

Trabalho 2 - Oscilador não-linear

Método de *Shooting* e Transformadas de Fourier

Física Computacional

João Inácio, 93039, PL7

27/05/2020

1 Sumário

O objetivo principal deste trabalho é estudar o movimento de um oscilador não-linear com recurso ao método da secante aplicado ao *shooting* e a transformadas de Fourier (FT, do inglês *Fourier Transform*). Na parte A do trabalho o objetivo é encontrar o parâmetro μ através do primeiro método e na parte B estudar a aceleração com recurso ao segundo método, FT. Usou-se o método `ode45()` para fazer todas as integrações numéricas neste trabalho. No desenvolvimento do trabalho houve concordância entre os resultados esperados e os obtidos de cada um dos métodos, excepto no final do trabalho que as acelerações calculadas com recurso ao método `fft()` têm erros de aproximação.

Nota: Todos os gráficos e resultados foram obtidos através dos *scripts* de MatLab presentes em anexo com este relatório. Alguma explicação dos códigos pode ser encontrada em comentário.

2 Introdução

A equação diferencial de movimento do oscilador não-linear é a seguinte

$$m \frac{d^2 y}{dt^2} + K(y + \beta y^3) = \mu \left[1 - \left(\frac{dy}{dt} \right)^2 \right] \frac{dy}{dt} \quad (1)$$

Para esta poder ser resolvida numericamente pela rotina `ode45()` do MatLab, é preciso dividi-la em duas equações diferenciais ordinárias (ODE, do inglês *Ordinary Differential Equation*) de primeira ordem. Assim se $\frac{dy}{dt} = v$, temos as equações (2) e (3)

$$\frac{dy}{dt} = v \quad (2)$$

$$\frac{dv}{dt} = \frac{\mu}{m} \left[1 - (v)^2 \right] v - \frac{K}{m} (y + \beta y^3) \quad (3)$$

que são duas ODE's de primeira ordem e podem ser usadas pela rotina `ode45()` para a integração numérica.

Nota: Resposta à alínea a).

2.1 Método de *Shooting*

O método de *shooting* é um método numérico que resolve problemas de valor de fronteira através da redução do sistema a um problema de valores iniciais. Este método também é útil na eventualidade de ser necessário encontrar um parâmetro desconhecido, por exemplo um coeficiente da ODE, sabendo que um dos valores de fronteira tem de ser de tomar um certo valor.

Para tal efeito, é preciso, em primeiro lugar, arbitrar duas estimativas, `guess(1)` e `guess(2)`, para o valor inicial ou o parâmetro que é pretendido encontrar. Estas duas primeiras suposições têm de ser próximas uma da outra no contexto do problema.

Em segundo lugar é necessário assumir que o valor a ser encontrado toma a forma de `guess(i)`, a primeira das estimativas, e integrar a expressão numericamente, neste caso foi usado a rotina `ode45()`.

Por fim, na próxima iteração, é preciso verificar se o resultado proveniente da integração, `result(i)`, ou seja, o valor de fronteira, satisfaz o valor pretendido e calcular uma nova estimativa, `guess(i + 1)`. Este calculo faz-se de acordo com o método da secante

$$m = \frac{\text{result}(i) - \text{result}(i - 1)}{\text{guess}(i) - \text{guess}(i - 1)} \quad (4)$$

$$b = \text{result}(i) - m.\text{guess}(i) \quad (5)$$

De seguida podemos calcular a nova estimativa por

$$\text{guess}(i + 1) = \text{guess}(i) + \frac{B - \text{result}(i)}{m} \quad (6)$$

onde B é o valor de fronteira pretendido. Este processo de estimativas só para quando o valor absoluto da diferença de duas suposições seguidas for menor do que o erro especificado, `tol`.

2.2 Método da Secante

Dadas duas estimativas, x_1 e x_2 , para o zeros de uma função $f(x)$, o método da secante encontra x_0 , tal que $f(x_0) = 0$. No caso a estudar, as duas estimativas são `guess(1)` e `guess(2)`, o valor que queremos encontrar, x_0 , é o valor aproximado da condição inicial ou do parametro e a função f é `result` em função de `guess`. Contudo, neste caso não se pretende encontrar $f(x) = 0$, mas sim $f(x) = B$, ou seja, o valor de `guess` quando `result` é B . Para uma função $f(x)$ genérica a expressão de x_{i+1} é

$$x_{i+1} = x_i + \frac{B - f(x_i)}{m} \quad (7)$$

onde m é o declive da secante ao gráfico de f e que passa em x_{i+1} e x_i e pode ser calculado por

$$\text{tg}(\alpha) = m = \frac{f(x_{i+1}) - f(x_i)}{x_{i+1} - x_i} \quad (8)$$

As equações para o problema em questão são as equações (4) e (6).

