



UNIVERSIDADE DE BRASÍLIA

RELATÓRIO DE ATIVIDADE DO MÓDULO 4
MÉTODOS NUMÉRICOS PARA ENGENHARIA

Interpolação Polinomial

Aluno:
Wilton Rodrigues

Matrícula:
13/0049212

6 de novembro de 2016

1 Introdução

O objetivo deste relatório é exercitar os conceitos aprendidos em aula, com relação ao tópico: Interpolação Polinomial.

Interpolar uma função $f(x)$ consiste em aproximar essa função por uma outra função $g(x)$. A função $g(x)$ é então usada em substituição à função $f(x)$. A necessidade de se efetuar esta substituição surge em várias situações, como por exemplo, quando são conhecidos somente os valores numéricos da função para um conjunto de pontos e é necessário calcular o valor da função em um ponto não tabelado.

O problema a ser solucionado é o que trata da seguinte tabela de resultados para a densidade da água, ρ , em várias temperaturas, como é mostrado a seguir:

T	0	5	10	15	20	25	30	35	40
ρ	0.9999	0.9998	0.9997	0.9991	0.9982	0.9971	0.9957	0.9941	0.9902

Tabela 1: Valores tabelados

O objetivos deste relatório será, através de interpolação polinomial de 2º e 3º graus, obter os valores de $\rho(13)$ e $\rho(27)$. E por fim, apresentar um gráfico com os pontos e os polinômios interpoladores obtidos.

2 Metodologia

Baseando-se no método da Interpolação de Lagrange:

$$P(x) = \sum_{i=0}^n y_i \frac{L_i(x)}{L_i(x_i)} \quad (1)$$

Onde: $L_i(x) = (x - x_0) \dots (x - x_{i-1})(x - x_{i+1}) \dots (x - x_n)$.

Para n pontos, obtemos um polinômio de grau $n - 1$. Baseado nisso, a fim de encontrar o valor de $\rho(13)$, como um polinômio de 2º utilizaremos os 3 seguintes pontos:

T	5	10	15
ρ	0.9998	0.9997	0.9991

Tabela 2: Valores para $\rho(13)$

Aonde, ao aplicarmos a equação (1) obtemos o seguinte resultado:

$$P(x) = f(x_0) \cdot L_0(x) + f(x_1) \cdot L_1(x) + f(x_2) \cdot L_2(x) \quad (2)$$

Onde:

$$\begin{aligned} L_0(x) &= \frac{(x-x_1) \cdot (x-x_2)}{(x_0-x_1) \cdot (x_0-x_2)} = \frac{(x-10) \cdot (x-15)}{(5-10) \cdot (5-15)} = \frac{x^2-25x+150}{50} \\ L_1(x) &= \frac{(x-x_0) \cdot (x-x_2)}{(x_1-x_0) \cdot (x_1-x_2)} = \frac{(x-5) \cdot (x-15)}{(10-5) \cdot (10-15)} = \frac{x^2-20x+75}{-25} \\ L_2(x) &= \frac{(x-x_0) \cdot (x-x_1)}{(x_2-x_0) \cdot (x_2-x_1)} = \frac{(x-5) \cdot (x-10)}{(15-5) \cdot (15-10)} = \frac{x^2-15x+50}{50} \end{aligned}$$

Obtendo assim o primeiro polinômio interpolador:

$$P(x) = 0.9998 \cdot \left(\frac{x^2 - 25x + 150}{50} \right) + 0.9997 \cdot \left(\frac{x^2 - 20x + 75}{-25} \right) + 0.9991 \cdot \left(\frac{x^2 - 15x + 50}{50} \right) \quad (3)$$

E para o $\rho(27)$ como um polinômio de 3º utilizaremos estes 4 pontos:

T	15	20	25	30
ρ	0.9991	0.9982	0.9971	0.9957

Tabela 3: Valores para $\rho(27)$

Aplicando a equação (1) mais uma vez:

$$P(x) = f(x_0) \cdot L_0(x) + f(x_1) \cdot L_1(x) + f(x_2) \cdot L_2(x) + f(x_3) \cdot L_3(x) \quad (4)$$

