



Universidad de Sevilla
Escuela Politécnica Superior de Sevilla



Trabajo de Fin de Estudios en Ingeniería en Diseño
Industrial y Desarrollo del Producto, e Ingeniería
Mecánica

Sobre la fricción no lineal

Realizado por: Joaquín Gabriel Salas-Boza

Supervisado por: Dr. Niurka R. Quintero

Junio 2019

Título: Sobre la fricción no lineal

Autor: Joaquín Gabriel Salas-Boza

Tutora académica: Dra. Niurka Rodríguez Quintero

Resumen: En este trabajo se analiza en detalle el efecto de la fricción de Stokes (lineal) y de la fricción seca (no lineal) en el oscilador armónico simple. Todos los sistemas estudiados se resuelven tanto analíticamente, como numéricamente en los capítulos 2 y 3. Se estudian las soluciones mediante el diagrama de fase y su relación con la ley de conservación de la energía. En el capítulo 4 se muestra cómo puede utilizarse el SolidWorks en el diseño de las piezas, en el estudio de la evolución de los sistemas disipativos estudiados, y en el desarrollo de animaciones. Finalmente, se presentan las conclusiones del trabajo, sus posibles extensiones y los programas numéricos realizados en *Matlab* y en *Mathematica*.

Palabras claves: fricción no lineal, retrato de fase, modelado con SolidWorks.

Title: About the nonlinear friction

Author: Joaquín Gabriel Salas-Boza

Academic supervisor: Dra. Niurka Rodríguez Quintero

Abstract: The effects of Stokes friction (linear) and dry friction (non-linear) on the simple harmonic oscillator are analyzed in detail in this bachelor thesis. All the studied systems are solved analytically and numerically in chapters 2 and 3. The solutions are studied by means of the phase diagram and its relation to the law of energy conservation. Chapter 4 shows how SolidWorks can be used in part design for building a simulation model, in studying the evolution of the studied dissipative systems, and in developing animations. Finally, the conclusions of the work, its possible extensions, and the numerical programs made in *Matlab* and *Mathematica* are presented.

Keywords: nonlinear friction, portrait phase, modeled with SolidWorks.

Índice general

1. Introducción	3
2. Movimiento armónico simple (MAS).	7
2.1. Sistema masa-resorte.	7
2.2. Resolución de la ecuación del movimiento armónico simple.	10
2.3. Vibraciones amortiguadas	11
2.4. Oscilaciones forzadas	14
2.4.1. Casos particulares	16
2.5. Planos de fase. Estabilidad de los puntos de equilibrio.	18
2.6. Trayectorias y leyes de conservación	23
2.6.1. Oscilador amortiguado	24
2.6.2. Oscilador amortiguado y forzado. Condiciones iniciales excepcionales.	25
2.6.3. Oscilador amortiguado y forzado con condiciones iniciales arbitrarias.	26
3. Fricción de Coulomb	29
3.1. Fricción estática y dinámica entre el sistema masa-resorte y una superficie horizontal.	29
3.2. Cálculo del primer tramo de desplazamiento.	31
3.3. Cálculo del segundo tramo.	33
3.4. Cálculo de las ecuaciones generales.	34
4. Modelado y animación en SolidWorks	39
4.1. Modelado de los elementos necesarios para la simulación	39
4.2. Creación del ensamblaje, animación y simulación	46
5. Apéndice: Programas numéricos en Matlab.	55
5.1. Apéndice	55

5.1.1. Acceso a MATLAB y salida del mismo. Instrucciones de utilidad.	55
5.1.2. Operaciones básicas. Funciones elementales.	56
5.1.3. Vectores y matrices	59
5.1.4. Funciones sobre los vectores	61
5.1.5. Editor de Matlab. Programación.	66
5.1.6. Gráficos	69
5.1.7. Grabar y leer datos en ficheros. Impresión de las gráficas .	71
5.2. MAS	73
5.3. Vibraciones amortiguadas subcríticas	77
5.4. Vibraciones amortiguadas subcríticas y forzadas	81
5.5. Fricción estática y dinámica	85
6. Apéndice: Programas en Mathematica.	91
7. Apéndice: Estudio de la estabilidad	97
8. Conclusiones	101
Bibliografía	107

Capítulo 1

Introducción

La fricción es uno de los fenómenos más complejos de la ingeniería [2]. Debido a su presencia en la vida diaria, es de vital importancia comprender las leyes físicas fundamentales por las que se rige. Depende tanto de las superficies o materiales entre las que se produce el contacto entre dos cuerpos, como de las condiciones de aceleración y velocidad de los cuerpos afectados, entre otros muchos factores [10].

La fricción es uno de los principales elementos a tener en cuenta en el diseño y control de las máquinas con el objetivo de ahorrar energía. Por otro lado, también en algunos sistemas físicos la fuerza de rozamiento estática es la causa del movimiento. En los libros de texto de Física General [1] se suele estudiar la fuerza de rozamiento seco entre dos superficies suponiendo que dicha interacción

1. se opone al movimiento de un bloque sobre otro
2. es proporcional a la normal que ejerce un bloque sobre el otro
3. no depende del área de contacto
4. es independiente de la velocidad una vez que comienza el movimiento.

Cuando incluimos este tipo de rozamiento en un bloque que desliza sobre otro, la segunda ley de Newton es lineal. Se integra dos veces y obtenemos la posición de una partícula con respecto a la otra [13]. Sin embargo, en la mayoría de los textos de Física y Mecánica general, al estudiar el movimiento armónico simple de un cuerpo sobre una superficie horizontal, este tipo de fricción no se tiene en cuenta [13, 1]. Generalmente se considera la fuerza disipativa, que surge como consecuencia del movimiento del cuerpo en un fluido, como por ejemplo, el aire, el agua o algún lubricante. Dicha fuerza es proporcional a la velocidad¹. Este tipo

¹Esta fuerza se denomina fuerza de Stokes.

de fuerza también puede ser proporcional a la velocidad al cuadrado o incluso ser tan compleja, que la ecuación de movimiento sólo se puede resolver mediante simulaciones numéricas partiendo de unas condiciones iniciales dadas [12].

Al incorporar el rozamiento seco² a este tipo de problemas, la segunda ley de Newton es no lineal. Esta es la razón fundamental de no considerar esta fuerza, en la ecuación del oscilador armónico simple, en los cursos de Física general o de Mecánica. Uno de los objetivos de esta memoria es precisamente analizar el efecto de este tipo de fricción en el oscilador armónico simple. Se muestra, que efectivamente la segunda Ley de Newton es no lineal. Sin embargo, a diferencia de muchos sistemas no lineales se puede resolver exactamente.

En el capítulo 2 comenzamos introduciendo tres sistemas físicos que conducen a la ecuación del oscilador armónico simple. Las oscilaciones armónicas (soluciones exactas del oscilador armónico simple) se distinguen de otro tipo de oscilaciones, porque presentan una sola frecuencia, llamada la frecuencia propia del sistema, que depende de los parámetros del mismo, pero no de las condiciones iniciales. Matemáticamente, son descritas por las funciones armónicas, senos o cosenos.

Ésta es la definición más básica con la que estamos habituados a trabajar sobre el comportamiento armónico de las oscilaciones. Gracias a su sencillez, generalmente se emplea una masa conectada a un resorte elástico, para explicar los conceptos relativos a este tipo de movimiento. Sin embargo, existen numerosos sistemas que presentan este tipo de oscilaciones, desde la rotación y oscilación de la tierra alrededor del eje polar a la oscilación de los átomos en torno a su punto de equilibrio (En el caso de la oscilación de la tierra, conocida como oscilación o “bamboleo” de Chandler, consultar [14]).

El sistema masa-resorte, es uno de los que aparece en los capítulos 2 y 3, donde paulatinamente se establecen distintas condiciones y consideraciones. Se discute la influencia no solo de la masa y de la constante elástica del resorte, sino también de las condiciones iniciales. Este último estudio se realiza a través del retrato de fases. Se aumenta gradualmente la complejidad del sistema, incluyendo las siguientes fuerzas: (i) la de amortiguación proporcional a la velocidad, (ii) la fricción estática y dinámica entre las superficies involucradas, y (iii) fuerzas externas constantes y periódicas.

En los modelos estudiados se conserva la energía mecánica cuando no hay fricción. La introducción de una fuerza de rozamiento a través de un término disipativo, hace que la energía mecánica se disipe. El cuerpo tiende hacia una si-

²También llamado fricción de Coulomb.

tuación de equilibrio (o sea el reposo, en ausencia de fuerzas externas). Cuando se añade al oscilador armónico disipación y fuerzas externas, se muestra como la energía disipada puede ser compensada. Entonces la partícula puede oscilar entorno a su punto de equilibrio. Precisamente el retrato de fases está compuesto por curvas, donde se conserva la energía total del sistema y por los puntos de equilibrio. En muchos de los casos estudiados, derivamos analíticamente dichas curvas.

Es por ello, que el objetivo de los siguientes capítulos no sólo es ofrecer una explicación sobre la influencia de la disipación sobre las oscilaciones armónicas, sino también la de realizar una presentación del tema de la manera más pedagógica posible.

Las ecuaciones diferenciales ordinarias que modelan nuestros sistemas físicos se estudian desde el punto de vista teórico y numérico. La metodología que guía esta memoria es la siguiente:

1. se utilizan las leyes de Newton para estudiar la dinámica de los sistemas físicos y se obtienen ecuaciones diferenciales ordinarias, algunas lineales, otras no.
2. se emplea el método estándar que utiliza el polinomio característico, para resolver las ecuaciones diferenciales lineales con coeficientes constantes.
3. se resuelven las ecuaciones no lineales en intervalos de tiempo, teniendo en cuenta que tanto la posición, como la velocidad son funciones continuas.
4. los puntos 2 y 3 también se estudian desde el punto de vista numérico, a través de códigos implementados en *Matlab*.
5. se realizan animaciones del sistema masa-muelle a partir del diseño de las piezas con el SolidWorks. Esto permite representar nuestros resultados a través de simulaciones y animaciones en las tres dimensiones.

Todo ello nos permitirá obtener no solo las explicaciones teóricas subyacentes a los comportamientos de las oscilaciones armónicas y de la fricción no lineal, sino también la representación y visualización más directa de los mismos.

Capítulo 2

Movimiento armónico simple (MAS).

Como indicábamos en la introducción, a continuación se presentan tres sistemas físicos que son modelados por la ecuación del movimiento armónico simple (MAS) que obtendremos a partir del estudio del sistema masa-resorte. A partir de la segunda Ley de Newton, se obtienen las ecuaciones que determinan la evolución de la posición en el tiempo. Posteriormente se presenta su resolución general que depende de las condiciones iniciales. Para más detalle, consúltese [1].

2.1. Sistema masa-resorte.

Supongamos que tenemos una superficie horizontal sin rozamiento, donde se coloca una masa m , unida por un extremo a un resorte de constante elástica k y masa despreciable. El otro extremo del muelle está fijo a una pared. Por el momento no tenemos en cuenta las fuerzas de rozamientos, ni la causada por el aire, ni la que existe entre las superficies en contacto. Situamos el eje horizontal X , haciendo coincidir el origen con la posición de la partícula cuando está en equilibrio (ver figura 2.1 (a)). Consideramos que en este momento el muelle está sin deformar. Si desplazamos el sistema hacia la derecha una distancia x , el alargamiento del resorte con respecto a su longitud natural también es x . Según la Ley de Hooke, sobre la partícula de masa m se ejerce una fuerza hacia la izquierda de módulo $k x$. Al soltar la partícula, su posición en cada instante de tiempo está determinada por la segunda Ley de Newton:

$$-k x = m \frac{d^2 x}{dt^2}, \quad (2.1)$$

que se reescribe convenientemente como

$$\frac{d^2x}{dt^2} + \omega_0^2 x = 0, \quad (2.2)$$

siendo $\omega_0^2 = k/m$ la frecuencia propia del sistema. La ecuación (2.2) es una ecuación diferencial lineal de segundo orden con coeficientes constantes, cuya resolución exacta depende de dos condiciones iniciales, por ejemplo: la posición y la velocidad iniciales de la partícula.

El segundo ejemplo que se analiza también es un sistema masa-resorte, pero ahora se encuentra suspendido verticalmente, de tal forma que el cuerpo oscila en dirección vertical (ver figura 2.1 (b)). El eje Y lo colocamos verticalmente hacia abajo, haciendo coincidir el origen con la posición del muelle sin deformar (como en el caso anterior). Estirando el resorte hacia abajo y soltándolo, su posición en cualquier instante de tiempo viene determinada por la segunda Ley de Newton:

$$mg - ky = m \frac{d^2y}{dt^2}, \quad (2.3)$$

que reescribimos como:

$$\frac{d^2y}{dt^2} + \omega_0^2 y = g. \quad (2.4)$$

La posición de equilibrio se determina a partir de la condición $\frac{d^2y}{dt^2} = 0$, por tanto $y_0 = g/\omega_0^2$. Si suponemos que

$$\tilde{y} = y - y_0, \quad (2.5)$$

entonces (2.3) se transforma en

$$\frac{d^2\tilde{y}}{dt^2} + \omega_0^2 \tilde{y} = 0, \quad (2.6)$$

que coincide con la ecuación (2.2).

Finalmente presentamos el péndulo físico, una varilla de longitud L y de masa m , uniformemente distribuida. La varilla cuelga de uno de sus extremos a través de una articulación y se mueve en el plano representado, rotando alrededor del eje vertical que pasa por la articulación (ver figura 2.2).

Supongamos que la varilla se mueve de tal forma que aumenta el valor del ángulo, el torque de la fuerza de gravedad con respecto a la articulación, será

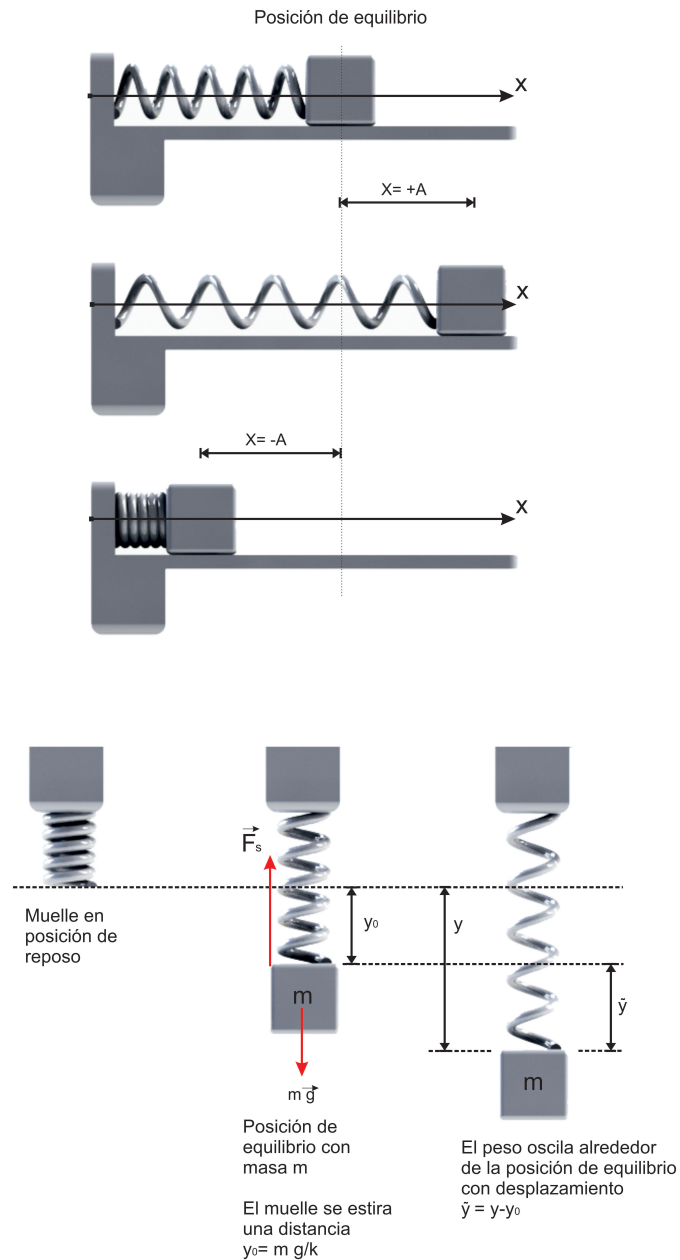


Figura 2.1: Sistema masa resorte. (a) Oscilaciones horizontales. (b) Oscilaciones verticales.

igual a

$$\begin{aligned}
 \vec{\tau}_A &= I_A \vec{\alpha}, \\
 \frac{L}{2} mg \operatorname{sen} \theta (-\vec{k}) &= \frac{1}{3} mL^2 \frac{d^2\theta}{dt^2} (\vec{k}), \\
 \frac{d^2\theta}{dt^2} + \omega_0^2 \operatorname{sen}(\theta) &= 0,
 \end{aligned} \tag{2.7}$$

donde el momento de inercia de la varilla con respecto a A es $I_A = mL^2/3$ de acuerdo con el Teorema de Steiner [1, 8], $\vec{\alpha}$ es la aceleración angular de la varilla, \vec{k} es un vector unitario dirigido hacia afuera del plano donde se encuentra la varilla y la frecuencia propia del sistema es $\omega_0^2 = 3g/(2L)$. Si la amplitud de las oscilaciones es pequeña, $\sin(\theta) \approx \theta$. Entonces, la ecuación (2.7), junto a sus dos condiciones iniciales, describe un movimiento armónico simple.

A continuación explicamos cómo se resuelven este tipo de ecuaciones.

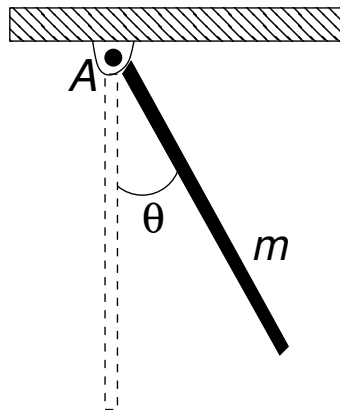


Figura 2.2: Péndulo físico

2.2. Resolución de la ecuación del movimiento armónico simple.

Reescribamos la ecuación del MAS como

$$\frac{d^2x}{dt^2} = -\omega_0^2 x, \quad (2.8)$$

$$x(0) = x_0, \quad (2.9)$$

$$x'(0) = v(0) = v_0. \quad (2.10)$$

Su solución $x(t)$ es una función, que es proporcional a su segunda derivada con respecto al tiempo. La función exponencial cumple con esa propiedad, por tanto suponemos que la solución de (2.8) viene dada por

$$x(t) = C e^{\lambda t}, \quad (2.11)$$

donde C y λ son constantes. Sustituyendo (2.9) en (2.8) obtenemos que

$$\lambda = \pm i\omega_0. \quad (2.12)$$

Quiere decir que $x(t) = Ce^{i\omega_0 t} + De^{-i\omega_0 t}$. Utilizando la identidad de Euler $e^{i\theta} = \cos \theta + i \operatorname{sen} \theta$, de forma alternativa, la solución de (2.8) es igual a

$$x(t) = A \operatorname{sen}(\omega_0 t - \phi), \quad (2.13)$$

donde la amplitud de las oscilaciones A , y la fase ϕ se determinan a partir de las condiciones iniciales (2.9)-(2.10). Si $v_0 = 0$,

$$\phi = -\frac{\pi}{2}, \quad A = -\frac{x_0}{\operatorname{sen} \phi}, \quad (2.14)$$

Si $v_0 \neq 0$

$$\phi = -\operatorname{atan} \left(\frac{\omega_0 x_0}{v_0} \right), \quad A = \frac{v_0}{\omega_0 \cos(\phi)}. \quad (2.15)$$

La función (2.13) describe oscilaciones de período $T = 2\pi/\omega_0$. Como podemos observar, el período T solo depende de las características del sistema, de manera que una vez fijado el sistema T será constante e independiente de las condiciones iniciales. Por ejemplo si $x(0) = 4$ y $v(0) = 0$, $x(t) = 4 \cos(\omega_0 t)$. Como la resolución de la ecuación (2.8) es exacta, en la sección 5 la utilizamos para verificar los programas numéricos que se han implementado en Octave/Matlab. Ejecutando el programa mainplot.m (véase sección 5), observamos la evolución tanto numérica como exacta de la posición y de la velocidad $x(t)$ y $v(t)$. Ambas se representan en la figura 2.3.

Otro conjunto de condiciones iniciales sólo afecta la amplitud de las oscilaciones y la fase, pero no al período de las mismas. Si deseamos oscilaciones más rápidas (períodos más cortos o frecuencias mayores) tenemos que aumentar el cociente k/m , ya sea aumentando k mientras se mantiene constante m o disminuyendo m sin variar k .

2.3. Vibraciones amortiguadas

Supongamos que la fuerza de rozamiento entre las partículas y el medio (aire o agua) es proporcional a la velocidad, o sea $F_r = -c dx/dt$, donde $c > 0$ es el coeficiente de fricción. En este caso, la ecuación de movimiento (2.8) queda

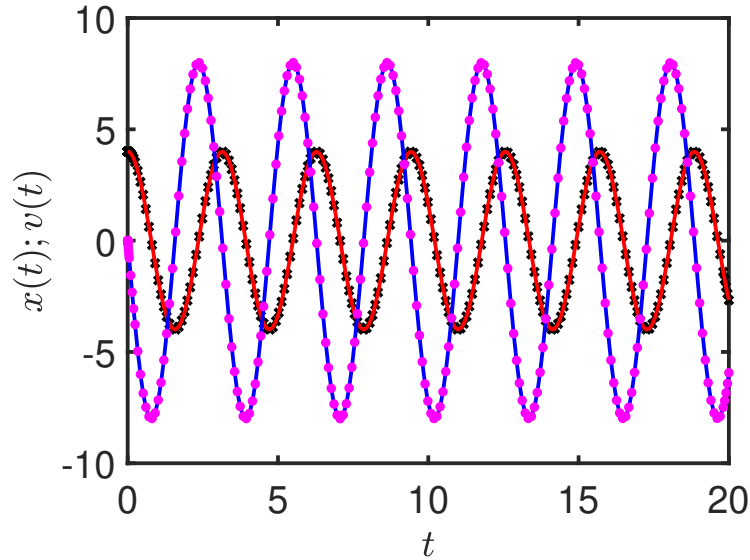


Figura 2.3: Oscilador armónico simple. La solución numérica de la ecuación (2.8), $x(t)$ (rojo) y $v(t)$ (azul) coinciden con la exacta (cruces y círculos, respectivamente). Condiciones iniciales: $x(0) = 4$ y $v(0) = 0$. Parámetros: $\omega_0 = 2$. Todas las magnitudes se considerarán adimensionales.

modificada

$$\frac{d^2x}{dt^2} = -\omega_0^2 x - 2\beta \frac{dx}{dt}, \quad (2.16)$$

donde $\beta = c/(2m)$ mientras que las condiciones iniciales siguen siendo

$$x(0) = x_0, \quad (2.17)$$

$$x'(0) = v_0. \quad (2.18)$$

Sustituyendo (2.9) en (2.16) se obtiene que λ satisface la ecuación $\lambda^2 + \omega_0^2 + 2\beta\lambda = 0$. Por tanto,

$$\lambda = -\beta \pm i\omega_d, \quad (2.19)$$

$$\omega_d = \sqrt{\omega_0^2 - \beta^2}. \quad (2.20)$$

Dependiendo de la relación entre β y ω_0 , existen tres tipos de amortiguamiento (ver [1]), que se enumeran a continuación:

1. Amortiguamiento subcrítico: si $\beta < \omega_0$. En este caso λ es compleja y su parte real es negativa. Teniendo en cuenta las condiciones iniciales, la solución

de (2.16) viene determinada por

$$x(t) = e^{-\beta t} A \operatorname{sen}(\omega_d t - \phi), \quad (2.21)$$

donde, si $v_0 = 0$, $\beta \neq 0$

$$\phi = -\operatorname{atan}\left(\frac{\omega_d}{d}\right), \quad A = -\frac{x_0}{\operatorname{sen}(\phi)}. \quad (2.22)$$

Además, si $v_0 = \beta = 0$, entonces $\phi = -\pi/2$ y $A = -x_0/\operatorname{sen}(\phi)$ y se recuperarían las condiciones indicadas en (2.14). Finalmente, si $v_0 \neq 0$

$$\phi = -\operatorname{atan}\left(\frac{\omega_d x_0}{v_0 + \beta x_0}\right), \quad A = \frac{v_0}{\omega_d \cos(\phi) + \beta \operatorname{sen}(\phi)}. \quad (2.23)$$

En este caso la partícula oscila con una amplitud que decae exponencialmente en el tiempo, de manera que la posición de la partícula queda acotada en la región comprendida entre las funciones $\pm A e^{-\beta t}$ representadas en la figura (2.4). Además, si $\beta = 0$ en (2.23), obtenemos las expresiones (2.15). Esto nos permite recuperar el caso no disipativo.

