MEFPY

1 de junho de 2020

Sumário

1 Método dos Elementos Finitos com Python

Universidade Federal Fluminense

Disciplina ministrada pelo Prof. Marco Ferro em 1sem/2020.

Anotações do Aluno Noé de Lima noe_lima@id.uff.br

2 Introdução

2.1 Histórico

- Início na década de 50 pelos engenheiros aeronáuticos Turner, Argyris e Associados
- Análise Matricial de Estruturas, PTV
- Anos 60, MEF entendido como um caso particular do Método de Rayleigh-Ritz, logo idealizado a partir de funcionais. Logo o MEF pôde ser aplicado a problemas de fluidos, meios porosos, termodinâmica, eletromagnetismo, dentre outros.
- No final dos anos 60, foi comprovado que o MEF pode também ser considerado como um caso particular do Método de Galerkin, dispensando a necessidade de um funcional.
- Nos anos 70 o MEF foi identificado como um caso particular do MRP - Método dos Resíduos Ponderados. Expansão do MEF com computadores mais modernos e o advento de linguagens de baixo nível (FORTRAN e BASIC). O FORTRAN foi criado no meio da década de 1950, mas começou a crescer com o FORTRAN IV (1962), tendo um grande avanço com o FORTRAN 77.
- Nos anos 80 houve a disponibilidade e a explosão dos “microcomputadores” e vários grupos de pesquisa e desenvolvimento do MEF foram criados ao redor do planeta.
- Dos anos 90 até os dias atuais, vários programas comerciais foram disponibilizados, alguns considerando pré e pós processamento e a computação paralela.
- Para o dimensionamento de estruturas, no Brasil, os mais usados são o TQS e o EBERICK.
- Além disso, diversas linguagens são usadas para programação do MEF, como FORTRAN, C++, Matlab, dentre outras e, mais recentemente, Python.

3 Método Numéricos

(FAZER DIAGRAMA E INCLUIR AQUI)

Segue abaixo a importação das bibliotecas que serão utilizadas.

In [1]:
```
from numpy import array,arange,math,linspace
from scipy.interpolate import interp1d
import matplotlib.pyplot as plt
plt.close('all') # apaga plotagens anteriores
```

3.1 Exemplo

Considere a seguinte equação diferencial e suas condições de contorno:

$$\left[\begin{array}{ll} \frac{d^2u}{dx^2} - u = 0; & 0 \leqslant x \leqslant 1 \\ u\,(0) = 0 & u\,(1) = 1 \end{array}\right. \tag{1}$$

A solução exata deste sistema é:

$$u\,(x) = \frac{e^x - e^{-x}}{e - e^{-1}} \tag{2}$$

Este exemplo será utilizado nas seções a seguir.

3.1.1 Vetor x (Definição do Domínio)

Este código define o domínio x para o exemplo, e será utilizado como entrada nos demais códigos.

Neste exemplo, o domínio é $0 \leqslant x \leqslant 1$, discretizado em intervalos de $0,1$.

```
In [2]: x = linspace(0.,1.,11)
```

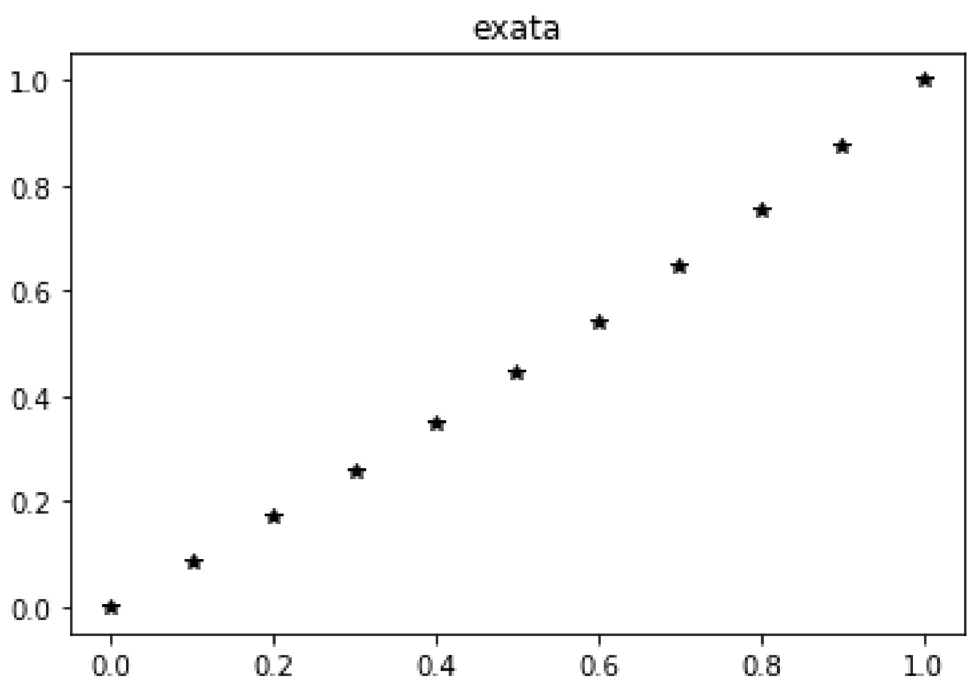
3.1.2 Definição da Solução Exata

Definição e plotagem (discreta) da solução exata para o exemplo proposto. Esta solução será utilizada para comparar os métodos de aproximação a seguir.

```
In [3]: def yexata(x):
            return (math.exp(1)**(x)-math.exp(1)**(-x))/(math.exp(1)-math.exp(-1))
```

```
In [4]: fig=plt.figure(1)
        plt.plot(x,yexata(x),'k*')
        plt.title('exata')
```

```
Out[4]: Text(0.5, 1.0, 'exata')
```



3.2 MRP - Método dos Resíduos Ponderados

Considere uma equação diferencial em $u(x)$ como mostrado a seguir, e suas condições de contorno.

$$\begin{bmatrix} L\left[u\left(x\right)\right] = f\left(x\right); & a \leqslant x \leqslant b \\ u\left(a\right) = u_a; & u\left(b\right) = u_b \end{bmatrix} \tag{3}$$

(1)

A solução aproximada de $u(x)$ é $\bar{u}(x)$ e possui a seguinte forma:

$$\bar{u}\left(x\right) = \phi_0\left(x\right) + \sum_{i=1}^{n} a_i \phi_i\left(x\right) \tag{4}$$

(2)

Onde ϕ_i são as funções de aproximação (a princípio arbitrárias) e a_i são coeficientes.