Nota: Parte da resposta à alínea d).

3 Métodos e Resultados

3.1 Parte A - Método de *Shooting*

Para implementar o método de *shooting* foram apresentadas duas estimativas para o valor de μ , $\text{guess}(1) = 1.4$ e $\text{guess}(2) = 1.6$ já que o valor real de μ é aproximadamente 1.5. De seguida, fez-se um ciclo `for` para poder ser feito o *shooting* i vezes até o parametro ter um erro menor que $\text{tol} = 10^{-12}$. Foi integrada a ODE numericamente com recurso à rotina `ode45()` e encontrados os máximos de $y(t)$ para o dado parametro de μ .

Para encontrar tais máximos usou-se a função `lagr.m`, que dado 3 pontos em que o ponto médio é o máximo numérico, encontra o ponto máximo da função contínua. Todos os pontos máximos da função foram colocados num *array*, A , bem como os índices dos respetivos valores, num *array* idx . Para obter uma melhor aproximação, foi calculada uma média aritmética de todos os máximos de $y(t)$ para ser comparado com o resultado esperado, $B = y_{\max} = 1.3$, ou seja, $\text{result}(i) = \text{média}(A)$.

Finalmente, é calculado o m e $\text{guess}(i + 1)$, com recurso ao método da secante (secção 2.2), equações (4) e (6), respetivamente. Isto é feito até que, como já foi mencionado, o módulo da diferença entre duas estimativas seguidas, ou seja, $|\text{guess}(i + 1) - \text{guess}(i)|$, tem de ser menor do que $\text{tol} = 10^{-12}$.

Para calcular o período de oscilação, o *array* idx foi percorrido e com o uso da função `lagr.m` foi calculado o tempo de cada um dos máximos e foi armazenada num vetor T a diferença do tempo entre dois máximos consecutivos. Por fim, o período de oscilação é a média de todos os períodos.

Nota: Estes parágrafos anteriores são a segunda parte da resposta para a alínea d).

Deste modo, os valores de μ e do período, T , para uma amplitude positiva de 1.3, são $\mu = 1,88$ e $T = 6,52s$. Também se pode ver nos gráficos 1 e 2, figura ao lado, o deslocamento em função do tempo e o espaço de fases para este oscilador não-linear.

Através de ambos os gráficos, verifica-se que o movimento do oscilador é periódico, com o período descrito a cima. Podemos ver também no gráfico 2 que ao longo do tempo, depois do movimento se estabilizar, o espaço de fases converge para um atrator de forma elíptica, figura 2, de centro $(0,0)$, como é indicado pelo ponto azul.

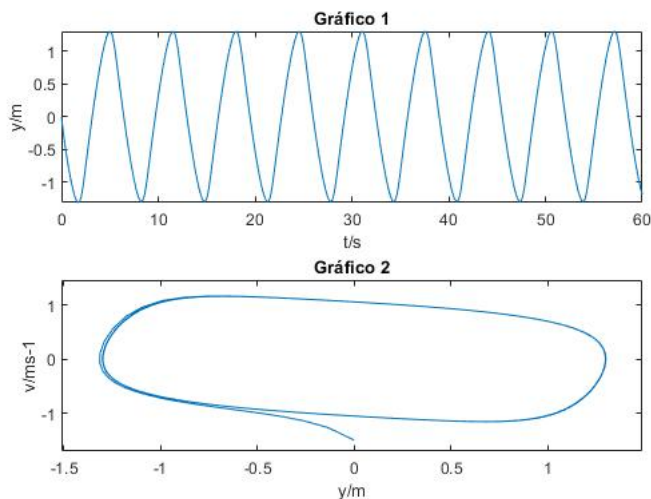


Figura 1: gráfico 1 representa o movimento do oscilador em função do tempo, $y(t)$, o gráfico 2 representa o espaço de fases do oscilador.

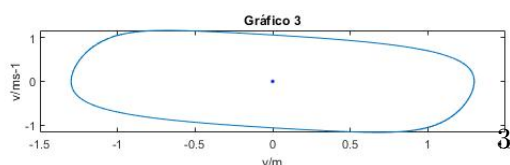


Figura 2: gráfico 3 representa o atrator do espaço de fases do oscilador não-linear, o ponto azul representa o centro do atrator, $(0,0)$.

Para fazer o gráfico 3, foram utilizados apenas os últimos 300 pontos de y e v , uma vez que para pontos

de t pequeno, o oscilador ainda estava a estabilizar o seu movimento harmónico.

