COMPARAÇÃO DE MÉTODOS ITERATIVOS DE RESOLUÇÃO DE EQUAÇÕES NÃO LINEARES IMPLEMENTADOS NO OCTAVE.

Renan Wesney Domingos Elias, Renan Muniz da Silva, Jane Kelly Barbosa de Almeida, Dalton Cézane Gomes Valadares
Departamento de Engenharia Mecânica
Instituto Federal de Educação, Ciência e Tecnologia de Pernambuco (IFPE)
Caruaru, Pernambuco, Brasil
renanwesney@gmail.com, renanmuniz18@hotmail.com, jane13k@hotmail.com, dalton.valadares@caruaru.ifpe.edu.br

***Resumo* — Esse trabalho tem o objetivo de mensurar o desempenho de seis métodos numéricos iterativos de resolução de equações não lineares implementados no GNU Octave. Os métodos utilizados são: Bissecção, Falsa Posição, Falsa Posição Modificado, Ponto Fixo, Newton-Raphson e Secante. O GNU Octave foi utilizado por ser *software* livre e seguir o mesmo estilo de programação do MATLAB. Foram solucionadas diferentes equações não lineares, buscando, ao longo das análises, comparar a eficiência de cada método, no que concerne a número de iterações e tempo de processamento necessário para se atingir uma solução aceitável. No fim, é apresentada uma classificação dos métodos de acordo com as métricas utilizadas.**

Palavras-chave: Cálculo Numérico, Métodos Numéricos, Octave, Métodos Iterativos, Equações Não Lineares.

I. INTRODUÇÃO

O uso constante de modelagens matemáticas, na busca de solucionar problemas físicos ou uma melhor interpretação dos dados de pesquisa, tornam as ferramentas computacionais, nos dias de hoje, fundamentais. No caso de cálculos numéricos, uma ferramenta bastante conhecida e utilizada é o Matlab, que, por ser pago, torna sua aquisição difícil por grande parte das instituições de ensino, principalmente as públicas.

Torna-se então necessário identificar quais softwares não possuem custo para aquisição e que possam ser utilizados nesse trabalho. Baseando-se em um artigo da Universidade Federal de Pelotas intitulado *Avaliação de Softwares Matemáticos Quanto a Sua Funcionalidade e Tipo de Licença em Sala de Aula*[3], o GNU Octave foi escolhido como mais adequado a esse estudo.

Assim, esse trabalho tem o objetivo de mensurar o desempenho de alguns métodos iterativos para solução de equações não lineares implementados no Octave. Os métodos numéricos iterativos implementados foram: Bissecção, Falsa Posição, Falsa Posição Modificado, Ponto Fixo, Newton-Raphson e Secante. Todos estes permitem determinar raízes de funções não lineares. O estudo realizado, neste trabalho, buscou, ao longo das análises, comparar a eficiência de cada método na solução de problemas relacionados à engenharia, propondo assim apresentar uma classificação de cada método de acordo com o grau de eficiência nas respostas, no intuito de oferecer ao leitor, dados e resultados no uso destes métodos com esta ferramenta.

TRABALHOS RELACIONADOS

Ao pesquisar sobre trabalhos realizados, foram identificados artigos que demonstraram comparação similar, a partir de outros *softwares*, tendo como resultado na maioria deles, o método de Newton-Raphson o mais eficiente.

Um dos artigos fala sobre "Métodos numéricos para aproximação de raízes de funções"[4] que, ao fazer análise a partir da ferramenta SciLab, apresentou como método mais eficiente o de Newton e o da Secante.

Outra abordagem acontece em "Estudo comparativo da convergência de métodos numéricos iterativos aplicados na solução da equação de transferência de calor proposta por rosenthal sob a variação de parâmetros de soldagem"[5], em que o método Newton-Raphson é formalmente o menos dispendioso computacionalmente, a partir da ferramenta Wolfram Mathematica 9.0®.

