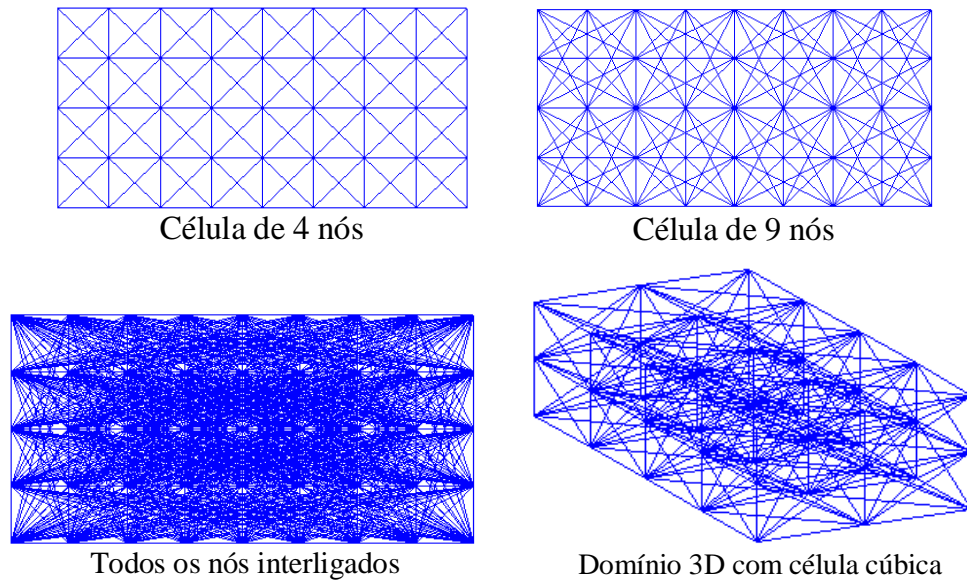


Fig.8.4.1: Domínios Iniciais de projeto para treliças.

As Figs.8.4.2, 8.4.3 e 8.4.4 mostram alguns exemplos de resultados de otimização topológica para treliças com os respectivos domínios iniciais de projeto.

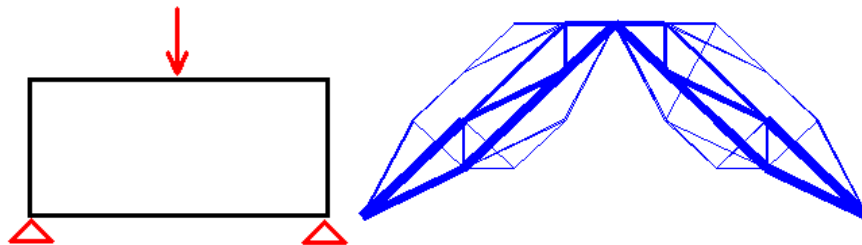


Fig.8.4.2: Domínio inicial bidimensional de projeto e resultado da otimização topológica para treliças.

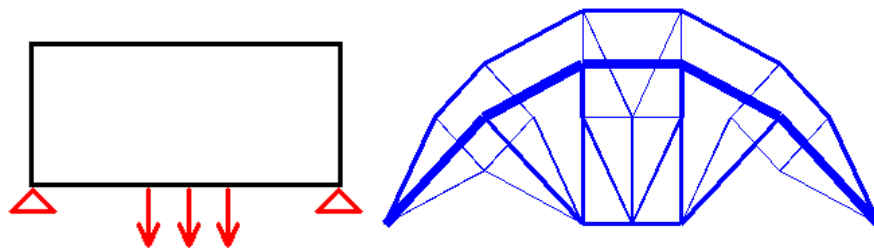


Fig.8.4.3: Domínio inicial bidimensional de projeto e resultado da otimização topológica para treliças.

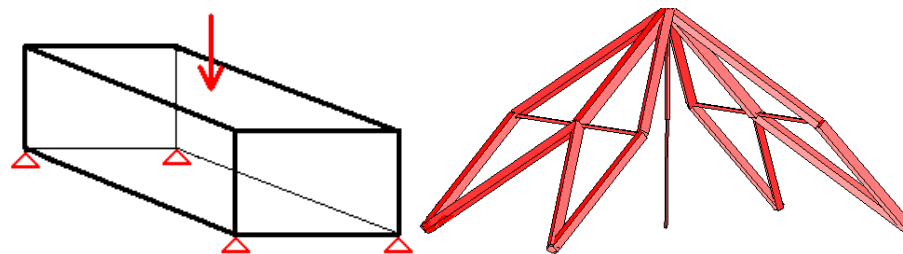


Fig.8.4.4: Domínio inicial tridimensional de projeto e resultado da otimização topológica para treliças.

