Télécom Physique Strasbourg
Projet Mathématiques-Informatique

January 2023

Chaotic double pendulum

Hugo PAGES, Tony PREDIERI, Antonin PIVARD, Clotilde ROUSSET,Moha

Contents

1 Introduction

Theory of chaos is a sector of mathematics and physics. In order to learn more about this topic, we studied the motion of the double pendulum. A simple pendulum consists of a string that holds a point mass on its extremity. The double pendulum is the chained association of two simple pendulums.

The study of the chaotic pendulum is a great example of a coupled oscillator and even if the system seems to be quite simple its evolution is described as chaotic. A chaotic system is a dynamic system very sensitive to the initial conditions. Indeed, an infinitesimal change in the initial conditions produces a totally different evolution which leads to impossible prediction of the movement in the long term. Before studying the double pendulum, we will look at the simple pendulum so that we could explain and reuse different concepts for the double pendulum.

Remark: We will not develop the python code used in this part, some part of the code will be developed only on the study of the double pendulum.

2 The simple pendulum

2.1 Movement equations

We know that motion's equations for the simple pendulum are quite easy to obtain using Newtonian mechanics. But for the double pendulum, it is a much longer and more laborious approach. Therefore, we decided to use the Lagrangian approach.

In order to discover the Lagrangian approach that was new for us, we decided to use this method on the simple pendulum which is an easier case than the double pendulum. This method is a strong way to get a motion equation.

For the simple pendulum, this approach consists of calculating first the difference between kinetic and gravitational potential energies. This difference is called the Lagrangian. Then use the Euler-Lagrange equation to finally get the equation of motion. We considered the string as non elastic and a Galilean reference frame without friction.

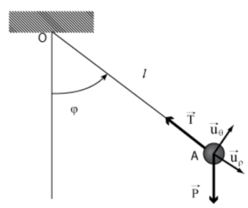


Figure 1: Simple pendulum schema, http://res-nlp.univ-lemans.fr/

We considered the motion's equation of the simple pendulum known, it is:

$$\ddot{\theta} + \frac{g}{l}\sin\theta = 0$$

Our Lagrangian approach will be true if we obtain the same equation.

$\overrightarrow{OA} = l\overrightarrow{u_r}$
$\overrightarrow{v} = l\frac{d\theta}{dt}\overrightarrow{u_\theta}$
$\overrightarrow{a} = l\frac{d^2\theta}{dt^2}\overrightarrow{u_\theta} - l(\frac{d\theta}{dt})^2\overrightarrow{u_r}$

Kinetic energy :
$E_c = \frac{1}{2}mv^2 = \frac{1}{2}ml^2\dot{\theta}^2$

Gravitational potential energy :
$E_p = mgz = mgl(1 - \cos\theta)$

Lagrangian calculation:
$L = E_c - E_p = \frac{1}{2}ml^2\dot{\theta}^2 - mgl(1-\cos\theta)$

Then we use the Euler-Lagrange equation :
$\frac{\partial L}{\partial \theta} - \frac{d}{dt}(\frac{\partial L}{\partial \dot{\theta}}) = 0$

Which gives us:
$-mgl\sin\theta - ml^2\ddot{\theta} = 0$

Then finally :
$\ddot{\theta} + \frac{g}{l}\sin\theta = 0$
Which is the same equation as using Newtonian mechanics.

2.2 Modelisation

To resolve numerically the equation of the simple pendulum, we used the Runge-Kutta 4 method. This method, which is based on the approximation of Taylor series permits resolving numerically differential equations.

Taylor : $\phi(t,u) = \displaystyle\sum_{k=1}^{p} f^{k-1}(t,u)\frac{h^{k-1}}{k!}$

Runge-Kutta : $\phi(t,u) = \displaystyle\sum_{k=1}^{p} \alpha_k f(t_a + \beta_k h, u_i + \beta_k^{'} h f(t_i^{'}, u_i^{'}))$

Runge-Kutta's method is based on the iteration principle. Indeed, a first estimation of the solution is used to calculate a second estimation, but more precisely then we repeat the operation. Therefore, the method of Runge-Kutta by order four uses four iterations to resolve the differential equation, so we can have both precision and interesting speed of calculation. To validate this resolution we plotted the analytic and the numerical solutions.

The analytical resolution is :

$\ddot{\theta} + \omega_0^2\theta = 0$, with $\omega_0^2 = \frac{g}{l}$

Here is the analytical solution :
$\theta(t) = \theta_0\sin(\omega_0 t)$, with period $T_0 = \frac{2\pi}{\omega_o}$
For the Runge-Kutta resolution we used a step of 1 and a steps time of 0.05 s and used a string of 1 meter. We considered the modeling only in 2 dimensions.

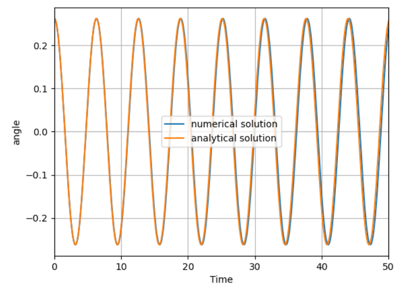


Figure 2: Analytical and numerical resolution of the equation

The analytical and the numerical resolution are overlaid on the graph which proof that our runge-kutta method is working and also that this method is precise enough to be used in the double pendulum resolution.

2.2.1 angular and speed evolution with RK4

The Runge-Kutta 4 method makes it possible to determine values of the angle and the speed of the pendulum over time.

