Resolução do Problema Unidimensional Transiente Não-Linear com Condição de Fronteira de Dirichlet Homogênea via o Método dos Elementos Finitos

João Victor Lopez Pereira

10 de março de 2025

Rio de Janeiro - RJ

Resumo:

Este documento apresenta a resolução de um sistema genérico de equações diferenciais ordinárias, com uma componente não linear, utilizando o Método dos Elementos Finitos, conforme abordado nas aulas da disciplina *Introdução ao Método dos Elementos Finitos*, ministradas pelo Marcello Goulart Teixeira e Bruno Alves do Carmo na Universidade Federal do Rio de Janeiro, durante o segundo semestre de 2024. Neste documento, apresentamos a resolução de uma equação do calor em uma dimensão espacial, com dependência temporal e não linearidade.

Abstract:

This document presents the solution of a generic system of ordinary differential equations, with a non-linear component, using the Finite Elements Method, as covered in the course *Introduction to the Finite Element Method*, taught by professors Marcello Goulart Teixeira and Bruno Alves do Carmo at the Federal University of Rio de Janeiro, during the second half of 2024. In this document, we present the solution of a one-dimensional heat equation, with temporal dependence and non-linearity.

Agradecimentos:

Agradeço ao professores Marcello Teixeira e Bruno Carmo e aos colegas Leonardo, Hashimoto e a vários outros que me ajudaram no entendimento do conteúdo necessário para a realização das contas e do Método dos Elementos Finitos como um todo.

Thanks:

I would like to thank professors Marcello Teixeira and Bruno Carmo and colleagues Leonardo, Hashimoto and several other who helped me understand the necessary content for carrying out the calculations and the Finite Element Method as a whole.

Sumário

Capítulo 1

Aspectos Teóricos

1.1 Definição da Formulação Forte

Dada uma função $f : [0,1] \times [0,T] \to \mathbb{R}$ e constantes $\alpha > 0$, $\beta \geq 0$, $T > 0$ e uma função $g(x)$, queremos encontrar $u : [0,1] \times [0,T] \to \mathbb{R}$ tal que:

$$(S) = \begin{cases} u_t(x,t) - \alpha u_{xx}(x,t) + \beta u(x,t) + g(u) = f(x,t) \ \forall x \in (0,1) \\ \\ u(0,t) = u(1,t) = 0 \\ \\ u(x,0) = u_0(x). \end{cases}$$

1.2 Transição entre a Formulação Forte e Fraca

Visto que $u_t(x,t) - \alpha u_{xx}(x,t) + \beta u(x,t) + g(u) = f(x,t)$, podemos multiplicar ambos por lados por uma função $v(x)$ tal que $v(1) = v(0) = 0$ que nos ajude a eliminar a segunda derivada u_{xx}:

$$\begin{aligned}
u_t(x,t) - \alpha u_{xx}(x,t) + \beta u(x,t) + g(u) &= f(x,t) \\
[u_t(x,t) - \alpha u_{xx}(x,t) + \beta u(x,t) + g(u)]v(x) &= f(x,t)v(x) \\
u_t(x,t)v(x) - \alpha u_{xx}(x,t)v(x) + \beta u(x,t)v(x) + g(u)v(x) &= f(x,t)v(x) \\
\int_0^1 u_t(x,t)v(x)\,dx - \int_0^1 \alpha u_{xx}(x,t)v(x)\,dx + \int_0^1 \beta u(x,t)v(x)\,dx + \int_0^1 g(u)v(x)\,dx &= \int_0^1 f(x,t)v(x)\,dx \\
\int_0^1 u_t(x,t)v(x)\,dx - \alpha \int_0^1 u_{xx}(x,t)v(x)\,dx + \beta \int_0^1 u(x,t)v(x)\,dx + \int_0^1 g(u)v(x)\,dx &= \int_0^1 f(x,t)v(x)\,dx
\end{aligned}$$

