

# Trabalho 02 - Equações Algébricas e Trascendentes

Arthur Vieira Silva

28 de Abril de 2025

# Sumário

<b>1</b>	<b>Introdução</b>	<b>3</b>
<b>2</b>	<b>Solução numérica dos problemas propostos</b>	<b>3</b>
2.1	Método da Bisseção . . . . .	3
2.2	Método de Newton-Raphson . . . . .	7
2.3	Método da Secante . . . . .	8
<b>3</b>	<b>Análise dos resultados</b>	<b>9</b>

# 1 Introdução

A determinação das raízes de funções é um problema recorrente em várias áreas, como engenharia, matemática, física, entre outras. Entretanto, para a maioria das funções, não é possível encontrar analiticamente uma raiz  $x$  que satisfaça  $f(x) = 0$ , com isso, é necessário aplicarmos métodos numéricos que aproximem tais raízes com uma determinada precisão. Duas funções serão utilizadas para verificar o funcionamento dos métodos implementados:

$$f(x) = 5 \times \ln(x) - 2 + 0.4x \quad (1)$$

$$f(x) = x - \frac{x^5 - 26}{5x^4} \quad (2)$$

Diante desse contexto, foram implementados, neste trabalho, três métodos numéricos utilizando a linguagem *Python*: o Método da Bissecção, o Método de Newton-Raphson e o Método da Secante. Além disso, como proposto no enunciado do problema, a precisão para o valor da variável independente  $x$  será  $1E-5$  e a precisão para o valor da função será  $1E-7$ . Por fim, para os *loops* dos algoritmos, também foi definido um valor máximo para o número de iterações igual a 50.

## 2 Solução numérica dos problemas propostos

### 2.1 Método da Bissecção

No primeiro método, temos que se uma função  $f$  for contínua em um determinado intervalo  $[a, b]$  e  $f(a) \times f(b) < 0$  (Teorema de Bolzano) então, há pelo menos uma raiz  $x^*$  no intervalo  $[a, b]$  na qual  $f(x^*) = 0$ . Dessa maneira, pode-se tomar uma aproximação para essa raiz dada pelo ponto médio do intervalo:  $x = \frac{a+b}{2}$ . Com isso, é possível determinar um novo intervalo  $[a, b]$  da seguinte forma:

- Se  $f(a) \times f(x) < 0$ : temos que  $x^* \in [a, x]$  e o valor de  $b$  receberá o valor de  $x$ ;
- Caso contrário, se  $f(x) \times f(b) < 0$ : temos que o  $x^* \in [x, b]$  e o valor de  $a$  receberá o valor de  $x$ .

O algoritmo funciona de maneira iterativa e, a cada iteração, um novo valor de  $x$  é calculado e um novo intervalo  $[a, b]$  também é obtido. A respeito das condições de parada, além do número máximo de iterações, também verifica-se a precisão da variável independente  $x$ :  $|\frac{b-a}{2}| < 1E-5$  ou a precisão de  $f(x)$ :  $|f(x)| < 1E-7$ . Assim sendo, o último valor médio obtido sobre o último intervalo  $[a, b]$  será a melhor aproximação encontrada para a raiz da função.

Sobre o intervalo inicial escolhido, a escolha foi feita observando-se o gráfico das duas funções. Primeiramente, nota-se, pela Figura 1, que a raiz da função está próxima do valor 1, sendo assim, o intervalo inicial  $[a, b]$  foi  $[1, 2]$ . Já para a segunda função, na Figura 2, é possível visualizar que a raiz da função está próxima do valor  $-2$  e, portanto, o intervalo inicial  $[a, b]$  foi  $[-2, -1]$ .

