

O Método De Rayleigh-Ritz aplicado a um Problema de Deflexão de Viga utilizando o Matlab

Yuri Elias Rodrigues* **Daniel da Silva Rodrigues****

Pontifícia Universidade Católica do Rio Grande do Sul – Faculdade de Matemática
90619-900, Av. Ipiranga, 6681 - Partenon - Porto Alegre, RS
E-mail: yuri.rodrigues@acad.pucrs.br

Eliete Biasotto Hauser

Pontifícia Universidade Católica do Rio Grande do Sul – Faculdade de Matemática
90619-900, Av. Ipiranga, 6681 - Partenon - Porto Alegre, RS
E-mail: eliete@pucrs.br

RESUMO

Este trabalho tem como objetivo avaliar os resultados do Método de Rayleigh-Ritz (MRR) aplicado ao seguinte Problema de Valor de Contorno (PVC) que modela a deflexão de uma viga apoiada sobre seus extremos:

$$\begin{cases} -\frac{d}{dx}\left(p(x)\frac{dy}{dx}\right) + q(x)y = f(x), & 0 \leq x \leq 1 \\ y(0) = y(1) = 0. \end{cases} \quad (1)$$

A solução exata de (1), $y(x)$, é aproximada por uma combinação linear de funções de base ϕ_i .

$$\Phi(x) = \sum_{i=1}^n c_i \phi_i(x). \quad (2)$$

As funções ϕ_i devem satisfazer as condições de fronteira e serem linearmente independentes. Para atender esta imposição selecionamos como funções de base, as funções lineares por partes expressas por:

$$\phi_i(x) = \begin{cases} 0, & \text{se } 0 \leq x \leq x_{i-1} \\ \frac{x-x_{i-1}}{h_{i-1}}, & \text{se } x_{i-1} < x \leq x_i \\ \frac{x_{i+1}-x}{h_i}, & \text{se } x_i < x \leq x_{i+1} \\ 0, & \text{se } x_{i+1} < x \leq 1, \end{cases} \quad (3)$$

ilustradas graficamente na figura 1.

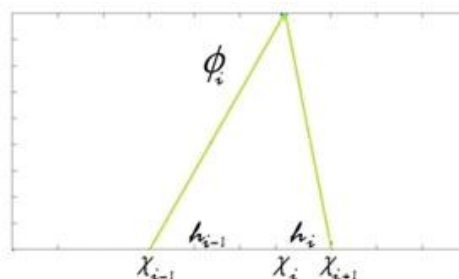


Figura 1. Representação gráfica de $\phi_i(x)$.

Em (3) temos que $h_i = x_{i+1} - x_i$. A escolha deste espaçamento não constante, possibilita que a aproximação seja melhorada utilizando o mesmo número nodos.

* Bolsista de Iniciação Científica PIBIC/CNPq

** Bolsista de Iniciação Científica BPA/PUCRS

Para determinar os coeficientes c_i é necessário, de acordo com [3], minimizar o funcional associado ao PVC (1), portanto, de forma geral temos que:

$$I(y) = \int_0^1 \left\{ p(x) \left[\frac{dy(x)}{dx} \right]^2 + q(x)[y(x)]^2 - 2f(x)y(x) \right\} dx. \quad (4)$$

Para tanto, substituímos (2) na equação (4) obtendo:

$$I = \int_0^1 \left\{ p(x) \left[\sum_{i=1}^n \frac{d\phi_i(x)}{dx} c_i \right]^2 + q(x) \left[\sum_{i=1}^n \phi_i(x) c_i \right]^2 - 2f(x) \sum_{i=1}^n \phi_i(x) c_i \right\} dx. \quad (5)$$

O valor mínimo do funcional ocorre quando sua derivada parcial em relação á c_i é igual a 0, ou seja

$$\frac{\partial I}{\partial c_j} = 0. \quad (6)$$

Então,

$$\int_0^1 f(x)\phi_j(x)dx = \sum_{i=1}^n \int_0^1 \left\{ p(x) \frac{d\phi_i(x)}{dx} \frac{d\phi_j(x)}{dx} + q(x)\phi_i(x)\phi_j(x) \right\} dx. \quad (7)$$

A equação (7) pode ser representada matricialmente na forma

$$A.C = B, \quad (8)$$

onde A é uma matriz tridiagonal de ordem $n \times n$, e seus elementos são expressos por

$$a_{ij} = \int_0^1 \left\{ p(x) \frac{d\phi_i(x)}{dx} \frac{d\phi_j(x)}{dx} + q(x)\phi_i(x)\phi_j(x) \right\} dx. \quad (9)$$