Sabemos que dadas funções f e g:

$$\int f(x)g'(x)dx = f(x)g(x) - \int f'(x)g(x)dx$$

Logo, realizando a integração por partes para eliminar a segunda derivada u_{xx} da equação:

$$\int_0^1 u_t(x,t)v(x)\,dx - \alpha\int_0^1 u_{xx}(x,t)v(x)\,dx + \beta\int_0^1 u(x,t)v(x)\,dx + \int_0^1 g(u)v(x)\,dx = \int_0^1 f(x,t)v(x)\,dx,$$

$$\int_0^1 u_t(x,t)v(x)\,dx - \alpha\left[u_x(x,t)v(x)\Big|_0^1 - \int_0^1 u_x(x,t)v_x(x)\,dx\right] + \beta\int_0^1 u(x,t)v(x)\,dx$$

$$+\int_0^1 g(u)v(x)\,dx = \int_0^1 f(x,t)v(x)\,dx$$

$$\int_0^1 u_t(x,t)v(x)\,dx - \alpha\left[u_x(1,t)v(1) - u_x(0,t)v(0) - \int_0^1 u_x(x,t)v_x(x)\,dx\right] + \beta\int_0^1 u(x,t)v(x)\,dx$$

$$+\int_0^1 g(u)v(x)\,dx = \int_0^1 f(x,t)v(x)\,dx,$$

$$\int_0^1 u_t(x,t)v(x)\,dx - \alpha\left[\cancel{u_x(1,t)v(1) - u_x(0,t)v(0)} - \int_0^1 u_x(x,t)v_x(x)\,dx\right] + \beta\int_0^1 u(x,t)v(x)\,dx$$

$$+\int_0^1 g(u)v(x)\,dx = \int_0^1 f(x,t)v(x)\,dx,$$

$$\int_0^1 u_t(x,t)v(x)\,dx - \alpha\left[-\int_0^1 u_x(x,t)v_x(x)\,dx\right] + \beta\int_0^1 u(x,t)v(x)\,dx + \int_0^1 g(u)v(x)\,dx = \int_0^1 f(x,t)v(x)\,dx,$$

$$\int_0^1 u_t(x,t)v(x)\,dx + \alpha\int_0^1 u_x(x,t)v_x(x)\,dx + \beta\int_0^1 u(x,t)v(x)\,dx + \int_0^1 g(u)v(x)\,dx = \int_0^1 f(x,t)v(x)\,dx.$$

Definição de Notação

Definição: $(f,g) = \displaystyle\int_0^1 f(x)g(x)\,dx$

Definição: $\kappa(f,g) = \alpha\displaystyle\int_0^1 f_x(x)g_x(x)\,dx + \beta\int_0^1 f(x)g(x)\,dx$

Ou seja, o problema:

$$\int_0^1 u_t(x,t)v(x)\,dx + \alpha\int_0^1 u_x(x,t)v_x(x)\,dx + \beta\int_0^1 u(x,t)v(x)\,dx + \int_0^1 g(u)v(x)\,dx = \int_0^1 f(x,t)v(x)\,dx$$

Pode ser escrito como:

$$(u_t(t), v) + \kappa(u(t), v) + (g(u(t)), v) = (f(t), v)$$

1.3 Definição da Formulação Fraca

Definição: Pelo restante do documento, considere que uma função g é "suficientemente suave" se ela respeitar:

- g é contínua em todo o domínio;
- g possui derivadas contínuas até a ordem necessária.

Seja H um espaço de funções formado por funções u suficientemente suaves que satisfazem (W) e as condições de contorno $u(0,t) = u(1,t) = 0$. Seja V um espaço das funções de teste, composto por funções v suficientemente suaves e que satisfazem as condições de contorno $v(0) = v(1) = 0$.

Dada uma função $f : [0,1] \times [0,T] \to \mathbb{R}$ e constantes $\alpha > 0$, $\beta \geq 0$, $T > 0$ e uma função $g(x)$, queremos encontrar $u : [0,1] \times [0,T] \to \mathbb{R}$, $u \in H$ tal que, $\forall v \in V$:

$$(W) = \begin{cases} (u_t(t), v) + \kappa(u(t), v) + (g(u(t)), v) = (f(t), v) \\ \\ u(0,t) = u(1,t) = 0 \\ \\ u(x,0) = u_0(x). \end{cases}$$

1.4 Problema Aproximado Totalmente Discreto via o método de Crank-Nicolson Galerkin linearizado

1.4.1 Problema Variacional no Ponto Médio

Discretizaremos o intervalo $[0,T]$ em $t_0, t_1, \ldots, t_N$ tal que $t_n - t_{n-1} = \tau$, $\forall n \in [0,N]$.

Temos então:

$$(u_t(t_{n-\frac{1}{2}}), v) + \kappa(u(t_{n-\frac{1}{2}}), v) + (g(u(t_{n-\frac{1}{2}})), v) = (f(t_{n-\frac{1}{2}}), v)$$

tal que:

$$t_{n-\frac{1}{2}} = \frac{t_n + t_{n-1}}{2}$$

é o ponto médio do intervalo $[t_{n-1}, t_n]$.

