UTILIZAÇÃO DO MÉTODO DE ELEMENTOS FINITOS PARA SIMULAÇÃO DE PROPAGAÇÃO DE FISSURAS

Palavras-Chave: MÉTODO DOS ELEMENTOS FINITOS, PROPAGAÇÃO DE FISSURAS, SIMULAÇÃO COMPUTACIONAL

Autores:

LARISSA MIEKO YOSHIZUMI HAGA, FECFAU-UNICAMP

Prof. Dr. NATHAN SHAUER (orientador), FECFAU-UNICAMP

INTRODUÇÃO:

O Método dos Elementos Finitos (FEM) é uma técnica numérica amplamente utilizada em engenharia e ciências aplicadas, que consiste na subdivisão de um domínio contínuo em pequenos elementos interligados, nos quais as equações diferenciais que regem o comportamento físico são aproximadas por funções conhecidas, geralmente polinomiais. Essa abordagem permite resolver sistemas complexos por meio do acoplamento dos elementos e aplicação de condições de contorno, possibilitando a análise precisa de fenômenos em áreas como mecânica dos sólidos, transferência de calor e eletromagnetismo, sendo essencial para o projeto, simulação e otimização de sistemas e estruturas em diversas disciplinas tecnológicas.

Entender como as fissuras se propagam nas estruturas é algo essencial na engenharia civil, já que essas falhas podem comprometer a segurança das construções ao longo do tempo, principalmente se não forem identificadas e corrigidas a tempo [1]. No entanto, esse tipo de análise não é simples, pois envolve muitos conceitos teóricos e cálculos bem complexos. Por isso, o uso de ferramentas computacionais se torna fundamental para facilitar esse tipo de estudo. Dentre as abordagens recentes, destaca-se a combinação do Método dos Elementos Finitos (FEM) com o método de campo de fase (phase field), que permite simular a nucleação e o avanço de trincas de forma contínua, sem a necessidade de definir previamente o trajeto da fissura. Em vez de representá-la como uma linha, o phase field utiliza uma variável que indica sua intensidade em cada ponto do domínio, possibilitando acompanhar seu surgimento, crescimento e eventual ramificação com maior precisão [2]. Baseada em princípios de energia do material, essa técnica oferece uma representação mais realista do comportamento estrutural [2], configurando-se como uma ferramenta promissora para a análise de diferentes cenários de carregamento e projeto de estruturas mais seguras e eficientes..

METODOLOGIA:

Inicialmente, foi feita uma revisão bibliográfica aprofundada abordando os fundamentos da mecânica dos sólidos, os mecanismos de propagação de fissuras em materiais estruturais e os princípios do Método dos Elementos Finitos (FEM). Essa etapa foi essencial para consolidar o embasamento teórico necessário para compreender o fenômeno físico em estudo e a correta aplicação das ferramentas computacionais. Também, foram estudadas as funcionalidades e os softwares Gmsh, Python, Mathematica, VSCode e Paraview, cada um com um uso específico para modelagem, simulação numérica e visualização dos resultados.

Com isso, foi desenvolvida uma malha geométrica no Gmsh. A qualidade e a densidade da malha foram criteriosamente avaliadas, dado o seu impacto direto na estabilidade numérica e na precisão dos resultados obtidos pelo FEM.

Posteriormente, utilizando o ambiente de programação do VSCode e a linguagem Python, foram realizadas modificações no código-fonte do script phasefield-jr-py [3], com o objetivo de importação e leitura direta de arquivos de malha no formato .msh, gerados pelo Gmsh. Essa adaptação permitiu maior flexibilidade na definição da geometria e das condições de contorno aplicadas ao modelo.

Finalizada a etapa de simulação, os dados computacionais gerados foram processados e visualizados no Paraview, possibilitando a análise qualitativa e quantitativa da evolução das fissuras, bem como a distribuição de tensões ao longo do domínio. Essa integração de ferramentas viabilizou uma abordagem robusta e eficiente para a simulação de fraturas em materiais frágeis, com potencial de aplicação em diversos contextos da engenharia estrutural.

