MS211 - Cálculo Numérico

Guilherme Nunes Trofino
217276

1 de abril de 2021

1 Questão

1.1 Modelagem

Teoria Sabe-se que a quantidade de poluentes a uma certa altura x do fundo de um corpo d'água de altura h pode ser descrito de acordo com a seguinte equação de *Problema de Valor de Contorno*:

$$\boxed{f(x) = -\alpha \cdot c''(x) + \nu \cdot c'(x) + \mu \cdot c(x)} \tag{1}$$

$$\boxed{c(0) = 0 \quad c'(H) = 0} \tag{2}$$

Onde:

1. α: Representa a **Difusibilidade** desse poluente na água;
2. ν: Representa a **Velocidade** e direção da densidade de partículas;
   (a) $\nu > 0$, deslocamento das partículas para cima;
   (b) $\nu < 0$, deslocamento das partículas para baixo;
3. μ: Representa o **Decaimento** desse poluente na água

Considera-se que poluentes vem apenas das chuvas e, portanto, sua entrada no sistema ocorre apenas pela superfície d'água. Este comportamento pode ser descrito como segue:

$$\boxed{f(x) = \begin{cases} k, & x = H \\ 0, & x \neq H \end{cases}} \tag{3}$$

Notações Adota-se as seguintes notações ao longo dos cálculos por simplicidade:

1. c_i para representar $c(x_i)$;
2. c'_i para representar $c'(x_i)$;
3. c''_i para representar $c''(x_i)$;

Em seguida utiliza-se as seguintes expressões para as derivadas de primeira e segunda ordem, respectivamente representadas, para modificar a primeira formulação do problema, apresentado em (1):

$$c'_i = \frac{c_{i+1} - c_{i-1}}{2\Delta x}$$

$$c''_i = \frac{c_{i-1} - 2c_i + c_{i+1}}{\Delta x^2}$$

$$f_i = -\alpha \cdot \left(\frac{c_{i-1} - 2c_i + c_{i+1}}{\Delta x^2}\right) + \nu \cdot \left(\frac{c_{i+1} - c_{i-1}}{2\Delta x}\right) + \mu \cdot c_1$$

Reformulação Reordenando as variavéis utilizadas na última equação tem-se a relação abaixo, base da solução numérica:

$$\boxed{f_i = \left(-\frac{\alpha}{\Delta x^2} - \frac{\nu}{2\Delta x}\right) c_{i-1} + \left(\frac{2\alpha}{\Delta x^2} + \mu\right) c_i + \left(-\frac{\alpha}{\Delta x^2} + \frac{\nu}{2\Delta x}\right) c_{i+1}} \tag{4}$$

Nota-se que a expressão será válida $\forall x_i$ e que $\Delta x = \frac{H}{n}$ será obtido dividindo a altura do corpo d'água pela quantidade de subintervalos a serem avaliados. Haverá portanto n incógnitas, $c_1, c_2, \ldots, c_n$ a serem avaliadas, pois como $c(0) = c(x_0)$ sabe-se que $c_0 = 0$ e, portanto, já conhecida.

Considerações A equação (4) estará definida para as $n-1$ primeiras linhas do sistema, resultando em vetor de variáveis $\vec{c} = [c_1, c_2, \dots, c_{n-1}, c_n]$. Analisando cada parcela dos termos das equações temos, considerando a condição (3):

1. Quando $i = 1$:

$$\cancelto{0}{f_1} = \left(-\frac{\alpha}{\Delta x^2} - \frac{\nu}{2\Delta x}\right)\cancelto{0}{c_0} + \left(\frac{2\alpha}{\Delta x^2} + \mu\right)c_1 + \left(-\frac{\alpha}{\Delta x^2} + \frac{\nu}{2\Delta x}\right)c_2$$

2. Quando $i = 2, 3, \dots, n-1$:

$$\cancelto{0}{f_i} = \left(-\frac{\alpha}{\Delta x^2} - \frac{\nu}{2\Delta x}\right)c_{i-1} + \left(\frac{2\alpha}{\Delta x^2} + \mu\right)c_i + \left(-\frac{\alpha}{\Delta x^2} + \frac{\nu}{2\Delta x}\right)c_{i+1}$$

