

Teaching classical mechanics using an applied example: Modelling and Software

M.A. García-March*, J.M. Isidro, M. Zacarés, M. Arevalillo,
González-Santander, Ll. Monreal
UNIVERSITAT POLITÈCNICA DE VALÈNCIA
* migarmal@upvnet.upv.es

C.I. López-Javier
INSTITUTO DE EDUCACIÓN SECUNDARIA JOSEP MARIA LLOMPART

Abstract

Presentamos la experiencia docente obtenida al utilizar un mecanismo muy común en ingeniería, como el mecanismo biela-manivela-deslizadera, como un ejemplo práctico en la asignatura "mecánica clásica" que se imparte en los grados de física e ingeniería. También presentamos una interfaz gráfica que permite al estudiante visualizar los resultados obtenidos al simular en el ordenador el movimiento de dicho mecanismo variando las condiciones en las que se realiza el experimento numérico.

We present our experience in using a common mechanism in engineering, such as a slider-crank, to explain classical mechanics in physics and engineering degrees. We also present a graphical user interface that permits the student to visualize the results of different simulations varying the conditions of the numerical experiment.

Keywords: *Graphical User Interface, Classical Mechanics, Slider-crank mechanism, multibody system*

1 Introduction

Classical mechanics is a core course in most physics and engineering degrees. It involves abstract concepts such as Action, Hamiltonian, or Lagrangian, and others much closer to the final applications, i.e., Energy, Force, Momentum, etc. [1, 2] These are usually explained in the context of easy-to-understand systems, such as the harmonic oscillator, or a parabolic projectile, instead of using other mechanisms commonly found in Engineering.

In this paper, we present our experience in using the slider-crank mechanism to illustrate classical mechanics concepts to students of the subject "Mecánica Clásica" of the physics degree at the University of Valencia (Spain). This subject is taught in the fourth year of the curricula of the the physics Degree in the Spanish Universities. It used to exist as an elective subject in the curricula of Industrial Engineering at the Polytechnical University of Valencia. The material introduced in this paper was designed for this subject but, unfortunately, it disappeared from the Engineering curricula. Hence, the software and the method is suitable to be applied in both in physics and Engineering degrees, whenever a subject that includes concepts of classical mechanics still exists. Since this subject does not exist any longer in the Engineering degrees of Valencian Universities, the experiments were carried out in a physics degree.

The slider-crank mechanism transforms a linear movement into a circular one or vice versa, and it is present in pumps, compressors, alternative motors, etc. The lesson includes three steps. Firstly, the professor explains the mathematical model of the mechanism. Secondly, he presents the numerical method and the software that solves the equations of motion numerically. Finally, the student can use a graphical interface to visualize the results of different simulations varying the conditions of the numerical experiment. The mathematical model and the software that solves the corresponding equations of motion numerically were presented in [3]. Here we also include a graphical user interface that permits the student to visualize the results varying the conditions of the numerical experiment. With these tools, the students can i) understand the theory using a mechanism that they will find in common applications and verify the utility of the abstract concepts; ii) obtain and solve the equations of motion using the numerical code; iii) use a graphical user interface to observe the evolution along the time axis of a number of variables, such as the Energy or the Momentum, and iv) study the behaviour of the system under different conditions, i.e. different initial conditions, different values of the coefficient of friction, etc.. Besides, if the students are able to program in MATLAB, they can also understand, or even code, the part of the application that solves the equations of motion. The structure of the application allows the students to understand the necessity of an abstract theory and all the steps in the resolution of the problem, that is, modelling, programming, and analysis of results.

2 The Slider-Crank Mechanism

The slider-crank mechanism is a multibody system, i.e., a system of interconnected rigid bodies (see Figure 3). This system consists of three bodies, which we have denoted as bodies A, B and C. The body A experiments a rotational movement around some point, which we denote a , and it is jointed with body B through a revolute joint in point b . Finally the body C is connected with body B also through a revolute joint at point c . This body is restricted to do only translational movements along the x coordinate. Therefore, the body B acts as a connection rod between a body that only experiments rotational movements (A) and another one that only experiments translational movements (C). If the driving body is the body A,

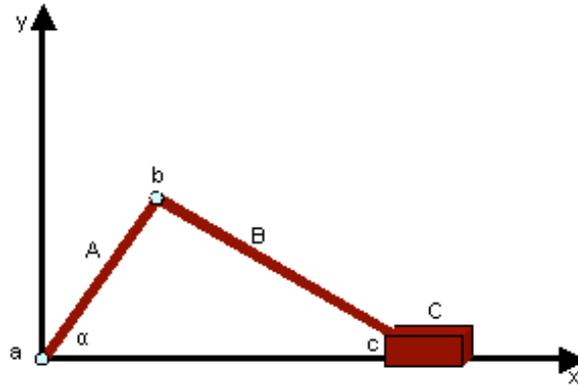


Figure 4.1: Schematical representation of a Slider-Crank mechanism

then this mechanism transforms a rotational movement into a translational one. If the driving body is the body C, then a translational movement is transformed into a rotational one.