RESULTADOS E DISCUSSÃO:

Para melhor entendimento do MEF foi proposto, primeiramente, o entendimento do método de Galerkin, já que o Método de Elementos Finitos é uma particularização dele. Para isso, foi proposto aplicar o Método de Galerkin para o seguinte problema, em que A é a área da seção transversal, l é o comprimento da barra, E é o módulo de elasticidade, e u(x) é o deslocamento a cada ponto da barra de treliça:

$$\begin{cases} EAu''(x) + f(x) = 0 \quad 0 < x < l \\ u(0) = 0 \\ EAu'(l) = 0 \end{cases}$$

Foi aplicado o Método de Galerkin utilizando as funções de forma $\{a_1x, a_2x^2\}$. E, com isso, o resultado aproximado foi comparado com a solução exata, o que gerou apenas uma curva, já que as funções de forma contemplam a exata ordem da função.

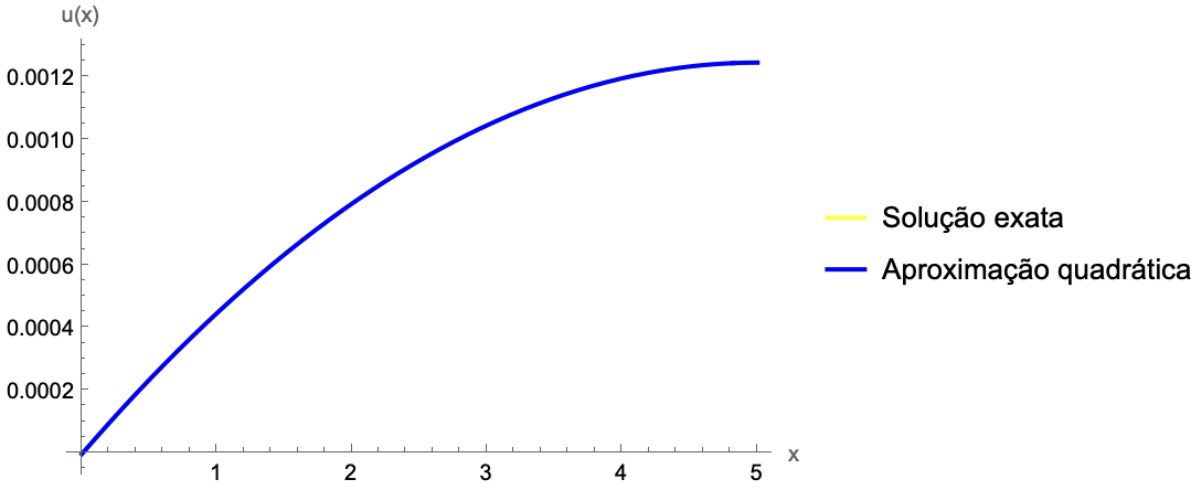

Figura 1 - Gráfico comparativo de soluções exata e aproximada
Fonte: Wolfram Mathematica

Também, utilizando o software CAST3M foi possível verificar a eficiência do método, comparando o resultado obtido por ele com o valor encontrado a partir da linha elástica