3. Quando $i = n$:

$$f_n = \left(-\frac{\alpha}{\Delta x^2} - \frac{\nu}{2\Delta x}\right)c_{n-1} + \left(\frac{2\alpha}{\Delta x^2} + \mu\right)c_n + \left(-\frac{\alpha}{\Delta x^2} + \frac{\nu}{2\Delta x}\right)c_{n+1}$$

Nota-se, a princípio, que não existe c_{n+1}, entretanto, sabe-se pela condição de contorno, que $c'(H) = 0$ e, aplicando esta informação na aproximação utilizada para a derivada de primeira ordem, obtem-se:

$$c'_n = \frac{c_{n+1} - c_{n-1}}{2\Delta x} \rightarrow 0 = \frac{c_{n+1} - c_{n-1}}{2\Delta x} \rightarrow \boxed{c_{n+1} = c_{n-1}}$$

Assim:

1. Quando $i = n$:

$$f_n = \left(-\frac{2\alpha}{\Delta x^2} - \frac{\nu}{2\Delta x}\right)c_{n-1} + \left(\frac{2\alpha}{\Delta x^2} + \mu\right)c_n$$

Sistema Desta maneira pode-se representar, tomando as considerações apresentadas acima, um sistema linear, onde n representa a quantidade de subintervalos:

$$A = -\alpha - \nu\frac{\Delta x}{2} \qquad B = +2\alpha + \mu\Delta x^2 \qquad C = -\alpha + \nu\frac{\Delta x}{2} \qquad D = -2\alpha \qquad \Delta x = \frac{H}{n}$$

$$\boxed{\underbrace{\begin{bmatrix} B_1 & C_1 & & & & & \\ A_2 & B_2 & C_2 & & & & \\ & A_3 & B_3 & C_3 & & & \\ & & \ddots & \ddots & \ddots & & \\ & & & A_{n-2} & B_{n-2} & C_{n-2} & \\ & & & & A_{n-1} & B_{n-1} & C_{n-1} \\ & & & & & D_n & B_n \end{bmatrix}}_{n\times n} \times \underbrace{\begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ \vdots \\ c_{n-2} \\ c_{n-1} \\ c_n \end{bmatrix}}_{n\times 1} = \underbrace{\begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ f\Delta x^2 \end{bmatrix}}_{n\times 1}} \tag{5}$$

1.2 Desenvolvimento

Problema Notou-se que a qualidade da água na lagoa do parque Ábaco pode ser descrita de acordo com seguinte PVC:

$$\begin{cases} 0.025 & = -\underbrace{(1 - 0.09(5-x))}_{\alpha} \cdot c''(x) + \underbrace{-(0.2 - 0.01(5-x))}_{\nu<0} \cdot c'(x) + \underbrace{0.05}_{\mu} \cdot c(x) \\ c(0) & = 0 \\ c'(5) & = 0 \end{cases}$$

Onde $c(x)$ representa a quantidade de poluente presente na água a uma certa profundidade $x \in [0, 5]$ do lago. As condições de contorno para este problema são respectivamente:

1. Não há poluentes no fundo do lago: $c(0) = 0$;
2. Não há evaporação dos polentes: $c'(5) = 0$;

Observação Nota-se que, diferentemente do desenvolvimento apresentado, este problema possui um valor constante, $f(x) = 0.025$, assim o sistema elaborado será ligeiramente distinto.

1.3 Respostas

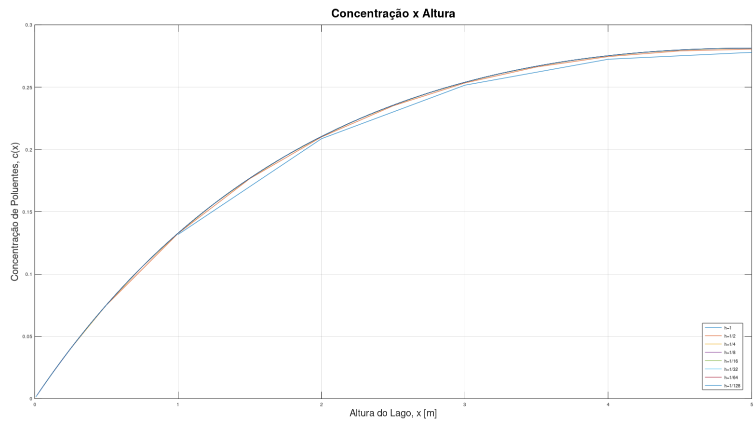
Questão 1. Sistema Geral de Equações obtido pelo Método de Diferenças Finitas:

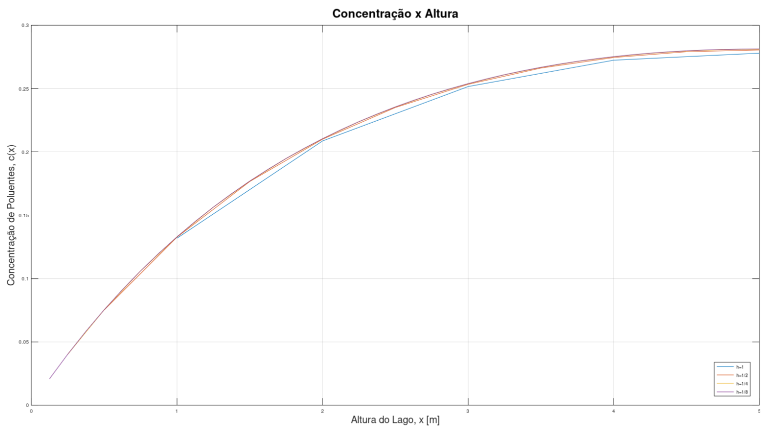
$$\underbrace{\begin{bmatrix} B_1 & C_1 & & & \\ A_2 & B_2 & C_2 & & \\ & \ddots & \ddots & \ddots & \\ & & A_{n-1} & B_{n-1} & C_{n-1} \\ & & & D_n & B_n \end{bmatrix}}_{n\times n} \times \underbrace{\begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_{n-1} \\ c_n \end{bmatrix}}_{n\times 1} = \underbrace{\begin{bmatrix} f\Delta x^2 \\ f\Delta x^2 \\ \vdots \\ f\Delta x^2 \\ f\Delta x^2 \end{bmatrix}}_{n\times 1}$$

Onde:

$$\begin{cases} A_i & = -1 + 0.09(5 - x_i) + (0.2 - 0.01(5 - x_i))\frac{\Delta x}{2} \\ B_i & = +2 - 0.18(5 - x_i) + 0.05\Delta x^2 \\ C_i & = -1 + 0.09(5 - x_i) - (0.2 - 0.01(5 - x_i))\frac{\Delta x}{2} \\ D_i & = +2 - 0.18(5 - x_i) \\ f & = 0.025 \end{cases}$$

Questão 2. Solução Numérica representada Graficamente:





Questão 3. Nota-se que a partir $h = \frac{1}{8}$ não ouve mais mudanças significativas na solução como pode ser observado comparando os gráficos. As condições iniciais do problema aparentam estarem satisfeitas, visto que $c(0) = 0$, partida da origem, e $c'(5) = 0$, curva aproximadamente horizontal.

1.4 Códigos

```
%============================================================================
function [X, C] = CriaSistemaLinear(A, V, U, f, H, dx)
    %========================================================================
    %Recebe Funcoes: A, V, U e f
    %Recebe Constantes: H e h
    %Retorna um Vetor X das Alturas e um Vetor C da Qntd. de Poluentes
    %========================================================================
    n = H/dx;               %Qtde Intervalos
    M = zeros(n, n);        %Inicializa a Matriz Base
    N = f*ones(n, 1)*dx^2; %Inicializa o Vetor Base

    X = linspace(dx, H, n); %Inicializa o Vetor de Intervalos

    for(i = 1: n)
        if(i != 1)
            M(i, i-1) = -A(X(i), H) -V(X(i), H)*dx/2; %Diagonal Inferior
        endif

        M(i, i) = 2*A(X(i), H) + U*dx^2; %Diagonal Principal

        if(i != n)
            M(i, i+1) = -A(X(i), H) +V(X(i), H)*dx/2; %Diagonal Superior
        endif
    endfor

    M(n, n-1) = -2*A(X(n), H); %Modifica a Entrada (n, n-1)
    C = M\N;                %Soluciona o Sistema Linear
endfunction

%============================================================================
%Declaracao das Funcoes e Variaveis
f = 0.025; %Funcao Constante
H = 5;     %Altura do Lago

A = @(x, H)  (1 - 0.09*(H - x));   %alpha
V = @(x, H) -(0.2 - 0.01*(H - x)); %nu
U = 0.05;                          %mu

%CriaSistemaLinear recebe alpha/nu/mu/f(x)/altura/intervalo
[X1, C1] = CriaSistemaLinear(A, V, U, f, H, 1    ); %Solucao h=1
[X2, C2] = CriaSistemaLinear(A, V, U, f, H, 1/2  ); %Solucao h=1/2
[X3, C3] = CriaSistemaLinear(A, V, U, f, H, 1/4  ); %Solucao h=1/4
[X4, C4] = CriaSistemaLinear(A, V, U, f, H, 1/8  ); %Solucao h=1/8
[X5, C5] = CriaSistemaLinear(A, V, U, f, H, 1/16 ); %Solucao h=1/16
[X6, C6] = CriaSistemaLinear(A, V, U, f, H, 1/32 ); %Solucao h=1/32
[X7, C7] = CriaSistemaLinear(A, V, U, f, H, 1/64 ); %Solucao h=1/64
[X8, C8] = CriaSistemaLinear(A, V, U, f, H, 1/128); %Solucao h=1/128


%Criacao do Grafico
LW = 1; FS = 20;
plot(X1,C1,'linewidth',LW,X2,C2,'linewidth',LW,
        X3,C3,'linewidth',LW,X4,C4,'linewidth',LW,
        X5,C5,'linewidth',LW,X6,C6,'linewidth',LW,
        X7,C7,'linewidth',LW,X8,C8,'linewidth',LW); grid;
title("Concentracao x Altura","fontsize",FS+4);
xlabel("Altura do Lago, x [m]","fontsize",FS);
ylabel("Concentracao de Poluentes, c(x)","fontsize",FS);
legend("h=1","h=1/2","h=1/4","h=1/8","h=1/16","h=1/32","h=1/64","h=1/128",
        "location", "southeast"); axis([0 H]);
```

