



UNIVERSIDAD DE GRANADA

GRADO EN FÍSICA

Física Computacional

Voluntario 1: Péndulo. Algoritmo de Runge-Kutta

José Álvarez García
2023/2024

Índice

1. Introducción	2
1.1. Desarrollo teórico. Ecuaciones del movimiento	2
2. Mapas de Poincaré	4
2.1. ϕ frente ψ	5
2.2. ϕ frente $\dot{\phi}$ y ψ frente $\dot{\psi}$	5
2.3. Oscilaciones pequeñas	6
3. Coeficientes de Lyapunov	7
3.1. Simulación	8
4. Pequeñas variaciones	9
5. Optimización	10
5.1. Comparación con el Superordenador Joel	10
6. Conclusión	11
7. Apéndices	13
7.1. Códigos	13
7.2. Características de mi PC	14
8. Bibliografía	15

Resumen

A lo largo de este informe se muestra la dinámica del sistema caótico del péndulo doble, realizando una simulación de este gracias al algoritmo de Runge-Kutta. Se realizarán diferentes análisis de este sistema físico junto con animaciones, además de un pequeño estudio de computación de alto rendimiento con el superordenador de la UGR

1. Introducción

Todos los programas utilizados para la realización del trabajo se encuentran en el repositorio [JoseAlvarez-Compu2324/Voluntario1](https://github.com/JoseAlvarez-Compu2324/Voluntario1) de GitHub.

1.1. Desarrollo teórico. Ecuaciones del movimiento

Un péndulo doble es un sistema compuesto por dos masas m_1 y m_2 unidas por dos alambres de longitud l_1 y l_2 , como se ve en la siguiente figura.

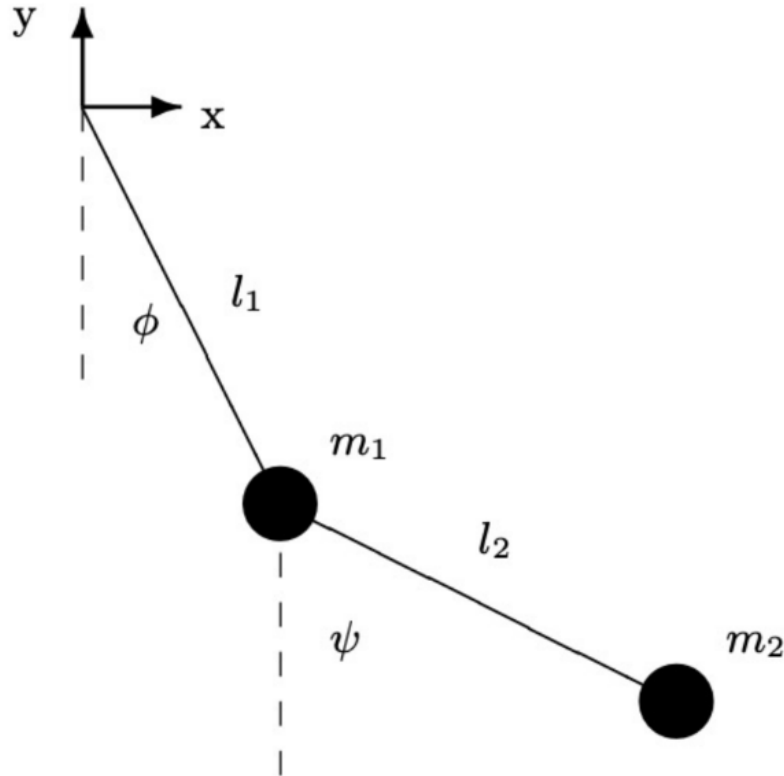


Figura 1: Esquema péndulo doble

Para encontrar unas ecuaciones de movimiento que podamos computar mediante el algoritmo de Runge-Kutta, para así simular el movimiento de este sistema, definimos las coordenadas generalizadas ϕ y ψ tal que cumplen:

$$\begin{aligned} x_1 &= l_1 \sin \phi & y_1 &= -l_1 \cos \phi \\ x_2 &= l_1 \sin \phi + l_2 \sin \psi & y_2 &= -l_1 \cos \phi - l_2 \cos \psi \end{aligned} \quad (1)$$

Donde $\{x_1, y_1\}, \{x_2, y_2\}$ hacen referencia a la posiciones de la masas 1 y 2 respectivamente.