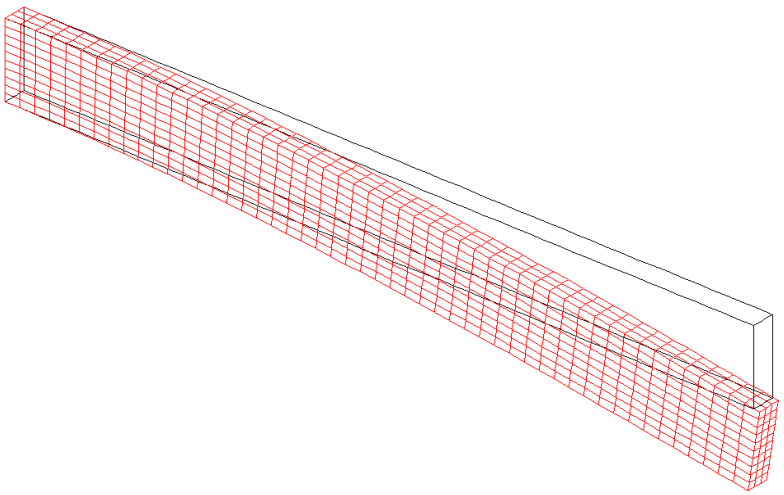

Figura 2: Projeção de deslocamento

Fonte: CAST3M

Cálculo por linha elástica: $f = \frac{PL^3}{3EI}$ P = -1000N L = 1 E = 30×10^9 $I = \frac{0.03\ x\ 0.1^3}{12}$

Por linha elástica tem-se f = $4{,}40 \times 10^{-3}$ m Valor encontrado pelo Cast3m $4{,}39 \times 10^{-3}$ m

Com base no Exemplo 1 apresentado por Shauer [5], que simula uma barra retangular sob tensão plana, com a extremidade esquerda engastada e deslocamento progressivo aplicado na extremidade oposta, resultando na fratura do material, foi construída, no Gmsh, uma malha com as mesmas dimensões do exemplo original (L = 200 mm e H = 20 mm), discretizada com 100 elementos

na direção horizontal e 10 na vertical. Foram utilizados elementos retangulares para representar a aplicação de tração oposta ao engaste, conforme a configuração proposta.

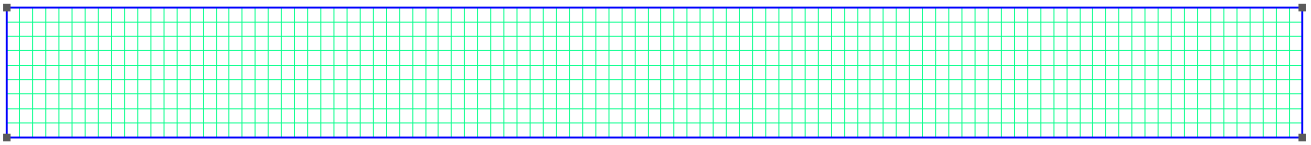

Figura 3: Malha gerada na interface do Gmsh

Fonte: Gmsh

Essa malha foi importada diretamente no código modificado, no qual a classe *CreateRectangularMesh* foi substituída por Mesh. Utilizou-se a biblioteca *meshio*, que permite extrair informações de arquivos no formato .msh, gerados pelo Gmsh. Com essa alteração, foi possível integrar de forma flexível geometrias definidas externamente, mantendo a precisão da simulação. Os resultados obtidos foram idênticos aos apresentados no exemplo original [5], validando a abordagem implementada.

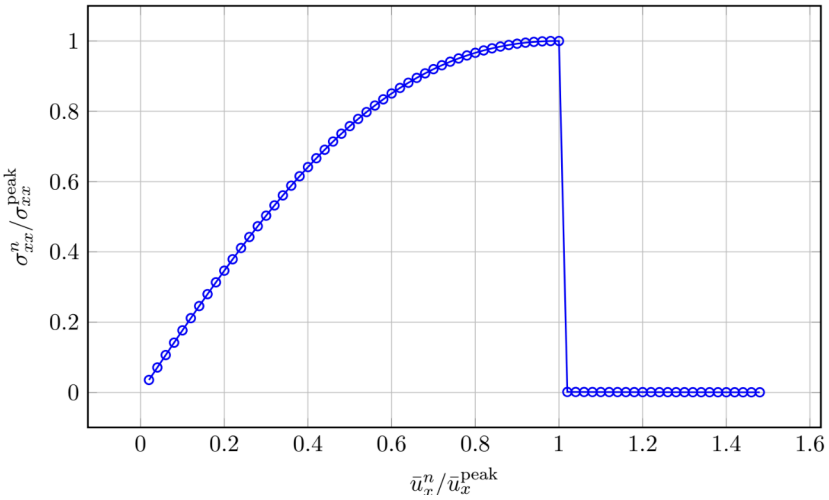

Figura 4: Gráfico apresentado como resultado do exemplo 1 em [5]