2 Questão

2.1 Modelagem

Teoria Sabe-se que, a partir de uma tabela de dados, pode-se obter uma função φ que interpola, isto é, relacionara todos os pontos fornecidos como descrito:

x	x_1	$\dots$	x_n
y	y_1	$\dots$	y_n

$$\boxed{\varphi(x_k) = y_k, \quad \forall k = 1, \dots, n} \tag{6}$$

Há diferentes métodos para se estimar a Interpolação Linear para um conjunto de dados, dos quais são notáveis:

1. Forma de Vandermonde;
2. Forma de Lagrande;
3. Forma de Newton;

Utiliza-se a Forma de Newton por sua simplicidade e reduzido custo computacional de execução quando comparado com o Forma de Lagrange, por exemplo, que apesar de direta é computacionalmente custosa.

Forma de Newton Nesta abordagem o sistema linear resultante será triangular inferior descrito pelas seguintes funções base:

$$\begin{aligned} N_0 &= 1 \\ N_1 &= x - x_0 \\ N_2 &= (x - x_0)(x - x_1) \\ &\vdots \\ N_n &= (x - x_0)(x - x_1) \cdots (x - x_{n-2})(x - x_{n-1}) \end{aligned}$$

Consequentemente os coeficientes do polinômio interpolador φ podem ser obtidos solucionando o sistemas linear apresentado anteriormente:

$$\boxed{\varphi_n(x) = \alpha_0 + \alpha_1(x - x_0) + \cdots + \alpha_n(x - x_0)(x - x_1) \cdots (x - x_{n-1})}$$

Operador Diferenças Dividas Alternativamente os coeficientes do polinômio interpolador podem ser calculados utilizando **Diferenças Dividas** denotadas em sequência:

$$\boxed{\alpha_k = f[x_0, x_1, \dots, x_k], \quad \forall k = 0, 1, \dots, n}$$

Erro da Interpolação Nas considerações apresentadas o erro poderá ser estimado pela seguinte expressão:

$$\boxed{\epsilon_n(x) \approx \max_{x \in [x_0, x_n]} |f^{(n+1)}(x)| \prod_{k=0}^{n} |(x - x_k)|}$$

Fenômeno de Runge Nota-se que em interpolações realizadas em intervalos igualmente espaçadas há oscilação nas bordas de um intervalo em polinomios de grau elevado. Isso implica que a medida que o grau do polinômio interpolador aumenta cresce também o erro associado.

