

## Research Article

H. Shanak, H. Khalilia, R. Jarrar, and J. Asad\*

# Numerical study of coupled oscillator system using the classical Euler-Lagrange equations

<https://doi.org/10.2478/mme-2021-0001>

Received Nov 13, 2020; accepted Dec 08, 2020

**Abstract:** Problems involving vibrations (mechanical or electrical) can be reduced to problems of coupled oscillators. For this, we consider the motion of coupled oscillators system using Lagrangian method. The Lagrangian of the system was initially constructed, and then the Euler-Lagrange equations (*i.e.*, equations of motion of the system) have been obtained. The obtained equations of motion are a homogenous second-order equation. These equations were solved numerically using the ode45 code, which is based on Runge-Kutta method.

**Keywords:** Lagrangian, Equations of Motion, Numerical Calculation, ode45, and Runge-Kutta method

## 1 Introduction

In classical mechanics, there are many useful techniques that can be used in solving physical systems. Newtonian mechanics is a technique based on vector concepts (*i.e.*, forces). In this technique, one has to write down all forces acting on the system, and then, applying Newton's second law. Another useful technique based on scalars quantities (*i.e.*, kinetic and potential energies) has been developed. This technique, known as Lagrangian mechanics, helped physicists in solving a large number of complicated systems, where Newtonian techniques cannot be applied effectively in many areas in physics, such as classical mechanics [1, 2], nonlinear systems, and control theory [3, 4].

To apply Lagrangian mechanics, one has to write down the kinetic and potential energies of the system, and as a result, the Lagrangian  $L(\dot{q}, q, t)$  is obtained, where  $q$  represents the generalized coordinate,  $\dot{q}$  is the generalized velocity, and  $t$  is the time. Applying the following equation  $\frac{\partial L}{\partial q} - \frac{d}{dt} \frac{\partial L}{\partial \dot{q}} = 0$  to the Lagrangian  $L(\dot{q}, q, t)$ , we obtained a set of differential equations known as Euler-Lagrange equations (ELE) or equations of motion. For more details, one can refer to [1]. The ELE can be solved analytically, and thus, the motion of the system is specified. In many cases, the ELE cannot be solved analytically. Therefore, we have to seek for suitable numerical techniques to apply, and in this case, some initial conditions are needed.

On the other hand, it is important to remember that all mechanical and electrical vibration problems reduce, in the case of small oscillations, to problems involving one or several coupled oscillators. Problems involving vibrations of strings, membranes, elastic solids, electrical, or acoustical can be reduced to problems of coupled oscillators; see for examples [5, 6] and references within them.

Many numerical methods have been introduced in literature to solve differential equations [7–12]. These numerical methods are based on MATLAB, MATHEMATICA, MATCAD software and other computerized programs. For example, MATLAB contains many built-in codes that can be used in such cases (*e.g.*, ode45). These numerical techniques enable us to solve many systems numerically, even when the analytical solution is not known. In addition, the ode45 code depends on a Runge-Kutta method, which can be used efficiently in solving different kinds of differential equations. One can refer to [13, 14] and references therein to construct a good idea about applying the Runge-Kutta method on solving differential equations.

On the other hand, many ideas have been carried out by applying MATLAB and built-in codes in solving some differential equations. For example in [15–17], the authors solved some differential equations describing physical systems by applying Lagrangian mechanics and obtaining the ELEs, which are in general differential equations that need to be solved numerically for certain initial conditions.

The outline of this paper is arranged as follows: In Section 2, the physical description of the system is presented,

\*Corresponding Author: J. Asad: Department of Physics, College of Sciences, P.O. box 7, Palestine Technical University-Kadoorie-Ptuk, Tulkarm 303, Palestine; Email: j.asad@ptuk.edu.ps

H. Shanak, R. Jarrar: Department of Physics, College of Sciences, P.O. box 7, Palestine Technical University-Kadoorie-Ptuk, Tulkarm 303, Palestine

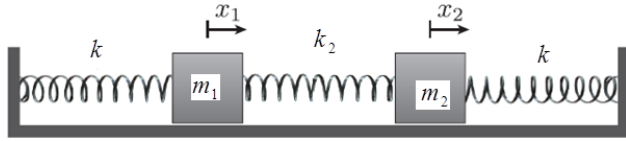
H. Khalilia: Department of Applied Computing, College of Sciences, P.O. box 7, Palestine Technical University-Kadoorie-Ptuk, Tulkarm 303, Palestine

and the Lagrangian of the coupled oscillator is obtained. In Section 3, the numerical method used is presented. Section 4 provides the numerical solutions of the derived equations of motion for different initial conditions. The results were discussed in Section 5. We finally end the paper with conclusions.

