

GERAÇÃO E OTIMIZAÇÃO DE MALHAS ESTRUTURADAS SOBRE DOMÍNIOS BIDIMENSIONAIS ARBITRÁRIOS

Márcio Ricardo Pivello

Faculdade de Engenharia Mecânica, Universidade Federal de Uberlândia, Uberlândia – MG.
e-mail: pivello@mecanica.ufu.br

Carlos Roberto Ribeiro

Faculdade de Engenharia Mecânica, Universidade Federal de Uberlândia, Uberlândia – MG.
e-mail: crribeiro@mecanica.ufu.br

Resumo. *Este trabalho apresenta uma metodologia para geração e suavização de malhas estruturadas sobre domínios bidimensionais arbitrários. O método consiste em dividir o domínio em áreas ou subdomínios quadrilaterais, chamados de blocos, e discretizá-los individualmente para depois conectar a malha entre blocos vizinhos. Uma formulação paramétrica da spline cúbica foi empregada tanto na caracterização do domínio como na aplicação das condições de contorno. A geração da malha foi feita através de um sistema acoplado de equações elípticas de Laplace, obtidas a partir do mapeamento conforme dos blocos. O processo de suavização consiste em realocar os nós de fronteira de cada bloco de modo a se obter elementos geometricamente regulares, tendo como variáveis de projeto as coordenadas paramétricas destes nós. Estas servem como condições de contorno de Dirichlet para o sistema gerador de malhas, que realocará os nós internos ao domínio. Para a continuidade da malha entre blocos adjacentes foram testados dois métodos, fornecendo continuidade C^0 ou C^1 , dependendo das necessidades do usuário.*

Palavras chave: *Malhas estruturadas, suavização de malhas, domínios bidimensionais arbitrários.*

1. REPRESENTAÇÃO DO DOMÍNIO POR SPLINE CÚBICA

A formulação paramétrica da fronteira utilizada neste trabalho baseia-se na analogia da viga de Euler-Bernoulli (Rogers, 1990). Segundo esta formulação, chamada de "spline" cúbica, as coordenadas de um ponto sobre uma curva representada por uma spline são dadas por

$$[x(t) \ y(t)] = [T][N][G_x \ G_y] \quad (1)$$

onde

$$[T] = [t^3 \ t^2 \ t \ 1], \quad 0 \leq t \leq 1 \quad (2)$$

$$[N] = \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \quad (3)$$

$$[G_x \ G_y]^T = \begin{bmatrix} x_k & x_{k+1} & x'_k & x'_{k+1} \\ y_k & y_{k+1} & y'_k & y'_{k+1} \end{bmatrix} \quad (4)$$

os valores de x_k , $k=1$ a n , são obtidos para uma spline natural resolvendo-se o sistema:

$$\begin{bmatrix} P_1' \\ P_2' \\ P_3' \\ \vdots \\ \vdots \\ \vdots \\ P_n' \end{bmatrix} = \begin{bmatrix} 1 & \frac{1}{2} & 0 & 0 & \cdot & \cdot & \cdot & 0 \\ 0 & 4 & 1 & 0 & 0 & \cdot & \cdot & 0 \\ 0 & 1 & 4 & 1 & 0 & \cdot & \cdot & 0 \\ \cdot & \cdot \\ \cdot & \cdot \\ \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & 0 & 1 & 4 & 1 \\ \cdot & \cdot & \cdot & \cdot & 0 & 0 & 2 & 4 \end{bmatrix}^{-1} \begin{bmatrix} 1.5(P_2 - P_1) \\ 3\{(P_2 - P_1) + (P_1 - P_0)\} \\ 3\{(P_3 - P_2) + (P_2 - P_1)\} \\ \cdot \\ \cdot \\ \cdot \\ 3\{(P_n - P_{n-1}) + (P_{n-1} - P_{n-2})\} \\ 6(P_n - P_{n-1}) \end{bmatrix} \quad (5)$$

onde P' tem componentes x e y.