All the movement of this system can be determined if the evolution with time of the angle $\alpha = \alpha(t)$ (see Figure 3) is known. The differential equation of motion of this system under the action of gravity and in the absence of external forces was explained in [3]. Let us summarize how to obtain this equation. First, the Lagrangian L of the system must be written as:

$$L = T_A + T_B + T_C - V(\alpha) \quad (4.1)$$

where T_A , T_B , and T_C , are the kinetic Energies of the bodies A, B, and C, and $V(\alpha)$ is the potential energy of the bodies A and B. The body C has not potential Energy since it is forced to remain along the x axis. All these Energies can be expressed in terms of only one spatial variables, i.e., the angle $\alpha = \alpha(t)$ and its derivative. Hence, the Lagrangian can be expressed as:

$$L(\alpha, \dot{\alpha}) = \frac{1}{2}F(\alpha)\dot{\alpha}^2 - G(\alpha) \quad (4.2)$$

where F and G are certain functions depending on α (for a detailed expression of these functions, see [3]). Using the Euler-Lagrange equations:

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{\alpha}} - \frac{\partial L}{\partial \alpha} = 0 \quad (4.3)$$

one can obtain the equation of motion:

$$\ddot{\alpha}[F(\alpha)] + \frac{1}{2}\dot{\alpha}^2 \frac{\partial F}{\partial \alpha} + G(\alpha) = 0 \quad (4.4)$$

This equation has to be solved numerically. To do that, a fourth order Runge-Kutta finite difference method has been programmed using MATLAB. Since it is a second order equation, the initial condition requires the definition of an initial position, $\alpha(0) = \alpha_0$, and an initial velocity $\omega(0) = \omega_0 = \frac{d\alpha}{dt}$. Finally, it has also been considered the possibility of the existence of a friction force of the form $R(\omega) = \frac{1}{2}\gamma\omega^2$ parametrized by a Rayleigh coefficient γ . This term is taken into account in the Euler-Lagrange equation as follows:

$$\frac{d}{dt} \frac{\partial T}{\partial \dot{\alpha}} - \frac{\partial T}{\partial \alpha} + \frac{\partial V}{\partial \alpha} + \frac{\partial R}{\partial \dot{\alpha}} = 0 \quad (4.5)$$

and, consequently, the equation of motion has to be modified according to this new term.

3 A MATLAB Graphical User Interface For Analysing The Slider-Crank Mechanism

MATLAB is a mathematical program for numerical computation that includes a visual Graphical User Interface (GUI) environment. It provides basic GUI components, such as push buttons, text fields, sliders (scroll bars), menus, check boxes, radio buttons, etc. So it is a valuable learning tool since it permits to use any MATLAB program from a graphical interface, permitting to vary parameters or functions and obtaining a graphical representation of the result [4]. We have developed a GUI that permits the student to analyse the dynamics of the slider-crank mechanism. This GUI offers four sliders that permit the control of:

- The initial position α_0 ,
- The initial velocity, ω_0 ,
- The total time of the simulation, and
- The Rayleigh coefficient γ .

The output of the GUI is the graphical representations of the evolution of α with time and the phase space, i.e., the graphical representation of the position and velocities at each time in the position-velocities space or phase space.

There are four different dynamical situations the student must recognize, i.e., the system can:

1. Oscillate in incomplete turns around the point a . In this case the position evolves in a periodic curve with time and the trajectory in the phase space is a closed curve.
2. Oscillate in incomplete turns around the point a and slow down to rest. In this case the position evolves with time in a periodic but amortigated function. In the phase space there is a spiral trajectory that converges to a rest point.
3. Spin on around point a . In this case the position evolves in a monotonic curve since the position α is a periodic variable. In the phase space it is represented as a periodic curve.
4. Spin on around point a and slow down to rest. This is a combination of the situations (2) and (3). The evolution with time of the position is monotonic but amortigated. The representation in the phase space is a periodic but amortigated curve. From certain time the mechanism does not spin around but oscillates. Then the evolution curve resembles a periodic amortigated movement and it is slowed down to rest. In the phase space the curve is closed and converges to the rest point.