Substituindo $\bar{u}$ em (1), tem-se:

$$L\left[\bar{u}\left(x\right)\right] - f\left(x\right) = R \tag{5}$$

(3)

Onde R é denominado Resíduo.

O MRP consiste em ponderar o Resíduo, de modo que a integral ao longo do domínio seja nula. Logo,

$$\int_D W_i R dv = 0 \tag{6}$$

(4)

(SRP - Sentença de Resíduos Ponderados)

Sendo, * $W_i \rightarrow$ as funções de ponderação; * $D \rightarrow$ o domínio e; * $dv \rightarrow$ os diferenciais relativos ao domínio considerado.

3.2.1 Analogia

É como se houvesse uma compensação de áreas situadas entre y e $\bar{y}$, com ponderação.

(INCLUIR GRÁFICO)

3.3 Subdomínio (MVF - Método dos Volumes Finitos)

3.3.1 Função de Ponderação

$$W_i\left(x\right) = \begin{bmatrix} 1 & x \in D \\ 0 & x \notin D \end{bmatrix} \tag{7}$$

(5)

3.3.2 Exemplo

Considere a seguinte equação diferencial e suas condições de contorno para a qual obtivemos a Solução Exata.

$$\left[\begin{array}{l} \frac{d^2u}{dx^2} - u = 0; \quad 0 \leqslant x \leqslant 1 \\ u\left(0\right) = 0 \qquad u\left(1\right) = 1 \end{array}\right. \tag{8}$$

A função de aproximação escolhida por simplicidade é um polinômio da forma x^i. Logo,

$$\bar{u}\left(x\right) = \sum_{i=0}^{n+1} C_i x^i \tag{9}$$

Para que as condições de contorno sejam atendidas, temos:

$$\bar{u}\left(0\right) = \sum_{i=0}^{n+1} C_i x^i = 0 = \sum_{i=0}^{n+1} C_i 0^i \tag{10}$$

Mas,

$$0^i = \left[\begin{array}{ll} 1 & i = 0 \\ 0 & i \neq 0 \end{array}\right. \tag{11}$$

Logo,

$$\sum_{i=0}^{n+1} C_i 0^i = C_0 0^0 + \sum_{i=1}^{n+1} C_i 0^i = C_0 + \sum_{i=1}^{n+1} C_i 0 = 0 \therefore C_0 = 0 \tag{12}$$

Bem como,

$$\bar{u}\left(1\right) = 1 \rightarrow \sum_{i=1}^{n+1} C_i 1^i = \sum_{i=1}^{n+1} C_i 1 = 1 \therefore \sum_{i=1}^{n+1} C_i = 1 \tag{13}$$

Logo,

$$\bar{u}\left(x\right) = \sum_{i=1}^{n+1} C_i x^i \tag{14}$$

Utilizando $n = 1$,

$$\bar{u}\left(x\right) = C_1 x + C_2 x^2 \tag{15}$$

Com $C_1 + C_2 = 1$.
Assim, tem-se:

$$R = \frac{d^2\bar{u}}{dx^2} - \bar{u} = 2C_2 - C_1 x - C_2 x^2 \tag{16}$$

Utilizando o fator de ponderação $W_i = 1$, tem-se:

$$\int_0^1 1\left[2C_2 - C_1 x - C_2 x^2\right] dx = \left[2C_2 x - C_1 \frac{x^2}{2} - C_2 \frac{x^3}{3}\right]_0^1 = 2C_2 - \frac{C_1}{2} - \frac{C_2}{3} = 0 \tag{17}$$

$$\left[\begin{array}{lll} -\frac{1}{2}C_1 + \frac{5}{3}C_2 & = & 0 \\ C_1 + C_2 & = & 1 \end{array}\right. \tag{18}$$

A solução deste sistema é:

$$\begin{bmatrix} C_1 & = & \frac{10}{13} \\ C_2 & = & \frac{3}{13} \end{bmatrix} \tag{19}$$

Logo, tem-se

$$\bar{u}\left(x\right) = \frac{10}{13}x + \frac{3}{13}x^2 \tag{20}$$

Somando-se e subtraindo-se $\frac{3}{13}x$, obtém-se

$$\bar{u}\left(x\right) = x + \frac{3}{13}\left(x^2 - x\right) \tag{21}$$

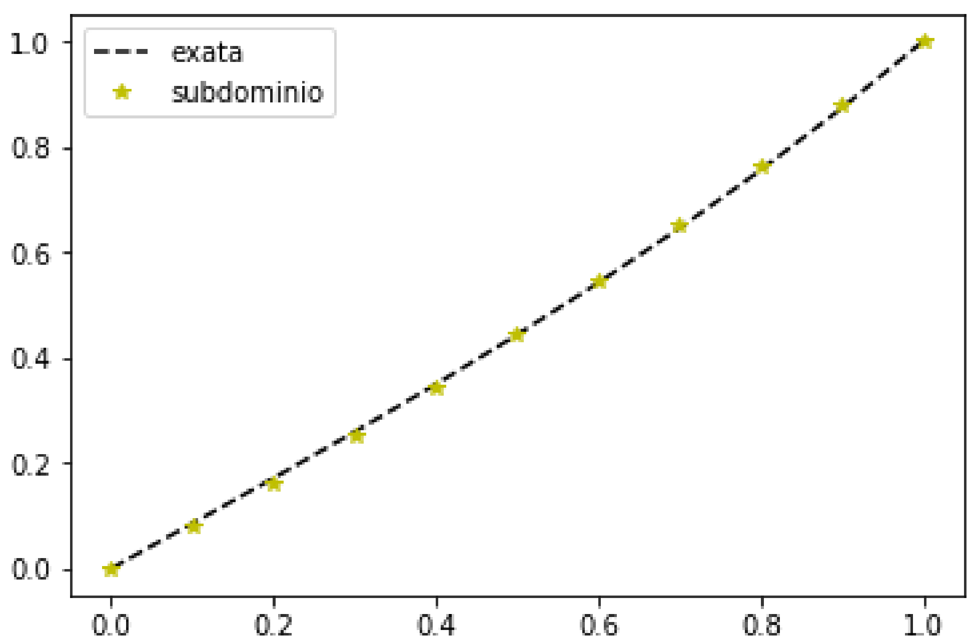
3.3.3 Implementação do Subdomínio

Abaixo, temos a implementação do MVF e o gráfico comparando com a solução exata.

```
In [5]: def ysubdominio(x):
            return x+3/13*x*(x-1)
```

```
In [6]: fig=plt.figure(2)
        plt.plot(x,yexata(x),'k--',x,ysubdominio(x),'y*')
        plt.legend(('exata','subdominio'))
```

```
Out[6]: <matplotlib.legend.Legend at 0x226440fc898>
```



3.4 Galerkin

No Método de Galerkin, a Função de Ponderação é igual a Função de Aproximação. Logo,

$$W_i(x) = \phi_i(x) \quad (i = 1, 2, 3, \ldots, n) \tag{22}$$

E a SRP - Sentença de Resíduos Ponderados, torna-se:

$$\int_D \phi_i R dv = 0 \tag{23}$$

Considere a mesma equação diferencial e as mesmas condições de contorno do exemplo anterior.

$$\begin{bmatrix} \frac{d^2u}{dx^2} - u = 0; & 0 \leqslant x \leqslant 1 \\ u(0) = 0 & u(1) = 1 \end{bmatrix} \tag{24}$$

Adota-se $W_i(x) = \phi_i(x) = x(x-1)$, com $n = 1$.