## 2 Physical description of the coupled oscillator

Let us consider the system shown in Figure 1, which consists of two coupled carts of masses  $m_1$  and  $m_2$ , attached to their respective walls by two identical springs (with force constant  $k$ ) and to each other by another spring (with force constant  $k_2$ ) [1]. It is of interest to mention that in some books, this system is known as: diatomic molecules.

Assuming all springs to be massless, we aim to obtain the ELE for the system, and then we require obtaining a numerical solution for these equations under different initial conditions.



**Figure 1:** Two coupled carts with two different masses

First of all, the kinetic energy ( $T$ ) of the system reads:

$$T = \frac{1}{2} m_1 \dot{x}_1^2 + \frac{1}{2} m_2 \dot{x}_2^2. \quad (1)$$

In order to write the potential energy ( $V$ ) of the system, we have to identify the extensions of the three springs shown in Figure 1 as ( $x_1$ ,  $x_1 - x_2$ , and  $-x_2$ ). As a result, the potential energy reads:

$$V = \frac{1}{2} (k + k_2) x_1^2 - k_2 x_1 x_2 + \frac{1}{2} (k_2 + k) x_2^2. \quad (2)$$

The classical Lagrangian can be immediately written as:

$$L = T - V = \frac{1}{2} m_1 \dot{x}_1^2 + \frac{1}{2} m_2 \dot{x}_2^2 - \frac{1}{2} (k + k_2) x_1^2 + k_2 x_1 x_2 - \frac{1}{2} (k_2 + k) x_2^2. \quad (3)$$

Our final step in this section is in obtaining the equations of motion. As such, applying our Lagrangian (eq. 3) to the following equation  $\frac{\partial L}{\partial q} - \frac{d}{dt} \frac{\partial L}{\partial \dot{q}} = 0$ , with  $q = x_1, x_2$ , we got the following ELE:

For  $q = x_1$ , we have:

$$m_1 \ddot{x}_1 = -(k + k_2) x_1 + k_2 x_2. \quad (4)$$

While for  $q = x_2$ , we have:

$$m_2 \ddot{x}_2 = k_2 x_1 - (k_2 + k) x_2. \quad (5)$$

Now, we aim to obtain the Hamilton's equations of motion. The first point is introducing the following generalized momenta:

$$p_{x_1} = \frac{\partial L}{\partial \dot{x}_1} = m \dot{x}_1. \quad (6)$$

$$p_{x_2} = \frac{\partial L}{\partial \dot{x}_2} = m \dot{x}_2. \quad (7)$$

As a result, the Hamiltonian  $H = \sum \dot{q}_i p_{x_i} - L$ , with  $i = 1, 2$  reads:

$$H = \frac{p_{x_1}^2}{2m_1} + \frac{p_{x_2}^2}{2m_1} + \frac{1}{2} (k + k_2) x_1^2 - k_2 x_1 x_2 + \frac{1}{2} (k_2 + k) x_2^2. \quad (8)$$

Finally, the Hamilton's equations can be obtained as:

$$\dot{x}_1 = \frac{\partial H}{\partial p_{x_1}} \Rightarrow m_1 \ddot{x}_1 = -(k + k_2) x_1 + k_2 x_2. \quad (9)$$

$$\dot{x}_2 = \frac{\partial H}{\partial p_{x_2}} \Rightarrow m_2 \ddot{x}_2 = k_2 x_1 - (k_2 + k) x_2. \quad (10)$$

In the section below, we aimed to obtain numerical solution for eqs. (9, 10) for some initial conditions.