2. GERADOR ELÍPTICO DE MALHAS ESTRUTURADAS

Uma das técnicas de geração de malhas estruturadas utiliza sistemas de equações diferenciais parciais elípticas (EDPEs). A principal vantagem desta técnica é que as soluções destes sistemas são funções harmônicas (ou seja, suas derivadas parciais de segunda ordem são contínuas) cujos valores máximo e mínimo ocorrem nas fronteiras. Este fato, além de garantir que os pontos de máximo e mínimo não anularão o Jacobiano da transformação, também garante a unicidade das soluções, isto é, duas superfícies coordenadas de mesmo valor nunca se interceptarão, o que é requisito básico para a geração de malhas estruturadas. Além disso, todos os problemas de campo são regidos por estas equações e possuem como soluções, portanto, iso-superfícies que podem ser empregadas como superfícies coordenadas (Maliska, 1995).

O método empregado neste trabalho consiste em fazer um mapeamento conforme – ou mudança de variáveis – no problema elíptico a ser resolvido, de modo que qualquer geometria quadrilateral seja transformada em um retângulo num plano computacional. Sejam ξ e η as variáveis dependentes antes do mapeamento e x e y as independentes. Após a aplicação da lei de mapeamento, ocorre uma troca de variáveis dependentes, e o par (x,y) passa a ser dependente de (ξ,η) . Assim, o sistema de equações dado pelas eqs. 6 e 7 é resolvido no plano computacional, e as iso-soluções são obtidas já no plano físico.

$$\alpha x_{\xi\xi} + \gamma x_{\eta\eta} - 2\beta x_{\xi\eta} = 0 \quad (6)$$

$$\alpha y_{\xi\xi} + \gamma y_{\eta\eta} - 2\beta y_{\xi\eta} = 0 \quad (7)$$

onde

$$\begin{aligned} \alpha &= x_\eta^2 + y_\eta^2 \\ \gamma &= x_\xi^2 + y_\xi^2 \\ \beta &= x_\xi x_\eta + y_\xi y_\eta \end{aligned} \quad (8)$$

Com relação às condições de contorno, uma possibilidade é especificar os valores de x e y na fronteira Γ em função de ξ e η (condições de Dirichlet). Outra abordagem seria impor derivada direcional nula nas fronteiras (condições de Neumann), que garantiriam a ortogonalidade do sistema coordenado. Como isto aumenta o custo computacional do problema, foram adotadas condições de Dirichlet, isto é:

$$\text{em } \Gamma_i : \begin{cases} x = f_i(t) \\ y = g_i(t) \end{cases} \quad i = 1,2,3,4 \quad (9)$$

onde

$$\Gamma = \bigcup_{i=1}^4 \Gamma_i \quad (10)$$

e $f_i(t)$ e $g_i(t)$ são as funções que definem as fronteiras do domínio em função do parâmetro t . Neste caso, foram empregadas splines cúbicas, conforme formulação da seção anterior.

3. SUAVIZAÇÃO DA MALHA

O processo de suavização da malha foi tratado como um problema de pseudo-otimização, que busca a minimização da deformação dos elementos da fronteira do domínio. Os nós localizados sobre a fronteira do domínio constituem as variáveis de projeto, e o critério de deformação adotado consiste em medir o desvio da forma dos elementos em relação a um quadrado. As equações para quantificação desta deformação foram determinadas empiricamente. O processo consiste em medir a variação do erro para cada elemento, e com base neste valor alterar localmente a posição dos nós do elemento, impondo uma perturbação na sua coordenada paramétrica. Como o emprego do gerador elíptico pressupõe que o domínio seja quadrilateral, domínios complexos devem ser decompostos em subdomínios quadrilaterais. Malhas desse tipo são normalmente chamadas na literatura de malhas multi bloco ou malhas estruturadas por bloco. Além disso, deve-se garantir a continuidade da malha sobre as faces compartilhadas pelos subdomínios adjacentes. Esta continuidade pode ser de dois tipos:

- De ordem C^0 , onde se garante apenas a coincidência dos nós nas fronteiras, impondo-se as mesmas condições de Dirichlet para os blocos adjacentes;
- De ordem C^1 , onde, além da coincidência destes nós exige-se também a continuidade da primeira derivada das linhas ao cruzar a fronteira. Nesta abordagem impõem-se condições de Neumann para os blocos adjacentes. Neste caso, também é permitido ao algoritmo reposicionar as faces internas ao domínio.