A SRP torna-se

$$\begin{gathered} \int_0^1 W_i R dx = \int_0^1 W_i \left(\frac{d^2\bar{u}}{dx^2} - \bar{u} \right) dx = \\ \left[W_i \frac{d\bar{u}}{dx} \right]_0^1 - \int_0^1 \left(\frac{dW_i}{dx} \frac{d\bar{u}}{dx} \right) dx - \int_0^1 W_i \bar{u} dx = 0 \end{gathered} \tag{25}$$

Lembrando que o resultado acima sai a partir da integração por partes, onde,

$$\int u dv = uv - \int v du \tag{26}$$

E, portanto,

$$\int W_i \frac{d^2\bar{u}}{dx^2} dx = \left[W_i \frac{d\bar{u}}{dx} \right] - \int \left(\frac{dW_i}{dx} \frac{d\bar{u}}{dx} \right) dx \tag{27}$$

Porém,

$$\left[W_i \frac{d\bar{u}}{dx} \right]_0^1 = 0 \tag{28}$$

Pois $W = x(x-1) = 0$ para $x = 0$ e $x = 1$. Logo,

$$\begin{gathered} -\int_0^1 \left(\frac{d\phi_i}{dx} \frac{d\bar{u}}{dx} \right) dx - \int_0^1 \phi_i \bar{u} dx = 0 \\ -\int_0^1 (2x-1)(C_1 + 2C_2 x) dx - \int_0^1 \left(x^2 - x\right) \left(C_1 x + C_2 x^2\right) dx = 0 \\ \int_0^1 \left(2C_1 x + 4C_2 x^2 - C_1 - 2C_2 x\right) dx + \\ \int_0^1 \left(C_1 x^3 + C_2 x^4 - C_1 x^2 - C_2 x^3\right) dx = 0 \\ \left[\tfrac{2}{2} C_1 x^2 + \tfrac{4}{3} C_2 x^3 - C_1 x - \tfrac{2}{2} C_2 x^2\right]_0^1 + \\ \left[\tfrac{1}{4} C_1 x^4 + \tfrac{1}{5} C_2 x^5 - \tfrac{1}{3} C_1 x^3 - \tfrac{1}{4} C_2 x^4\right]_0^1 = 0 \end{gathered} \tag{29}$$

Logo, simplificando,

$$\begin{gathered} \cancel{C_1} + \tfrac{4}{3} C_2 - \cancel{C_1} - C_2 + \tfrac{1}{4} C_1 + \tfrac{1}{5} C_2 - \tfrac{1}{3} C_1 - \tfrac{1}{4} C_2 = \\ \left(\tfrac{1}{4} - \tfrac{1}{3}\right) C_1 + \left(\tfrac{4}{3} - 1 + \tfrac{1}{5} - \tfrac{1}{4}\right) C_2 = 0 \\ \left(\tfrac{3-4}{12}\right) C_1 + \left(\tfrac{80-60+12-15}{60}\right) C_2 = 0 \\ -\tfrac{1}{12} C_1 \left(\tfrac{5}{5}\right) + \tfrac{17}{60} C_2 = 0 \end{gathered} \tag{30}$$

Portanto,

$$\begin{bmatrix} -5C_1 & + & 17C_2 & = & 0 \\ C_1 & + & C_2 & = & 0 \end{bmatrix} \tag{31}$$

A solução desse sistema é:

$$\begin{cases} C_1 & = & \frac{17}{22} \\ C_2 & = & \frac{5}{22} \end{cases} \tag{32}$$

Assim,

$$\bar{u}\left(x\right) = x + \frac{5}{22}x\left(x-1\right) \tag{33}$$

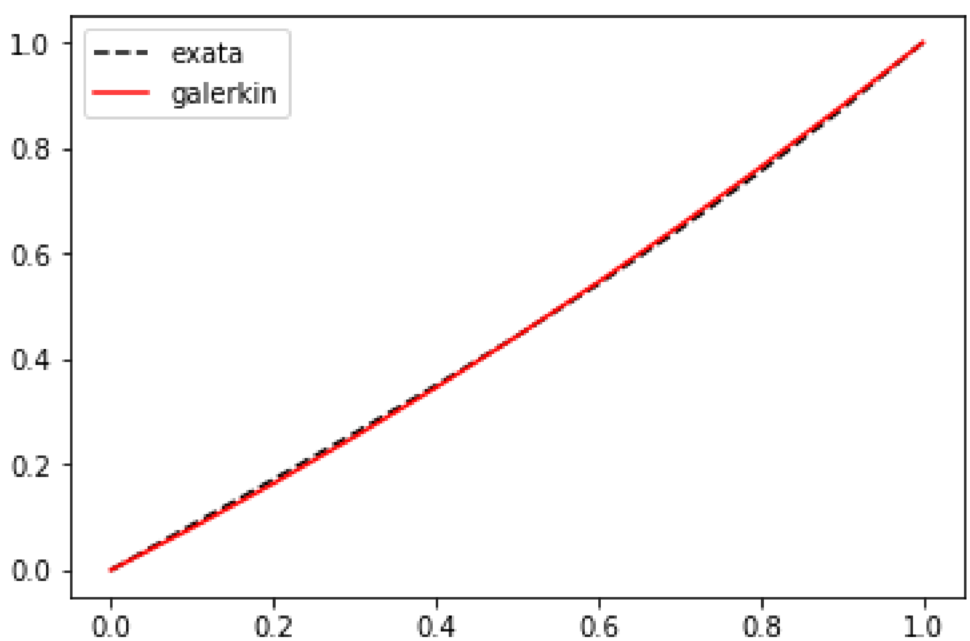
3.4.1 Implementação do Método de Galerkin

Abaixo, segue a implementação do Método de Galerkin e o gráfico comparando a solução exata.

```
In [7]: def ygalerkin(x):
            return x+5/22*x*(x-1)
```

```
In [8]: fig=plt.figure(3)
        plt.plot(x,yexata(x),'k--',x,ygalerkin(x),'r-')
        plt.legend(('exata','galerkin'))
```

```
Out[8]: <matplotlib.legend.Legend at 0x22646770240>
```



3.5 Rayleigh-Ritz

É utilizado quando existe um Funcinoal equivalente à uma equação diferencial.

