

Análise do Método dos Elementos Finitos para a Equação de Poisson

Pedro Menezes Pimenta

Engenharia de Computação



Estrutura da apresentação

- 1.Motivação e objetivo
- 2.Fundamentação
- 3.Metodologia
- 4.Resultados e considerações finais
- 5.Referências

O problema

A Equação de Poisson é uma EDP muito importante nos campos da física e da matemática. A equação é definida em um domínio Ω , dada por

$$-\Delta u = f$$

A solução é conhecida no contorno, essa é a condição de Dirichlet, que define as fronteiras do domínio com valor conhecido.

Nosso objetivo é resolver essa EDP usando o Método dos Elementos Finitos e desejamos disponibilizar uma análise de seu funcionamento para auxiliar estudos futuros.

Método dos Elementos Finitos

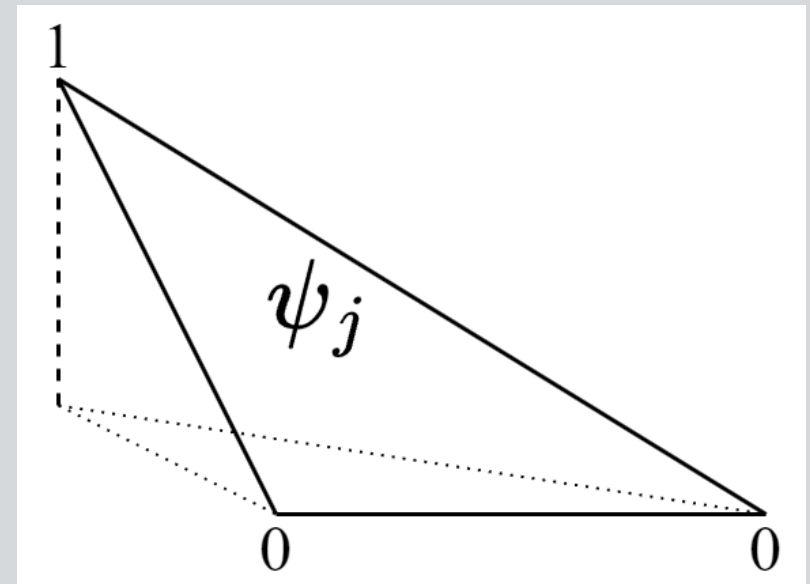
O Método dos Elementos serve principalmente para resolver EDPs de até três dimensões espaciais. Para aplicá-lo dividimos um domínio maior em vários subdomínios menores, chamados de elementos finitos. Para a definição desses subdomínios discretizamos o espaço em estudo e utilizamos esses pontos e as arestas entre eles como limites dos subdomínios. Ao final queremos escrever a solução como uma combinação linear de funções mais simples.

$$u_h(x) = \sum_{j=1}^N u_j \psi_j(x)$$

Funções de interpolação

Após definirmos o formato dos elementos da malha, devemos encontrar as funções de interpolação. Por definição essa função deve possuir valor unitário no nó em questão e valor nulo em qualquer outro nó, isso se repete para todos os outros nós da malha.

O método de Galerkin descreve a solução do problema como sendo a combinação linear das funções de interpolação.



Obtenção da matriz A

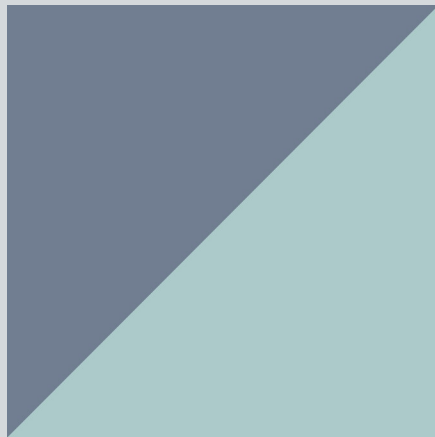
O triângulo é a forma mais simples para se utilizar, sua escolha também é estratégica, pois facilita a obtenção das funções de interpolação.

Para encontrar as funções de interpolação de um elemento escrevemos os três vértices, no sentido anti-horário, numa matriz 3x3.

