Anhanguera

Aqui o seu esforço ganha força.

Faculdade Anhanguera de Taubaté – Unidade II

Curso de Ciência da Computação e Engenharias

Disciplina: ***Métodos Numéricos***
Professor: ***Marcello Benevides***

Aula 01 & 02 – Introdução

Ementa:

- Erros
- Equações
- Interpolação
- Integração
- Autovalores
- Equações diferenciais
- Mínimos Quadrados

Livro texto para Métodos Numéricos:

E. KREYSZIG

Advanced Engineering Mathematics

Wiley

Avaliação:

- Listas de exercícios
- Provas
- Ava

Software para métodos numéricos

SCILAB

https://www.scilab.org/

https://www.avaeduc.com.br/pluginfile.php/3239402/mod_page/content/2/Introdu%C3%A7%C3%A3o%20ao%20Scilab.pdf

CAPÍTULO I - ERROS

1) Conversão de nºs binários em decimais:

$$N = (b_m\, b_{m\text{-}1}\ \ldots\ b_1\, b_0\,)_2 = (b_m 2^{m} + b_{m-1} 2^{m-1} + \ldots + b_1 2^{1} + b_0 2^{0})_{10}$$

Onde $b_i \in \{0,1\}\ \forall i=1,\ldots,m$

Ex$_1$: $(1001)_2 = (b_3\ b_2\ \ b_1\ b_0\,)_2 =$

$$= (1\times 2^3 + 0\times 2^2 + 0\times 2^1 + 1\times 2^0)_{10} =$$

$$= \ (8 + 0 + 0 + 1)_{10} =$$

$$= (9\,)_{10}$$

2) Conversão de nºs decimais em binários:

$$N=(d_n\ d_{n\text{-}1}\ \dots\ d_1\ d_0\ )_{10} = (b_m\ b_{m\text{-}1}\ \dots\ b_1\ b_0\ )_2 = \sum_{k=0}^{m} b_k 2^k$$

Onde m é a maior potência de 2 tal que $2^m \leq$ N

$$\text{Ex}_2\text{: } (47)_{10} = (b_5\ \ b_4\ \ b_3\ b_2\ \ b_1\ b_0\ )_2 = \sum_{k=0}^{5} b_k 2^k$$

$$= b_5 2^5 + b_4 2^4 + b_3 2^3 + b_2 2^2 + b_1 2^1 + b_0 2^0 =$$

$$= 32b_5 + 16b_4 + 8b_3 + 4b_2 + 2b_1 + b_0 =$$

$$= (\ 1\qquad 0\qquad 1\qquad 1\qquad 1\qquad 1\ )_2$$

3) Representação de nºs decimais fracionários:

$$f=(0.d_1\, d_2\ \ldots\, d_k \ldots)_{10} = d_1 10^{-1} + d_2 10^{-2} + \ldots + d_k 10^{-k} + \ldots$$

Onde $d_j \in \{0,1,\ldots 9\}$

Se existir m tal que $d_k=0\ \forall k > m \Rightarrow f$ tem representação decimal finita

Ex$_3$: $f = 1/8 = 0.125 = 1\times10^{-1} + 2\times10^{-2} + 5\times10^{-3}$ $\Rightarrow$ finita

Ex$_4$: $f = 1/9 = 0.111\ldots = 1\times10^{-1} + 1\times10^{-2} + 1\times10^{-3} + \ldots$ $\Rightarrow$ não finita

4) Conversão de nºs decimais fracionários em binários:

$f=(0.d_1\, d_2\ \dots\, d_k \dots)_{10} = (0.b_1\, b_2\ \dots\, b_k \dots)_2 = b_1 2^{-1} + b_2 2^{-2} + \dots + b_k 2^{-k} + \dots$

Onde $b_j \in \{0,1\}$

Como converter uma fração decimal em uma fração binária?

Parte inteira binária de 2*f*

$$f = \sum_{k=1}^{\infty} b_k 2^{-k} \quad \Rightarrow \quad 2f = b_1 + \sum_{k=1}^{\infty} b_{k+1} 2^{-k}$$

$\Rightarrow b_1 = 0$ ou 1

Parte frac. binária de 2*f*

$$(2f)_f = \sum_{k=1}^{\infty} b_{k+1} 2^{-k} \quad \Rightarrow \quad 2(2f)_f = b_2 + \sum_{k=1}^{\infty} b_{k+2} 2^{-k}$$

Seguindo esse raciocínio, obtemos b_3, ..., b_k, que são os dígitos que compõem a representação binária!

5) Aritmética de ponto flutuante

Seja *x* um número qualquer na base β em aritmética de ponto flutuante de *t* dígitos:

$$x = \pm(.d_1\ d_2\ \ldots\ d_t)_\beta \times \beta^e$$

Onde: (i) $\pm(.d_1\ d_2\ \ldots\ d_t)_\beta \times \beta^e$ é uma fração na base β

(ii) $d_j \in \{0,1,2,\ldots,\beta-1\}$

(iii) $e \in [m, M]$

(iv) *t* = número máximo de dígitos da mantissa

Um número não pode ser representado se o expoente “e” estiver fora dos limites m e M.

“Underflow” se e < m

“Overflow” se e > M

Números cuja representação em aritmética de ponto flutuante de *t* dígitos extrapolam os *t* dígitos da mantissa são armazenados por arredondamento ou por truncamento.