Outro documento importante de citar é o de título "Equações algébricas: Aspectos históricos e um estudo sobre métodos algébricos, geométricos e computacionais de soluções"[6] , que explana e analisa melhor todos os métodos e aplicações, servindo também como base técnica para análise dos métodos

II. TEORIA

Na utilização de uma abordagem de iteração numérica para encontrar raízes reais de uma equação, pode-se obter seus resultados dividindo o processo em três principais etapas:

1) Definição inicial de um intervalo: para se fazer um refinamento inicial, é necessário se obter um intervalo no qual

pode-se afirmar que, dentro deste, existe uma raiz da equação, para que a partir daí o método aplicado possa convergir para o resultado aproximado. Este intervalo geralmente é definido pelo Teorema de Bolzano;

2) Precisão: é necessária a determinação da tolerância adotada para a aproximação do zero da equação localizado no intervalo definido;

3) O número de iterações: escolher um número máximo de iterações para evitar, por exemplo, que o programa entre em um *loop* infinito.

Para a análise do intervalo, deve-se verificar se a função é contínua no intervalo e se ela possui uma raiz no mesmo. Sendo assim, tem-se os seguintes critérios:

$f'(a)\ e\ f'(b)$ devem ter o mesmo sinal e, portanto:

$$f'(a) * f'(b) > 0$$

A precisão é determinada pelo usuário e não pode ser maior que a média do intervalo. O número de iterações é também definido pelo usuário, porém é interessante determinar um valor máximo evitando que o programa não entre em uma quantidade exorbitante de *loops*.

Definindo o intervalo, aplica-se o método desejado realizando as etapas necessárias para a sua execução. Tal procedimento pode se tornar comum a todos os métodos na obtenção da raiz com a precisão desejada, conforme fluxo ilustrado na Figura 1.

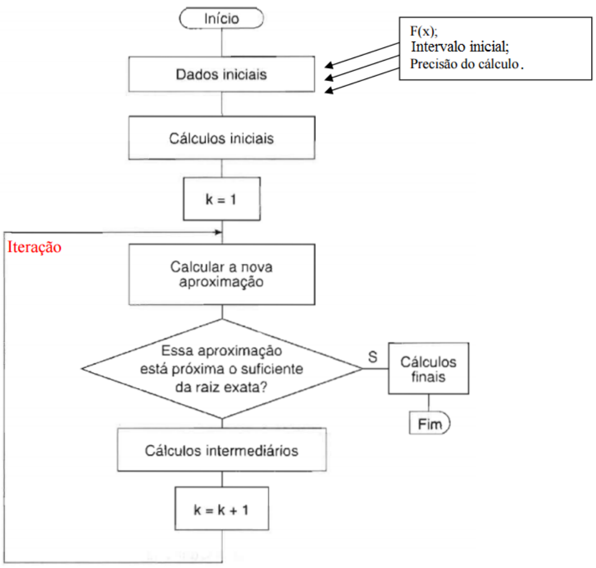

Figura 1 - Fluxograma da funcionalidade dos métodos numéricos [1].

Desta forma, faz-se necessário então definir-se separadamente cada método, conforme a próxima Seção.

III. METODOLOGIAS E MATERIAIS

Com o intuito de medir a eficiência de cada método, foi realizada uma comparação, utilizando uma linguagem computacional desenvolvida para a computação matemática, o GNU Octave, um *software* de licença livre, na versão 4.0.3.

O programa executou todas as programações e obteve os resultados em um notebook ASUS RBASQ550LF-BSI7T21, cuja especificação segue:

- Intel® Core™ i7-4500U, 4ª Geração, 1.8 GHz mobile processor ;
- 8 GB DDR3L 64bits de memória RAM;
- NVIDIA GT 745M placa gráfica com 2048 MB de memória de 1800 MHz e Core Speed de 837 MHz.
- Windows 10 Pro

Foram utilizados os seguintes métodos:

1. BISSECÇÃO

Este método busca reduzir um dado intervalo [a,b], que contém uma raiz da equação trabalhada, em valores o mais próximos possíveis da raiz da função, reduzindo a cada passo esse intervalo pela metade e verificando se a raiz se encontra nele, ou seja, dependendo da tolerância adotada. O código será executado até obter um resultado que satisfaça a tolerância: $(b-a)<\varepsilon$ (tolerância), conforme pode ser visto na Figura 2.