1.4.2 Diferenças Finitas no Tempo

Sabemos que as seguintes aproximações são válidas:

$$u_t(t_{n-\frac{1}{2}}) = \frac{u(t_n) - u(t_{n-1})}{\tau} + O(\tau^2)$$

$$u(t_{n-\frac{1}{2}}) = \frac{u(t_n) + u(t_{n-1})}{2} + O(\tau^2)$$

$$u(t_{n-\frac{1}{2}}) = \frac{3u(t_{n-1}) - u(t_{n-2})}{2} + O(\tau^2).$$

Substituindo-as em nossa equação:

$$\left(\frac{u(t_n) - u(t_{n-1})}{\tau}, v\right) + \kappa \left(\frac{u(t_n) + u(t_{n-1})}{2}, v\right) + \left(g\left(\frac{3u(t_{n-1}) - u(t_{n-2})}{2}\right), v\right) \approx (f(t_{n-\frac{1}{2}}), v).$$

Visto que nosso sistema agora não é exato — por conta das parcelas $O(\tau^2)$ que foram desprezadas — queremos determinar $U^n \in H$ tal que $U^n \approx u(t_n)$ e seja solução da equação:

$$\left(\frac{U^n - U^{n-1}}{\tau}, v\right) + \kappa \left(\frac{U^n + U^{n-1}}{2}, v\right) + \left(g\left(\frac{3U(t_{n-1}) - U(t_{n-2})}{2}\right), v\right) = (f(t_{n-\frac{1}{2}}), v)$$

1.5 Definição do Problema Aproximado Totalmente Discreto

Seja H^m um espaço de funções finito-dimensional composto por funções U^h suficientemente suaves que satisfazem (A) e as condições de contorno $U^h(0,t) = U^h(1,t) = 0$. Analogamente, seja V^m um espaço de funções finito-dimensional das funções de teste, formado por funções v^h suficientemente suaves que também atendem às condições de contorno $v^h(0) = v^h(1) = 0$.

$$\left(\frac{U_h^n - U_h^{n-1}}{\tau}, v_h\right) + \kappa \left(\frac{U_h^n + U_h^{n-1}}{2}, v_h\right) + \left(g\left(\frac{3U_h^{n-1} - U_h^{n-2}}{2}\right), v_h\right) = (f(t_{n-\frac{1}{2}}), v_h)$$

Dada uma função $f : [0,1] \times [0,T] \to \mathbb{R}$ e constantes $\alpha > 0$, $\beta \geq 0$, $T > 0$ e uma função $g(x)$, queremos determinar $U_h^n \in H^m$ tal que, $\forall v_h \in V^m$:

$$(A) = \begin{cases} (u_t(t), v) + \kappa(u(t), v) + (g(u(t)), v) = (f(t), v) \\ \\ u(0, t) = u(1, t) = 0 \\ \\ u(x, 0) = u_0(x). \end{cases}$$

1.6 Transição entre o Problema Aproximado e a Forma Matriz-vetor

Tomando U_h^n como combinação linear das funções da base de H^m, temos:

$$U_h^n(x) = \sum_{j=1}^{m} c_j^n \varphi_j(x).$$

Além disso, temos que (A) é válido $\forall v_h \in V^m$, logo:

$$\left(\frac{\left[\sum_{j=1}^m c_j^n - \sum_{j=1}^m c_j^{n-1}\right] \varphi_j}{\tau}, \varphi_i \right) + k \left(\frac{\left[\sum_{j=1}^m c_j^n + \sum_{j=1}^m c_j^{n-1}\right] \varphi_j}{2}, \varphi_i \right)$$
$$+ \left(g \left(\frac{\left[3 \sum_{j=1}^m c_j^{n-1} - \sum_{j=1}^m c_j^{n-2}\right] \varphi_j}{2} \right), \varphi_i \right) = (f(t_{n-\frac{1}{2}}), \varphi_i)$$

$$\sum_{j=1}^{m} \left(\frac{[c_j^n - c_j^{n-1}]\varphi_j}{\tau}, \varphi_i \right) + \kappa \sum_{j=1}^{m} \left(\frac{[c_j^n + c_j^{n-1}]\varphi_j}{2}, \varphi_i \right)$$
$$+ \left(g \left(\frac{\sum_{j=1}^m [3c_j^{n-1} - c_j^{n-2}]\varphi_j}{2} \right), \varphi_i \right) = (f(t_{n-\frac{1}{2}}), \varphi_i)$$