8.5 Aspectos Práticos da OT

Quando se obtém um resultado de OT para meios contínuos, além das escalas de cinza devemos estar atentos a outros fenômenos. O primeiro deles é o “instabilidade de tabuleiro” (ou “checkerboard” em inglês). A Fig.8.5.1 ilustra o fenômeno "tabuleiro de xadrez".

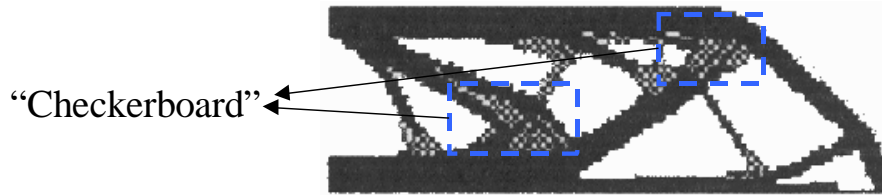


Fig.8.5.1: Tabuleiro de xadrez.

Ou seja, aparecem elementos de cor preta (material sólido) próximo a elementos de cor branca (“ar”) numa disposição de tabuleiro de xadrez. Isso ocorre devido a formulação de MEF usada, mas pode ser corrigida.

Outro ponto a ser analisado é a dependência da discretização da malha de MEF. A Fig.8.5.2 ilustra a dependência do resultado de OT para diferentes discretizações de malha.



Fig.8.5.2: Dependência da discretização da malha de MEF.

Essa dependência está relacionada com o modelo de material empregado como já discutido acima.

Finalmente, um ponto importante a ser analisado não somente em OT mas em qualquer otimização, é a curva de convergência da função objetivo e das restrições. A Fig.8.5.3 ilustra uma curva de convergência do tipo monotônica. Dependendo do tipo de função objetivo adotada (por exemplo, frequência de ressonância) essa curva pode variar de forma, no entanto a presença de oscilações na parte "assintótica" da curva indica que alguns parâmetros iniciais da otimização devem ser alterados.

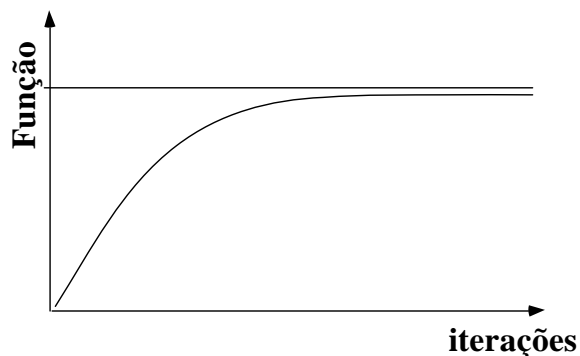


Fig.8.5.3: Exemplo de curva de convergência da função objetivo.

8.6 Áreas de Atuação

A seguir são apresentados alguns exemplos que ilustram o potencial da OT.

O primeiro exemplo consiste na otimização da viga mostrada na Fig.8.6.1a para máxima rigidez e menor volume. Dois métodos são usados, a otimização de forma e otimização topológica. O resultado obtido com a otimização de forma é apresentado na Fig.8.6.1b. Apresenta uma redução de 5% em relação ao volume inicial. Nota-se uma grande redução de espessura com conseqüente deformação da malha em alguns pontos da estrutura, não sendo possível realizar o remalhamento e continuar a otimização. O resultado de OT é mostrado na Fig.8.6.1c e a sua interpretação final na Fig.8.6.1d. A estrutura final obtida apresenta uma redução de volume de 42% em relação ao volume inicial e uma deflexão 7% menor do que a configuração inicial!!