## 3 Numerical solution method

A standard solver for ordinary differential equations (ODEs) in the simulation of MATLAB's framework is the ode45 function. ODE45 executes the Runge-Kutta method with a variable time step for optimal computation [18–20]. A fourth-order Runge-Kutta algorithm is useful in solving non-linear differential equations, taking into consideration some given initial conditions [21]. Any problem in the following manner

$$\frac{dx}{dt} = f(t, x). \quad (11)$$

With  $x(t_0) = x_0$

Where  $t$  is the independent variable,  $x$  is a vector of dependent variables to be found, and  $f(t, x)$  is a function of  $t$ , and  $x$  can be solved using ode45. Specifying  $f(t, x)$  and the initial conditions enable us to solve the problem [22].

Bellow a short explanation is presented to explain how Runge-Kutta works [23].

In general, given that:

$$y' = f(x, y). \quad (12)$$

$$y(x_n) = y_n. \quad (13)$$

Then, one can compute in turn:

$$k_1 = hf(x_n, y_n). \quad (14)$$

$$k_2 = hf\left(x_n + \frac{h}{2}, y_n + \frac{k_1}{2}\right). \quad (15)$$

$$k_3 = hf\left(x_n + \frac{h}{2}, y_n + \frac{k_2}{2}\right). \quad (16)$$

$$k_4 = hf(x_n + h, y_n + k_3). \quad (17)$$

$$y_{n+1} = y_n + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4). \quad (18)$$

Now, everything becomes in first-order form, so the list of commands is needed in the code [24].

ODE45 is used to implement “first try” solution for most problems [25, 26]. For more details about the Runge-Kutta method, one can refer to [23, 27].

Repetition and conditional structures are used to repeat sequences in MATLAB, and from these structures are: if and if-else statements, do while, for each, while, and for loops.

In literature, there are many examples of repetition and conditional structures [23, 27].

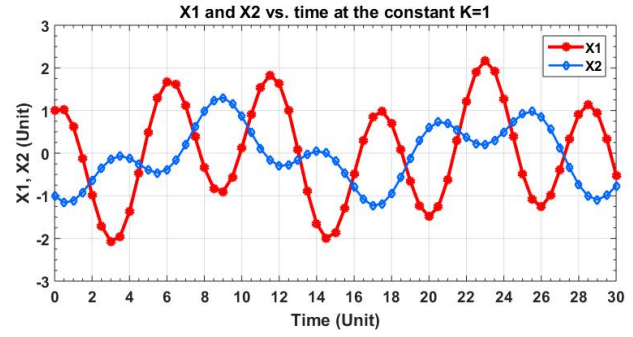
In Section 4, we propose the numerical solution for eqs. (4 and 5) using loop structure. We consider some initial conditions, and the numerical solution for these particular conditions has been obtained.

## 4 Simulation results

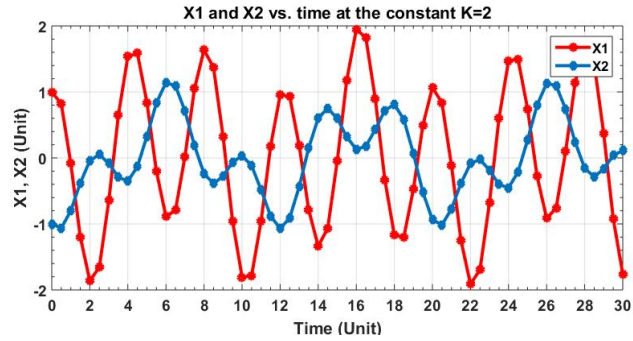
Below, we consider some selected cases to obtain the numerical solution of eqs. (4, 5):

### Case 1: Symmetric Case

In this case, we considered the two initial conditions  $x_1(0) = -x_2(0) = 1$  and  $\dot{x}_1(0) = -\dot{x}_2(0) = 0.5$ ; on the other hand, the other parameters were assumed to be  $k = 1, 2$ , with  $k_2 = 2k$ , while  $m_1 = 3$  and  $m_2 = 7$ , see Figures 2, 3:



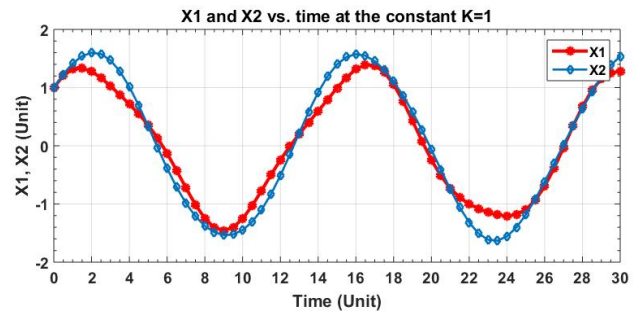
**Figure 2:** The positions  $x_1$  and  $x_2$  of masses  $m_1$  and  $m_2$ , respectively, against time  $t$  (the red line for mass  $m_1$ , and the black line for mass  $m_2$ ).