El Lagrangiano del sistema valdrá:

$$\mathcal{L} = \frac{1}{2}(m_1 + m_2)l_1^2\dot{\phi}^2 + m_2l_1l_2\dot{\phi}\dot{\psi}\cos(\phi - \psi) + \frac{1}{2}m_2l_2^2\dot{\psi}^2 + (m_1 + m_2)gl_1\cos\phi + m_2gl_2\cos\psi \quad (2)$$

Para simplificar utilizaremos los parámetros $l_1 = l_2 = m_1 = m_2 = 1$. Quedando el Lagrangiano del sistema como:

$$\mathcal{L} = \dot{\phi}^2 + \frac{1}{2}\dot{\psi}^2 + \dot{\phi}\dot{\psi}\cos(\phi - \psi) + 2g\cos\phi + g\cos\psi \quad (3)$$

De la ecuación de Euler-Lagrange obtenemos las ecuaciones de movimiento (dos ecuaciones diferenciales de segundo orden):

$$\frac{d}{dt}\left(\frac{\partial\mathcal{L}}{\partial\dot{\phi}}\right) = \frac{\partial\mathcal{L}}{\partial\phi} \quad , \quad \frac{d}{dt}\left(\frac{\partial\mathcal{L}}{\partial\dot{\psi}}\right) = \frac{\partial\mathcal{L}}{\partial\psi} \quad (4)$$

Resolviendo nos queda:

Para ϕ :

$$2\ddot{\phi} + \ddot{\psi}\cos(\phi - \psi) - \dot{\psi}\sin(\phi - \psi)(\dot{\phi} - \dot{\psi}) = -\dot{\phi}\dot{\psi}\sin(\phi - \psi) - 2g\sin\phi \quad (5)$$

$$2\ddot{\phi} + \ddot{\psi}\cos(\phi - \psi) + \dot{\psi}^2\sin(\phi - \psi) + 2g\sin\phi = 0$$

Para ψ :

$$\ddot{\psi} + \ddot{\phi}\cos(\phi - \psi) - \dot{\phi}\sin(\phi - \psi)(\dot{\phi} - \dot{\psi}) = \dot{\phi}\dot{\psi}\sin(\phi - \psi) - g\sin\psi \quad (6)$$

$$\ddot{\psi} + \ddot{\phi}\cos(\phi - \psi) - \dot{\phi}^2\sin(\phi - \psi) + g\sin\psi = 0$$

Para el algoritmo de Runge-Kutta nos conviene expresar este sistema de 2 ecuaciones diferenciales de orden 2 en un sistema de 4 ecuaciones diferenciales de orden 1.

Definimos:

$$\begin{aligned} y_1 &= \phi \\ y_2 &= \psi \\ y_3 &= \dot{\phi} \\ y_4 &= \dot{\psi} \end{aligned} \quad (7)$$

Despejando $\ddot{\phi}$ y $\ddot{\psi}$ de las ecuaciones (5) y (6) respectivamente y haciendo dos sustituciones las ecuaciones del movimiento, podemos despejar $\ddot{\phi}$ y $\ddot{\psi}$ en términos de ϕ , ψ , $\dot{\phi}$ y $\dot{\psi}$.

Obtenemos entonces el siguiente sistema de 4 ecuaciones diferenciales.

$$\begin{aligned} y_1 &= \phi = y_3 \\ y_2 &= \psi = y_4 \\ y_3 &= \dot{\phi} = \frac{1}{2-\cos^2(y_1-y_2)} (gsiny_2\cos(y_1-y_2) - 2gsiny_1 - y_3^2\cos(y_1-y_2)\sin(y_1-y_2) - y_4^2\sin(y_1-y_2)) \\ y_4 &= \dot{\psi} = \frac{1}{1-\frac{1}{2}\cos^2(y_1-y_2)} (gsiny_1\cos(y_1-y_2) - gsiny_2 + \frac{1}{2}y_4^2\cos(y_1-y_2)\sin(y_1-y_2) + y_3^2\sin(y_1-y_2)) \end{aligned} \quad (8)$$

Este es sistema de ecuaciones diferenciales que utilizaremos para el desarrollo del algoritmo de Runge-Kutta, y teniendo en cuenta el cambio de variable definido en (7) obtendremos las coordenadas y velocidades angulares.

Para las condiciones iniciales calcularemos el Hamiltoniano del sistema mediante su definición:

$$\mathcal{H} = \sum_i p_i \dot{q}_i - \mathcal{L} = p_\phi \dot{\phi} + p_\psi \dot{\psi} - \mathcal{L} \quad (9)$$

con los momentos asociados a cada coordenada generalizada calculados como:

$$p_\phi = \frac{\partial \mathcal{L}}{\partial \dot{\phi}} = 2\dot{\phi} + \dot{\psi} \cos(\phi - \psi) \quad , \quad p_\psi = \frac{\partial \mathcal{L}}{\partial \dot{\psi}} = \dot{\psi} + \dot{\phi} \cos(\phi - \psi) \quad (10)$$

Obteniendo:

$$\mathcal{H} = \dot{\phi}^2 + \frac{1}{2}\dot{\psi}^2 + \dot{\phi}\dot{\psi}\cos(\phi - \psi) - 2g\cos\phi - g\cos\psi = E \quad (11)$$

que será igual a la energía total del sistema al tratarse de un sistema en un campo conservativo y no tener el Lagrangiano dependencia explícita con el tiempo $\frac{\partial \mathcal{L}}{\partial t} = 0$

De aquí encontramos la dependencia de $\dot{\phi}$ con la energía total del sistema

$$\dot{\phi} = \frac{1}{2} \left(-\dot{\psi} \cos(\phi - \psi) \pm \sqrt{\dot{\psi}^2 \cos^2(\phi - \psi) - 4\left(\frac{1}{2}\dot{\psi}^2 - 2g\cos\phi - g\cos\psi - E\right)} \right) \quad (12)$$

Donde el \pm indica si la velocidad angular $\dot{\phi}$ va en sentido horario ($-$) o antihorario ($+$)

* Nota: En nuestra elección de coordenadas la energía mínima que puede tener el sistema será $E_{min} = -3 \cdot g$, teniendo como condiciones iniciales $\dot{\phi} = 0$, $\dot{\psi} = 0$, $y_1 = -1$, $y_2 = -2$.