2. Amortiguamiento crítico: $\beta = \omega_0$. $\lambda = -\beta < 0$. La partícula no oscila, su amplitud se amortigua en el tiempo. Exactamente, su posición es igual a

$$x(t) = e^{-\beta t} (A + B t). \quad (2.24)$$

3. Amortiguamiento supercrítico: $\beta > \omega_0$. Los dos valores de $\lambda_{1,2} = -\beta \pm \sqrt{\beta^2 - \omega_0^2}$ son reales y negativos. La amplitud inicial simplemente se atenúa,

$$x(t) = A e^{\lambda_1 t} + B e^{\lambda_2 t}. \quad (2.25)$$

En los dos últimos casos las constantes A y B también están determinadas por las condiciones iniciales.

Al ejecutar el programa `mainplotv2.m` (véase sección 5.3), comparamos la solución numérica del sistema (2.16) con su solución exacta (ver la figura 2.4, donde los parámetros escogidos cumplen con la condición del amortiguamiento subcrítico).

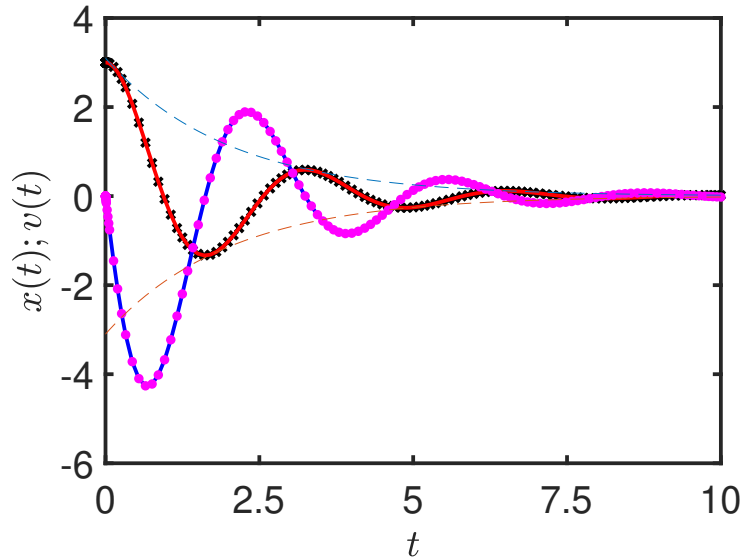


Figura 2.4: Oscilador armónico simple con amortiguamiento subcrítico. La solución numérica de $x(t)$ (rojo) y $v(t)$ (azul) coinciden con la exacta (cruces y círculos, respectivamente). Las líneas discontinuas representan las funciones $\pm Ae^{-\beta t} = \pm 3e^{-0,5t}$ (ver explicación en el texto). Condiciones iniciales: $x(0) = 3$, $v(0) = 0$. Parámetros: $\omega_0 = 2$ y $\beta = 0,5$.

2.4. Oscilaciones forzadas

Supongamos que sobre el sistema modelado por (2.16) actúa una fuerza externa que depende del tiempo, $F(t)$. Entonces la segunda ley de Newton para la partícula está dada por

$$\frac{d^2x}{dt^2} + \omega_0^2 x + 2\beta \frac{dx}{dt} = \frac{F(t)}{m}, \quad (2.26)$$

y se complementa con sus condiciones iniciales

$$x(0) = x_0, \quad (2.27)$$

$$x'(0) = v_0, \quad (2.28)$$

y $\beta = c/(2m)$. La ecuación (2.26) sigue siendo lineal, pero como el término de la derecha no es igual a cero es no homogénea. Como este término es independiente de x , y de sus derivadas, su solución es igual a la suma de la solución de la ecuación homogénea y de una solución particular.

$$x(t) = x_h(t) + x_p(t). \quad (2.29)$$

La solución homogénea es la representada por la expresión (2.21) y la solución particular depende de la función $F(t)$. Por ejemplo, supongamos que $F(t)$ es la función periódica,

$$F(t) = a_0 m \operatorname{sen} \omega t + b_0 m \operatorname{cos} \omega t, \quad (2.30)$$

entonces la solución particular se busca de la forma

$$x_p(t) = A_p \operatorname{sen} \omega t + B_p \operatorname{cos} \omega t, \quad (2.31)$$

donde A_p y B_p se determinan sustituyendo (2.31) en (2.26),

$$A_p = \frac{(\omega_0^2 - \omega^2) a_0 + 2 \beta \omega b_0}{(\omega_0^2 - \omega^2)^2 + 4 \beta^2 \omega^2}, \quad (2.32)$$

$$B_p = \frac{(\omega_0^2 - \omega^2) b_0 - 2 \beta \omega a_0}{(\omega_0^2 - \omega^2)^2 + 4 \beta^2 \omega^2}. \quad (2.33)$$

Teniendo en cuenta las condiciones iniciales, la solución de la ecuación (2.26) sería igual a

$$x(t) = e^{-\beta t} A \operatorname{sen}(\omega_d t - \phi) + A_p \operatorname{sen} \omega t + B_p \operatorname{cos} \omega t, \quad (2.34)$$

donde $\omega_d = \sqrt{\omega_0^2 - \beta^2}$, y A_p y B_p están determinadas por (2.32)-(2.33). Además, la fase ϕ y la amplitud A son iguales a

$$\phi = -\operatorname{atan} \left(\frac{\omega_d}{\beta} \right), \quad A = -\frac{x_0 - B_p}{\operatorname{sen} \phi}, \quad (2.35)$$

si $v_0 - \omega A_p = 0$ y $\beta \neq 0$. Si por el contrario $\beta = 0$, entonces $\phi = \pi/2$ y $A = -(x_0 - B_p)/\operatorname{sen} \phi$.

En la ecuación (2.35) se observa que $x_0 - B_p = 0$ implica que $A = 0$. Como consecuencia $v(0) = \omega A_p$. Bajo estas condiciones la solución de la ecuación homogénea es cero y que la solución $x(t) = x_p(t)$ tiene el mismo periodo que la fuerza externa. El sistema "olvida" la frecuencia ω_d y las condiciones iniciales desde el primer momento.

Si $v_0 - \omega A_p \neq 0$

$$\phi = -\operatorname{atan} \left(\frac{\omega_d (x_0 - B_p)}{v_0 - \omega A_p + \beta (x_0 - B_p)} \right), \quad A = \frac{v_0 - \omega A_p}{\omega_d \operatorname{cos} \phi + \beta \operatorname{sen} \phi}. \quad (2.36)$$

Si no hay fuerza externa, o sea si $a_0 = b_0 = 0$, entonces $A_p = B_p = 0$ y $x(t) = x_h(t)$, y las ecuaciones (2.35) y (2.36) se reducen a las relaciones (2.22) y (2.23).

Al ejecutar el programa mainplotv3.m (véase sección 5.3), comparamos la

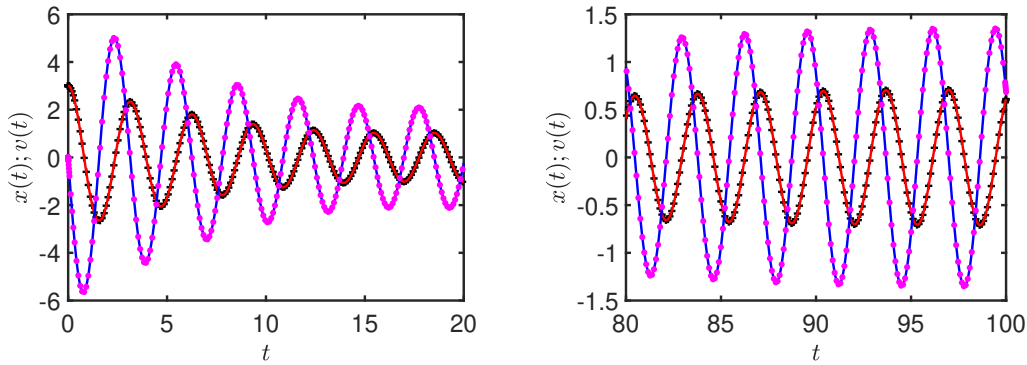


Figura 2.5: Oscilador armónico simple, amortiguado y forzado a tiempos cortos (panel izquierdo) y a tiempos largos (panel derecho). La solución numérica de $x(t)$ (rojo) y $v(t)$ (azul) coinciden con la exacta (signo más y asterisco, respectivamente). Condiciones iniciales: $x(0) = 3$, $v(0) = 0$. Parámetros: $\omega_0 = 2$, $\beta = 0,05$, $a_0 = 0,3$, $b_0 = 0$, y $\omega = 1,9$.

solución numérica del sistema (2.26) con la solución exacta encontrada en este apartado (ver las figuras 2.5 y 2.6). Comparando las posiciones y las velocidades, representadas a tiempos cortos y largos en la Fig. 2.5 se observa que en el mismo intervalo de tiempo ($\Delta t = 20$) la partícula realiza menos oscilaciones a tiempos largos que las que realiza a tiempos cortos. Esto se debe a que la frecuencia que prevalece a tiempos largos es $\omega = 1,9 < \omega_d = 1,99$, por tanto el período de las oscilaciones a tiempos largos es mayor que el observado a tiempos cortos. En la figura 2.6, se han escogido las condiciones iniciales de tal forma que se anula el régimen transitorio. Como hemos mantenido los mismos parámetros de la Fig. 2.5, se puede apreciar que los resultados obtenidos en la Fig. 2.6 coinciden con los de la Fig. 2.5 a tiempos largos. Sobre este detalle volveremos a hacer hincapié cuando analicemos las órbitas correspondientes a estos dos conjuntos de condiciones iniciales en el plano de fase.

2.4.1. Casos particulares

Cuando la frecuencia de la fuerza externa, ω tiende a la frecuencia propia del sistema, A_p y B_p alcanzan su valor máximo. En este caso, estas amplitudes tienden a

$$\lim_{\omega \rightarrow \omega_0} A_p = \lim_{\omega \rightarrow \omega_0} \frac{(\omega_0^2 - \omega^2) a_0 + 2 \beta \omega b_0}{(\omega_0^2 - \omega^2)^2 + 4 \beta^2 \omega^2} = \frac{b_0}{2 \beta \omega_0} \quad (2.37)$$

$$\lim_{\omega \rightarrow \omega_0} B_p = \lim_{\omega \rightarrow \omega_0} \frac{(\omega_0^2 - \omega^2) b_0 - 2 \beta \omega a_0}{(\omega_0^2 - \omega^2)^2 + 4 \beta^2 \omega^2} = -\frac{a_0}{2 \beta \omega_0}. \quad (2.38)$$

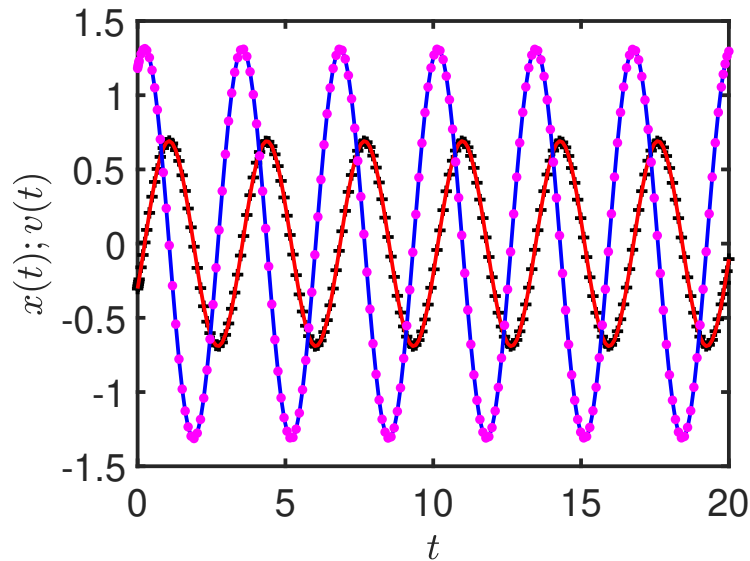


Figura 2.6: Oscilador armónico amortiguado y forzado, para el caso particular. Las condiciones iniciales hacen que el sistema oscile con la frecuencia de la fuerza que actúa sobre el mismo desde $t = 0$. Condiciones iniciales: $x(0) = B_p$, $v(0) = \omega A_p = 1,18$. Parámetros: $\omega_0 = 2$, $\beta = 0,05$, $a_0 = 0,3$, $b_0 = 0$, y $\omega = 1,9$.

Como consecuencia la amplitud de las oscilaciones es máxima. De hecho, se puede comprobar que tiende a infinito si β tiende a cero. Este fenómeno se conoce como resonancia, y a la frecuencia $\omega = \omega_0$ se le llama la frecuencia de resonancia.

Otro caso particular, es el de la fuerza externa constante. Si consideramos $a_0 = 0$ y $\omega = 0$ en la ecuación (2.30), $F(t) = m b_0$ es constante. La solución particular (2.31) es igual a $x_p(t) = b_0/\omega_0^2$ y la solución general del oscilador armónico amortiguado bajo una fuerza constante es

$$x(t) = e^{-\beta t} A \operatorname{sen}(\omega_d t - \phi) + \frac{b_0}{\omega_0^2}, \quad (2.39)$$

donde $\omega_d = \sqrt{\omega_0^2 - \beta^2}$, y las constantes A y ϕ vienen determinadas por las condiciones iniciales. A diferencia del forzado periódico, aquí la partícula tiende al punto b_0/ω_0^2 y su velocidad tiende a cero.

2.5. Planos de fase. Estabilidad de los puntos de equilibrio.

En la sección anterior se representa la evolución temporal tanto de la posición como de la velocidad, a partir de unas condiciones iniciales determinadas. Esto nos permite conocer en cada instante de tiempo, donde está la partícula y que velocidad tiene.

Sin embargo, también es de interés conocer el concepto de plano de fase. Se utiliza para caracterizar el sistema dinámico desde el punto de vista cualitativo. Nos permite conocer de antemano el conjunto de condiciones iniciales para las cuales tendremos un tipo de trayectorias, en lugar de otras. Para comenzar, podemos definir un plano o retrato de fase como una representación geométrica de las trayectorias representativas de un sistema dinámico. Cada trayectoria no es más que la representación paramétrica de la velocidad de la partícula en función de su posición, donde el parámetro es el tiempo. Cada curva se obtiene a partir de una energía determinada (que depende de las condiciones iniciales). El retrato de fase se complementa con los puntos de equilibrio del sistema. También nos permite estudiar la estabilidad de los puntos de equilibrio.

Así, recordemos que la ecuación (2.2) es una ecuación diferencial lineal de segundo orden, dependiente de las condiciones de posición y velocidad iniciales. Como tal, multiplicando toda la ecuación por la derivada de la posición en función del tiempo y luego integrando, obtenemos:

$$\frac{dx}{dt} \frac{d^2x}{dt^2} + \omega_0^2 \frac{dx}{dt} x = 0, \quad (2.40)$$

$$\frac{1}{2} \left(\frac{dx}{dt} \right)^2 + \frac{\omega_0^2}{2} x^2 = C, \quad (2.41)$$

donde $C \geq 0$ es una constante. Si ahora multiplicamos la ecuación por la masa m de la partícula y consideramos:

$$mC = E, \quad (2.42)$$

$$m\omega_0^2 = k, \quad (2.43)$$

$$\frac{m}{2E} \left(\frac{dx}{dt} \right)^2 + \frac{k}{2E} x^2 = 1, \quad (2.44)$$

donde k es la constante elástica, y E es la energía mecánica del sistema que depende de las condiciones iniciales.

El punto fijo o punto de equilibrio tiene coordenadas $x = 0$ y $\frac{dx}{dt} = 0$. Al punto fijo le corresponde la menor energía y podemos observar que la ecuación (2.44) se corresponde con la ecuación de una elipse, al estar formada por la suma de los productos de dos constantes por la velocidad al cuadrado y posición de la partícula al cuadrado, respectivamente. Por ello, si representamos la velocidad frente a la posición, obtendremos la elipse de la figura 2.7.

De aquí podemos extraer a su vez algunas conclusiones:

1. Todos los puntos (x, v) de esta elipse representan un conjunto de condiciones iniciales que tienen la misma energía mecánica.
2. Otras elipses centradas en $(0, 0)$ se obtienen partiendo de otras condiciones iniciales.
3. Los vectores representados en el plano de fase son tangentes a las trayectorias, que nunca se cruzan. Ayudan a determinar la evolución cualitativa del sistema para unas condiciones iniciales concretas de manera gráfica. Aunque hemos utilizado el programa *Matlab* para dibujar los planos de fases, resulta útil analizar la dirección de los vectores que aparecen en la Fig. 2.7. Si reescribimos la ecuación diferencial de segundo orden como un sistema de dos ecuaciones de primer orden

$$\begin{pmatrix} \frac{dx}{dt} \\ \frac{dy}{dt} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ -\omega_0^2 & 0 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = A \begin{pmatrix} x \\ y \end{pmatrix},$$

se pueden interpretar los vectores representados en el plano de fase xy (o xv) como vectores velocidades V cuyas componentes son:

$$\vec{V} = \frac{dx}{dt}\vec{i} + \frac{dy}{dt}\vec{j} = y\vec{i} - \omega_0^2 x\vec{j}. \quad (2.45)$$

Así, nos es fácil comprender las direcciones de los vectores en la Fig. 2.7. Por ejemplo, los vectores donde $x = 0$, sólo tienen componente en el eje x . Esta componente es $V_x = y$, por tanto será positiva si $y > 0$ (los vectores están dirigidos hacia la derecha) y negativa si $y < 0$ (vectores hacia la izquierda). Este análisis puede extenderse a los demás planos de fase.

4. En la sección 2.2, los dos valores de λ obtenidos en la resolución de la ecuación (2.8) eran imaginarios. Estos valores son los autovalores correspondientes a la matriz A en la ecuación lineal (2.45). El punto fijo se denomina centro y se dice que es estable (ver Fig. 2.7).

5. Las trayectorias son órbitas cerradas, porque el movimiento es periódico. Podemos observar a continuación, de hecho, que en el caso del oscilador armónico amortiguado las trayectorias tienden al punto fijo. La elipse pasará a ser, por ejemplo, una espiral (ver figura 2.8). En este caso, como se ha demostrado en la sección 2.3 al resolver (2.16), $\lambda = -\beta \pm i\omega_d$ (ver la ecuación (2.19)). Para los parámetros representados en la Fig. 2.8, estos autovalores son complejos conjugados, con parte real negativa, por tanto las trayectorias son espirales que convergen hacia el punto fijo, que es un foco estable. Este es el caso de amortiguamiento subcrítico, discutido en la sección 2.3. Si el amortiguamiento es supercrítico, los dos autovalores son negativos y el punto fijo es un nodo estable, y si el amortiguamiento es crítico, para $\beta = \omega_0$, $\lambda = -\beta < 0$, el nodo es degenerado y el plano de fase resultante es el de la figura (2.9).

Finalmente, tenemos los casos de las oscilaciones forzadas. Si la fuerza es constante, el sistema (2.26) con $F(t) = m b_0$ puede reescribirse como

$$\frac{d^2\tilde{x}}{dt^2} + \omega_0^2\tilde{x} + 2\beta\frac{d\tilde{x}}{dt} = 0, \quad (2.46)$$

donde $\tilde{x} = x - b_0/\omega_0^2$ [5]. Esta es la ecuación del oscilador armónico amortiguado. Su punto fijo tiene coordenadas $\tilde{x} = 0$ y $d\tilde{x}/dt = 0$. Por tanto, el punto fijo en las variables x y dx/dt es $(b_0/\omega_0^2, 0)$ y las trayectorias en el plano de fase serán similares al caso del oscilador armónico amortiguado.

Si la fuerza es periódica, entonces el sistema dinámico no es autónomo, y al plano de fase hay que agregarle una dimensión extra (el tiempo). La solución exacta nos proporcionaba dos casos representativos. Primero, para el caso en el que al forzar el movimiento anulábamos el régimen transitorio, esto es, $x_0 - B_p = 0$ y $v_0 - \omega_0 A_p = 0$, obtendremos una elipse (ver figura 2.10), mientras que en cualquier otro caso la trayectoria tiende a la elipse anterior (ver figura 2.11). El primer caso es curioso porque la órbita es cerrada a pesar de que la energía mecánica no se conserva. En la siguiente sección veremos en detalle la relación que existe entre la ley de conservación de la energía y las curvas en el plano de fase.

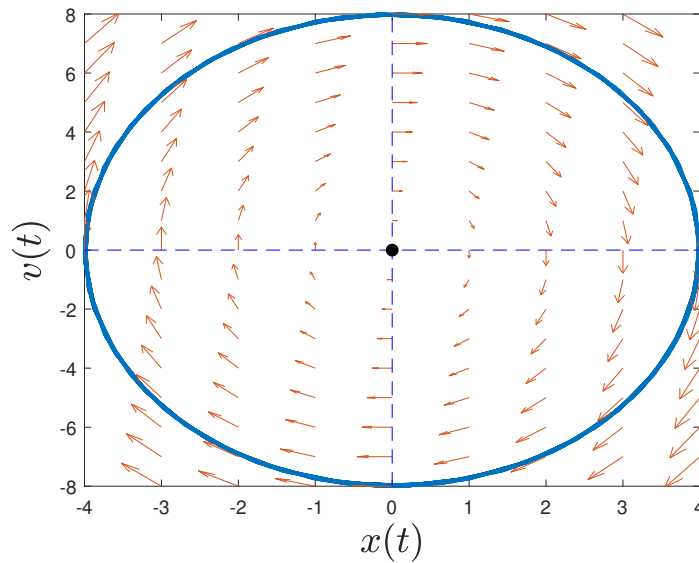


Figura 2.7: Plano de fase del oscilador armónico simple con una sola órbita. Como se demuestra en la ecuación (2.44) la representación de la velocidad frente a la posición para unas condiciones iniciales determinadas genera una elipse. Condiciones iniciales: $x(0) = 4$ y $v(0) = 0$. Parámetro: $\omega_0 = 2$.

