MAP2310 - Métodos Numéricos em Equações Diferenciais I
Primeiro Semestre de 2012 (Noturno)

Exercício Programa 2

1 Introdução

Este exercício programa tem como objetivo a implementação de um método Runge-Kutta com controle automático do tamanho do passo. Ele será usado para alguns testes envolvendo sistemas de equações diferenciais ordinárias.

Seu programa deve ser entregue na página do curso no sistema Moodle do Stoa (*disciplinas.stoa.usp.br*) até o dia 25 de junho. O programa deve ser escrito em Linguagem C e ser compilado e executado com o compilador DevC++ (ver link na página da disciplina). Caso você desnvolva o programa em outro compilador, verifique se ele compila e executa no DevC++. Programas que não compilarem terão notas muito baixas.

2 Pares embutidos de métodos Runge-Kutta

Conforme discutido em aula, o controle do passo será feito usando um par embutido de ordens 4 e 5. Os parâmetros do método são os de Cash-Karp apresentados na tabela abaixo:

Parâmetros do par Runge-Kutta embutido de Cash-Karp

i	a_i	b_{ij}					$\tilde{c}_i$	c_i
1							$\frac{37}{378}$	$\frac{2825}{27648}$
2	$\frac{1}{5}$	$\frac{1}{5}$					0	0
3	$\frac{3}{10}$	$\frac{3}{40}$	$\frac{9}{40}$				$\frac{250}{621}$	$\frac{18575}{48384}$
4	$\frac{3}{5}$	$\frac{3}{10}$	$-\frac{9}{10}$	$\frac{6}{5}$			$\frac{125}{594}$	$\frac{13525}{55296}$
5	1	$-\frac{11}{54}$	$\frac{5}{2}$	$-\frac{70}{27}$	$\frac{35}{27}$		0	$\frac{277}{14336}$
6	$\frac{7}{8}$	$\frac{1631}{55296}$	$\frac{175}{512}$	$\frac{575}{13824}$	$\frac{44275}{110592}$	$\frac{253}{4096}$	$\frac{512}{1771}$	$\frac{1}{4}$
$j =$		1	2	3	4	5		

Os coeficientes c_i e $\tilde{c}_i$ são dos métodos de ordem 4 e 5, respectivamente.

Em linhas gerais, o algoritmo é descrito da seguinte forma. Deseja-se aproximar a solução do *sistema de equações diferenciais ordinárias*

$$\dot{x} = f(t, x)$$

onde o campo f é uma função de um subconjunto aberto de $R \times R^N$ em R^N (se $N = 1$, temos uma equação escalar). Suponha que estamos no instante t_j com a aproximação η_j (que é um vetor N-dimensional) e desejamos avançar um passo no tempo, usando um valor h como candidato ao tamanho do passo. Então:

1) Calcule

$$k_1 = f(t_j, \eta_j)$$
$$k_i = f\left(t_j + ha_i, \eta_j + h\sum_{l=1}^{i-1} b_{il}k_l\right), \quad 2 \le i \le 6$$

2) Calcule a estimativa do erro

$$d = \sum_{i=1}^{6}(\tilde{c}_i - c_i)k_i$$

3) Calcule

$$q = \max_{1\le n\le N} \frac{|d(n)|}{EPS * (|\eta_j(n)| + |k_1(n)| + \delta)}$$

onde $d(n)$, $\eta_j(n)$ e $k_1(n)$ são as n-ésimas componentes de d, η_j e k_1, respectivamente, EPS é uma precisão especificada pelo usuário e δ é um número pequeno para evitar uma eventual divisão por zero.

4) Se $q < 1$, *aceite* o valor h, avance um passo

$$\eta_{j+1} = \eta_j + h\sum_{i=1}^{6}\tilde{c}_i k_i$$

e atualize o candidato a tamanho do passo

$$h \leftarrow \min\left\{5h, 0.9q^{-0.2}h\right\}$$

para o novo instante. *Saia do laço.*

5) Se $q \ge 1$, *rejeite* h, diminua o seu valor para

$$h \leftarrow \max\left\{0.1h, 0.9q^{-0.25}h\right\}$$

e volte para **1)**.

3 Implementação e resultados

Implemente o algoritmo acima na linguagem C. Use dupla precisão (double), $EPS = 10^{-6}$ e $\delta = 10^{-30}$. Para cada um dos testes abaixo, imprima os valores mínimo e máximo de h utilizados, a quantidade de passos mal e bem sucedidos, quantas vezes o campo f foi avaliado e quantos instantes intermediários foram armazenados. Como primeiro valor do passo a ser testado, use $h = 0.001$. Você deverá gerar um arquivo de saída contendo as aproximações para apresentar gráficos.

4 Testes

Teste o seu programa com os seguintes exemplos:

Exemplo 1 (Equação escalar)

$$\dot{x} = 10e^{-\frac{1}{2}\left(\frac{t-2}{0.075}\right)^2} - 0.6x, \quad x(0) = 0.5, \quad 0 \leq t \leq 4.$$

Apresente um gráfico de x como função de t, destacando os instantes intermediários. Explique a variação do tamanho do passo obtida.

Exemplo 2 (Um modelo presa-predador)

$$\begin{aligned} \dot{x} &= 1.2x - x^2 - \frac{xy}{x+0.2} \\ \dot{y} &= \frac{1.5xy}{x+0.2} - y \end{aligned}$$

Faça simulações no intervalo $0 \leq t \leq 60$ para dois conjuntos de condições iniciais: $x(0) = 1$, $y(0) = 0.75$ e $x(0) = 0.75$, $y(0) = 0.25$. Represente as soluções no plano de fase (x, y). O que você pode concluir do resultado?

Exemplo 3 (Sistema de Lorenz)

$$\begin{aligned} \dot{x} &= \sigma(y - x) \\ \dot{y} &= rx - y - xz \\ \dot{z} &= xy - bz \end{aligned}$$

Usando $\sigma = 10$, $b = 8/3$ e $(x(0), y(0), z(0)) = (5, 5, 5)$, faça simulações no intervalo $0 \leq t \leq 40$ para os seguintes valores do parâmetro r: 1, 15, 20, 22 e 28. Em cada caso, apresente gráficos de x, y e z como funções de t e represente também a solução no espaço de fase (x, y, z). No caso $r = 28$, faça também uma simulação com condições iniciais 5, 5.001 e 5, e compare o resultado com o anterior.

5 Bibliografia

Para ler mais sobre pares Runge-Kutta embutidos, veja por exemplo

R. L. Burden & J. D. Faires, *Análise Numérica*, Thomson, 2003

J. Stoer & R. Bulirsch, *Introduction to Numerical Analysis*, Springer, 2002

Discussões sobre modelos presa-predador e o sistema de Lorenz podem ser encontradas em

M. W. Hirsch, S. Smale & R. L. Devaney, *Differential Equations, Dynamical Systems, and an Introduction to Chaos*, Elsevier, 2004