

Sistemas Não-lineares

Pós-graduação em Modelagem Computacional

Universidade Federal de Juiz de Fora

Nome: Yago Pereira dos Anjos Santos
Matrícula: 130120017
Disciplina: Métodos Numéricos

1 Resolução de equações não-lineares

No processo de busca de raízes para as equações não-lineares a seguir, será utilizada uma tolerância pré-fixada $\varepsilon = 10^{-6}$. Empregaremos os três métodos para cada uma das quatro funções fornecidas pelo exercício. Para todos eles, o número máximo de iterações será pré-definido como $k_{max} = 100$. A implementação dos métodos e sua aplicação para cada uma das funções encontra-se em [Pereira (2025a)] ou em arquivo via anexo. O critério de parada adotado para o Método da Bissecção é dado por $|x_k - x_{k-1}| < \varepsilon$ e o critério de parada adotado para ambos os Métodos de Newton e da Secante são $|f(x_k)| < \varepsilon$ e $|x_k - x_{k-1}| < \varepsilon$.

1.1 Análise da função $f(x) = x^3 - 2x - 5$

Aqui consideramos a função $f(x) = x^3 - 2x - 5$. Inicialmente observamos que $f(0) = -5 < 0$ e $f(3) = (3)^3 - 2(3) - 5 = 16 > 0$. Além disso, a função f é contínua. Desta forma, vamos procurar por uma raiz no intervalo $[0, 3]$. Note também que $f'(x) = 3x^2 - 2$. Os resultados obtidos com a implementação dos métodos para esta função, encontram-se na Tabela 1.

Método	Raiz aproximada	Iterações	Erro
Bissecção	2.09455132484436	22	7.15256e-07
Newton	2.094551481542327	6	3.16014e-12
Secante	2.094551481483225	5	5.56951e-07

Tabela 1: Valores obtidos pelos métodos da Bissecção, de Newton e da Secante para a função $f(x) = x^3 - 2x - 5$.

Observe que para o método da bissecção, podemos desenvolver analiticamente o número máximo de iterações para que o método convirja. A fórmula é dada por:

$$k_{max} = \frac{\log(\frac{b-a}{\varepsilon})}{\log(2)} - 1 = \frac{\log(\frac{3-0}{10^{-6}})}{\log(2)} - 1 \approx 20.52.$$

Como k_{max} deve ser inteiro, temos $k_{max} = 21$. Observa-se pela tabela que o número de iterações para a convergência do método da bissecção é igual a 22 dentro da tolerância especificada. A diferença com o cálculo pela fórmula poderia ser explicada, por exemplo, por erros de arredondamento na aritmética de ponto flutuante. O Método da Bissecção tem uma ordem de convergência linear e, como podemos ver, é preciso um número maior de iterações para a convergência se comparado com os métodos de Newton e o da Secante. Além disso, para os métodos de Newton e da Secante foram usadas as aproximações iniciais $x_0 = 1.5$ e $x_1 = 2.25$ obtidos pelas duas iterações iniciais do Método da Bissecção. A ordem de convergência para o

Método de Newton é quadrática (significando que a convergência é rápida e o número de casas decimais corretas na aproximação dobra a cada iteração) enquanto a ordem de convergência para o Método da Secante é superlinear. No entanto, conforme verifica-se na Tabela 1, o Método da Secante precisou de 5 iterações para convergir, ao passo que o Método de Newton precisou de 6 iterações para a convergência, embora sua taxa de convergência seja quadrática. A razão para este fenômeno poderia ser o fato que o Método de Newton precisa não só da avaliação da função f em um ponto a cada iteração como também a avaliação da derivada f' em um ponto. Se for mais custoso avaliar a derivada em um ponto, o método de Newton pode se tornar ligeiramente mais lento. Além disso, a distância do chute inicial para a raiz pode também influenciar a convergência no Método de Newton. Por fim, destaca-se que o erro obtido para o Método de Newton é da ordem de 10^{-12} , ao passo que o erro para o Método da Secante é da ordem de 10^{-7} .