$$\sum_{j=1}^{m} \left[\frac{c_j^n - c_j^{n-1}}{\tau} (\varphi_j, \varphi_i) \right] + \sum_{j=1}^{m} \left[\frac{c_j^n + c_j^{n-1}}{2} \kappa(\varphi_j, \varphi_i) \right]$$
$$+ \left(g \left(\frac{\sum_{j=1}^m [3c_j^{n-1} - c_j^{n-2}]\varphi_j}{2} \right), \varphi_i \right) = (f(t_{n-\frac{1}{2}}), \varphi_i)$$

Mas veja que (A) é válido $\forall i \in \{1, \ldots, m\}$, logo,:

$$\begin{cases} \cdots + (\varphi_j, \varphi_1)\dfrac{c_j^n - c_j^{n-1}}{\tau} + \kappa(\varphi_j, \varphi_1)\dfrac{c_j^n + c_j^{n-1}}{2} + \left(g\left(\dfrac{\sum_{j=1}^m [3c_j^{n-1} - c_j^{n-2}]\varphi_j}{2}\right), \varphi_i\right) + \cdots = (f(t_{n-\frac{1}{2}}), \varphi_1) \\ \vdots \\ \cdots + (\varphi_j, \varphi_i)\dfrac{c_j^n - c_j^{n-1}}{\tau} + \kappa(\varphi_j, \varphi_i)\dfrac{c_j^n + c_j^{n-1}}{2} + \left(g\left(\dfrac{\sum_{j=1}^m [3c_j^{n-1} - c_j^{n-2}]\varphi_j}{2}\right), \varphi_i\right) + \cdots = (f(t_{n-\frac{1}{2}}), \varphi_i) \\ \vdots \\ \cdots + (\varphi_j, \varphi_m)\dfrac{c_j^n - c_j^{n-1}}{\tau} + \kappa(\varphi_j, \varphi_m)\dfrac{c_j^n + c_j^{n-1}}{2} + \left(g\left(\dfrac{\sum_{j=1}^m [3c_j^{n-1} - c_j^{n-2}]\varphi_j}{2}\right), \varphi_m\right) + \cdots = (f(t_{n-\frac{1}{2}}), \varphi_m) \end{cases}$$

Perceba que podemos organizar essas equações em produtos matriz-vetor tal que:

$$\begin{bmatrix} (\varphi_1, \varphi_1) & \dots & (\varphi_m, \varphi_1) \\ \vdots & \ddots & \vdots \\ (\varphi_1, \varphi_m) & \dots & (\varphi_m, \varphi_m) \end{bmatrix} \frac{c^n - c^{n-1}}{\tau} + \begin{bmatrix} \kappa(\varphi_1, \varphi_1) & \dots & \kappa(\varphi_m, \varphi_1) \\ \vdots & \ddots & \vdots \\ \kappa(\varphi_1, \varphi_m) & \dots & \kappa(\varphi_m, \varphi_m) \end{bmatrix} \frac{c^n + c^{n-1}}{2}$$
$$+ \begin{bmatrix} \left(g\left(\dfrac{\sum_{j=1}^m [3c_j^{n-1} - c_j^{n-2}]\varphi_j}{2}\right), \varphi_1\right) \\ \vdots \\ \left(g\left(\dfrac{\sum_{j=1}^m [3c_j^{n-1} - c_j^{n-2}]\varphi_j}{2}\right), \varphi_m\right) \end{bmatrix} = \begin{bmatrix} (f(t_{n-\frac{1}{2}}), \varphi_1) \\ \vdots \\ (f(t_{n-\frac{1}{2}}), \varphi_m) \end{bmatrix}$$

Seja $\mathcal{M} = \begin{bmatrix} (\varphi_1, \varphi_1) & \dots & (\varphi_m, \varphi_1) \\ \vdots & \ddots & \vdots \\ (\varphi_1, \varphi_m) & \dots & (\varphi_m, \varphi_m) \end{bmatrix}$.

Seja $\mathcal{K} = \begin{bmatrix} \kappa(\varphi_1, \varphi_1) & \dots & \kappa(\varphi_m, \varphi_1) \\ \vdots & \ddots & \vdots \\ \kappa(\varphi_1, \varphi_m) & \dots & \kappa(\varphi_m, \varphi_m) \end{bmatrix}$.