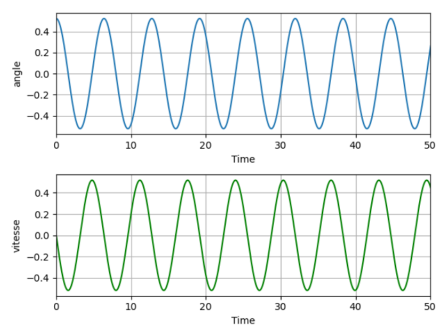


Figure 3: Angular and speed variation during time of the pendulum with omega_0 =0 and $\theta_0 = \frac{\pi}{12}$

We can see that without any friction there is no attenuation which means that the pendulum moves with the same amplitude without slowing down.

2.2.2 Trajectory of the pendulum

In order to have a better visualisation of the system we wanted to animate the pendulum. Therefore, we transformed the polar coordinates from the resolution on Cartesian coordinates plot his trajectory. The coordinates of the first pendulum are calculated with this formula: $x_1 = l_1 \sin\theta_1$ and $y_1 = -l_1 \cos\theta_1$

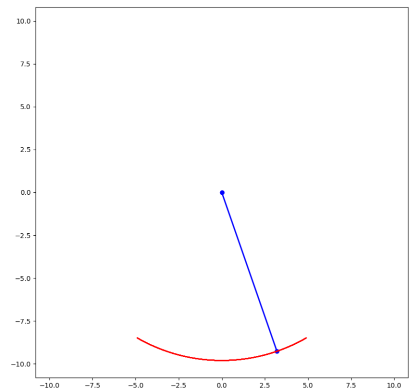


Figure 4: Modeling of the pendulum

2.3 Propriety of the pendulum

2.3.1 Energies of the system

Mechanical energy of the simple pendulum :
$E_m = E_c + E_p = \frac{1}{2}ml^2\dot{\theta}^2 + mgl(1 - \cos\theta)$

This energy is constant, so its time derivative is zero. Indeed in a closed systeme the Mechanical energy is conserved.

Thanks to the precedent data, we are able to determine the kinetic, potential and mechanical energies of the system. If the modeling of the system is correct we should find a constant mechanical energy because of the conservation of energies in a closed system.

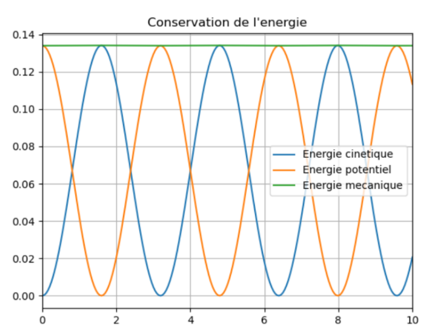


Figure 5: Kinetic, potential and mechanical energies during time

We can see that the mechanical energy is a constant function of time which implies a conservation of the energies in the system. The theory is confirmed.

2.3.2 Phase portrait

We placed ourselves in a small angle condition and traced the phase portrait which is the speed as the function of the angle.

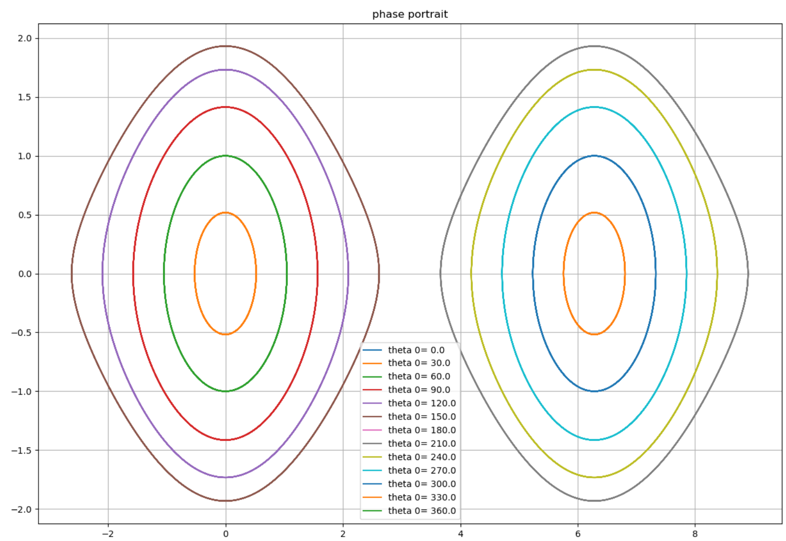


Figure 6: Phase portrait with theta

This phase portrait shows two different phenomena, the first shows that for small angle we tend to have a sinusoidal oscillation and an almost circular trajectory. The non sinusoidal oscillation is shown for large angle. The fact that every portrait is "closed" is because we neglected friction.

2.4 Conclusion on the utility of studying the simple pendulum before the double pendulum

Even if the simple pendulum is not a chaotic object, its study is a great mean to verify our resolution method and also to create python function that we could reuse for the double pendulum as our phase portrait function or our energies conservation function. Also, with some conditions the double pendulum can act like a simple pendulum so the precedent study is an interesting way to verify the modeling. Finally the impact of the initial conditions and the modification of mass and length will be only studied and developed for the double pendulum.

3 Double pendulum

3.1 Equation of motion using the Lagrange method

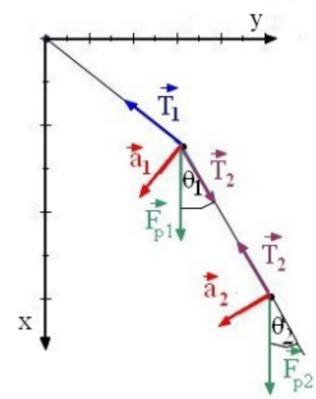


Figure 7: Noémie Jaquier Ma2-3 Travail de maturité 2009-2010 page 6.