1.2 Análise da função $f(x) = e^{-x} - x$

O interesse aqui é encontrar $x \in \mathbb{R}$ tal que $e^{-x} = x$, isto é, deseja-se encontrar uma raiz para a função $f(x) = e^{-x} - x$. Neste caso, como $e^5 > 1$, segue que $e^{-5} < 1$, ou seja, $e^{-5} - 1 < 0$. Desta forma, temos $e^{-5} - 5 < -4 < 0$. Logo, $f(5) < 0$. Por outro lado, verifica-se que $f(-1) = e + 1 > 0$. Além disso, a função f é contínua com $f'(x) = -e^{-x} - 1$. Diante disto, o intervalo de busca da raiz para a função f será o intervalo $[-1, 5]$. Para a aplicação dos Métodos de Newton e da Secante, utilizamos as duas aproximações iniciais $x_0 = 2.0$ e $x_1 = 0.5$ obtidas a partir de duas iterações iniciais do Método da Bissecção. Para o Método da Bissecção, o número máximo de iterações para a convergência obtido pela fórmula analítica é:

$$k_{max} = \frac{\log(\frac{b-a}{\epsilon})}{\log(2)} - 1 = \frac{\log(\frac{5+1}{10^{-6}})}{\log(2)} - 1 \approx 21.52.$$

Como k_{max} deve ser inteiro, temos $k_{max} = 22$. Os resultados obtidos podem ser vistos na Tabela 2.

Método	Raiz aproximada	Iterações	Erro
Bissecção	0.5671432018280029	23	7.15256e-07
Newton	0.5671432904097838	5	3.01456e-11
Secante	0.5671432904230098	4	4.28666e-07

Tabela 2: Valores obtidos pelos métodos da Bissecção, de Newton e da Secante para a função $f(x) = e^{-x} - x$.

Novamente, pode-se observar pela tabela que o Método da Bissecção, por ter uma taxa de convergência linear, demora mais para convergir para a raiz procurada em comparação com os Métodos de Newton e da Secante. Na Bissecção temos 23 iterações, enquanto para os Métodos de Newton e da Secante temos 5 iterações e 4 iterações, respectivamente. Agora, destaca-se o fato que, embora o Método de Newton em geral tenha uma taxa de convergência quadrática, esta taxa não foi atingida para o presente caso. Isto porque o Método da Secante foi mais rápido do que o de Newton, mesmo tendo uma taxa de convergência superlinear, ligeiramente inferior à quadrática. Além disso, no Método de Newton temos o erro na ordem de 10^{-11} , enquanto no Método da Secante temos o erro na ordem de 10^{-7} .

1.3 Análise da função $f(x) = x \sin(x) - 1$

Aqui, queremos encontrar $x \in \mathbb{R}$ tal que $x \sin(x) = 1$, ou seja, queremos encontrar uma raiz para a função $f(x) = x \sin(x) - 1$. Observe que $f(0) = -1 < 0$. Por outro lado, temos

$f(\pi/2) = \pi/2 - 1$. Agora, note que $\pi > 2$, de modo que $\pi/2 > 1$, isto é, $f(\pi/2) > 0$. Além disso, f é uma função contínua com $f'(x) = \sin(x) + x \cos(x)$. Desta forma, a busca pela raiz será feita no intervalo $[0, \pi/2]$. Para o Método da Bissecção, o número máximo de iterações para a convergência obtido através de fórmula é:

$$\frac{\log(\frac{b-a}{\varepsilon})}{\log(2)} - 1 = \frac{\log(\frac{\pi/2-0}{10^{-6}})}{\log(2)} - 1 \approx 19.58.$$

Como k_{max} deve ser inteiro, temos $k_{max} = 20$. Para os Métodos de Newton e da Secante utilizamos as aproximações iniciais $x_0 = 0.7853981633974483$ e $x_1 = 1.1780972450961724$ obtidos a partir das duas primeiras iterações do Método da Bissecção. Os resultados obtidos são mostrados na Tabela 3.

Método	Raiz aproximada	Iterações	Erro
Bissecção	1.114157660155812	21	7.49014e-07
Newton	1.11415714087193	4	4.71787e-11
Secante	1.11415714087193	4	2.58117e-10

Tabela 3: Valores obtidos pelos métodos da Bissecção, de Newton e da Secante para a função $f(x) = x \sin(x) - 1$.