No caso de um domínio quadrilateral simples, o domínio Ω é modelado como um objeto computacional, composto pelas seguintes propriedades:

- Fronteira Γ , composta por quatro curvas splines cúbicas ($\Gamma_s, \Gamma_E, \Gamma_N, \Gamma_W$, em referência a sul, leste, norte e oeste), que serão chamadas de faces;
- Conjunto de nós internos (x,y) no interior do domínio;
- Quantidade de elementos presentes em cada face do domínio (m);
- Conjunto de condições de contorno definidas parametricamente sobre Γ pelo vetor $\{t\} = \{t_1, t_2, t_3, \dots, t_{4m}\}$;
- Um valor de mérito E , que quantifica a distorção dos elementos localizados na fronteira Γ .

Para um nó genérico de fronteira k ($1 \leq k \leq 4m$), sua coordenada paramétrica t_k é dada por

$$t_k = k + g_k \quad (11)$$

onde k representa a parte inteira da coordenada e g_k sua parte decimal. Por exemplo, para $t = 3.8$, tem-se $k = 3$ e $g_k = 0.8$. Como a malha inicial é gerada com os nós igualmente espaçados na fronteira, na primeira iteração adota-se $g_k = 0$. A partir daí, inicia-se um processo de busca pelo conjunto ótimo de condições de contorno que minimize a distorção geométrica dos elementos de fronteira, alterando o valor de g . Formula-se então o seguinte problema de otimização:

Dados:

- Um domínio bidimensional Ω , limitado pela fronteira $\Gamma = \Gamma_S \cup \Gamma_E \cup \Gamma_N \cup \Gamma_W$;

- Uma Malha \mathbf{M} , caracterizada por:
 - conjunto de nós internos (\mathbf{x}, \mathbf{y}) no interior do domínio;
 - A quantidade de elementos presentes em cada face do domínio (m);
 - conjunto de parâmetros $\{\mathbf{t}\} = \{\mathbf{t}_1, \mathbf{t}_2, \mathbf{t}_3, \dots, \mathbf{t}_{4m}\}$ que definem as condições de contorno sobre a fronteira;
 - Um valor de mérito \mathbf{E} , que quantifica a distorção dos elementos localizados na fronteira Γ ;

Obter:

Conjunto de condições de contorno $\{\mathbf{t}^*\}$ que minimize \mathbf{E} .

Na seqüência de descrição do algoritmo, apresenta-se primeiramente a formulação do critério de otimização, e depois o método de busca.

Uma vez que todos os elementos são quadrilaterais, o critério adotado neste trabalho consiste em medir a distorção de cada elemento da fronteira em relação a um quadrado, e o desempenho da malha será medido pela soma destas distorções. Para o elemento genérico da figura 1, a tabela 1 mostra as condições adotadas neste trabalho que determinam o desvio em relação à forma considerada ideal, bem como as equações que quantificam este desvio.

Tabela 1: Condições a serem satisfeitas por um elemento quadrilateral genérico

Condição	Equação
1. $A = F$	$a = (x_0 - x_1) - (x_2 - x_3) = 0$ (12a)
2. $D = I$	$b = (y_0 - y_1) - (y_2 - y_3) = 0$ (12b)
3. $D = F$	$c = (y_1 - y_3) - (x_2 - x_3) = 0$ (12c)
4. $C = G$	$d = (x_1 - x_3) - (y_3 - y_2) = 0$ (12d)

O erro associado ao elemento será dado por:

$$\varepsilon = a^2 + b^2 + c^2 + d^2 \quad (13)$$

Note que este critério é necessário mas não suficiente para se obter um elemento quadrado. Para isso seriam necessários testes de ortogonalidade, que aumentariam o custo computacional em até 20%, segundo alguns cálculos preliminares. Como a qualidade da malha obtida foi praticamente a mesma para todos os casos, optou-se por um critério menos rigoroso, a favor do desempenho computacional.