The student must decide the values of the different controls and recognize one of these different situations. If the student chooses an initial position different from zero, an initial velocity equal to zero and Rayleigh coefficient equal to zero, he or she will observe the situation described in

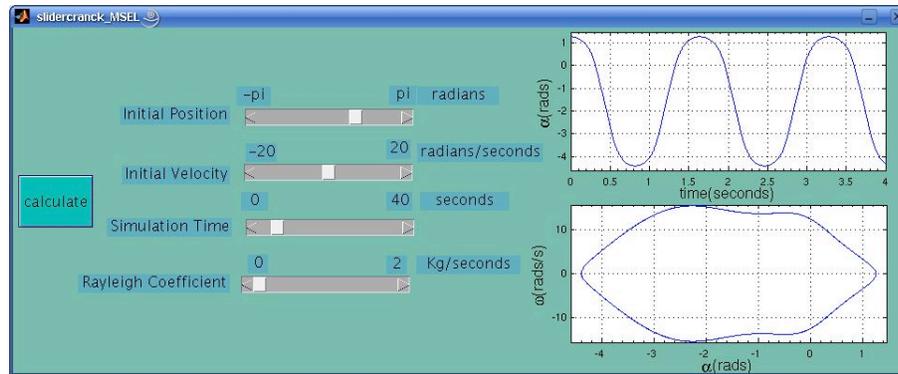


Figure 4.2: Oscillating movement

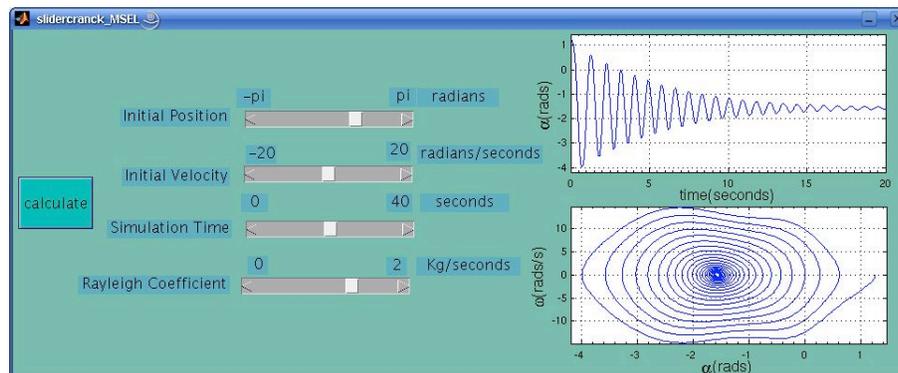


Figure 4.3: Amortigated oscillating movement.

(1). In Figure 4 we present an example where $\alpha = 1.25$ rads, $\omega_0 = 0$ rad/s, $\gamma = 0$ Kg/s and the total time of simulation is $t = 4$ s. As it can be observed, the evolution of the position with time is periodic and the trajectory in the phase space is closed.

If the student chooses an initial position different from zero, an initial velocity equal to zero and Rayleigh coefficient different from zero she will observe the situation described in (2). In Figure 4 we present an example where $\alpha = 1.25$ rads, $\omega_0 = 0$ rad/s, $\gamma = 1.4$ Kg/s and the total time of simulation is $t = 20$ s. In this case, the evolution of the position with time resembles an amortigated movement and the trajectory in the phase space tends to a rest or equilibrium point.

If the student chooses an initial position different from zero, an initial velocity different from zero and Rayleigh coefficient equal to zero she could observe either situation (1) or (3), depending on the value of ω_0 . If ω_0 is smaller than certain threshold, the initial kinetic energy will be insufficient to force the system to spin around its axis, and it will oscillate. In Figure 3 we present an example where $\alpha = 1.25$ rads, $\omega_0 = 1.6$ rad/s, $\gamma = 0$ Kg/s and the total time of simulation is $t = 4$ s. In this case, the evolution of the position with time and the trajectory in the phase space are similar to the first case. Note that at the beginning of the simulation the position grows slightly due to the initial velocity. If the initial kinetic energy is sufficient to permit the system to spin around its axis, the system will evolve as described in situation (3). In Figure 3 we present an example where $\alpha = 1.25$ rads, $\omega_0 = 4$ rad/s, $\gamma = 0$ Kg/s and $t = 4$ s. In this case the angle α grows continuously and the trajectory in the phase space is a periodic curve.