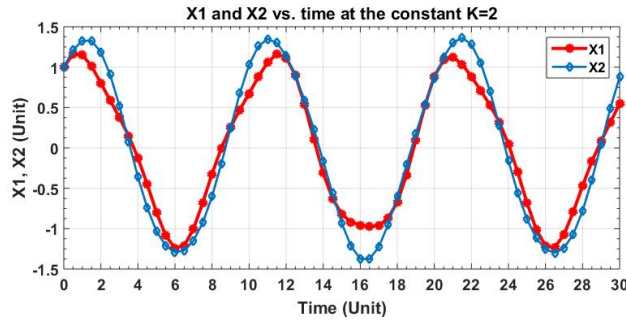
**Figure 3:** The positions  $x_1$  and  $x_2$  of masses  $m_1$  and  $m_2$ , respectively, against time  $t$  (the red line for mass  $m_1$  and the black line for mass  $m_2$ ).

### Case 2: Anti-symmetric Case

In this case, we considered the following two initial conditions  $x_1(0) = x_2(0) = 1$  and  $\dot{x}_1(0) = \dot{x}_2(0) = 0.5$ , and the other parameters were assumed to be  $k = 1$ , and 2, with  $k_2 = 2k$ , while  $m_1 = 3$  and  $m_2 = 7$ , see Figures 4, 5:



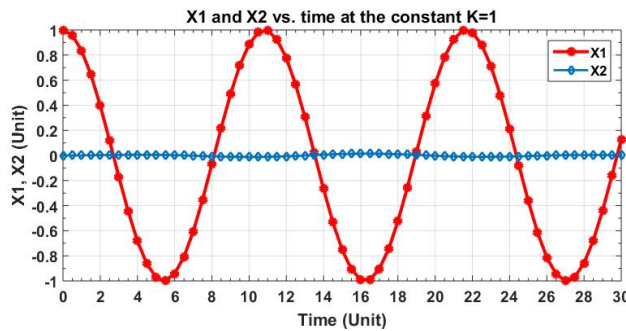
**Figure 4:** The positions  $x_1$  and  $x_2$  of masses  $m_1$  and  $m_2$ , respectively, against time  $t$  (the red line for mass  $m_1$  and the black line for mass  $m_2$ ).



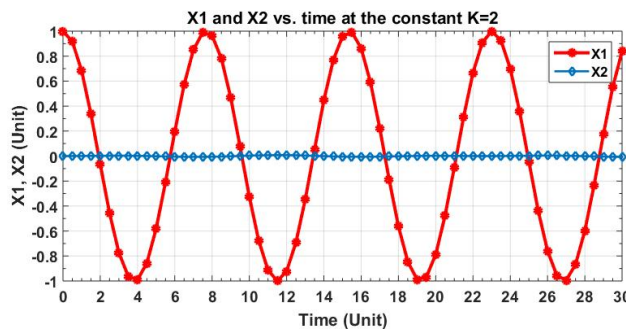
**Figure 5:** The positions  $x_1$  and  $x_2$  of masses  $m_1$ , and  $m_2$ , respectively, against time  $t$  (the red line for mass  $m_1$  and the black line for mass  $m_2$ ).

### Case 3: Weak Coupled Oscillator ( $k_2 \ll k$ )

Finally, we considered the weak coupled oscillator case, which is of special importance. In this case, the following parameters are considered  $k = 1$ , and  $2$ , with  $k_2 = 0.01$ , while  $m_1 = 3$  and  $m_2 = 7$ , with the following two initial conditions  $x_1(0) = 1$ ,  $x_2(0) = 0$  and  $\dot{x}_1(0) = \dot{x}_2(0) = 0$ , see Figures 6, 7:



**Figure 6:** The positions and  $x_2$  of masses  $m_1$ , and  $m_2$ , respectively against time  $t$  (the red line for mass  $m_1$ , and the black line for mass  $m_2$ ).