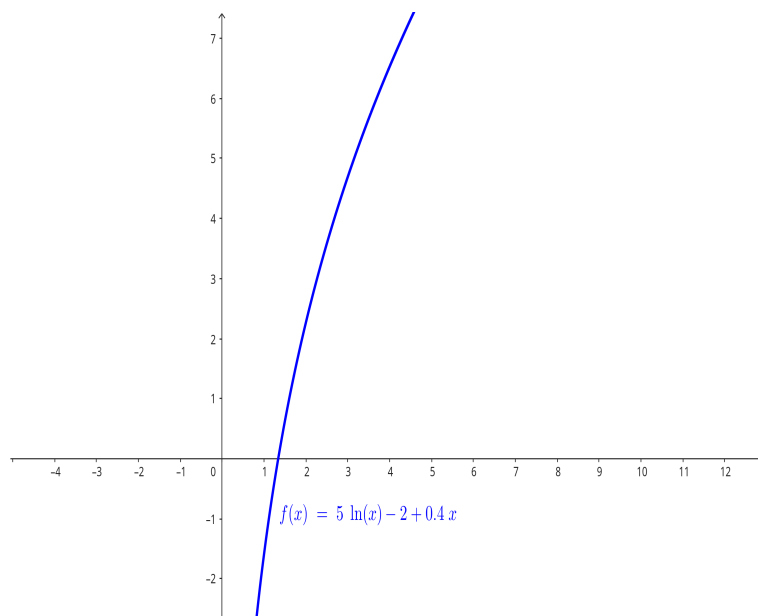


Figura 1: Gráfico da Função 1

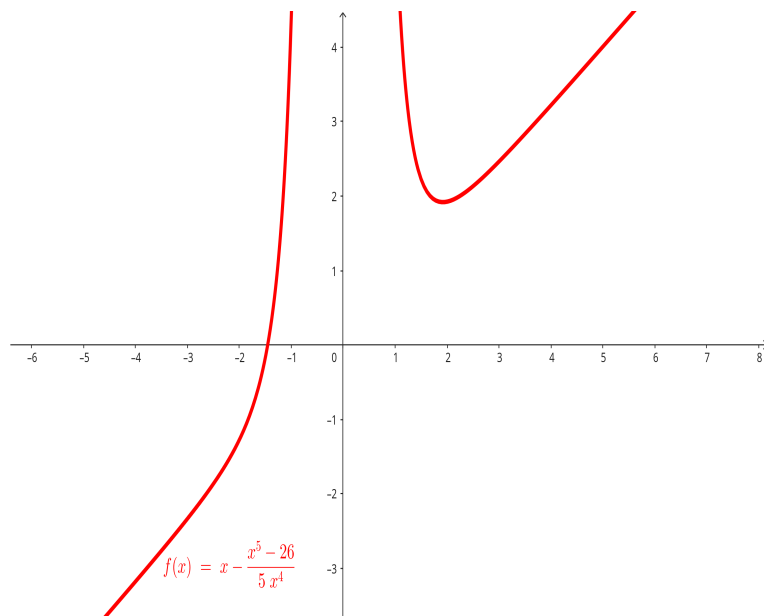


Figura 2: Gráfico da Função 2

Após a execução do algoritmo nos dois exemplos, a saída encontrada foi a seguinte:

Tabela 1: Saída do Método da Bisseção para a Função 1

Iteração	Intervalo [a, b]	x	f(x)
1	[1, 2]	1.5	0.6273255405408222
2	[1, 1.5]	1.25	-0.3842822434289512
3	[1.25, 1.5]	1.375	0.1422686555926729
4	[1.25, 1.375]	1.3125	-0.1153314225817911
5	[1.3125, 1.375]	1.34375	0.0148210644691794
6	[1.3125, 1.34375]	1.328125	-0.0499091343467768
7	[1.328125, 1.34375]	1.3359375	-0.0174585370847866
8	[1.3359375, 1.34375]	1.33984375	-0.0012974865681127
9	[1.33984375, 1.34375]	1.341796875	0.0067670859142078
10	[1.33984375, 1.341796875]	1.3408203125	0.0027361258425898
11	[1.33984375, 1.3408203125]	1.34033203125	0.0007196514211629
12	[1.33984375, 1.34033203125]	1.340087890625	-0.0002888345972724
13	[1.340087890625, 1.34033203125]	1.3402099609375	0.0002154291522170
14	[1.340087890625, 1.3402099609375]	1.34014892578125	-3.669753698754e-05
15	[1.34014892578125, 1.3402099609375]	1.340179443359375	8.936710394059e-05
16	[1.34014892578125, 1.340179443359375]	1.3401641845703125	2.633510756522e-05
17	[1.34014892578125, 1.3401641845703125]	1.3401565551757812	-5.181133687970e-06