- truncagem: descartar todos os decimais a partir de um específico
- arredondamento:
  - para cima, descartado para > 5
  - para baixo, descartado para < 5

0,57 → 0,5
0,52 → 0,5

0,57 → 0,6

0,52 → 0,5

Ex_5: Seja um sistema de aritmética de ponto flutuante cuja mantissa tenha $t=3$ dígitos, base $\beta=10$, $m=-4$ e $M=4$.

x	Representação por arredondamento	Representação por truncamento
1.25	0.125×10	0.125×10
10.053	0.101×10^2	0.100×10^2
-238.15	-0.238×10^3	-0.238×10^3
2.71828	0.272×10^1	0.271×10^1
0.000000007	*Underflow*	*Underflow*
718235.82	*Underflow*	*Overflow*

Ex_6: Dados $x = 0.937 \times 10^4$ e $y = 0.127 \times 10^2$, calcule $x + y$ para um sistema em que $t=4$ e $\beta=10$.

$$x + y = 0.9370 \times 10^4 + 0.0013 \times 10^4 = 0.9383 \times 10^4$$

Estimativa de erros

Definição de erro:

$\varepsilon = a - \tilde{a}$, onde

$\tilde{a}$ = valor aproximado

a = valor verdadeiro (não conhecido)

erro relativo:

$$\varepsilon_r = \frac{\varepsilon}{a} = \frac{a - \tilde{a}}{a} \quad (a \neq 0)$$

$$\varepsilon_r \cong \frac{\varepsilon}{\tilde{a}} \qquad se \quad \varepsilon << \tilde{a}$$

Na prática, ε também não é conhecido. Assim, devemos definir um valor limite para o erro: $\beta \geq |\, \varepsilon \,|$

Tipos de erros:

operações (truncagens e arredondamento)

experimentais

Propagação de erros

Seja y uma função das variáveis $x_1, x_2, x_3, \ldots x_n$, ou seja,

$$y = f(x_1, x_2, x_3, \ldots x_n)$$

onde x_i é uma medida com um erro experimental Δx_i, ou seja

$$x_i = x_i \pm \Delta x_i$$

O erro Δy em y devido aos erros Δx_i das medidas de x_i pode ser obtido como:

$$\Delta y = \left|\frac{\partial y}{\partial x_1}\Delta x_1\right| + \left|\frac{\partial y}{\partial x_2}\Delta x_2\right| + \left|\frac{\partial y}{\partial x_3}\Delta x_3\right| + \ldots.$$

Ex$_7$: Para determinar o período de oscilação de um sistema massa-mola, um aluno mediu a constante elástica da mola e a massa do bloco, encontrando:

$$m = (100{,}36 \pm 0{,}03)\ \text{g}$$

e

$$k = (200{,}4 \pm 0{,}7)\text{x}10^2\ \text{N/m}$$

O período de oscilação do sistema é:

$$T = 2\pi\sqrt{\frac{m}{k}} = 1{,}406 \times 10^{-2}\ s$$

O erro ΔT *no período será dado por*

$$\Delta T = \left|\frac{\partial T}{\partial m}\Delta m\right| + \left|\frac{\partial T}{\partial k}\Delta k\right| = \frac{\pi}{\sqrt{mk}}\Delta m + \pi\sqrt{\frac{m}{k^3}}\Delta k$$

onde $\Delta m = 0{,}03\text{x}10^{-3}$ kg e $\Delta k = 0{,}7 \text{ x } 10^2$ N/m

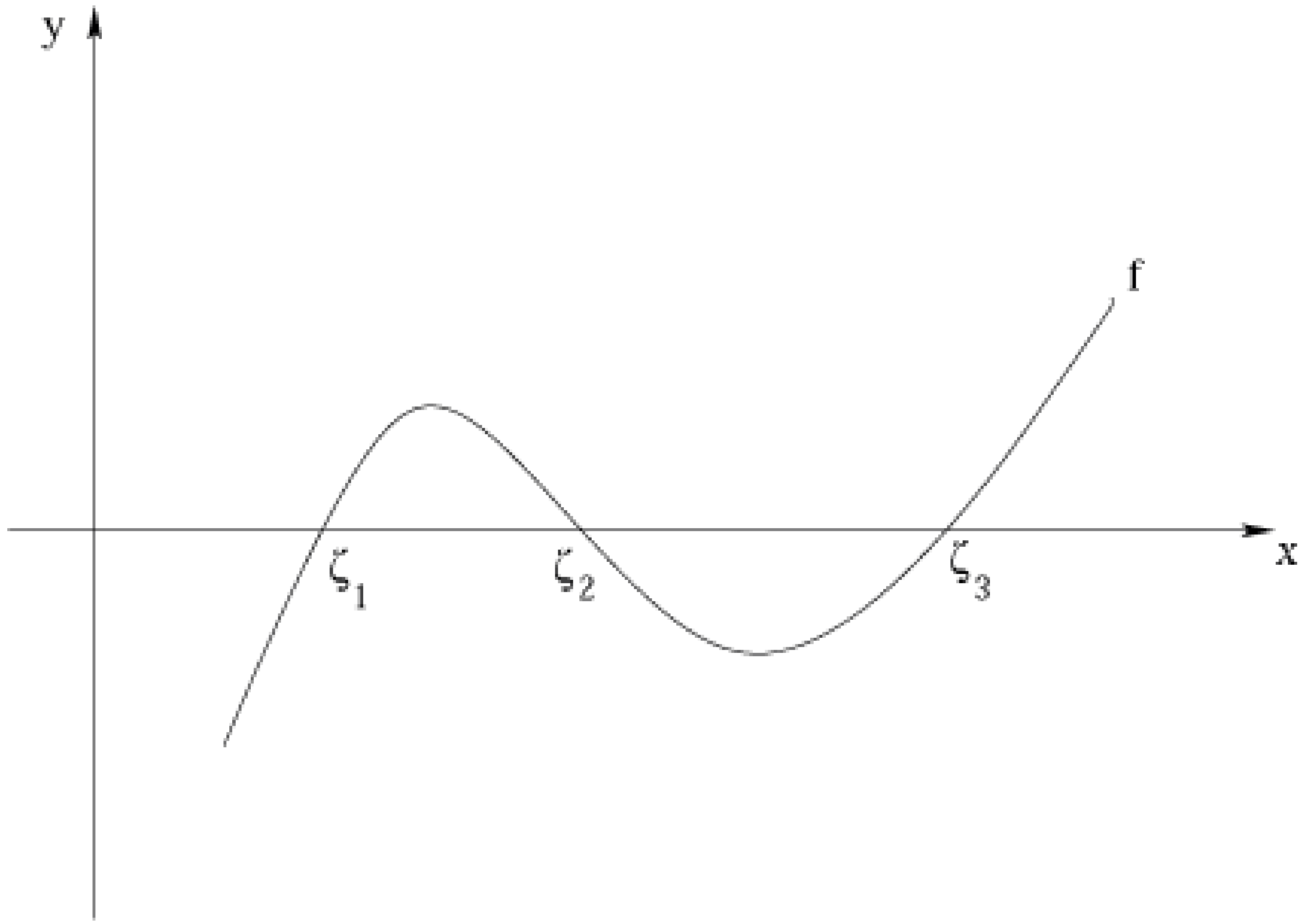
Substituindo esses valores na equação, obtém-se