Seja $\mathcal{G}(c) = \begin{bmatrix} (g(\sum_{j=1}^m c_j\varphi_j), \varphi_1) \\ \vdots \\ (g(\sum_{j=1}^m c_j\varphi_j), \varphi_m) \end{bmatrix}$.

Seja $\mathcal{F}^{n-\frac{1}{2}} = \begin{bmatrix} (f(t_{n-\frac{1}{2}}), \varphi_1) \\ \vdots \\ (f(t_{n-\frac{1}{2}}), \varphi_m) \end{bmatrix}$.

1.7 Definição do Problema na Forma Matriz-vetor

Dadas matrizes duas $\mathcal{M}$ e $\mathcal{K}$, um vetor $\mathcal{F}^{n-\frac{1}{2}}$ e $\mathcal{G}(\rfloor)$, queremos encontrar C^n tal que:

$$(M) = \left\{ \mathcal{M}\frac{C^n - C^{n-1}}{\tau} + \mathcal{K}\frac{C^n + C^{n-1}}{2} + \mathcal{G}\left(\frac{3C^{n-1} - C^{n-2}}{2}\right) = \mathcal{F}^{n-\frac{1}{2}} \right.$$

Em breve falaremos melhor sobre os casos em que $n = 0$ e $n = 1$.

1.8 Solução da Forma Matriz-vetor

$$\mathcal{M}\frac{C^n - C^{n-1}}{\tau} + \mathcal{K}\frac{C^n + C^{n-1}}{2} + \mathcal{G}\left(\frac{3C^{n-1} - C^{n-2}}{2}\right) = \mathcal{F}^{n-\frac{1}{2}}$$

$$\mathcal{M}2\tau\frac{C^n - C^{n-1}}{\tau} + \mathcal{K}2\tau\frac{C^n + C^{n-1}}{2} + 2\tau\mathcal{G}\left(\frac{3C^{n-1} - C^{n-2}}{2}\right) = 2\tau\mathcal{F}^{n-\frac{1}{2}}$$

$$\mathcal{M}2\cancel{\tau}\frac{C^n - C^{n-1}}{\cancel{\tau}} + \mathcal{K}\cancel{2}\tau\frac{C^n + C^{n-1}}{\cancel{2}} + 2\tau\mathcal{G}\left(\frac{3C^{n-1} - C^{n-2}}{2}\right) = 2\tau\mathcal{F}^{n-\frac{1}{2}}$$

$$\mathcal{M}2[C^n - C^{n-1}] + \mathcal{K}\tau[C^n + C^{n-1}] + 2\tau\mathcal{G}\left(\frac{3C^{n-1} - C^{n-2}}{2}\right) = 2\tau\mathcal{F}^{n-\frac{1}{2}}$$

$$2\mathcal{M}C^n - 2\mathcal{M}C^{n-1} + \tau\mathcal{K}C^n + \tau\mathcal{K}C^{n-1} + 2\tau\mathcal{G}\left(\frac{3C^{n-1} - C^{n-2}}{2}\right) = 2\tau\mathcal{F}^{n-\frac{1}{2}}$$

Colocando as equações em um formato solucionável utilizando sistemas lineares:

$$2\mathcal{M}C^n + \tau\mathcal{K}C^n = 2\tau\mathcal{F}^{n-\frac{1}{2}} + 2\mathcal{M}C^{n-1} - \tau\mathcal{K}C^{n-1} - 2\tau\mathcal{G}\left(\frac{3C^{n-1} - C^{n-2}}{2}\right)$$

$$[2\mathcal{M} + \tau\mathcal{K}]\, C^n = 2\tau\mathcal{F}^{n-\frac{1}{2}} + [2\mathcal{M} - \tau\mathcal{K}]\, C^{n-1} - 2\tau\mathcal{G}\left(\frac{3C^{n-1} - C^{n-2}}{2}\right)$$

$$\left[\mathcal{M} + \left(\frac{\tau}{2}\right)\mathcal{K}\right] C^n = \tau\mathcal{F}^{n-\frac{1}{2}} + \left[\mathcal{M} - \left(\frac{\tau}{2}\right)\mathcal{K}\right] C^{n-1} - \tau\mathcal{G}\left(\frac{3C^{n-1} - C^{n-2}}{2}\right)$$

$$\left[\mathcal{M} + \left(\frac{\tau}{2}\right)\mathcal{K}\right] C^n = \tau\mathcal{F}^{n-\frac{1}{2}} + \left[\mathcal{M} - \left(\frac{\tau}{2}\right)\mathcal{K}\right] C^{n-1} - \tau\mathcal{G}\left(\frac{3C^{n-1} - C^{n-2}}{2}\right).$$

Seja $\mathcal{A} = \left[\mathcal{M} + \left(\frac{\tau}{2}\right)\mathcal{K}\right]$.