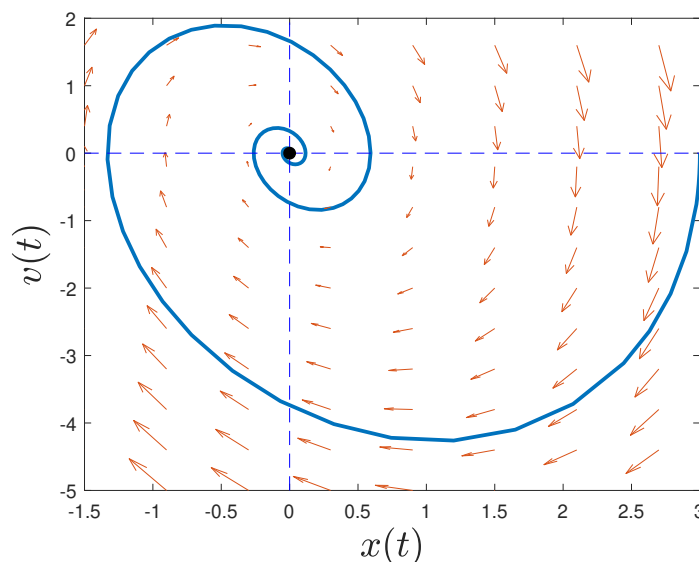


Figura 2.8: Plano de fase del oscilador armónico simple y amortiguado, donde se representa una trayectoria. Como observábamos en la gráfica del comportamiento del oscilador armónico simple y amortiguado la trayectoria representada tiende al punto de equilibrio o punto fijo. Condiciones iniciales: $x(0) = 3$, $v(0) = 0$. Parámetros: $\omega_0 = 2$ y $\beta = 0,5$.

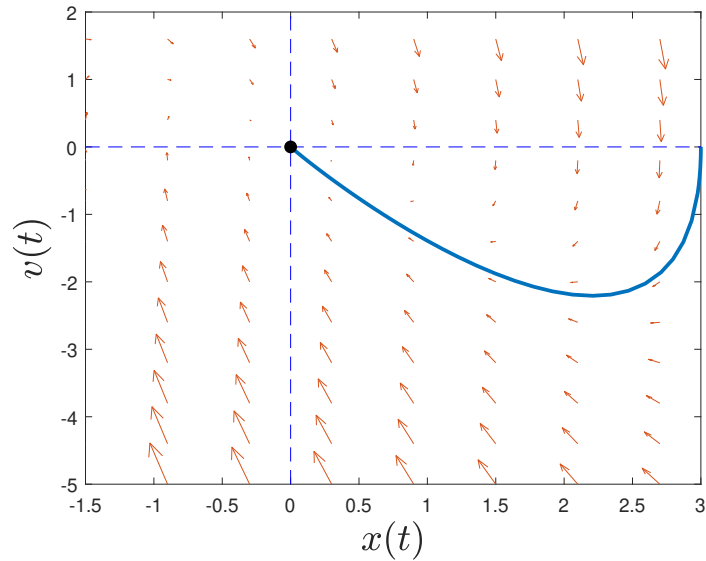


Figura 2.9: Plano de fase del oscilador armónico simple amortiguado crítico. Condiciones iniciales: $x(0) = 3$, $v(0) = 0$. Parámetros: $\omega_0 = 2 = \beta$.

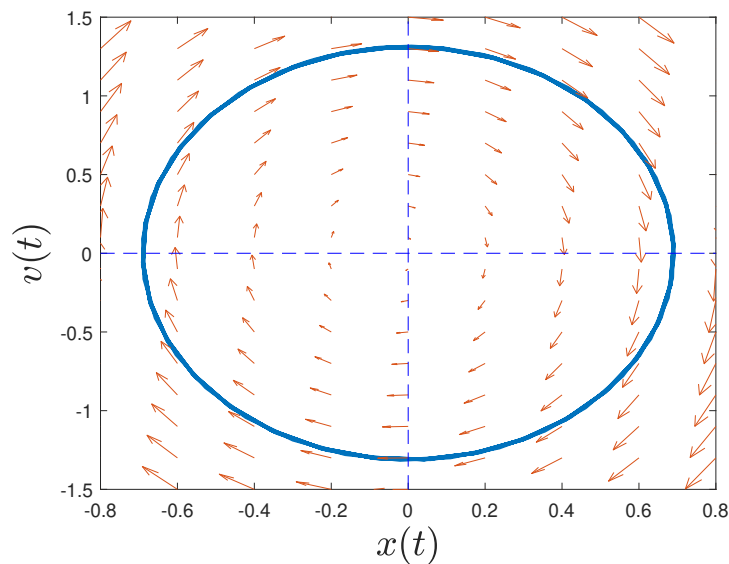


Figura 2.10: Plano de fase del oscilador armónico forzado, para $x_0 - B_p = 0$. Como vimos en el apartado de las oscilaciones forzadas, la condición inicial de $x_0 - B_p = 0$ y $v(0) - \omega A_p = 0$ hace que el sistema oscile desde el inicio con la frecuencia de la fuerza que actúa sobre el mismo, de manera que obtenemos de nuevo una elipse como en el caso del oscilador armónico simple. Condiciones iniciales: $x(0) = B_p = -0,302869287991498$, $v(0) = \omega A_p = 1,18$. Parámetros: $\omega_0 = 2$, $a_0 = 0,3$, $b_0 = 0$, $\omega = 1,9$ y $\beta = 0,05$.

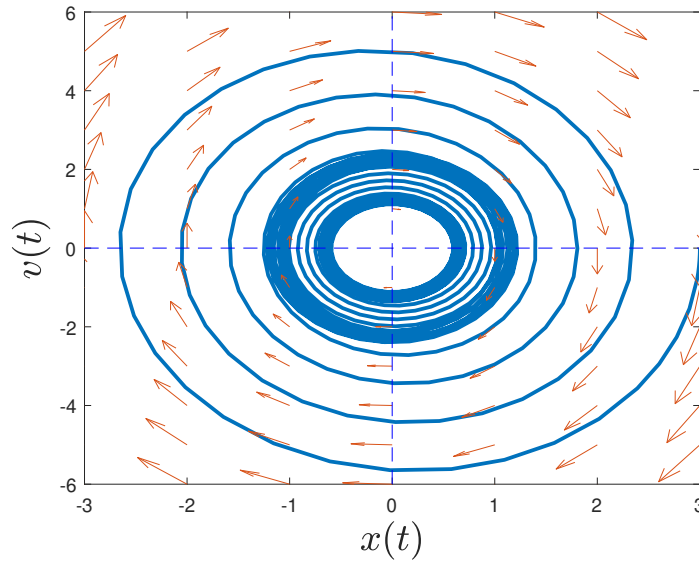


Figura 2.11: Plano de fase del oscilador armónico simple, amortiguado y forzado. Como habíamos explicado, la espiral que se forma inicialmente tiende hacia una elipse concreta, que indica que el sistema ha alcanzado la estabilidad. Condiciones iniciales: $x(0) = 3$, $v(0) = 0$. Parámetros: $\omega_0 = 2$, $\beta = 0,05$, $a_0 = 0,3$, $b_0 = 0$ y $\omega = 1,9$

2.6. Trayectorias y leyes de conservación

Todas las trayectorias del oscilador armónico son elipses. Como hemos demostrado, estas curvas cerradas se obtienen a partir de la ley de conservación de la energía mecánica (suma de la energía cinética y de la energía potencial).

En el plano de fase correspondiente al oscilador armónico amortiguado, se obtienen curvas que se dirigen desde la condición inicial hasta el punto fijo (nodo). La energía mecánica no se conserva, como se puede apreciar al sustituir la ecuación (2.26) en la derivada de la energía mecánica con respecto al tiempo

$$\frac{dE}{dt} = m \frac{dx}{dt} \frac{d^2x}{dt^2} + kx \frac{dx}{dt} = -2m\beta v^2(t) + F(t)v(t). \quad (2.47)$$

El primer término de la derecha representa la potencia disipada por el rozamiento y el segundo es la potencia asociada a la fuerza externa $F(t)$. En el oscilador armónico amortiguado y forzado periódicamente, consideramos básicamente dos tipos de condiciones iniciales: (i) las que “eliminan el régimen transitorio” y (ii) las que lo mantienen. En el primer caso las trayectorias vuelven a ser elipses, mientras que en el segundo las trayectorias tienden a esta elipse. Sin embargo, la energía mecánica no se conserva.

Estas observaciones plantean una cuestión fundamental y es ¿cuál es la relación existente entre las espirales, las elipses y las curvas que tienden a estas (ver figuras 2.8-2.11) y las leyes de conservación?

En el oscilador armónico demostramos que las trayectorias estaban relacionadas con la conservación de la energía mecánica, que a su vez es igual a la energía total del sistema. A continuación vamos a demostrar que en los demás casos que hemos estudiado, las trayectorias del plano de fase están conectadas con la ley de conservación de la energía total del sistema.

Para ello, multiplicamos la ecuación (2.26) por $(mdx/dt = mv(t))$ e integramos cada término $\int_0^t \dots d\tau$. Término a término, obtenemos

$$m \int_0^t \frac{dx}{d\tau} \frac{d^2x}{d\tau^2} d\tau = m \int_{v(0)}^{v(t)} \frac{dx}{d\tau} d\left(\frac{dx}{d\tau}\right) = \frac{m}{2} v^2(t) - \frac{m}{2} v^2(0) = \Delta E_c, \quad (2.48)$$

$$m \int_0^t \omega_0^2 x \frac{dx}{d\tau} d\tau = m \int_{x(0)}^{x(t)} \omega_0^2 x dx = \frac{k}{2} x^2(t) - \frac{k}{2} x^2(0) = \Delta E_p, \quad (2.49)$$

$$m \int_0^t \frac{dx}{d\tau} 2\beta \frac{dx}{d\tau} d\tau = 2\beta m \int_0^t \left(\frac{dx}{d\tau}\right)^2 d\tau = -W_r \quad (2.50)$$

$$\int_0^t \frac{dx}{d\tau} F(\tau) d\tau = W_{ex}. \quad (2.51)$$

La relación (2.48) representa la variación de energía cinética entre el tiempo inicial y un tiempo dado. Asimismo, (2.49) no es más que la variación de la energía potencial elástica. Los trabajos realizados por la fuerza de rozamiento y la fuerza externa están definidos en la ecuaciones (2.50) y (2.51), respectivamente. Finalmente, sumando estas ecuaciones, obtenemos que la variación de energía mecánica entre el tiempo inicial y cualquier instante de tiempo posterior está determinada por

$$\Delta E_c(v(t), v(0)) + \Delta E_p(x(t), x(0)) = W_r(t) + W_{ex}(t). \quad (2.52)$$

Esta ecuación define las trayectorias en el plano de fase $(x(t), v(t))$.

2.6.1. Oscilador amortiguado

Por ejemplo, si no hay fuerza externa, $W_{ex} = 0$ y (2.52) se reduce a

$$v^2(t) - v^2(0) + \omega_0^2 [x^2(t) - x^2(0)] = -4d \int_0^t \left(\frac{dx}{d\tau}\right)^2 d\tau. \quad (2.53)$$

Para obtener las ecuaciones correspondientes a las espirales, primero sustituimos la derivada de $x(t)$ (2.21) en el término de la derecha e integramos. Luego, sumamos a ambos miembros la función $2 dx(t)v(t)$. Finalmente teniendo en cuenta las relaciones entre las condiciones iniciales $x(0)$, $v(0)$ y la amplitud A y la fase ϕ de la solución (2.21), obtenemos

$$[v(t) + dx(t)]^2 + [x(t)\omega_d^2]^2 = A^2 \omega_d^2 e^{-2\beta t}. \quad (2.54)$$

En las nuevas variables

$$V(t) = v(t) + dx(t), \quad (2.55)$$

$$X(t) = x(t)\omega_d, \quad (2.56)$$

la ecuación (2.54) se escribe como

$$V^2(t) + X^2(t) = A^2 \omega_d^2 e^{-2\beta t}. \quad (2.57)$$

En las coordenadas polares $\rho = \sqrt{V^2(t) + X^2(t)}$ y $\phi = \omega_d t$, obtenemos la ecuación

$$\rho = A \omega_d e^{-\beta\phi/\omega_d}, \quad (2.58)$$

correspondiente a una espiral logarítmica (ver figura 2.8). Los cálculos detallados se han realizado en el programa *Mathematica* [6] `osciladormortiguado.nb` que se adjunta en el apéndice 6.

2.6.2. Oscilador amortiguado y forzado. Condiciones iniciales excepcionales.

En este caso (2.52) se reescribe como

$$v^2(t) - v^2(0) + \omega_0^2[x^2(t) - x^2(0)] = -4\beta \int_0^t \left(\frac{dx}{d\tau}\right)^2 d\tau + \frac{2}{m} \int_0^t F(\tau) v(\tau) d\tau. \quad (2.59)$$

Ahora sumamos en el miembro derecho y en el izquierdo el término $\omega^2 x^2(t)$ y reagrupando los términos, obtenemos

$$\begin{aligned} v^2(t) + \omega^2 x^2(t) &= v^2(0) - \omega_0^2 x^2(0) + (\omega^2 - \omega_0^2)x^2(t) \\ &- 4\beta \int_0^t \left(\frac{dx}{d\tau}\right)^2 d\tau + \frac{2}{m} \int_0^t F(\tau) v(\tau) d\tau. \end{aligned} \quad (2.60)$$

Sustituyendo la solución $x(t)$ (2.31) y su derivada en la parte derecha de la ecuación (2.60) y teniendo en cuenta que $A = 0$, obtenemos la ecuación de la siguiente elipse

$$v^2(t) + \omega^2 x^2(t) = \omega^2 (A_p^2 + B_p^2), \quad (2.61)$$

donde A_p y B_p las constantes definidas en (2.32)-(2.33). Esta elipse es la que está representada en la figura 2.10. Los cálculos detallados se han realizado en el programa de *Mathematica* *osciladorforzado1.nb* que adjuntamos en el apéndice 6.

2.6.3. Oscilador amortiguado y forzado con condiciones iniciales arbitrarias.

Análogamente al caso anterior, obtenemos la ecuación (2.60), donde $x(t)$ viene determinada por la ecuación (2.31), sin embargo ahora $A \neq 0$. Esto significa que la solución de la parte homogénea de la ecuación diferencial del oscilador armónico amortiguado y forzado no es cero. Reescribimos (2.31) como $x(t) = x_h(t) + x_p(t)$ y la sustituimos en la parte derecha de la ecuación (2.60). De esta forma obtenemos,

$$\begin{aligned} v^2(t) + \omega^2 x^2(t) &= [v_h(0) + v_p(0)]^2 - \omega_0^2 [x_h(0) + x_p(0)]^2 + \\ &+ (\omega^2 - \omega_0^2) [x_h(t) + x_p(t)]^2 \\ &- 4\beta \int_0^t \left(\frac{dx_h}{d\tau} + \frac{dx_p}{d\tau} \right)^2 d\tau + \frac{2}{m} \int_0^t F(\tau) \left(\frac{dx_h}{d\tau} + \frac{dx_p}{d\tau} \right) d\tau. \end{aligned} \quad (2.62)$$

De acuerdo con la ecuación (2.21), la solución de la ecuación homogénea es igual a

$$x_h(t) = e^{-\beta t} A \operatorname{sen}(\omega_d t - \phi). \quad (2.63)$$

Y según la ecuación (2.31), $x_p(t)$ está determinada por

$$x_p(t) = A_p \operatorname{sen} \omega t + B_p \operatorname{cos} \omega t. \quad (2.64)$$

Sustituyendo la función $F(t)$ (2.30) y las expresiones (2.63) y (2.64) en la parte derecha de la ecuación (2.62), e integrando obtenemos

$$v^2(t) + \omega^2 x^2(t) = (A_p^2 + B_p^2) \omega^2 + e^{-2\beta t} g(t) + e^{-\beta t} h(t), \quad (2.65)$$

donde $g(t)$ y $h(t)$ son funciones acotadas, que omitimos por simplicidad. Por tanto, después del tiempo transitorio $t \gg 1/\beta$, esta ecuación tiende a la ecuación de

la elipse obtenida con las condiciones iniciales excepcionales.

Capítulo 3

Fricción de Coulomb

Todos los sistemas estudiados en el capítulo 2 son lineales¹. Los sistemas lineales cumplen con la siguiente propiedad: si $x_1(t)$ y $x_2(t)$ son dos soluciones de la ecuación diferencial lineal, entonces cualquier combinación lineal de estas soluciones $\alpha_1 x_1(t) + \alpha_2 x_2(t)$ es también una solución de dicha ecuación. Esto se conoce como el principio de superposición lineal. Si la ecuación diferencial es no homogénea y de orden n , entonces la solución general se puede escribir como la superposición de una particular y de n soluciones independientes. Casi todas las ecuaciones lineales son resolubles, pero muy pocas ecuaciones diferenciales no lineales se pueden resolver exactamente [12]. En este capítulo se discute el efecto de la fricción de Coulomb (no lineal) sobre el sistema masa-resorte y se demuestra que es analíticamente resoluble.

3.1. Fricción estática y dinámica entre el sistema masa-resorte y una superficie horizontal.

Mientras que en el capítulo anterior considerábamos términos disipativos derivados del rozamiento entre la superficie de la partícula y el medio (agua, aire, etc.), y por tanto las fuerzas de rozamiento eran dependientes de $\frac{dx}{dt}$, en este capítulo analizaremos la llamada fricción de Coulomb, donde la fuerza disipativa es independiente de la velocidad. Así, consideraremos la partícula a estudiar como un bloque (de masa m) situado en una superficie horizontal, con fricción y sujeto a un muelle de constante elástica k [9].

¹El péndulo físico es no lineal, a pequeñas amplitudes se considera lineal.

Teniendo en cuenta la segunda ley de Newton obtenemos:

$$m \frac{d^2 x}{dt^2} + k x + \mu_k N \operatorname{sgn} \left(\frac{dx}{dt} \right) = 0, \quad (3.1)$$

donde μ_k es el coeficiente de fricción dinámico entre las superficies del bloque y la horizontal sobre la que se desplaza; N es la fuerza normal del bloque contra la superficie horizontal (en cuyo caso será equivalente a $N = mg$) y $\operatorname{sgn} \left(\frac{dx}{dt} \right)$ hace referencia a la función signo, de manera que:

$$\operatorname{sgn} \left(\frac{dx}{dt} \right) = \begin{cases} = -1, & \text{si } \left(\frac{dx}{dt} \right) < 0 \\ = 0, & \text{si } \left(\frac{dx}{dt} \right) = 0 \\ = 1, & \text{si } \left(\frac{dx}{dt} \right) > 0 \end{cases} \quad (3.2)$$

Cabe destacar que la fuerza de rozamiento en (3.1) no depende de la magnitud del modulo de la velocidad, sino de su signo. Además, deberán conocerse las condiciones iniciales $x(0) = x_0$ y $\frac{dx}{dt}(0) = v_0$, así como el coeficiente de rozamiento estático $\mu_s > \mu_k$ entre las superficies en contacto del bloque y la horizontal.

De esta forma, si suponemos que el bloque parte del reposo (velocidad inicial nula, $v_0 = 0$) y que se cumple la condición de que la fuerza elástica ejercida por el muelle debe de superar la debida al rozamiento estático,

$$k x_0 > \mu_s N, \quad (3.3)$$

el bloque se desplaza desde la posición inicial. De donde podemos deducir que la condición necesaria para que el bloque se desplace es:

$$x_0 > \mu_s \frac{mg}{k}, \quad (3.4)$$

Y teniendo en cuenta que $\omega_0 = \sqrt{\frac{k}{m}}$ podemos obtener:

$$x_0 > \frac{\mu_s g}{\omega_0^2}. \quad (3.5)$$

Como vemos en la figura 3.1, cuando se cumpla la condición (3.5), el bloque comenzará a desplazarse hacia la izquierda. Si por el contrario la fuerza elástica es igual o menor que la fuerza de rozamiento estático, entonces el bloque se mantendrá en reposo.

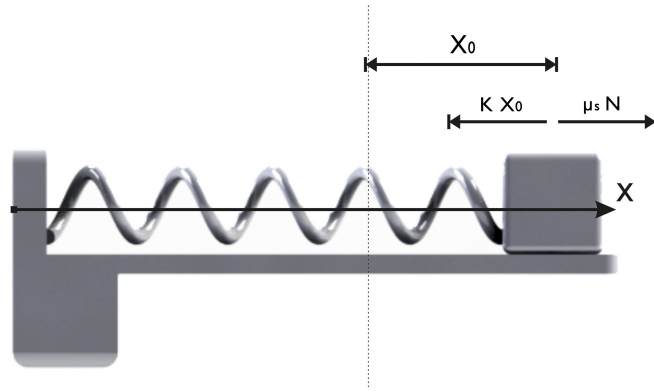


Figura 3.1: Se muestra un sistema masa-resorte, que descansa sobre una superficie horizontal con rozamiento estático/dinámico.

Siempre que el bloque se detenga, habrá que considerar una condición de desplazamiento para saber si continua desplazándose o se detiene.

3.2. Cálculo del primer tramo de desplazamiento.

Ahora, supongamos que la posición inicial cumple con la condición (3.4), suficiente para que se produzca desplazamiento, deducimos que la velocidad tendrá sentido negativo si $t \in (0, t_1)$, donde t_1 representa el primer instante de tiempo donde se anula la velocidad. Para una posición y velocidades iniciales tal que $x(0) > 0$ y $\frac{dx}{dt}(0) = 0$, la aceleración se puede obtener de la ecuación (3.1) y toma el valor de $m \frac{d^2x}{dt^2}(0) = -k x(0) < 0$. Eso significa que $\text{sgn}\left(\frac{dx}{dt}\right) = -1$ y nuestra ecuación (3.1) pasará a ser:

$$m \frac{d^2x}{dt^2} + kx - \mu_k N = 0, \quad (3.6)$$

donde si dividimos todo por la masa, y si denotamos que $\alpha = \mu_k g$ tendremos:

$$\frac{d^2x}{dt^2} + \omega_0^2 x - \alpha = 0, \quad (3.7)$$

Este sistema además será equivalente a:

$$\frac{d^2y}{dt^2} + \omega_0^2 y = 0; \quad (3.8)$$

donde:

$$y(t) = x(t) - \frac{\alpha}{\omega_0^2}, \quad (3.9)$$

y las condiciones iniciales serán

$$y(0) = x(0) - \frac{\alpha}{\omega_0^2}; \quad (3.10)$$

$$\frac{dy(0)}{dt} = \frac{dx(0)}{dt} = 0. \quad (3.11)$$

La ecuación (3.8) ha sido estudiada en detalle en el capítulo anterior. Por tanto, su solución será igual a:

$$y(t) = \left(x(0) - \frac{\alpha}{\omega_0^2} \right) \cos(\omega_0 t). \quad (3.12)$$

Deshaciendo el cambio de variables obtenemos entonces:

$$x(t) = y(t) + \frac{\alpha}{\omega_0^2} = x(0) \cos(\omega_0 t) + \frac{\alpha}{\omega_0^2} [1 - \cos(\omega_0 t)]. \quad (3.13)$$

De las ecuaciones anteriores obtenemos la posición, la velocidad y la aceleración de la partícula:

$$x(t) = \frac{\alpha}{\omega_0^2} + \left(x(0) - \frac{\alpha}{\omega_0^2} \right) \cos(\omega_0 t), \quad 0 \leq t \leq t_1, \quad (3.14)$$

$$v(t) = -\omega_0 \left(x(0) - \frac{\alpha}{\omega_0^2} \right) \text{sen}(\omega_0 t), \quad (3.15)$$

$$a(t) = -\omega_0^2 \left(x(0) - \frac{\alpha}{\omega_0^2} \right) \cos(\omega_0 t), \quad (3.16)$$

$$\text{para } x(0) > \frac{\mu_s g}{\omega_0^2} > \frac{\mu_k g}{\omega_0^2} = \frac{\alpha}{\omega^2}.$$

La posición $x(t)$ representa entonces una función que oscila de manera periódica en torno a $\frac{\alpha}{\omega_0^2}$. Además, la amplitud de las oscilaciones es $\left(x(0) - \frac{\alpha}{\omega_0^2} \right)$ mientras que la velocidad sea distinta de cero. En el momento en que la velocidad $\frac{dx}{dt} = 0$ la ecuación (3.1) ya no será válida.

