1. Métodos numéricos

Nestas seções do **anexo numérico**, apresento algumas palavras sobre métodos numéricos que foram em parte usados no texto de *mecânica com MuPAD*. As explicações dadas podem servir para melhor entender as ferramentas que MuPAD nos oferece. Os programas são todos escritos na linguagem de programação do MuPAD. Está planejado de incorporar, no futuro, mais tópicos numéricos neste anexo.

Na *primeira seção* tratamos sobre representação de dados experimentais (interpolação)

Na *segunda* parte discutimos as equações diferenciais ordinarias da primeira ordem e os métodos numéricos de Euler, Heun e Runge -Kutta

Na *terceira* parte representamos as equações diferenciais de ordem superior de um e sistemas de equações.

1.1 Representação de dados experimentais (Interpolação)

No parágrafo 5.3.4 (Espalhamento de partículas alfa (II)) tivemos que fazer um gráfico dos dados experimentais de Geiger e Marsden do ano 1913. Afortunadamente existia uma função (fórmula de Rutherford) que pudemos traçar junto com os dados experimentais. Mas, na maioria das vezes, não conhecemos *a priori* tal função e é, precisamente, a nossa tarefa, de encontrar uma função aproximada, por exemplo em forma de um polinômio. Esse trabalho chama-se de **interpolação**.

Interpolar uma função f(x) consiste em aproximar essa função por uma outra função g(x), por exemplo por um polinômio $p_n(x) = a_o + a_1x + a_2x^2 + \cdots + a_nx^n$. A condição que tal polinômio debe satisfazer é passar pelos pontos experimentais $(x_o, f(x_o))$, $(x_1,f(x_1))$, ... , $(x_n,f(x_n))$, ou seja, $p_n(x_k) = f(x_k)$ para k = 0, 1, 2, ... , n. Existem várias maneiras para se obter tal polinômio. Uma consiste na resolução de um sistema linear para obter os coeficiente a_o, a_1, ... ,a_n. (Este será o nosso método.) Mais adiante estudaremos também dois outros métodos populares, a saber, os de Lagrange e o de Newton.

Seguimos agora a descrição dos passos aplicando MuPAD na buscada de um polinômio para os dados de Geiger e Marsden de 1913. (O material foi tirado da SciFace-publicação "Analysis mit MuPAD" de Kai Gehrs e Vera Verspohl.)

Os 7 pontos fornecidos das experiências são

P1:=[45,1435] ; P2:=[60,477] ; P3:=[75,211] ; P4:=[105,69.5] ; P5:=[120,51.9] ; P6:=[35,43] ; P7:=[150,33.1]

(P1[1] = 45 significa: o valor-x é de 45^o; P1[2] é o valor-y, que corresponde a 1435 impulsos contados.)

O polinômio é $p_n(x)$:= $p(x) = a_o + a_1x + a_2x^2 + a_3x^3 + a_4x^4 + a_5x^5 + a_6x^6$.

As 7 condições $p(x_k) = f(x_k)$ para o polinômio são

(O valor de g na posição x = P1[1] = 45 é igual a y = P1[2] =1435)