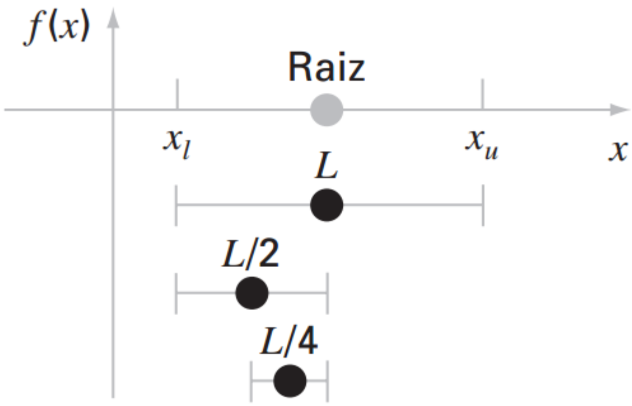

Figura 2 - Representação gráfica do método de Bissecção[2].

2. FALSA POSIÇÃO (FP)

É utilizado para resolver equações lineares definidas em um intervalo [a,b], utilizando a ideia de que há uma solução contida dentro deste espaço amostral e que se pode encontrá-la a partir de um refinamento deste espaço, dividindo esses intervalos em sub intervalos e assim dividindo estes em partes cada vez menores. Pode-se dizer que ele calcula a média ponderada dos limites do intervalo que contém a raiz [a,b] , conforme pode ser visto na Figura 3.

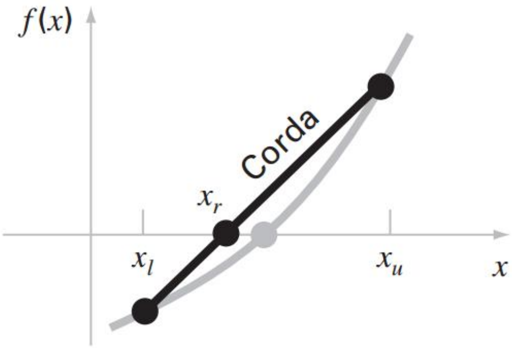

Figura 3 - Representação gráfica do método de Falsa posição [2].

3. FALSA POSIÇÃO MODIFICADA (FPM)

O método propõe dividir um intervalo em várias subdivisões sucessivas até alcançar o valor mais aproximado da raiz. Basicamente este método verifica algumas condições predefinidas como: $f(a)*f(b)<0$ e sinal da derivada constante.

Em seguida, calculando o valor de x_k , realiza-se vários testes para verificar o novo intervalo e o código se repete até obter resultados cada vez mais aproximados do valor da raiz da função.

$$x_k = \frac{[a*|f(b)|+b*|f(a)|}{|f(b)|+|f(a)|}$$

No caso, se $f(a)*f(b)<0$ for verdadeiro, faz-se $f(a)/2$ e, se for falso, faz-se $f(b)/2$.

4. PONTO FIXO

Um dos métodos que pode determinar a raiz de uma função é o ponto fixo. Para isso, é necessário, previamente, encontrar uma função de iteração a partir da função inicial, de forma que essa função de iteração satisfaça: $g(x_{k+1}) = g(x_k)$. Pode-se observar isso por meio da Figura 4.

$$f(x)=0 \qquad x = g(x)$$

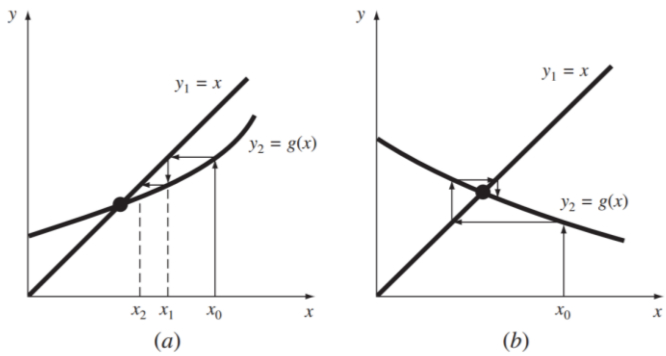

Figura 4 - Representação gráfica do método de Ponto fixo[2].