O método de busca por gradiente usado é baseado no método da máxima descida. As variáveis de projeto são as coordenadas paramétricas dos nós de fronteira do domínio, e as variáveis dependentes são as distorções dos elementos que definem a fronteira. Portanto, interessa agora obter uma relação entre a variação do posicionamento destes nós e a distorção destes elementos, quantificada pelas eqs. 12.

A atualização de t_k na iteração i é dada por

$$t_k^{(i)} = \left(t_k - \Delta t_k \cdot \left(\frac{\partial \mathbf{E}}{\partial t_k} \right) \right)^{(i-1)} \quad (14)$$

Substituindo-se a eq. 14 na eq. 11, obtém-se a expressão para a atualização de $g_k^{(i)}$, que é a perturbação imposta à coordenada paramétrica do nó k na iteração i e \mathbf{E} é a distorção ou erro de todos os elementos da malha analisada.

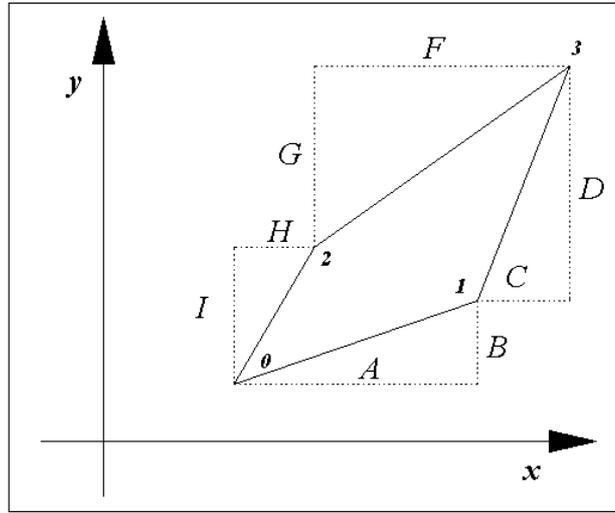


Figura 1: Desvio de forma para um elemento quadrilátero genérico.

$$\mathbf{g}_k^{(i)} = \left(\mathbf{g}_k - \Delta \mathbf{g}_k \cdot \left(\frac{\partial \mathbf{E}}{\partial \mathbf{t}_k} \right) \right)^{(i-1)} \quad (15)$$

Sendo o desvio em cada elemento dado pela equação 13, o erro total da malha será

$$\mathbf{E} = \sum_{e=1}^{4m} (a_e^2 + b_e^2 + c_e^2 + d_e^2) = S_a + S_b + S_c + S_d \quad (16)$$

onde

$$S_a = \sum_{e=1}^{4m} (a_e^2), \quad S_b = \sum_{e=1}^{4m} (b_e^2), \quad S_c = \sum_{e=1}^{4m} (c_e^2) \quad \text{e} \quad S_d = \sum_{e=1}^{4m} (d_e^2)$$

A variação do erro é calculada considerando-se que apenas os nós da fronteira estejam livres, uma vez que os nós interiores serão modificados pelo gerador elíptico (nesta etapa, os nós que definem vértices não serão alterados). Então, as variáveis de projeto serão as coordenadas dos nós, que podem ser representadas parametricamente pelo parâmetro t . As componentes do gradiente do erro serão calculadas pela eq.(17).

$$\frac{\partial \mathbf{E}}{\partial \mathbf{t}_k} = \frac{\partial S_a}{\partial \mathbf{t}_k} + \frac{\partial S_b}{\partial \mathbf{t}_k} + \frac{\partial S_c}{\partial \mathbf{t}_k} + \frac{\partial S_d}{\partial \mathbf{t}_k} \quad (17)$$

onde

$$\frac{\partial S_a}{\partial \mathbf{t}_k} = 2 \cdot \frac{\partial x_k}{\partial \mathbf{t}_k} \{a_k - a_{k-1}\} \quad (18a)$$

$$\frac{\partial S_b}{\partial \mathbf{t}_k} = 2 \cdot \frac{\partial y_k}{\partial \mathbf{t}_k} \{b_k - b_{k-1}\} \quad (18b)$$

$$\frac{\partial \text{Sc}}{\partial t_k} = 2 \cdot \frac{\partial y_k}{\partial t_k} \cdot c_k \quad (18c)$$

$$\frac{\partial \text{Sd}}{\partial t_k} = 2 \cdot \frac{\partial x_k}{\partial t_k} \cdot d_k \quad (18d)$$

e a_k , b_k , c_k e d_k são dados pelas equações 12.