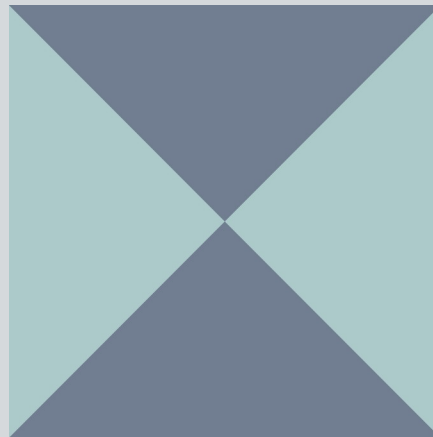
$$\psi_i(x, y) = a_i + b_i x + c_i y \quad \begin{vmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{vmatrix} \cdot \begin{vmatrix} a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \\ c_1 & c_2 & c_3 \end{vmatrix} = \begin{vmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{vmatrix}$$

Criação da malha

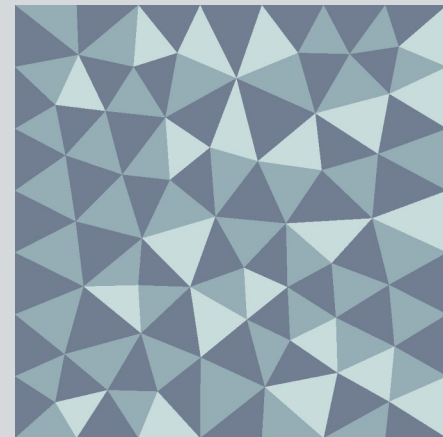
As malhas abaixo foram usadas como base para o trabalho. As três foram definidas utilizando o formato de arquivo reconhecido pela biblioteca MFEM. Partimos de malhas mais simples, pois foi possível obter mais elementos através do refinamento uniforme e mesmo assim mantendo a estrutura da malha.



triang-ret.mesh



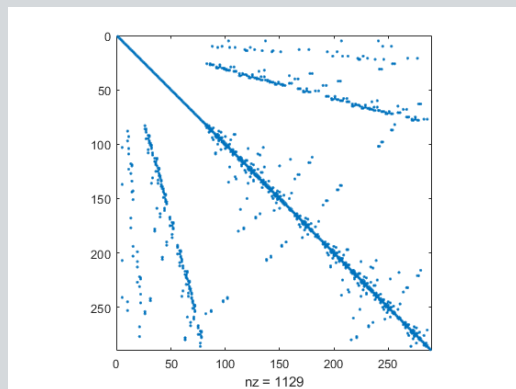
Estru.mesh



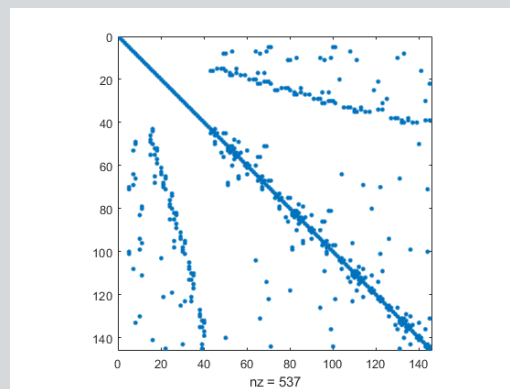
unEstru.mesh

Esparsidade da matriz A

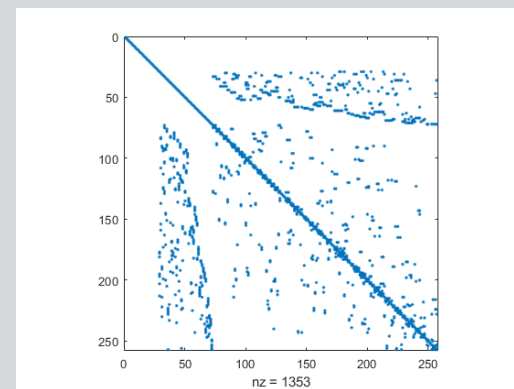
Abaixo estão as imagens da matriz A do sistema linear para cada malha que tenta aproximar a quantidade de elementos. Os pontos representam os coeficientes não nulos da matriz. Assim podemos observar como os vértices interagem entre si. A primeira contém 512 elementos, a segunda 256 e a última 456.



triang-ret.mesh



Estru.mesh



unEstru.mesh

Solução dos sistemas lineares

Os métodos numéricos para a resolução do sistema linear utilizados foram o Gradiente Conjugado Precondicionado e o Iterativo Estacionário Jacobi Precondicionado. Ambos os métodos requerem um valor de tolerância para a análise de convergência.

