Métodos Numéricos para la Ciencia e Ingeniería
Informe Tarea 2

Benjamín Guerrero

1 de Octubre, 2015

1. Introducción

Sea una partícula de masa *m* que solo se mueve verticalmente en el eje *Y*, rebotando contra un suelo que oscila de forma sinusoidal con amplitud *A* y frecuencia ω. El choque es inelástico, y sigue la siguiente regla:

$$v_p'(t^*) = (1+\eta)v_s(t^*) - \eta v_p(t^*)$$

En la fórmula, t^* es el instante del bote, v_p y v_p' son las velocidades justo antes y justo después del bote, v_s es la velocidad del suelo en ese instante y η es un coeficiente de restitución (η entre 0, y 1; η=1 corresponde al caso elástico).

Inicialmente (t=0), la partícula está en contacto con el suelo, y su velocidad hacia arriba es mayor a la velocidad del suelo (se despega de éste).

Los parámetros del problema son (*A*, ω, η, *m*, *g*) y la condición inicial (*y*(0), *v*(0)). Se adimensializa el problema con *m*=1, *g*=1, y *A*=1. *y*(0) ya se escogió (pegado al suelo), así que solo queda por elegir *v*(0), η y ω.

Primero, se pide escribir una rutina que permita calcular (y_{n+1}, v'_{n+1}) dados (y_n, v'_n), que son la posición y velocidad luego del n-ésimo choque.

Luego, usando η=0.15 y para ω = 1.66, se pide estimar N_{relax}, el número de botes necesarios para relajar el sistema.

Después, se pide probar con un par de otros valores para ω entre 1.66 y 1.7, y comparar el resultado de N_{relax} .

Finalmente, se pide hacer un gráfico de v'_n versus ω con ω entre 1.66 y 1.79 y n =2×N_{relax},..., 2×N_{relax} + 49, es decir, plotear 50 valores de v'_n por cada valor de ω.

2. Procedimiento

Para empezar, se crea un programa llamado rebote*_pelota.py*, que realizará lo pedido. Por conveniencia, se programan las siguientes funciones:

Ppiso_oscilador, que determina la posición del piso. Como es una función sinusoidal, esta es así:$y(t) = Asen(\omega t)$. Derivando esta función se obtiene *Vpiso_oscilador*, que determina la velocidad del piso de la siguiente manera: $V(t) = A\omega\cos(\omega t)$.

Luego tenemos *arco_pelota* y *velocidad_pelota*, que determinan la posición y la velocidad de la pelota, respectivamente. Como es un movimiento uniforme acelerado, las ecuaciones respectivas son $y(t) = y_o + V_o - \frac{1}{2}gt^2$, y $V(t) = V_o - gt$.

Finalmente, tenemos *rebote_pelota*, que calcula la velocidad de la pelota después del rebote. Esta función es la descrita en la introducción:

$$v_p'(t^*) = (1+\eta)v_s(t^*) - \eta v_p(t^*)$$

Una vez hecho esto, se realizan una serie de inputs para que el usuario entre los valores de $v_n, y_n, \eta,$ y ω. También se especifica si el piso va subiendo o bajando. Se añadieron while loops que aseguran que los valores de $y_n,$ y η no estén fuera de sus límites.

Luego, haciendo el reverso de la función *Ppiso_oscilador*, se saca t_n, el tiempo inicial asociado (este tiempo dependerá si el piso sube o baja).

Después, se usa el método de Brent para sacar la raíz de *Ppiso_oscilador- arco_pelota*, en función de t. Esto da t^*, el tiempo del siguiente rebote. Luego, con ese tiempo, se sacan $Y_{s(n+1)}$ y $V_{s(n+1)}$, la posición y velocidad del piso en el momento t^*+t_n. También se puede sacar la velocidad de la pelota justo antes del choque, y, usando *rebote_pelota*, se saca la velocidad de la pelota después del rebote. Así, se tiene lo pedido.

3. Resultados

4. Conclusión