Pela tabela observa-se que o Método da Bissecção converge com uma taxa linear com 21 iterações. Os Métodos de Newton e da Secante convergem com o mesmo número de iterações. Sabemos que o Método da Secante possui uma taxa de convergência superlinear, de modo que o Método de Newton converge também com uma taxa superlinear com um erro da ordem de 10^{-11} . Já o Método da Secante converge com o erro na ordem de grandeza de 10^{-10} .

1.4 Análise da função $f(x) = x^3 - 3x^2 + 3x - 1$

Primeiramente, observe que $f(-1) = -8 < 0$ e $f(2) = 7 > 0$. Além disso, a função f é contínua com $f'(x) = 3x^2 - 6x + 3$. Desta forma, a busca pela raiz será feita no intervalo $[-1, 2]$. Para o Método da Bissecção, o número máximo de iterações para a convergência do método obtida por fórmula é:

$$k_{max} = \frac{\log(\frac{b-a}{\varepsilon})}{\log(2)} - 1 = \frac{\log(\frac{2+1}{10^{-6}})}{\log(2)} - 1 \approx 20.52.$$

Como k_{max} deve ser inteiro, então temos $k_{max} = 21$. Para os Métodos de Newton e da Secante, foram usadas as aproximações iniciais $x_0 = 0.5, x_1 = 1.25$, obtidas a partir das duas primeiras iterações do Método da Bissecção. Os resultados obtidos para este caso são mostrados na Tabela 4.

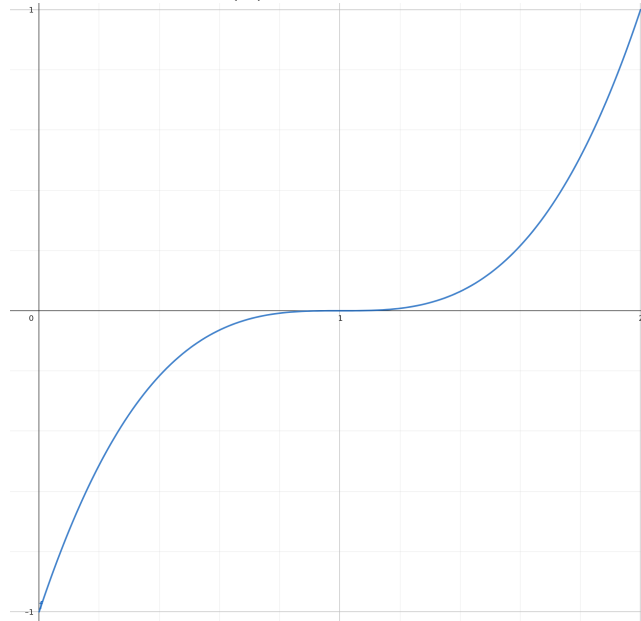
Método	Raiz aproximada	Iterações	Erro
Bissecção	1.000014543533325	22	7.15256e-07
Newton	0.9999992292331121	32	0.0
Secante	1.000000205887511	38	0.0

Tabela 4: Valores obtidos pelos métodos da Bissecção, de Newton e da Secante para a função $f(x) = x^3 - 3x^2 + 3x - 1$.

Neste caso, observa-se a partir da tabela que ocorre um fenômeno interessante. O Método da Bissecção, que possui uma taxa de convergência linear, foi mais rápido do que os Métodos

de Newton e da Secante obtendo um número de iterações igual a 22 para a convergência. Os Métodos de Newton e da Secante convergiram com 32 iterações e 38 iterações, respectivamente. Além disso, observa-se também que o Método de Newton convergiu com menos iterações que o Método da Secante. Neste caso, pode-se dizer que a convergência de ambos os Métodos, de Newton e da Secante, é linear. A razão para a ocorrência do fenômeno mencionado anteriormente, em vista dos resultados obtidos, encontra-se no fato que a raiz que está sendo procurada $\alpha = 1$ é mal condicionada. De fato, temos $f'(1) = 0$. Isto pode ser constatado também através do gráfico da função f na Figura 1 plotado com Geogebra. Desta forma, o critério de parada $|x_k - x_{k-1}| < \varepsilon$ mostra-se mais adequado para este caso.

Figura 1: Gráfico da função $f(x) = x^3 - 3x^2 + 3x - 1$ em tem-se $f'(1) = 0$.