$\Delta T = 2{,}66 \text{ x}10^{-5}$ s $= 0{,}00266 \text{ x } 10^{-2}$ s

$$T=(1{,}406 \pm 0{,}003) \text{ x } 10^{-2}\ \text{s}$$

CAPÍTULO II - EQUAÇÕES

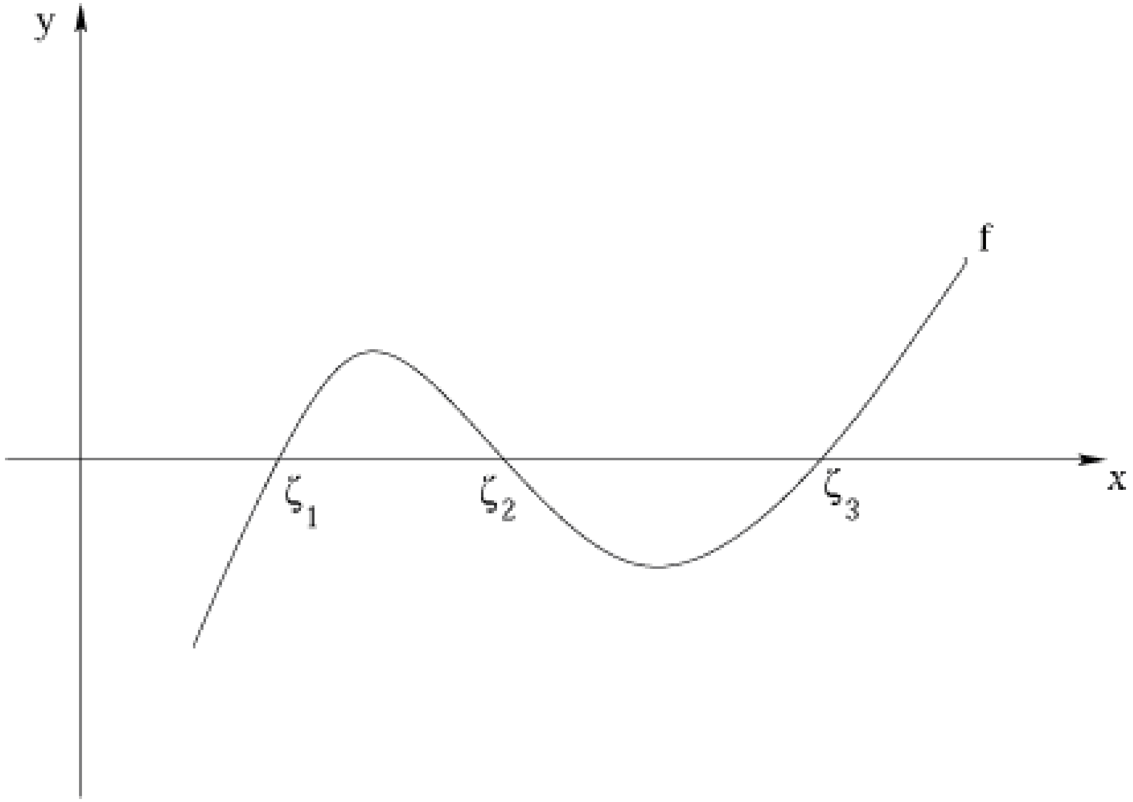
Objetivo: Resolver $f(x) = 0$, isto é, encontrar números ξ_i tais que $f(\xi_i)=0$



Fase I: isolar as raízes

Teorema de Cauchy-Bolzano: Seja f uma função contínua em [a,b]. Se $f(a)\times f(b) < 0$ então existe pelo menos uma raiz $\xi \in [a,b]$.

Teorema: Se f'preservar o sinal em [a,b] então a raiz ξ é única.



Processo I (Esboço do gráfico - varredura): Determinar um ponto inicial, um passo *h* e um ponto final de busca

Ex_8: Isolar a(s) raíz(es) positiva(s) de $f(x) = 2x - \cos(x) = 0$;

Façamos $a = 0$, $h=1$, $b = 10$

x	0	1	2	3	4	5	6	7	8	9	10
f(x)	-1	1.46	4.42	6.99	8.65	9.72	11.03	...	...	...	20.84

Conclusão: Há raiz $\xi \in [0,1]$.

Como $f'(x) = 2 + \text{sen}(x) > 0 \; \forall x \in [0,1]$ então ξ é única.

Processo II: Transformar $f(x) = 0$ em $g(x) – h(x) = 0$ e encontrar os pontos de interseção de g e h.

Ex_8: Isolar a(s) raíz(es) positiva(s) de $f(x) = 2x – \cos(x) = 0$;

Resp.: $\xi \in [0,\pi/2]$.