Os elementos do vetor B são dados por

$$b_i = \int_0^1 f(x)\phi_i(x)dx. \quad (10)$$

Para validar a solução aproximada de (1) consideremos o PVC que tem solução conhecida:

$$\begin{cases} -\frac{d}{dx} \left(e^x \frac{dy}{dx} \right) + e^x y = x + (2-x)e^x, & 0 \leq x \leq 1 \\ y(0) = y(1) = 0. \end{cases} \quad (11)$$

O qual tem solução exata

$$y(x) = (x-1)(e^{-x} - 1). \quad (12)$$

O algoritmo MRR foi implementado em linguagem de programação do software Matlab. Com $n = 2$, $n = 5$ e $n = 10$, e h assumindo respectivamente os valores $1/3$, $1/6$ e $1/11$, ilustramos graficamente os resultados na figura 3, em que: a aproximação obtida pelo MRR, $\Phi(x)$, esta em azul; a solução exata, $y(x)$, esta em vermelho; o erro absoluto, $E(x) = |y(x) - \Phi(x)|$, esta em verde.

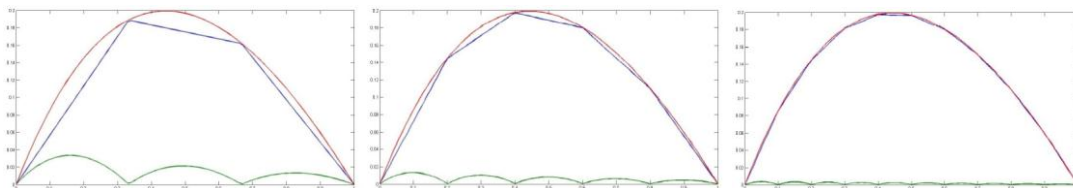


Figura 3. $y(x)$, $\Phi(x)$ e $E(x)$, com $n = 2$, $n = 5$ e $n = 10$.

Na tabela 1, são apresentados os valores dos erros absolutos para as simulações ilustradas na figura 3, e seu padrão sugere a convergência de $\Phi(x)$ para $y(x)$, à medida que aumentarmos a quantidade de nodos.

x_i	$n_1 = 2$	$n_2 = 5$	$n_3 = 10$
0	0	0	0
0.1	0.029025685184959	0.013232147308362	0.000078048779597
0.2	0.031774120372259	0.000187044619065	0.000148785680326
0.3	0.011565329774764	0.010275050194529	0.000201733484602
0.4	0.014444794880438	0.000331934640629	0.000231318797700
0.5	0.021429919308394	0.007923088068827	0.000235833486994
0.6	0.013229021389989	0.000328219150663	0.000216594059407
0.7	0.032409319228047	0.005978229257140	0.000177264949847
0.8	0.024737309003027	0.000188974377407	0.000123317302782
0.9	0.026966270242601	0.004370417626366	0.000615995439889
1	0	0	0

Tabela 1. Erros absolutos, $E(x)$.

Adicionalmente podemos verificar nas simulações que o erro é maior no centro do intervalo, portanto a liberdade de escolha de cada h_i pode ser usada para limitar o erro absoluto máximo.

Por exemplo, para $n = 2$ com $h_0 = 0.26222$, $h_1 = 0.32799$ e $h_2 = 0.58999$, o erro absoluto máximo é aproximadamente 0.0215.

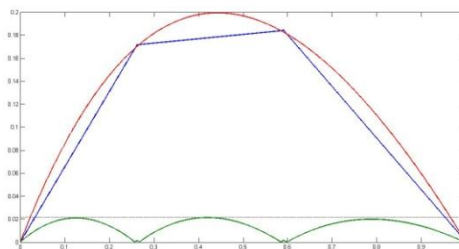


Figura 4. $y(x)$, $\Phi(x)$ e $E(x)$ para $n = 2$ e h_0, h_1 e h_2 .

Palavras-chave: *Método de Rayleigh-Ritz, Equações Diferenciais Ordinárias, Problemas de Valor de Contorno*

Referências

[1] A.W. Leissa, “The historical bases of the Rayleigh and Ritz methods”, *Journal of Sound and Vibration* 287. pp. 961 – 978, 2005.

[2] A. E. Assan, “Análise Numérica”, São Paulo, SP. Ed. Thomson. 2003.

[3] L. Burden, e D. Faires, “Numerical analysis”, 9º edition. Cengage Learning, pp. 673-679. 2011.

[4] J. Júdice, “Sistemas de Equações Lineares”, Coimbra, Portugal, Departamento de Matemática da Universidade de Coimbra, Capítulo 5, 1996

[5] M. Schultz, “Spline Analyses”, Prentice Hall, pp. 88-89, 1973.

[6] MATHWORKS, “User’s Guide - The Student Edition of Matlab, The Ultimate Computing Environment for Technical Education”, Prentice Hall, 1995.