2 Método do Ponto Fixo para Sistemas Não-lineares

O Sistema a ser resolvido numericamente é dado por:

$$\begin{cases} 16x^4 + 16y^4 + z^4 = 16 \\ x^2 + y^2 + z^2 = 3 \\ x^3 - y = 0. \end{cases}$$

A função de iteração a ser utilizada $g : \mathbb{R}^3 \rightarrow \mathbb{R}^3$ é definida por:

$$g(x, y, z) = \left(\sqrt[4]{1 - y^4 - \frac{z^4}{16}}, x^3, \sqrt{3 - x^2 - y^2} \right).$$

A partir desta função de iteração, o algoritmo do Método do Ponto Fixo foi implementado e o programa pode ser encontrado no repositório [Pereira (2025b)] ou no arquivo em anexo. Para a solução do sistema, a tolerância de $\varepsilon = 10^{-6}$ foi pré-fixada. Além disso, o número máximo de iterações foi definido como $k_{max} = 100$ e o critério de parada adotado foi definido por $\frac{|x_k - x_{k-1}|}{x_k} < \varepsilon$. Usando como aproximação inicial o vetor $[0.9, 0.7, 1.0]$, obteve-se a solução aproximada:

$$\mathbf{x} = [0.8779656209097886, 0.6767574591741983, 1.3308550237295367].$$

A solução aproximada foi obtida com 29 iterações e o erro associado da ordem de 10^{-7} . Agora, observamos que a matriz Jacobiana da função g é dada por:

$$J(g)(w) = \begin{pmatrix} 0 & \frac{-y^3}{\sqrt[4]{(1-y^4-\frac{z^4}{16})^3}} & \frac{-z^3}{\sqrt[4]{(1-y^4-\frac{z^4}{16})^3}} \\ 3x^2 & 0 & 0 \\ \frac{-x}{\sqrt{3-x^2-y^2}} & \frac{-y}{\sqrt{3-x^2-y^2}} & 0 \end{pmatrix},$$

onde $w = (x, y, z) \in \mathbb{R}^3$. Lembre que o raio espectral é o maior autovalor da matriz em valor absoluto. Avaliando a matriz Jacobiana de g no ponto \mathbf{x} da solução aproximada, obtemos o raio espectral $r = 0.9916051336451428 < 1$. Portanto, o método converge para a solução correta.

3 Método de Newton-Raphson com Backtracking Line Search

Para a solução do sistema e implementação do método, o ponto de partida é a definição do vetor de funções $F(x, y) = (f_1(x, y), f_2(x, y)) = (x^2 + y^2 - 4, e^x + y - 1)$ e da matriz Jacobiana do sistema:

$$J(F)(x, y) = \begin{pmatrix} 2x & 2y \\ e^x & 1 \end{pmatrix}.$$

Desta forma, a cada iteração, resolve-se o sistema $J(\mathbf{x}_k) p_k = -F(\mathbf{x}_k)$, sendo \mathbf{x}_k a aproximação no passo k , a fim de obter a direção de Newton. Em seguida, aplica-se *backtracking line search* para encontrar um passo α que satisfaça a condição de Armijo $\|F(\mathbf{x}_k + \alpha p_k)\| \leq (1 - c\alpha) \|F(\mathbf{x}_k)\|$ para $c = 10^{-4}$ e α inicial igual a 1, sendo reduzido multiplicativamente com o fator $\beta = 0.5$. Por fim, foi adotado o critério de parada $\|F(\mathbf{x}_k)\| < 10^{-6}$. A implementação do programa pode ser encontrada no repositório [Pereira (2025c)] ou em arquivo em anexo.

Em seguida, a implementação foi testada com o ponto inicial $x_0 = (1.0, 1.0)^T$ obtendo-se a solução aproximada para o sistema $\mathbf{x} = (-1.8162640706773958, 0.8373678007388251)$ com 7 iterações. A trajetória das iterações foi plotada no plano (x, y) conforme pode ser visto na Figura 2.

