Universidade de Brasília

Relatório de atividade do módulo 6

Métodos Numéricos para Engenharia

Resolução de Equações Diferenciais

Aluno:
Wilton Rodrigues

Matrícula:
13/0049212

5 de dezembro de 2016

1 Introdução

O objetivo deste relatório é exercitar os conceitos aprendidos em aula, com relação ao tópico: Resolução de Equações Diferenciais.

O problema a ser solucionado é uma equação diferencial de primeira ordem com condições iniciais que trata da velocidade de queda de um corpo considerando a resistência do ar, que é dada pela seguinte equação:

$$m\frac{dv}{dt} = mg - kv \tag{1}$$

Onde: $g = 10m/s^2$ é a aceleração gravitacional, $m = 1kg$ é a massa do corpo e $k = 10^-2kg/s$ é o coeficiente de amortecimento do ar. Considerando $v(0) = 0$

O objetivo deste relatório será, através do método de Runge-Kutta, plotar o gráfico da equação (1) com 100 valores de t no intervalo $0 < x < 10$.

2 Metodologia

Baseando-se no método de Runge-Kutta de 3ª ordem, como dito anteriormente, tem-se a seguinte relação:

$$\frac{df(x)}{dx} = g(x, f(x)), f(x_0) = f_0 \tag{2}$$

A idéia básica deste método é aproveitar as qualidades dos métodos da série de Taylor e ao mesmo tempo eliminar seu maior defeito que é o cálculo de derivadas de $f(x, y)$ que torna os métodos de série de Taylor computacionalmente ineficientes. O método de Runge-Kutta faz uma aproximação para $f(x + \Delta x)$ em um passo único. Para o passo $i + 1$, a seguinte equação produz o resultado:

$$\begin{aligned} f_{i+1} = f_i + \frac{\Delta x}{6}(k_1 + 4k_2 + k3),\ \textit{onde}: \\ k_1 = g(x, f(x)) \\ k_2 = g(x + \frac{\Delta x}{2}, f(x) + k_1\frac{\Delta x}{2}) \\ k_3 = g(x + \Delta x, f(x) - k_1\Delta x + 2k_2\Delta x) \end{aligned} \tag{3}$$

Já que $f_i = f(x) \Rightarrow f_{i+1} = f(x + \Delta x)$, ou seja, o problema se torna do tipo PVI (Problema de Valor Inicial), onde é preciso conhecer um valor inicial para $f(x)$ para

então os valores já calculados para este termo, sejam utilizados para calcular os demais termos.

3 Diagrama esquemático de execução

Nesta seção, encontra-se o fluxo de execução utilizando a linguagem C. Que é apresentada na próxima sessão.

