Computational Physics

2º Trabalho Prático de Avaliação Continua

Shooting Method and Fast Fourier Transform

João Maria Machado ,89132, PL4

May 28, 2020

Contents

1 Summary

In this report it will be made a computational analysis regarding a non-linear harmonic oscillator using a shooting method and the fast Fourier transform routine native to Matlab

After explaining the thought process behind the produced code an analysis over the obtained results will be made in order to verify that **there is accordance in between the different methods when both are applied to the same situation**.

2 Introduction

The studied physical system can be described by the following differential equation:

$$m\frac{d^2y}{dt^2} + K(y+\alpha y^2) = \mu\left[sen\left(\frac{dy}{dt}\right)\right]\frac{dy}{dt} \tag{1}$$

As we can see the equation clearly describes a non-linear harmonic oscillator. More over the equation can be rearranged in order to make it easier to integrate.

$$\frac{d^2y}{dt^2} = \frac{-K(y+\alpha y^2) + \mu\left[sen\left(\frac{dy}{dt}\right)\right]\frac{dy}{dt}}{m} \tag{2}$$

This ordinary differential equation can be integrated using the ODE45 routine. However integrating is not enough! After the integration process the physical value of the amplitude must be attested (because it can be incorrect), and the integration process must be remade and adjusted accordingly. It is a boundary value problem and it will be solved using the **Shooting method**. In this case the shooting will be made in regards to μ using two initial guesses (suggested in the test sheet) afterwards, the code will produce an **educated guess** and verify if the result produced by the guesses meet the established tolerance.

3 Methods and Results

The first step done in order to start writing the code was to write the equations that describe the physical system in a way that can be easily written into Matlab.In the previous section there is already an equation 2 that describes the oscillator's acceleration. After there is only need to substitute the derivatives by their physical meaning ($\frac{dy}{dt} = v$ and $\frac{d^2y}{dt^2} = \frac{dy}{dt}$, where v stands for velocity). **This system is the answer to question a) from part 1**

$$\begin{cases} \frac{dv}{dt} = \frac{-K(y+\alpha y^2)+\mu\left[sen\left(\frac{dy}{dt}\right)\right]\frac{dy}{dt}}{m} \\ \frac{dy}{dt} = v \end{cases} \tag{3}$$

After taking the necessary mathematical considerations the code can now be written. It is important to note that the main purpose of the code is not to only to integrate but to integrate while respecting the boundary conditions. As stated before, the code will "shoot" in respect to μ in order to satisfy the condition that the value of the amplitude is 2.15.

After implementing the code the following vector of values of μ was obtained (**the following row matrix is the answer to question b) from part 1**):

$$\begin{bmatrix} 0.3500 & 0.4000 & 0.5074 & 0.4957 & 0.4965 & 0.4965 & 0.4965 \end{bmatrix}$$

The last value is the value of μ that made the amplitude converge to the expected value. It should be noted that the value of μ as the number of integrations increases approaches **0.5**

The next step in order to study the system is to analyse the trajectory in the phase space and find the period of the studied oscillator. The following graph, was obtained (the following set of figures contains the results found in order to **answer question c) from part 1**) :

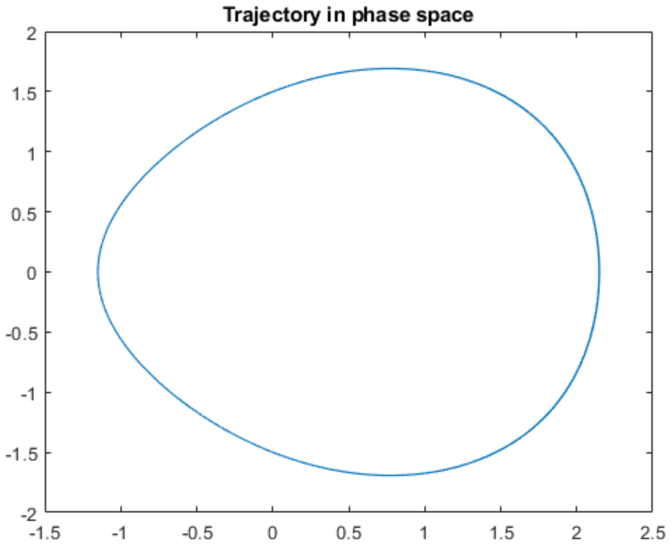

Figure 1: Graph of Trajectory in phase space, as we can see the graph does not resemble an harmonic oscillator, due to the elliptical shape. The energy is being conserved