O gradiente do erro para cada elemento será dado então por

$$\nabla \mathbf{E}_k = 2 \cdot \left\{ \frac{\partial x_k}{\partial t_k} \cdot [a_k - a_{k-1} + d_k] + \frac{\partial y_k}{\partial t_k} \cdot [b_k - b_{k-1} + c_k] \right\} \quad (19)$$

O cálculo de $\frac{\partial x_k}{\partial t_k}$ e $\frac{\partial y_k}{\partial t_k}$ é feito derivando-se as eqs. 1 e 2 em relação a t .

Domínios complexos são modelados como a união de blocos, definidos pelo usuário e discretizados independentemente pelo algoritmo descrito acima. A continuidade da malha nas faces comuns aos subdomínios adjacentes é obtida aplicando-se o mesmo conjunto de condições de contorno às fronteiras comuns. Quando a forma destas faces não é importante para a análise do problema, elas são tratadas como isolinhas, de modo que o algoritmo possa reposicionar livremente os nós localizados sobre elas, minimizando assim a distorção dos elementos na fronteira comum. Em alguns casos, o modelo pode ter não somente faces internas mas também vértices internos ao domínio, que também devem ser realocados durante a alteração das faces na otimização da malha. A condição para que um vértice seja interno ao domínio é que ele seja compartilhado por pelo menos 3 blocos, e cada um destes blocos tenha dois vizinhos neste conjunto. Com base neste critério, foi formulada uma regra empírica para o reposicionamento de vértices internos ao domínio, ponderando a posição deste ponto pela posição dos vizinhos. Atribui-se peso 3 para os nós localizados *sobre* as faces que o compartilham ($x_1^{(i)}$ e $x_2^{(i)}$) e peso 1 para os nós vizinhos imediatos *interiores ao domínio* ($x_3^{(i)}$). A equação 20 mostra como são reposicionados os vértices internos.

$$x_p = \frac{\sum_{i=1}^{n_blocos} [3 \cdot (x_1^{(i)} + x_2^{(i)}) + x_3^{(i)}]}{7 \cdot n_blocos} \quad (20)$$

$$y_p = \frac{\sum_{i=1}^{n_blocos} [3 \cdot (y_1^{(i)} + y_2^{(i)}) + y_3^{(i)}]}{7 \cdot n_blocos}$$

onde n_blocos é o número de blocos que compartilham o vértice interno.

O algoritmo para geração e otimização de uma malha sobre um domínio bidimensional arbitrário pode ser assim resumido:

1. Definir a fronteira parametricamente a partir dos pontos de suporte fornecidos pelo usuário.
2. Para cada bloco, identificar seus vizinhos, faces internas e vértices internos ao domínio.
3. Aplicar um método de continuidade de malha.
4. Resolver o sistema de equações geradoras.
5. Realocar os vértices internos, se houverem.

6. Atribuir mérito à malha.
7. Voltar ao passo 3 até a convergência.

4. RESULTADOS

Serão apresentados dois casos de domínios multiplamente conexos, mostrando o domínio e sua divisão em blocos, respectivamente (fig. 3). As figuras 4 e 5 mostram as malhas geradas sobre cada um dos domínios com continuidade entre blocos do tipo Dirichlet (fig. superior) e Neumann (fig. inferior). Para cada caso são apresentados o tempo de execução (em segundos), o número de iterações e o erro máximo obtido. Estes dados são mostrados nas tabelas 2 e 3.

