Otimização Topológica Evolucionária de Estruturas Periódicas: aplicação no projeto Mecanismos Flexíveis

Palavras-Chave: Otimização topológica, Estruturas periódicas, Mecanismos flexíveis

Autores:
Thiago Henrique Martins da Silva, FEM/Unicamp
Prof. Dr. Renato Pavanello (orientador), DMC/FEM/Unicamp

---

1 Introdução

Nas últimas décadas, o avanço da análise computacional tem permitido a solução de problemas complexos de engenharia, integrando conhecimentos teóricos clássicos com alta capacidade de processamento. Nesse contexto, a otimização topológica ganhou destaque a partir dos anos 1980 como uma ferramenta eficiente para o projeto estrutural.

Mecanismos flexíveis são estruturas capazes de gerar um movimento desejado ao sofrerem deformação elástica, sem que sejam empregadas ligações e juntas, como ocorre nos mecanismos de corpos rígidos. Esses mecanismos são aplicáveis no ramo da engenharia de precisão, especialmente em campos que envolvem micromanipulação e microposicionamento. Isso se deve principalmente à natureza monolítica dos mecanismos flexíveis que, como consequência, leva a vantagens como a desnecessidade de montagem, a ausência de atrito (e, portanto, de lubrificação), bem como a ausência de folgas e consequente redução de vibrações em funcionamento [1]. A otimização topológica traz benefícios para o projeto desses mecanismos, visto que pode ser utilizada para obter geometrias ótimas em termos da força aplicada e do deslocamento resultante.

2 Metodologia

Na primeira etapa do projeto, realizou-se a revisão bibliográfica, com a leitura de Lima [2] a fim de compreender de maneira geral o problema de otimização topológica para mecanismos flexíveis, assim como as ferramentas matemáticas utilizadas na sua realização. Em seguida, um código para aproximação por elementos finitos das estruturas com comportamento estático foi realizado seguindo o método proposto em Kwon et al. [3].

2.1 Método dos Elementos Finitos

O elemento finito utilizado na análise segue o modelo quadrilateral bilinear, com quatro nós. Após o cálculo da matriz de rigidez de cada elemento, definiu-se a malha com o respectivo padrão de numeração dos nós e elementos. Com a matriz de incidência, foi possível montar a matriz global de rigidez da estrutura:

$$\boldsymbol{K} = \sum_{i=1}^{N} \boldsymbol{K}_i \tag{1}$$

A resolução do problema é feita por meio da aplicação das condições de contorno e do carregamento, resolvendo-se o sistema linear:

$$\boldsymbol{K}\boldsymbol{u} = \boldsymbol{f} \tag{2}$$

em que $\boldsymbol{u}$ é o vetor de deslocamentos nodais e $\boldsymbol{f}$ o vetor de forças nodais aplicadas.

2.2 Método BESO

Para o desenvolvimento do código BESO foi utilizado o método descrito em Huang et al, [4]. O programa em MATLAB foi desenvolvido tendo como base o código proposto por Sigmund [5], sendo realizadas algumas alterações. A principal alteração consiste na implementação do código do método dos Elementos Finitos descrito anteriormente, o que envolveu a adaptação de algumas variáveis e matrizes no código. Além disso, foi implementado um filtro de sensibilidades utilizando a Toolbox de processamento de imagem do MATLAB como proposto em [6]. A seguir, é apresentada uma breve explicação do procedimento adotado.

O problema de otimização consiste em satisfazer as seguintes condições:

$$\begin{aligned} \text{Minimizar: } C &= \frac{1}{2}\boldsymbol{f}^t\boldsymbol{u} \\ \text{Sujeito a: } V^* - \sum_{i=1}^{N} V_i x_i &= 0 \\ x_i = x_{min} \text{ ou } x_i &= 1 \end{aligned} \tag{3}$$

onde V_i é o volume do elemento i, enquanto V^* é o volume prescrito da estrutura final. A variável binária de projeto x_i indica a densidade relativa do elemento i e assume um valor mínimo $x_{\min}$ (por exemplo, 0.001) para representar os elementos vazios.