Em relação à parte A do trabalho podemos concluir que para o parametro μ determinado, $\mu = 1,88$, pelo método de *shooting* temos um movimento periódico, com $T = 6,52s$, que apesar de haver uma força dissipadora, coeficiente de v , há também uma força restauradora, βy^3 , que faz com que não haja perdas de energia, tal como é mostrado no espaço de fases, ou seja, para $\mu = 1,88$, há um equilíbrio nas forças, $F_{restauradora} = F_{dissipadora}$.

Nota: Estes parágrafos a cima respondem às questões b) e c).

3.2 Parte B - Transformadas de Fourier

Para estudar o movimento do oscilador com o uso de transformadas de Fourier, é usado o método `fft()`, uma função do MatLab que usa o algoritmo *fast fourier transform*. Este método não calcula FT, $F(\omega)$, por si, calcula sim a transformada discreta de Fourier (DFT, do inglês *Discrete Fourier Transform*), $F(k)$, e a inversa usando `ifft()`. FT e DFT estão relacionadas da seguinte forma

$$F(\omega) = \Delta t F(k) \quad (9)$$

A densidade espectral pode ser calculada da seguinte forma

$$Densidade\ Espectral = |F(\omega)|^2 \quad (10)$$

No caso da derivação da função TF, a relação é a seguinte equação

$$\frac{d^n f(t)}{dt^n} = (i\omega)^n F(\omega) \quad (11)$$

A transformada de Fourier passa uma função de domínio t para uma função de domínio ω . O algoritmo `fft()` calcula $F(k)$, de uma maneira eficiente e rápida, dado o nome, visto que tem uma complexidade computacional de $\mathcal{O}(N \log N)$, N é o número de elementos do *array* de posições.

Para implementar o código desta questão, em primeiro lugar, usando um vetor de tempo com comprimento igual a 1024, N , e com um passo $dt = 0.1$, foi criado um vetor de frequências angulares ω para o domínio de TF. As fórmulas para calcular os extremos e passo desse vetor são

$$d\omega = \frac{2\pi}{Ndt} \quad (12)$$

$$\omega_{max} = \left(\frac{N}{2} - 1\right) d\omega \quad (13)$$

$$\omega_{min} = -\frac{N}{2} d\omega \quad (14)$$

Com as últimas três equações, criou-se o vetor de frequências. De seguida, foram resolvidas numericamente as equações (2) e (3), pela rotina `ode45()`.

Seguidamente, foi usado a função `fft()` para determinar a DFT, `fftshift()` para centrar os valores da DFT e, por fim, isto foi multiplicado por dt para obter $F(\omega)$, como dita a relação (9). Foi calcula da densidade espectral de acordo com a equação (10) e daí resultou a seguinte figura,

Nota: Estes parágrafos a cima correspondem a metade da resposta à questão g).

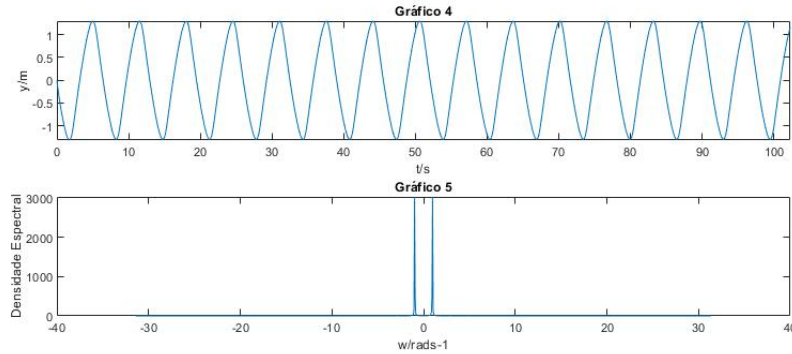


Figura 3: gráfico 4 representa o deslocamento, y , do oscilador em função do tempo, o gráfico 5 representa a densidade espectral em função do domínio das frequências.

Pela observação do gráfico 5, podemos concluir, mais uma vez, que o movimento do oscilador é de facto periódico, já que no domínio de frequências, a função apresenta um pico à volta de $\omega = 1$, e o seu simétrico em $\omega = -1$. Ou seja, o sistema oscila apenas com uma e uma só frequência. O calculo do período associado a esta frequência vai ser calculado mais a baixo para comparar com o valor encontrado na secção 3.1.

Nota: Este parágrafo a cima corresponde à resposta da alínea e).

Para poder ser mais fácil a comparação da frequência máxima, temos de ampliar o gráfico, ou seja, diminuir $d\omega$ por um fator de 4. Para tal, multiplicou-se o valor de dt por 4, já que segundo a fórmula (12) $d\omega$ é proporcional a $\frac{1}{dt}$. Com esta alteração foi calculado outra vez um vetor de frequências e feito e a densidade espectral da mesma forma descrita a cima.