Tabela 2: Saída do Método da Bissecção para a Função 2

Iteração	Intervalo [a, b]	x	f(x)
1	[-2, -1]	-1.5	-0.1728395061728396
2	[-1.5, -1]	-1.25	1.1299199999999998
3	[-1.5, -1.25]	-1.375	0.35476401885117137
4	[-1.5, -1.375]	-1.4375	0.06778867285351331
5	[-1.5, -1.4375]	-1.46875	-0.05759187434588453
6	[-1.46875, -1.4375]	-1.453125	0.003749432308857825
7	[-1.46875, -1.453125]	-1.4609375	-0.027247686337204202
8	[-1.4609375, -1.453125]	-1.45703125	-0.011832058823291414
9	[-1.45703125, -1.453125]	-1.455078125	-0.004062213353638988
10	[-1.455078125, -1.453125]	-1.4541015625	-0.0001616366115735346
11	[-1.4541015625, -1.453125]	-1.45361328125	0.0017925836823713404
12	[-1.4541015625, -1.45361328125]	-1.453857421875	0.0008151453248137486
13	[-1.4541015625, -1.453857421875]	-1.4539794921875	0.0003266723453043685
14	[-1.4541015625, -1.4539794921875]	-1.45404052734375	8.249736919974104e-05
15	[-1.4541015625, -1.45404052734375]	-1.454071044921875	-3.957474495774882e-05
16	[-1.454071044921875, -1.45404052734375]	-1.4540557861328125	2.146003109793071e-05
17	[-1.454071044921875, -1.4540557861328125]	-1.4540634155273438	-9.057677175849932e-06

## 2.2 Método de Newton-Raphson

Para o Método de Newton-Raphson, seja  $x^*$  o zero de uma função  $f(x)$  continuamente diferenciável, ou seja,  $f(x^*) = 0$ . Por meio da iteração do ponto fixo, nota-se  $x^*$  é um ponto fixo da função e  $g(x) = x + A(x) \times f(x)$ ,  $A(x) \neq 0$ . Com isso, em  $x = x^*$ , sabemos que  $f(x^*) = 0$  e, assim, temos que:  $g'(x^*) = 1 + A(x^*) \times f'(x^*)$ . O método iterativo possui uma convergência maior quanto menor for  $|g'(x)|$  próximo a  $x^*$ , dessa forma, escolhemos  $g'(x^*) = 0$  e obtemos  $A(x^*) = -\frac{1}{f'(x^*)}$ ,  $f'(x^*) \neq 0$ .

Dessa maneira, no Método de Newton-Raphson, o valor de  $x$ , em cada iteração, é obtido de acordo com a seguinte equação:

$$x^{(n+1)} = x^{(n)} + \Delta x, n \geq 1 \quad (3)$$

onde  $\Delta x = -\frac{f(x^{(n)})}{f'(x^{(n)})}$ .  $x^{(1)}$  é uma aproximação inicial escolhida.

Para as condições de parada para o algoritmo implementado, a cada iteração, verifica-se se o valor de  $|\Delta x| < 1E - 5$  ou se  $|f(x)| < 1E - 7$ ,

além de certificar-se de não ultrapassar o número máximo de iterações. Na escolha de um  $x^{(1)}$  inicial, foram levados em consideração dois fatores:

1.  $x^{(1)}$  estar próximo da raiz;
2.  $f'(x^{(1)}) \neq 0$ , para evitar divisões por 0.

Com isso, observando o Gráfico 1, é possível notar que a raiz da função está próxima de 1 e  $f'(1) = 5.4$  logo,  $x^{(1)} = 1$ . Em relação ao Gráfico 2, a raiz está próxima de -2 e  $f'(-2) = 1.45$  e, neste caso,  $x^{(1)} = -2$ .

Após a execução do algoritmo nos dois exemplos, a saída encontrada foi a seguinte:

Tabela 3: Saída do Método de Newton-Raphson para a Função 1

Iteração	$x^{(n)}$	$f(x^{(n)})$	$ \Delta x^{(n)} $
1	1	-1.6	0.2962962962962963
2	1.2962962962962963	-0.18392550405605845	0.04320397746283252
3	1.3395002737591288	-0.002716818613239136	0.0006573899353663645
4	1.3401576636944952	-6.019472472695853e-07	1.457180322960277e-07

Tabela 4: Saída do Método de Newton-Raphson para a Função 2

Iteração	$x^{(n)}$	$f(x^{(n)})$	$ \Delta x^{(n)} $
1	-2	-1.275	0.8793103448275862
2	-1.1206896551724137	2.4000151579740407	0.19098961021633712
3	-1.3116792653887508	0.7073368836515179	0.11488257837504383
4	-1.4265618437637946	0.11432051988013336	0.02645972454940838
5	-1.453021568313203	0.00416428567186089	0.001038096286977394
6	-1.4540596646001804	5.946015423585749e-06	1.486497772334187e-06

## 2.3 Método da Secante

Por fim, o Método da Secante é uma variação do Método de Newton-Raphson não necessitando do cálculo da derivada da função pois, ela pode ser difícil ou até inviável de se obter. Assim sendo, dada uma função  $f(x)$ , a aproximação da sua derivada será dada por  $f'(x) \approx \frac{f(x) - f(x_0)}{x - x_0}$ ,  $x \approx x_0$ .