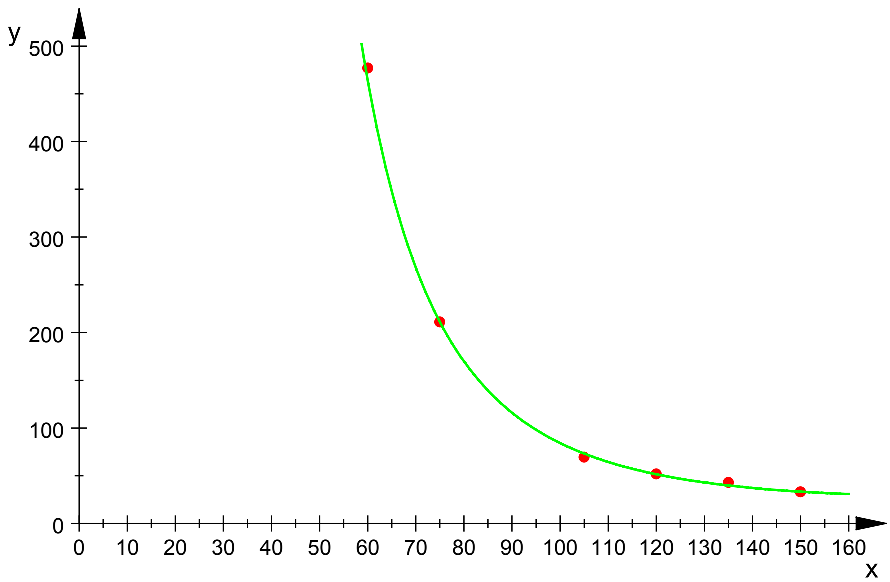
C1:= subs(g, x=P1[1]) = P1[2];
C2:= subs(g, x=P2[1]) = P2[2];
C3:= subs(g, x=P3[1]) = P3[2];
C4:= subs(g, x=P4[1]) = P4[2];
C5:= subs(g, x=P5[1]) = P5[2];
C6:= subs(g, x=P6[1]) = P6[2];
C7:= subs(g, x=P7[1]) = P7[2];

-
```
reset()://Rutherford
P1:=[45,1435]:
P2:=[60,477]:
P3:=[75,211]:
P4:=[105,69.5]:
P5:=[120,51.9]:
P6:=[135,43]:
P7:=[150,33.1]:
f:=29/(sin(w*PI/360))^4:

lista_de_pontos:= [P1,P2,P3,P4,P5,P6,P7]:

Pontos:=plot::Point2d(lista_de_pontos[i],Color=RGB::Red)
$ i=1..7:
Teoria:=plot::Function2d(f,Color=RGB::Green,w=50..160):
```

```
• plot(Pontos,Teoria, ViewingBoxXRange =0..160,
  ViewingBoxYRange =0..500)
```




```
• reset()://Marsden com interpolação

  P1:=[45,1435]:
  P2:=[60,477]:
  P3:=[75,211]:
  P4:=[105,69.5]:
  P5:=[120,51.9]:
  P6:=[135,43]:
  P7:=[150,33.1]:

   p := a0 + a1*x + a2*x^2 + a3*x^3 + a4*x^4 + a5*x^5 +
  a6*x^6:

  C1:= subs(p, x=P1[1]) = P1[2]:
  C2:= subs(p, x=P2[1]) = P2[2]:
  C3:= subs(p, x=P3[1]) = P3[2]:
  C4:= subs(p, x=P4[1]) = P4[2]:
  C5:= subs(p, x=P5[1]) = P5[2]:
  C6:= subs(p, x=P6[1]) = P6[2]:
  C7:= subs(p, x=P7[1]) = P7[2]:
  L:=solve({C1,C2,C3,C4,C5,C6,C7},{a0,a1,a2,a3,a4,a5,a6,a7}):
```

```
p:=subs(p,op(L[1])):

lista_de_pontos:= [P1,P2,P3,P4,P5,P6,P7]:

plot(plot::Function2d(p,Color=RGB::Green,x=50..160),
plot::Point2d(lista_de_pontos[i],Color=RGB::Red) $ i=1..7,
ViewingBoxXRange =0..160,ViewingBoxYRange =0..500)
```

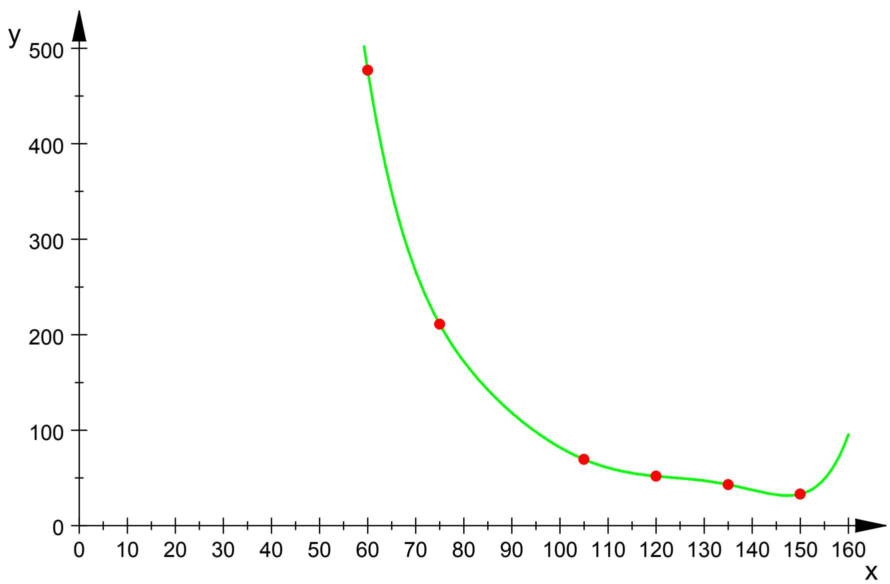


Fig.:1.1-2

Agora extraímos alguns valores usando diferentes métodos:

```
p:=subs(p,op(L[1])):

subs(p,x=75);
```

211.0

- ```
  a0:=op(op(op(L[1])[1])[2]);
  a1:=op(op(op(L[1])[2])[2]);
  a2:=op(op(op(L[1])[3])[2]);
  a3:=op(op(op(L[1])[4])[2]);
  ```

29374.9

-1620.836048

37.72595132

-0.4683297178

```
//ou mais curto:

a0:=op((L[1][1]),1);
a1:=op((L[1][2]),1);
a2:=op((L[1][3]),1);// etc.
```
```