Utilizaremos $g = 1$ para simplificar las ecuaciones.

2. Mapas de Poincaré

Un mapa de Poincaré es una técnica usada en el análisis de sistemas dinámicos, especialmente en el estudio de sistemas caóticos. Este mapa, también conocido como sección de Poincaré, ayuda a visualizar y entender el comportamiento a largo plazo de un sistema dinámico en el espacio de fases.

Se realizan en este apartado 5 simulaciones con condiciones iniciales

- $E = \{1, 3, 5, 10, 15\}$
- $\phi = 45^\circ$
- $\psi = 45^\circ$
- $\dot{\psi} = 0$
- $\dot{\phi}$ dada por la ecuación (12)

Los resultados se muestran en las animaciones "*animacion_con_mapa_i.mp4*" donde i hace referencia a la energía del sistema. Las animaciones se pueden encontrar en el repositorio, accediendo a través del siguiente [enlace](#). Se recomienda su visualización para mejor comprensión del problema.

Se hace un análisis de la trayectoria en el espacio de fases, comparados los casos de diferentes energías en un mismo gráfico con el programa "mapas_poincare.py".

En todos los casos el tiempo transcurrido es $T = 100$ y una precisión $h = 0,01$, es decir, un total de 10000 iteraciones

2.1. ϕ frente ψ

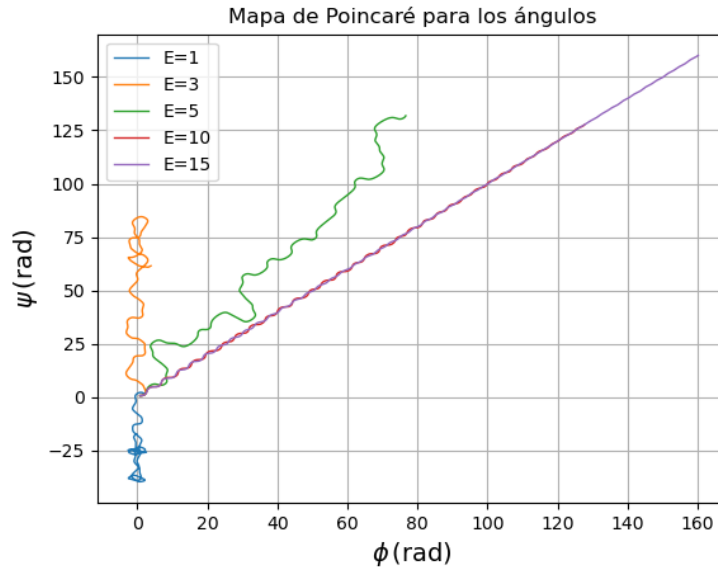


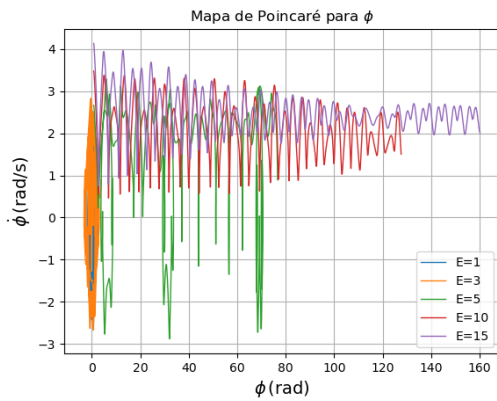
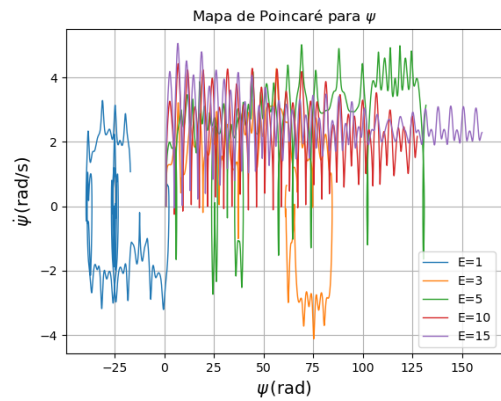
Figura 2: Mapa de Poincaré de los ángulos ϕ y ψ

En esta gráfica vemos la relación entre las dos coordenadas generalizadas en las que se aprecia como para bajas energías ($E = \{1, 3\}$) el movimiento se encuentra confinado en un recinto del espacio de fases, característica de un sistemas con cierta periodicidad en sus coordenadas.

