Aula 2

• Erros:

→ de Arredondamento e de Truncatura

→ **Global e Local**

• Problemas de valor inicial (PVI)

Método de Euler-Cromer

Métodos Implícitos:

Euler Implícito

Crank-Nicolson

Exemplo de Aplicação: o oscilador hamónico simples (OHS)

17-02-2020

Erros de Arredondamento e de Truncatura

Erros de arredondamento (round-off error)

Erros devidos ao número limite de dígitos que podem ser retidos pelo computador. Em geral os cáculos computacionais envolvem números com *dupla precisão*, i.e., os números são representados com 16 dígitos. Esta limitação pode ter um impacto significativo nos cáculos numéricos e em particular para métodos iterativos.

Erros de truncatura ou discretização (truncation error)

Erros de truncatura, ou discretização, são erros causados pela natureza das técnicas empregues para a estimativa de uma dada solução da ODE. Por exemplo, no método de Euler, o declive à curva no intervalo h, é aproximado pelo declive à curva no 1º ponto do intervalo,sendo assim uma fonte de erro.

Estes erros são constituidos por duas partes:

o erro local de truncatura, que resulta da aplicação do método em questão numa única iteração;

o erro global de truncatura, que resulta das aproximações feitas em iterações prévias. A soma dá-nos o erro global de truncatura

Erro Local

É o erro cometido em cada iteração. Para se estimar o erro local considera-se a expansão da solução em série de Taylor, e compara-se com aproximação da mesma obtida pelo método de Euler. A diferença entre as duas dá-nos uma estimativa do erro local, que é cometido em cada iteração.

Consider-se a expansão da solução em série de Taylor da solução y, supondo que a função possui derivadas contínuas:

$$y(t+h) = y(t) + \frac{dy}{dt}(t) * h + \frac{d^2y}{dt^2}(c) * \frac{h^2}{2!}$$

Em que $c \in [t,\ t+h]$. Como $\frac{dy}{dt} = f(t,y)$, a equação anterior reduz-se ao esquema iterativo de Euler,

$$y_{n+1} = y_n + h * f\left(t_{n,} y_n\right) + O(h^2)$$

Então, o erro de truncatura é $O(h^2)$, sendo dado por $\frac{d^2y}{dt^2}(c_n) * \frac{h^2}{2!}$. Ou seja, o erro cometido na iteração k+1, designa-se por **erro local**, e é dado por:

$$\epsilon_{k+1} = const * h^2$$

Erro Global

O erro global é o erro acumulado durante todo o processo de cálculo. Considere que a equação diferencial é resolvida numericamente, deste o instante inicial t_0 a um instante final t_f, em que o número total de pontos N é dado por

$\frac{(t_f - t_0)}{h} + 1.$

Se for conhecida a solução exata da equação y, num dado intante t_k o **erro global** será dado pela diferença entre a solução exata $y(t_k)$ e a solução numérica y_k:

$$E_k = y(t_k) - y_k$$

OBS: no final de todas as iterações o erro global será dado por $y(t_f) - y_N$.

E se a solução exata não for conhecida?

→Para uma iteração, o erro de truncatura é $O(h^2)$, sendo dado por $\frac{d^2y}{dt^2}(c) * \frac{h^2}{2!}$. Após k iterações uma estimativa do erro global poderá ser $\sum_{j=1}^{k} \frac{d^2y}{dt^2}(c_j) * \frac{h^2}{2!}$.

Pode mostra-se que o **erro global** do método de Euler é da ordem de h, ou seja,

$$E_k = const * h$$

Pode explicar-se qualitativamente este resultado notando que:

- O erro local é da ordem de h^2.
- O número de pontos aumenta linearmente com h^{-1} .

O erro global aumenta com o número de pontos e com o erro introduzido em cada iteração, sendo da ordem de h.

Erro Global e número de iterações

Dado que h é inversamente proporcional ao número de pontos, (com rigor a N-1), então,

$$y(t_f) - y_N = const * N^{-1}$$

Pelo que se pode escrever:

$$y_N = const' * N^{-1} + y(t_f)$$

Se a equação anterior se verificar, um gráfico da solução numérica y_k em função de N^{-1} deve ter como ordenada na origem uma estimativa do valor exato $y(t_f)$.