As sensibilidades dos elementos podem ser calculadas empregando o método BESO soft-kill, o qual está baseado no método de interpolação de material com penalização p, geralmente igual a 3. Efetuando as manipulações necessárias, obtém-se a seguinte expressão para a sensibilidade do i-ésimo elemento:

$$\alpha_i = -\frac{1}{p}\frac{\partial C}{\partial x_i} = \begin{cases} -\frac{1}{2}\boldsymbol{u}_i^T\boldsymbol{K}_i\boldsymbol{u}_i & \text{para } x_i = 1 \\ -\frac{1}{2}x_{min}^{p-1}\boldsymbol{u}_i^T\boldsymbol{K}_i\boldsymbol{u}_i & \text{para } x_i = x_{min} \end{cases} \tag{4}$$

Após a filtragem das sensibilidades de cada elemento, a última etapa do procedimento de otimização corresponde à aplicação do critério de adição e remoção de elementos. Conforme apresentado em Sigmund [5], utiliza-se as sensibilidades e os volumes-alvo de cada iteração, calculados nas etapas anteriores, para determinar o novo conjunto de densidades de material. Como critério de parada, utiliza-se uma equação que impõe que a diferença entre o somatório da função objetivo nas últimas 5 iterações e o somatório das 5 iterações anteriores, normalizada pelo somatório das últimas 5 iterações, seja menor que um parâmetro de tolerância τ.

2.3 Método BESO para estruturas periódicas

Na etapa seguinte, realizou-se a implementação do problema de otimização para estruturas periódicas, descrito a seguir. Denotando-se por i o índice da célula periódica e por j o índice do elemento finito dentro de cada célula, o problema pode ser formulado como:

$$
\begin{aligned}
\text{Minimizar:} \quad & C = \frac{1}{2}\boldsymbol{f}^T\boldsymbol{u} \\
\text{Sujeito a:} \quad & V^* - mV_i = 0 \\
& V_i = \sum_{j=1}^{n} V_{i,j}\, x_{i,j} \\
& x_{1,j} = x_{2,j} = \cdots = x_{m,j} \\
& x_{i,j} \in \{x_{\min},\, 1\}, \quad \text{para } j = 1, \ldots, n,\ i = 1, \ldots, m
\end{aligned} \tag{5}
$$

No contexto do método *soft-kill*, a sensibilidade da função objetivo em relação à variável de projeto x_j é dada por:

$$
\alpha_j = -\frac{1}{p}\frac{dC}{dx_j} = \frac{1}{2}\sum_{i=1}^{m} x_{i,j}^{p-1}\, \boldsymbol{u}_{i,j}^T\, \boldsymbol{K}_{i,j}^0\, \boldsymbol{u}_{i,j} \tag{6}
$$

Os mesmos procedimentos de filtragem, adição e remoção de material e critério de parada explicados anteriormente são novamente utilizados para a obtenção da estrutura final.

2.4 Método BESO para Mecanismos Flexíveis

O projeto de mecanismos flexíveis com otimização topológica exige um equilíbrio entre flexibilidade (movimento) e rigidez (resistência). A formulação do problema envolve múltiplas funções objetivo para capturar esse comportamento multiobjetivo essencial à performance do mecanismo. [7]

Considere um domínio de projeto onde uma força F_{in} é aplicada na entrada de um mecanismo flexível, com um deslocamento esperado u_{out} na saída. Uma mola de constante de rigidez k_s é conectada ao porto de saída para representar a interação com a carga externa, gerando uma força de saída $F_{\text{out}} = k_s u_{\text{out}}$.

A performance do mecanismo pode ser avaliada pela sua flexibilidade (u_{out}), e sua rigidez estrutural, representada pela compliance média:

$$C = \frac{1}{2}F_{\text{in}}u_{\text{in}} - \frac{1}{2}F_{\text{out}}u_{\text{out}} = \frac{1}{2}F_{\text{in}}u_{\text{in}} - k_s u_{\text{out}}^2 \tag{7}$$

Finalmente, dadas as condições de contorno, o método BESO pode ser novamente aplicado para obter o mecanismo otimizado.

3 Resultados e discussão

O problema analisado para a validação do código desenvolvido consiste em uma viga engastada em ambas as extremidades com uma força $\boldsymbol{P}$ de 20 kN aplicada no seu centro. O deslocamento total máximo obtido no código MATLAB foi idêntico ao valor obtido no programa de simulação para engenharia ANSYS. Na Figura 1, são mostrados os deslocamentos totais na estrutura deformada obtidos pelo código MATLAB. Os resultados permitiram concluir a validação do código.

