Simulación Numérica de un Sistema Masa-Resorte-Amortiguador mediante el Método de Runge-Kutta (RK4)

Nombre del Estudiante 1
Nombre del Estudiante 2

30 de diciembre de 2025

Resumen

En este documento se describe el desarrollo e implementación de un simulador físico para un sistema de oscilación amortiguada. Se detalla el modelo matemático basado en la Segunda Ley de Newton y la Ley de Hooke, así como la transformación de la ecuación diferencial de segundo orden en un sistema de primer orden. Finalmente, se explica la integración numérica utilizando el método de Runge-Kutta de cuarto orden (RK4) implementado en lenguaje C para la visualización en tiempo real de los regímenes de amortiguamiento.

Índice

1. Introducción

El estudio de las vibraciones mecánicas es fundamental en ingeniería civil, mecánica y automotriz. Desde el diseño de suspensiones de vehículos hasta la protección sísmica de edificios, comprender cómo se disipa la energía en un sistema oscilatorio es crucial.

El objetivo de este proyecto es simular numéricamente la evolución temporal de un sistema de un grado de libertad compuesto por una masa, un resorte elástico y un amortiguador viscoso, permitiendo la modificación interactiva de sus parámetros físicos (m, k, c).

2. Modelo Físico y Matemático

2.1. Planteamiento del Problema

Consideramos un cuerpo de masa m sujeto a un resorte con constante de rigidez k y un amortiguador con coeficiente de viscosidad c. El sistema se mueve en una sola dimensión (eje vertical y).

Las fuerzas que actúan sobre la masa son:

- **Fuerza del Resorte (Ley de Hooke):** Se opone al desplazamiento.

$$F_k = -k \cdot y$$

- **Fuerza del Amortiguador:** Se opone a la velocidad.

$$F_c = -c \cdot v = -c \cdot \frac{dy}{dt}$$

Aplicando la Segunda Ley de Newton ($\sum F = m \cdot a$), obtenemos la ecuación de movimiento:

$$m\frac{d^2y}{dt^2} = -ky - c\frac{dy}{dt} \tag{1}$$

Reordenando los términos, llegamos a la Ecuación Diferencial Ordinaria (EDO) lineal homogénea de segundo orden:

$$m\ddot{y} + c\dot{y} + ky = 0 \tag{2}$$

2.2. Reducción de Orden

Los métodos numéricos estándar, como Runge-Kutta, están diseñados para resolver ecuaciones de primer orden de la forma $\dot{\mathbf{y}} = f(t, \mathbf{y})$. Para resolver la ecuación (2), debemos reducir su orden definiendo un **Vector de Estado**.

Definimos el estado del sistema $\mathbf{S}$ como:

$$\mathbf{S} = \begin{bmatrix} y \\ v \end{bmatrix} \tag{3}$$

Donde y es la posición y v es la velocidad ($\dot{y}$).

Al derivar el vector de estado respecto al tiempo, obtenemos:

$$\frac{d\mathbf{S}}{dt} = \begin{bmatrix} \dot{y} \\ \dot{v} \end{bmatrix} = \begin{bmatrix} v \\ a \end{bmatrix} \tag{4}$$

Sustituyendo la aceleración a despejada de la ecuación (2), obtenemos el sistema de primer orden acoplado:

$$\frac{d}{dt} \begin{bmatrix} y \\ v \end{bmatrix} = \begin{bmatrix} v \\ \frac{-ky-cv}{m} \end{bmatrix} \tag{5}$$

Este sistema es el que se implementa computacionalmente.

3. Método Numérico: Runge-Kutta 4 (RK4)

Para obtener la posición y velocidad en el instante $t_{n+1} = t_n + h$ (donde h es el paso de tiempo o Δt), utilizamos el método RK4. Este método es superior al método de Euler porque evalúa la pendiente en cuatro puntos distintos del intervalo, reduciendo el error de truncamiento local a $O(h^5)$.

Dada la ecuación de estado $\dot{\mathbf{S}} = f(t, \mathbf{S})$, el procedimiento es:

3.1. Cálculo de las Pendientes (k)

Debemos calcular cuatro vectores auxiliares (k_1, k_2, k_3, k_4). Cada k tiene dos componentes: una para la posición (dy) y otra para la velocidad (dv).

1. **Inicio del intervalo (k_1):** Pendiente en el estado actual.

$$k_1 = f(t_n, \mathbf{S}_n)$$

2. **Punto medio A (k_2):** Pendiente estimada a mitad del paso usando k_1.

$$k_2 = f(t_n + \frac{h}{2}, \mathbf{S}_n + k_1 \frac{h}{2})$$

3. **Punto medio B (k_3):** Pendiente mejorada a mitad del paso usando k_2.

$$k_3 = f(t_n + \frac{h}{2}, \mathbf{S}_n + k_2 \frac{h}{2})$$

4. **Final del intervalo (k_4):** Pendiente estimada al final del paso.

$$k_4 = f(t_n + h, \mathbf{S}_n + k_3 h)$$

3.2. Actualización del Estado

El nuevo estado se calcula como un promedio ponderado de estas pendientes:

$$\mathbf{S}_{n+1} = \mathbf{S}_n + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4) \tag{6}$$

4. Implementación en Código

En el software desarrollado, la función `CalcularAceleracion` representa la parte física del modelo:

```
float a = (-k * y - c * v) / m;
```

Mientras que la función `PasoRK4` realiza la integración vectorial descrita anteriormente. Se ha añadido la capacidad de aplicar impulsos externos (fuerzas instantáneas), lo que matemáticamente se modela como un cambio discontinuo en la velocidad:

$$v(t^+) = v(t^-) + \frac{J}{m}$$

Donde J es el impulso aplicado.

5. Análisis de Resultados: Regímenes de Amortiguamiento

El simulador permite visualizar en tiempo real los tres comportamientos posibles dependiendo del factor de amortiguamiento ζ:

$$\zeta = \frac{c}{2\sqrt{mk}}$$

- **Sub-amortiguado ($\zeta < 1$):** El sistema oscila con una amplitud que decae exponencialmente. Es el comportamiento por defecto de resortes ligeros.
- **Sobre-amortiguado ($\zeta > 1$):** No hay oscilación. El sistema retorna al equilibrio lentamente. Típico de mecanismos de seguridad o puertas hidráulicas.
- **Amortiguamiento Crítico ($\zeta = 1$):** El sistema retorna al equilibrio lo más rápido posible sin oscilar. Es el diseño ideal para la suspensión de vehículos.

6. Conclusión

La implementación del método de Runge-Kutta de cuarto orden permite una simulación estable y precisa del sistema masa-resorte-amortiguador. A diferencia de métodos más simples como Euler, RK4 conserva mejor la energía en sistemas conservativos ($c = 0$) y predice con exactitud la envolvente de decaimiento en sistemas disipativos, lo que lo convierte en una herramienta robusta para la enseñanza y análisis de vibraciones mecánicas.