APLICAÇÃO DE CONDIÇÕES DE CONTORNO USANDO O MÉTODO DE ELEMENTOS FINITOS

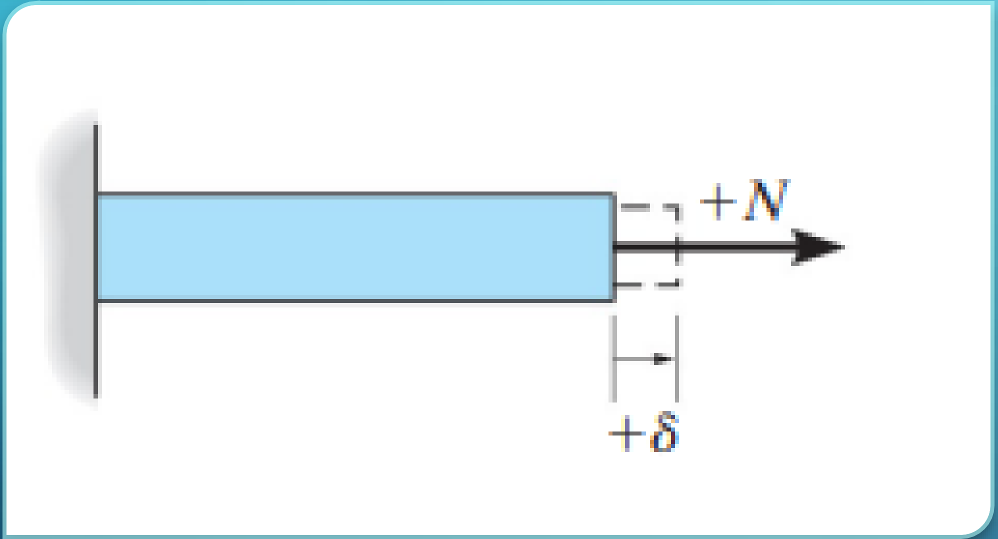
MODIFICAÇÃO DA MATRIZ DE RIGIDEZ E DO VETOR DE FORÇA PRA REALIZAR O CÁLCULO DO DESLOCAMENTO

MÉTODO DE ELEMENTOS FINITOS

- Conceitos básicos
- Função residual ponderada
- Formulação de Galerkin*
- Funções de forma
- Cálculo manual das funções de forma para o problema proposto

$$\int_0^L W(x)R(T,x)dx = 0 \qquad (2.11)$$

EQUAÇÕES DA MECÂNICA DOS SÓLIDOS



- Problema base
- Equações governantes
- Adaptação da equação de tensão e deformação, em termos de força, deslocamento e uma matriz constitutiva "C"
- Forma fraca e formulação de elementos finitos

MATRIZ DE RIGIDEZ

- Formulação do livro
- Implementação independente em Matlab com integrais simbólicas e coordenadas cartesianas
- Estudo do código já existente
- Comparação entre os dois

CONDIÇÕES DE CONTORNO - ABORDAGEM DIRETA

```
erro_abaqus =

   1.0e-12 *

         0
         0
         0
         0
    0.0027
   -0.0245
    0.0027
    0.0245
    0.3809
    0.0186
    0.3809
   -0.0186
```

- Estudo da formulação do livro
- Implementação direta no código independente para cálculo dos deslocamentos
- Comparação dos resultados com o Abaqus com uma carga N de 20E3, N usando a formulação de tensão plana

USO DO CÓDIGO EXISTENTE

- Problema formulado com os métodos do código já existente
- Matrix de rigidez calculada com método geral
- Resolução do mesmo problema, agora usando o código principal