We define theta1 between the X axes and the first pendulum and theta 2 between the X axes and the second pendulum.

Kinetic and gravitational potential energies calculation
$\overrightarrow{V}(M_1/R) = \frac{d}{dt}\overrightarrow{OM_1} = l_1\dot{\theta}_1 \vec{u_{\theta 1}}$ thus $\overrightarrow{V}^2(M_1/R) = (l_1\dot{\theta}_1)^2$

$\overrightarrow{V}(M_2/R) = \frac{d}{dt}\overrightarrow{0M_2} = \frac{d}{dt}(\overrightarrow{0M_1} + \overrightarrow{M_1M_2}) = l_1\dot{\theta}1\vec{u_{\theta 1}} + l_2\dot{\theta}_2\vec{u_{\theta 2}}$
thus $\overrightarrow{V}^2(M_1/R) = (l_1\dot{\theta}_1)^2 + (l_2\dot{\theta}_2)^2 + 2l_1l_2\dot{\theta}_1\dot{\theta}_2\cos(\theta_2 - \theta_1)$

Finally
$Ec = Ec_1 + Ec_2 = \frac{1}{2}(m_1 + m_2)(l_1\dot{\theta}_1)^2 + \frac{1}{2}m_2(l_2\dot{\theta}_2)^2 + m_2l_1l_2\dot{\theta}_1\dot{\theta}_2\cos(\theta_2 - \theta_1)$

$\delta W = (\overrightarrow{P_1} + \overrightarrow{T_1} - \overrightarrow{T_2}).d\overrightarrow{OM_1} + (\overrightarrow{P_2} + \overrightarrow{T_2}).d\overrightarrow{OM_2}$

Only weight's work is not zero. So we have :

$\delta W = \overrightarrow{P_1}.d\overrightarrow{OM_1} + \overrightarrow{P_2}.d\overrightarrow{OM_2}$
$= d(m_1gl_1\cos(\theta_1) + m_2gl_1\cos(\theta_1) + m_2gl_2\cos(\theta_2))$
$= -dE_p$

Thus
$E_p = -(m_1 + m_2)gl_1\cos(\theta_1) - m_2gl_2\cos(\theta_2) + (cst = 0)$

So we can calculate the Lagrangian:
$L = E_c - E_p$
$= \frac{1}{2}(m_1+m_2)(l_1\dot{\theta}_1)^2+\frac{1}{2}m_2(l_2\dot{\theta}_2)^2+m_2l_1l_2\dot{\theta}_1\dot{\theta}_2\cos(\theta_2-\theta_1)+(m_1+m_2)gl_1\cos(\theta_1)+ m_2gl_2\cos(\theta_2)$

Euler-Lagrange equation :
$\frac{dL}{dt}(\frac{dL}{d\dot{\theta}_1}) - \frac{\partial L}{\partial \theta_1} = 0$
$(m_1+m_2)l_1\ddot{\theta}_1+m_2l_2\ddot{\theta}_2\cos(\theta_1-\theta_2)-m_2l_2(\dot{\theta}_2)^2\sin(\theta_1-\theta_2)+(m_1+m_2)g\sin(\theta_1) = 0$

$\frac{dL}{dt}(\frac{dL}{d\dot{\theta}_2}) - \frac{\partial L}{\partial \theta_2} = 0$
$m_2l_2\ddot{\theta}_2 + m_2l_1\ddot{\theta}_1\cos(\theta_1-\theta_2) - m_2l_1(\dot{\theta}_1)^2\sin(\theta_1-\theta_2) - m_2g\sin(\theta_2) = 0$

Finally, we obtain equations of motion of the double pendulum :

$$\begin{cases} (m_1+m_2)l_1\ddot{\theta}_1 + m_2l_2\ddot{\theta}_2\cos(\theta_1-\theta_2) - m_2l_2(\dot{\theta}_2)^2\sin(\theta_1-\theta_2) + (m_1+m_2)g\sin(\theta_1) = 0 \\ m_2l_2\ddot{\theta}_2 + m_2l_1\ddot{\theta}_1\cos(\theta_1-\theta_2) - m_2l_1(\dot{\theta}_1)^2\sin(\theta_1-\theta_2) - m_2g\sin(\theta_2) = 0 \end{cases}$$

3.2 The resolution method

Firstly, we wondered why do we need to resolve numerically the movement equation of the double pendulum. It is because the motion's equations consists of a system of two non linear combined equations of the second order that we cannot solve analytically. This means that we can only approach the solution by an numerical approach, and the precision of the approach depends of the method.
As said before we used the Runge-Kutta 4 method and we wrote a function that will take as argument:the masses and the length of the two pendulums, the initial positions of θ_1 and θ_2, the initial speed, the number of points we want for the acquisition and the time of the acquisition. The last two parameters permit us calculating the step time used for the Runge-Kutta 4 method.
This function will return three arrays, the first one with the angles of the two pendulums $\theta = [\theta_1, \theta_2]$, the second with speed of the two pendulums $v = [v_1, v_2]$ and the third is the time. This resolution function is composed of three functions that we will develop in the next part.