Onde:

$$\begin{aligned} L_0(x) &= \frac{(x-x_1) \cdot (x-x_2) \cdot (x-x_3)}{(x_0-x_1) \cdot (x_0-x_2) \cdot (x_0-x_3)} = \frac{(x-20) \cdot (x-25) \cdot (x-30)}{(15-20) \cdot (15-25) \cdot (15-30)} = \frac{x^3-75x^2+1850x-15000}{-750} \\ L_1(x) &= \frac{(x-x_0) \cdot (x-x_2) \cdot (x-x_3)}{(x_1-x_0) \cdot (x_1-x_2) \cdot (x_1-x_3)} = \frac{(x-15) \cdot (x-25) \cdot (x-30)}{(20-15) \cdot (20-25) \cdot (20-30)} = \frac{x^3-70x^2+1575x-11250}{250} \\ L_2(x) &= \frac{(x-x_0) \cdot (x-x_1) \cdot (x-x_3)}{(x_2-x_0) \cdot (x_2-x_1) \cdot (x_2-x_3)} = \frac{(x-15) \cdot (x-20) \cdot (x-30)}{(25-15) \cdot (25-20) \cdot (25-30)} = \frac{x^3-65x^2+1350x-9000}{-250} \\ L_3(x) &= \frac{(x-x_0) \cdot (x-x_1) \cdot (x-x_2)}{(x_3-x_0) \cdot (x_3-x_1) \cdot (x_3-x_2)} = \frac{(x-15) \cdot (x-20) \cdot (x-25)}{(30-15) \cdot (30-20) \cdot (30-25)} = \frac{x^3-60x^2+1175x-7500}{750} \end{aligned}$$

Obtendo assim o segundo polinômio interpolador:

$$\begin{aligned} P(x) &= 0.9991 \cdot \left(\frac{x^3 - 75x^2 + 1850x - 15000}{-750} \right) + 0.9982 \cdot \left(\frac{x^3 - 70x^2 + 1575x - 11250}{250} \right) + \\ &+ 0.9971 \cdot \left(\frac{x^3 - 65x^2 + 1350x - 9000}{-250} \right) + 0.9957 \cdot \left(\frac{x^3 - 60x^2 + 1175x - 7500}{750} \right) \quad (5) \end{aligned}$$

3 Diagrama esquemático de execução

Nesta seção, encontra-se o fluxo de execução do sistema proposto na equação (??) utilizando a linguagem C. Que é apresentada na próxima sessão.