Para energías más altas ($E = \{5, 10, 15\}$), las dos masas del péndulo dan un número mayor de vueltas o incluso no dejan de girar, es decir, el ángulo recorrido aumenta y el movimiento en el espacio de fases no se queda confinado. Cabe destacar como para $E = 15$ cuanto más aumenta el tiempo, la trayectoria en el espacio fásico se aproxima a la recta $\phi = \psi$, comportándose el péndulo doble como un péndulo simple ya que al estar las masas acopladas, estas entran en resonancia.

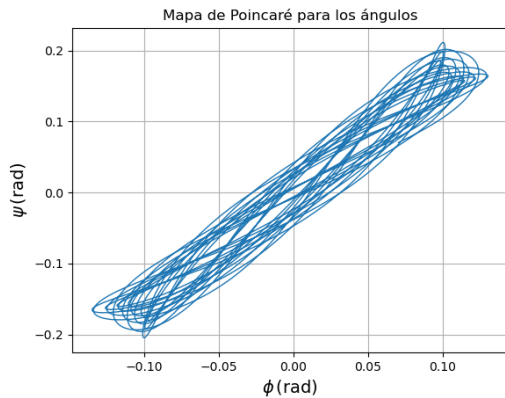
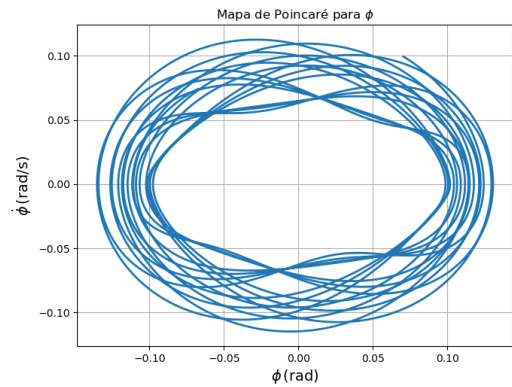
2.2. ϕ frente $\dot{\phi}$ y ψ frente $\dot{\psi}$

Como se ha comentado anteriormente para energías altas la trayectoria en el espacio de fases avanza constantemente en el eje X mientras que para energías bajas "orbitan" en torno a un punto del espacio de fases. Esto se conoce como **atractor**, un punto en el espacio fásico en torno al que se crean órbitas de las trayectorias. Para $E = 5$ es destacable como en el espacio $(\phi, \dot{\phi})$ se combinan movimientos periódicos en los que se orbita alrededor de un atractor, y movimientos en los que la trayectoria avanza a lo largo del eje X .

(a) Mapa de Poincaré de ϕ con $\dot{\phi}$ (b) Mapa de Poincaré de ψ con $\dot{\psi}$ **Figura 3:** Mapas de Poincaré con condiciones iniciales $\phi = \psi = 45^\circ$, $\dot{\phi} = \dot{\psi} = 0$

2.3. Oscilaciones pequeñas

Realizamos ahora el estudio de los espacios de fases para unas condiciones iniciales $\phi = \psi = 4^\circ$ y $\dot{\phi} = \dot{\psi} = 0,1$, donde tanto las coordenadas angulares como las velocidades son iguales y pequeñas.

(a) Mapa de Poincaré de los ángulos ϕ y ψ (b) Mapa de Poincaré de ϕ con $\dot{\phi}$ **Figura 4:** Mapas de Poincaré con condiciones iniciales $\phi = \psi = 4^\circ$, $\dot{\phi} = \dot{\psi} = 0,1$

Se observa en este caso un movimiento periódico en ambos ángulos, en el que en todo momento $\phi \approx \psi$ y el ángulo ϕ describe una trayectoria en el espacio de fases muy similar a la del péndulo simple.

Concluimos que la dinámica del péndulo doble para oscilaciones pequeñas tiene bastante similitud con la del péndulo simple.

3. Coeficientes de Lyapunov

Los coeficientes o exponentes de Lyapunov en un sistema dinámico cuantifican la separación entre dos trayectorias en el espacio de fases a lo largo del tiempo. La cantidad de coeficientes corresponderá a la dimensión n del espacio de fases.

Los sistemas caóticos se caracterizan por la divergente separación entre trayectorias en el espacio de fases que inicialmente eran muy similares (es decir, que tenían condiciones iniciales parecidas). Estos coeficientes λ_L serán la herramienta capaz de cuantificar esta divergencia entre trayectorias. La separación en el espacio de las fases en función del tiempo de diferentes trayectorias la denominaremos $\delta(t)$. La relación a estudiar es la siguiente:

$$\|\delta(t)\| = e^{\lambda_L t} \|\delta(t_0)\| \quad (13)$$

Donde $\delta(t_0)$ es la separación inicial en el espacio-fase.

Discretizando el tiempo, podemos encontrar una expresión para obtener los λ_L tomando logaritmos a ambos lados y convirtiendo la variable t en variaciones temporales (con $t_0 = 0$ en nuestro caso).

$$\lambda_L = \lim_{m \rightarrow \infty} \frac{1}{m} \sum_{i=1}^m \ln \left(\frac{\delta(t_i)}{\delta(t_0)} \right) \quad (14)$$

Donde m es el número total de iteraciones.