- PTV - Princípio dos Trabalhos Virtuais
- EPM - Energia Potencial Mínima

Considere $\Pi(u)$ o funcional equivalente à equação diferencial

$$\frac{d^2u}{dx^2} - u = 0 \tag{34}$$

O MRR - Método de Rayleigh-Ritz considera que a solução aproximada $\bar{u}(x)$ é uma combinação linear de funções de aproximação $\phi_i(x)$, ou seja,

$$\bar{u}\left(x\right) = \sum_{i=1}^{n} a_i \phi_i\left(x\right) \tag{35}$$

Sendo $a_i, (i = 1.2. \ldots, n)$ coeficientes arbitrários e $\phi_i(x)$ devem ser escolhidos para atenderem as condições de contorno:

$$\begin{bmatrix} \sum_{i=1}^{n} a_i \phi_i\left(a\right) & = & u_a \\ \sum_{i=1}^{n} a_i \phi_i\left(b\right) & = & u_b \end{bmatrix} \tag{36}$$

A solução aproximada $\bar{y}$ deve tornar o Funcional "estacionário" em relação às constantes a_i, ou seja, nulo. Logo,

$$\frac{\partial \Pi}{\partial a_i} = 0 \quad (i = 1, 2, \ldots, n) \tag{37}$$

O Funcional Π da equação diferencial $\frac{d^2u}{dx^2} - u = 0$ vale

$$\Pi\left(u\right) = \int_0^1 \left[\frac{1}{2}\left(\frac{du}{dx}\right)^2 + \frac{1}{2}u^2\right] dx \tag{38}$$

Supondo $\bar{u} = x + ax(x+1)$,

$$\begin{matrix} \Pi\left(\bar{u}\right) = \int_0^1 \left[\frac{1}{2}\left(1 + a\left(2x - 1\right)\right)^2 + \frac{1}{2}\left(x + a\left(x^2 - x\right)\right)^2\right] dx \\ \Pi\left(\bar{u}\right) = \frac{2}{3} - \frac{a}{12} + \frac{a^2}{3} \end{matrix} \tag{39}$$

Como

$$\begin{matrix} \frac{\partial \Pi(\bar{u})}{\partial a} = 0 \therefore -\frac{1}{12} + \frac{2}{3}a = 0 \\ a = \frac{1}{8} \end{matrix} \tag{40}$$

Portanto,

$$\bar{u}\left(x\right) = x + \frac{1}{8}x\left(x - 1\right) \tag{41}$$

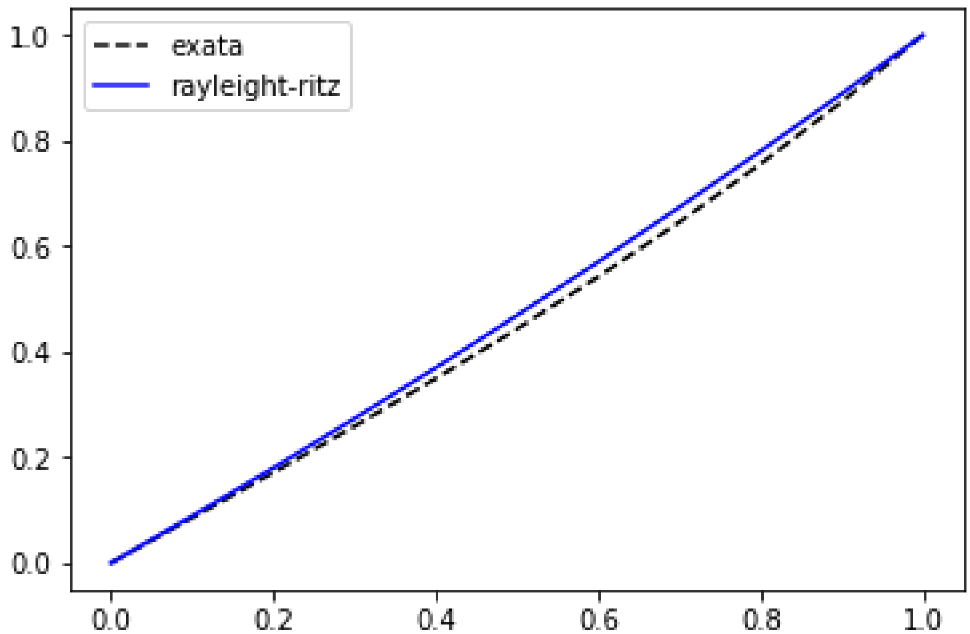
3.5.1 Implementação do Método de Rayleigh-Ritz

A seguir, a implementação do Método de Rayleigh-Ritz, bem como o gráfico de comparação com a solução exata.

```
In [9]: def yRR (x):
            return x+1/8*x*(x-1)
```

```
In [10]: fig=plt.figure(4)
         plt.plot(x,yexata(x),'k--',x,yRR(x),'b-')
         plt.legend(('exata','rayleight-ritz'))
```

```
Out[10]: <matplotlib.legend.Legend at 0x2264415ecf8>
```



4 MEF - Método dos Elementos Finitos

O Método dos Elementos Finitos pode ser entendido como derivado do Método de Galerkin, onde, na SRP (Sentença de Resíduos Ponderados), $W_i = \phi_i$, considera-se o domínio $1 - D$ (unidimensional) dividido em segmentos ou elementos, sendo os seus pontos extremos denominados nós.

