Física Computacional

2022/2023

Universidade de Aveiro

Departamento de Física

Folha de Revisões 1

Problema FR1.1: Oscilador quártico — Método de Crank–Nicolson

a) Vamos voltar a estudar o oscilador quártico do Problema 2.3 (Problema 3 do Trabalho 2).
As propriedades são: massa m = 1 kg, K = 1 N/m e energia potencial

$$V(x) = \frac{1}{2}\,K\,x^2(1+\alpha x^2)$$

Com $\alpha = -0.1\ m^{-1}$. A força restauradora é

$$F_x(x) = -K\,x(1+2\alpha x^2)$$

As equações diferenciais de primeira ordem são:

$$\begin{cases} \dfrac{dx}{dt} = v_x \\ \dfrac{dv_x}{dt} = -\dfrac{K}{m}(x+2\alpha x^3) \end{cases}$$

Como a segunda equação é não linear, a aplicação do método exige uma abordagem diferente da que foi usada, por exemplo, na alínea d) do Problema 2.1.
Aplicando o método de Crank-Nicolson ao sistema, obtém-se,

$$\begin{cases} x_{k+1} = x_k + (\,v_k + v_{k+1})\dfrac{h}{2} \\ v_{k+1} = v_k - \dfrac{Kh}{2m}[x_k + x_{k+1} + 2\alpha({x_k}^3 + {x_{k+1}}^3)]\dfrac{h}{2} \end{cases} \Leftrightarrow$$

Estas equações podem ser escritas como:

$$\begin{cases} x_{k+1} - x_k - (\,v_k + v_{k+1})\dfrac{h}{2} \;\; = 0 \\ v_{k+1} - v_k + \dfrac{Kh}{2m}[x_k + x_{k+1} + 2\alpha({x_k}^3 + {x_{k+1}}^3)]\dfrac{h}{2} \;\; = 0 \end{cases}$$

Neste problema, em cada passo de Crank–Nicolson, vamos usar um método numérico fornecido pelo MATLAB para determinar os valores de x_{k+1} e $v_{x,k+1}$ que são raízes do sistema escrito acima.
(Vamos usar a função **`fsolve`** que permite resolver um sistema de equações não lineares, do tipo `F(x)= 0`. Note que no nosso caso temos duas 2 equações, `F(1) = 0 e F(2) = 0`).

<u>Como implementar em MATLAB?</u>

No corpo do programa, pode começar por definir um vetor de constantes
e as opções para a função **`fsolve`**,

```
const = [h/2, K*h/(2*m), 2*alfa];
options = optimset('Display','off','Tolx',1e-10,'TolFun',1e-10);
```

O ciclo do método de Crank–Nicolson escreve-se,

```
for k=1:N-1
      func = @(xv) fcr(xv,x(k),vx(k),const);

      xv0 = [x(k),vx(k)];

      aux = fsolve(func,xv0,options);

      x(k+1) = aux(1);

      vx(k+1) = aux(2);
end
```

Começa-se por definir uma função anónima, baseada numa função escrita num outro ficheiro.

Quando se chama a função **`fsolve`**, indica-se `xv0 = [x(k),  vx(k)]` como os valores perto dos quais deverão ser encontradas as soluções.

Falta é claro, escrever o ficheiro **`fcr.m`** !!

Complete o seguinte, escrevendo as expressões F(1) e F(2) que têm que ser iguais a zero, de acordo com o sistema.

```
function F = fcr(xv,xold,vold,const)

    % const(1), const(2) e const(3) estão definidas no programa
    principal.

    % xold é x(k) e vold é vx(k).

    % xv(1) é x(k+1) e xv(2) é vx(k+1).

    F(1)=_______;

    F(2)=_______;

end
```

Usando as condições iniciais x_0= 1m vx_0 = 1 m/s, faça os gráficos de *x* e *vx* em função do tempo e também de *vx* em função de *x*.

b) Obtenha a amplitude e o período da oscilação.

Problema FR1.2: <u>Oscilador quártico — Métodos de Runge-Kutta</u>

Repita a alínea **a)** do problema anterior usando o método de Runge–Kutta de 3º ordem representado na seguinte tabela de Butcher .

0			
1/2	1/2		
3/4	0	3/4	
	2/9	1/3	4/9

Calcule a energia total para h = 0.01, e compare com o valor obtido pelo método de Euler-Cromer (Problema 2.3 do Trabalho 2).

Problema FR1.3: <u>Oscilador quártico — ode45</u>

Repita a alínea **a)** do problema anterior usando a `ode45` do MATLAB. Compare o valor obtido para a energia total, e compare com os resultado obtido pelométodo de Runge -Kutta de 3ª ordem (FR1.2)