The value found for the period is **6.3619** (the value has no unit because no units are mentioned in the test sheet). This value can be confirmed for the integration done while analysing the graph of position over time:

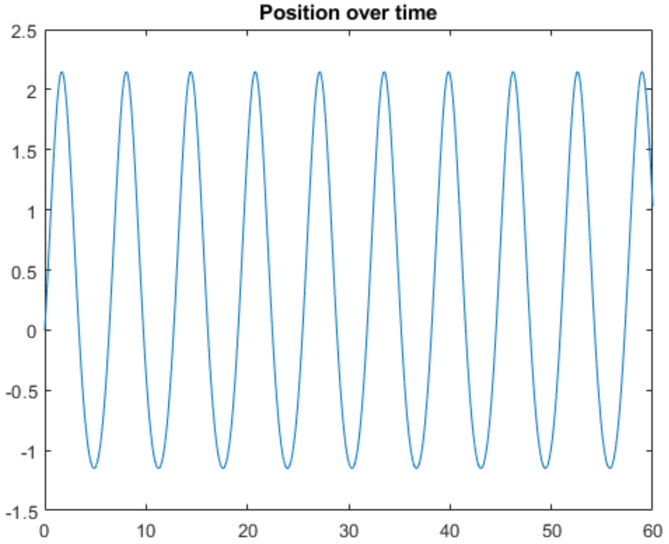

Figure 2: Graph of position over time, after taking a close look, it can be confirmed that value of period is close to the one computed previously, the graph confirms that the algorithm responsible for computing the period is correct

Finally, the first part shall be concluded with an in depth analysis of the particular shooting method that was used. **The shooting and secant method** (**This part will answer question d) of part 1)**. As stated before the shooting method is used to ensure the accuracy of the integration and that it respects the boundary conditions. The method starts with 2 guesses given by the the programmer, afterwards it starts the integration process, when the first integration is completed it proceeds to access how accurate it was, this can be easily explained by the following expression:

$$|last.value.computed - boundary.value| < tolerance$$

where "**last.value.computed**" is the last computed value for a list of values, "**boundary.value**" is the of the boundary condition, the goal per say and "**tolerance**" is how the two compared values can differ.

If the comparison step is successful the shooting is stopped, however if the estimate does not meet the boundary condition with the established tolerance the method restarts with **1 new guess and reuses one of the latest pair**. Unlike the first set these 2 new guesses are not random, one is reused and the other one can be calculated using the following formula:

$$guess(3) = guess(2) + \frac{B - result(2)}{m}$$

where B is the coordinated at origin and **m** is the slope of a particular linear equation. The slope can be calculated just like any slope but using the previous guesses and the results originated by them, as shown in the following expression:

$$m = \frac{result(2) - result(1)}{guess(2) - guess(1)}$$

the **B** parameter can be computed by the following expression:

$$B = result(2) - m \times guess(2)$$

After producing the 2 new educated guesses the program proceeds to check if the guesses produce an adequate result if so the program stops, otherwise the program does the algorithm explained before again.

With these advancements made the analysis can now move on, by studying the system with regards to **Fourier analysis**. The first step in order to study the system is to produce the array of values of position and its length. Afterwards the vector of frequencies is ready to be calculated. The following formulas are used to allocate the vector of frequencies.

$$\Delta\omega = \frac{2\pi}{N\Delta t}$$

$$\omega = \Delta\omega * [0 : N - 1]$$

Where N is the length (cardinality) of the array of the positions computed previously. $\Delta\omega$ is the step of the frequencies array and Δt is the time step. After successfully calculating the frequencies the program is now ready to initiate the **FFT** routines. The following graph was obtained (**The next graph contains the answer the question e) and explain question g) of part 2**):