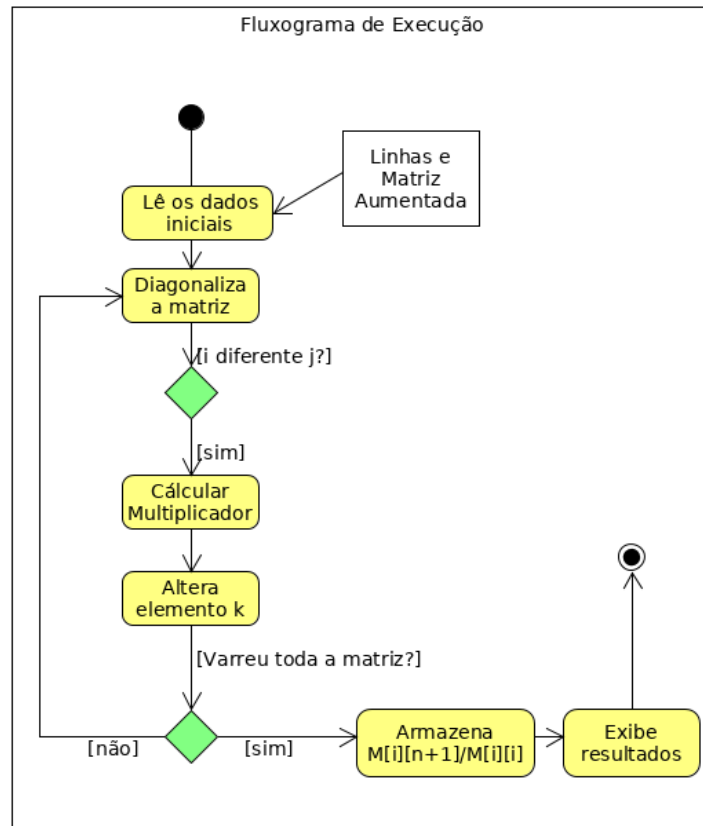


Figura 1: Fluxo de execução da solução

A solução elaborada neste relatório funciona da seguinte maneira. É necessário inserir a quantidade de linhas do sistema de equações lineares, em formato de matriz aumentada, que se quer resolver. Após isso o programa solicitará a inserção dos elementos de cada uma das linhas da matriz. Após completar a matriz de entrada o sistema irá fazer a diagonalização de acordo com o método de eliminação de Gauss-Jordan. Onde caso os índices i e j sejam diferentes, ou seja não fazem parte da diagonal principal, será aplicado o algoritmo de eliminação. Após haver apenas os elementos da diagonal principal, o método de solução se torna direto e com isso é possível encontrar os valores das incógnitas que se busca. As limitações do programa são entradas de matrizes de no máximo 10×10 e apenas para equações lineares.

4 Código Fonte

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <assert.h>
4 #define MAX_SIZE 10
5
6 int main() {
7
8     int qtd_pontos;
9
10    printf("Insira a quantidade de pontos conhecidos da função a ser
        interpolada: ");
11    scanf("%d",&qtd_pontos);
12
13    assert(qtd_pontos <= MAX_SIZE);
14
15    double temperatura[MAX_SIZE], ro[MAX_SIZE];
16
17    printf("Valores de x pré existentes: \n");
18
19    for(int i = 0; i < qtd_pontos; i++){
20        printf("Insira o valor de x_%d :", i);
21        scanf("%lf", &temperatura[i]);
22    }
23
24    printf("Valores de f(x) pré existentes: \n");
25
26    for(int i = 0; i < qtd_pontos; i++){
27        printf("Insira o valor de x_%d :", i);
28        scanf("%lf", &ro[i]);
29    }
30
31    double x;
32
33    printf("Insira o valor de x para o qual deseja obter f(x): ");
34    scanf("%lf", &x);
35
36    double l[MAX_SIZE];
37
38    for (int i = 0; i < MAX_SIZE; i++){
39        l[i] = 0.0;
40    }
41
42    for (int i = 0; i < qtd_pontos; i++){
43        double num = 1.0, den = 1.0;
44        for (int j = 0; j < qtd_pontos; j++){
45            if (i != j){
```

```
46         num *= (x - temperatura[j]);
47         den *= (temperatura[i] - temperatura[j]);
48     }
49 }
50 l[i] += num/den;
51 }
52
53 double p = 0.0;
54
55 for (int i = 0; i < qtd_pontos; i++){
56     p += ro[i] * l[i];
57 }
58
59 printf("O valor de f(x) é: %lf\n", p);
60
61 return EXIT_SUCCESS;
62 }
```

5 Resultados e discussões

Nesta seção discutiremos os resultados obtidos após a execução.