Igualando la ecuación (3.15) a 0, obtenemos el valor del momento t_1 que delimita nuestro primer tramo:

$$\text{sen}(\omega_0 t) = 0; \quad \Rightarrow \quad \omega_0 t = n\pi; \quad n = 0, 1, 2... \quad (3.17)$$

Para $n = 1$, $\omega_0 t_1 = \pi$; $t_1 = \frac{\pi}{\omega_0}$, y por tanto entonces la solución (3.14) será válida para $0 \leq t \leq \frac{\pi}{\omega_0}$.

3.3. Cálculo del segundo tramo.

Lo que hemos descrito entonces es el comportamiento de la partícula para el primer tramo que va desde el momento inicial hasta $t_1 = \frac{\pi}{\omega_0}$. Nuestro objetivo será conocer el comportamiento del bloque en cualquier instante de tiempo, de manera que repetiremos el procedimiento anterior para obtener la ecuación del desplazamiento y la velocidad en el siguiente tramo $\frac{\pi}{\omega_0} \leq t \leq t_2$.

En este segundo tramo, la partícula o bien permanece en reposo, o se mueve siempre que la condición de desplazamiento se verifique. De forma análoga a (3.3), la partícula se mueve si:

$$k |x(t_1)| > \mu_s N, \quad (3.18)$$

donde

$$x(t_1) = \frac{2\alpha}{\omega_0^2} - x(0). \quad (3.19)$$

También es importante determinar el signo de la aceleración en $t_1 = \pi/\omega_0$. De la ecuación (3.16), obtenemos

$$a(t_1) = \omega_0^2 \left(x(0) - \frac{\alpha}{\omega_0^2} \right), \quad (3.20)$$

que es mayor que cero como consecuencia de que $\mu_s > \mu_k$ y de que $x(0)$ cumple con la condición (3.3). Esto significa que el cuerpo se moverá a la derecha y por tanto, el signo de la velocidad será positivo. De manera que ahora la ecuación de movimiento será:

$$\frac{d^2x}{dt^2} + \omega_0^2 x + \alpha = 0, \quad t_1 < t \leq t_2, \quad (3.21)$$

y las condiciones iniciales estarán dadas por (3.19) y por $v(t_1) = 0$. El instante de tiempo t_2 , es el primer instante de tiempo, posterior a t_1 donde la velocidad se anula. Este sistema de nuevo será equivalente a:

$$\frac{d^2y}{dt^2} + \omega_0^2 y = 0; \quad (3.22)$$

$$y(t_1) = x(t_1) + \frac{\alpha}{\omega_0^2}; \quad (3.23)$$

$$\frac{dy(t_1)}{dt} = \frac{dx(t_1)}{dt} = 0, \quad (3.24)$$

estudiado en detalle en el capítulo anterior. Si usamos el siguiente cambio de

variable

$$y(t) = x(t) + \frac{\alpha}{\omega_0^2}. \quad (3.25)$$

Como consecuencia, la solución de la ecuación (3.22) está determinada por

$$y(t) = \left(x(0) - \frac{3\alpha}{\omega_0^2} \right) \cos(\omega_0 t). \quad (3.26)$$

En la variable original, la posición, la velocidad y la aceleración de la partícula son, respectivamente,

$$x(t) = y(t) - \frac{\alpha}{\omega_0^2} = -\frac{\alpha}{\omega_0^2} + \left(x(0) - \frac{3\alpha}{\omega_0^2} \right) \cos(\omega_0 t); \quad t_1 \leq t \leq t_2, \quad (3.27)$$

$$v(t) = -\omega_0 \left(x(0) - \frac{3\alpha}{\omega_0^2} \right) \text{sen}(\omega_0 t); \quad (3.28)$$

$$a(t) = -\omega_0^2 \left(x(0) - \frac{3\alpha}{\omega_0^2} \right) \cos(\omega_0 t). \quad (3.29)$$

De la condición $v(t_2) = 0$, obtenemos que $t_2 = \frac{2\pi}{\omega_0}$. Además,

$$x(t_2) = x(0) - \frac{4\alpha}{\omega_0^2}, \quad (3.30)$$

$$a(t_2) = -\omega_0^2 \left(x(0) - \frac{3\alpha}{\omega_0^2} \right). \quad (3.31)$$

Después del instante t_2 si la fuerza elástica vence a la del rozamiento estático, entonces el cuerpo continua moviéndose. De lo contrario, a partir de ese momento permanece en reposo. El procedimiento que hemos explicado habría que continuarlo hasta que el cuerpo se detenga.

3.4. Cálculo de las ecuaciones generales.

Una vez hemos obtenido las ecuaciones anteriores, podemos deducir las ecuaciones generales en función del tramo en que nos encontramos.

Primero, indicaremos que la ecuación del desplazamiento en función de $n = 0, 1, 2, 3, \dots$, es igual a

$$x(t) = \frac{\alpha}{\omega_0^2} (-1)^n + \left(x(0) - \frac{(2n+1)\alpha}{\omega_0^2} \right) \cos(\omega_0 t), \quad t_n \leq t \leq t_{n+1}. \quad (3.32)$$

El valor de $n + 1$ indica el tramo en el que nos encontramos. A partir de esta expresión general, podemos obtener la velocidad y la aceleración en cada inter-

valo.

Segundo, es importante determinar los momentos para el cual la derivada de x sea nula (momentos en que generalmente cambia el sentido de desplazamiento):

$$t_n = \frac{n\pi}{\omega_0}. \quad (3.33)$$

Finalmente, cada uno de los tramos lo habíamos calculado considerando que en el momento en que cambiaba el sentido de desplazamiento y la velocidad era nula, la posición del bloque era tal que la fuerza del muelle era capaz de superar la fuerza de rozamiento estática necesaria para que el bloque comenzara a desplazarse de nuevo. Así, teníamos:

$$x_0 > \frac{\mu_s g}{\omega_0^2}; \quad (3.34)$$

$$|x(t_1)| > \frac{\mu_s g}{\omega_0^2}; \quad (3.35)$$

$$|x(t_2)| > \frac{\mu_s g}{\omega_0^2}. \quad (3.36)$$

En $t = t_n$,

$$|x(t_n)| > \frac{\mu_s g}{\omega_0^2}. \quad (3.37)$$

Sustituyendo (3.32) en (3.37), y evaluando la expresión obtenida en $t = t_n$ (3.33),

$$\left| x(0) - \frac{2n\alpha}{\omega_0^2} \right| > \frac{\mu_s g}{\omega_0^2}. \quad (3.38)$$

Si $x(0) > \frac{2n\alpha}{\omega_0^2}$, entonces

$$n < \left(\frac{x_0 k}{\mu_k N} - \frac{\mu_s}{\mu_k} \right) \frac{1}{2}. \quad (3.39)$$

En este caso debemos escoger el número entero que más se acerque a $\left(\frac{x_0 k}{\mu_k N} - \frac{\mu_s}{\mu_k} \right) \frac{1}{2}$.

Sin embargo, si $x(0) < \frac{2n\alpha}{\omega_0^2}$, entonces

$$n > \left(\frac{x_0 k}{\mu_k N} + \frac{\mu_s}{\mu_k} \right) \frac{1}{2}. \quad (3.40)$$

Aquí tendríamos que escoger el menor número entero que cumpla con esta condición.

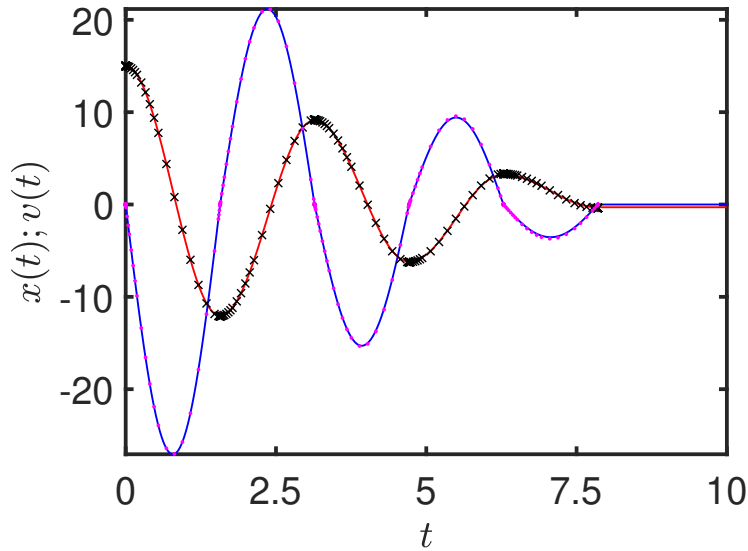


Figura 3.2: Oscilador armónico amortiguado por fricción estática y dinámica entre superficies. Podemos observar cómo el oscilador realiza una serie de semiciclos antes de quedarse anclado debido a que no es capaz de superar la fuerza ejercida por el rozamiento estático. Condiciones iniciales: $x(0) = 15$, $v(0) = 0$. Parámetros: $\omega_0 = 2$, $\alpha = 5,88$ y $\mu_s/\mu_k = 1,2$.

De las ecuaciones generales podemos obtener la representación gráfica de $x(t)$ y $v(t)$ para los parámetros indicados en la figura 3.2. Como se observa, los resultados analíticos han sido verificados numéricamente. Curiosamente, en la representación de $v(t)$ se observa que en los puntos t_n , la aceleración es una función discontinua. Esta discontinuidad puede calcularse con las soluciones analíticas de la aceleración en los distintos tramos. Por ejemplo, si fijamos un valor de t_n y calculamos el límite de $a(t)$ por la derecha y por la izquierda, podemos comprobar que son diferentes. Esta discontinuidad es debida a la fuerza de rozamiento. En dos tramos consecutivos, $t_{j-1} \leq t \leq t_j$ y $t_j \leq t \leq t_{j+1}$ donde la partícula se mueva las ecuaciones de movimiento estarán representadas por

$$m \frac{d^2 x}{dt^2} + k x + \mu_k N = 0, \quad (3.41)$$

$$m \frac{d^2 x}{dt^2} + k x - \mu_k N = 0, \quad (3.42)$$

respectivamente. Tomando el límite cuando $t \rightarrow t_j$ en las dos ecuaciones anteriores y restándolas, obtenemos

$$\lim_{t \rightarrow t_j^-} \frac{d^2 x}{dt^2} - \lim_{t \rightarrow t_j^+} \frac{d^2 x}{dt^2} = -2\mu_k g. \quad (3.43)$$

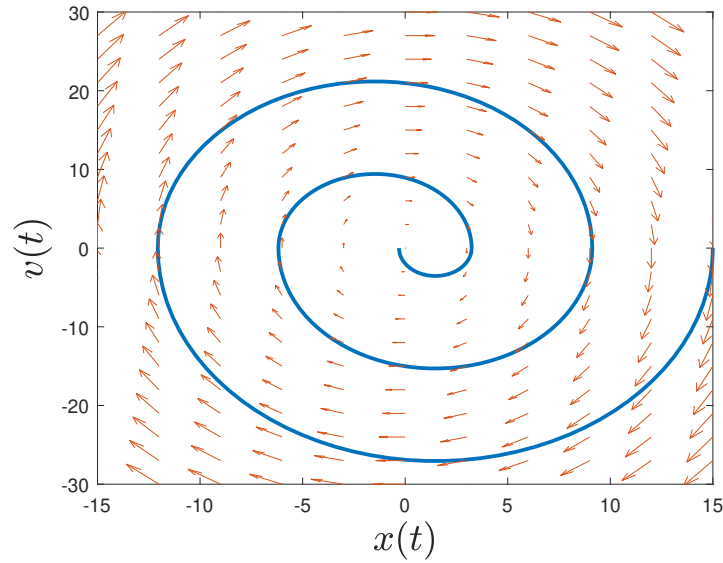


Figura 3.3: Plano de fase del oscilador armónico amortiguado por fricción estática y dinámica entre superficies. Parámetros y condiciones iniciales como en la figura 3.2.

Para los parámetros de la figura 3.2, la relación (3.39) predice que $n < 4,5$ y la (3.40) que $n > 5,7$, por tanto $n = 4$. Quiere decir que la partícula se mueve 5 tramos completos hasta que se detiene, tal y como se observa en la figura 3.2.

Como obtuvimos en el capítulo anterior, también podemos observar que las órbitas en el plano de fase correspondiente tienden a un punto. Aunque la espiral que se forma en la figura 3.3 no es exactamente igual a la obtenida en el análisis del oscilador armónico con la disipación de Stokes y además no tienden al punto $(0, 0)$.

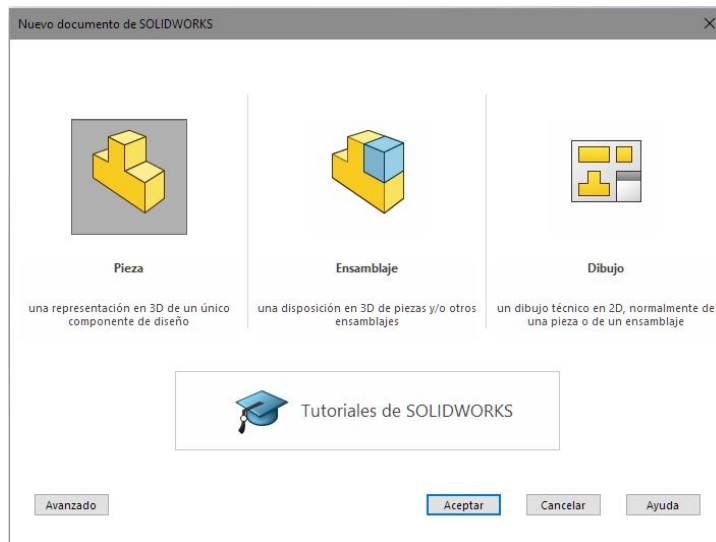
Capítulo 4

Modelado y animación en SolidWorks

Una vez que hemos obtenido las ecuaciones en los capítulos anteriores será de interés poder generar una serie de datos que nos permitan simular el comportamiento de los osciladores para las condiciones estudiadas y representar gráficamente del fenómeno. En este capítulo se detalla la creación y animación de un modelo (sistema masa-resorte) a través del software de modelado sólido Solidworks, con el fin de obtener una animación a partir de las ecuaciones detalladas en los capítulos anteriores y los conjuntos de datos obtenidos con los programas de *Matlab*. También se ahonda en los conocimientos de modelado sólido, de simulación, de análisis de movimiento, y en las herramientas que pone SolidWorks a nuestra disposición.

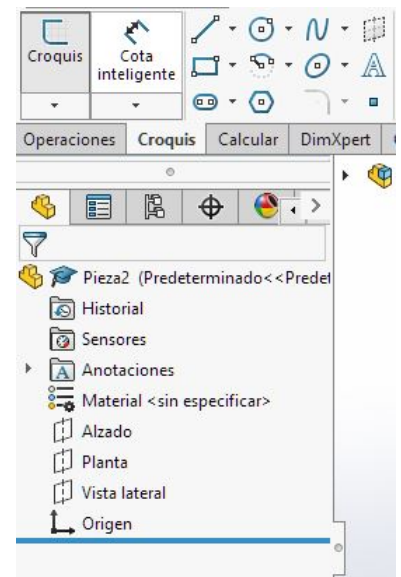
4.1. Modelado de los elementos necesarios para la simulación

Para comenzar, podemos crear un nuevo archivo pasando el ratón por el icono de **Solidworks** situado arriba a la izquierda. Se desplegará una barra en la que seleccionaremos **Archivo nuevo**, luego Pieza en la ventana que se abrirá y finalmente **Aceptar**.



Situada en la parte superior de la ventana se encuentran las barras de cada apartado de Solidworks; seleccionamos croquis, y veremos las herramientas con las que trabajaremos, que serán las siguientes:

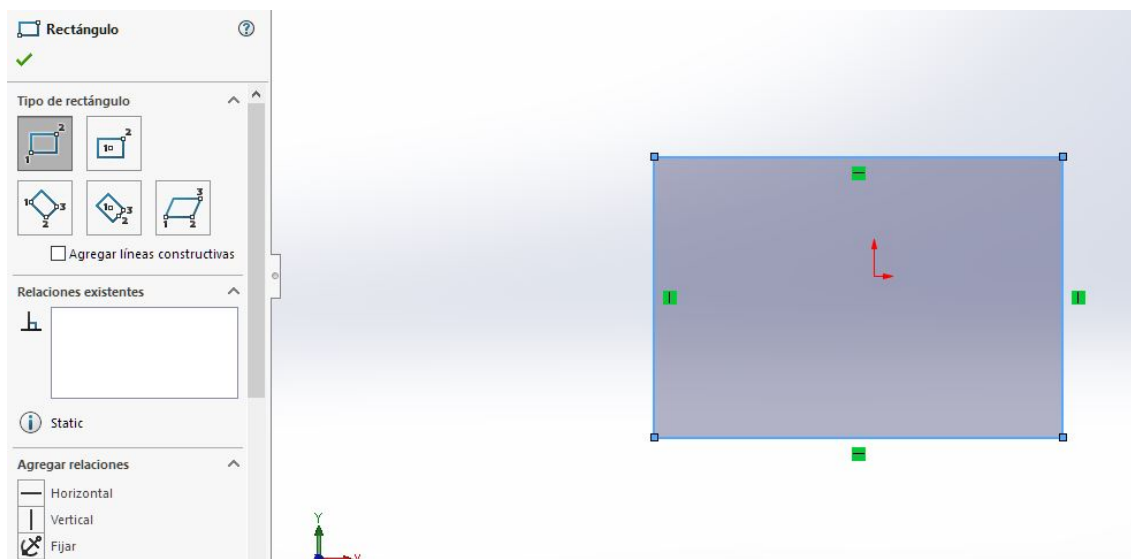
- Línea: Permite crear segmentos.
- Círculo: Permite crear círculos en base a distintos parámetros.
- Rectángulo: Permite crear cuadrados y rectángulos.



También podremos ver en el lado izquierdo de la pantalla el árbol de elementos y operaciones de la pieza, donde debemos seleccionar el **Alzado** de los tres planos principales sobre el que vamos a trabajar.

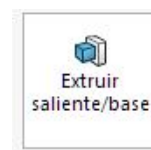
Comenzaremos creando un bloque que nos servirá de elemento móvil para realizar las simulaciones, luego usaremos la herramienta **Rectángulo**, y a la izquierda cambiará la ventana contextual para mostrar las opciones sobre cómo crear el rectángulo. Pinchamos en la zona de trabajo, desplazamos y volvemos a pinchar, y tendremos un rectángulo al cual podremos escoger los parámetros para las longitudes de los lados en la ventana contextual de la izquierda.

4.1. MODELADO DE LOS ELEMENTOS NECESARIOS PARA LA SIMULACIÓN41

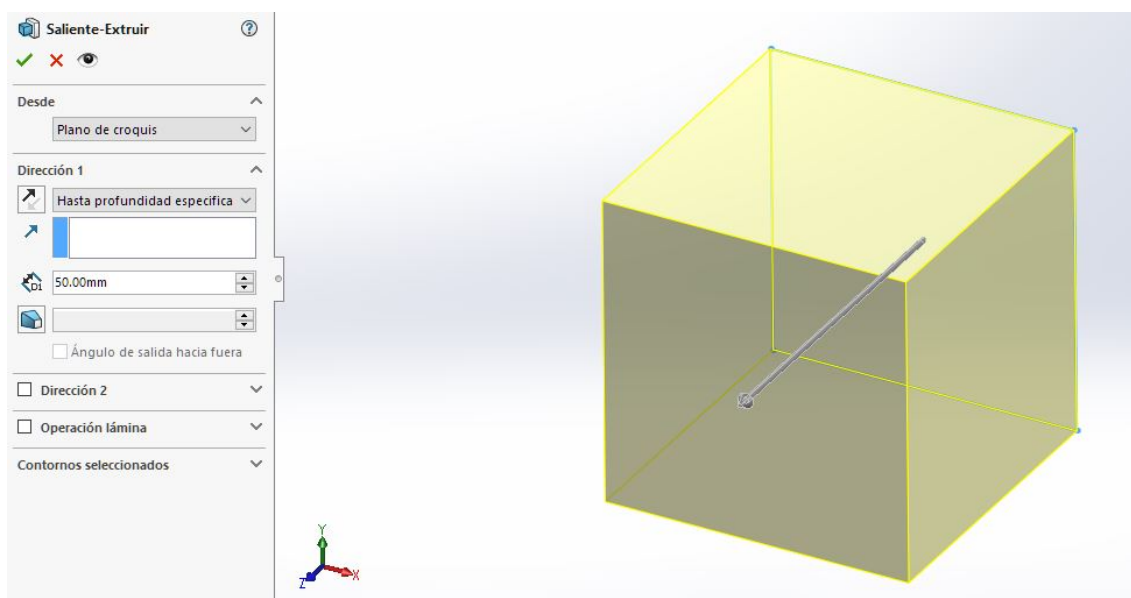


Finalmente tendremos que darle a la pequeña flecha verde en la parte superior de la ventana contextual.

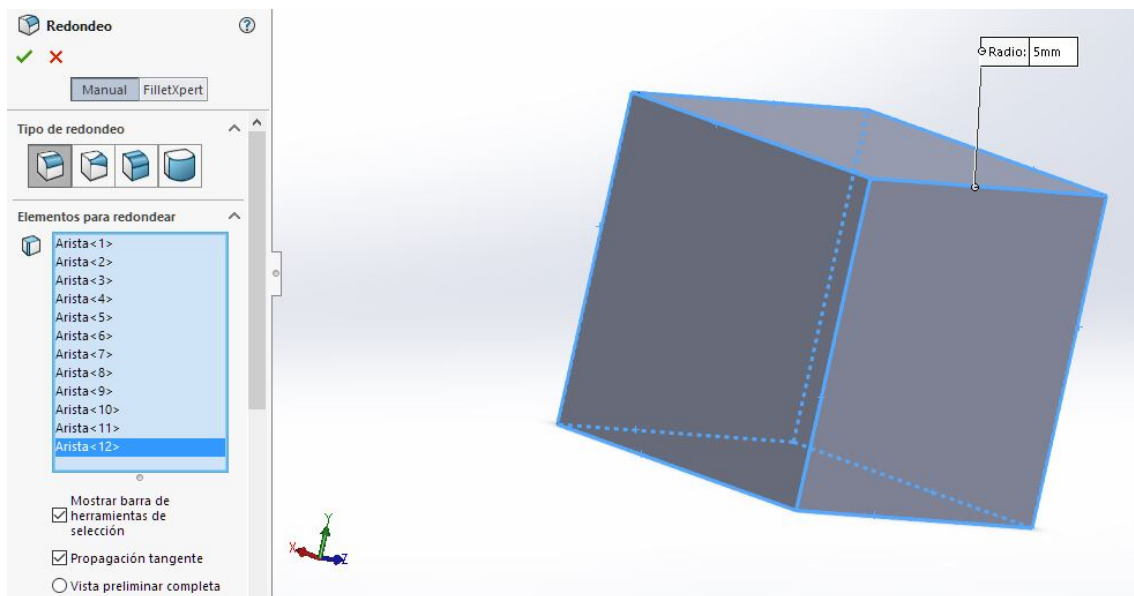
Para salir del croquis daremos al icono azul con una flecha situado arriba a la derecha, y procederemos a seleccionar **Operaciones**, situado en la barra superior junto a **Croquis**. De todas las herramientas que veremos en la parte superior, seleccionaremos **Extruir Saliente/Base**, arriba a la izquierda mientras tenemos seleccionado el cuadrado.