LEITURA_1

Método de Euler –Exemplo de Aplicação (Chapra)

Considere o exemplo da Leitura_1 da Aula_1:

$$\frac{dy}{dt} = -2t^3 + 12t^2 - 20t + 8.5$$

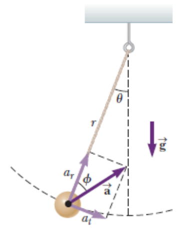
No exemplo anterior a equação foi integrada numericamente pelo método de Euler de t=0 a t=3, com passo temporal de 0.5. A condição inicial em t=0 é y=1. A solução analítica é dada por: $y = -0.5\, t^4 + 4t^3 - 10t^2 + 8.5t + 1$.

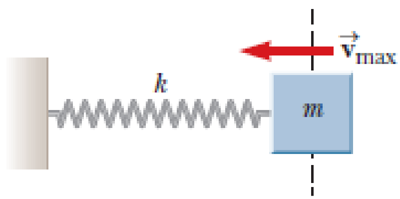
Com o auxilio de uma calculadora obtenha os erros local e global, absolutos e relativos, para h=0.5.

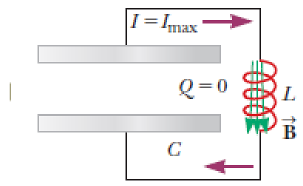
Instabilidade do método de Euler aplicado ao OHS

Na presente aula não vão ser discutidas formalmente as questões de convergência e estabilidade dos métodos numéricos estudados, mas o exemplo seguinte ajuda a entender o que está em causa. Até ao presente já estudou aguns sistemas físicos, que em determindas condições podem ser considerados osciladores harmónicos simples.

Recorde por exemplo, o pêndulo simples, o sistema massa-mola, e o circuito LC .







No caso do pêndulo, a posição angular do massa, solução da equação do movimento, pode ser descrita por $\theta(t) = \theta_{max}\cos(\omega\, t + \varphi)$, sendo $\omega = \sqrt{\frac{g}{l}}$, a frequência da oscilação e φ a fase inicial. De modo semelhante, no circuito LC, a evolução da carga aos terminais do condensador é dada por $C(t) = C_{max}\cos(\omega\, t + \varphi)$, com $\omega = 1/\sqrt{LC}$ e φ a fase inicial.

Considere-se o sistema massa-mola, de massa m e cuja constante da mola é K. Pela 2ª lei de Newton sabe-se que $m\,\frac{d^2y}{dt^2} = -Ky$. Para oscilações de pequena amplitude o movimento é hamónico simples, i.e, $y(t) = A\cos(\omega\, t + \varphi)$, com $\omega = \sqrt{\frac{K}{m}}$ e φ a fase inicial. (Confirme que esta expressão verifica a equação do movimento).

Atendendo a que a ODE é de 2ª ordem, e como já foi anteriormente apresentado, o método de Euler permite escrever:

$$\begin{cases} v_{k+1} &= v_k + f\left(t_{k,} y_k, v_k\right) * h \\ y_{k+1} &= y_k + v_k * h \end{cases}$$

Então, para o sistema massa-mola (OHS), estas equações podem escrever-se como:

$$\begin{cases} v_{k+1} &= v_k - \omega^2 y_k * h \\ y_{k+1} &= y_k + v_k * h \end{cases}$$

Recorde também que a energia total do sistema massa mola, no instante t_k, se pode escrever como:

$$E_k^{total} = \frac{1}{2}m\omega^2{y_k}^2 + \frac{1}{2}\,m\,{v_k}^2 = \frac{1}{2}\,K\,A^2$$

Então, num instante posterior t_{k+1} a energia pode escrever-se como:

$$E_{k+1}^{total} = \frac{1}{2}m\omega^2{y_{k+1}}^2 + \frac{1}{2}\,m\,{v_{k+1}}^2$$

$$= \frac{1}{2}m\omega^2(y_k + v_k * h)^2 + \frac{1}{2}\,m\,(v_k - \omega^2 y_k * h)^2$$

$$= \frac{1}{2}m\omega^2\left({y_k}^2 + {v_k}^2 * h^2 + 2 * y_k * v_k * h\right) + \frac{1}{2}\,m\left({v_k}^2 + \omega^4{y_k}^2 * h^2 - 2 * v_k * \omega^2 y_k * h\right)$$

$$= \left(\frac{1}{2}m\omega^2{y_k}^2 + \frac{1}{2}m{v_k}^2\right) + \frac{1}{2}m\omega^2h^2\left({v_k}^2 + \omega^2{y_k}^2\right)$$

$$= E_k^{total} + \frac{1}{2}m\omega^2h^2\left({v_k}^2 + \omega^2{y_k}^2\right)$$