```
[0][wilton@asus]~/Dropbox/UnB/Métodos Numéricos/solutions/m4/solution
$ ./a.out
Insira a quantidade de pontos conhecidos da função a ser interpolada: 3
Valores de x pré existentes:
Insira o valor de x_0 :5
Insira o valor de x_1 :10
Insira o valor de x_2 :15
Valores de f(x) pré existentes:
Insira o valor de x_0 :0.9998
Insira o valor de x_1 :0.9997
Insira o valor de x_2 :0.9991
Insira o valor de x para o qual deseja obter f(x): 13
O valor de f(x) é: 0.999400
```

Figura 2: Resultado da execução do programa

Após a execução do programa obtemos os valores dos coeficientes que resulta no seguinte polinômio:

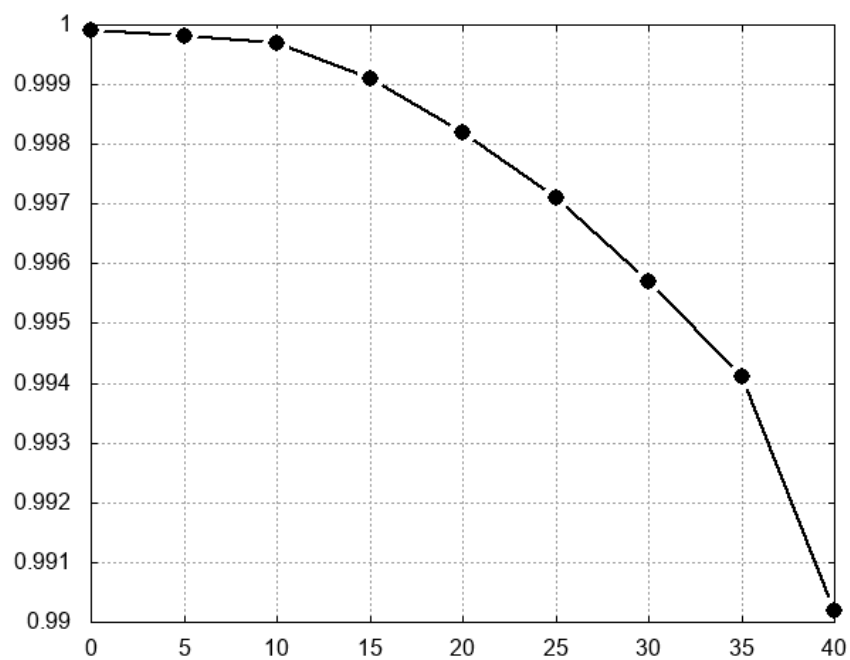
$$C(x) = 0,6523 - 0,2438x + 0,5682x^2 \quad (6)$$

O resultado encontrado a partir da solução proposta é condizente. Pois ao fazermos a comparação dos valores iniciais com os obtidos através da equação (6) encontramos valores bem próximos uns dos outros, como pode ser visto na tabela abaixo:

Valores iniciais do experimento										
$\frac{A}{A_1}$	0.10	0.20	0.30	0.40	0.50	0.60	0.70	0.80	0.90	1.00
C	0.62	0.63	0.64	0.66	0.68	0.71	0.76	0.81	0.89	1.00
Valores obtidos após o método										
$\frac{A}{A_1}$	0.10	0.20	0.30	0.40	0.50	0.60	0.70	0.80	0.90	1.00
C	0.63	0.63	0.63	0.65	0.67	0.71	0.76	0.82	0.89	0.98

Tabela 4: Valores experimentais e de regressão

E também graficamente, como é mostrado abaixo:



Sendo assim, o objetivo proposto no início do relatório foi satisfatoriamente alcançado.

6 Ferramentas

Todas as ferramentas utilizadas neste relatório são ferramentas open source (software livre). Permitindo assim que qualquer um possa reproduzir e contestar as afirmações presentes neste documento.

1. Arch Linux (<https://www.archlinux.org>)
 - Sistema operacional utilizado.
2. GCC (<https://gcc.gnu.org>)
 - Compilador de C utilizado para compilar a solução.
3. Python (<https://www.python.org>)
 - Linguagem de programação utilizada para conferir os valores da solução.
4. vim (<http://www.vim.org>)
 - Editor de texto.
5. L^AT_EX (<https://www.latex-project.org>)
 - Sistema tipográfico de alta qualidade (utilizado para elaborar o relatório).
6. Gnuplot (<http://www.gnuplot.info>)
 - Utilitário de representação gráfica (utilizado para plotagem do gráfico).
7. UMLet (<http://www.umlet.com>)
 - Ferramenta de UML (utilizado para criar o fluxo de execução).
8. Shutter (<http://shutter-project.org>)
 - Programa de captura de tela (utilizado para capturar os resultados).