Ex_9: Isolar a(s) raíz(es) de $f(x) = x + \ln(x) = 0$;

Resp.: $\xi \in [? , ?]$.

Ex_{10}: Isolar a(s) raíz(es) de $f(x) = x\ln(x) - 1 = 0$;

Resp.: $\xi \in [?, ?]$.

Fase II: Refinar cada raiz

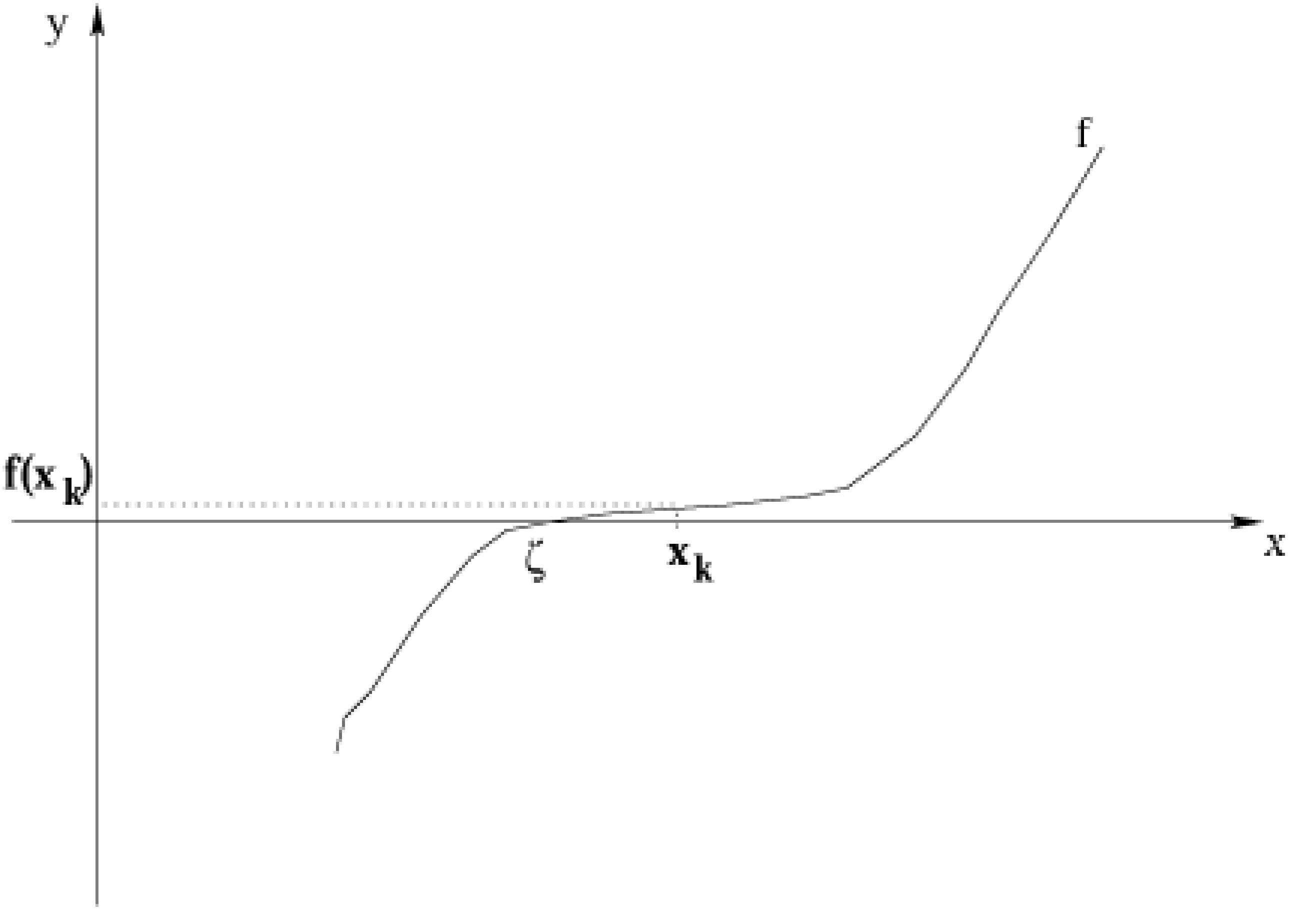
Diz-se que x_k é uma "boa" aproximação para a raiz ξ se:

(i) $|f(x_k)| < \varepsilon$

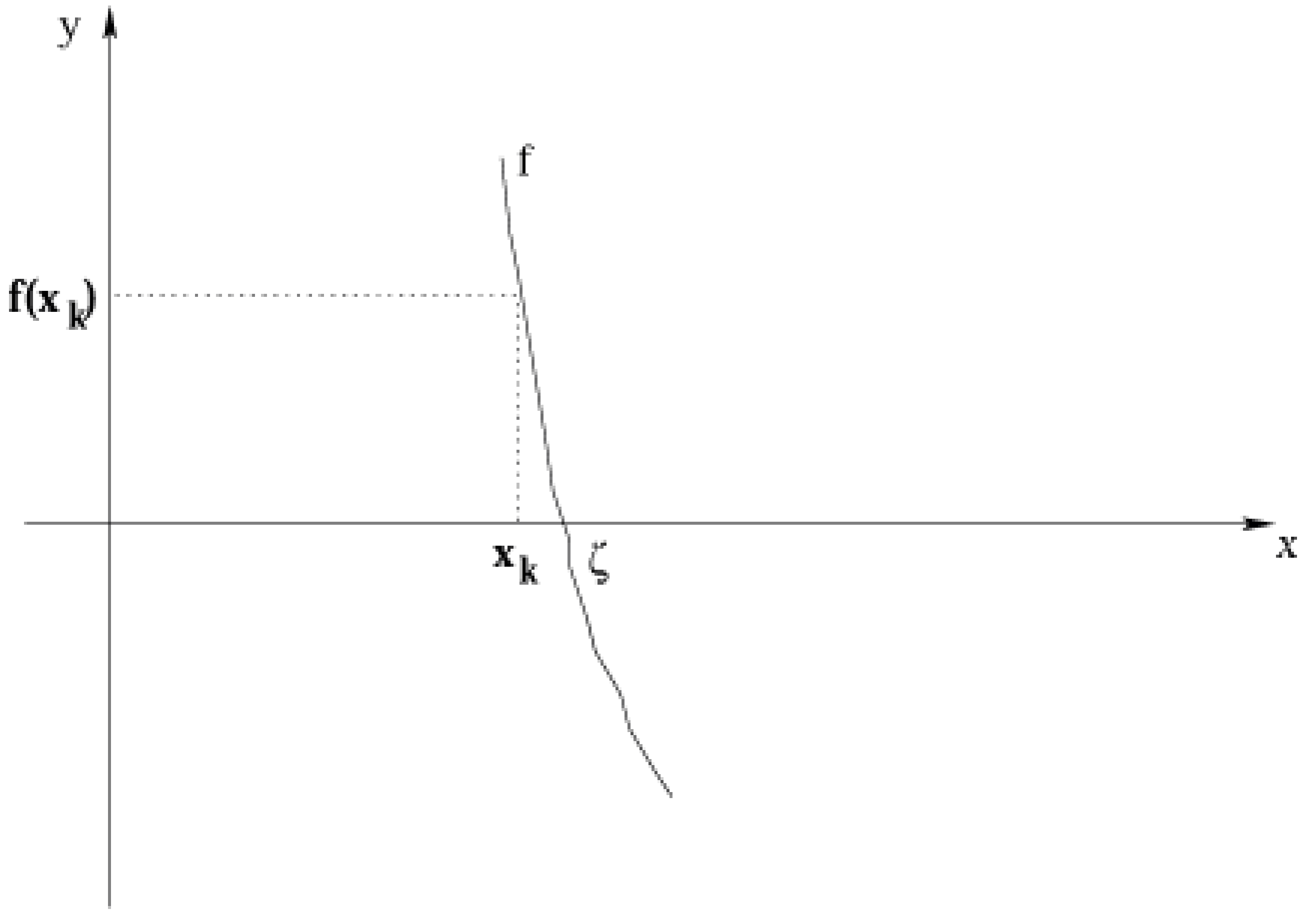
(ii) $|x_k - \xi| < \varepsilon$

Sendo ε a tolerância máxima admissível.

Estes dois critérios não são equivalentes!



$|f(x_k)| < \varepsilon$, mas $|x_k - \xi| >> \varepsilon$



$|x_k - \xi| < \varepsilon$, mas $|f(x_k)| >> \varepsilon$

Solução: Impor os dois critérios:

i) $|f(x_k)| < \varepsilon$

ii) $|x_k - \xi| < \varepsilon$

Como utilizar o segundo critério não se conhecendo ξ ?

Solução 1: Reduzir o intervalo [a,b] que contém a raiz até que sua amplitude seja menor que ε, isto é, que $b - a < \varepsilon$.

Se $b - a < \varepsilon \Rightarrow \forall x_k \in [a,b]$ tem-se: $|x_k - \xi| < b - a < \varepsilon$

Obs.: Como um método numérico pode não convergir é comum impor um número máximo de iterações como critério adicional de parada.

Solução 2: Aplicar o teorema:

TEOREMA: Sejam f e f'contínuas em [a,b]. Se f'preserva o sinal em [a,b] e se m=min|f'(x)| e M=max|f'(x)| para x $\in$[a,b], então:

$$|x_k - \xi| \leq ((M-m)/m)|x_k - x_{k-1}|$$

Além disso, se [a,b] é tão pequeno que M $\leq$ 2m então:

$$|x_k - \xi| \leq |x_k - x_{k-1}|$$

Conclusão: Para intervalo [a,b] suficientemente pequeno:

$$|x_k - x_{k-1}| < \in \quad \text{substitui} \quad |x_k - \xi| < \in$$

Método da Bisseção

Idéia: Reduzir o intervalo que contém a raiz, dividindo-o ao meio a cada iteração.

procedimento *Bissecao(a,b, ε, ITERMAX,x)*;

$k \leftarrow 0$;

$x \leftarrow (a+b)/2$;

enquanto $((b - a \geq \varepsilon$ ou $f(x) \geq \varepsilon)$ e $k \leq ITERMAX)$ faça

  se $(f(a) \times f(x) < 0)$

    então $b \leftarrow x$

    senão $a \leftarrow x$;

  $x \leftarrow (a+b)/2$;

  $k \leftarrow k+1$;

fim-enquanto;

se $(k \leq ITERMAX)$

  então Retorne x como aproximação para a raiz;

  senão Imprima: Não foi obtida uma aproximação com a precisão requerida em k iterações;

fim *Bissecao*;

Determinar com precisão $\varepsilon < 0,01$ e com um máximo de 10 iterações, a raiz da equação $f(x) = 2x - \cos x = 0$.

Solução:

(a) Isolamento da raiz:

Já foi visto que $\xi \in [0, 1]$.