$E_{k+1}^{total} > E_k^{total}$ *A energia total, que se devia conservar, aumenta em todas as iterações!*

Método de Euler-Cromer

Uma moficação do método de Euler, usada especialmente para conseguir estabilidade em problemas de movimento oscilatório, é o método de **Euler-Cromer**, também conhecido como **Euler semi-implícito** ou **Euler semi-explícito**, que se descreve em seguida.

MÉTODO de EULER - CROMER

$$\begin{cases} v_{k+1} = v_k + f(t_{k,} y_k, v_k) * h \\ y_{k+1} = y_k + v_{k+1} * h \end{cases}$$

No caso do OHS, fica,

$$\begin{cases} v_{k+1} = v_k - \omega^2 y_k * h \\ y_{k+1} = y_k + v_{k+1} * h \end{cases}$$

Note-se que estas equações podem ser usadas diretamente no programa, desde que se use a ordem certa. No método de Euler-Cromer, a energia varia durante cada ciclo de oscilação, mas conserva-se de ciclo para ciclo. (Não vamos fazer a demonstração formal).

Métodos IMPLÍCITOS

Note que o método de Euler-Cromer é aplicável apenas a casos partículares. Num problema de valor inicial descrito por uma ODE de primeira ordem, o método não pode sequer ser aplicado. Uma abordagem geral para tentar conseguir a estabilidade é usar o método de **Euler Implicíto:**

MÉTODO de EULER - IMPLÍCITO

$$y_{k+1} = y_k + f\left(t_{k+1}, y_{k+1}\right) * h$$

Ou o método dos trapézios, conhecido como método de **Crank-Nicolson**:

MÉTODO de CRANK-NICOLSON

$$y_{k+1} = y_k + \left[f\left(t_k, y_k\right) + f\left(t_{k+1}, y_{k+1}\right)\right] * \frac{h}{2}$$

Métodos IMPLÍCITOS

Nestes casos, y_{k+1} aparece nos dois membros da equação, logo temos que resolver em cada passo uma equação que pode ser não linear se *f* for não linear. Estamos perante **métodos implícitos** que ganham em estabilidade mas perdem pelo aumento da complexidade de cada iteração.

Ordem dos Métodos Implícitos

Pode provar-se que o **erro global** do método:

- de **Euler-Implícito** é de ordem **h**;
- de **Crank-Nicolson** é de ordem $\mathbf{h^2}$.

Aplicação do método de Euler Implícito ao OHS

Atendendo a que a ODE é de 2ª ordem, vamos obter um sistema de duas ODEs de 1ª ordem. (*Note que:a 1ª integração dá-nos a velocidade, e a 2ª integração dá-nos a posição*). Aplicando método de Euler Implícito ao sistema de equações, obtém-se,

$$\begin{cases} v_{k+1} = v_k + f(t_{k+1}, y_{k+1}, v_{k+1}) * h \\ y_{k+1} = y_k + v_{k+1} * h \end{cases}$$

Ou seja,

$$\begin{cases} v_{k+1} = v_k - \omega^2 y_{k+1} * h \\ y_{k+1} = y_k + v_{k+1} * h \end{cases}$$

Este sistema pode ser resolvido numericamente nesta forma? Porquê?

Aplicação do método de Euler Implícito ao OHS

Se substituirmos a 1ª expressão na 2ª, e resolvermos em ordem a y_{k+1}, obtém-se o seguinte sistema:

$$\begin{cases} y_{k+1} = \dfrac{y_k + v_k * h}{1 + \omega^2 * h^2} \\ v_{k+1} = v_k - \omega^2 y_{k+1} * h \end{cases}$$

Este sistema de equações já pode ser programado!