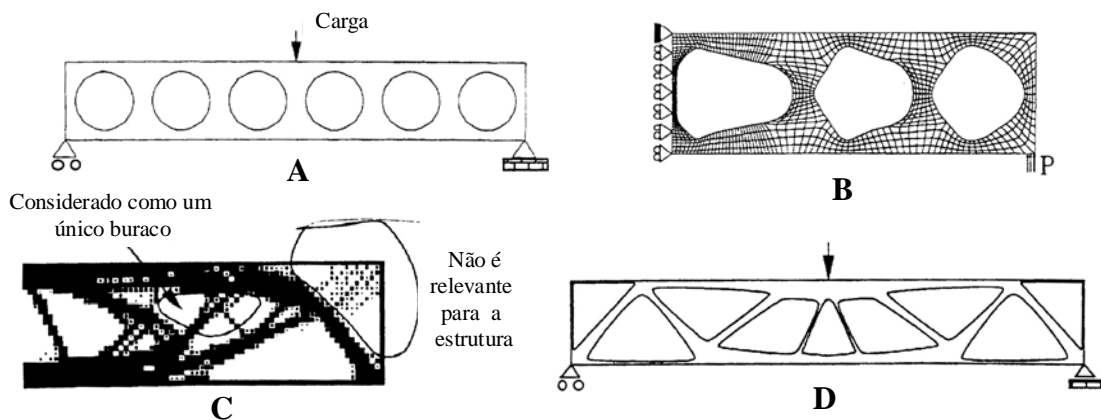


Fig.8.6.1: Otimização de uma viga para máxima rigidez e menor peso.

A tabela 8.6.1 mostra os resultados de volume, deslocamento e máxima tensão de Von Mises. Note que a tensão máxima no resultado final é apenas 4,5% maior do que no inicial.

Tabela 8.6.1: Volume, deslocamento e tensão máxima para as estruturas apresentadas na Fig.8.6.1.

| | Fig. | Volume | Deflection | Max. stress |
|----------------------------|------|--------|------------|-------------|
| Initial, infeasible design | A | 1.07 | 10.1 | 292 |
| Optimal circular holes | | 1.10 | 9.4 | 248 |
| Optimal boundary of holes | B | 1.02 | 9.4 | 372 |
| Optimal topology | C | 1.10 | 6.0 | 227 |
| Final design | D | 0.62 | 9.4 | 305 |

O próximo exemplo ilustra a reprodução do problema resolvido por Michell em 1904 mostrado na seção 1.1. Esse problema é considerado uma "benchmark" para os softwares de OT e demonstra que o método de OT é matematicamente correto. A Fig.8.6.2a ilustra o domínio inicial, a Fig.8.6.2b o resultado de OT e a Fig.8.6.2c o resultado analítico obtido por Michell.

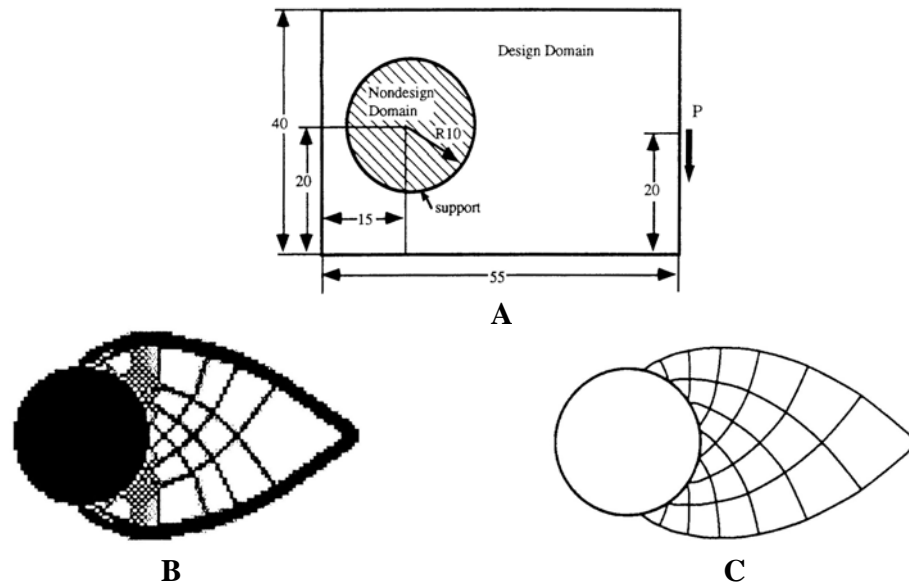


Fig.8.6.2: Reprodução do resultado de Michell.

A Fig.8.6.3 ilustra outra aplicação da OT que é o cálculo do reforçamento ótimo de chapas e cascas de forma a resultar na máxima rigidez com o menor volume de reforçamento. O exemplo é um capô de automóvel (Bendsoe e Kikuchi, 1988), onde a área escura indica a presença do reforçamento enquanto a área clara indica a espessura normal da casca. O reforçamento pode ser implementado soldando-se vigas em "U" no capô, por exemplo.