(INSERIR IMAGEM)

Tem-se que:

$$\bar{u} = \sum_{i=1}^{n+1} u_i N_i(x) \tag{42}$$

No segmento e, temos:

$$h_e = x_e^{(2)} - x_e^{(1)} \tag{43}$$

E, tem-se que

$$\bar{u}(x) = \sum_{i=1}^{n+1} u_i N_i(x) \tag{44}$$

Assim, localmente,

$$N_1(e) = \frac{x_e^{(2)} - x}{x_e^{(2)} - x_e^{(1)}} = \frac{x_e^{(2)} - x_e^{(1)} - x_e}{x_e^{(2)} - x_e^{(1)}} \tag{45}$$

$$N_1(e) = \frac{h_e - x_e}{h_e} \quad \begin{bmatrix} N_1(x_e) = 1 \\ N_1(h_e) = 0 \end{bmatrix} \tag{46}$$

$$N_2\left(e\right) = \frac{x - x_e^{(1)}}{x_e^{(2)} - x_e^{(1)}} \tag{47}$$

$$N_1\left(e\right) = \tfrac{x_e}{h_e} \quad \begin{bmatrix} N_2\left(x_e\right) = 0 \\ N_2\left(h_e\right) = 1 \end{bmatrix} \tag{48}$$

Usando o Método de Galerkin na SRP, temos:

$$\int_0^1 N_i R dx = 0 \tag{49}$$

Ou,

$$\begin{gathered} \int_0^1 N_i \left(\tfrac{d^2\bar{u}}{dx^2} - \bar{u}\right) dx = \\ \left[N_i \tfrac{d\bar{u}}{dx}\right]_0^1 - \int_0^1 \left(\tfrac{dN_i}{dx}\tfrac{d\bar{u}}{dx} + N_i\bar{u}\right) dx = 0 \end{gathered} \tag{50}$$

Logo, lembrando que

$$\bar{u}\left(x\right) = \sum_{i=1}^{n+1} u_i N_i\left(x\right) \tag{51}$$

Obtém-se

$$\begin{gathered} \textstyle\sum_{j=1}^{n+1} \int_0^1 \left[\left(\tfrac{dN_i}{dx}\tfrac{dN_j}{dx}\right) u_j + \left(N_i N_j\right) u_j\right] dx - \left[N_i \tfrac{d\bar{u}}{dx}\right]_0^1 = 0 \\ \therefore \\ \textstyle\sum_{j=1}^{n+1} \int_0^1 \left(\tfrac{dN_i}{dx}\tfrac{dN_j}{dx} + N_i N_j\right) u_j dx - \left[N_i \tfrac{d\bar{u}}{dx}\right]_0^1 = 0 \end{gathered} \tag{52}$$

A partir da equação acima, define-se a Matriz de Rigidez K.

4.0.1 Matriz de Rigidez e Matriz de Força (ou Carregamento) Nodal

$$\begin{gathered} K_{ij} \equiv \int_0^1 \left(\tfrac{dN_i}{dx}\tfrac{dN_j}{dx} + N_i N_j\right) dx \qquad (i,j) \geqslant 1 \\ f_i \equiv \left[N_i \tfrac{d\bar{u}}{dx}\right]_0^1 \qquad\qquad (i,j) \leqslant n+1 \end{gathered} \tag{53}$$

4.1 MEF com 1 Elemento

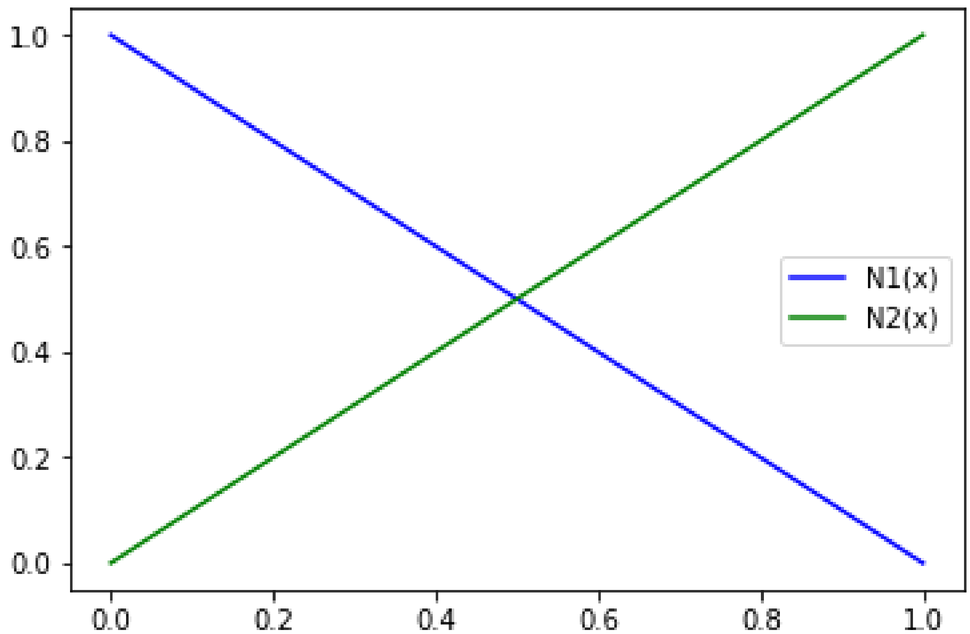
Aplicando o MEF ao exemplo utilizado até aqui:

$$\frac{d^2 u}{dx^2} - u = 0 \tag{54}$$

Subdividindo o domínio em um segmento (ou seja, sen divisão interna), temos:

```
In [11]: fig=plt.figure(8)
         plt.plot(x,1-x,'b-',x,x,'g-')
         plt.legend(('N1(x)','N2(x)'))

Out[11]: <matplotlib.legend.Legend at 0x226468457f0>
```



Onde,

- $N_1(x) = \frac{1-x}{1} = 1 - x$
- $N_2(x) = \frac{x}{1} = x$

Logo,

$$K_{ij} = \int_0^1 \frac{dN_i}{dx}\frac{dN_j}{dx} + N_i N_j dx \tag{55}$$

Se $i = j$,

$$\begin{gathered} K_{11} = K_{22} = \int_0^1 \left[(-1) \times (-1) + (1-x)^2\right] dx = \\ \int_0^1 \left(1 + 1 - 2x + x^2\right) dx = \int_0^1 \left(x^2 - 2x + 2\right) dx = \\ \left[\frac{x^3}{3} - \frac{\cancel{2}x^2}{\cancel{2}} + 2x\right]_0^1 = \frac{1}{3} - 1 + 2 = \frac{4}{3} \end{gathered} \tag{56}$$

Ou, alternativamente,

$$\begin{gathered} K_{11} = K_{22} = \int_0^1 \left[(1) \times (1) + (x)^2\right] dx = \int_0^1 (1+x)^2 dx = \\ \left[x + \frac{x^3}{3}\right]_0^1 = 1 + \frac{1}{3} = \frac{4}{3} \end{gathered} \tag{57}$$