3.2.1 The Lagrange function

The Lagrange function (called f_double_pendulum in our python program) is the first of the three functions written in the resolve function. It contains the equation found previously with the Lagrangian approach. The state in argument is an array corresponding to the instantaneous pendulum state.

```
def f_double_pendulum(M1,M2,R1,R2,state):
    n_state = np.zeros_like(state)
    n_state[0] = state[1]  #theta1 = d_theta1
    n_state[2] = state[3]  #theta2 = d_theta2

    theta1 = state[0]
    theta2 = state[2]
    omega1 = state[1]
    omega2 = state[3]
    denom = (R1*(2*M1+M2-M2*np.cos(2*theta1-2*theta2)))

    n_state[1] = (-g*(2*M1 + M2)*np.sin(theta1) - M2*g*np.sin(theta1 - 2*theta2)
- 2*np.sin(theta1 - theta2)*M2*((omega2**2)*R2+(omega1**2)*R1*np.cos(theta1-
theta2)))/denom
    n_state[3] = (2*np.sin(theta1 - theta2)*((omega1**2)*R1*(M1+M2)
+g*(M1+M2)*np.cos(theta1)+(omega1**2)*R2*M2*np.cos(theta1-theta2)))/denom
    return(n_state)
```

Figure 8: python program corresponding to the Lagrange equation.

3.2.2 The Runge-Kutta 4 function

We used the same Runge-Kutta 4 method than for the simple pendulum because of its precision and its speed time. The program corresponding to this method is:

```
def RK4(M1,M2,R1,R2,state, h, F=None):
    k1 = F(M1,M2,R1,R2,state)
    k2 = F(M1,M2,R1,R2,state + 0.5 * h * k1)
    k3 = F(M1,M2,R1,R2,state + 0.5 * h * k2)
    k4 = F(M1,M2,R1,R2,state + h * k3)
    return (state + h * (k1 + 2. * k2 + 2. * k3 + k4) / 6)
```

Figure 9: python program corresponding to the RK 4 resolution.

The impact of the step on the Runge-Kutta 4 method will be studied after a first approach of the movement of the double pendulum (cf 3.3.4).

3.2.3 The final resolution

The third function registers inside two arrays (theta, omega), the state of the double pendulum at every step and return these two arrays. We finally wrote a loop in the resolve function to repeat the operation until the time limit and we obtained the numerical solution of the two combined equations.

```
def resolve(M1,M2,R1,R2,theta1,theta2, omega1,omega2,n_samples,t_lim,frame_s=60):
    time = np.array([0.])
    step = float(t_lim/n_samples)
    state = np.array([theta1, omega1, theta2, omega2])  #instantaneous pendulum state
    skip_frames = int(n_samples/(t_lim*frame_s))

    def f_double_pendulum(M1,M2,R1,R2,state):
        n_state = np.zeros_like(state)
        n_state[0] = state[1]  #theta1 = d_theta1
        n_state[2] = state[3]  #theta2 = d_theta2

        theta1 = state[0]
        theta2 = state[2]
        omega1 = state[1]
        omega2 = state[3]
        denom = (R1*(2*M1+M2-M2*np.cos(2*theta1-2*theta2)))

        n_state[1] = (-g*(2*M1 + M2)*np.sin(theta1) - M2*g*np.sin(theta1 - 2*theta2) - 2*np.sin(theta1
theta2)*M2*((omega2**2)*R2+(omega1**2)*R1*np.cos(theta1-theta2)))/denom
        n_state[3] = (2*np.sin(theta1 - theta2)*((omega1**2)*R1*(M1+M2)+g*(M1+M2)*np.cos(theta1)+
(omega1**2)*R2*M2*np.cos(theta1-theta2)))/denom
        return(n_state)


    def RK4(M1,M2,R1,R2,state, h, F=None):
        k1 = F(M1,M2,R1,R2,state)
        k2 = F(M1,M2,R1,R2,state + 0.5 * h * k1)
        k3 = F(M1,M2,R1,R2,state + 0.5 * h * k2)
        k4 = F(M1,M2,R1,R2,state + h * k3)
        return (state + h * (k1 + 2. * k2 + 2. * k3 + k4) / 6)

    def states_to_lists(states):
        if isinstance(states, list):
            n_samples = len(states)
            n_rods = int(len(states[0])/2)
            theta = [np.zeros(n_samples) for _ in range(n_rods)]
            omega = [np.zeros(n_samples) for _ in range(n_rods)]
            for i, state in enumerate(states):
                for j in range(n_rods):
                    theta[j][i] = state[(j*2)]
                    omega[j][i] = state[(j*2)+1]
        else:
            n_rods = int(len(states)/2)
            theta = [np.zeros(1) for _ in range(n_rods)]
            theta = [np.zeros(0) for _ in range(n_rods)]
            for i in range(n_rods):
                theta[i] = states[(j*2)]
                omega[i] = states[(j*2)+1]
        return(theta, omega)


    list_states = [state]
    for i in range(n_samples):
        state = RK4(M1,M2,R1,R2,state, step, f_double_pendulum)
        list_states.append(state)
        time = np.append(time, (i+1)*step)
    Theta, Omega = states_to_lists(list_states)
    return Theta, Omega, time
```

Figure 10: python program corresponding to the RK 4 resolution.

3.3 Dynamics of the system

The numerical resolution gave us two arrays composed by the evaluation of the two angles and omega at every step of time during the time evaluation. The study of this data will permit plotting the angular evolution or the speed evolution of the system during time or also to observe properties of the chaotic system.