En la ventana contextual a la izquierda podremos modificar la longitud de la extrusión, y cuando demos a aceptar podremos ver el sólido realizado, que intentaremos que sea cúbico.



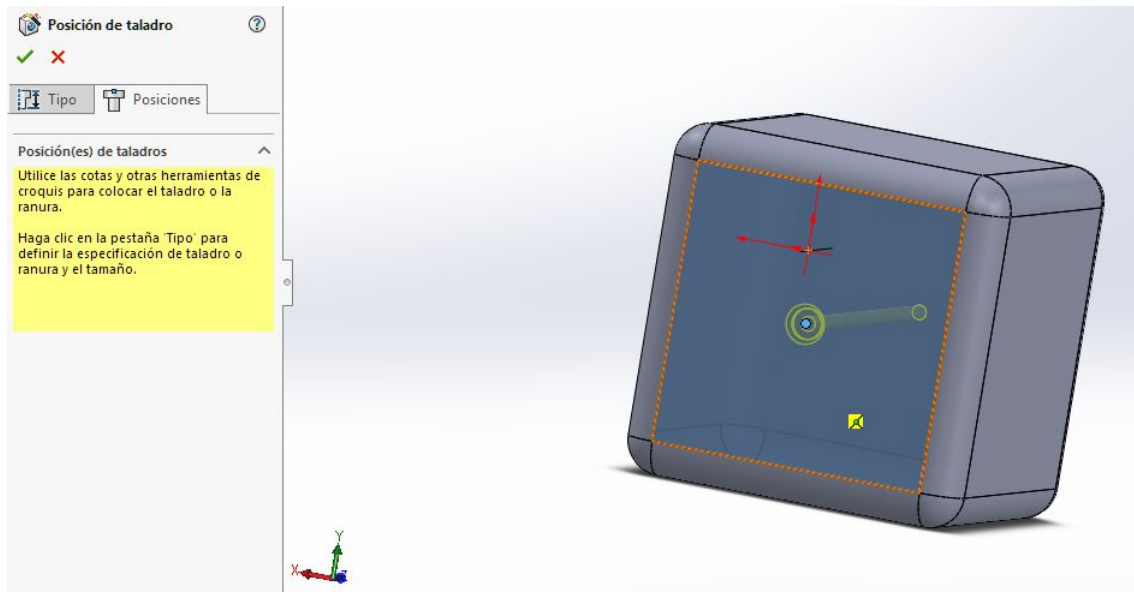
Ahora seleccionaremos **Redondeado**, situado arriba en el centro y escogeremos todas las aristas de nuestro sólido empleando el clic central del ratón, manteniendo y arrastrando para girar el sólido. En la ventana contextual, si nos desplazamos hacia abajo usando la rueda del ratón podremos ajustar el radio del redondeo, y ejecutando aceptaremos observaremos el resultado.



Finalmente, en las herramientas escogeremos el **Asistente para taladros**, y en la ventana contextual elegimos **Posiciones** y luego **Croquis 3D**, que nos permitirá seleccionar en una de las caras del cubo dónde queremos realizar el taladro; intentaremos que esté lo más centrado posible, aunque no nos influirá realmente la posición. Damos a aceptar y podemos ver el taladro realizado.

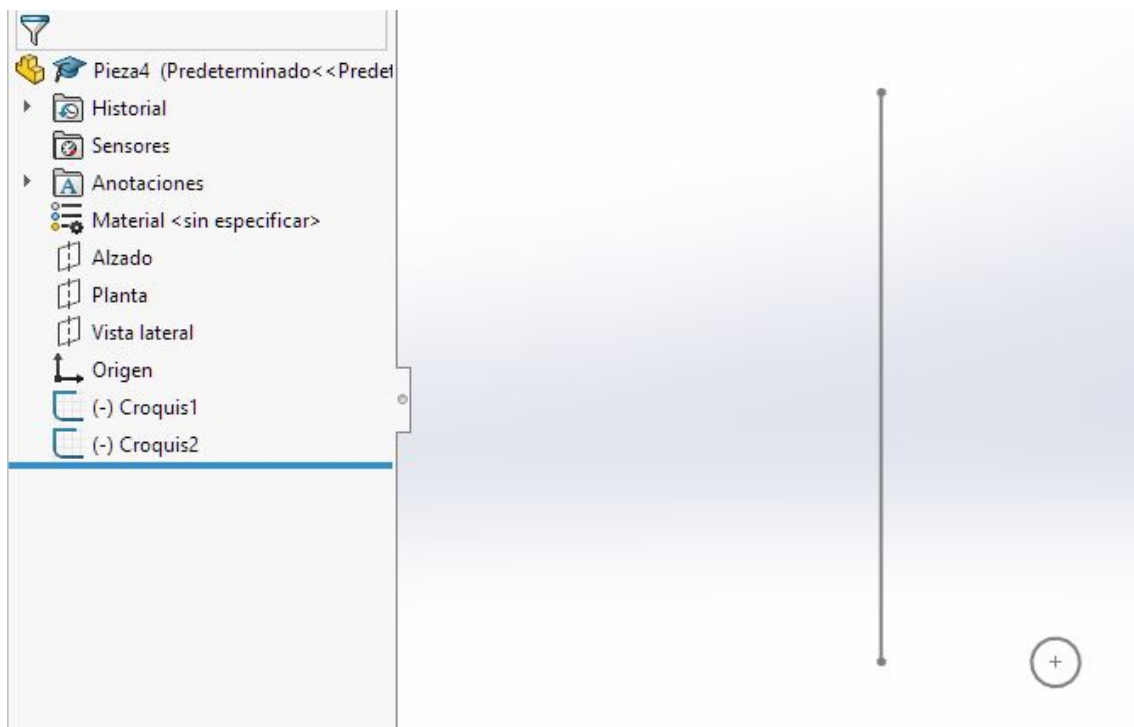


4.1. MODELADO DE LOS ELEMENTOS NECESARIOS PARA LA SIMULACIÓN⁴³



Para acabar, hacemos clic en el icono de **Solidworks**, seleccionamos **Archivo** > **Guardar como**, y guardamos nuestra primera pieza (Formato .sldprt).

Ahora, para el modelado del muelle dinámico volvemos a crear una nueva pieza y abrimos la pestaña de Croquis. El primer croquis que realizaremos será un segmento vertical, salimos del croquis y haciendo clic derecho sobre el árbol seleccionamos crear un nuevo croquis, y creamos un círculo.



Es muy importante no establecer la longitud del segmento, pues durante el ensamblaje será una medida que variará con la posición de los elementos, y

también será muy importante que el segmento y el círculo estén en croquis diferentes.

Vamos a la pestaña de Operaciones de nuevo y seleccionamos la herramienta de **Saliente/Base barrido**. El círculo será el elemento de barrido y el segmento será la guía, que seleccionamos haciendo clic en cada casilla y luego en el croquis correspondiente.

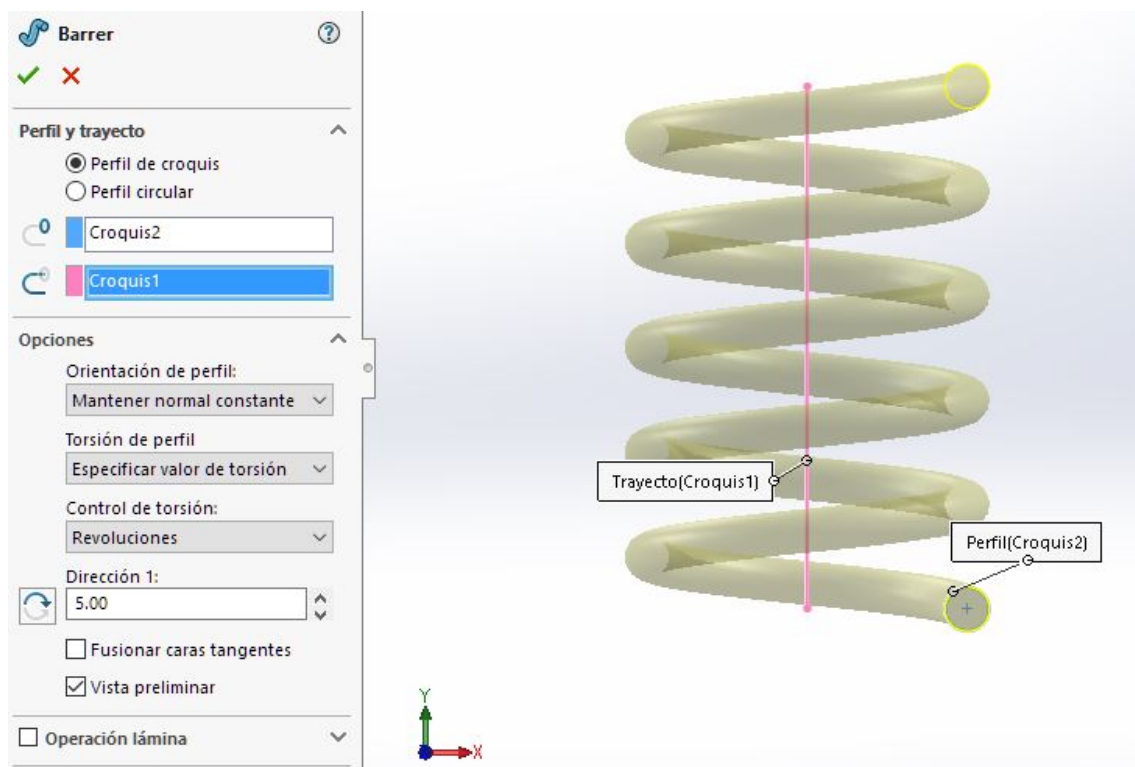
Luego, tenemos que seleccionar en las pestañas desplegables

Orientación > Mantener la normal constante

Torsión > Especificar valor de torsión

Control > Revoluciones

Y en la **Dirección 1** introduciremos un valor arbitrario.

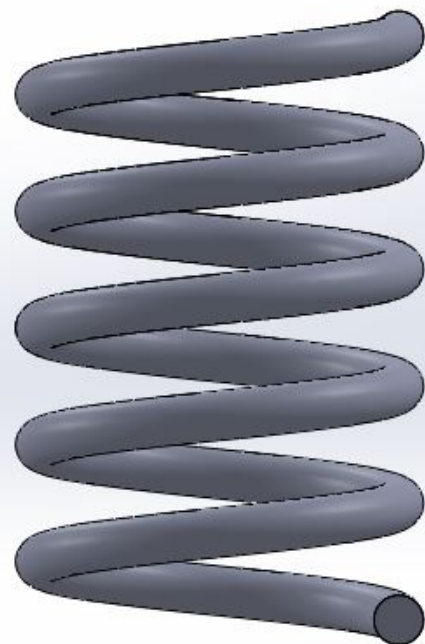
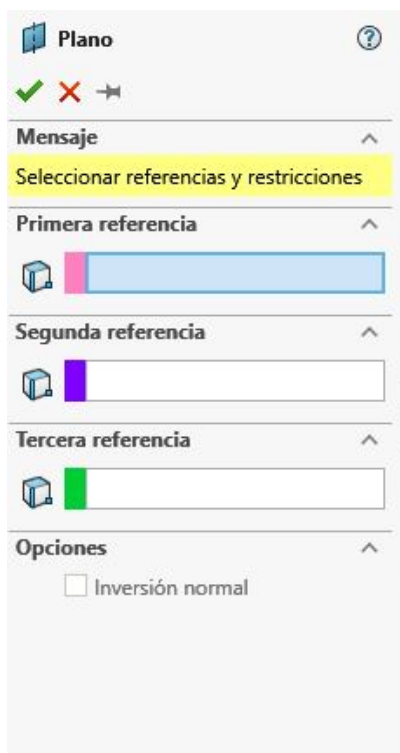


Lo que estamos haciendo es crear una espiral alrededor del segmento donde el valor introducido en Dirección 1 es el número de vueltas que da la espira en su trayectoria, luego lo ajustaremos a nuestro gusto.

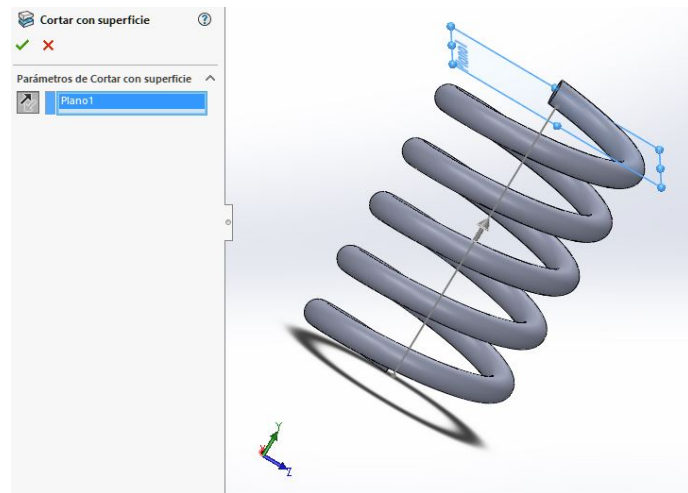
Damos aceptar y veremos la espiral creada. En el árbol debemos seleccionar la operación que acabamos de realizar, **Barrer**, y haciendo clic en la flecha de al lado desplegaremos los croquis que usamos originalmente. Haremos clic derecho sobre el croquis del segmento y luego le damos a **Ver** (Símbolo de un ojo abierto), lo que nos permitirá ver el segmento que se había ocultado automáticamente.

4.1. MODELADO DE LOS ELEMENTOS NECESARIOS PARA LA SIMULACIÓN⁴⁵

Ahora recortaremos los extremos de la espiral para que se apoye en dos superficies planas. Para ello, arriba a la derecha podemos encontrar la herramienta de **Geometría de referencia**, y al hacer clic podremos seleccionar **Plano** y crear uno con las referencias que podemos introducir en la ventana contextual de la izquierda. La primera referencia será el extremo superior del segmento, y la segunda referencia será el plano de Planta que podemos seleccionar en el árbol, de manera que cuando le demos a aceptar podremos ver el nuevo plano creado.



Ahora, necesitaremos activar las herramientas de **Superficies**. Para ello hacemos clic derecho sobre la barra en la que se encuentran Croquis y **Operaciones**, lo cual nos abrirá una pestaña donde podemos seleccionar que nos active Superficies. En **Superficies**, seleccionamos la herramienta arriba de Cortar con superficie. Haremos clic en el plano que acabamos de crear, y dándole a la flecha doble de la casilla nos aseguraremos de que la flecha que aparece sobre nuestro espiral apunte hacia el exterior, pues indicará las partes a eliminar. Damos a aceptar, lo cual nos dejará un extremo plano, y repetimos, pero empleando el plano de planta para obtener el otro extremo plano también.

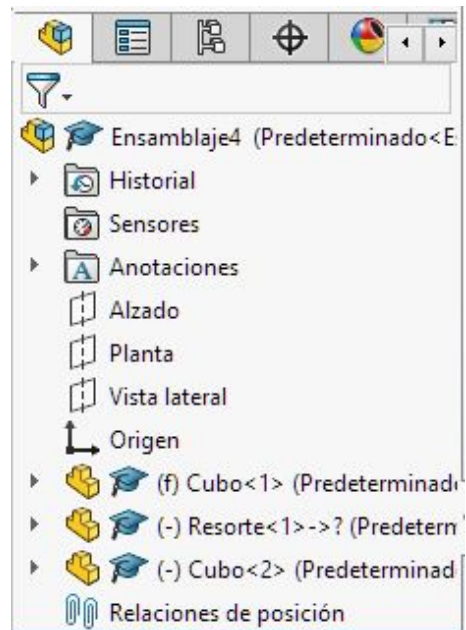


Guardamos la pieza, y ya tenemos las dos preparadas.

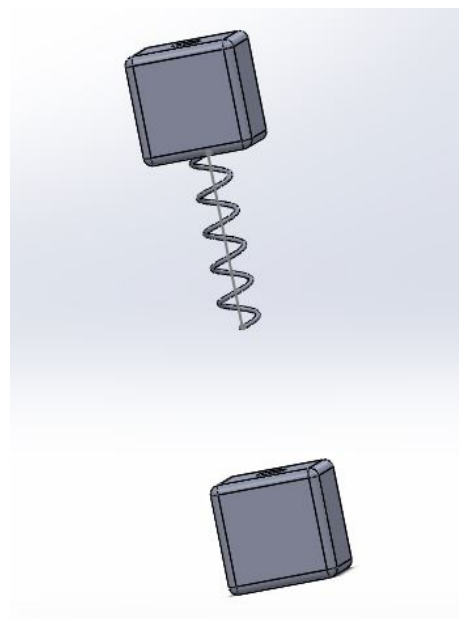
4.2. Creación del ensamblaje, animación y simulación

En **Archivo nuevo** seleccionaremos **Ensamblaje**, y se nos abrirá una pestaña donde poder importar el cubo y el muelle que hemos creado anteriormente.

Si nos fijamos en el árbol veremos que cada pieza tiene un (f) o (-) junto a su nombre, que indican si la pieza está fija o flota respectivamente, y que podremos cambiar haciendo clic derecho sobre cada pieza en el árbol. Debemos fijar el bloque y dejar libre el muelle.



Ahora, seleccionamos **Archivo**, situado arriba, luego **Insertar > Componente > Pieza existente**, y volvemos a insertar el cubo, lo cual nos creará una segunda instancia de la pieza, de manera que nos quedará algo similar a lo de la siguiente captura.



Ahora crearemos las relaciones de posición necesarias para el correcto funcionamiento del movimiento de los elementos.

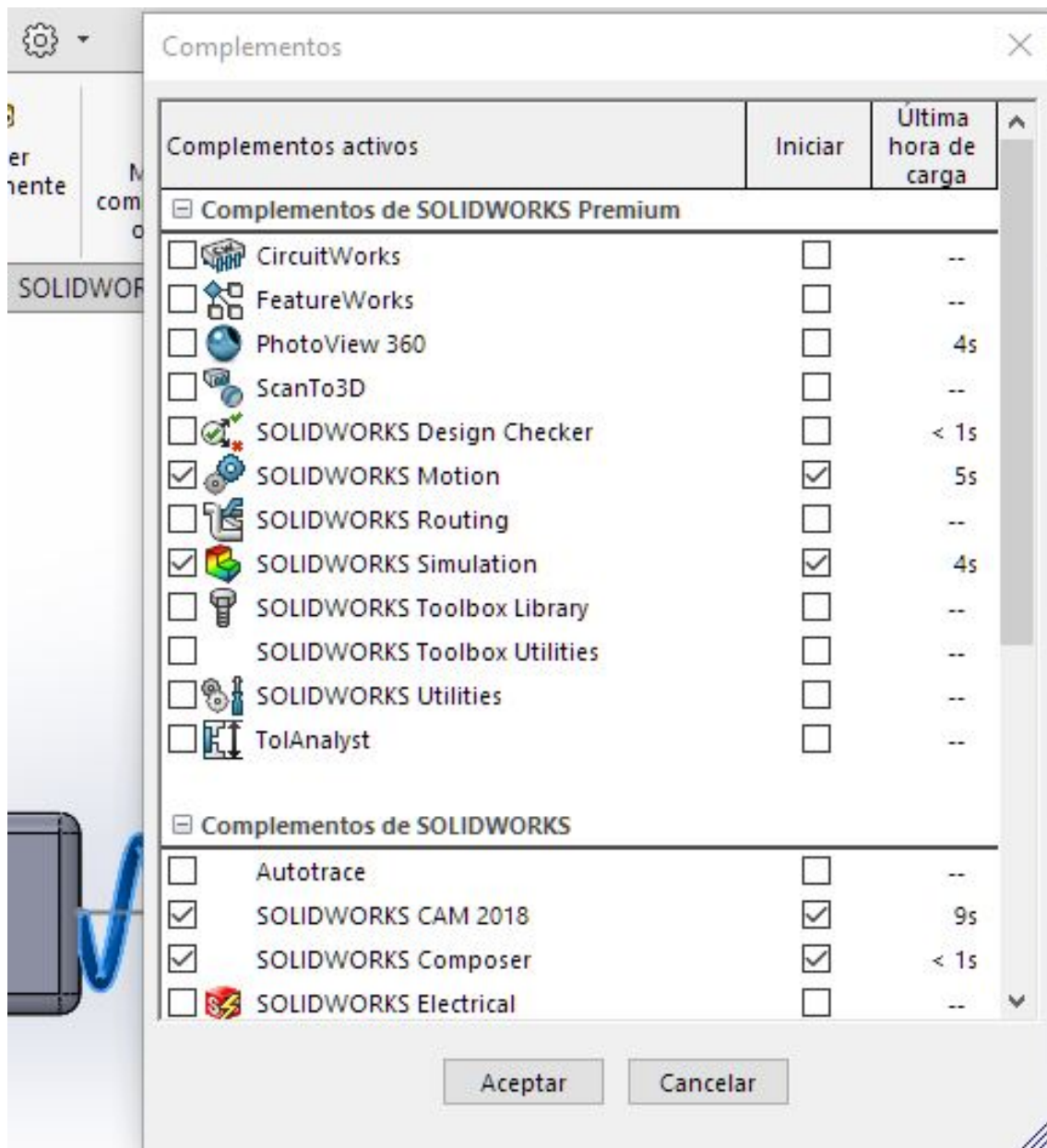
Queremos que el cubo libre se desplace respecto del fijo como si se moviera por unas guías, luego seleccionaremos **Relaciones de posición**, indicado arriba por un símbolo de un clip. Se abrirá la ventana contextual, y empleando Ctrl+Clic izquierdo seleccionaremos las caras de los cubos de dos en dos, seleccionando la relación Coincidentes en la ventana contextual y dándole a aceptar. Será suficiente hacerlo para dos caras perpendiculares de los cubos, y comprobamos que si hacemos clic en el cubo libre y arrastramos el movimiento que presenta se corresponde con el deseado.