Os pontos da trajetória estão representados em azul, com destaque para o ponto inicial x_0 (representado na cor rosa) e para a solução aproximada para o sistema \mathbf{x} (representado na cor vermelha). Além disso, sobrepostas no gráfico, estão representadas as curvas de nível $f_1(x, y) = x^2 + y^2 - 4 = 0$ e $f_2(x, y) = e^x + y - 1 = 0$ nas cores verde e marrom, respectivamente. Note que o sistema em questão possui duas soluções, dadas pelas interseções das curvas de nível. Por fim, a implementação foi testada com o ponto inicial $x_0 = (65.0, 45.0)^T$ para análise da robustez do método e comparação do desempenho com o método de Newton sem o Backtracking Line Search. Os resultados obtidos são mostrados na Tabela 5.

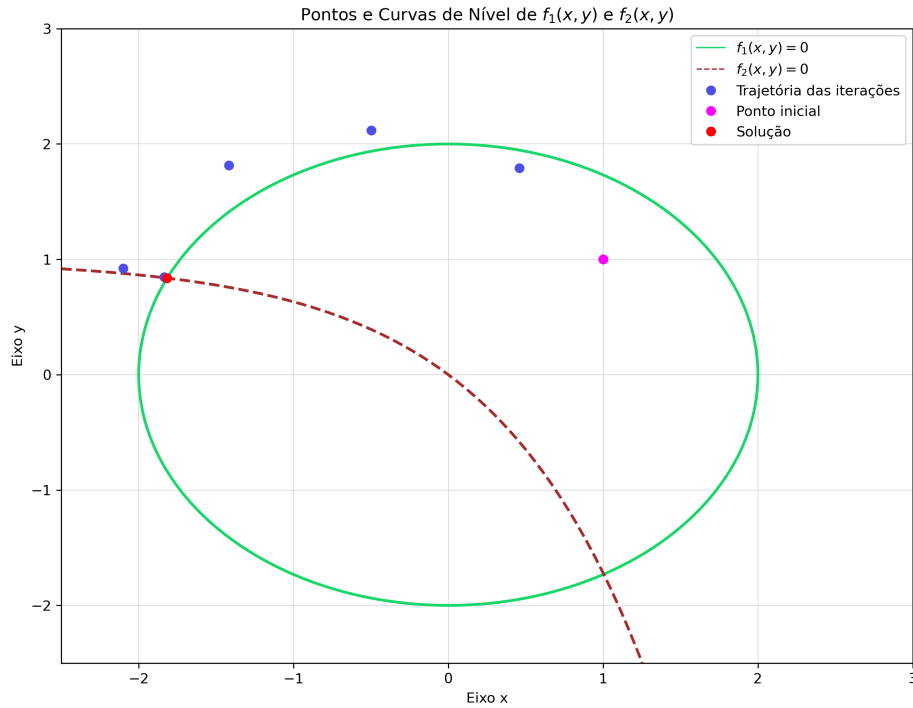
Método de Newton-Raphson	Solução aproximada	Iterações
Com Backtracking LS	(1.00416878, -1.72963729)	68
Sem Backtracking LS	(-1.81626407, 0.8373678)	70

Tabela 5: Comparação do Método de Newton-Raphson com busca linear e sem busca linear.

Conforme observa-se na tabela, com a aplicação do backtracking o método de Newton precisou de 68 iterações para convergir para a solução aproximada:

$$\mathbf{x} = (1.00416878, -1.72963729).$$

Figura 2: Trajetória das iterações a partir do ponto inicial $x_0 = (1.0, 1.0)^T$.



Por outro lado, o método de Newton sem a implementação do backtracking precisou de 70 iterações para convergir para a solução aproximada:

$$\mathbf{x} = (-1.81626407, 0.8373678).$$

Isto nos mostra que o método de Newton com backtracking line search é mais robusto podendo encontrar a solução do sistema com maior rapidez, dado que a cada iteração garante-se que a norma do resíduo é reduzida.

Referências

- Pereira, Y. 2025a, Repositório de Código para o Exercício Computacional 1, https://github.com/yapeansa/trabalho5_MN/blob/main/exe_1/equacoes_ponto_fixo.ipynb
- . 2025b, Repositório de Código para o Exercício Computacional 2, https://github.com/yapeansa/trabalho5_MN/blob/main/exe_2/sistema_ponto_fixo.ipynb
- . 2025c, Repositório de Código para o Exercício Computacional 3, https://github.com/yapeansa/trabalho5_MN/blob/main/exe_3/sistema_newton_btls.ipynb