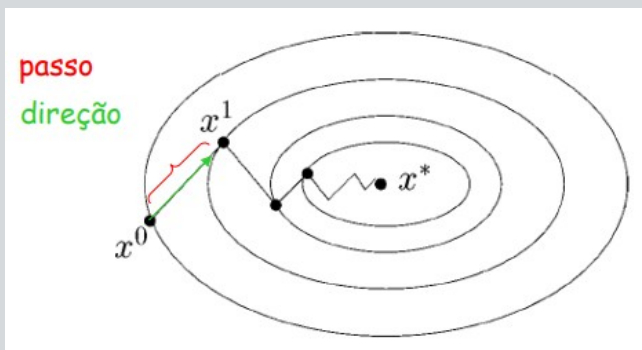


Imagem ilustrativa do funcionamento do Gradiente Conjugado

$$\begin{aligned}x_1^{(k+1)} &= \frac{1}{a_{11}}(b_1 - a_{12}x_2^{(k)} - a_{13}x_3^{(k)} - \dots - a_{1n}x_n^{(k)}) \\x_2^{(k+1)} &= \frac{1}{a_{22}}(b_2 - a_{21}x_1^{(k)} - a_{23}x_3^{(k)} - \dots - a_{2n}x_n^{(k)}) \\&\vdots \\x_n^{(k+1)} &= \frac{1}{a_{nn}}(b_n - a_{n1}x_1^{(k)} - a_{n2}x_2^{(k)} - \dots - a_{n,n-1}x_{n-1}^{(k)})\end{aligned}$$

Imagem das iterações do método iterativo estacionário

Problema escolhido

Para avaliar o funcionamento do método, realizamos a escolha de uma função da qual já conhecemos a solução para podermos calcular o erro real e assim realizar comparações. A EDP escolhida foi:

$$-\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right) = 2\pi^2 \text{sen}(\pi x) \text{sen}(\pi y)$$

E a função exata para realizarmos o cálculo do erro foi a solução da EDP, que é:

$$u(x, y) = \text{sen}(\pi x) \text{sen}(\pi y)$$

Então realizamos a resolução da primeira função e ao final comparamos o resultado obtido com o valor da segunda função.

Tabelas de erro

Abaixo a tabela de análise de convergência da solução obtida com a malha triang-ret.mesh.

Elementos	Erro	Razão	$\log_2(\text{Razão})$
8	0.233701	-	-
32	0.0530293	4.40701649842634	2.13980229760317
128	0.0129507	4.09470530550472	2.03375962534964
512	0.00321897	4.02324345986449	2.00835904288814
2048	0.000803564	4.0058663653424	2.00211429402292
8192	0.00020081	4.00161346546487	2.00058181732132
32768	5.02E-05	4.00009561505379	2.00003448542882

Tabelas de erro

Abaixo a tabela com os erros obtidos e as iterações necessárias para a malha triang-ret.mesh.

Elementos	Erro(PCG)	Iterações	Erro(SLI)	Iterações
8	0.233701	2	0.233701	2
32	0.0530293	6	0.0530289	18
128	0.0129507	11	0.0129497	68
512	0.00321897	19	0.00321785	265
2048	0.000803564	32	0.000801212	1057
8192	0.00020081	60	0.000199664	4225
32768	5.02E-05	114	4.90E-05	16900

Tabelas de erro

Abaixo a tabela com os erros obtidos e as iterações necessárias para a malha Estru.mesh.

Elementos	Erro(PCG)	Iterações	Erro(SLI)	Iterações
4	0.644934	2	0.644934	2
16	0.0483114	3	0.0483108	11
64	0.0368201	7	0.0368208	36
256	0.0151014	12	0.0151024	136
1024	0.0052067	21	0.00520788	534
4096	0.00165696	38	0.00165812	2119
16384	0.000502921	71	0.000504057	8458

Tabelas de erro

Abaixo a tabela com os erros obtidos e as iterações necessárias para a malha unEstru.mesh.