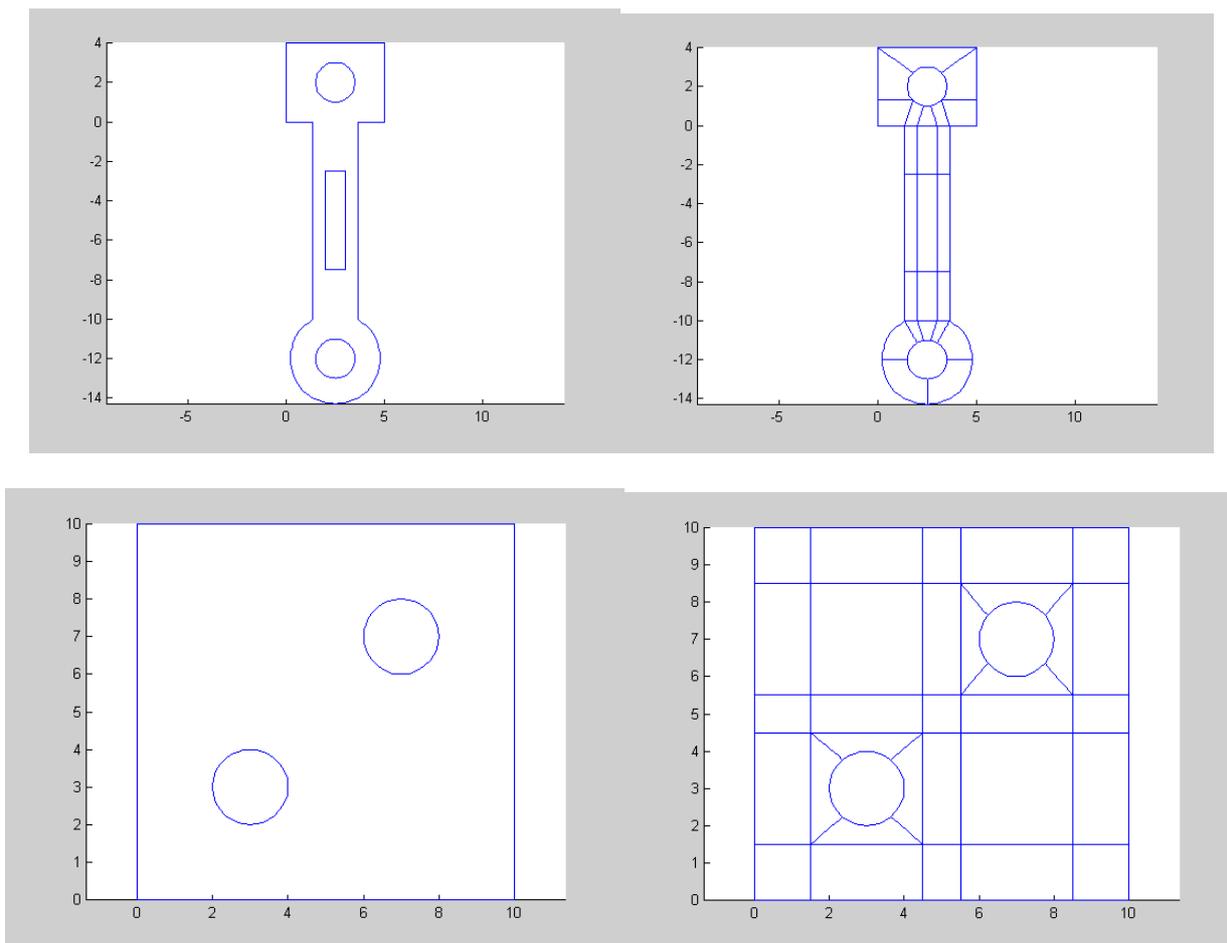


Figura 3: Domínio e configuração dos blocos nos dois casos estudados.