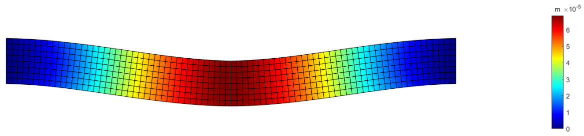
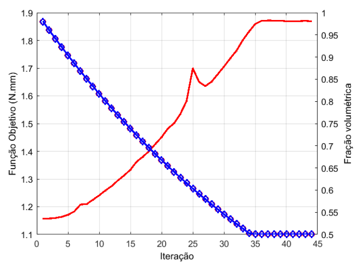


Figura 1: Deslocamento Total (em metros), obtida no código MATLAB

Para a validação código de implementação do método BESO, os resultados obtidos foram comparados àqueles obtidos por Huang et al. [4] para determinados problemas. A seguir, é apresentado um dos exemplos resolvidos. Este exemplo trata da otimização da rigidez de uma viga curta com uma das extremidades engastada enquanto uma força é aplicada no centro da extremidade livre. Na Figura 2, apresentam-se os resultados do problema de otimização obtido pelo código BESO desenvolvido. Conclui-se que a topologia obtida é idêntica à apresentada em Huang et al. [4], logo, o código pôde ser validado.

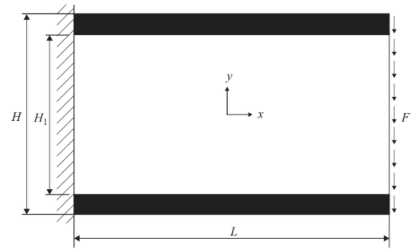
(a) Topologia otimizada obtida pelo código BESO



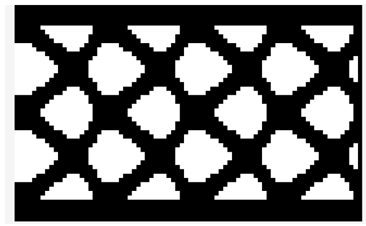
(b) Evolução da otimização topológica

Figura 2: Método BESO aplicado

Para estruturas periódicas, alguns problemas foram resolvidos, incluindo o apresentado na Figura 3a. A otimização levou à estrutura apresentada na Figura 3b. O resultado assemelha-se ao apresentado em Huang et al. [4].



(a) Problema a ser resolvido (L = 32 m, H = 20 m e H_1 = 16 m)



(b) Topologia otimizada obtida pelo código BESO

Figura 3: Método BESO aplicado

4 Conclusão

A partir da metodologia adotada, foi possível alcançar uma compreensão sólida da otimização topológica aplicada a diferentes tipos de estruturas. No entanto, não se obteve sucesso no desenvolvimento de um método que convergisse de forma eficaz ao combinar, em uma única formulação, as abordagens para estruturas periódicas e mecanismos flexíveis. A periodicidade só pôde ser implementada por meio da justaposição de elementos previamente otimizados de forma independente, o que se mostrou insuficiente para garantir a funcionalidade da maioria dos mecanismos analisados. Conclui-se, portanto, que integrar essas duas abordagens representa um desafio significativo, mas também uma oportunidade promissora para explorar aplicações ainda pouco investigadas dos mecanismos flexíveis.

Referências

[1] Zhang, X., Zhu, B. Topology Optimization of Compliant Mechanisms. Alemanha: Springer Nature, 2018.

[2] Lima, V. Otimização Topológica Evolucionária de Mecanismos Flexíveis Atuados por Fluido, Tese (Mestrado em Engenharia Mecânica) - Universidade Estadual de Campinas. Campinas, p. 72. 2024.

[3] Kwon, Y. W., Bang, H. The Finite Element Method using MATLAB. CRC Press, 1996.

[4] Huang, X., Xie, M. Evolutionary Topology Optimization of Continuum Structures: Methods and Applications. Alemanha: Wiley, 2010.

[5] Sigmund, O. (2001). A 99 line topology optimization code written in MATLAB. Struct. Multidisc. Optim. 21:120–7.

[6] FERRARI, Federico; SIGMUND, Ole. A new generation 99 line Matlab code for compliance topology optimization and its extension to 3D. Structural and Multidisciplinary Optimization, [S.l.], v. 65, p. 63–79, 2022. DOI: https://doi.org/10.1007/s00158-022-03197-x.

[7] HUANG, X.; ZHANG, X.; XIE, Y. M. Topology optimization of compliant mechanisms with desired structural stiffness. Engineering Structures, [S.l.], v. 79, p. 13–21, 2014. DOI: https://doi.org/10.1016/j.engstruct.2014.08.007.