En nuestro sistema dinámico el espacio de fases tiene 4 dimensiones $(\phi, \dot{\phi}, \psi, \dot{\psi})$

Para estudiar el coeficiente de Lyapunov, se escogerán unas condiciones iniciales $(\phi_0, \dot{\phi}_0, \psi_0, \dot{\psi}_0)$ y otras condiciones iniciales, separadas infinitesimalmente de las anteriores $(\phi'_0, \dot{\phi}'_0, \psi'_0, \dot{\psi}'_0)$

La distancia en el espacio de fases para un instante t_i será la euclídea:

$$\delta(t_i) = \sqrt{(\phi_i - \phi'_i)^2 + (\psi_i - \psi'_i)^2 + (\dot{\phi}_i - \dot{\phi}'_i)^2 + (\dot{\psi}_i - \dot{\psi}'_i)^2} \quad (15)$$

Podemos calcular a su vez coeficientes de Lyapunov para cada una de las coordenadas de forma aproximada como:

$$\lambda_j = \lim_{m \rightarrow \infty} \frac{1}{m} \sum_{i=1}^m \ln \left(\frac{\delta_j(t_i)}{\delta_j(t_0)} \right) \quad (16)$$

Donde el subíndice j hace referencia a cada una de las coordenadas del espacio de fases las distancias se calculan como:

$$\begin{aligned} \delta_\phi(t_i) &= |\phi_i - \phi'_i| \\ \delta_\psi(t_i) &= |\psi_i - \psi'_i| \\ \delta_{\dot{\phi}}(t_i) &= |\dot{\phi}_i - \dot{\phi}'_i| \\ \delta_{\dot{\psi}}(t_i) &= |\dot{\psi}_i - \dot{\psi}'_i| \end{aligned} \quad (17)$$

El valor de estos coeficientes nos arrojan una descripción cuantitativa del comportamiento caótico del sistema, de esta manera, si $\lambda_j < 0$ las trayectorias en el espacio de fases convergen, y si $\lambda_j > 0$ estas divergen, indicador del caos. Si $\lambda_j = 0$ significará que la distancia entre trayectorias no cambia, es decir, existe una separación constante.

3.1. Simulación

En nuestra simulación tendremos como condiciones iniciales $\phi_0 = 45,0^\circ$, $\psi_0 = 45,0^\circ$, $\dot{\psi}_0 = 0$ y $\dot{\phi}_0$ dada por la ecuación (12), se utilizarán de igual forma las energías $E = \{1, 3, 5, 10, 15\}$

Para el caso perturbado se utilizan las condiciones iniciales $\phi'_0 = 50,0^\circ$; $\psi'_0 = 5,0^\circ$; $\dot{\psi}'_0 = 0,1$, con un tiempo transcurrido $T = 10^5$ y una precisión $h = 0,01$, es decir, un total de 10^7 iteraciones.

Obtenemos los siguientes resultados con el programa "calcula_lyapunov.ipynb":

E	λ_L	λ_ϕ	λ_ψ	$\lambda_{\dot{\phi}}$	$\lambda_{\dot{\psi}}$
1	6.12	$-\infty$	6.90	$-\infty$	$-\infty$
3	9.64	10.11	9.93	$-\infty$	$-\infty$
5	9.14	9.01	9.76	$-\infty$	$-\infty$
10	9.57	8.82	10.21	$-\infty$	$-\infty$
15	9.02	8.62	9.64	$-\infty$	$-\infty$

Tabla 1: Coeficientes de Lyapunov

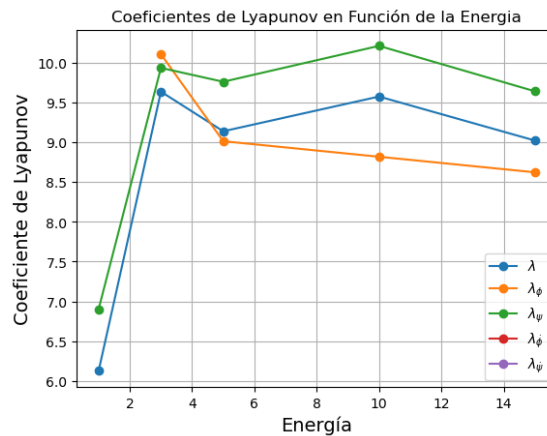


Figura 5: Coeficientes de Lyapunov

El coeficiente de Lyapunov del sistema λ_L representado en color azul en la figura 5 es para todos los valores de la energía mayor que 0, indicando el comportamiento caótico del péndulo doble.

Para las coordenadas angulares (ϕ, ψ) se observa también un comportamiento caótico, mucho más destacable en la coordenada ψ , la cual es más susceptible a las condiciones iniciales.