FORMULAÇÃO GERAL PARA AS CONDIÇÕES DE CONTORNO

- Estudo das formulações diretas e gerais do livro
- Adaptação para código iterativo, alterando a matriz de rigidez e o vetor de força
- Implementação das condições de contorno de forma geral porém com vetores de Neumann e Dirichlet definidos de forma direta (na mão)

$$\begin{bmatrix} u_2 \\ u_3 \end{bmatrix} = \begin{bmatrix} k^{(1)} + k^{(2)} & -k^{(1)} \\ -k^{(1)} & k^{(1)} \end{bmatrix}^{-1} \left\{ \begin{bmatrix} -4 \\ 10 \end{bmatrix} - \begin{bmatrix} -k^{(2)} \\ 0 \end{bmatrix} [4/k^{(2)}] \right\},$$

$$\left[\begin{array}{c:cc} k^{(2)} & -k^{(2)} & 0 \\ \hdashline -k^{(2)} & k^{(1)} + k^{(2)} & -k^{(1)} \\ 0 & -k^{(1)} & k^{(1)} \end{array}\right] \left[\begin{array}{c} \bar{u}_1 \\ \hdashline u_2 \\ u_3 \end{array}\right] = \left[\begin{array}{c} r_1 \\ \hdashline -4 \\ 10 \end{array}\right] \quad \text{or} \quad \begin{bmatrix} \mathbf{K}_E & \mathbf{K}_{EF} \\ \mathbf{K}_{EF}^T & \mathbf{K}_F \end{bmatrix} \begin{bmatrix} \bar{\mathbf{d}}_E \\ \mathbf{d}_F \end{bmatrix} = \begin{bmatrix} \mathbf{r}_E \\ \mathbf{f}_F \end{bmatrix},$$

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & k^{(1)} + k^{(2)} & -k^{(1)} \\ 0 & -k^{(1)} & k^{(1)} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} = \begin{bmatrix} \bar{u}_1 \\ -4 - (-k^{(2)})\bar{u}_1 \\ 10 - (0)\bar{u}_1 \end{bmatrix}.$$

$$\mathbf{d}_F = \mathbf{K}_F^{-1}(\mathbf{f}_F - \mathbf{K}_{EF}^T \bar{\mathbf{d}}_E).$$

USO DA BCCLASS

- O código usado para resolver o problema já usa todos os métodos disponíveis no código original
- Transferência do código de implementação das condições de contorno para dentro da classe BCClass
- Código de exemplo usando a aplicação como um método dentro da BCClass

```matlab
%% Neumann vector and K matrix transformation
function [Kmod, Neumod] = KFTransform(Neu, Dir, K)
    k = zeros(size(Dir, 1), size(K, 2));
    for i = 1:size(Dir, 1)
        if Neu(Dir(i, 1)) ~= 0
            error('Make sure that there are not two types of BC at the same node.')
        end
        k(i, :) = K(Dir(i, 1), :);
        K(Dir(i, 1),:) = 0;
        K(:, Dir(i, 1)) = 0;
        K(Dir(i, 1), Dir(i, 1)) = 1;
    end
    Neumod = (Neu - k'*Dir(:, 2));
    Kmod = K;
end
```