Portanto, a iteração do Método da Secante é dada pela mesma Equação 3 descrita anteriormente, obtendo-se, a cada iteração, o valor  $x^{(n+1)}$ . Porém, temos que:  $\Delta x = -f(x^{(n)}) \times \frac{x^{(n)} - x^{(n-1)}}{f(x^{(n)}) - f(x^{(n-1)})}$ , com diferença de que  $n \geq 2$ . Além disso, os critérios de parada utilizados também são os mesmos do Método de Newton-Raphson.

Neste caso, note que, para inicializar as iterações, é necessário definir duas aproximações iniciais:  $x^{(1)}$  e  $x^{(2)}$ . Dessa maneira, os valores iniciais escolhidos foram próximos do valor da raiz. Analisando a Figura 1, como a raiz está próxima de 1, os valores escolhidos foram  $x^{(0)} = 1$  e  $x^{(1)} = 1.5$ . Já na Figura 2, a raiz da função está próxima de  $-2$  e, assim, os valores escolhidos foram  $x^{(0)} = -2$  e  $x^{(1)} = -1.5$ .

Após a execução do algoritmo nos dois exemplos, a saída encontrada foi a seguinte:

Tabela 5: Saída do Método da Secante para a Função 1

Iteração	$x^{(n-1)}$	$x^{(n)}$	$x^{(n+1)}$	$f(x^{(n+1)})$	$ \Delta x $
1	1	1.5	1.3591751566794992	0.07806012921884953	0.14082484332050083
2	1.5	1.3591751566794992	1.339161507102758	-0.004117011473191323	0.020013649576741115
3	1.3591751566794992	1.339161507102758	1.3401641755362663	2.6297788948914658e-05	0.0010026684335082282
4	1.339161507102758	1.3401641755362663	1.3401578115500945	8.830054243347263e-09	6.363986171893645e-06

Tabela 6: Saída do Método da Secante para a Função 2

Iteração	$x^{(n-1)}$	$x^{(n)}$	$x^{(n+1)}$	$f(x^{(n+1)})$	$ \Delta x $
1	-2	-1.5	-1.4215905908709043	0.13595260013570498	0.07840940912909555
2	-1.5	-1.4215905908709043	-1.456112079045066	-0.008180634514027396	0.03452148817416149
3	-1.4215905908709043	-1.456112079045066	-1.454152727488713	-0.00036625941702106424	0.0019593515563529266
4	-1.456112079045066	-1.454152727488713	-1.4540608927674639	1.0333344997448535e-06	9.183472124905717e-05
5	-1.454152727488713	-1.4540608927674639	-1.4540611511335366	-1.3015877264876963e-10	2.5836607269866824e-07

### 3 Análise dos resultados

Após a execução dos algoritmos, é possível afirmar que todos os métodos respeitaram os erros estabelecidos no enunciado do problema e também ficaram distantes de atingir o número máximo de iterações definido.

O Método da Bissecção possui uma convergência linear e, apesar do intervalo inicial ter sido escolhido corretamente, em ambos os casos, o método precisou de 17 iterações para encontrar um resultado adequado logo, o método converge mas é lento, justamente por reduzir o intervalo pela metade a cada iteração.

O Método de Newton-Raphson, diferentemente do Método da Bissecção, possui uma convergência quadrática, sendo muito mais rápido se a aproximação inicial for boa. Note que isso ocorreu, uma vez que o algoritmo levou 4 e 6 iterações, respectivamente, para solucionar os dois problemas.

Por fim, o Método da Secante foi o mais eficiente, uma vez que levou 4 iterações para o primeiro problema (igual ao Newton-Raphson) e 5 iterações para o segundo problema (menor número de iterações obtido). Além disso, possui a vantagem de não ser preciso realizar os cálculos das derivadas das funções.