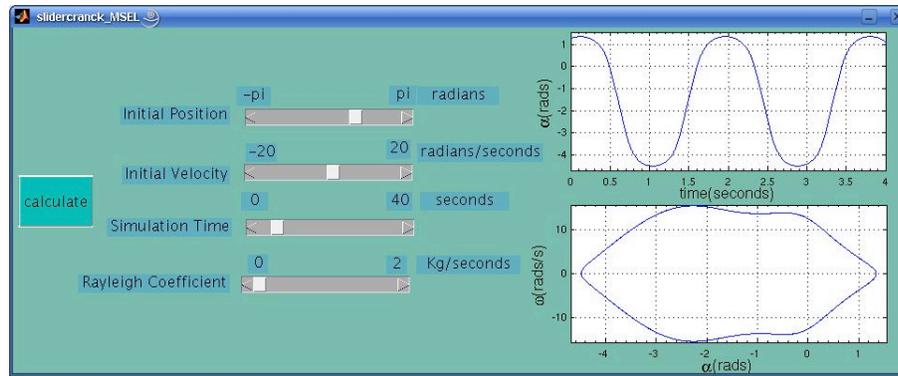


Figure 4.4: Oscillating movement with initial velocity different from zero

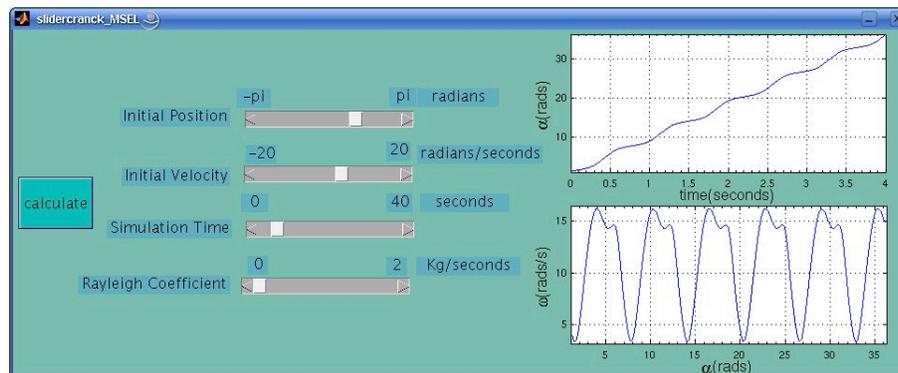


Figure 4.5: Spinning movement

Finally, if under the conditions of the last case we assume there are frictional forces, the student will observe the situation (4). In Figure 3 we present an example where $\alpha = 1.25$ rads, $\omega_0 = 4$ rad/s, $\gamma = 0.6$ Kg/s and $t = 28$ s. As it can be observed, the position grows with time until a value is reached. Then, an amortigated oscillating movement starts. This is translated into a closed curve in the phase space that tends to an equilibrium point.

4 Application For A Practical Lesson In Classroom

The previous software can be used for designing a practical lesson for the subject "Mecánica clásica" in a physical or engineering degree, whenever a subject including these concepts still exists. A practical lesson based on this software can be divided into three parts. First, the teacher explains the system model, inferring the equation of motion as described in [3]. Then, students are asked to propose different strategies to solve the equation. If the students have a sufficient level of programming, they can try to program the simulation by themselves. Otherwise, the lecturer explains the use of the software. Finally, the student uses the GUI described in this paper to analyse the dynamical behaviour of the mechanism and tries to analyse different situations. With these simulations the student can check:

- (i) that the theory is of a great utility to model complex mechanisms,
- (ii) that the numerical methods are necessary when it is not possible to obtain an analytical

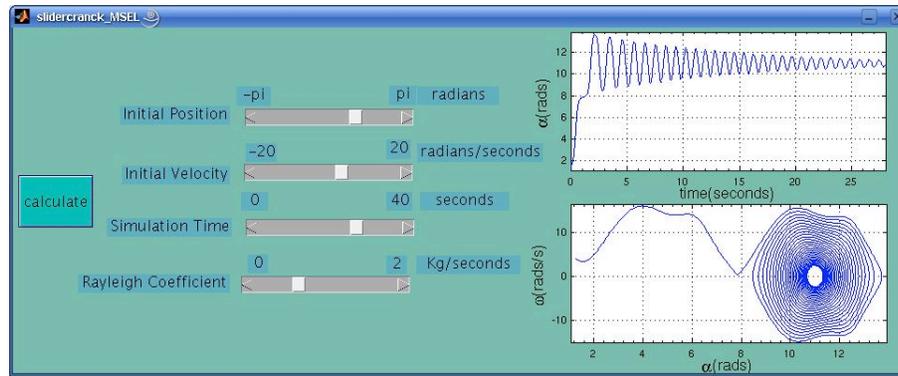


Figure 4.6: Amortigated spinning-oscillating movement

solution of the equation, and

- (iii) that although the dynamics of the system are different under different conditions they can be easily understood by analysing the evolution or the phase space curves.