Elementos	Erro(PCG)	Iterações	Erro(SLI)	Iterações
114	0.0408172	10	0.0408161	56
456	0.0111453	19	0.0111443	211
1824	0.0030611	35	0.00306002	817
7296	0.000833829	70	0.000832715	3249
29184	0.000225499	140	0.000224379	12983

Conclusão e considerações finais

Com este trabalho analisamos o funcionamento do Método dos Elementos Finitos, explicando seu passo a passo, os resultados obtidos com dois métodos numéricos distintos e malhas com estruturas diferentes.

Para trabalhos futuros, devemos considerar a utilização de malhas adaptativas e talvez buscar comparar outros métodos numéricos.

Referências

- ANDERSON, R. et al. MFEM: A modular finite element methods library. Computers & Mathematics with Applications, v. 81, p. 42-74, 2021. Disponível em: <<https://doi.org/10.1016/j.camwa.2020.06.009>>.
- AZEVEDO, A. F. M. MÉTODO DOS ELEMENTOS FINITOS. [S.l.: s.n.], 2003. 1–153 p.
- BURDEN, R. L.; FAIRES, J. D. Numerical Analysis. [S.l.]: Cengage Learning, 2010. 450–495 p. ISBN 978-0-538-73351-9.
- CUNHA, M. C. C. Métodos numéricos. [S.l.: s.n.], 2000. ISBN 978-85-268-0877-5.
- FISCHER, P. et al. Scalability of high-performance pde solvers. The International Journal of High Performance Computing Applications, v. 34, n. 5, p. 562–586, 2020. Disponível em: <<https://doi.org/10.1177/1094342020915762>>.
- FRANGO, J. V. d. S. O MÉTODO DOS GRADIENTES CONJUGADOS PRÉ_x005F_x0002_CONDICIONADO POR FATORAÇÃO INCOMPLETA LU DE NÍVEL ZERO. 7 2018. Dis_x005F_x0002_ponível em: <
<https://app.uff.br/riuff/bitstream/handle/1/11619/113035017%20Joseana%20Veiga%20de%20Souza%20Frango.pdf?sequence=1&isAllowed=y>
>.
- JOHNSON, C. Numerical solutions of partial differential equations by the finite element method. [S.l.]: Cambridge University Press, 1988. 1–33 p. ISBN 978-05-21347-58-7.
- KREMER, I. MÉTODOS ITERATIVOS PARA SISTEMAS LINEARES. 2 2009. Disponível em: <
<https://repositorio.ufsc.br/xmlui/bitstream/handle/123456789/96623/lvandra.pdf?sequence=1&isAllowed=y>>.
- LEE, J. H.; GRIFFITH, B. E. On the lagrangian-eulerian coupling in the immersed finite element/difference method. Journal of Computational Physics, v. 457, 2022. ISSN 0021-9991. Disponível em: <
<https://www.sciencedirect.com/science/article/pii/S0021999122001048>>.
- LIU, W. K.; LI, S.; PARK, H. S. Eighty years of the finite element method: Birth, evolution, and future. Archives of Computational Methods in Engineering, 2022. Disponível em: <<https://doi.org/10.1007/s11831-022-09740-9>>.
- MOZOLEVSKI, I.; MURAD, M. A.; SCHUH, L. A. High order discontinuous galerkin method for reduced flow models in fractured porous media. Mathematics and Computers in Simulation, v. 190, p. 1317–1341, 2021. ISSN 0378-4754. Disponível em: <
<https://www.sciencedirect.com/science/article/pii/S0378475421002627>>.
- PEDROSO, N. M. C. M. O Método de Elementos Finitos: Aspectos da Implementação Computacional. 12 2019. Disponível em: <
https://app.uff.br/riuff/bitstream/handle/1/22769/TCC_NataliaPedroso.pdf?sequence=1&isAllowed=y>.
- QUARTERONI, A.; SALERI, F. CÁLCULO CIENTÍFICO com MATLAB e Octave. [S.l.: s.n.], 2007. 240–270 p. ISBN 978-88-47007-17-8.
- RINCON, M. A.; LIU, I.-S. Introdução ao Método de Elementos Finitos. [S.l.]: Instituto de Matemática/UFRJ, 2020. 15–102 p. ISBN 978-65-86502-00-8.
- SAAD, Y. Iterative Methods for Sparse Linear Systems. [S.l.: s.n.], 2003. 47–108 p. ISBN 978-05-34947-76-7.

Contato

E-mail: pedro.mp@grad.ufsc.br

Telefone: (62) 98316-3067