**Figure 7:** The positions  $x_1$  and  $x_2$  of masses  $m_1$  and  $m_2$ , respectively against time  $t$  (the red line for mass  $m_1$  and the black line for mass  $m_2$ ).

## 5 Results and Discussion

In this paper, the motion of the coupled oscillator system has been studied numerically using the ode45 code, and the solutions  $x_1(t)$  and  $x_2(t)$  have been plotted for the several considered cases as in Figures 2–7.

Figures 2, 3 present the positions  $x_1(t)$  and  $x_2(t)$  of masses  $m_1$  and  $m_2$  against the time, respectively, for the symmetric case. The results in the same figures show that the periodicity of the oscillations has disappeared. Also, the influence on mass  $m_2$  is clearer than that on  $m_1$ , because it has a larger value. In addition, because  $m_1$  and  $m_2$  were displaced initially in opposite directions (i.e.,  $x_1(0) = -x_2(0)$ ) and then released, this is the main reason for disappearing the periodicity of the oscillations.

Figures 4, 5 present the positions  $x_1(t)$  and  $x_2(t)$  of masses 1 and 2 against the time, respectively, for the anti-symmetric case. Again, the periodicity of the oscillation has disappeared, but less than that in Figures 2, 3. The reason here refers to the fact that the two masses  $m_1$  and  $m_2$  were displaced initially in the same direction (i.e.,  $x_1(0) = x_2(0)$ ) and then released.

For Cases 1 and 2, it is expected that the periodicity of the oscillations will be destroyed. This is due to the fact that the spring-connected the two masses together have a larger stiffness than the two other springs.

In Figures 6, 7, the positions  $x_1(t)$  and  $x_2(t)$  of masses 1 and 2, respectively, is displaced against time for the weak coupled oscillator, where the stiffness of the spring connecting the two masses is very small compared with the other springs stiffness. In our case, we consider  $k_2 = 0.01$ , whereas  $k = 1$  is shown in Figure 6, and  $k = 2$  is shown in Figure 7. It is clear from these two figures that mass  $m_1$  undergoes a simple harmonic oscillations, with a constant amplitude and a constant period, while mass  $m_2$  nearly vibrates but with very small oscillations around the equilibrium position; this is due to the fact that  $m_2$  has a larger value than  $m_1$ .

Finally, one can see from the figures that if the stiffness of the spring connecting the two masses is larger than the other springs' stiffness, then this will not allow the system to undergo simple harmonic motion. As a result, the periodicity of the motion is destroyed.

## 6 Conclusion

In conclusion, the periodicity of the oscillator disappeared, especially in the symmetric case. While in the anti-symmetric case, the periodicity is nearly still there and not



destroyed. In the weak coupled case, one can see from Figures 6, 7 that mass  $m_2$  does not move, because it is larger than  $m_1$ .

After releasing the system, the displacement in the symmetric and anti-symmetric cases exceeds the initial displacement. This can be explained by the effects of springs on each other. Nonetheless, for the weak coupled oscillator, mass  $m_1$  undergoes a simple harmonic motion and not affected by mass  $m_2$ .

**Acknowledgement:** The authors would like to acknowledge Palestine Technical University-Kadorrie (PTUK) for research grants and funding.