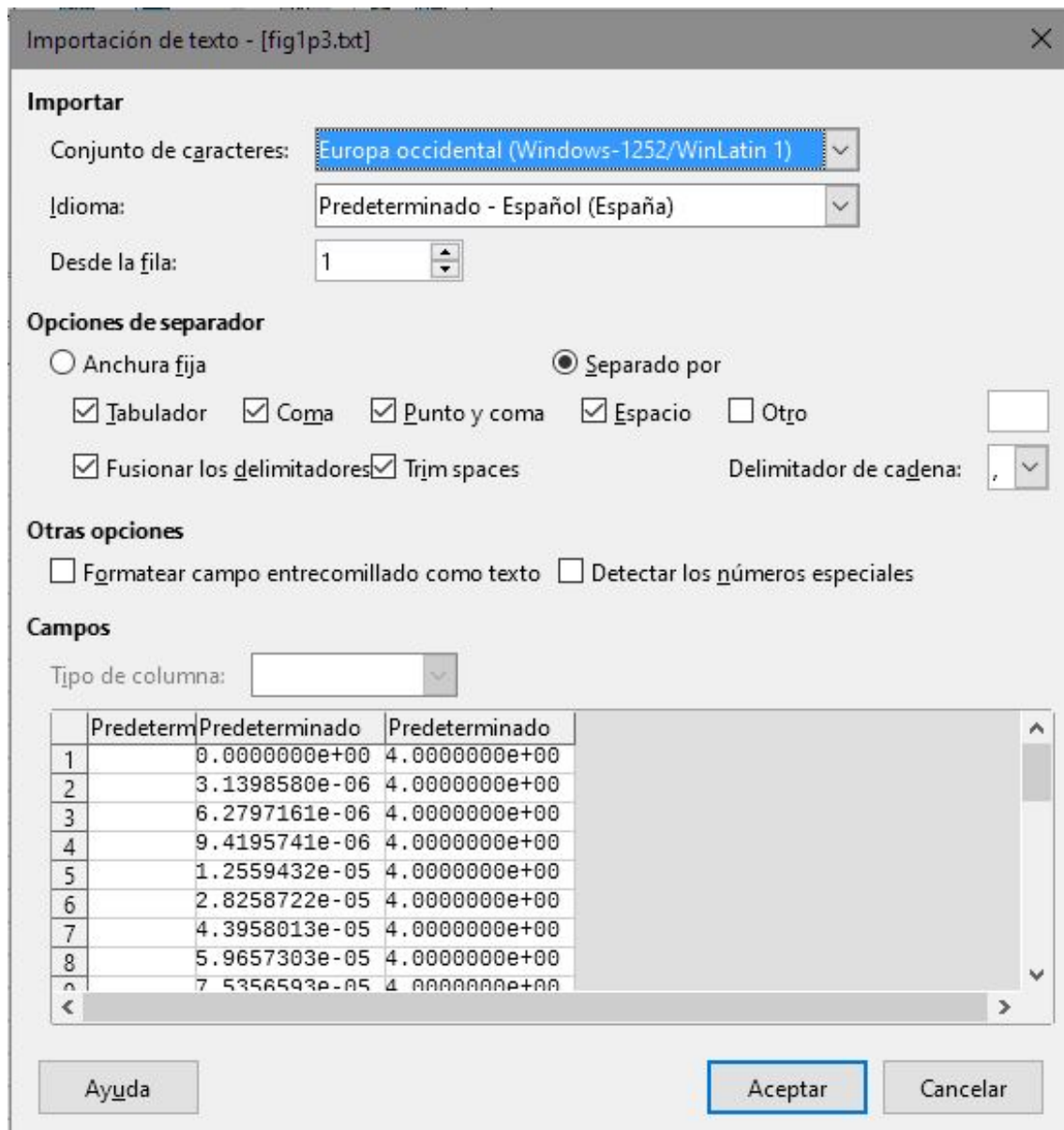
Ahora seleccionamos la guía (Eje) del muelle y uno de los taladros en cualquiera de los dos cubos y repetimos para una relación concéntrica. Será importante hacer clic en la casilla de **Bloquear rotación** situada justo debajo para que el muelle no rote al desplazar los cubos, y al darle a aceptar veremos los elementos alineados.

Ahora, también estableceremos una relación de coincidencia para la superficie plana inferior del muelle con la cara adyacente del cubo correspondiente, y finalmente usaremos el árbol para entrar en el croquis del segmento de eje del muelle, donde seleccionaremos el extremo libre del segmento y **el borde del taladro del cubo más cercano** (Cualquier otra superficie no haría aparecer el comando de **Relación de posición** siquiera) y estableceremos una relación de coincidencia.

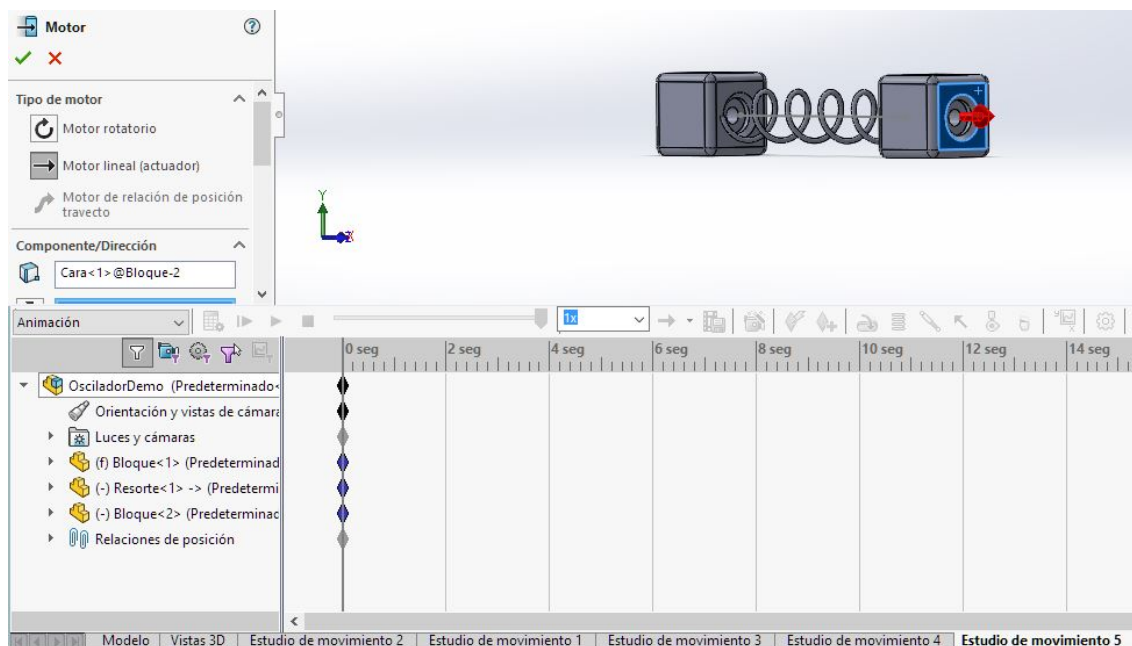
Al salir del croquis, si movemos el cubo libre veremos que el muelle se desplace con él, y si le damos a reconstruir, indicado por un símbolo de un **Semáforo** situado arriba de la pantalla, el modelo se actualizará con la forma y elongación del muelle correctas.



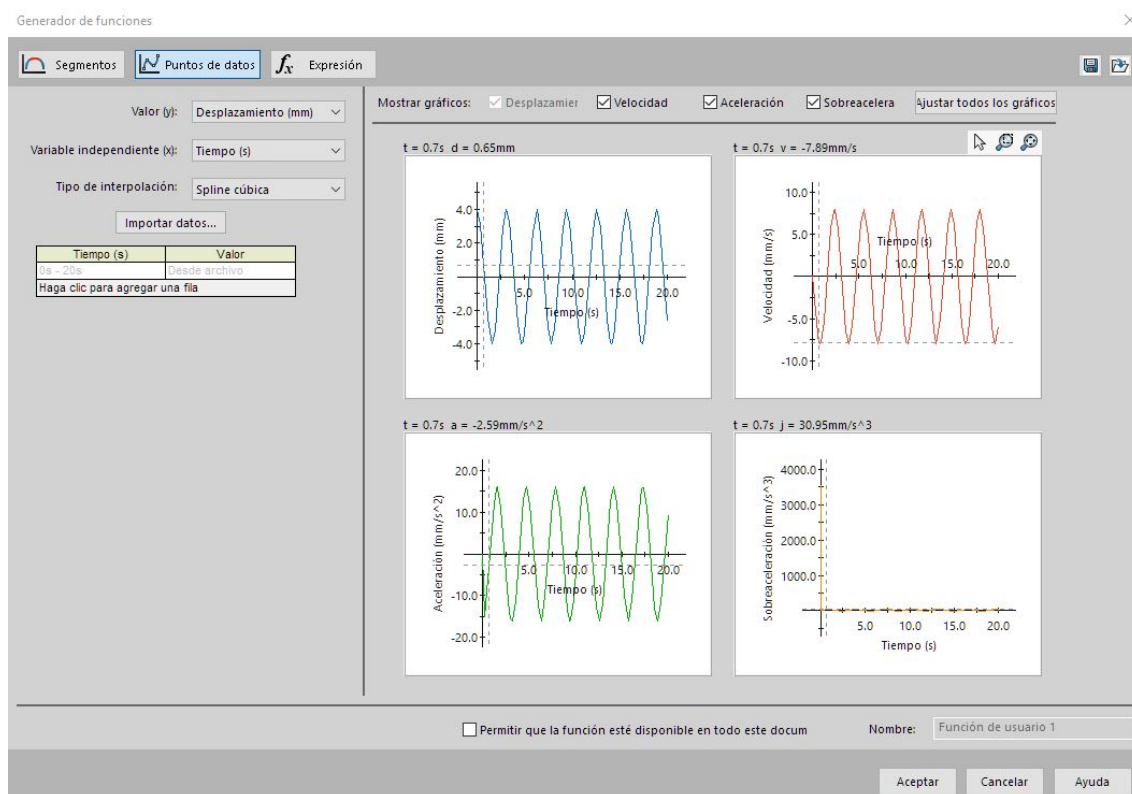
Para realizar la animación y la simulación, debemos primero seleccionar **Complementos** en las opciones, situadas arriba, haciendo clic en la flecha junto al símbolo del engranaje; seleccionamos **Solidworks Motion** y marcamos la casilla, le damos a aceptar y dejamos que cargue. Una vez haya cargado el complemento abajo a la izquierda seleccionaremos **Estudio de Movimiento**, lo cual nos desplegará una línea de tiempo para la animación y simulación.



Para crear la animación primero necesitamos un conjunto de datos apto, y para ello generaremos una tabla de datos desde los programas de *Matlab* que contenga tiempo y posición (En formato .txt). Recomendamos importar el archivo .txt a un programa de cálculo como LibreOffice Calc, de manera que obtengamos un documento con dos columnas independientes. Para ello en LibreOffice Calc seleccionaremos **Archivo > Abrir**, y ajustamos las opciones de separador hasta que tengamos las columnas preparadas, aceptamos y eliminamos cualquier columna sobrante de manera que el tiempo sea la primera columna y la posición la segunda, y guardamos el archivo (En formato .csv).



Regresando a Solidworks, en **Estudio de Movimiento** seleccionamos **Motor lineal**, el cual nos abre la ventana contextual donde debemos seleccionar Motor lineal y seleccionamos la cara externa del cubo móvil perpendicular a la dirección de desplazamiento, y empleando la rueda del ratón nos desplazamos hacia abajo en la ventana contextual hasta ver la pestaña desplegable de **Movimiento**, donde debemos seleccionar **Puntos de datos**.



Se nos abrirá una ventana donde se pueden introducir los datos de manera manual y Solidworks calculará las ecuaciones de movimiento. Debemos seleccionar que el valor Y será el desplazamiento (Por defecto en milímetros) y hacemos clic en Importar datos, que nos dejará seleccionar el archivo (En formato .csv) que contiene los datos. Podemos ver cómo al seleccionar los datos en la ventana anterior aparecen las gráficas correspondientes no sólo al desplazamiento introducido, sino también la velocidad, aceleración e incluso sobreaceleración, calculados de manera automática por el programa.

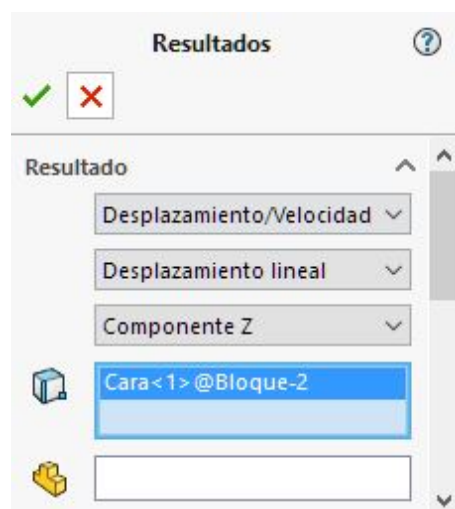


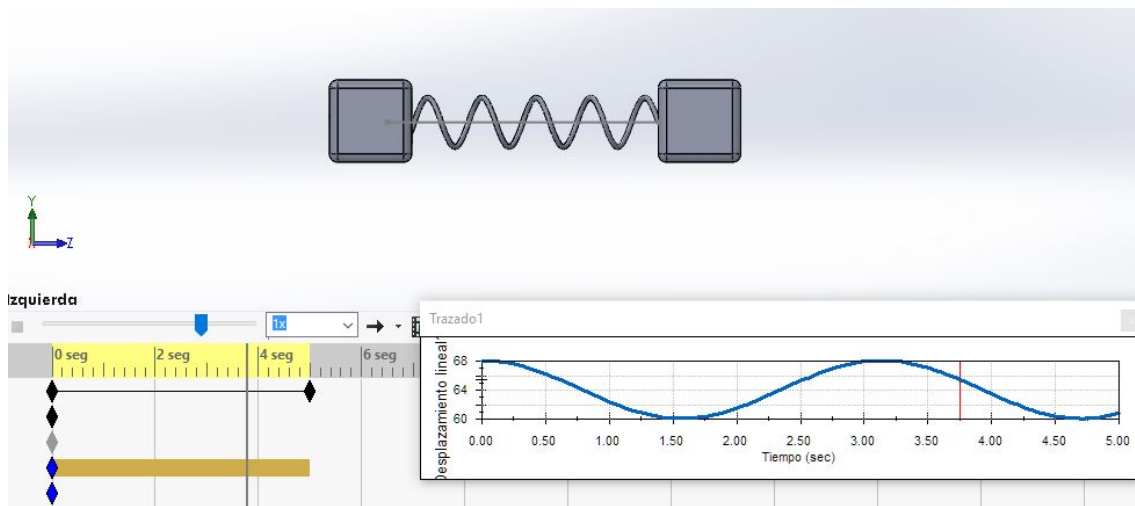
Aceptamos, aceptamos en la ventana contextual del motor, y en la línea de tiempo, junto al símbolo de **Play**, veremos un icono de una tabla llamado Calcular, sobre el que haremos clic y dejaremos que procese. Una vez que haya terminado, podemos hacer clic en **Play** y se activará la animación.

Si queremos ajustar la suavidad de la animación le podemos dar a **Opciones** de la línea de tiempo (Símbolo del engranaje) y en la ventana contextual podremos seleccionar el número de tramas por segundo, que será el número de frames por segundo y por tanto la suavidad de la animación (se recomienda un valor de 30), y volvemos a calcular.

Si además en la línea de tiempo, a la derecha donde pone **Animación**, seleccionamos **Análisis de movimiento** (si no aparece, podemos activarlo como hemos visto antes a través de la pestaña de **Complementos**) y luego en el extremo de la línea de tiempo seleccionamos el símbolo de la gráfica junto al de las opciones, podremos crear una gráfica que muestre resultados en tiempo real.

Seleccionamos los datos que queremos observar y calcular, en qué dirección y para qué elemento (En este caso una de las caras del cubo, en la dirección Z), aceptamos y volvemos a darle a **Calcular**.





Se generará una gráfica que representa lo que hemos seleccionado, y aparecerá un marcador que nos indicará en tiempo real la posición en la gráfica si le damos al **Play** y activamos la animación.

Estas animaciones se pueden guardar. Hemos empleado este método para obtener los siguientes vídeos, que complementan esta memoria:

1. El vídeo 1 es el correspondiente al oscilador armónico simple representado en la figura 2.3.
2. El vídeo 2 es el correspondiente al oscilador armónico simple con amortiguamiento subcrítico representado en la figura 2.4.
3. El vídeo 3 es el correspondiente al oscilador armónico simple, amortiguado y forzado representado en la figura 2.5.
4. El vídeo 4 es el correspondiente al oscilador armónico sometido a fricción estática y dinámica representado en la figura 3.2.

Capítulo 5

Apéndice: Programas numéricos en Matlab.

5.1. Apéndice

Matlab es un programa específico de **Cálculo Numérico**. Puede considerarse como una calculadora programable muy potente. Es muy popular entre estudiantes, **ingenieros**, técnicos e **investigadores** debido a las siguientes características:

- Es un programa **interactivo**.
- Tiene unas **Capacidades Gráficas** potentes y sencillas.
- Posee gran cantidad de **Funciones** de todo tipo.
- Es un **Lenguaje de Programación** de alto nivel similar a *Fortran, C, Pascal* o *Basic*, pero **más fácil** de aprender.

Existen versiones del programa MATLAB no sólo para pequeños PC's, sino también para superordenadores.



5.1.1. Acceso a MATLAB y salida del mismo. Instrucciones de utilidad.

Para acceder a Matlab desde un entorno Unix, tenemos que ejecutar la instrucción matlab desde cualquier ventana. Desde el sistema operativo Windows,

hacemos *doble click* sobre el **icono** de **MATLAB**. Entonces, aparece la ventana de matlab con el *prompt* : >>

```
>>
```

Para salir del mismo ejecutamos `exit` o `quit` desde la ventana de matlab.

A continuación enumeramos algunas instrucciones de utilidad del programa, como por ejemplo:

```
>> pwd
```

```
ans=
```

```
C:\MATLAB\bin
```

- nos dice en que directorio nos encontramos

```
>> ls
```

- realiza un listado de los ficheros y los directorios

```
>> mkdir filename
```

- crea un directorio llamado *filename*

```
>> cd filename
```

nos permite cambiar al directorio *filename*.

5.1.2. Operaciones básicas. Funciones elementales.

En las siguientes tablas se relacionan las operaciones más sencillas y que realiza Matlab y sus funciones elementales. Con respecto a otros programas, destacamos la operación de división por la izquierda. En el recuadro siguiente pueden observarse ejemplos de utilización de estas funciones y operaciones. Observese que el argumento de las funciones se escribe entre paréntesis.

+	adición
-	sustracción
*	multiplicación
^	potenciación
\	división izquierda
/	división derecha

exp	log	log10
sin	cos	tan
asin	acos	atan
abs	sqrt	sign
round	floor	ceil

Ejemplos

```
>> 2 + 3                >> 2 * 2
ans =                   ans =
5                        4

>> sin(pi/6)           >> 2/6
ans =                   ans =
0.5000                  0.333

>> log(5^3)            >> 2\6
ans =                   ans =
4.8283                  3

>> 3*(log10(14.7) - 4/6)/atan(2.3)
ans =
1.2940
```

Si queremos información sobre las funciones que están en el recuadro, podemos utilizar el comando **help**. Por ejemplo:

```
>> help atan

atan    Inverse tangent, result in radians.
atan(X) is the arctangent of the elements of X.

See also atan2, tan, atand, atan2d.

Reference page for atan
Other functions named atan
```

También destacamos que las **flechas**: \uparrow y \downarrow permiten recuperar mandatos. Las **flechas**: \leftarrow y \rightarrow permiten movernos a lo largo de una línea de instrucciones y luego corregirla. A la hora de definir variables o cuando utilizamos funciones es importante recordar que *Matlab* distingue entre mayúsculas y minúsculas.

Así por ejemplo si ejecutamos:

```
>> ceil(2.3)
ans =
3
```

Matlab redondea por exceso al entero más cercano. Sin embargo, al ejecutar

```
>> Ceil(2.3)
Undefined function 'Ceil' for input arguments of type 'double'.

Did you mean:
>> ceil(2.3)
```

Matlab nos devuelve un mensaje de error con una sugerencia ya que en este caso encuentra una función muy parecida.

En Matlab podemos asignar variables con determinados nombres a las expresiones numéricas (*números, constantes*). A continuación definimos la masa y la carga del electrón y le asignamos las variables *m* y *q*, respectivamente. Una vez que estas variables están definidas, podemos hacer operaciones con ellas.

```
>> m = 9.11e-31;
>> q = -1.6e-19;
>> r = abs(q)/m
r=
1.7563e+011
>> 3e+8
ans=
300000000
>> m*(ans^2)
ans =
8.1990e-014
```

Además, los nombres de estas variables pueden formarse utilizando letras, dígitos, etc. A su vez, las variables se pueden borrar con el comando `clear nombre`. Si una expresión numérica no le asignamos un nombre, *Matlab* crea por defecto la variable `ans`. El comando `who` nos permite conocer los nombres de las variables asignadas.

5.1.3. Vectores y matrices

Un vector o una matriz en *Matlab* es un conjunto ordenado de números $a_{11}, a_{12}, \dots, a_{nm}$. Los vectores pueden ser vectores filas o columnas como los que están representados en la figura 5.1.

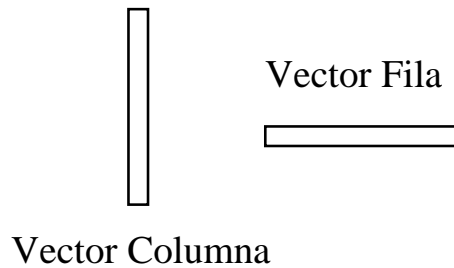


Figura 5.1: Vector columna y vector fila.

Para definir *vectores* o *matrices* utilizamos los **corchetes []**. Los elementos de una fila se separan mediante espacios en blanco o comas y los elementos de una columna se separan por puntos y comas o por nuevas líneas. En el siguiente recuadro se definen algunos vectores filas y columna. Si al final de la definición aparece el signo de punto y coma, los elementos del vector no aparecen en pantalla. (El signo de % se utiliza para escribir comentarios.)

```
>> A=[ 1 2 3 4 5 6 7 8 9];      % vector fila

>> vecf=[1,2,3,4,5,6,7,8,9];  % vector fila

>> B=[ 1
      2
      3
      4 ];                    % vector columna

>> vecc=[1;2;3;4];           % vector columna
```

En ocasiones resulta útil definir los vectores utilizando los dos puntos `:`. De forma general tenemos:

- *1er elemento del vector : incremento : último elemento*

Si omitimos el incremento, Matlab toma por defecto como *incremento = 1*

- *1er elemento del vector : último elemento*

```
>> A=1:2:5
A =
1     3     5

>> B=[5:-1:3] '
B =
5
4
3

>> x = [0:0.1:2*pi]';
>> y = sin(x); c=[x y];
>> M1=[1 2 3;4 5 6;7 8 9]

M1=
1     2     3
4     5     6
7     8     9
```

Como mostramos en el siguiente ejemplo en el recuadro, también se pueden formar vectores o matrices utilizando bloques, siempre que las dimensiones de los bloques sean las apropiadas.

```
>> a = [1 2];
>> b = [3 4];
>> c = [a b 5:1:7 [8 9]]
c =
1     2     3     4     5     6     7     8     9
```


5.1.4. Funciones sobre los vectores

Los argumentos de las funciones en Matlab pueden ser vectores. Si en la variable **v** hemos definido un vector, el $\sin(v)$ será igual a otro vector cuyos elementos se corresponden al seno de cada uno de los elementos del vector **v**.

```
>> v=[0.1:0.1:0.6 1]
v=
0.1000 0.2000 0.3000 0.4000 0.5000 0.6000 1.0000

>> sin(v)
ans =
0.0998 0.1987 0.2955 0.3894 0.4794 0.5646 0.8415

>> length(v)
ans =
7

>> norm(v) % calcula la norma de un vector
ans =
1.3820
```

Hay determinadas funciones que nos dan información sobre los vectores, como por ejemplo:

1. **length(argumento)** calcula el número de elementos de un vector (longitud de un vector). Su argumento es el propio vector.
2. **norm(argumento)** nos permite calcular la normal del vector.
3. **size(argumento)** nos da las dimensiones de la matriz definida como argumento de esta función.