(a) Refinamento da solução:

k	a	b	x_k	$f(x_k)$	$b-a$	Conclusão
0	0	1	0.500	0.122	1	$\xi \in [0.000, 0.500]$
1	0	0.500	0.250	-0.469	0.500	$\xi \in [0.250, 0.500]$
2	0.250	0.500	0.375	-0.181	0.250	$\xi \in [0.375, 0.500]$
3	0.375	0.500	0.438	-0.031	0.125	$\xi \in [0.438, 0.500]$
4	0.438	0.500	0.469	0.045	0.063	$\xi \in [0.438, 0.469]$
5	0.438	0.469	0.453	0.007	0.031	$\xi \in [0.438, 0.453]$
6	0.438	0.453	0.445	-0.012	0.016	$\xi \in [0.445, 0.453]$
7	0.445	0.453	0.449	-0.002	0.008	Pare! pois $b - a < \varepsilon$ e $\lvert f(x_k)\rvert < \varepsilon$

Na iteração 7, tanto a amplitude do intervalo $[a, b]$ quanto a imagem, em módulo, de x_7 são menores que a precisão requerida, isto é, $b - a = 0.453 - 0.445 = 0.008 < \varepsilon = 0.01$ e $\lvert f(x_7)\rvert = 0.008 < \varepsilon = 0.01$. Desta forma, dizemos que $x_7 = 0.449$ é uma aproximação para a raiz ξ da equação $f(x) = 2x - \cos x = 0$ com uma precisão $\varepsilon < 0.01$.

Método da Falsa Posição

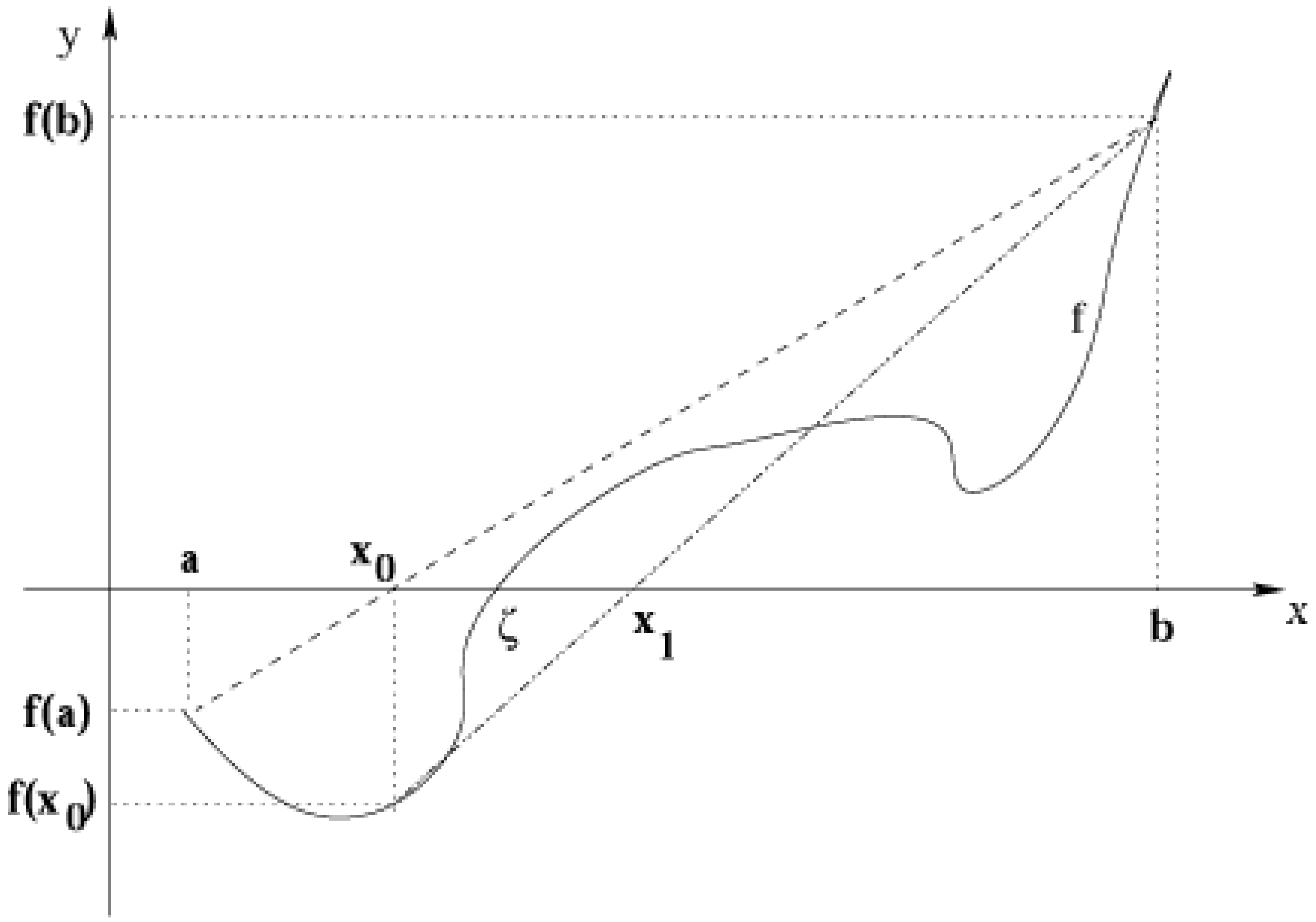
Idéia: Tomar como aproximação x para a raiz ξ a média ponderada dos extremos do intervalo [a,b] com pesos |f(b)| e |f(a)| respectivamente.

$$x = \frac{a|f(b)| + b|f(a)|}{|f(b)| + |f(a)|}$$

Desta forma, x estará mais próximo do extremo cuja imagem for menor.