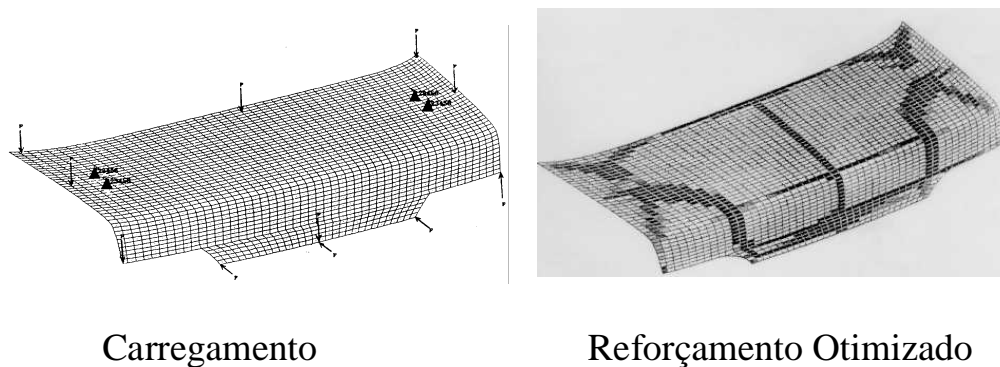


Fig.8.6.3: Reforçamento ótimo de um capô de automóvel.

Além da minimização de flexibilidade com o menor volume de material, ainda na área mecânica estrutural, outras funções objetivo foram implementadas como maximização da frequência de ressonância (ou carga de flambagem), minimização da resposta em frequência, maximização da absorção de energia de impacto, etc. Algumas dessas funções objetivos já estão disponíveis nos softwares comerciais de OT.

Nos últimos cinco anos a OT foi expandida na área acadêmica para outras áreas da engenharia. Assim, a OT foi implementada para o projeto de: mecanismos flexíveis, ou seja, mecanismos onde o movimento é dado pela flexibilidade da estrutura ao invés da presença de juntas e pinos; atuadores piezoelétricos; dispositivos eletromagnéticos, onde o objetivo é maximizar a relação torque/volume nos motores elétricos ou a emissividade e receptividade de antenas; e projeto de materiais compostos com propriedades desejadas. As pesquisas em OT se concentram nesses tópicos atualmente.

9. SOFTWARES DE OTIMIZAÇÃO COMERCIAIS DISPONÍVEIS MAIS USADOS EM ENGENHARIA

Nesse capítulo são apresentados os mais comumente usados softwares de otimização utilizados em engenharia disponíveis no mercado e suas principais áreas de atuação para conhecimento geral do leitor.

- TSO (WASP) - projeto preliminar da asa e cauda de aviões com restrições aeroelásticas combina análise estrutural (modelos simplificados) e otimização (método de penalização interior);
- ACCESS, FASTOP, OPSTAT, OPTCOMP, OPTIMUM, ASOP, STARS, DESAP - integram algoritmos de MEF específicos com otimização;
- OPTSYS - integra os programas de MEF genérico ABAQUS e ASKA com algoritmos de otimização;
- ASTROS - software de otimização estrutural baseado no NASTRAN;
- NISAOPT (SHAPE e STROPT) - baseado no NISA II;
- ADS, DOT e DOC (empresa VMA) - derivados do CONMIN. DOT é uma coleção de subrotinas em FORTRAN para otimização. DOC é um programa de controle que facilita o uso do DOT na solução de problemas com multiobjetivos;
- NEWSUMT - pacote de otimização baseado em métodos de penalização;
- OPT - software de otimização baseado no algoritmo de gradientes reduzidos;
- IDESIGN - software de otimização baseado no PQS (ou “SQP”);
- NASTRAN, I-DEAS, ANSYS - possuem algoritmos de otimização, cálculo de sensibilidade, e otimização topológica baseado no método das densidades;
- GENESYS - apresenta o MEF já integrado com algoritmos de otimização, incluindo otimização topológica;
- OPSTRUCT (empresa ALTAIR - EUA) e OPTISHAPE (empresa QUINT - Japão) - softwares de otimização topológica baseados no método de homogeneização;

10. Bibliografia

[1] Haftka , R. T., "*Elements of Structural Optimization*", Solid Mechanics and its Applications, Kluwer Academic Publishers, Dordrecht, Netherlands, 1993.

[2] Vanderplaats, G. N., "Numerical Optimization Techniques for Engineering Design With Applications", McGraw-Hill, Inc., NY, USA, 1984.