3.3.1 Style of the angular and speed evolution during time

For the resolution we chose M1=M2=0.1 and R1=1, R2=2.

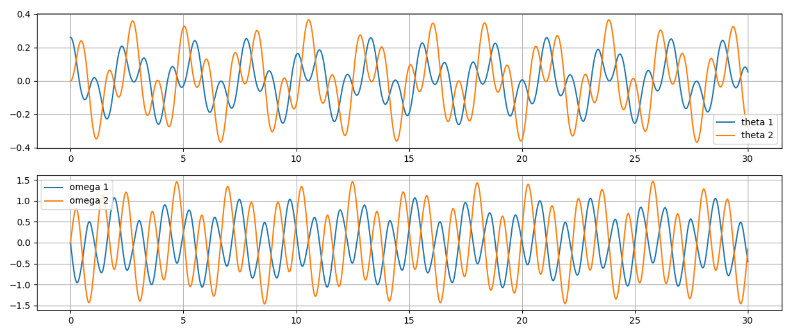


Figure 11: angular and speed evolution for theta1 0 =15°and theta2 0 = 0°

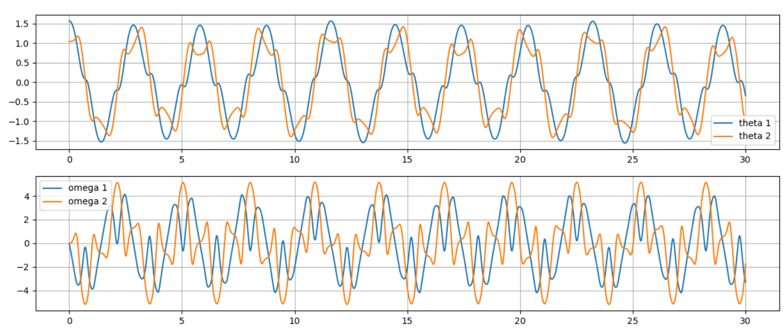


Figure 12: angular and speed evolution for theta1 0 =90° and theta2 0 =60°

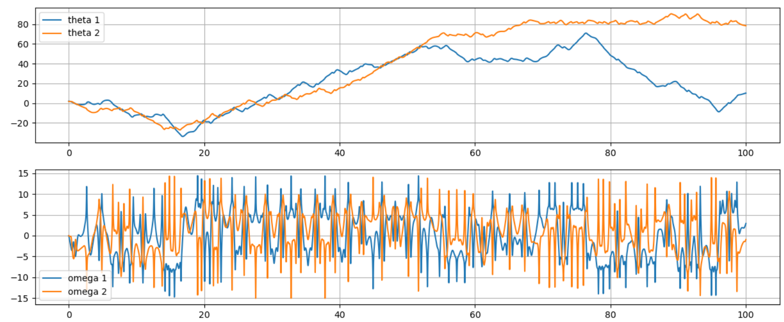


Figure 13: angular and speed evolution for theta1 0 =115° and theta2 0 =100°

We can observe that for small angles the double pendulum seems to have a sinusoidal movement which is not the case for large angles.

3.3.2 Animation of the double pendulum

To observe the differences between small and large angles we animated the double pendulum for the two cases. To do this animation we calculated the Cartesian coordinates of the two pendulum for each theta values of the resolution.

We calculated the x and y coordinates of the first pendulum with formula determined in the study of the simple pendulum: $x_1 = l_1 * sin\theta_1$ and $y_1 = -l_1 * cos\theta_1$ The position of the second pendulum depends on the position of the first pendulum $x_2 = x_1 + l_2 * sin\theta_2$ and $y_2 = y_1 - l_2 * cos\theta_2$ To calculate the coordinates of any pendulum without regarding if its the first or the second we decided to code the following program:

```
def polar_to_cartesian(r, theta=None, offset_x=0, offset_y=0):
    """
    Convert polar to cartesian

    r = scalar.
    return has the same type of theta.
    offset_x = scalar or list.
    offset_y = scalar or list.
    """
    if isinstance(theta, (list, np.ndarray)):
        if isinstance(offset_x , int) and offset_x == 0:
            offset_x = np.zeros(len(theta))
            offset_y = np.zeros(len(theta))
        X = r*np.sin(theta) + offset_x
        Y = -r*np.cos(theta) + offset_y
        return(X, Y)
```

Figure 14: function that calculates the Cartesian coordinates of any pendulum

We are now able to animate the double pendulum by only having the values of θ_1 and θ_2

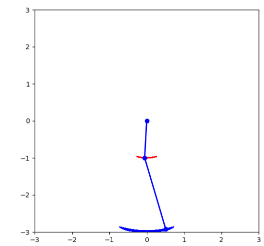


Figure 15: pendulum animation for theta1 0 =15° and theta2 0 =0°

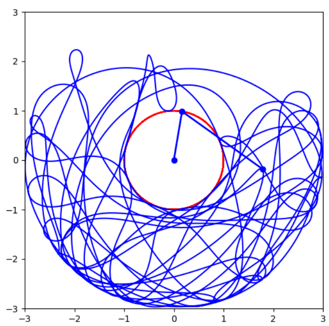


Figure 16: pendulum animation for theta1 0 =115° and theta2 0 =100°

For large angles, there is a moment when both pendulums made a full 360° rotation.

3.3.3 Impact of the initial angular position on the system

This system is a chaotic system which means that a small change in the initial conditions produces a totally different movement. To observe this phenomenon we plotted the angular position of the pendulum for 6 different angular positions close of 0.5 degrees, which means a change of 0.01 radians.