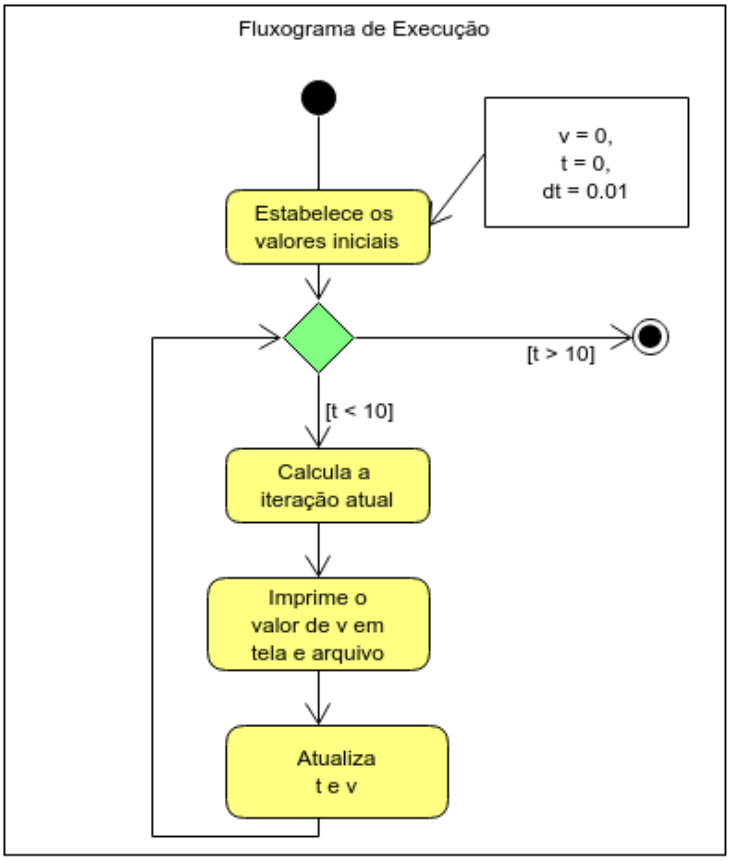

Figura 1: Fluxo de execução da solução

A solução elaborada neste relatório funciona da seguinte maneira. Baseando-se nos valores iniciais fornecidos pela questão, o programa calcula os valores para a iteração atual, de acordo com o método de Runge-Kutta. Após isto é exibido o valor atual de v, então os valores são incrementados. Após isso, caso o limite máximo não tenha sido atingido, o programa executa o loop novamente.

4 Código Fonte

```c
#include <stdio.h>
#include <stdlib.h>

char file_name[]="in.dat";
FILE *file;
double m = 1.0, g = 10.0, k = 1e-2;

double f(double v) {
  return g - k / m * v;
}

int main() {
  double v = 0, t = 0, dt = 1e-2;
  file = fopen(file_name, "w");

  if(file == NULL){
    printf("Erro, nao foi possivel abrir o arquivo\n");
    return EXIT_FAILURE;
  }
  else{
    while(t <= 10) {
      double k1 = f(v), k2 = f(v + k1 * dt / 2), k3 = f(v - k1 * dt + 2
    * k2 * dt);
      printf("%.2lf %.15lf\n", t, v);
      fprintf(file, "%.2lf %.15lf\n", t, v);
      v += (k1 + 4 * k2 + k3) * dt / 6;
      t += dt;
    }
    fclose(file);
  }
  return EXIT_SUCCESS;
}
```

5 Resultados e discussões

Nesta seção discutiremos os resultados obtidos após a execução do programa apresentado na seção anterior. Abaixo é apresentada a plotagem do gráfico da equação (1) para o intervalo $[0, 10]$, como havia sido estabelecido no início do relatório. Para gerar estes pontos, basta executar o programa. Um arquivo será gerado com todos os pontos e os respectivos valores. Então, através da ferramenta Gnuplot, foi feita a plotagem do gráfico, com os seguintes comandos:

```
#Instruções para plotar
set grid
set xtics 0,1,10
plot "in.dat" using 1:2 title 'Gráfico com 100 valores'
set terminal png
set output 'graph.png'
replot
#Fim
```

