Tarefa 2

Danilo Miranda de Medeiros Galvão

Setembro de 2020

1

Os cálculos foram feitos a partir de código em C e com critérios de parada pré estabelecidos. Todas as fórmulas foram retiradas da vídeo-aula sobre "Zeros de funções". Os códigos seguem em anexo e seus nomes correspondem a cada um dos cálculos. Abaixo são listados os resultados de cada método, onde os chutes iniciais, iterações feitas e raízes encontradas são todos respectivos às suas posições. Os critérios usados foram:
e = 0.01
Iterações máximas: 12

1.1 Bisseção

Intervalo: [-10,10]
Raízes encontradas: 4.8
Iterações feitas: 12

1.2 Ponto Fixo

Chutes iniciais: -1, 4.799805
Raízes encontradas: -1, 4.8
Iterações feitas: 1

1.3 Newton

Chutes iniciais: -5, 0.5, 4
Raízes encontradas: -2.1, -1, 4.8
Iterações feitas: 6, 4, 8

1.4 Secante

Chutes iniciais: (0,4), (0,5), (4,10)
Raízes encontradas: -2.1, -1, 4.8
Iterações feitas: 8, 5, 6

1.5 Falsa posição

Chutes iniciais: (-3,-1.5), (-2,0), (4,6)
Raízes encontradas: -2.1, -1, 4.8
Iterações feitas: 11, 10, 7

2

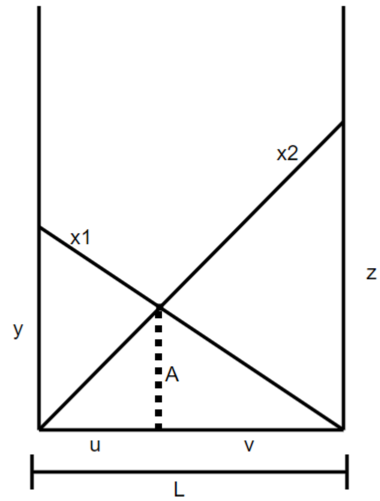
Ficou claro após os cálculos que alguns dos métodos funcionam melhor quanto mais próximo da raiz você tem seu chute inicial, como é visível na comparação entre o método da secante e o da falsa posição, onde pudemos encontrar as três raízes com um intervalo mais abrangente usando o método da secante enquanto com a falsa posição tivemos que estreitar o intervalo e, ainda assim, foram necessárias mais iterações. Além disso, o método da bisseção se provou ser mais difícil de se trabalhar por sua intuição ser recursiva, porém é um método iterativo, o que ocasionou em testes menos claros e resultados incompletos, além do problema limitante do caso onde dois pontos do intervalo tem sinal igual, mas existe uma raiz no meio. Algo similar ocorreu ao usar o método do ponto fixo, pois este método aposta muito ao calcular o valor da próxima iteração, assim só pude encontrar as raízes ao inserir como chute as próprias raízes. Por fim, o método de Newton conseguiu encontrar todas as raízes com uma velocidade considerável em relação aos outros métodos.

3

O cálculo foi feito por meio de código em C (anexado como q3.c).
A questão pede para encontrarmos o tempo exato em segundos que o objeto leva para chegar ao chão. Assim, podemos enxergar a queda como uma função s(t) que nos retorna a distância do chão para cada segundo que inserimos e que depende de vários outros fatores, que foram dados pela questão. Assim, precisamos apenas encontrar t tal que s(t) é 0. Usamos então o método da secante, por escolha, e utilizamos como valores iniciais 0 e 10, com o critério de parada sendo —s(t)— ¡ 0.00001. O método resultou em 5.818174 segundos.

4

Inicialmente, note que foi dado nome a outros componentes da configuração dada pela questão. Segue abaixo as nomenclaturas:



Assim, podemos começar a calcular L. Note que, por semelhança de triângulos, temos:

$$\frac{y}{L} = \frac{A}{v} \implies \frac{y \cdot v}{L} = A \implies y \cdot v = A \cdot L \implies v = \frac{A \cdot L}{y}$$
$$\frac{z}{L} = \frac{A}{u} \implies \frac{z \cdot u}{L} = A \implies z \cdot u = A \cdot L \implies u = \frac{A \cdot L}{z}$$

Temos, ainda, que L = v+u. Assim, podemos substituir:

$$L = \frac{A \cdot L}{y} + \frac{A \cdot L}{z} \implies \frac{1}{A} = \frac{1}{y} + \frac{1}{z} \implies \frac{1}{A} = \frac{1}{\sqrt{x_1^2 - L^2}} + \frac{1}{\sqrt{x_2^2 - L^2}}$$

Assim, podemos formar uma função s(L) tal que:

$$s(L) = \frac{1}{\sqrt{x_1^2 - L^2}} + \frac{1}{\sqrt{x_2^2 - L^2}} - \frac{1}{A}$$

Agora, basta encontrar uma raiz desta função e então obteremos o valor de L. Para isso, foi feito um código em C (anexado como q4.c) utilizando o método da bisseção no intervalo [10,20]. Assim, descobrimos que o valor de L é 16.212158 com precisão de 0.00001.