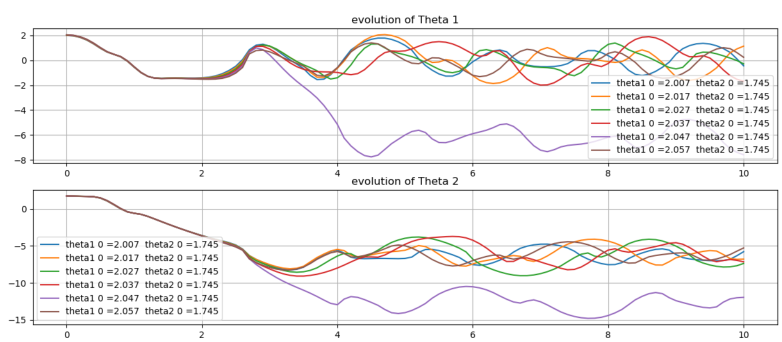


Figure 17: angular evolution with a change of 0.5° in theta1 0

We can see that just a simple change in the initial conditions modifies the dynamics of the entire system. This is a interesting way to observe the chaotic behavior of this system.

3.3.4 Impact of the steps resolution on the system dynamics

We wanted to visualise the impact of the step on the precision of movement. As seen before, we plotted theta 1 and theta 2 but the angular position is calculated with an adaptive step, we chose to start with a step of 0.003s and add 0.05s for each graph.

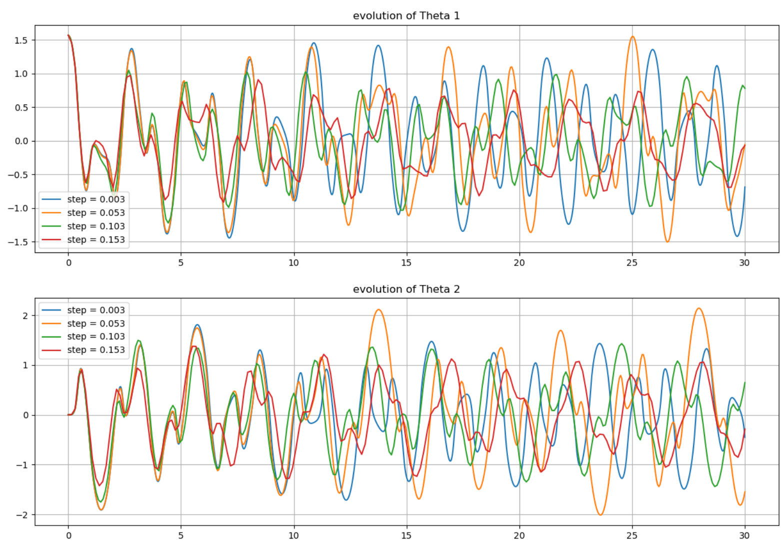


Figure 18: angular evolution with different steps

The precision decreases when the step increases, we lost a lot of precision between a step of 0.035 and 0.103 as we can see in the graph.

3.3.5 Impact of the resolution method on the dynamics of the system

We know that our resolution method is correct and precise because of the previous studies. However we may wonder if we could use a less precise resolution as the Euler resolution. To compare these methods we plotted theta1 and theta2 calculated with Runge-Kutta 4 and Euler. We used a step of 0.001 for the Euler resolution.

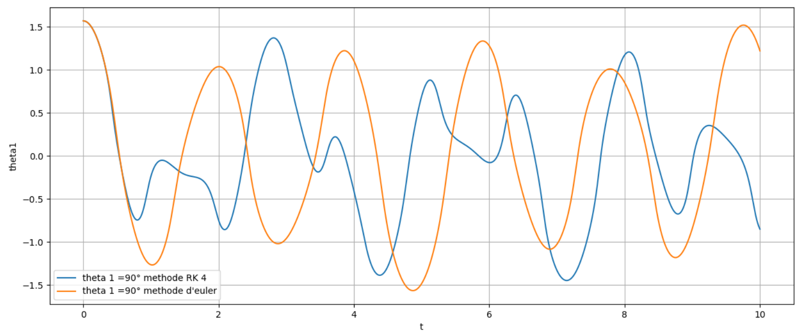


Figure 19: Theta1 during time with the Euler resolution and the RK 4 resolution

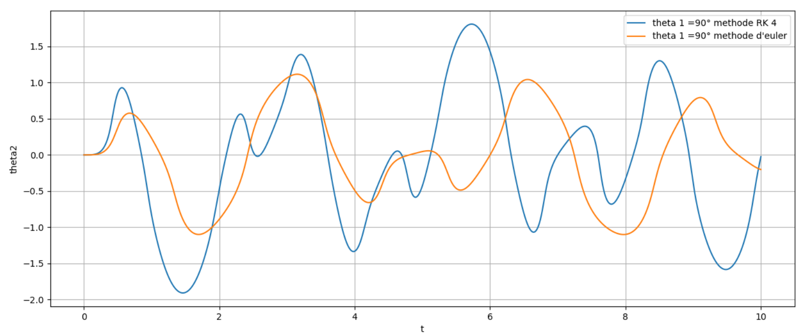


Figure 20: Theta2 during time with the Euler resolution and the RK 4 resolution

The two graphs show that a lot of information are lost in the Euler resolution. That's why it's more interesting to used the Runge Kutta method in order to obtain a more precis resolution.