Para $i \neq j$, temos:

$$\begin{gathered} K_{12} = K_{21} = \int_0^1 \left[(-1) \times (1) + (1-x)\, x\right] dx = \\ \int_0^1 \left(-1 + x - x^2\right) dx = \left[-x + \frac{x^2}{2} - \frac{x^3}{3}\right]_0^1 = \\ -\frac{1}{1} + \frac{1}{2} - \frac{1}{3} = \frac{-6+3-2}{6} = -\frac{5}{6} \end{gathered} \tag{58}$$

Portanto, temos:

$$\mathbf{K} = \begin{bmatrix} \frac{4}{3} & -\frac{5}{6} \\ -\frac{5}{6} & \frac{4}{3} \end{bmatrix} \tag{59}$$

Tem-se, também, que $\mathbf{K}\bar{u} = \vec{f}$, Ou,

$$\begin{bmatrix} \frac{4}{3} & -\frac{5}{6} \\ -\frac{5}{6} & \frac{4}{3} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} -\frac{d\bar{u}(0)}{dx} \\ -\frac{d\bar{u}(1)}{dx} \end{bmatrix} \tag{60}$$

Sabe-se, ainda, que:

$$\begin{bmatrix} u_1\,(0) = 0 \\ u_1\,(1) = 1 \end{bmatrix} \tag{61}$$

Assim,

$$\begin{bmatrix} \frac{4}{3} & -\frac{5}{6} \\ -\frac{5}{6} & \frac{4}{3} \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} -\frac{d\bar{u}(0)}{dx} \\ -\frac{d\bar{u}(1)}{dx} \end{bmatrix} \tag{62}$$

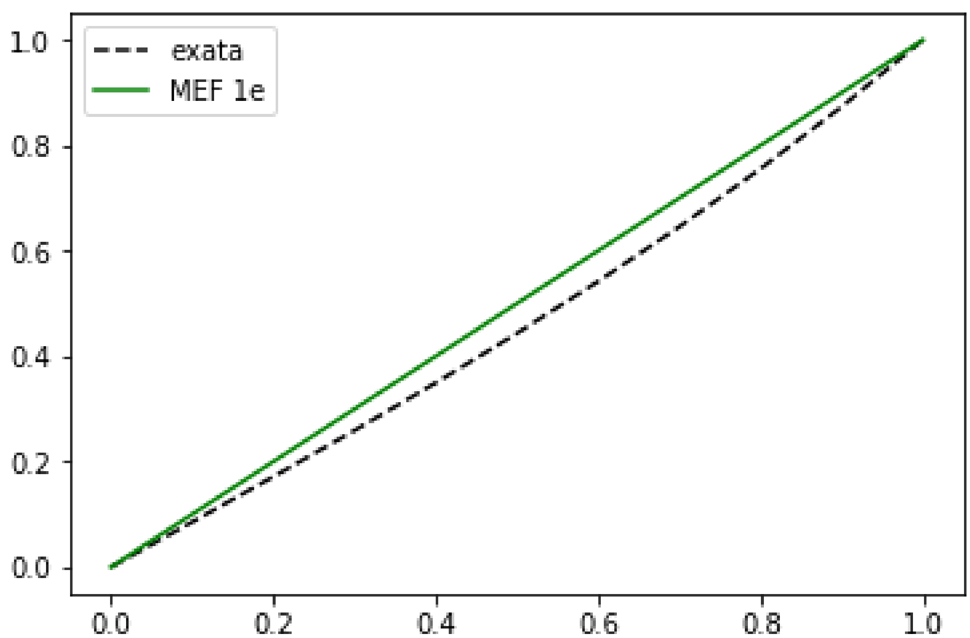
Resolvendo este sistema, obtém-se:

$$\begin{bmatrix} \frac{d\bar{u}(0)}{dx} & = & \frac{5}{6} \\ \frac{d\bar{u}(1)}{dx} & = & -\frac{4}{3} \end{bmatrix} \tag{63}$$

```
In [12]: def yFEM1E (x):
             return x
```

```
In [13]: fig=plt.figure(5)
         plt.plot(x,yexata(x),'k--',x,yFEM1E(x),'g-')
         plt.legend(('exata','MEF 1e'))
```

```
Out[13]: <matplotlib.legend.Legend at 0x22646f4aa90>
```



4.2 MEF com 3 Elementos

Agora veremos o mesmo exemplo aplicando o MEF com 3 Elementos.

$$\bar{u}\,(x) = \sum_{i=1}^{4} u_i N_i \tag{64}$$

Dividindo o domínio em 3 segmentos, temos 4 nós igualmente espaçados. São eles: $N_1 \to (x = 0)$, $N_2 \to (x = \frac{1}{3})$, $N_3 \to (x = \frac{2}{3})$ e $N_4 \to (x = 1)$.

Localmente (no elemento), temos:

$$\begin{bmatrix} N_1^e & = & \frac{h_e - x}{h_e} \\ N_2^e & = & \frac{x}{h_e} \end{bmatrix} \tag{65}$$

Onde $h_e = \frac{1}{3}$.

Portanto,

$$\begin{bmatrix} N_1^e & = & \frac{\frac{1}{3} - x}{\frac{1}{3}} & = & 1 - 3x \\ N_2^e & = & \frac{x}{\frac{1}{3}} & = & 3x \end{bmatrix} \tag{66}$$

Assim,

$$K_{ij} = \int_0^{\frac{1}{3}} \left[\left(\frac{dN_i}{dx} \frac{dN_j}{dx} \right) + N_i N_j \right] dx \tag{67}$$

$$f_i = \left[N_i \frac{d\bar{u}}{dx} \right] \therefore \vec{f} = \begin{bmatrix} -\frac{d\bar{u}(0)}{dx} \\ 0 \\ 0 \\ -\frac{d\bar{u}(1)}{dx} \end{bmatrix} \tag{68}$$

Ressaltando que só existe f_i em N_1 e N_4.

Assim,

$$\mathbf{K} = \begin{bmatrix} \frac{28}{9} & -\frac{53}{18} \\ -\frac{53}{18} & \frac{28}{9} \end{bmatrix} \tag{69}$$

Considerar que o nó N_2 é influenciado pelos elementos N_1 e N_2 e o nó N_3 pelos elementos N_2 e N_3.