5. NEWTON-RAPHSON

Seja f(x) uma função contínua no intervalo [a.b], no qual existe a raiz da função e adotando uma tolerância δ. É possível determinar a raiz, pois f(ε) = 0 e f'(ε) ≠ 0, com isso, quando calcularmos a cada laço o valor de x, por meio da equação abaixo é possível se aproximar da raiz da função, até que o resultado satisfaça a tolerância desejada, conforme pode ser visto na Figura 5.

$$x_k = x_{k-1} - \frac{f(x_k)}{f'(x_k)}$$

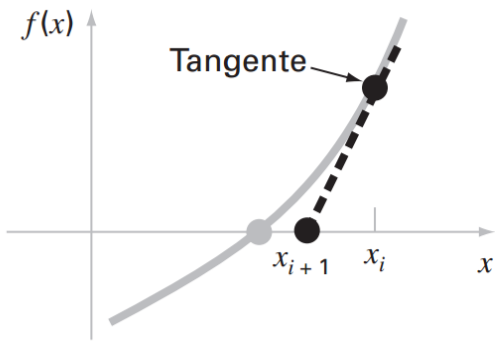

Figura 5 - Representação gráfica do método Newton-Raphson[2].

Para este trabalho, a $f'(x_k)$ foi calculada analiticamente e então inserida no código no formato da equação anterior.

6. SECANTE

Utiliza-se de uma sequência de linhas secantes à curva da função, ou seja, a partir de um intervalo inicial é possível traçar linhas secantes que tendem a chegar cada vez mais próximo da raiz da equação. A figura 6 ilustra graficamente como isso ocorre.

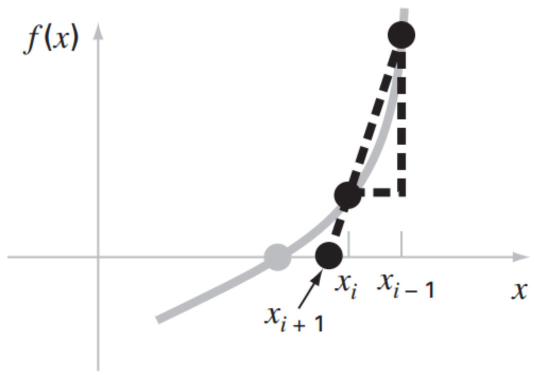

Figura 6 - Representação gráfica do método da Secante [2].

Estes métodos foram aplicados às equações listadas na Tabela 1, que foram escolhidas por representarem uma gama de situações, como o a necessidade de se trabalhar com funções logarítmicas, trigonométricas, algébricas e exponenciais.

Tabela 1: Apresentação das equações utilizadas para comparação dos métodos

Num	*Equações*
1	$x^3 - x - 1$
2	$x\log_{10}(x) - 1$
3	$sen(x) - x$
4	$x^3 + 4x^2 - 10$
5	$x^2 - 5x + 2$
6	$(x+1)^2 * e^{(x^2-2)} - 1$
7	$100 - 150x + 10x^2$

IV. RESULTADOS E DISCUSSÕES

Foram implementados códigos referentes à cada método no Octave e, assim, executados para todas as equações a fim de realizar a análise comparativa, restringindo um número de iterações de até 10^{25}, e uma tolerância de 10^{-9}. Os resultados são exibidos da Tabela 2 a 8. No final, uma análise referente à média desses valores foi realizada para se comparar a efetividade geral dos métodos, e definir qual o melhor.