Además instrucción **v(1)** da como resultado el primer elemento de dicho vector, mientras que **M1(2,3)** nos da el elemento de la matriz M1 situado en la fila 2 y en la columna 3.

```
>> v=[0.1:0.1:0.6 1]
v=
0.1000 0.2000 0.3000 0.4000 0.5000 0.6000 1.0000

>> v(1)      % extrae el 1er elemento del vector
ans =
0.1000

>> v([1:3]) %extrae los 3 primeros elementos de v
ans =
0.1000 0.2000 0.3000
```

La notación de dos puntos “:” también puede resultarnos útil no sólo para definir matrices, sino también para extraer elementos, filas o columnas de la misma, como se muestra en los ejemplos del siguiente cuadro.

```
>> A = [1:2:5 ; 1:4:9 ; 1:0.1:1.2]
A =
1.0000    3.0000    5.0000
1.0000    5.0000    9.0000
1.0000    1.1000    1.2000

>> B=A(:,3) % extrae todos los elementos de la 3era columna de A
B =
5.0000
9.0000
1.2000

>> C=A(1:2,:) % extrae todos los elementos de las
% filas 1 y 2 de A
C =
1     3     5
1     5     9

>> D=A(:,[1 3]) % extrae las columnas 1 y 3 de A
D =
1.0000    5.0000
1.0000    9.0000
1.0000    1.2000

>> E=A([1 2],[2 3])
% extrae los elementos de las filas 1 y 2 y
% de las columnas 2 y 3 de A
E =
3     5
5     9
```

Matlab también tiene algunas funciones que definen matrices elementales, como la matriz unidad, o una matriz de ceros. El argumento de estas funciones es simplemente la dimensión de la matriz.

eye(n,m)	matriz identidad $n \times m$
zeros(n,m)	matriz de ceros $n \times m$
ones(n,m)	matriz de unos $n \times m$
rand(n,m)	matriz de números aleatorios $n \times m$

A continuación, exponemos algunos ejemplos de utilización de estas funciones.

```
>> a=eye(2)
a =
1     0
0     1

>> b=zeros(2,5)
b =
0     0     0     0     0
0     0     0     0     0
```

Algunas operaciones entre números, se trasladan simplemente cuando hablamos de operaciones que pueden realizarse con vectores o matrices, siempre que sus dimensiones lo permitan. Así, por ejemplo, dos vectores o dos matrices se pueden sumar o restar. También se pueden multiplicar, dividir, transponer, elevarlos a una potencia. Estas operaciones se pueden realizar de dos formas, o bien sobre el vector o sobre la matriz o bien sobre cada elemento del vector o sobre cada elemento de la matriz. Estas últimas se definen a través de los operadores puntuales.

Operadores Matriciales	Operadores Puntuales
+ suma	+ suma
- resta	- resta
* multiplicación	.* multiplicación
/ división derecha	./ división derecha
\ división izquierda	.\ división izquierda
^ potenciación	.^ potenciación
' conjugada transpuesta	.' transpuesta

A continuación mostramos algunos ejemplos.

```
>> A = [ 1 2; 3 4];
>> B = [ 5 6; 7 8];

>> C = A*B
C =

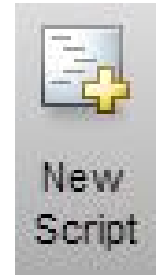
    19    22
    43    50

>> D = A.*B
D =

     5    12
    21    32
>> x = [1 -2 4 2]; y = [3 -5 4 0];
>> 2*y
ans=
     6 -10  8  0
>> sqrt(x(1)^2+x(2)^2+x(3)^2+x(4)^2)
ans=
     5
>> x*y
??? Error using ==> *
Inner matrix dimensions must agree.
>> pe=x*y'
pe=
    29
>> x.*y
ans=
     3    10    16     0
```

5.1.5. Editor de Matlab. Programación.

Para editar un programa nuevo, desde la misma ventana de *Matlab*, pinchar con el ratón **New** y elegir **Script**. Entonces aparece el editor de Matlab donde podemos escribir nuestro programa, que puede ser simplemente un archivo de instrucciones "filename.m" para ser ejecutado desde la ventana de Matlab. En el programa prog.m del siguiente recuadro se guardan los valores de la intensidad (I) y el voltage (V) y luego con el comando plot se representan los valores de I frente a V. Los ficheros deben ser escritos en *ASCII* y la extensión del programa es **.m**



```
I=[0.01;0.02;0.03;0.036;0.032;0.028;0.024;0.018;0.012;0.008];
V=[3.04;6.41;9.84;11.73;10.61;9.02;7.65;5.71;3.79;2.55]; plot(I,V,')
```

Desde la ventana del Matlab, al llamar este programa se representa la gráfica de la intensidad frente al voltaje. El programa debe de ejecutarse desde el directorio donde se encuentre.

```
>> clear
>> help prac
% En el programa prac.m se guardan los valores
% de la intensidad (I) y el voltage (V) y se representan
>> prac          % ejecutamos el programa
```

También podemos seleccionar en **New** la opción **Function**, entonces en el editor de Matlab aparecen los comandos principales para definir una función en Matlab:

```
function [ output_args ] = untitled6( input_args )
%UNTITLED6 Summary of this function goes here
% Detailed explanation goes here

end
```

La primera palabra sin comentar que aparece en estos ficheros debe ser: `function`. El nombre de la función `untitled` debe ser el **mismo** que el nombre del fichero sin extensión `.m`. Los parámetros de entrada (`input_arg`) de la función, deben de ir entre paréntesis `()`. Los parámetros de salida (`output_args`) van delante del nombre de la función. Para definir los nombres de los parámetros de entrada y salida se utilizan variables **auxiliares**, es decir, su nombre puede ser cualquiera. Las variables definidas dentro del fichero son locales, es decir, fuera de la función **no** existen. Una vez creadas, las funciones se utilizan como cualquier función definida en *Matlab*. También se pueden llamar desde cualquier otro programa de *Matlab*.

En el siguiente ejemplo definimos la función `num.m`, cuyo argumento es un número entero n , y que nos da n números aleatorios comprendidos entre el intervalo $[1,49]$.

```
% La función num.m nos da n números (enteros) aleatorios
% comprendidos entre [1,49]. Su argumento debe ser un número entero.
function auxs=num(auxe)
auxs=rand(1,auxe);
auxs=round(48*auxs+1);
```

Si desde Matlab llamamos a la función con argumento 6, el resultado serán 6 números aleatorios comprendidos en el intervalo dado.

```
>> clear
>> help num
>> num(6)           % se evalua la funcion en 6
ans =

37    20    32     9    35     3
```

A continuación relacionamos algunos de los comandos que luego utilizaremos en la implementación de nuestros programas.

- Las sentencias `if`, `elseif`, `else`, `while` nos permiten la elección entre dos o más secuencias. Cuando se verifican ciertas relaciones, se ejecutan una serie de instrucciones, mientras que el resto no se ejecuta. La estructura de la sentencia sería la siguiente

```
if (relaciones1)
    instrucciones1
elseif (relaciones2)
    instrucciones2
else
    instrucciones3
end
```

Junto a estas sentencias se pueden utilizar los siguientes operadores relacionales y lógicos

<	menor que	>	mayor que
<=	menor o igual que	>=	mayor o igual que
==	igual	~=	diferente

&	y		o	~	no
---	---	--	---	---	----

- la sentencia `while` permite ejecutar unas instrucciones mientras se cumplan ciertas relaciones. Su estructura general sería:

```
while (relaciones)
    instrucciones
end
```

- La sentencia `for` se utiliza para repetir las instrucciones un determinado número de veces. Su estructura viene determinada por las líneas

```
for k=1:n
    instrucciones
end
```


5.1.6. Gráficos

En uno de los ejemplos anteriores aparece el comando **plot** que utilizamos para hacer representaciones gráficas en dos dimensiones. Dado un conjunto de n puntos (x_i, y_i) , $i = 1, 2, \dots, n$, definidos previamente en los vectores x e y de *Matlab*; la instrucción **plot(x,y)** nos dibuja los pares de puntos (x_i, y_i) unidos por líneas. Este comando se puede utilizar con algunas opciones para fijar el tipo y color de la línea, etc. Por ejemplo,

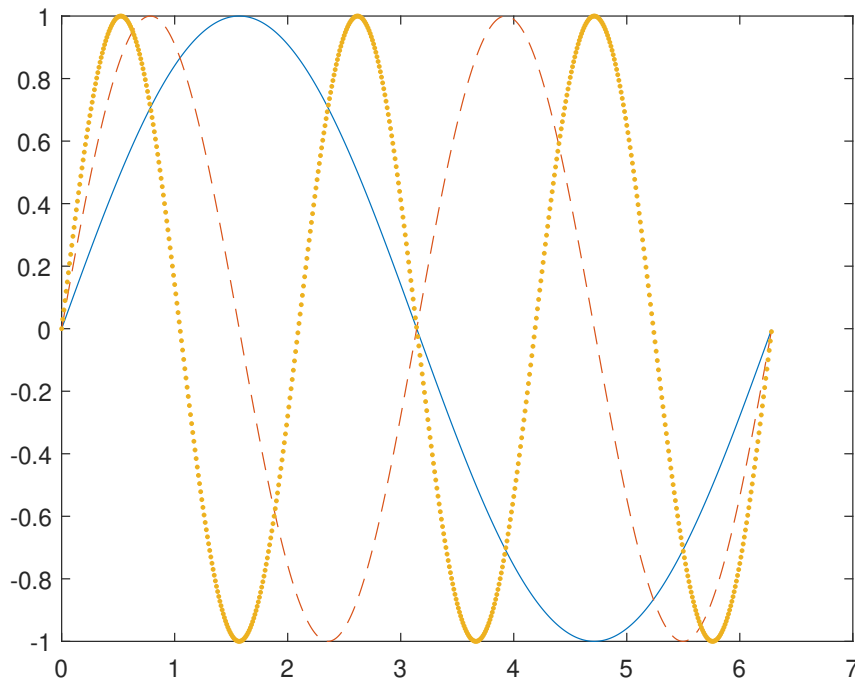
- **plot(x,y,'cts')**, se representan los puntos de los elementos de y frente a x , donde **c** es el color de las líneas, **t** el tipo de línea y **s** el símbolo que usa *Matlab* para dibujar los puntos. En la siguiente tabla recogemos algunos de los colores que podemos usar, los tipos de líneas y los símbolos.

Color	Tipos de líneas	Símbolos
y yellow	. point	- solid
m magenta	o circle	: dotted
c cyan	x x-mark	-. dashdot
r rojo	+ plus	-- dashed
g green	* star	
b blue	s square	
w white	d diamond	
k black	v triangle(down)	
^	triangle(up)	
<	triangle(left)	
>	triangle(right)	
p	pentagram	
h	hexagram	

Con el comando **plot** también podemos realizar gráficos múltiples. En el ejemplo del recuadro se representa $\sin(x)$ con líneas continuas que unen los puntos, $\sin(2x)$ con líneas discontinuas y $\sin(3x)$ con puntos.

```
>> x=0:.01:2*pi;
>> y1=sin(x);y2=sin(2*x);y3=sin(3*x);
>> plot(x,y1,x,y2,'--',x,y3,'.')
```

Que nos generaría la siguiente gráfica:



Algunas sentencias de utilidad para utilizar en combinación con plot son las siguientes:

- las instrucciones **hold on** (congela la pantalla gráfica), **hold off** (desactiva el hold on) y el **clf** (borra la pantalla gráfica).
- `axis([xmin,xmax,ymin,ymax])` escala la ventana gráfica.
- `grid` dibuja una retícula cuadrada.
- `xlabel(' nombre_del_eje_x')`, `ylabel(' nombre_eje_y')`, `title(' título')`.
- `gtext(' texto')` se añade un texto al gráfico de forma interactiva. Se escribe `gtext(' texto')` y se marca con el ratón en la gráfica donde se desee insertar el *texto*.
- `text(' x,y,texto')` se inserta el texto deseado al gráfico en la posición *x*, *y*
- `subplot(' n,m,k')` se utiliza (junto con el `plot`) para incorporar *n* por *m* gráficas en una misma figura.

5.1.7. Grabar y leer datos en ficheros. Impresión de las gráficas

Para trabajar con nuestras propias bases de datos de variables:

- La instrucción `save('fname.dat','x z')` graba las variables `x` y `z` en un fichero (binario) de nombre `fname.dat` (*archivos de datos*). Si queremos grabar los datos en un archivo `ascii`, debemos utilizar `save('fname.dat','x z','-ascii')`
- La instrucción `load fname.dat x` recupera la variable `x` guardada en el archivo (binario) `fname.dat`. Observe que en `fname.dat` podría haber almacenadas otras variables aparte de la variable `x`. Si queremos recuperar un archivo `ascii`, utilizamos `load('fname.dat','-ascii')`

Ejemplo

```
>> x = [0:pi/60:2*pi]'; y = sin(x.^2);
>> save('datos.dat','x','-ascii')
>> clear
>> who
>> load datos.dat
>> who
>> x
>> z = [x y];
>> save('datos.dat','z','-ascii')
>> clear
>> who
>> load('datos.dat','-ascii')
>> who
>> plot(datos(:,1),datos(:,2))
```

- Es recomendable también guardar las variables en formato `.txt` en lugar de `.dat`, puesto que luego se pueden importar a otros programas y obtener tablas de cálculos o realizar simulaciones, como vimos en el capítulo anterior.
- Con el comando `diary('filename')` grabamos la sesión de *Matlab*, es decir todas las entradas y las salidas que tengan lugar en la ventana de *Matlab* después de haber escrito este comando. Para dejar de grabar en este archivo es necesario ejecutar la instrucción `diary off`.

- Para imprimir la figura en un archivo postscript utilizamos el comando `print` (`'nfile.ps', '-dps'`). Por ejemplo, **`print ('nfile.ps', '-dps')`** crea el archivo postscript, `fig.ps`, de la figura que este en la ventana gráfica de *Matlab*.

En este capítulo se recopilan los códigos numéricos escritos en Matlab, que se han utilizado para resolver los modelos físicos planteados en la memoria.

5.2. MAS

En esta sección mostramos cómo el Matlab resuelve la ecuación diferencial de segundo orden representada por (2.8). El primer paso, es transformar (2.8) en el siguiente sistema de dos ecuaciones diferenciales de primer orden:

$$\begin{pmatrix} x' \\ v' \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ -\omega_0^2 & 0 \end{pmatrix} \begin{pmatrix} x \\ v \end{pmatrix} \quad (5.1)$$

Este sistema está definido en el programa osc.m y es utilizado en el programa que obtiene su solución numérica mas.m.

```
% PROGRAMA osc.m
function xprime=osc(t,y)
% x es un vector de dos componentes: la posicion y la velocidad
%parametro, frecuencia propia del sistema
w0=2;
% matriz asociada al sistema de dos ecuaciones diferenciales,
% que determinan la posicion y la velocidad de una particula
% d=[0 1;-w0^2 0]; xprime=d*x
xprime=[y(2);-(w0^2)*y(1)];

% RESOLUCION DEL SISTEMA DE ECUACIONES
% Ejecucion del programa desde octave utilizando lsode
% desde octave: t= linspace(0, 2, 100)';
% se define un vector que comienza en 0, termina en 2; y
%tiene 100 elementos equidistantes.
% desde octave: x0=[4;0] esta es la condicion inicial,
% 4 es la posicion inicial y 0 es la velocidad inicial.
% desde octave: y = lsode ("osc",x0, t);

% NOTAS sobre la ejecución del programa desde Matlab
% en Matlab nos funciona poniendo osc(t,x) en la
% primera linea. en Matlab: ejecutar el comando
% [t,y]=ode45(@osc,[0 2],[4;0]);
```

```
% REPRESENTACION GRAFICA
% en Matlab o en octave: ejecutar plot(t,y(:,1))
% dibuja la posicion en funcion del tiempo
% en Matlab o en octave: ejecutar plot(t,y(:,2))
% dibuja la velocidad en funcion del tiempo
```

Al ejecutar el programa `mas.m`, observamos la representación de $x(t)$ y de $v(t)$. En este programa las condiciones iniciales son iguales a $x(0) = 4$ y $v(0) = 0$, como el ejemplo que se discute en la sección 2.2.

```
% PROGRAMA mas.m
% RESOLUCION DEL SISTEMA DE ECUACIONES
% Ejecucion del programa desde octave utilizando lsode
% desde octave: t= linspace(0, 2, 100)'; se define un
% vector que comienza en 0, termina en 2; y tiene 100
% elementos equidistantes.
% desde octave: x0=[4;0] esta es la condicion inicial,
% 4 es la posicion inicial y 0 es la velocidad inicial.
% desde octave: y = lsode ("osc",x0, t);
t= linspace(0, 2, 100)';
x0=[4;0];
y = lsode ("osc",x0, t);

% NOTAS sobre la ejecución del programa desde Matlab
% en Matlab nos funciona poniendo osc(t,x) en la primera linea.
% en Matlab: ejecutar el comando [t,y]=ode45(@osc,[0 2],[4;0]);

% REPRESENTACION GRAFICA
% en Matlab o en octave: ejecutar plot(t,y(:,1)) dibuja
% la posicion en funcion del tiempo
% en Matlab o en octave: ejecutar plot(t,y(:,2)) dibuja
% la velocidad en funcion del tiempo
plot(t,y(:,1),t,y(:,2),'*')
```

A continuación presentamos el programa `mainplot.m`, donde se compara la solución numérica con la solución exacta (2.13)-(2.15) y se añaden las leyendas a la figura. Este programa también utiliza el sistema de ecuaciones definido en `osc.m`.

```

% PROGRAMA mainplot.m
% Solucion numerica del oscilador armonico sin disipacion
% Parametros
clear
clf
% ti: tiempo inicial; tfinal: tiempo final;
% N: numero de puntos entre ti y tfinal
ti=0;
tfinal=20;
N=1000;
% x0 posicion inicial y v0 velocidad inicial
x0=4;
v0=0;
% w0: frecuencia propia del sistema
w0=2;
%global w0
%t= linspace(ti, tfinal, N);
xi=[x0, v0];
% en octave aparece lode en lugar de ode45
[t,y] = ode45(@osc,[ti,tfinal],xi);
tr=t';
% construccion de la solucion exacta
% determinacion de las constantes a partir de las
% condiciones iniciales
if (v0==0)
phi=pi/2;
A=-x0/sin(phi);
else
phi=-atan(w0*x0/v0);
A=v0/(w0*cos(phi));
% en octave aparece endif
end
xe=A*sin(w0*tr-phi); %posicion exacta
ve=A*w0*cos(w0*tr-phi); %velocidad exacta

x1=[-4,4];
y1=zeros(size(x1));

```

```

y2=[-8,8];
x2=zeros(size(y2));

% representacion grafica
% y(:,1) posicion numerica
% y(:,2) velocidad numerica
figure(1)
p1=plot(tr,y(:,1),'kx',tr,xe,'r',tr,ve,'b');
hold on
p2=plot(tr,y(:,2),'m.');
```



```

lx=xlabel('$t$', 'Interpreter', 'LaTeX');
ly=ylabel('$x(t); v(t)$', 'Interpreter', 'LaTeX');
h=get(gcf, 'currentaxes');
set(h, 'fontsize', 20, 'linewidth', 2);
set(p2, 'MarkerSize', 15)
set(p1, 'linewidth', 2)
set([lx ly], 'linewidth', 2);
set([lx ly], 'fontsize', 20)
format longG
datos=[tr', xe'];
save('prueba.txt', 'datos', '-ascii')
```



```

figure(2)
p3=plot(y(:,1),y(:,2));

hold on
[X,Y]=meshgrid(-4:1:4,-8:1:8);
U=Y;V=-w0*w0*X;
p7=quiver(X,Y,U,V);
axis([-4 4 -8 8])

p5=plot(x1,y1,'b--',x2,y2,'b--');
plot(0,0,'-o','MarkerSize',6,'MarkerEdgeColor','k','MarkerFaceColor','k');
lx=xlabel('$x(t)$', 'Interpreter', 'LaTeX');
```



```

ly=ylabel('$v(t)$','Interpreter','LaTeX');
h=get(gcf,'currentaxes');
set(p3,'MarkerSize',15)
set(p3,'linewidth',2)
set([lx ly],'linewidth',2);
set([lx ly],'fontsize',20)
format longG

figure(3)
p9=plot3(y(:,1),y(:,2),tr,'b');
hold on
plot3(y(:,1),y(:,2),1+0*tr,'r'); %proyeccion en XY
plot3(y(:,1),10+0*y(:,2),tr,'r'); %proyeccion en XZ
plot3(4+0*y(:,1),y(:,2),tr,'r'); %proyeccion en ZY
lx=xlabel('$x(t)$','Interpreter','LaTeX');
ly=ylabel('$v(t)$','Interpreter','LaTeX');
lz=zlabel('$t$', 'Interpreter','LaTeX');

```

5.3. Vibraciones amortiguadas subcríticas

En esta sección describimos la resolución numérica de la ecuación diferencial (2.16), que es equivalente al sistema:

$$\begin{pmatrix} x' \\ v' \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ -\omega_0^2 & -d \end{pmatrix} \begin{pmatrix} x \\ v \end{pmatrix}, \quad (5.2)$$

donde se considera el caso subcrítico, $d < \omega_0$. Este sistema está definido en el programa `oscv1.m`, utilizado posteriormente en el programa que obtiene su solución numérica `mainplotv2.m` y que también la compara con su solución exacta.

```

% Programa oscv1.m
function xprime=oscv1(t,y)
%parameters
global w0
global b
C=[0 1;-w0^2 -b];
y=[y(1);y(2)];

```

```
xprime=C*y;
```

```
% Programa mainplotv2.m
% Solucion numerica del oscilador armonico AMORTIGUADO
% Parametros
clear
clf
% ti y tfinal tiempos inicial y final de la integracion
% N: numero de puntos entre ti y tfinal
ti=0;
tfinal=10;
tr1=linspace(ti,tfinal,30);
N=100;
% x0 posicion inicial
% v0 velocidad inicial
x0=3;
v0=0;
% w0 frecuencia propia del sistema
% b coeficiente de friccion
global w0
global b
w0=2;
b=1;
d=b/2;
% resolucion numerica de la ecuacion diferencial
% definida en ocv1.m
xi=[x0,v0];
[t,y] = ode45(@oscv1,[ti,tfinal],xi);
tr=t';
% construccion de la solucion exacta
wd=sqrt(w0^2-d^2);
if (v0==0)
if (d==0)
phi=pi/2;
else
```

```

phi=-atan(wd/d);
end
A=-x0/sin(phi);
else
phi=-atan(wd*x0/(v0+d*x0));
A=v0/(wd*cos(phi)+d*sin(phi));
end

xe=A*sin(wd*tr-phi).*exp(-d*tr); %posicion exacta amortiguada
ve=A*wd*cos(wd*tr-phi).*exp(-d*tr)+A*(-d)*sin(wd*tr-phi).*exp(-d*tr);
pru=A*exp(-d*tr);
%velocidad exacta amortiguada

x1=[-1.5,3];
y1=zeros(size(x1));

y2=[-5,2];
x2=zeros(size(y2));

% representacion grafica
figure(1)
p1=plot(tr,y(:,1),'kx',tr,xe,'r',tr,ve,'b');
hold on
p2=plot(tr,y(:,2),'m. ');
hold on
p4=plot(tr, pru,'--', tr, -pru,'--');

lx=xlabel('$t$', 'Interpreter', 'LaTeX');
ly=ylabel('$x(t) ; v(t) $', 'Interpreter', 'LaTeX');
h=get(gcf, 'currentaxes');
set(gca, 'XTick', [0:2.5:10]);
set(h, 'fontsize', 20, 'linewidth', 2);
set(p1, 'linewidth', 2)
set(p2, 'MarkerSize', 15)
set([lx ly], 'linewidth', 2);
set([lx ly], 'fontsize', 20)
datos=[tr', xe'];

```

```

save('fig1p4.txt','datos','-ascii')

figure(2)
p3=plot(y(:,1),y(:,2));
hold on
[X,Y]=meshgrid(-1.5:0.6:3,-5:0.6:2);
U=Y;V=-w0*w0*X-b*Y;
p7=quiver(X,Y,U,V);
axis([-1.5 3 -5 2])
p5=plot(x1,y1,'b--',x2,y2,'b--');
plot(0,0,'-o','MarkerSize',6,'MarkerEdgeColor','k','MarkerFaceColor','k');
lx=xlabel('$x(t)$','Interpreter','LaTeX');
ly=ylabel('$v(t)$','Interpreter','LaTeX');
h=get(gcf,'currentaxes');
set(p3,'MarkerSize',15)
set(p3,'linewidth',2)
set([lx ly],'linewidth',2);
set([lx ly],'fontsize',20)
format longG

figure(3)
p9=plot3(y(:,1),y(:,2),tr,'b');
hold on
plot3(y(:,1),y(:,2),1+0*tr,'r'); %proyeccion en XY
plot3(y(:,1),10+0*y(:,2),tr,'r'); %proyeccion en XZ
plot3(4+0*y(:,1),y(:,2),tr,'r'); %proyeccion en ZY
lx=xlabel('$x(t)$','Interpreter','LaTeX');
ly=ylabel('$v(t)$','Interpreter','LaTeX');
lz=zlabel('$t$', 'Interpreter','LaTeX');

%print -deps xvamortiguado.eps

```

Al ejecutar el programa mainplotv2.m, comparamos la solución numérica de $x(t)$ y $v(t)$ con la solución exacta (ver la figura 2.4).