2.2 Desenvolvimento

Problema Notou-se que o registro do número de casos graves de COVID-19 ao longo de 2 meses na Sobolévia apresentou falha em 15/11 como mostrado:

Data	Casos
29/11	2899
22/11	3118
15/11	???
08/11	2821
01/11	2294
25/10	2261
18/10	2219
10/10	2347
04/10	2546

Afim de estimar, por meio da Interpolação Linear, o número de casos em 15/11 será necessário, primeiramente, modificar esta tabela para que sejam considerados apenas os dias das datas fornecidas, tomando 04/10 como dia 0. Em seguida, exclui-se o dado a ser calculado, modificando também a nomenclatura das colunas.

Data	Casos
29/11	2899
22/11	3118
15/11	???
08/11	2821
01/11	2294
25/10	2261
18/10	2219
10/10	2347
04/10	2546

$\rightarrow$

Dia	Casos
54	2899
47	3118
44	???
33	2821
28	2294
21	2261
14	2219
6	2347
0	2546

$\rightarrow$

X	Y
54	2899
47	3118
33	2821
28	2294
21	2261
14	2219
6	2347
0	2546

Observação Caso fosse considerado 04/10 como dia 4 os resultados seriam ligeiramente distintos e, portanto deve-se atentar a escolha de datas.

2.3 Respostas

Questão 1. Levando em consideração os conceitos empregados no **Fenômeno de Runge**, apesar do problema do problema não possuir intervalos igualmente espaçados, espera-se que quanto maior o grau do polinômio mais sucetível a erro a aproximação estará. Assim escolheu-se um Polinômio de Grau 4 para realização da interpolação, utilizando o seguinte conjunto de dados:

X	Y
54	2899
47	3118
33	2821
28	2294

Escolhendo este conjunto de dados para um polinômio interpolador estima-se aproximadamente 3161.4 casos graves de COVID-19 com erro relativo de 150.33 casos para o dia 15/11.

2.4 Código

```
%==================================================================================
function [T] = CoeficientesNewton (X, Y)
    %==============================================================================
    %Recebe Vetores: X e Y
    %Retorna um Vetor C com os Coeficientes de Newton do Conjunto
    %==============================================================================
    nPontos = length(X);          %Qtde de Dados
    T = zeros(nPontos, nPontos); %Matriz das Diferencas Dividas
    T(:,1) = Y;                   %Inicializa a Primeira Coluna

    for(j = 2:nPontos)
        for(i = 1:(nPontos-j+1))

            T(i,j) = (T(i+1,j-1) - T(i,j-1))/(X(j+i-1) - X(i)); %Calculo das Entradas

        endfor
    endfor
endfunction

%==================================================================================
function y = CalculoInterpolacao(A, X, T)
    %==============================================================================
    %Recebe o Valor A, o Vetor X e a Matrix T
    %Retorna um Valor y da Interpolacao
    %==============================================================================
  C = T(1,:); %Obtem os Coeficientes de Interpolacao
  y = C(1);   %Valor Base da Interpolacao

  for(i = 2:length(C))
    produto = C(i); %Inicializa o Produto

    for(j = 1:i-1)
      produto = produto*(A - X(j)); %Realiza o Produtorio
    endfor

    y = y + produto; %Realiza o Somatorio
  endfor
endfunction

%==================================================================================
function e = ErroInterpolacao(A, X, T)
    %==============================================================================
    %Recebe Vetores: X e Y
    %Retorna um Vetor C com os Coeficientes de Newton do Conjunto
    %==============================================================================
  e = 1; %Inicializa o Erro

  for(i = 1:length(X))
    e = abs(e*(A - X(i))); %Realiza o Produtorio
  endfor

  maxF = 0; %Inicializa o Maximo
  for(i = 1:(size(T)(1)))
    if(abs(T(i,(size(T)(2)))) > maxF)
      maxF = abs(T(i,(size(T)(2)))); %Modifica o Maximo
    endif
  endfor

  e = e*maxF; %Retorna o Erro da Interpolacao
endfunction
```

```matlab


%====================================================================================
%Declaracao das Colunas de Constantes
X = [0 6 14 21 28 33 47 54]';                    %Contagem em Dias
Y = [2546 2347 2219 2261 2294 2821 3118 2899]'; %Numero de Casos

T = CoeficientesNewton(X, Y); %Obtencao dos Coeficientes de Newton

S = 5;  %Start
E = 8;  %End
A = 44; %Dia 44

y = CalculoInterpolacao(A, X(S:E), T(S:end,1:E-S+1)) %Retorna Numero de Casos
E = ErroInterpolacao(A, X(S:E), T(:,1:E-S+2))        %Retorna Erro da Interpolacao
```