Seja $\mathcal{B} = \left[\mathcal{M} - \left(\frac{\tau}{2}\right)\mathcal{K}\right]$.

Assim, temos:

$$\mathcal{A}C^n = \tau\mathcal{F}^{n-\frac{1}{2}} + \mathcal{B}C^{n-1} - \tau\mathcal{G}\left(\frac{3C^{n-1} - C^{n-2}}{2}\right)$$

$$C^n = \mathcal{A} \setminus \tau\mathcal{F}^{n-\frac{1}{2}} + \mathcal{B}C^{n-1} - \tau\mathcal{G}\left(\frac{3C^{n-1} - C^{n-2}}{2}\right).$$

Mas veja que ao utilizarmos $u(t_{n-\frac{1}{2}}) = \dfrac{3u(t_{n-1}) - u(t_{n-2})}{2} + O(\tau^2)$ — para calcularmos C^n — estamos utilizando os valores de C^{n-1} e C^{n-2}. Mas veja que isso nos restringe a $n \geq 2$, visto que para calcular os valores de C^0 e C^1 teríamos um problema. Sendo assim, faremos:

para $n = 0$: Temos diversas opções de inicialização. Veremos quatro delas em breve no documento. Por enquanto, assuma que já tenhamos tenhamos C^0.

para $n = 1$:

$$\hat{C} = \mathcal{A} \setminus \tau\mathcal{F}^{n-\frac{1}{2}} + \mathcal{B}C^{n-1} - \tau\mathcal{G}(C^{n-1})$$

Tal que $\hat{C}$ seja apenas um valor intermediário auxiliar que não utilizaremos de fato em nossa aproximação. Perceba que $\hat{C}$ apresenta ordem de erro de $O(\tau)$. por isso, no caso $n = 1$, utilizaremos $\hat{C}$ e C^{n-1} para estimar C^n visto que não temos C^{n-2} (pois não existe):

$$C^n = \mathcal{A} \setminus \tau\mathcal{F}^{n-\frac{1}{2}} + \mathcal{B}C^{n-1} - \tau\mathcal{G}\left(\frac{\hat{C} + C^{n-1}}{2}\right)$$

Dessa forma, C^n apresenta erro na ordem de $O(\tau^2)$. Ou seja, nosso método continua tendo erro na ordem de $O(\tau^2)$.

para $n \geq 2$:

$$C^n = \mathcal{A} \setminus \tau\mathcal{F}^{n-\frac{1}{2}} + \mathcal{B}C^{n-1} - \tau\mathcal{G}\left(\frac{3C^{n-1} - C^{n-2}}{2}\right)$$

Que é como calculamos anteriormente. Visto que em $n \geq 2$, já se tem os valores de ao menos 2 pontos anteriores.

1.9 Solução do Sistema no Momento Inicial

Primeira Opção

Visto que U^0 é dado de entrada e sabemos que:

$$U^0(x) = \sum_{j=1}^{m} C_j^0 \varphi_j(x)$$

Visto que nossa escolha de $\varphi_j(x)$ vale 1 nos pontos da discretização, logo:

$$C_i^0 = U^0(x)$$

$$C^0 = \begin{bmatrix} u_0(x_1) \\ \vdots \\ u_0(x_m) \end{bmatrix}$$

1.9.1 Segunda Opção

Seja $U^0 \in V_m$ tal que:

$$(U^0 - u_0, v_h) = 0 \quad \forall v_h \in V_m$$

Tomando $U^0(x) = \sum_{j=1}^{m} C_j^0 \varphi_j(x)$ e $v_h = \varphi_i$ para $i \in \{1, \dots, m\}$, temos:

$$(\sum_{j=1}^{m} C_j^0 \varphi_j(x) - u_0, \varphi_i) = 0$$

$$\sum_{j=1}^{m} C_j^0 (\varphi_j(x) - u_0, \varphi_i) = 0$$

$$\sum_{j=1}^{m} C_j^0 (\varphi_j(x), \varphi_i) - (u_0, \varphi_i) = 0$$

$$\sum_{j=1}^{m} C_j^0 (\varphi_j(x), \varphi_i) = (u_0, \varphi_i)$$

Que pode ser escrito na forma matriz-vetor tal que, usando a notação definida anteriormente, temos:

$$MC^0 = \begin{bmatrix} (u_0, \varphi_1) \\ \vdots \\ (u_0, \varphi_m) \end{bmatrix}$$