## References

- [1] **Marion J. B., and Thornton S. T.** Classical Dynamics of Particles and Systems, 3rd edn. (Harcourt Brace Jovanovich), **1988**.
- [2] **Hand L. N., and Finch J. D.** Analytical Mechanics. Cambridge University Press, **2012**.
- [3] **Murray R. M.** Nonlinear control of mechanical systems: A Lagrangian perspective. Annual Reviews in Control. Vol. 21, pp. 31-42, **1997**.
- [4] **Musielak Z. E., Roy D., and Swift L. D.** Method to derive Lagrangian and Hamiltonian for a nonlinear dynamical system with variable coefficients. Chaos, Solitons and Fractals. Vol. 38, no. 3, pp. 894-902, **2008**.
- [5] **Hajimiri A., and Lee T.** A General Theory of Phase Noise in Electrical Oscillators," IEEE Solid State Circuits, vol. 33, no. 2, pp. 179-194, **1998**.
- [6] **Liu M. L.** Principle and Application of Oscillator, Higher Education Press, Beijing, pp. 41-43, **1984**.
- [7] **Butcher J. C.** Numerical methods for ordinary differential equations, second revised ed., Wiley, Chichester, **2008**.
- [8] **Hairer E., Lubich C., and Wanner G.** Geometric numerical integration. Structure preserving algorithms for ordinary differential equations. 2nd revised ed., Springer, Berlin, **2006**.
- [9] **Deufhard P., and Bornemann F.** Scientific computing with ordinary differential equations, Springer, New York, **2002**.
- [10] **Atkinson K., Han W., and Stewart D.** Numerical Solution of Ordinary Differential Equations. A John Wiley and sons, Inc., Publication, **2008**.
- [11] **Potra F. A., and Yen J.** Implicit Numerical Integration for Euler-Lagrange Equations via Tangent Space Parameterization. Mechanics of Structures and Machines, vol. 19, no. 1, pp. 77-98, **2007**.
- [12] **Rheinboldt W. C.** Performance Analysis of Some Methods for Solving Euler-Lagrange Equations. Appl. Math. Lett. vol. 8, no. 1, pp. 77-82, **1995**.
- [13] **Hussain K. A., Ismail F., Senu N., and Rabiei F.** Fourth-Order Improved Runge-Kutta Method for Directly Solving Special Third-Order Ordinary Differential Equations. Iranian Journal of Science and Technology, Transaction A: Science. Vol. 41, no. 2, **2017**.
- [14] **Chauhan, V., Srivastava, P.K.** Computational techniques based on Runge-Kutta method of various order and type for solving differential equations. International Journal of Mathematical, Engineering and Management Sciences Vol. 4, no. 2, **2019**.
- [15] **Khalilia H., Jarrar R., and Asad J.** Numerical study of motion of a spherical particle in a rotating parabola using Lagrangian. Journal of the Serbian Society for Computational Mechanics. Vol. 12, no. 1, **2018**.
- [16] **Asad J., Florea O., and Khalilia H.** Numerical study of the motion of a heavy ball sliding on a rotating wire. Bulletin of the Transilvania University of Brasov, Series III: Mathematics, Informatics, Physics. Vol. 13, no. 1, **2020**.
- [17] **Asad J., and Florea O.** Numerical aspects of two coupled harmonic oscillators. Analele Stiintifice ale Universitatii Ovidius Constanta, Seria Matematica. Vol. 28, no. 1, **2020**.
- [18] **Rutia W.** Beitrag zur naherungsweisen Integration totaler Differentialgleichungen. Zeitschrift Math. Phys. Vol. 46, pp. 435- 453, **1901**.
- [19] **Nystrom E. I.** Acta Societatis Scientiarum Fennicae 50, No. 13, pp.1- 55, **1925**.
- [20] **Moore H.** Matlab for Engineers. Third Edition, **2005**.
- [21] **Marsavina L., Nurse A. D., Braescu L., and Craciun E. M.** Stress singularity of symmetric free-edge joints with elasto-plastic behavior," Computational Materials Science. Vol. 52, no. 1, pp. 282-286, **2012**.
- [22] **Shampine L. F.** Numerical Solution of Ordinary Equations. Chapman and Hall, **1994**.
- [23] **Attawa S.** Matlab: A Practical Introduction to Programming and Problem Solving, College of Engineering, Boston University, Boston, MA, **2009**.
- [24] **Press W. H., Teukolsky S. A., Vetterling W. T., and Flannery B. P.** Numerical recipes in C: The art of scientific computing. Second Edition. Cambridge University Press, **1992**.
- [25] **Dormand J. R., and Prince P. J.** A family of embedded Runge-Kutta formulae," J. Comp. Appl. Math, vol. 6, pp. 19-26, **1980**.
- [26] **Houcque D., and Robert R.** Applications of MATLAB: Ordinary Differential Equations (ODE), McCormick School of Engineering and Applied Science – Northwestern University, **2007**.
- [27] **Gilat A.** Matlab an Introduction with Applications, Fourth Edition, Ohio University, **2011**.