Simplificação:

$$x = \frac{af(b) - bf(a)}{f(b) - f(a)}$$



y
f(b)
f
a
x_0
ζ
x_1
b
x
f(a)
$f(x_0)$

procedimento *FalsaPosicao(a,b, ε, ITERMAX,x)*;

$k \leftarrow 0$;
$x_{\text{ant}} \leftarrow a$;
$x \leftarrow \frac{a \times f(b) - b \times f(a)}{f(b) - f(a)}$;
enquanto $(|x - x_{\text{ant}}| \geq \varepsilon$ ou $|f(x)| \geq \varepsilon)$ e $k \leq ITERMAX$ faça
    se $(f(a) \times f(x) < 0)$
        então $b \leftarrow x$
        senão $a \leftarrow x$;
    $x_{\text{ant}} \leftarrow x$;
    $x \leftarrow \frac{a \times f(b) - b \times f(a)}{f(b) - f(a)}$;
    $k \leftarrow k + 1$;
fim-enquanto;
se $(k \leq ITERMAX)$
    então Retorne x como aproximação para a raiz;
    senão Imprima: Não foi obtida uma aproximação com a precisão
                requerida em k iterações;

fim *FalsaPosicao*;

Determinar, com erro $\varepsilon < 0,01$, a raiz da equação $f(x) = 2x - \cos x = 0$ no intervalo $[0, 1]$.

Solução:

Gerando as aproximações x_k segundo a expressão 5.6, obteremos a seguinte tabela:

k	a	b	x_k	$f(x_k)$	$\lvert x_k - x_{k-1}\rvert$	Conclusão
0	0	1	0.407	-0.105	0.407	$\xi \in [0.407, 1.000]$
1	0.407	1.000	0.447	-0.009	0.040	$\xi \in [0.447, 1.000]$
2	0.447	1.000	0.450	-0.001	0.003	Pare! pois $\lvert f(x_2)\rvert < 0.01$ e $\lvert x_2 - x_1\rvert < \varepsilon$

Logo, $x_2 = 0.450$ é uma aproximação para a raiz ξ da equação $f(x) = 2x - \cos x = 0$ com uma precisão $\varepsilon < 0.01$.

Método da Iteração Linear

- f(x) = 0 , solução é o número $x = s$ tal que f(s) = 0

- métodos iterativos:

    iniciar com um valor tentativo x_o, calcular iterativamente os valores x_1, x_2 ....

Algoritmos estáveis e instáveis

Ponto fixo:

transformar f(x) = 0 em x = g(x)

$x_o \rightarrow x_1 = g(x_o)$

$x_1 \rightarrow x_2 = g(x_1)$

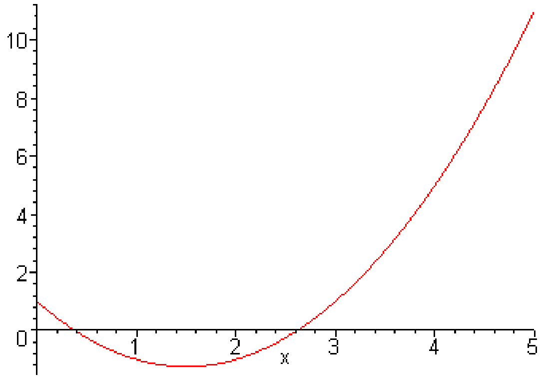
....

a solução da equação é o ponto fixo do processo $x_{n+1} = x_n = x^*$

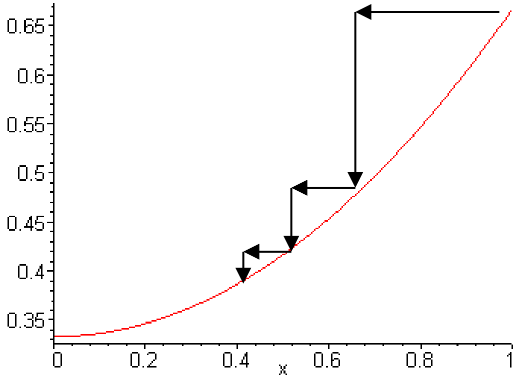
Exemplo:

$$x_{n+1} = (x_n^{\ 2} + 1)/3$$

$$f(x) = x^2 - 3x + 1 = 0$$



1	3
0,666667	3,333333
0,481481	4,037037
0,410608	5,765889
0,389533	11,41516
0,383912	43,76863
0,382463	638,8975
0,382093	136063,7
0,381998	6,17E+09

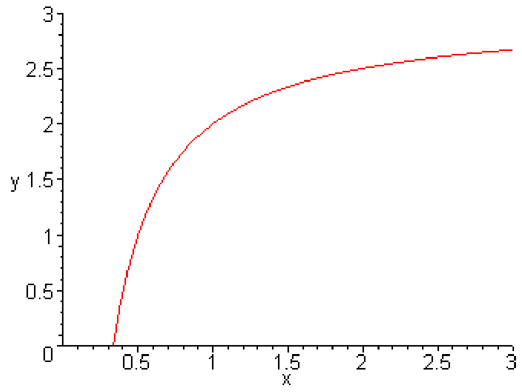


converge

diverge

1	3
2	2,666667
2,5	2,625
2,6	2,619048
2,615385	2,618182
2,617647	2,618056
2,617978	2,618037

$$x_{n+1} = (3 - \frac{1}{x_n})$$



Teorema: sendo x=s uma solução de x=g(x) e supondo que g(x) tem uma derivada contínua no intervalo J que contém s;

então, se | g´(x) | <= k < 1 em J, o processo iterativo definido por x_{n+1} = g(x_n) para qualquer x_o em J é ***convergente.***