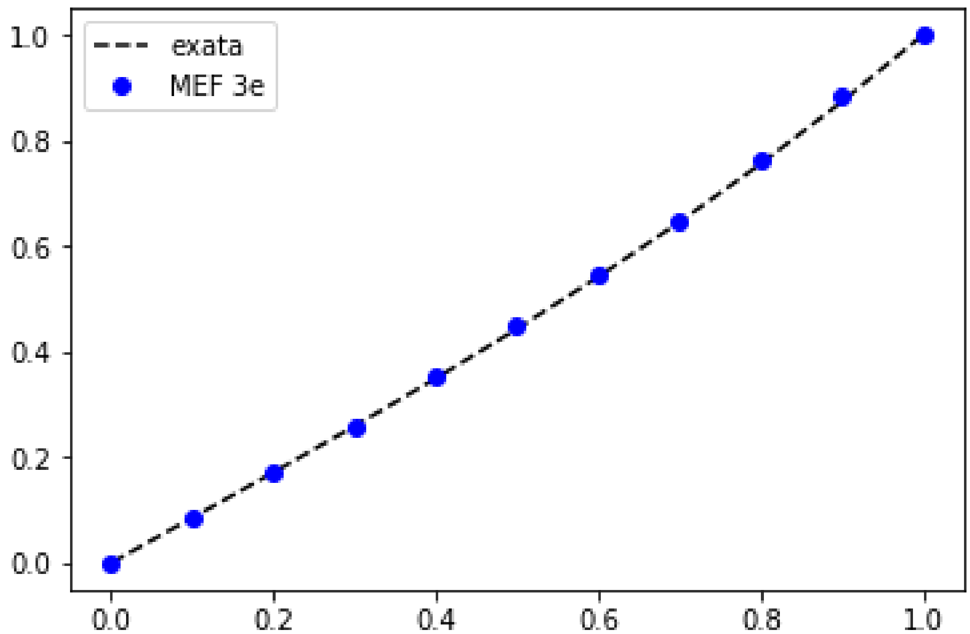
As condições de contorno de Dirichlet DEVEM ser atendidas, ou seja, $u_1 = 0$ e $u_4 = 1$. Logo, as incógnitas serão:

$$\frac{du(0)}{dx}, \quad u_2, \quad u_3, \quad \frac{du(1)}{dx} \tag{70}$$

```
In [14]: xfemtres=array([0.,0.333,0.667,1.])
         yfemtres=array([0,0.2885,0.6098,1.])
         ylin=interp1d(xfemtres,yfemtres,kind='linear')(x)
```

```
In [15]: fig=plt.figure(6)
         plt.plot(x,yexata(x),'k--',x,ylin,'bo')
         plt.legend(('exata','MEF 3e'))
```

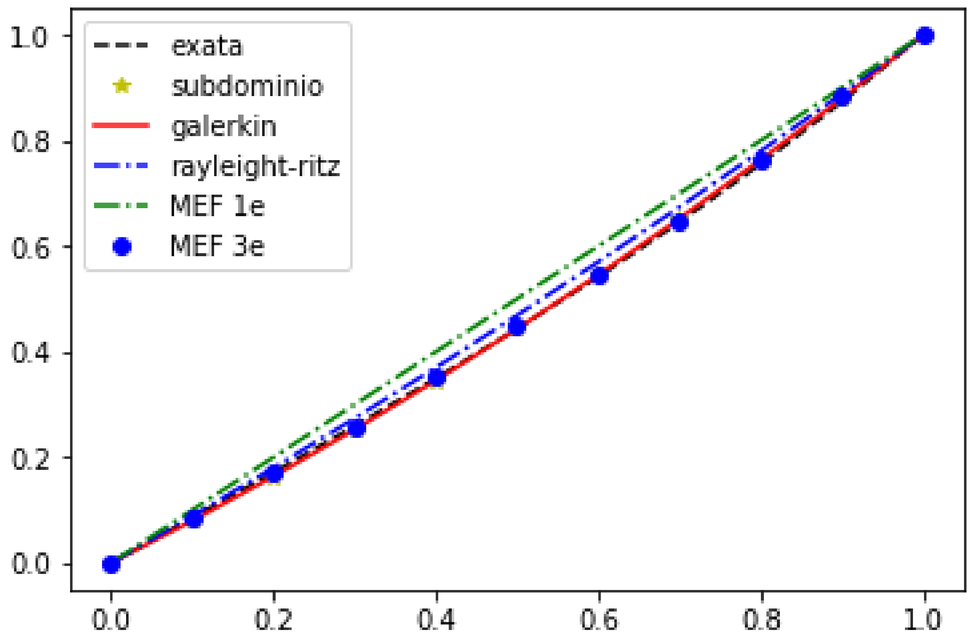
```
Out[15]: <matplotlib.legend.Legend at 0x22646a8a2b0>
```



5 Comparação das Soluções

```
In [16]: fig=plt.figure(7)
         plt.plot(x,yexata(x),'k--', x,ysubdominio(x),'y*',x,ygalerkin(x),'r-',x,yRR(x),'b-.',x,
         plt.legend(('exata','subdominio','galerkin','rayleight-ritz','MEF 1e','MEF 3e'))

Out[16]: <matplotlib.legend.Legend at 0x22646f8f320>
```



6 PTV - Princípio dos Trabalhos Virtuais

Considere a barra de treliça sujeita a uma força de tração f, com área de seção transversal A, comprimento l e Módulo de Elasticidade E.

(INSERIR FIGURA)

A medida que a treliça está sendo carregada, ocorre um alongamento até a posição de equilíbrio.

A curva Tensão × Deformação é mostrada a seguir:

(INSERIR FIGURA)

(INSERIR DETALHE)

No detalhe, temos:

$$\Delta U_{0m} = Area_{\square_{ABDE}} + Area_{curva-AB} \tag{71}$$

Assim,

$$\Delta U_{0m} = \underbrace{\sigma_m \delta\epsilon_m}_{\delta U_{0m}^{(1)}} + \underbrace{\frac{1}{2}\delta\sigma_m \delta\epsilon_m}_{\delta U_{0m}^{(2)}} + erro_{U_{0m}} \tag{72}$$

$$\Delta U_{0m} = \delta U_{0m}^{(1)} + \delta U_{0m}^{(2)} + erro \tag{73}$$

6.1 Incremento do Trabalho Interno (Energia de Deformação)

$$\Delta U_m = \int_0^{U_m} \Delta U_{0m} dU_m \tag{74}$$

$$\Delta U_m = \int_0^{U_m} \sigma_m \delta\epsilon_m dU_m + \int_0^{U_m} \frac{1}{2}\delta\sigma_m \delta\epsilon_m dU_m + \int_0^{U_m} erro\ dU_m \tag{75}$$

$$\delta U_m^{(1)} = \int_0^{U_m} \sigma_m \delta\epsilon_m dU_m \tag{76}$$

$$\delta U_m^{(2)} = \int_0^{U_m} \frac{1}{2}\delta\sigma_m \delta\epsilon_m dU_m \tag{77}$$

Para o caso de n_e barras,

$$\delta U^{(1)} = \sum_{n=i}^{n_e} \int_0^{U_m^{(i)}} \sigma_m^{(i)} \delta\epsilon_m^{(i)} dU_m^{(i)} \tag{78}$$

$$\delta U^{(2)} = \sum_{n=i}^{n_e} \int_0^{U_m^{(i)}} \frac{1}{2}\delta\sigma_m^{(i)} \delta\epsilon_m^{(i)} dU_m^{(i)} \tag{79}$$

Sendo,

- $\sigma_m^{(i)}$ = Tensão na barra i;
- $\delta\sigma_m^{(i)}$ = incremento de Tensão na barra i;
- $\delta\epsilon_m^{(i)}$ = incremento de Deformação na barra i.