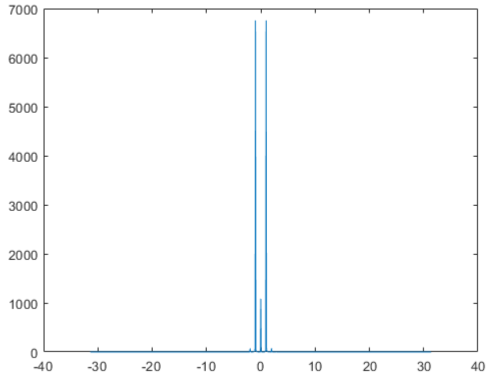

Figure 3: Fast Fourier transform (centred at 0) in which we can see the spectral density over frequencies, the graph suggests that the movement seen is **not typical of an harmonic oscillator, because it has more than two natural frequencies (peaks in the graph)**.

The next step is now to take the obtained results and select the values of spectral density and frequency bellow 10 and amplify them by a factor of 4. The produced graphs are the following (**this contains the answer to question f) from part 2**) :

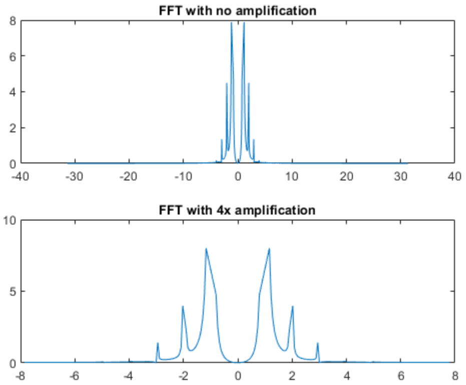

Figure 4: Comparison between different FFT's with different amplification, as the factor of amplification increases the peaks grow larger and more evident, **the oscillator is still not harmonic. There are more than two natural frequencies.**

One thing to note is the way that the spectral density is computed. The theoretical formula is simple (**the next equation is meant to respond to question g) from part 2**):

$$dens = (\Delta t \times |FFT|)^2$$

Finally the last step is to check if the Fourier analysis is correct and apply it to the same context as before. A comparison will be made between the acceleration over time computed by two different methods. Via ODE45 and shooting method and the FFT route. However before presenting the results, the method used to derive the Fourier transform must be cited. The Fourier transform has a special rule when it comes to deriving it:

$$F\left[\frac{d^2f(t)}{dt^2}\right] = (i\omega)^2 F(\omega)$$

The logic of the question is simple, the current Fourier transform is made using the position over time if the transform is derived twice and then applied the IFFT (Inverse Fourier Transform) routine the following graphs can be obtained (**the following graph is meant to respond to question h) from part 2**):

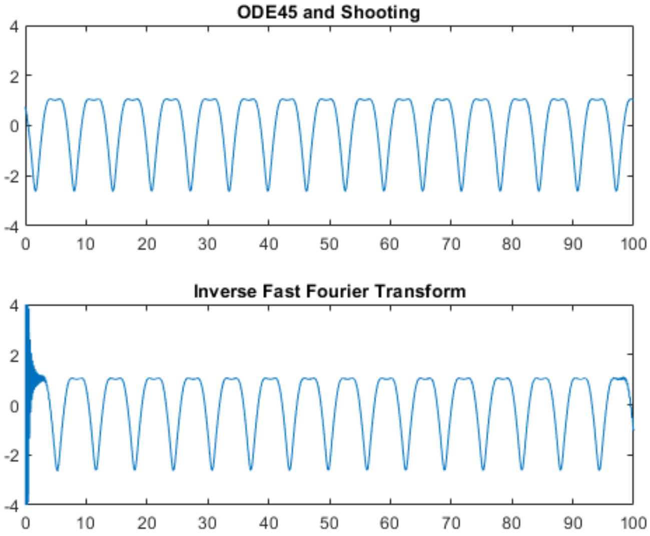

Figure 5: Comparison between the two graphs of acceleration over time, the different methods agree with each other however the FFT method presents some noise near t=0

4 Discussion and Conclusion

The obtained results were precise and were coherent in their physical interpretation, there does not seem to appear any significant error, however minor errors can be lowered in the program if the time step is decreased thus making the integration more precise and making the sampling larger (in the case of the FFT).

As stated before the results are coherent with the physics thus one could say that the goal was achieved.