This GUI has been used in the teaching of "Mecánica Clásica", a module in the degree of physics at the University of Valencia (Spain). Since it was not possible to use a computer laboratory, the professor used a data projector to show the GUI on a screen and asked the students to vary the conditions and to analyse the results. The experiment took around 30 minutes and the number of students per group were around 40 people. The experiment was carried out in two different groups, without a control group. Personal enquiries showed that students were quite satisfied with the experience. They appreciated the effort made in presenting a complex system and the graphical representation. Nevertheless, it was not possible to do a rigorous test of their opinions. By the way, some questions related to this demonstration were included in the final exam, but it was not possible to do a detailed study of the results, since we did not use a control group.

This GUI was first developed as part of the laboratory work for the subject "Mecánica Clásica" in engineering in the Engineering School of the Universitat Politècnica de València (Spain). Unfortunately, this module has disappeared from the Engineering curricula due to a recent reform of the degree contents. Hence, it was not possible to perform any experiment in an engineering degree.

5 Conclusions

Our experience in developing a software application to simulate a complex system has been presented. We have used a slider-crank mechanism, which has been modelled using classical mechanics concepts. We have presented a GUI which permits the student to do numerical experiments with this model and to understand the dynamical behaviour of the system. Finally, the possibilities of using this software in practical lessons and our experience utilizing it in the context of the module "Mecánica Clásica" in a physics degree have also been outlined. It can be used also in an Engineering degree, but it was not possible to perform any experiment in this case, since this subject has disappeared from the Engineering curricula recently. Finally, the possibility of adding external forces to the system is left for future research. This case is very interesting since it leads to a scenario of chaotical dynamics. This case can be implemented in

the method easily from the theoretical point of view, but, in such a case, the numerical method has to be designed carefully, due to the chaotic response.

6 User's Guide for programs

- **bilmarcr.m**: This is a program to be run directly from the command line of MATLAB with the sentence:

```
[alfa,omega,b,bp,E,t,p]=bilmarcr(alfazero,omegazero,t,raylcoeff);
```

where the outputs are:

1. alfa: a vector with α as a function of time at each time step.
2. omega: a vector with ω as a function of time at each time step.
3. b: a vector with β as a function of time at each time step.
4. bp: a vector with the time derivative β as a function of time at each time step.
5. E: a vector with the Energy as a function of time at each time step. Should be constant if the Rayleigh Coefficient vanishes.
6. t: a vector with each time step.
7. p: a vector with the momentum as a function of time at each step it also gives different graphical representations.

and the inputs are:

1. alfazero: a scalar α_0 .
 2. omegazero: a scalar ω_0 .
 3. t: a scalar with the total time of simulation.
 4. raylcoeff: a scalar indicating the value of the Rayleigh Coefficient.
- **slidercrank_MSEL.m** This program is the GUI presented in the paper. It is an interface to run the previous program. To run this program write "guide" in the commands line of MATLAB. Once GUIDE is running, select "open" and open the program slidercrank_MSEL.m . Then select "run".

Referencias

- [1] H.Goldstein, *Mecánica Clásica*. Ed. Reverté, Barcelona, (1987).
- [2] C.Lanczos, *The variational principles of mechanics*. Dover, New York, (1986).
- [3] M.A. García-March, I. Orquín, P. Fernández de Córdoba, A. Montero, J. Urchueguía, M.H. Giménez y J.A. Monsoriu, *Formulación Lagrangiana del Sistema Biela-Manivela-Deslizadera*. Anales de Ingeniería Mecánica. Revista de la Asociación Española de Ingeniería Mecánica. 2 1077-1086 (2004).
- [4] M. A. García-March, M. Arevalillo-Herráez, M. Zacarés, J. M. Isidro, J.L. González-Santander, and C. I. López-Javier, *The graphical user interface of matlab as an educational tool*. Proceedings of the International Technology, Education, and Development conference. INTED 2008.