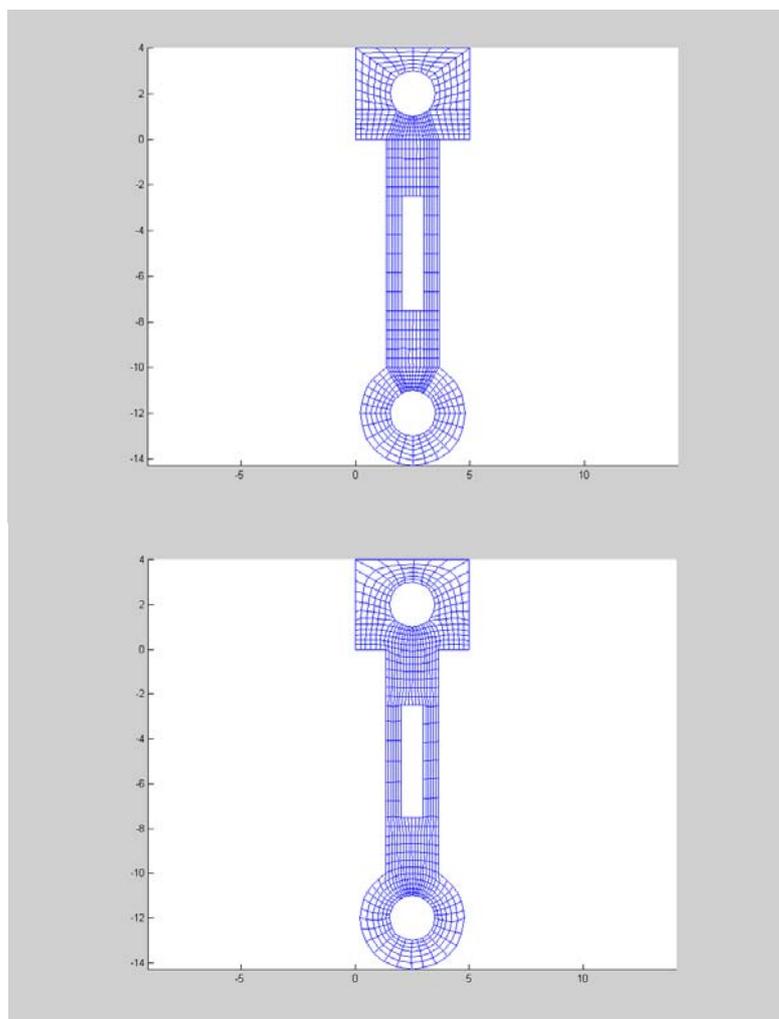


Figura 4: Discretização do primeiro caso apresentado. Tipo Dirichlet(acima) e Neumann (abaixo).

Tabela 1. Dados resultantes da discretização do exemplo 1

Elementos	Malha Inicial	Condições de Neumann			Condições de Dirichlet		
		Tempo em segundos	Tempo em segundos	Número de Iterações	Erro Máximo	Tempo em segundos	Número de Iterações
1472(8)	0.05	0.281	6	33.5	0.09	4	36.8
5888(16)	0.16	1.382	10	17.6	0.4	4	18.7
13248(24)	0.99	6.9	13	12.4	1.5	4	12.6
23552(32)	3.2	21.35	16	9.8	4.4	4	9.5

Tabela 2. Dados resultantes da discretização do exemplo 2

Elementos	Malha Inicial	Condições de Neumann			Condições de Dirichlet		
		Tempo em segundos	Tempo em segundos	Número de Iterações	Erro Máximo	Tempo em segundos	Número de Iterações
1984(8)	0.05	2.452	18	26.56	0.101	4	30.412
7936(16)	0.221	6.38	31	17.63	0.541	4	15.42
17856(24)	1.502	31.465	42	14.07	1.783	4	10.36
31744(32)	4.437	101.115	52	12.15	5.448	4	7.8