Destacamos además como para los coeficientes $\lambda_{\dot{\phi}}$, $\lambda_{\dot{\psi}}$ obtenemos siempre valores $-\infty$ que, fijándonos en la ecuación (14), podemos deducir que la distancia δ en el espacio fásico entre las dos trayectorias en un momento t_i es nula. Esto puede deberse a las limitaciones computacionales y a como el programa aproxima distancias muy pequeñas a 0. Lo que quieren decir estos coeficientes es que el comportamiento caótico no es visible en las coordenadas de velocidad angular $(\dot{\phi}, \dot{\psi})$

4. Pequeñas variaciones

En esta sección veremos de forma gráfica el carácter caótico de este sistema.

Se simulan dos sistemas con una pequeña variación en las condiciones iniciales y se observará como la trayectoria del péndulo varía drásticamente cuando se deja pasar un tiempo lo suficientemente grande.

Para la primera simulación se toman como condiciones las utilizadas en apartados anteriores: $\phi_0 = 45,0^\circ$, $\psi_0 = 45,0^\circ$, $\dot{\psi}_0 = 0$ y $\dot{\phi}_0$ dada por la ecuación (12) con energía $E = 3$. Para la segunda simulación tan solo variaremos en $0,1^\circ$ ambos ángulos iniciales, teniendo $\phi'_0 = 45,1^\circ$, $\psi'_0 = 45,1^\circ$, $\dot{\psi}'_0 = 0$ y $E' = 0$.

El resultado se puede ver en la animación "*pendulo_comparacion.mp4*" en la carpeta [Animaciones](#) dentro del repositorio y algunos frames en la siguiente figura:

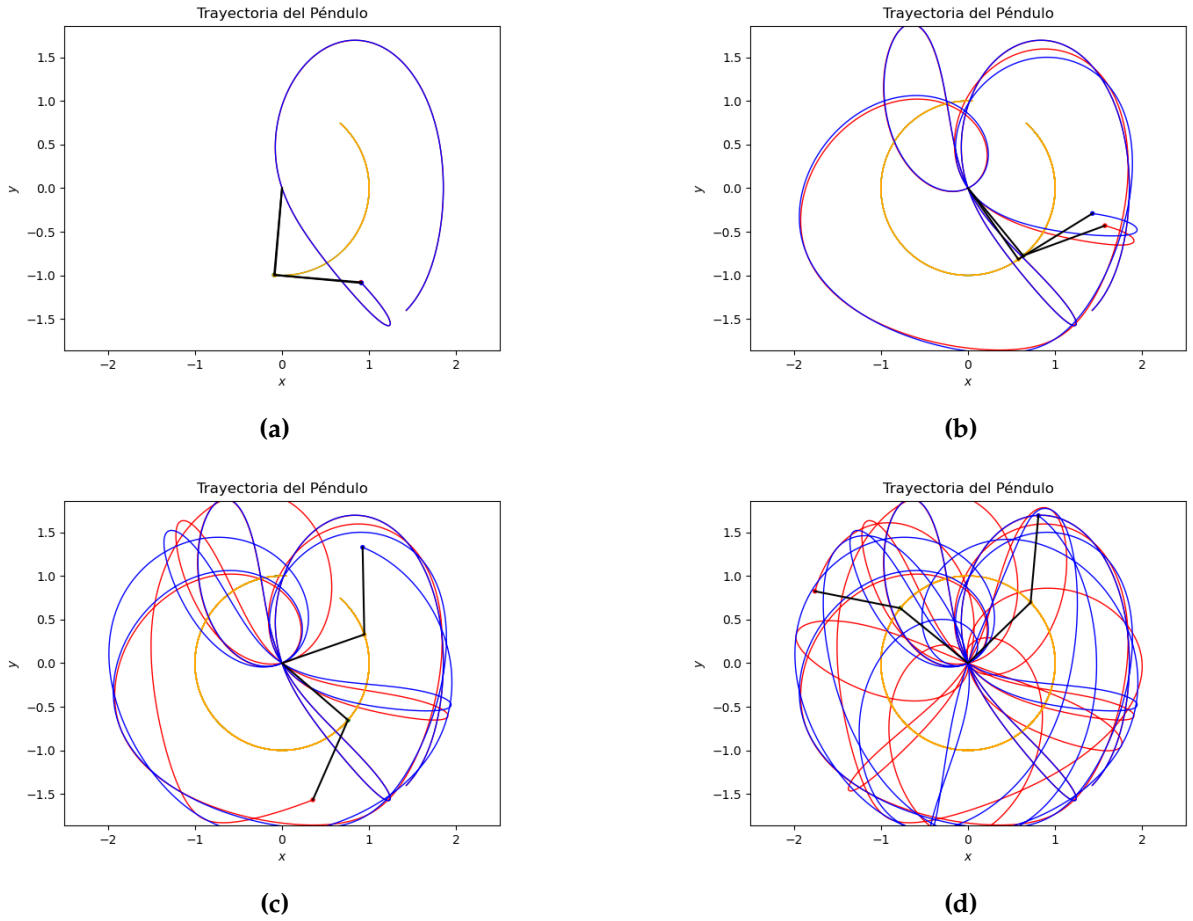


Figura 6: Evolución temporal del dos péndulos con condiciones iniciales distintas

5. Optimización

En esta sección obtendremos tiempos de ejecución del código "*pendulo_doble.c*" en función del tiempo transcurrido en la simulación T y la precisión h de este.

