Lab 7: Equações Diferenciais Ordinárias (EDO)

INF1608 – Análise Numérica

Leonardo Quatrin Campagnolo

lquatrin@tecgraf.puc-rio.br

Departamento de Informática, PUC-Rio

Considere a solução de equações diferenciais ordinárias expressas por:

$$y'(t) = f(t, y(t))$$

O método de Runge-Kutta de ordem 4, considerando passos h constantes, é dado por:

$$\begin{aligned}
k_1 &= hf(t, y(t)) \\
k_2 &= hf(t + h/2, y(t) + k_1/2) \\
k_3 &= hf(t + h/2, y(t) + k_2/2) \\
k_4 &= hf(t + h, y(t) + k_3) \\
y(t + h) &= y(t) + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4)
\end{aligned}$$

Para se ter uma estratégia com passos adaptativos, pode-se adotar o método que avalia um avanço com passo h, obtendo $y_1(t + h)$, e um avanço com dois passos $h/2$, obtendo y_2. Como o erro é proporcional a h^5, tem-se que o erro local associado a $y_2(t + h)$ pode ser estimado por:

$$\Delta = \frac{y_2 - y_1}{15}$$

Considerando ϵ a tolerância (erro local aceitável), o fator de ampliação/redução do passo é dado por:

$$f = \sqrt[5]{\frac{\epsilon}{|\Delta|}}$$

Se $f \geq 1.0$, validamos o passo, e adotamos como solução a resposta de ordem superior acrescida da estimativa do erro: $y_2 + \Delta$. Neste caso, a próxima iteração pode ser avaliada com um novo passo:

$$h' = \min(1.2, f)\ h$$

Caso contrário, o passo é invalidado e tem que ser reavaliado com h atualizado:

$$h' = 0.8\ f\ h$$

1. Pede-se:

   (a) Implemente o método de Runge Kutta com passo constante. Sua função deve receber como parâmetros o tempo inicial t_0, o tempo final t_1, o passo de integração h, o valor inicial $y(t_0)$ e a função derivada $f(t, y(t))$, tendo como retorno o valor no tempo final $y(t_1)$, seguindo o protótipo:

   ```
   double RungeKutta (double t0, double t1, double y0,
                      double (*f) (double t, double y), double h);
   ```

   (b) Implemente o método de Runge Kutta adaptativo, conforme apresentado acima. Sua função deve receber como parâmetros o tempo inicial t_0, o tempo final t_1, o valor inicial $y(t_0)$, a função derivada $f(t, y(t))$ e a tolerância do erro local tol, tendo como retorno o valor no tempo final $y(t_1)$. Como valor de passo inicial, pode-se adotar $h_0 = 10^{-7}$. A função deve ter o seguinte o protótipo:

   ```
   double RungeKuttaAdapt (double t0, double t1, double y0,
                           double (*f) (double t, double y), double tol);
   ```

2. Para testar suas funções, avalie $y(2.4)$ sabendo que $y' = ty + t^3$, com $y(0) = -1$. Para o método com passos constante, use $h = 0.001$; para o método com passo adaptativo, use $tol = 10^{-12}$.

   Sabe-se que a solução desta EDO para $y(0) = -1$ é:

   $$y(t) = e^{\frac{t^2}{2}} - t^2 - 2$$

   Compare os resultados obtidos pelos métodos calculando o *erro relativo* para cada caso. Compare também o número de vezes que a função foi avaliada por cada método.

Em seguida, avalie $y(3.0)$ para $y' = 10(1 - y)$, com $y(0) = 0$ (solução $y(t) = 1 - \frac{e^{-10t}}{2}$).

Agrupe os protótipos das funções pedidas em um módulo "ode.h" e as implementações em um módulo "ode.c". Escreva o teste em outro módulo "main.c".

Entrega: O código fonte deste trabalho (isto é, os arquivos "ode.c", "ode.h" e "main.c", *e eventuais códigos de laboratórios passados usados na solução*) devem ser enviados via página da disciplina no EAD até domingo, dia 20/10. O sistema não receberá trabalhos com atraso.