COMPARAÇÃO COM MÉTODO ANALÍTICO

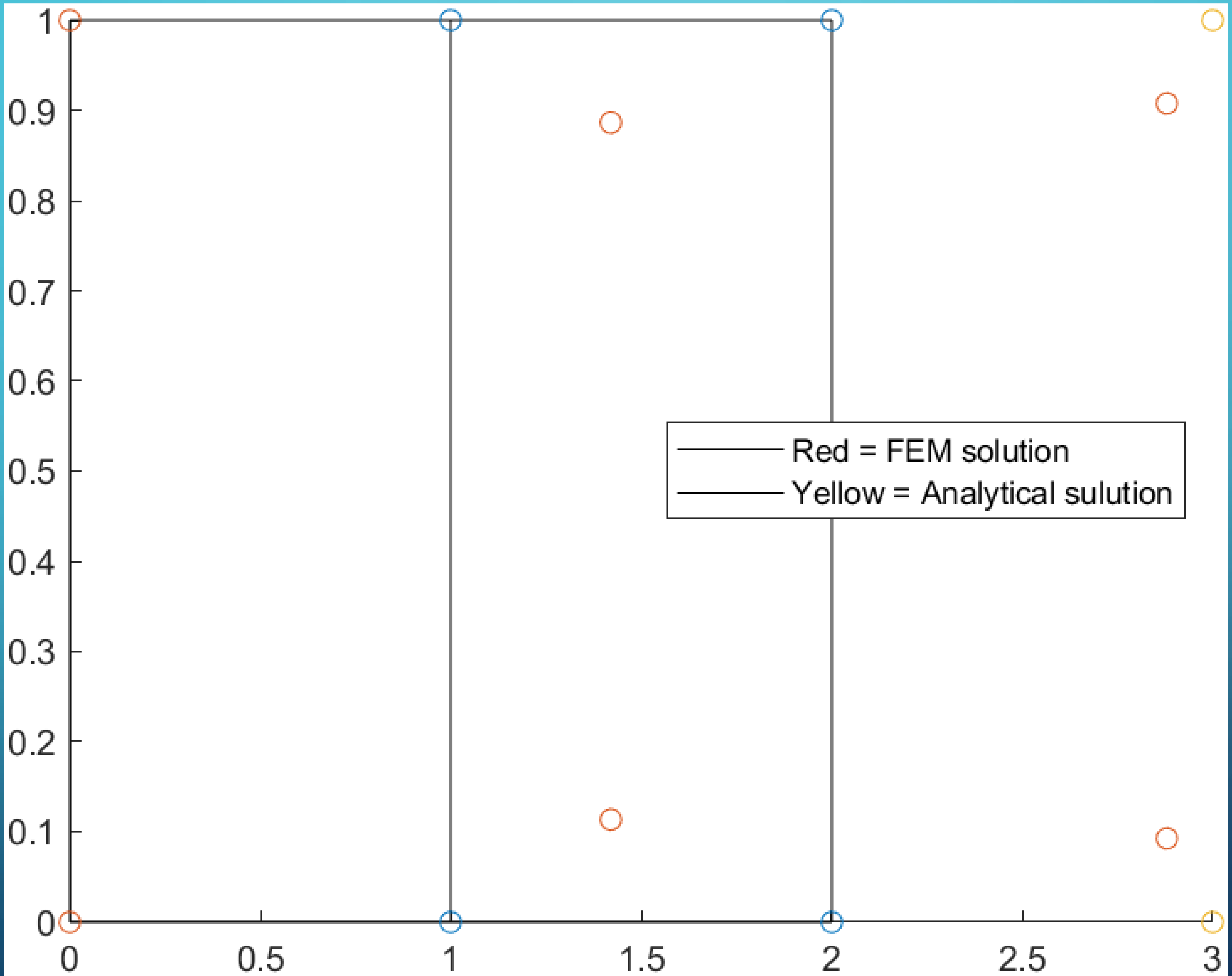
- Deformação elástica axial
- Comparação de resultados
- Diferenças de formulação de tensão e deformação plana
- Foi também comparado o resultado usando a formulação de <u>deformação plana</u>, que gerou um erro de 2.4047e-05.

```
N =

   20000000


A =

     1


L =

     2


E =

  2.0000e+11
```

```
delta =

   2.0000e-04


delta_FEM =

   1.7595e-04


erro =

   2.4047e-05
```

Constant Load and Cross-Sectional Area. In many cases the bar will have a constant cross-sectional area A; and the material will be homogeneous, so E is constant. Furthermore, if a constant external force is applied at each end, Fig. 4–3a, then the internal force N throughout the length of the bar is also constant. As a result, Eq. 4–1 when integrated becomes

$$\boxed{\delta = \frac{NL}{AE}} \qquad (4\text{–}2)$$



Red = FEM solution
Yellow = Analytical sulution
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
0
0.5
1
1.5
2
2.5
3