Los datos se recogen en la siguiente tabla:

T	h	Iteraciones $\frac{T}{h}$	Tiempo de ejecución (s)
10	0,01	1000	$0,25 \pm 0,05$
1000	0,01	10^5	$1,33 \pm 0,15$
10^4	0,01	10^6	$9,46 \pm 0,86$
$5 \cdot 10^4$	0,01	$5 \cdot 10^6$	$36,35 \pm 2,04$
10^7	0,01	10^7	$74,22 \pm 12,53$
10^6	0,01	10^8	$738,60 \pm 19,52$

Tabla 2: Tiempos de ejecución del programa con PC

5.1. Comparación con el Superordenador Joel

Para esta ultima sección comparamos los tiempos de ejecución del programa en el superordenador con los obtenidos en mi PC obtenidos en la sección anterior.

El procedimiento para obtener estos resultados ha sido:

- Subir el programa a mi repositorio por medio de GitHub (git push).
- Conectarse a Joel con clave ssh
- Descargar el programa en el superordenador desde el repositorio en GitHub (git pull)
- Ejecutar el programa desde la terminal con el comando "*gcc pendulo_doble.c -o pendulo_doble*"
- Se obtienen los resultados del tiempo de ejecución por pantalla

Se recogen los resultado obtenidos en la siguiente tabla

T	h	Iteraciones $\frac{T}{h}$	Tiempo de ejecución (s)
10	0,01	1000	$0,19 \pm 0,04$
1000	0,01	10^5	$1,35 \pm 0,15$
10^4	0,01	10^6	$6,65 \pm 0,94$
$5 \cdot 10^4$	0,01	$5 \cdot 10^6$	$54,20 \pm 2,17$
10^7	0,01	10^7	$154,26 \pm 11,17$
10^6	0,01	10^8	$1557,48 \pm 20,63$

Tabla 3: Tiempos de ejecución del programa con superordenador Joel

Han existido diferentes complicaciones para hacer esta sección como el apagón que dejó al superordenador **Proteus** completamente inútil durante un periodo de tiempo y la larga espera para conectarse a Joel, seguramente por la demanda de computo por parte del resto de alumnos de la asignatura y la distancia física entre mi PC y Proteus, llegando a esperar 30 aproximadamente minutos tan solo para tener acceso.

Se comparan los resultados de ambos ordenadores en la siguiente figura:

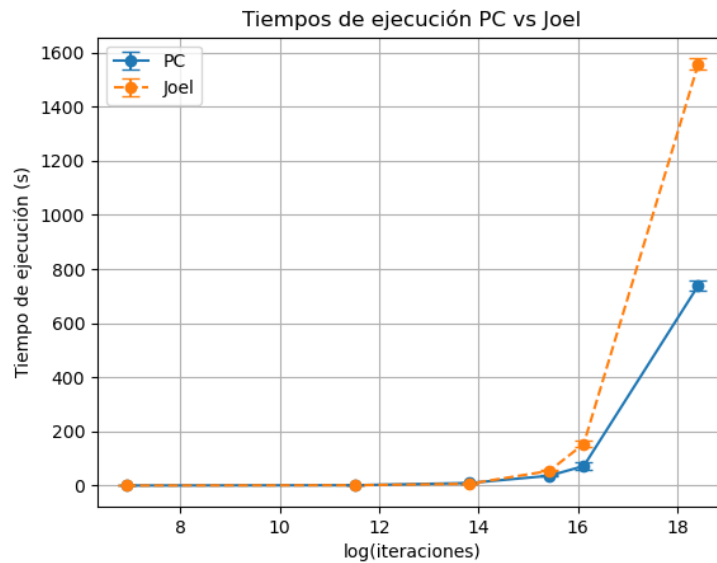


Figura 7: Tiempos ejecución en mi ordenador personal Vs en el superordenador

Los resultados obtenidos muestran como, la tediosa tarea que resulta trabajar con Joel, dadas las complicaciones previamente mencionadas, no proporciona mayor capacidad de computo en mi caso y no aporta una mejor realización de la simulación, sin embargo no lo considero una perdida de tiempo ya que he podido aprender como trabajar con repositorios y ordenadores remotos, y saber como operar en el sistema operativo Linux.

6. Conclusión

En este trabajo, hemos explorado la dinámica compleja del péndulo doble mediante simulaciones numéricas utilizando el método de Runge-Kutta de cuarto orden. A través del desarrollo teórico, hemos derivado las ecuaciones del movimiento para este sistema y procedimos a resolverlas numéricamente bajo ciertas condiciones simplificadas.

Las simulaciones realizadas han demostrado de manera efectiva el comportamiento caótico del péndulo doble, caracterizado por su alta sensibilidad a las condiciones iniciales. Este comportamiento se ha analizado en detalle utilizando herramientas como los mapas de Poincaré y los coeficientes de Lyapunov. Los mapas de Poincaré nos han permitido visualizar las trayectorias en el espacio de fases, revelando la estructura compleja del sistema. Por su parte, los coeficientes de Lyapunov han proporcionado una medida cuantitativa del caos, confirmando la presencia de exponentes positivos que indican sensibilidad exponencial a las condiciones iniciales.