1.9.2 Terceira Opção

Seja $U^0 \in V_m$ tal que:

$$((U^0 - u_0)_x, v_{hx}) = 0 \quad \forall v_h \in V_m$$

Tomando $U^0(x) = \sum_{j=1}^{m} C_j^0 \varphi_j(x)$ e $v_h = \varphi_i$ para $i \in \{1, \ldots, m\}$, temos:

$$((\sum_{j=1}^{m} C_j^0 \varphi_j - u_0)_x, \varphi_{ix}) = 0$$

$$\sum_{j=1}^{m} C_j^0((\varphi_j - u_0)_x, \varphi_{ix}) = 0$$

$$\sum_{j=1}^{m} C_j^0(\varphi_{jx} - u_{0x}, \varphi_{ix}) = 0$$

$$\sum_{j=1}^{m} C_j^0(\varphi_{jx}, \varphi_{ix})(-u_{0x}, \varphi_{ix}) = 0$$

$$\sum_{j=1}^{m} C_j^0(\varphi_{jx}, \varphi_{ix}) = (u_{0x}, \varphi_{ix})$$

Que pode ser escrito na forma matriz-vetor tal que, usando a notação definida anteriormente, temos:

$$\begin{bmatrix} (\varphi_{1x}, \varphi_{1x}) & \cdots & (\varphi_{mx}, \varphi_{1x}) \\ \vdots & \ddots & \vdots \\ (\varphi_{1x}, \varphi_{mx}) & \cdots & (\varphi_{mx}, \varphi_{mx}) \end{bmatrix} C^0 = \begin{bmatrix} (u_{0x}, \varphi_{1x}) \\ \vdots \\ (u_{0x}, \varphi_{mx}) \end{bmatrix}$$

1.9.3 Quarta Opção

Seja $U^0 \in V_m$ tal que:

$$\kappa(U^0 - u_0, v_{hx}) = 0 \quad \forall v_h \in V_m$$

Tomando $U^0(x) = \sum_{j=1}^{m} C_j^0 \varphi_j(x)$ e $v_h = \varphi_i$ para $i \in \{1, \ldots, m\}$, temos:

$$\kappa(\sum_{j=1}^{m} C_j^0 \varphi_j - u_0, \varphi_i) = 0$$

$$\sum_{j=1}^{m} C_j^0 \kappa(\varphi_j - u_0, \varphi_i) = 0$$

$$\sum_{j=1}^{m} C_j^0 \kappa(\varphi_j, \varphi_i) + \kappa(-u_0, \varphi_i) = 0$$

$$\sum_{j=1}^{m} C_j^0 \kappa(\varphi_j, \varphi_i) = \kappa(u_0, \varphi_i)$$

Que pode ser escrito na forma matriz-vetor tal que, usando a notação definida anteriormente, temos:

$$\mathcal{K}C^0 = \begin{bmatrix} \kappa(u_0, \varphi_1) \\ \vdots \\ \kappa(u_0, \varphi_m) \end{bmatrix}$$

Capítulo 2

Detalhes de Implementação

2.1 Inicialização de C0

2.1.1 Primeira Opção

Vimos que a primeira opção para inicializar C^0 é:

$$C^0 = \begin{bmatrix} u_0(h) \\ \vdots \\ u_0(1-h) \end{bmatrix}$$

que é facilmente implementável como veremos a seguir.

2.1.2 Segunda Opção

Vimos que a segunda opção para inicializar C^0 é:

$$MC^0 = \begin{bmatrix} (u_0, \varphi_1) \\ \vdots \\ (u_0, \varphi_m) \end{bmatrix}$$

$$C^0 = M \setminus \begin{bmatrix} (u_0, \varphi_1) \\ \vdots \\ (u_0, \varphi_m) \end{bmatrix}$$

A matriz M pode ser facilmente inicializada utilizando o inicializador da K e o vetor da parte direita da equação pode ser facilmente inicializado utilizando construtor da F, como veremos em breve.