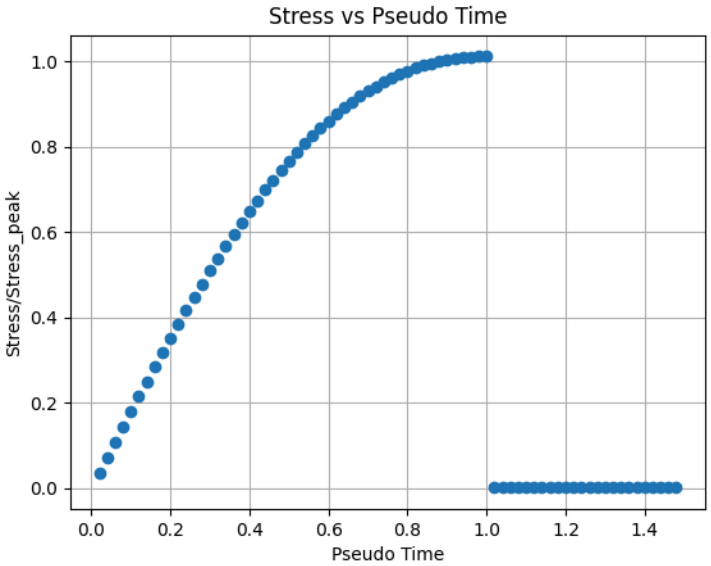

Figura 5: Gráfico gerado pelo código utilizando a malha importada diretamente

CONCLUSÕES:

A aplicação do Método dos Elementos Finitos aliado ao método de campo de fase demonstrou-se eficaz para a simulação da propagação de fissuras em materiais frágeis. A utilização de ferramentas como o Gmsh, Python e Paraview permitiu a construção de uma malha personalizada, a modificação do código-fonte original e a visualização detalhada dos resultados, proporcionando maior flexibilidade e controle sobre o modelo numérico. A comparação dos resultados obtidos com soluções analíticas e exemplos da literatura validou a precisão do método e a confiabilidade da implementação computacional adotada.

Além disso, a adaptação do código para importar diretamente malhas externas contribuiu para tornar o processo mais eficiente, eliminando a necessidade de definir manualmente os parâmetros geométricos no código-fonte. Essa abordagem facilita a análise de diferentes configurações estruturais e condições de contorno, tornando-se uma ferramenta promissora para estudos avançados de fratura em engenharia civil. O trabalho desenvolvido reforça o potencial da integração entre modelagem numérica e simulação computacional como suporte para análises estruturais mais precisas, seguras e automatizadas.

BIBLIOGRAFIA

[1] FONSECA, Eliana. Fissuras, trincas, causas, prevenções e terapia em edificações. Universidade Presbiteriana Mackenzie, São Paulo, p. 18-26, 2009. Disponível em: https://dspace.mackenzie.br/bitstream/handle/10899/184/Eliana%20Amorim%20Coutinho%20Fonseca1.pdf?sequence=1&isAllowed=y

[2] KRISTENSEN, P. K.; NIORDSON, C. F.; MARTÍNEZ & PAÑEDA, E. (2021). **An assessment of phase field fracture: crack initiation and growth.** Philosophical transactions. Series A, Mathematical, physical, and engineering sciences, 379(2203), 20210021. https://doi.org/10.1098/rsta.2021.0021

[3] SHAUER, Nathan. phasefield-jr-py: **Implementação em Python do método de campo de fase para fratura.** Jun. 2025. Disponível em: https://github.com/nathanshauer/phasefield-jr-py.

[4]Wolfram Mathematica: Modern technical computing. Disponível em: <https://www.wolfram.com/mathematica/>

[5] Cast3M. Disponível em: <https://www-cast3m.cea.fr/index.php>

[6] SHAUER, Nathan (2025). **Less than 500 Lines Self-Contained Python Finite Element Implementation of the Phase-Field Method for Fracture Mechanics.** doi:10.2139/ssrn.5293414

[7] Gmsh. Disponível em: <https://gmsh.info/>