

Detalles Matemáticos del Código

Consideremos la ecuación presentada en el documento (4):

$$u'' + (1 + \epsilon\lambda\omega^2 \cos(\omega\tau)) \left(u - \frac{\epsilon^2 u^3}{6} \right) = 0. \quad (1)$$

Donde:

$$\begin{aligned} \tau &= \omega t, \\ \omega_0 &= \frac{g}{r_0}, \\ \epsilon\lambda &= \frac{\delta z}{r_0}, \\ \omega &= \frac{\omega}{\omega_0}. \end{aligned}$$

Después de realizar con detalle los pasos para convertir la ecuación diferencial (4) en un sistema de EDOs, vamos a utilizar el ****método de Euler mejorado**** (también conocido como ****método de Heun****) para aproximar las soluciones.

El sistema de EDOs resultante es:

$$\begin{cases} v_1' = v_2, \\ v_2' = -(1 + \epsilon\lambda\omega^2 \cos(\omega\tau)) \left(v_1 - \frac{\epsilon^2 v_1^3}{6} \right). \end{cases} \quad (2)$$

Donde $v_1 = u$ y $v_2 = u'$.

Implementación Numérica

Para resolver este sistema con el ****método de Heun****, definimos las siguientes variables iniciales:

$$\begin{aligned} \epsilon &= 0.5, \\ \lambda &= 1.0, \\ \Omega &= 1.96, \\ h & \text{ (paso temporal),} \\ t_{\text{inicio}} &= 0, \\ t_{\text{parada}} &= 50.0, \\ \omega_0 & \text{ (constante de frecuencia angular).} \end{aligned}$$

Los valores iniciales son $(0, 0.1)$, es decir, la posición angular es 0 y la velocidad inicial es 0.1.

Definimos dos funciones f y g que corresponden a las ecuaciones (1) y (2):

$$\begin{cases} u^* = u_n + hf(t_n, u_n, v_n), \\ v^* = v_n + hg(t_n, u_n, v_n). \end{cases} \quad (3)$$

Estas son las **sucesiones predictoras**, que nos permiten aproximar la posición y la velocidad angular en el tiempo t_n . Luego, aplicamos la corrección:

$$\begin{cases} u_{n+1} = u_n + \frac{h}{2} (f(t_n, u_n, v_n) + f(t_n + h, u^*, v^*)), \\ v_{n+1} = v_n + \frac{h}{2} (g(t_n, u_n, v_n) + g(t_n + h, u^*, v^*)). \end{cases} \quad (4)$$

En el código, para que esto funcione iterativamente, creamos un ciclo **while** cuya condición de parada es $t_{\text{parada}} = 50.0$. Este ciclo ejecuta las actualizaciones y almacena los datos en un vector para analizar los cambios en la solución.

Ejecutando el código podemos ver que la siguiente grafica que muestra como varia la velocidad y posicion angular en el tiempo t

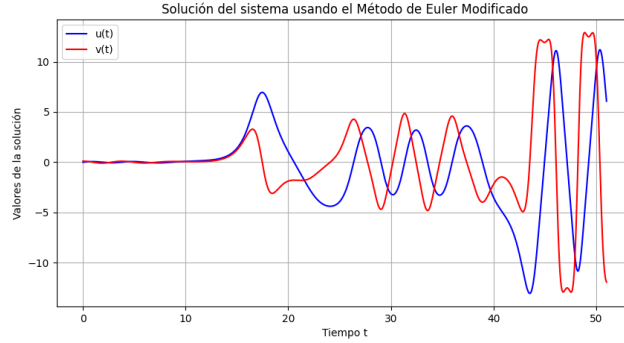


Figure 1: Posicion angular vs velocidad angular