3.4 Energy conservation

3.4.1 calculus and plotting of the mechanical energy

To validate our modeling, a possibility is to verify if the total energy of the system is conserved. The model could be considered as true if the total mechanical energy is constant. To calculate this energy of the system we need to calculate the mechanical energy of the two pendulums

$$\begin{aligned}
&E_c = \tfrac{1}{2}(m_1+m_2)l_1^2\dot{\theta}_1^2 + \tfrac{1}{2}m_2 l_2^2\dot{\theta}_2^2 + m_2 l_1 l_2 \dot{\theta}_1\dot{\theta}_2\cos(\theta_2-\theta_1)\\
&E_p = -(m_1+m_2)gl_1\cos\theta_1 - m_2 g l_2\cos\theta_2\\
&E_m = E_c + E_p = \tfrac{1}{2}(m_1+m_2)l_1^2\dot{\theta}_1^2 + \tfrac{1}{2}m_2 l_2^2\dot{\theta}_2^2 + m_2 l_1 l_2\dot{\theta}_1\dot{\theta}_2\cos(\theta_2-\theta_1) - (m_1+m_2)gl_1\cos\theta_1 - m_2 g l_2\cos\theta_2
\end{aligned} \tag{1}$$

We plot the mechanical energy for the three system studied in the part 3.3.1

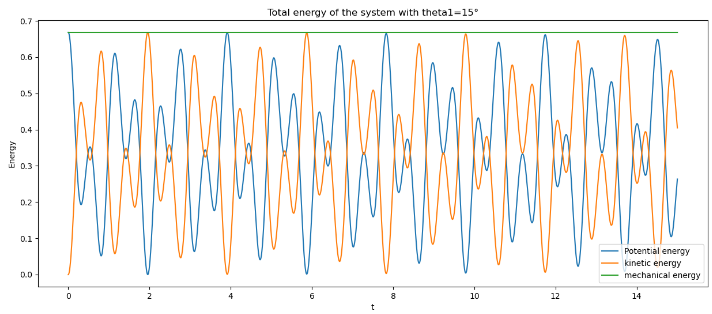


Figure 21: mechanical energy for theta1=15° theta2=0°

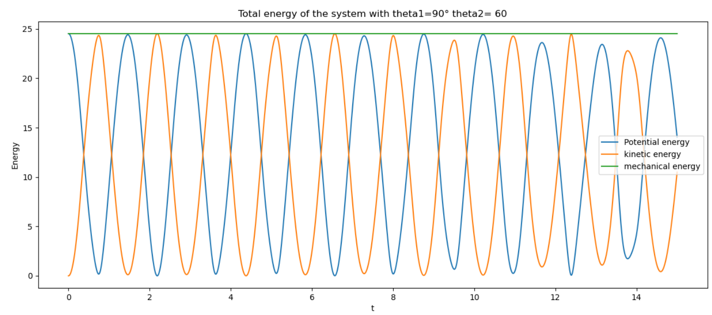


Figure 22: mechanical energy for theta1=90° theta2=60°

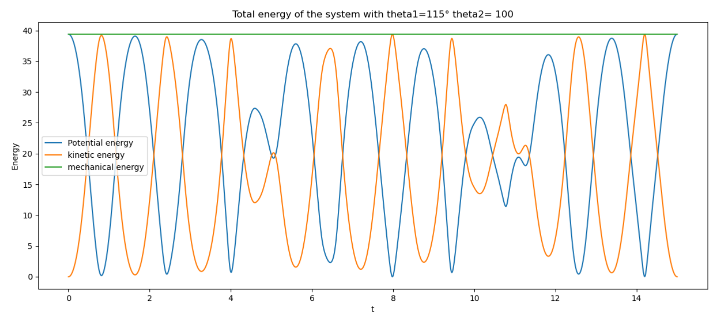


Figure 23: mechanical energy for theta1=115° theta2=100°

We observe that the energy is conserved which valid our modelisation.

3.4.2 Stability of the Runge kutta method on the energy

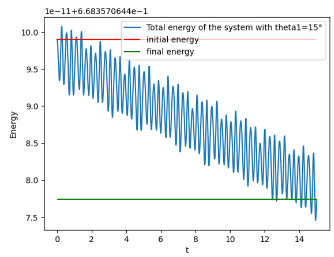


Figure 24: variation of the total energy for theta1=15° theta2=0°

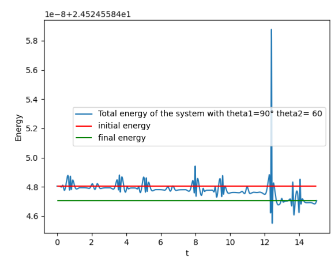


Figure 25: variation of the total energy for theta1=90° theta2=60°

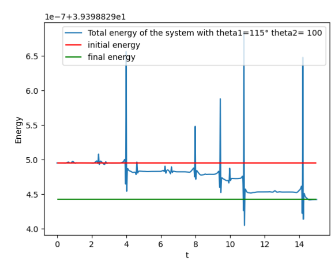


Figure 26: variation of the total energy for theta1=115° theta2=100°

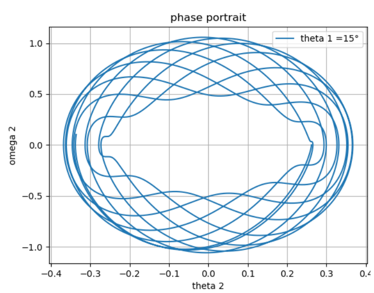
We can see that between the beginning and the end of the acquisition the total energy decrease these is due to an instability of the runge kutta method that is not precise at 100%.However this imprecision is infinitesimal regarding to the step of time used for the acquisition.