Tabela 2: $y=x^3$ - x - 1

MÉTODO	INTERVALO	ITERAÇÕES (I)	TEMPO DE EXECUÇÃO [S]	RAÍZ
BISSECÇÃO	[0,2]	31	0.0190141	1.324717957
FP	[0,2]	29	0.0370259	1.324717957
FPM	[0,2]	10	0.00200391	1.324717957
PONTO FIXO	1	13	0.00400305	1.324717957
NEWTON	1	5	0.000999928	1.324717957
SECANTE	[0,2]	32	0.020014	1.324717957

Tabela 3: y=x*Log10(x)-1

MÉTODO	INTERVALO	ITERAÇÕES (I)	TEMPO DE EXECUÇÃO [S]	RAÍZ
BISSECÇÃO	[1,3]	31	0.0146091	2.506184145
FP	[1,3]	7	0.005476	2.506184145
FPM	[1,3]	5	0.00182509	2.506184144
PONTO FIXO	1	250	0.079061	2.506184144
NEWTON	1	5	0.00242901	2.506184146
SECANTE	[1,3]	5	0.00255084	2.506184146

Tabela 4: y=sin(x)-x

MÉTODO	INTERVALO	ITERAÇÕES (I)	TEMPO DE EXECUÇÃO [S]	RAÍZ
BISSECÇÃO	[-1,1]	1	0.000974178	0.000000000
FP	[-1,1]	1	0.000749111	0.000000000
FPM	[-1,1]	1	0	0.000000000
PONTO FIXO	1	42.163	11.9915	0.008434308
NEWTON	1	16	0.00661111	0.001476886
SECANTE	[-1,1]	1	0.00100183	0.000000000

Tabela 5: $y=x^3$ +4*x^2-10

MÉTODO	INTERVALO	ITERAÇÕES (I)	TEMPO DE EXECUÇÃO [S]	RAÍZ
BISSECÇÃO	[1,2]	30	0.0100079	1.365230014
FP	[1,2]	17	0.0183589	1.365230013
FPM	[1,2]	16	0.00441289	1.365230013
PONTO FIXO	1	34	0.0130329	1.365230013
NEWTON	1	4	0.002002	1.365230013
SECANTE	[1,2]	6	0.00626898	1.365230013

Tabela 6: $y=x^2$-5*x+2

MÉTODO	INTERVALO	ITERAÇÕES (I)	TEMPO DE EXECUÇÃO [S]	RAÍZ
BISSECÇÃO	[0,1]	30	0.013011	0.438447188
FP	[0,1]	10	0.0101821	0.438447187
FPM	[0,1]	9	0.00333881	0.438447187
PONTO FIXO	1	13	0.00400901	0.438447187
NEWTON	1	4	0.00202703	0.438447187
SECANTE	[0,1]	5	0.00299501	0.438447187

Tabela 7: $y=((x+1)^2)*exp^(x^2 -2) -1$

MÉTODO	INTERVALO	ITERAÇÕES (I)	TEMPO DE EXECUÇÃO [S]	RAÍZ
BISSECÇÃO	[0,1]	30	0.0118792	0.866873543
FP	[0,1]	14	0.0137079	0.866873543
FPM	[0,1]	10	0.00199986	0.866873543
PONTO FIXO	1	42	0.0160341	0.866873544
NEWTON	1	4	0.0026269	0.866873543
SECANTE	[0,1]	9	0.866873543	0.866873543

Tabela 8: $y=100-150*x+10*x^2$

MÉTODO	INTERVALO	ITERAÇÕES (I)	TEMPO DE EXECUÇÃO [S]	RAÍZ
BISSECÇÃO	[10,15]	33	0.0209639	14.300735255
FP	[10,15]	10	0.00601101	14.300735254
FPM	[10,15]	8	0.00200105	14.300735254
PONTO FIXO	10	43	0.0136011	14.300735254
NEWTON	10	6	0.00200105	14.300735254
SECANTE	[10,15]	6	0.00300407	14.300735254

Fazendo-se uma média entre todos estes valores, chega-se ao seguinte gráfico, visualizado na figura 2:

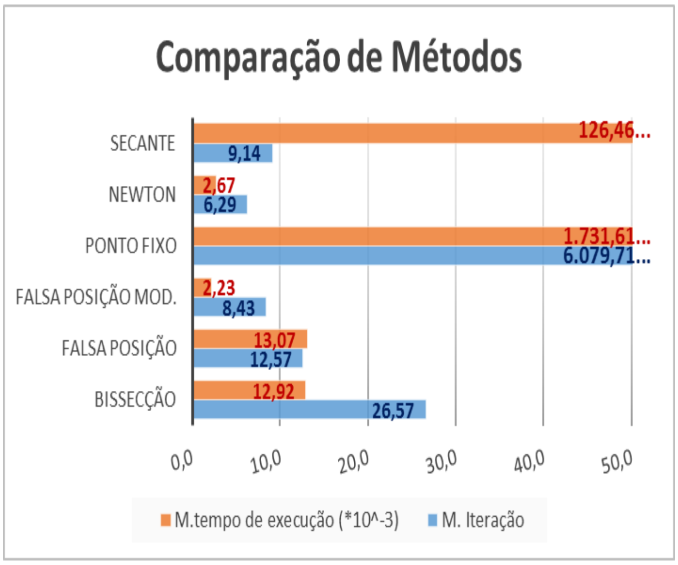

Figura 2- Gráfico Método x Tempo[cs] e iterações, *OBS:Foi retirada a iteração de ponto fixo da tabela 4, para melhor visualização

De acordo com o gráfico, percebe-se que o método mais eficiente é o de Newton-Raphson, porém, na implementação no Octave, é necessário ser feito uma derivação implícita, manualmente, em todas as equações, antes de implementar este método, o que, portanto, leva à conclusão que, para o Octave, a melhor opção é a falsa posição modificada, pois apresenta um baixo número de iterações, associado a baixa demanda de tempo.

CONCLUSÕES

Conclui-se que computacionalmente o método de Newton Raphson é o mais eficiente, porém, este exige mais esforço do operador da ferramenta, o que leva-se a eleger como melhor método, no OCTAVE, o da falsa posição modificada, pois, apresenta ótimo desenho, precisão, e boa eficiência computacional.

Devido ao satisfatório resultado desta pesquisa, pretende-se, no futuro, realização de comparações futuras com outros métodos existentes e outras ferramentas existentes no mercado.

AGRADECIMENTOS

- Dalton Valadares – Orientador – Pelo auxílio, incentivo e ajuda de forma geral;
- Instituto Federal De Pernambuco Campus Caruaru – Pelo conhecimento e o espaço cedido de forma geral;
- A toda turma de Cálculo Numérico de 2016.1
- A Deus por proporcionar tudo de forma geral.

REFERÊNCIAS

[1] CHAPRA, Steven C.; Canale. Raymond P. Métodos Numéricos para Engenharia, 5ª ed, McGraw Hill, 2008, ISBN: 978-85-86804-87-8..

[2] PILLING, Sérgio. II – Métodos numéricos para encontrar raízes (zeros) de funções reais - Cálculo Numérico, UNIVAP.

[3] SANTOS, R. Avaliação de softwares matemáticos quanto a sua funcionalidade e tipo de licença para uso em sala de aula. REnCiMa, v. 1, n. 1, p. 47-65, 2010.

[4] ALVEZ, A. N. S. et al. Estudo comparativo da convergência de métodos numéricos iterativos aplicados na solução da equação de transferência de calor proposta por rosenthal sob a variação de parâmetros de soldagem. CNMAI2014-0088. Novembro de 2014.

[5] RIBEIRO, R.R.J.; MENEZES, M. S.. Métodos numéricos para aproximação de raízes de funções. CMAC. ISSN 2317-3297. 2012.

[6] FERREIRA, G. S. S.. Equações algébricas: Aspectos históricos e um estudo sobre métodos algébricos, geométricos e computacionais de soluções. Fortaleza, 2014.