A força externa atuando na treliça relaciona-se com o deslocamento de acordo com o gráfico:
(INSERIR GRÁFICO)
(INSERIR DETALHE)

6.2 Incremento do Trabalho Externo

$$\Delta W_i = \underbrace{f_i \delta d_i}_{\delta W_i^{(1)}} + \underbrace{\frac{1}{2}\delta f_i \delta d_i}_{\delta W_i^{(2)}} + erro_{W_i} \tag{80}$$

Considerando-se n_e elementos de treliça,

$$dW^{(1)} = \sum_{i=1}^{n_e} f_i \delta d_i \tag{81}$$

$$dW^{(2)} = \sum_{i=1}^{n_e} \frac{1}{2}\delta f_i \delta d_i \tag{82}$$

Ou, na forma vetorial,

$$dW^{(1)} = \sum_{i=1}^{n_e} \vec{f} \cdot \delta\vec{d} \tag{83}$$

$$dW^{(2)} = \sum_{i=1}^{n_e} \frac{1}{2}\delta\vec{f} \cdot \delta\vec{d} \tag{84}$$

Sendo:

- $\vec{f} \rightarrow$ vetor de forças nas barras da treliça;
- $\delta\vec{f} \rightarrow$ incremento de $\vec{f}$; e
- $\delta\vec{d} \rightarrow$ incremento dos deslocamentos das barras da treliça.

7 PDV - Princípio dos Deslocamentos Virtuais

Nesse caso, equivale ao PTV.

Obs: Existe também o Princípio das Forças Virtuais, mas não será tratado neste contexto.

PDV: Em um sistema Estrutural em equilíbrio, o incremento de 1^a ordem da energia de deformação é igual ao incremento de 1^a ordem do trabalho externo.

Logo, $\delta U' = \delta W'$, ou

$$\sum_{n=1}^{n_e} \int_0^{U_m^{(i)}} \sigma_m^{(i)} \delta \epsilon_m^{(i)} dU_m^{(i)} = \vec{f} \cdot \delta \vec{d} \tag{85}$$

$\sigma_m^{(i)}$ e $\vec{f}$ são grandezas reais e em equilíbrio, enquanto $\delta \epsilon_m^i$ e $\delta \vec{d}$ são grandezas virtuais.

8 Energia Potencial Total do Sistema

$$\Pi = U + W \therefore \Pi(d) = U(d) + W(d) \tag{86}$$

U é a energia (interna) que volta a estrutura para a configuração inicial se houver o descarregamento total.

W é o trabalho que será realizado caso a estrutura voltasse para a configuração inicial, mantido o carregamento.

Π é a energia total necessária para a estrutura voltar à condição inicial, mantido o carregamento. Ou seja, para voltar a condição inicial, a estrutura precisa "vencer" o carregamento e a deformação sofridos.

Tem-se que

$$W(d) = -\vec{f} \cdot \vec{d} \tag{87}$$

9 Princípio da Energia Potencial Total Mínima

Um sistema estrutural carregado sofrerá deformação de modo a gastar o mínimo de Energia Potencial Mínima.

Logo, no equilíbrio,

$$\frac{\partial \Pi(d)}{\partial d} = 0 \tag{88}$$

$\Pi(d)$ é denominado Energia Potencial Total do Sistema, ou FUNCIONAL DO SISTEMA.

Logo, a equação $\Pi(d) = U(d) + W(d)$ equivale às Equações de Equilíbrio do Sistema.

9.1 Exemplo

Considere a equação

$$\tfrac{d^2u}{dx^2} - u = 0; \quad 0 \leqslant x \leqslant 1 \tag{89}$$

Prove que o Funcional da EDO anterior vale:

$$\Pi(u) = \int_0^1 \left[\frac{1}{2}\left(\frac{du}{dx}\right)^2 + \frac{1}{2}u^2\right] dx \tag{90}$$

9.1.1 Solução

Tem-se que

$$d\Pi\left(u\right) = \int_0^1 \left[\frac{du}{dx} d\left(\frac{du}{dx}\right) + u\delta u\right] dx \tag{91}$$

Obs:

$$v = \frac{du}{dx} \rightarrow \frac{1}{2}v^2 = \frac{1}{2}\left(\frac{du}{dx}\right)^2 \tag{92}$$

$$d\left(\frac{1}{2}v^2\right) = \frac{1}{\cancel{2}} \times \cancel{2} \times v \times dv = vdv = \frac{du}{dx} d\left(\frac{du}{dx}\right) \tag{93}$$

$$d\left(\frac{1}{2}u^2\right) = \frac{1}{\cancel{2}} \times \cancel{2} \times u \times du = udu \tag{94}$$

$d\Pi$ pode ser reescrito como

$$d\Pi = \int_0^1 \left[\frac{du}{dx} d\left(\frac{du}{dx}\right) + udu\right] dx \tag{95}$$

Integrando por partes,

$$\int_0^1 \left(\frac{du}{dx} d\left(\frac{du}{dx}\right) + udu\right) dx = \left[\frac{du}{dx} du\right]_0^1 - \int_0^1 \left[\frac{d}{dx}\frac{du}{dx} du - udu\right] dx \tag{96}$$

As condições de contorno de Dirichlet devem ser atendidas:

$$du = 0 \quad se \quad \left[\begin{array}{l} x = 0 \\ x = 1 \end{array}\right. \tag{97}$$

Logo,

$$\left[\frac{du}{dx} du\right] = 0 \tag{98}$$

$$d\Pi = -\int_0^1 \left(\frac{d^2u}{dx^2} - u\right) dudx \tag{99}$$

Condição de "Estacionariedade" $d\Pi = 0$; logo,

$$\int_0^1 \left(\frac{d^2u}{dx} - u\right) dudx = 0 \tag{100}$$

Logo,

$$\frac{d^2u}{dx^2} - u = 0 \tag{101}$$

10 Cálculo Variacional, Funcional