3.5 Phase portrait and Chaos characterisation

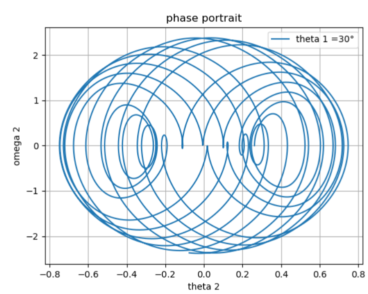
3.5.1 Utility of the phase portrait on the study of a chaotic behavior system

A phase portrait is a geometrical representation of the trajectories in a dynamic system. Each set of initial conditions corresponds to a curve in the $(\dot{\theta},\theta)$ space.

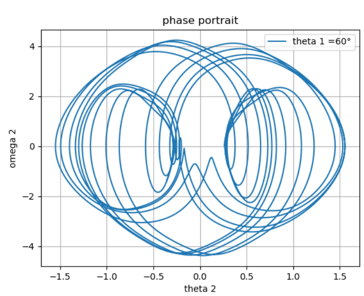
The phase Portrait is a great tool to study dynamic system because it can caracterise the presence of an attractor or a repulsor. An attractor is a set of state where a system tend to approach irreversibly without any perturbation. At the opposite, which is a repulsor is a set of states that the system tend to postpone. To describe our system we plotted the Portrait phase for five different position of theta 1 initial {15,30,60,90,140} degree.



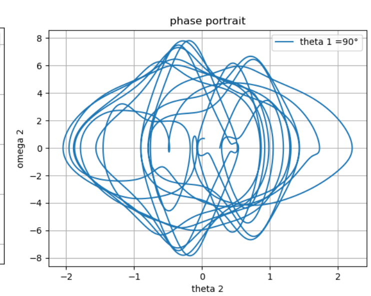
(a) initial theta1 =15°



(b) initial theta1 =30°



(a) initial theta1 =60°



(b) initial theta1 =90°

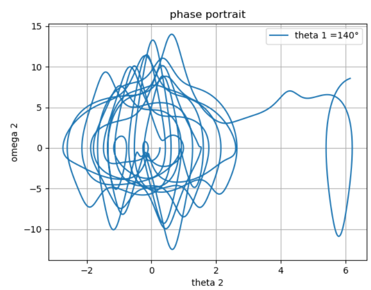


Figure 29: initial theta1 =140°

For small angle of theta1 the system turns around two points but never stabilise around them. When we increase the initial angle, we can see the chaotic behavior of the system. There is no state of stability that the system tend to approach. In conclusion the second pendulum follow a chaotic movement and more the initial angle is huge, more chaotic is the evolution.

3.5.2 Time for the second pendulum to make a full 360° turn

We want to calculate the time made by the second pendulum to make a full turn. We want to plot the result as a 2D color plot, where the x and the y axis are the initial condition on theta 1 and theta2. However to plot this graph we need to resolve the differential equation for every couple of initial condition (θ_1,θ_2). This will produce heavy computation and the smaller the steps is more time will be spend on the computation.

To solve this temporal problem we stop directly the resolution if the second pendulum made a full 360° turn. Also to made the computation more fast we calculate the energy needed to made a full turn. If the initial condition don't give enough energy to the pendulum to do a full turn the resolution is immediately stop. We can notice that the yellow mass at the center, because

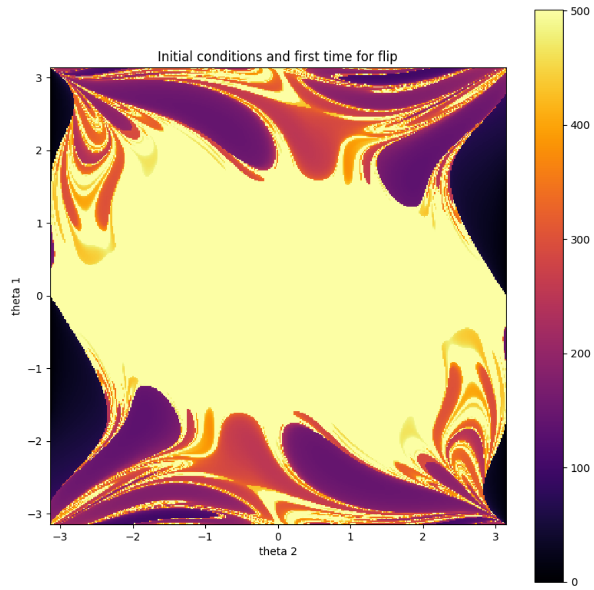


Figure 30: Time spend before first second pendulum flip for every couple of initial condition (colors in hundredth of a second, the more yellow, the more it take time to flip

4 Conclusion

To conclude this study, we can say that the double pendulum is not only a very interesting system because of its chaotic side, but also very remarkable because of the initial conditions which can completely modify its behavior.

5 Bibliography

Noémie Jaquier,”Le pendule double, un système chaotique”, from the website https://vdocuments.mx/le-pendule-double-un-systme-chaotique.html?page=1, dated 2009-2010

Boutrih Lhoucine, ”Le Pendule Double : Exemple D’un Système Sensible Aux Conditions Initiales”, dated 2013-2014

Rubin H.Landau, Manuel J.Paez, Cristian C. Bordeianu,Computational Physics,Problem Solving with Python, 3rd completely revised edition, p177-178

Every graph shown on this report was plot on Python by herself.