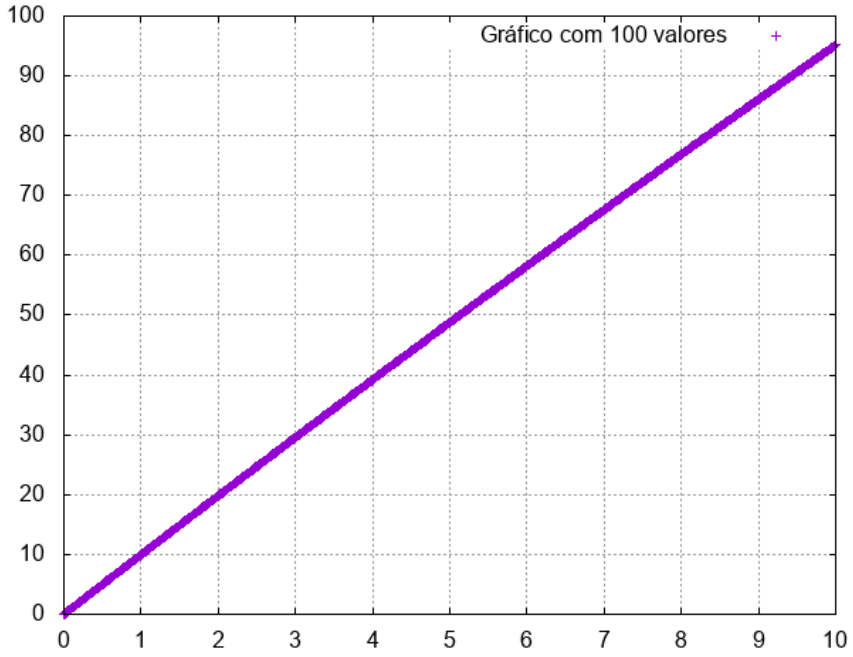

Figura 2: Plotagem de 100 valores no intervalo $[0, 10]$

Abaixo, segue alguns dos resultados obtidos após a execução do programa. A solução proposta gera 1000 valores, devido a esta quantidade de resultados apenas alguns foram exibidos para comprovar o funcionamento da solução.

```
[0][wilton@asus]~/Dropbox/UnB/Métodos Numéricos/solutions/m6/solution
$ ./a.out
0.00    0.000000000000000       0.01    0.099995000166667       0.02    0.199980001333275
0.03    0.299955004499675       0.04    0.399920010665617       0.05    0.499875020830750
0.06    0.599820035994626       0.07    0.699755057156693       0.08    0.799680085316303
0.09    0.899595121472705       0.10    0.999500166625050       0.11    1.099395221772389
0.12    1.199280287913671       0.13    1.299155366047748       0.14    1.399020457173370
0.15    1.498875562289189       0.16    1.598720682393754       0.17    1.698555818485519
0.18    1.798380971562833       0.19    1.898196142623948       0.20    1.998001332667017
0.21    2.097796542690090       0.22    2.197581773691121       0.23    2.297357026667961
0.24    2.397122302618363       0.25    2.496877602539980       0.26    2.596622927430364
0.27    2.696358278286970       0.28    2.796083656107150       0.29    2.895799061888158
0.30    2.995504496627149       0.31    3.095199961321176       0.32    3.194885456967195
0.33    3.294560984562059       0.34    3.394226545102526       0.35    3.493882139585249
0.36    3.593527769006785       0.37    3.693163434363591       0.38    3.792789136652023
0.39    3.892404876868338       0.40    3.992010656008693       0.41    4.091606475069147
1.71    16.954624817999981      1.72    17.052924440455147      1.73    17.151214233439550
1.74    17.249494197936084      1.75    17.347764334927554      1.76    17.446024645396658
1.77    17.544275130326000      1.78    17.642515790698084      1.79    17.740746627495319
1.80    17.838967641700012      1.81    17.937178834294375      1.82    18.035380206260516
1.83    18.133571758580452      1.84    18.231753492236098      1.85    18.329925408209270
1.86    18.428087507481688      1.87    18.526239791034975      1.88    18.624382259850648
1.89    18.722514914910136      1.90    18.820637757194767      1.91    18.918750787685767
1.92    19.016854007364266      1.93    19.114947417211297      1.94    19.213031018207793
1.95    19.311104811334591      1.96    19.409168797572431      1.97    19.507222977901950
1.98    19.605267353303692      1.99    19.703301924758097      2.00    19.801326693245514
2.01    19.899341659746188      2.02    19.997346825240271      2.03    20.095342190707814
2.04    20.193327757128770      2.05    20.291303525482999      2.06    20.389269496750252
2.07    20.487225671910192      2.08    20.585172051942383      2.09    20.683108637826283
9.45    90.172265373973616      9.46    90.263243598449179      9.47    90.354212725557176
9.48    90.445172756207285      9.49    90.536123691309115      9.50    90.627065531772175
9.51    90.717998278505888      9.52    90.808921932419580      9.53    90.899836494422473
9.54    90.990741965423737      9.55    91.081638346332412      9.56    91.172525638057451
9.57    91.263403841507738      9.58    91.354272957592059      9.59    91.445132987219111
9.60    91.535983931297480      9.61    91.626825790735680      9.62    91.717658566442125
9.63    91.808482259325160      9.64    91.899296870293000      9.65    91.990102400253804
9.66    92.080898850115631      9.67    92.171686220786427      9.68    92.262464513174081
9.69    92.353233728186382      9.70    92.443993866731006      9.71    92.534744929715558
9.72    92.625486918047557      9.73    92.716219832634422      9.74    92.806943674383476
9.75    92.897658444201952      9.76    92.988364142997014      9.77    93.079060771675699
```

Figura 3: Alguns resultados da execução do programa

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. LaTeX (`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).