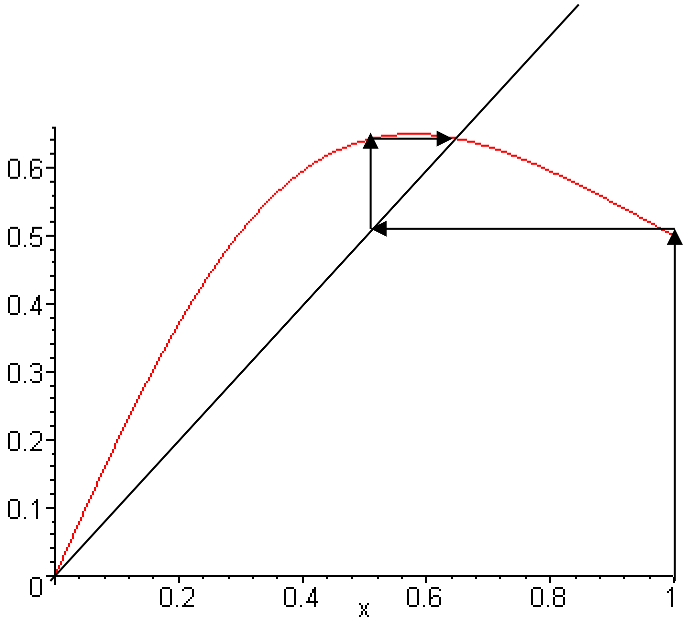
Ex.:

f(x) = $x^3 + x - 1$

$$x_{n+1} = g(x_n) = \frac{1}{1 - x_n^{\;2}}$$

$$|\dot{g}(x)| = \frac{2|x|}{(1+x^2)^2}$$

1
0,5
0,64
0,644196
0,643491
0,643612
0,643591

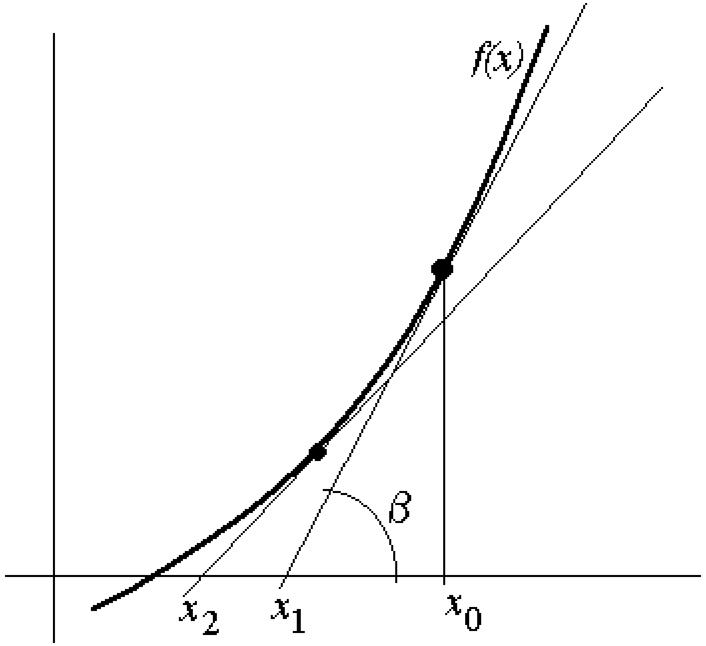


Método de Newton (Newton-Raphson)

f(x) tem uma derivada contínua f´(x)

$$\beta = \dot{f}(x_0) = \frac{f(x_0)}{x_0 - x_1}$$

$$x_1 = x_0 - \frac{f(x_0)}{\dot{f}(x_0)}$$



Exigências para convergência:

(i) f'e f'' devem preservar o sinal em [a,b] e não se anularem

(ii) x_0 deve ser tal que $f(x_0) \times f''(x_0) > 0$.

Algoritmo para calcular a solução de $f(x)=0$, sendo dada a aproximação inicial x_o . A função $f(x)$ é contínua, bem como sua derivada $f'(x)$. ε é a tolerância máxima e N é o número máximo de iterações

Dados: $f(x)$, $f'(x)$, x_o, ε, N

Algoritmo NEWTON

Entrada: $f(x)$, $f'(x)$, x_o, ε, N

Saída: solução aproximada x_n ($n \leq N$) ou mensagem de erro

For $n = 0, 1, 2, 3 \ldots\ldots N$

Calcule $f'(x_n)$

If $f''(x_n) = 0$, then "Mensagem de erro" (*procedimento malsucedido*)

Else: calcular $$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

If $\left| x_{n+1} - x_n \right| \leq \varepsilon$ then "Raiz é" x_{n+1} ; STOP

End

"Mensagem de erro" ; STOP (*não convergiu após N iterações*)

End NEWTON

Determinar pelo Método de Newton-Raphson, com precisão $\varepsilon < 0,01$ em um máximo de 10 iterações, a raiz da equação $f(x) = 2x - \cos x = 0$.

(a)Isolamento

Já foi visto que $\xi \in [0,1]$.

(b)Determinação de x_0:

$f'(x) = 2 + \operatorname{sen} x \implies f'(x) > 0 \;\; \forall\, x \in [0,1]$

$f''(x) = \cos x \implies f''(x) > 0 \;\; \forall\, x \in [0,1]$

Sendo $f(0) = -1$, $f(1) = 1.46$ e $f''(x) > 0$ então podemos tomar como aproximação inicial $x_0 = 1$, pois $f(1)f''(1) > 0$.

(c)Refinamento:

k	x_k	$f(x_k)$	$f'(x_k)$	$\lvert x_k - x_{k-1}\rvert$	Conclusão
0	1	1.460	2.841	-	
1	0.486	0.088	2.467	0.514	
2	0.450	0.001	2.435	0.036	
3	0.450	0.000	2.435	0.000	Pare! pois $\lvert f(x_3)\rvert < 0.01$ e $\lvert x_3 - x_2\rvert < \varepsilon$

Logo, $x_3 = 0.450$ é uma aproximação para a raiz ξ da equação $f(x) = 2x - \cos x = 0$ com uma precisão $\varepsilon < 0.01$.

kroton

paixão por educar

Anhanguera fama pitágoras GRUPO UNIASSELVI unic UNIVERSIDADE ANHANGUERA UNIDERP unime UNIRONDON unopar