(Exercício: resolva o mesmo problema em ordem a v_{k+1})

O sistema pode ser resolvido usando-se uma função intrínseca do MATLAB, a rotina **`LINSOLVE.`** Para tal há que escrever o sistema de forma adequada (Escreva ***help linsolve*** na linha de comando do MATLAB).

Aplicação do método de Euler Implícito ao OHS

Reescrevemos o sistema isolando à *esquerda* os termos com indíce $k+1$ e à *direita* os termos com indíce k bem como os termos independentes (se for o caso):

$$\begin{cases} y_{k+1} + (-h) * v_{k+1} = y_k \\ \omega^2 * h * y_{k+1} + v_{k+1} = v_k \end{cases}$$

Este sistema pode ser escrito em notação matricial,

$$A\boldsymbol{Z} = \boldsymbol{b}$$

Onde A é a matriz do sistema, e é uma matriz constante no exemplo presente,

$$A = \begin{pmatrix} 1 & -h \\ \omega^2 h & 1 \end{pmatrix}$$

$\boldsymbol{Z}$ é o vetor que queremos calcular (a solução do sistema), e $\boldsymbol{b}$ um vetor de termos independentes que varia de iteração para iteração,

$$\boldsymbol{Z} = \begin{pmatrix} y_{k+1} \\ v_{k+1} \end{pmatrix}, \qquad \boldsymbol{b} = \begin{pmatrix} y_k \\ v_k \end{pmatrix}$$

Aplicação do método de Crank-Nicolson ao OHS

Relembrando que no exemplo considerado, o OHS, a ODE é de 2ª ordem, pelo vamos obter um sistema de duas ODEs de 1ª ordem, vamos aplicar o método de Crank-Nicolson ao OHS.

Obtém-se neste caso,

$$\begin{cases} v_{k+1} = v_k + \left[f\left(t_{k,} y_k\right) + f\left(t_{k+1,} y_{k+1}\right)\right] * \dfrac{h}{2} \\ y_{k+1} = y_k + \left(v_k + v_{k+1}\right) * \dfrac{h}{2} \end{cases}$$

Substituindo a função *f* e trocando a ordem das equações, obtém-se,

$$\begin{cases} y_{k+1} = y_k + \left(v_k + v_{k+1}\right) * \dfrac{h}{2} \\ v_{k+1} = v_k - \omega^2\left(y_k + y_{k+1}\right) * \dfrac{h}{2} \end{cases}$$

Aplicação do método de Crank-Nicolson ao OHS

Tal com se referiu, o sistema pode ser resolvido em MATLAB, com a função **LINSOLVE.** Há que escrever o sistema de forma adequada!

Na forma matricial, tem-se:

$$A\boldsymbol{Z} = \boldsymbol{b}$$

Isolando à esquerda os termos de indíce *k+1* e à direita os termos de indíce *k*, obtém-se:

$$\begin{cases} y_{k+1} + \left(-\frac{h}{2}\right) * v_{k+1} = y_k + \frac{h}{2} * v_k \\ \omega^2 * \frac{h}{2} * y_{k+1} + v_{k+1} = v_k - \omega^2 * \frac{h}{2} * y_k \end{cases}$$

Assim, a matriz constante A é dada por

$$A = \begin{pmatrix} 1 & -\frac{h}{2} \\ \omega^2 \frac{h}{2} & 1 \end{pmatrix}$$

Enquanto que o vetor $\boldsymbol{Z}$, a solução do sistema que queremos calcular, e o vetor $\boldsymbol{b}$, de termos independentes que varia de iteração para iteração, são dados por,

$$\boldsymbol{Z} = \begin{pmatrix} y_{k+1} \\ v_{k+1} \end{pmatrix} , \qquad \boldsymbol{b} = \begin{pmatrix} y_k + \frac{h}{2} * v_k \\ v_k - \omega^2 * \frac{h}{2} * y_k \end{pmatrix}$$

OBS1: ***b*** *depende da solução da iteração anterior, já conhecida na presente iteração.*

OBS2: *Os procedimentos que foram usados na aplicação destes dois métodos implícitos ao OHS, deixam, em geral de ser exequíveis quando a equação diferencial* ***não é linear****. Uma abordagem diferente será necessária nesses casos.*