Como os pontos máximos da densidade espectral estão próximos de 3000, é pedido para considerar pontos cuja densidade espectral é menor do que 10. Assim, foi usado uma sintaxe do MatLab que permite escolher os índices dos pontos de um *array* que têm um determinado valor com o intuito de ir buscar os valores associados a esses índices noutro *array*. Exemplo:

Supor que se tem os seguintes vetores $a = [1, 2, 4, 8]$, $b = [4, 8, 10, 11]$ e c , definido por $c = a(b < 9)$. A operação $(b < 9)$ vai buscar os índices dos valores de b que são maiores do que 9, logo $(b < 9) = [3, 4]$. Desde modo, c são os valores de $a(3)$ e $a(4)$, $c = [4, 8]$.

Este método foi usado para encontrar as frequências cuja densidade espectral associada é menor do que 10. Foi feito um `plot()` destas frequências e densidades espectrais, assim temos a figura 4.

Nota: Estes parágrafos correspondem ao resto da questão g).

Analisando ambos os gráficos verificar-se que já não há só uma frequência que sobressalta, em vez disso temos 4. Isto pode ser devido a erros de arredondamento no método `fft()`, já

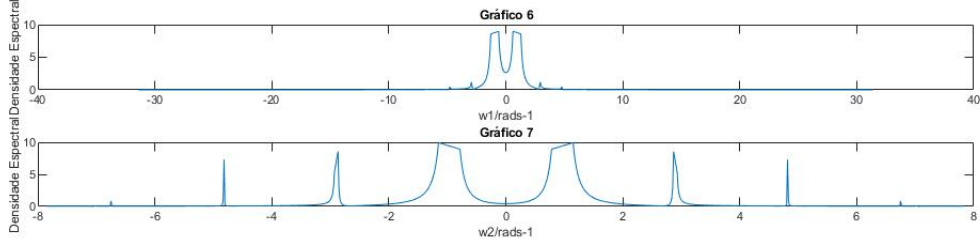


Figura 4: gráficos 6 e 7 correspondem às densidades espectrais menores que 10 em função das frequências. O gráfico 7 tem uma resolução 4 vezes maior.

que estas frequências estão numa densidade espectral de 10, enquanto as outras numa ordem de 3000. Se for calculado o período ($T = \frac{2\pi}{\omega}$) associado ao primeiro e segundo pico no gráfico ($\omega = 0.7823$ e $\omega = 1.15$), obtemos um intervalo $[5.46, 8.03]$ em que o período determinado em 3.1 ($T = 6.52$) pertence. Ou seja, o movimento é sim periódico, os gráficos apenas têm erros de arredondamento devido algoritmo `fft()`.

Nota: Este parágrafo corresponde à resposta à questão f).

Por fim, foi calculada a velocidade através da TF usando a relação das derivadas, equação (11) e usando y e v determinados pela `ode45()`, que foi calculado através da equação (1), já que $a = \frac{d^2y}{dt^2}$. Resultou a figura 5.

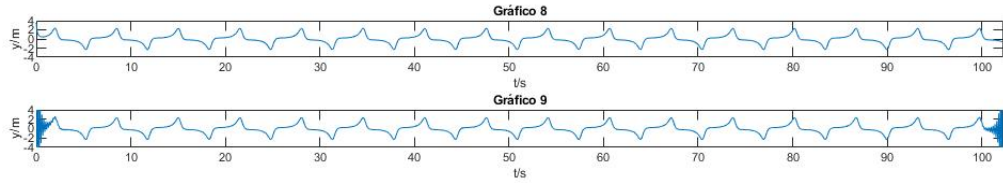


Figura 5: gráficos 8 e 9 representam a aceleração em função do tempo. O gráfico 8 foi feito através dos cálculos da `ode45()` e o 9 através da derivada de FT.

Ambos os gráficos estão muito parecidos o que indica que quer pela `ode45()` que por FT se pode calcular a aceleração de uma forma correta, contudo para os extremos de tempos, o gráfico 9 tem ruído, o que indica que há pequenos erros de arredondamentos e aproximações no que toca a fazer a transformada de Fourier inversa.

Nota: Estes dois parágrafos correspondem à resposta à questão h).

4 Discussão e Conclusão

A maioria dos resultados foram precisos, com excepção dos dois últimos gráficos da parte B. Esses erros podem estar associados a erros de truncatura e arredondamento dentro do algoritmo *fast fourier transform* que o MatLab usa.

Em suma, foi estudado o movimento do oscilador não-linear com recurso ao método de *shooting* e `fft()`, foi determinado o valor de μ quando a amplitude máxima positiva é 1.3, o valor do período e a aceleração de duas formas, chegando à conclusão que se trata de um oscilador periódico para $\mu = 1.88$.