2.1.3 Terceira Opção

Vimos que a terceira opção para inicializar C^0 é:

$$\begin{bmatrix} (\varphi_{1x}, \varphi_{1x}) & \dots & (\varphi_{mx}, \varphi_{1x}) \\ \vdots & \ddots & \vdots \\ (\varphi_{1x}, \varphi_{mx}) & \dots & (\varphi_{mx}, \varphi_{mx}) \end{bmatrix} C^0 = \begin{bmatrix} (u_{0x}, \varphi_{1x}) \\ \vdots \\ (u_{0x}, \varphi_{mx}) \end{bmatrix}$$

$$C^0 = \begin{bmatrix} (\varphi_{1x}, \varphi_{1x}) & \dots & (\varphi_{mx}, \varphi_{1x}) \\ \vdots & \ddots & \vdots \\ (\varphi_{1x}, \varphi_{mx}) & \dots & (\varphi_{mx}, \varphi_{mx}) \end{bmatrix} \setminus \begin{bmatrix} (u_{0x}, \varphi_{1x}) \\ \vdots \\ (u_{0x}, \varphi_{mx}) \end{bmatrix}$$

Essa matriz pode ser facilmente inicializada utilizando o inicializador da K e o vetor teremos que fazer um construtor próprio. Cada termo desse vetor é:

$$\begin{aligned} \int_0^1 u_{0x}(x)\varphi_{ax}(x)\,dx &= \int_{-1}^1 u_{0x}(x(\xi,e))\varphi_{ax}^e(x(\xi,e))\frac{h}{2}\frac{2}{h}\,d\xi \\ &= \int_{-1}^1 u_{0x}(x(\xi,e))\varphi_{ax}^e(x(\xi,e))\,d\xi \\ &= \int_{-1}^1 u_{0x}(x(\xi,e))\phi_{ax}(\xi)\,d\xi \end{aligned}$$

Sendo assim, podemos fazer seu inicializador como veremos em breve.

2.1.4 Quarta Opção

Vimos que a quarta opção para inicializar C^0 é:

$$\mathcal{K}C^0 = \begin{bmatrix} \kappa(u_0, \varphi_1) \\ \vdots \\ \kappa(u_0, \varphi_m) \end{bmatrix}$$

A matriz K pode ser facilmente inicializada usando seu próprio construtor e o vetor teremos que realizar as contas. Cada um de seus termos é:

$$
\begin{aligned}
\kappa(u_0, \varphi_i) &= \alpha \int_0^1 u_{0x}(x)\varphi_{ix}(x)\, dx + \beta \int_0^1 u_0(x)\varphi_i(x)\, dx \\
&= \alpha \int_{-1}^1 u_{0x}(x(\xi, e))\varphi_{ix}((\xi, e))\frac{h}{2}\, d\xi + \beta \int_{-1}^1 u_0(x(\xi, e))\varphi_i(x(\xi, e))\frac{h}{2}\, d\xi \\
&= \alpha \int_{-1}^1 u_{0x}(x(\xi, e))\phi_{ix}(\xi)\frac{2}{h}\frac{h}{2}\, d\xi + \beta \int_{-1}^1 u_0(x(\xi, e))\phi_i(\xi)\frac{h}{2}\, d\xi \\
&= \alpha \int_{-1}^1 u_{0x}(x(\xi, e))\phi_{ix}(\xi)\, d\xi + \beta\frac{h}{2} \int_{-1}^1 u_0(x(\xi, e))\phi_i(\xi)\, d\xi
\end{aligned}
$$

Mas veja que a parte esquerda da equação é o C^0 que calculamos para a terceira opção multiplicado pela constante α. Além disso, veja que a parte direita da equação pode ser calculada utilizando o construtor da F multiplicado pela constante β.

Bibliografia

[1] Bruno Alves do Carmo. *Elementos Finitos*. Acessado em 2 de Outubro de 2024. URL: `https://github.com/bacarmo/Elementos_Finitos`.

[2] Thomas J. R. Hughes. *The Finite Element Method: Linear Static and Dynamic Finite Element Analysis*. Dover Civil and Mechanical Engineering. Dover Publications, 2000. ISBN: 0486411818; 9780486411811. URL: `libgen.li/file.php?md5=6ccd7300b50cdf2ec244ba5c248c4bc1`.

[3] João Victor Lopez Pereira. *Finite-Elements-Method*. Acessado em 20 de Outubro de 2024. URL: `https://github.com/joaovictorlopezpereira/Finite-Elements-Method`.