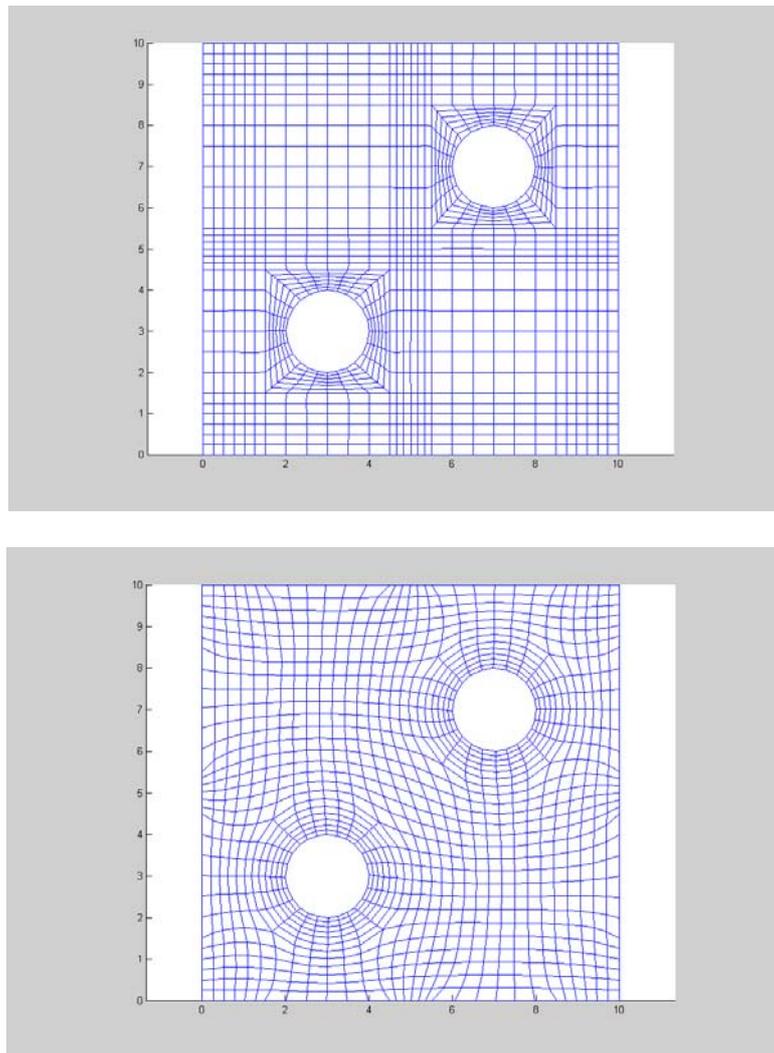


Figura 5: Discretização do segundo caso apresentado Tipo Dirichlet(acima) e Neumann (abaixo).

5.1 Conclusões e Perspectivas

Os resultados mostram que o método de suavização empregado é robusto quanto ao tratamento de geometrias complexas. A decomposição em multi blocos para a discretização de geometrias complexas provou ser bastante versátil, mas também pode aumentar o custo computacional. Porém, se em alguns casos a otimização de faces internas ao domínio eleva o tempo de cálculo, em outros mostrou-se extremamente interessante, principalmente quando aplicada a geometrias típicas de problemas de CFD, que exigem uma discretização cuidadosa ao redor dos corpos submersos no fluido. Para a aplicação deste método em problemas de geometria variável, a geração automática dos blocos deverá ser estudada nas próximas etapas.

6. REFERÊNCIAS BIBLIOGRÁFICAS

Maliska, C. R., 1995, Transferência de Calor e Mecânica dos Fluidos Computacional, LTC – Livros Técnicos e Científicos Editora S. A., Rio de Janeiro.

Roger, D. F., Adams J. A., 1990, Mathematical Elements for Computer Graphics, McGraw-Hill, Inc, New York.

GENERATION AND OPTIMIZATION OF STRUCTURED MESHES ON GENERIC TWO DIMENSIONAL DOMAINS

Márcio Ricardo Pivello

Mechanical Engineering College, Federal University of Uberlândia, Uberlândia – MG.

e-mail: pivello@mecanica.ufu.br

Carlos Roberto Ribeiro

Mechanical Engineering College, Federal University of Uberlândia, Uberlândia – MG.

e-mail: crribeiro@mecanica.ufu.br

***Abstract.** The aim of this work is to develop an algorithm for optimizing structured meshes generated over generic two dimensional domains, by splitting complex shapes into a collection of quadrilateral blocks, which are then meshed by a Laplacian elliptic mesh generator, and the mesh smoothness between adjacent areas is imposed later. A parametric formulation for cubic spline interpolation is used both to define the domain and apply the boundary conditions. The smoothing process relocates the boundary nodes on each block in order to obtain square elements. The process is solved as an optimization problem, where the design variables are the boundary nodes coordinates. These are relocated by a gradient-based search method and the elliptic generator uses them as Dirichlet boundary conditions to smooth the inner elements. The grid smoothness between adjacent blocks may be ensured by two methods, resulting in C^0 - or C^1 - smoothness.*

Keywords: structured meshes, optimization, general shape two dimensional domains