Además, se ha examinado cómo pequeñas variaciones en las condiciones iniciales pueden llevar a trayectorias divergentes, enfatizando la naturaleza impredecible del sistema a largo plazo. Estos hallazgos son consistentes con la teoría del caos y refuerzan la comprensión de cómo los sistemas dinámicos no lineales pueden comportarse de manera aparentemente errática aunque sean deterministas.

El uso del método de Runge-Kutta ha demostrado ser una herramienta robusta y precisa para la simulación de sistemas dinámicos complejos como el péndulo doble. La precisión de este método ha permitido obtener resultados detallados y fiables, facilitando un análisis profundo del comportamiento del sistema.

En resumen, este trabajo no solo ha permitido una comprensión más profunda de la dinámica del péndulo doble, sino que también ha demostrado la importancia y utilidad de los métodos numéricos en el estudio de sistemas caóticos. Las técnicas y enfoques utilizados aquí pueden extenderse a otros sistemas dinámicos no lineales, proporcionando una base sólida para futuros estudios e investigaciones en este campo.

7. Apéndices

7.1. Códigos

Se explican a continuación cada uno de los códigos utilizados en este informe

- **"pendulo_doble.c"**

Este código es el principal del informe, con él se calculan las posiciones del péndulo en cada instante.

Se definen las funciones $y_i = f(y_1, y_2, y_3, y_4)$ para aplicar el algoritmo de Runge-Kutta segun las ecuaciones (8).

Los parámetros que podemos cambiar son: Ángulos iniciales ϕ_0 , ψ_0 , velocidad angular inicial $\dot{\phi}_0$, energía del sistema E , dirección de la velocidad angular inicial $\dot{\phi}_0$, tiempo total transcurrido T y precisión de la simulación h .

Se realiza el algoritmo de Runge-Kutta y se almacenan los ángulos y velocidades angulares para cada instante en diferentes variables.

El programa devuelve 5 ficheros de datos en los que se incluyen posiciones x e y de ambas masas del péndulo, ángulos ϕ y ψ y velocidades angulares $\dot{\phi}$ y $\dot{\psi}$ en cada iteración temporal.

Estos ficheros se utilizan en el resto de códigos para realizar las animaciones, gráficas y cálculos pertinentes.

- **"animacion_con_mapa.py"**

Este código crea animaciones como *subplots* para comparar el movimiento real del péndulo doble con el movimiento de las trayectorias en los diferentes mapas de Poincaré.

Se recogen los datos de 4 ficheros de datos que contienen: posiciones x e y de ambas masas del péndulo; ángulos ϕ y ψ ; ángulo ϕ junto a su velocidad angular $\dot{\phi}$; y ángulo ψ junto a su velocidad angular $\dot{\psi}$.

El programa devuelve un archivo *.mp4* o muestra en pantalla un *plot* con las 4 animaciones.

- **"mapas_poincare.py"**

Este código se encarga de realizar los gráficos de la sección 2.

Se recogen los conjuntos de datos que se quieran mostrar y devuelve por pantalla un *plot* con con el mapa de Poincaré correspondiente .

- **"calculo_lyapunov.py"**

Este código realiza los cálculos pertinentes para encontrar los coeficientes de Lyapunov del apartado 3.

Se recogen los conjuntos de datos del espacio fásico de dos configuraciones iniciales distintas y se devuelven por pantalla los 5 coeficientes de Lyapunov en función de la energía y una gráfica que recoge todos ellos.

- **"animacion2_pendolo.py"**

Este código crea una animación del movimiento real de dos péndulos en un mismo *plot* con el objetivo de compararlos entre si.

Se recogen los datos de 2 ficheros de datos que incluyen las posiciones x e y de las dos masas para dos condiciones iniciales distintas.

El programa devuelve un archivo *.mp4* o muestra en pantalla un *plot* con la animación de ambos péndulos.

■ **"animacion_pendolo.py"**

Este código no se ha usado directamente en la realización del informe aunque ha servido para llegar a los resultados. Funciona igual que el anterior pero este solo recoge datos y devuelve la animación de un solo péndulo doble

7.2. Características de mi PC

- **Procesador:** Intel® Core™ i7-10870H (Caché: 16MB SmartCache, 2.20GHz hasta 5.00GHz, 64-bit)
- **Memoria RAM:** 16GB (8GB*2) DDR4 3200MHz
- **Disco duro:** 1TB SSD M.2 NVMe™ PCIe® 3.0
- **Controlador gráfico:** NVIDIA® GeForce® GTX 1650/ 4GB GDDR6 VRAM

8. Bibliografía

- [1] Exponente de lyapunov.
- [2] J. Bermejo Vega. Física computacional. algoritmo runge-kutta, 2024.
- [3] H. Sayama. 9.3: Exponente de lyapunov.