5.4. Vibraciones amortiguadas subcríticas y forzadas

En esta sección describimos la resolución numérica de la ecuación diferencial (2.16), que es equivalente al sistema:

$$\begin{pmatrix} x' \\ v' \\ \zeta' \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 \\ -\omega_0^2 & -d & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ v \\ \omega \end{pmatrix} + \begin{pmatrix} 0 \\ f(\zeta) \\ 0 \end{pmatrix}, \quad (5.3)$$

donde $f(\zeta) = a_0 \sin(\zeta) + b_0 \cos(\zeta)$. Los parámetros a_0 , b_0 y ω representan las amplitudes y la frecuencia de la fuerza externa periódica $f(\zeta)$, respectivamente. Este sistema está definido en el programa `oscv3.m` y luego se utiliza en el programa que obtiene su solución numérica `mainplotv3.m` y que también la compara con su solución exacta.

```
% PROGRAMA oscv3.m
function xprime=oscv3(t,x)
%parameters
global w0
global b
global w
global a0
global b0
d=[0 1;-w0^2 -b];
xprime=d*x+[0; a0*sin(w*t)+b0*cos(w*t)];

% Programa mainplotv3.m
% Solucion numerica del oscilador armonico AMORTIGUADO y FORZADO
% Parametros
clear
clf
% ti: tiempo final; tfinal: tiempo final;
ti=0;
tfinal=100;
```

```

% N: numero de puntos entre ti y tfinal;
N=200;
% parametros de la ecuacion
% w0 frecuencia propia
% b disipacion
% w frecuencia externa
% a0 y b0: amplitudes de la fuerza externa
global w0
global b
global w
global a0
global b0
w0=2;
b=0.1;
d=b/2;
a0=0.3;
b0=0;
w=1.9;
zeta=10;
%%%%%
w1=w0^2-w^2;d1=2*d*w;
Ap=(w1*a0+d1*b0)/(w1^2+d1^2);
Bp=(w1*b0-d1*a0)/(w1^2+d1^2);
wd=sqrt(w0^2-d^2);
% condiciones iniciales para la posicion y la velocidad
x0=3;
v0=0;
%v0=w*Ap
%x0=Bp

xi=[x0,v0];
[t,y] = ode45(@oscv3,[ti,tfinal],xi);
tr=t';

% solucion exacta
if ((v0-w*Ap)==0)
if (d==0)

```

```

phi=pi/2;
else
phi=-atan(wd/d);
end
A=-(x0-Bp)/sin(phi);
else
phi=-atan(wd*(x0-Bp)/(v0-w*Ap+d*(x0-Bp)));
A=(v0-w*Ap)/(wd*cos(phi)+d*sin(phi));
end
xe=A*exp(-d*tr).*sin(wd*tr-phi)+Ap*sin(w*tr)+Bp*cos(w*tr); %posicion exacta
ve=-d*A*exp(-d*tr).*sin(wd*tr-phi)+wd*A*exp(-d*tr).*cos(wd*tr-phi)+...
w*(Ap*cos(w*tr)-Bp*sin(w*tr)); %velocidad exacta

x1=[-3,3];
y1=zeros(size(x1));

y2=[-6,6];
x2=zeros(size(y2));

% representacion grafica
% y(:,1) posicion numerica
% y(:,2) velocidad numerica

figure(1)
p1=plot(tr,y(:,1),'k+',tr,xe,'r',tr,ve,'b');
hold on
p2=plot(tr,y(:,2),'m.');
```

pause

```

axis([0 20 -6 6])

lx=xlabel('$t$', 'Interpreter', 'LaTeX');
ly=ylabel('$x(t) ; v(t) $', 'Interpreter', 'LaTeX');
h=get(gcf, 'currentaxes');
set(h, 'fontsize', 20, 'linewidth', 2);
set(p1, 'linewidth', 2);
set(p2, 'MarkerSize', 15);
set([lx ly], 'linewidth', 2);
```

```

set ([lx ly], 'fontsize', 20)
datos=[tr', xe'];
save('fig1p5a.txt','datos','-ascii')

figure(2)
p3=plot(y(:,1),y(:,2));
hold on
[X,Y]=meshgrid(-3:1:3,-6:1:6);
U=Y;V=-w0*w0*X-b*Y+ones(size(X))*(a0*sin(zeta)+b0*cos(zeta));
p7=quiver(X,Y,U,V);
axis([-3 3 -6 6])
p5=plot(x1,y1,'b--',x2,y2,'b--');
lx=xlabel('$x(t)$','Interpreter','LaTeX');
ly=ylabel('$v(t)$','Interpreter','LaTeX');
h=get(gcf, 'currentaxes');
set(p3,'MarkerSize',15)
set(p3,'linewidth',2)
set([lx ly],'linewidth',2);
set ([lx ly], 'fontsize', 20)
format longG

figure(3)
p9=plot3(y(:,1),y(:,2),tr,'b');
hold on
plot3(y(:,1),y(:,2),1+0*tr,'r');    %proyeccion en XY
plot3(y(:,1),10+0*y(:,2),tr,'r');    %proyeccion en XZ
plot3(4+0*y(:,1),y(:,2),tr,'r');    %proyeccion en ZY
lx=xlabel('$x(t)$','Interpreter','LaTeX');
ly=ylabel('$v(t)$','Interpreter','LaTeX');
lz=zlabel('$t$', 'Interpreter','LaTeX');

print -deps xvforzado1.eps

```

Al ejecutar el programa mainplotv3.m, comparamos la solución numérica de $x(t)$ y $v(t)$ con la solución exacta (ver la figura 2.5).

5.5. Fricción estática y dinámica

En esta sección describimos la resolución numérica de la ecuación diferencial (3.32), que es equivalente al sistema:

$$\begin{pmatrix} x' \\ v' \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ -\omega_0^2 & 0 \end{pmatrix} \begin{pmatrix} x \\ v \end{pmatrix} + \begin{pmatrix} 0 \\ -\alpha \end{pmatrix} \text{sign}(v), \quad (5.4)$$

Los parámetros α y ω_0 representan el producto de g y μ_k y la frecuencia de oscilación natural del sistema.

```
%Oscilador armonico amortiguado por rozamiento, para v0=0;
% Programa oscrozv0nula.m
clear
clf
ti=0;
tfinal=10;
tfinal1=10;
N=300;
global w0
global alpha
w0=2;
alpha=5.88; %mu_d=alpha/g=0.6
C=1.2; % mu_s/mu_d mu_s=0.72
x0=15;
v0=0;

t= linspace(ti, tfinal1, N);
%t2= linspace(ti, tfinal, N);
xi=[x0 v0];
[t2,y] = ode15s(@osc4,[ti tfinal],xi);

%Xlim indica el valor minimo de x0 para el cual la particula va a oscilar.
%Si es inferior a xlim, las graficas seran nulas por la propia naturaleza
% de las ecuaciones.
xlim=(alpha*C/w0^2);

%Limsup es el ciclo limite para el cual la particula no posee energia
% suficiente para vencer el rozamiento.
```

%A partir de dicho valor la partícula se estanca.

```
limsup=(0.5)*((w0^2)*x0/alpha-C);
```

%Para trabajar con los ciclos, tenemos un vector con tantos elementos como
% tiene el vector t

%El vector p será el que nos indique en que ciclo nos encontramos.

```
n=1;
```

```
g=0;
```

```
lim1=(g+1)*pi/w0;
```

```
q=0;
```

%Primero está vacío.

```
p=zeros(1,N);
```

%q será el vector que nos indique si la partícula continúa en movimiento.

```
q=zeros(1,N);
```

%Luego este bucle reescribe p para darles valores según el tramo al que
% pertenece.

%También reescribe a q para darle valores de 1 (Si sigue en movimiento)

% o 0 (Si se queda estancada).

```
while (n<=N)
```

```
    lim=(g+1)*pi/w0;
```

```
    if (t(n)>lim)
```

```
        g=g+1;
```

```
    end
```

```
    if(n<limsup)
```

```
        q(n)=1;
```

```
    end
```

```
        p(n)=g;
```

```
        if(p(n)<limsup)
```

```

    %Mientras este dentro del limite para el cual moverse, q toma el
    %valor de 1.
    q(n)=1;

    tlimite=t(n);
    xestanq=((x0-(2*p(n)+1)*(alpha/(w0^2)))*cos(w0*tlimite)+...
        (alpha/(w0^2))*(-1)^p(n));
    %tlimite nos indicara en que momento t exactamente se quedara
    % parada la particula, y xestanq en que posicion lo hace
    end
    n=n+1;
end

tr=t';
D=alpha/(w0^2);
tq=q';

%Aqui vemos la influencia de p (El numero del tramo en el que nos
% encontramos.
%Y todo multiplicado por q. Es decir, cuando p sea mayor que el limite,
%xf pasa a valer xestanq y vf pasa a valer 0.
xf=((x0-(2*p+1)*D).*cos(w0*t)+D*(-1).^p).*q -xestanq*(q-1);
vf=(-w0*(x0-(2*p+1)*D).*sin(w0*t)).*q;

% representacion grafica
% y(:,1) posicion numerica
% y(:,2) velocidad numerica
figure(1)
p1=plot(tr,xf,'r',tr,vf,'b');
hold on
p2=plot(t2',y(:,1),'kx',t2',y(:,2),'m. ');

%t1=title("Oscilador armonico simple");
%ly=ylabel("Posicion (+) / Velocidad (*) ");
%ly=ylabel('$q(t)$', 'Interpreter', 'LaTeX');

```

```

% en octave aparecen las 2 lineas anteriores,
% en matlab se puede cambiar por
lx=xlabel('$t$', 'Interpreter', 'LaTeX');
ly=ylabel('$x(t) ; v(t) $', 'Interpreter', 'LaTeX');
h=get(gcf, 'currentaxes');
set(gca, 'XTick', [0:2.5:20]);
set(h, 'fontsize', 20, 'linewidth', 2);
set(p1, 'linewidth', 1)
set([lx ly], 'linewidth', 2);
set([lx ly], 'fontsize', 20)
datos=[tr, xf'];
save('fig2p1.txt', 'datos', '-ascii')

```

```

figure(2)
p3=plot(xf, vf);
hold on
[X, Y]=meshgrid(-15:3:15, -30:3:30);
U=Y; V=-w0*w0*X-alpha*sign(Y);
p7=quiver(X, Y, U, V);
axis([-15 15 -30 30])

```

```

lx=xlabel('$x(t)$', 'Interpreter', 'LaTeX');
ly=ylabel('$v(t)$', 'Interpreter', 'LaTeX');
h=get(gcf, 'currentaxes');
set(p3, 'MarkerSize', 15)
set(p3, 'linewidth', 2)
set([lx ly], 'linewidth', 2);
set([lx ly], 'fontsize', 20)
format longG

```

```

figure(3)
p9=plot3(xf, vf, tr, 'b');
hold on
plot3(xf, vf, 1+0*tr, 'r'); %proyeccion en XY
plot3(xf, 20+0*vf, tr, 'r'); %proyeccion en XZ
plot3(20+0*xf, vf, tr, 'r'); %proyeccion en ZY
lx=xlabel('$x(t)$', 'Interpreter', 'LaTeX');

```

```
ly=ylabel('$v(t)$', 'Interpreter', 'LaTeX');
```

```
lz=zlabel('$t$', 'Interpreter', 'LaTeX');
```

```
%p1=plot(tr,xf,tr,vf,tr,p,tr,q);
```


Capítulo 6

Apéndice: Programas en Mathematica.

En este apéndice incluimos los programas de Mathematica `osciladormortiguado.nb` y `osciladorforzado1.nb`. Se han utilizado para demostrar que a partir de la ley de conservación de la energía se derivan las ecuaciones que representan las curvas en el retrato de fase del oscilador armónico amortiguado y del oscilador armónico amortiguado y forzado (con condiciones iniciales excepcionales).

Quit

(*Definimos la posición y la velocidad del oscilador armónico amortiguado*)

$$x[t_] = A * \text{Exp}[-\beta * t] * \text{Sin}[\omega d * t - \phi]$$

$$v[t_] = D[x[t], t]$$

$$-A e^{-t\beta} \text{Sin}[\phi - t\omega d]$$

$$A e^{-t\beta} \omega d \text{Cos}[\phi - t\omega d] + A e^{-t\beta} \beta \text{Sin}[\phi - t\omega d]$$

(*El trabajo de la fuerza disipativa esta relacionado con la integral de $[v(\tau)]^2$. Calculo de dicha integral*)

$$\text{Inte}[t_] = \text{Integrate}[\text{Expand}[v[t]^2], t]$$

$$\frac{1}{4\beta} A^2 e^{-2t\beta} (-\beta^2 - \omega d^2 + \beta^2 \text{Cos}[2(\phi - t\omega d)] - \beta \omega d \text{Sin}[2(\phi - t\omega d)])$$

(*Integral evaluada entre 0 y t, y luego multiplicada por $(-4*\beta)$ *)

$$\text{parte1} = \text{Simplify}[-4 * \beta * (\text{Inte}[t] - \text{Inte}[0])]$$

$$\text{parte2} = \text{Simplify}[\text{parte1} + 2 * \beta * x[t] * v[t]]$$

$$A^2 e^{-2t\beta} (\beta^2 - e^{2t\beta} \beta^2 + \omega d^2 - e^{2t\beta} \omega d^2 + e^{2t\beta} \beta^2 \text{Cos}[2\phi] - \beta^2 \text{Cos}[2(\phi - t\omega d)] - e^{2t\beta} \beta \omega d \text{Sin}[2\phi] + \beta \omega d \text{Sin}[2(\phi - t\omega d)])$$

$$A^2 (-\beta^2 - \omega d^2 + e^{-2t\beta} \omega d^2 + \beta^2 \text{Cos}[2\phi] - \beta \omega d \text{Sin}[2\phi])$$

$$\omega_0 = \text{Sqrt}[\omega d^2 + \beta^2]; (*Relación entre ω_0 y ωd *)$$

$$\text{cte} = v_0^2 + \omega_0^2 * x_0^2; (*Definición de los términos$$

constantes que aparecen en la Ley de conservación de la energía*)

$$\text{parte3} = \text{Simplify}[\text{parte2} + \text{cte}]$$

$$v_0^2 + x_0^2 (\beta^2 + \omega d^2) + A^2 (-\beta^2 - \omega d^2 + e^{-2t\beta} \omega d^2 + \beta^2 \text{Cos}[2\phi] - \beta \omega d \text{Sin}[2\phi])$$

$$\text{parte4} = \text{Simplify}[\text{parte3} - A^2 * e^{-2t\beta} \omega d^2]$$

$$v_0^2 - A^2 \beta^2 + x_0^2 \beta^2 - A^2 \omega d^2 + x_0^2 \omega d^2 + A^2 \beta^2 \text{Cos}[2\phi] - A^2 \beta \omega d \text{Sin}[2\phi]$$

(*Definimos las condiciones iniciales

y demostramos que la parte4 es igual a cero*)

$$x_0 = x[0];$$

$$v_0 = v[0];$$

$$\text{FullSimplify}[\text{parte4}]$$

0

(*Ley de conservación de la energía, eq=0*)

$$\text{eq}[x1_, v1_] = \text{Simplify}[$$

$$-v1^2 + \text{cte} - \omega_0^2 * x1^2 - 2 * \beta * x1 * v1 - 4 * \beta * (\text{Inte}[t] - \text{Inte}[0]) + 2 * \beta * x[t] * v[t]]$$

$$-v1^2 - 2 v1 x1 \beta + A^2 e^{-2t\beta} \omega d^2 - x1^2 (\beta^2 + \omega d^2)$$

$$X = x1 * \omega d; Y = v1 + x1 * \beta;$$

$$\rho = \text{Sqrt}[X^2 + Y^2]$$

$$\text{FullSimplify}[\text{eq}[X/\omega d, Y - X * \beta/\omega d]]$$

$$\sqrt{(v1 + x1 \beta)^2 + x1^2 \omega d^2}$$

$$- (v1 + x1 \beta)^2 + A^2 e^{-2 t \beta} \omega d^2 - x1^2 \omega d^2$$

(*Esta es la ecuación de la espiral logarítmica*)

Quit

(*Definimos la posición, la velocidad del oscilador armónico amortiguado. Definimos la fuerza externa f(t)*)

$$Ap = ((\omega_0^2 - \omega^2) * a_0 + 2 * \beta * \omega * b_0) / ((\omega_0^2 - \omega^2)^2 + (2 * \beta * \omega)^2);$$

$$Bp = ((\omega_0^2 - \omega^2) * b_0 - 2 * \beta * \omega * a_0) / ((\omega_0^2 - \omega^2)^2 + (2 * \beta * \omega)^2);$$

$$At = \text{Sqrt}[Ap^2 + Bp^2];$$

$$\text{Cos}\delta = Ap / At;$$

$$\text{Sin}\delta = Bp / At;$$

$$x[t_] = A * \text{Exp}[-\beta * t] * \text{Sin}[\omega d * t - \phi] + At * \text{Sin}[\omega * t + \delta]$$

$$v[t_] = D[x[t], t]$$

$$f[t_, a_0_, b_0_] = a_0 * \text{Sin}[\omega * t] + b_0 * \text{Cos}[\omega * t]$$

$$\sqrt{\left((2 b_0 \beta \omega + a_0 (-\omega^2 + \omega_0^2))^2 / (4 \beta^2 \omega^2 + (-\omega^2 + \omega_0^2)^2) + (-2 a_0 \beta \omega + b_0 (-\omega^2 + \omega_0^2))^2 / (4 \beta^2 \omega^2 + (-\omega^2 + \omega_0^2)^2) \right)^2} \text{Sin}[\delta + t \omega] - A e^{-t \beta} \text{Sin}[\phi - t \omega d]$$

$$\omega \sqrt{\left((2 b_0 \beta \omega + a_0 (-\omega^2 + \omega_0^2))^2 / (4 \beta^2 \omega^2 + (-\omega^2 + \omega_0^2)^2) + (-2 a_0 \beta \omega + b_0 (-\omega^2 + \omega_0^2))^2 / (4 \beta^2 \omega^2 + (-\omega^2 + \omega_0^2)^2) \right)^2} \text{Cos}[\delta + t \omega] + A e^{-t \beta} \omega d \text{Cos}[\phi - t \omega d] + A e^{-t \beta} \beta \text{Sin}[\phi - t \omega d]$$

$$b_0 \text{Cos}[t \omega] + a_0 \text{Sin}[t \omega]$$

A = 0; (*Condiciones iniciales excepcionales*)

$$x_0 = x[0];$$

$$v_0 = v[0];$$

(*El trabajo de la fuerza disipativa esta relacionado con la integral de [v(t)]^2. Calculo de dicha integral*)

$$\text{Inte}[t_] = \text{Simplify}[\text{Integrate}[\text{Expand}[v[t]^2], t]]$$

$$\left((a_0^2 + b_0^2) \omega (2 (\delta + t \omega) + \text{Sin}[2 (\delta + t \omega)]) \right) / \left(4 (4 \beta^2 \omega^2 + (\omega^2 - \omega_0^2)^2) \right)$$

(*Integral evaluada entre 0 y t*)

$$\text{parte1} = \text{Simplify}[-4 * \beta * (\text{Inte}[t] - \text{Inte}[0])]$$

$$- \left((a_0^2 + b_0^2) \beta \omega (2 t \omega - \text{Sin}[2 \delta] + \text{Sin}[2 (\delta + t \omega)]) \right) / \left(4 \beta^2 \omega^2 + (\omega^2 - \omega_0^2)^2 \right)$$

(*Calcular la integral relacionada con la fuerza externa si b0=0*)

$$\text{Inte1}[t_] = \text{Integrate}[f[t, a_0, 0] * v[t], t];$$

$$\text{parte2} = 2 * (\text{Inte1}[t] - \text{Inte1}[0])$$

$$2 \left(\frac{1}{4} a_0 \sqrt{\left((a_0^2 + b_0^2) / (4 \beta^2 \omega^2 + (\omega^2 - \omega_0^2)^2) \right)} \text{Cos}[\delta] - \frac{1}{4} a_0 \sqrt{\left((a_0^2 + b_0^2) / (4 \beta^2 \omega^2 + (\omega^2 - \omega_0^2)^2) \right)} (\text{Cos}[\delta + 2 t \omega] + 2 t \omega \text{Sin}[\delta]) \right)$$

Simplify[parte1 + parte2]

$$\frac{1}{2} a_0 \sqrt{\left((a_0^2 + b_0^2) / \left(4 \beta^2 \omega^2 + (\omega^2 - \omega_0^2)^2 \right) \right)} \cos[\delta] -$$

$$\frac{1}{2} a_0 \sqrt{\left((a_0^2 + b_0^2) / \left(4 \beta^2 \omega^2 + (\omega^2 - \omega_0^2)^2 \right) \right)} (\cos[\delta + 2 t \omega] + 2 t \omega \sin[\delta]) -$$

$$\left((a_0^2 + b_0^2) \beta \omega (2 t \omega - \sin[2 \delta] + \sin[2 (\delta + t \omega)]) \right) / \left(4 \beta^2 \omega^2 + (\omega^2 - \omega_0^2)^2 \right)$$

(*Definimos los términos constantes que aparecen en la ley de conservación de la energía*)

b0 = 0;

cte = v0^2 + ω0^2 * x0^2;

parte3 = Simplify[parte1 + parte2 + cte]

c = Coefficient[Expand[parte3], t]

$$\left(a_0^2 \left(\frac{a_0 \cos[\delta]}{\sqrt{\frac{a_0^2}{4 \beta^2 \omega^2 + (\omega^2 - \omega_0^2)^2}}} + 2 \omega^2 \cos[\delta]^2 + 2 \omega_0^2 \sin[\delta]^2 - \right. \right.$$

$$\left. \left. \frac{2 \beta \omega (2 t \omega - \sin[2 \delta] + \sin[2 (\delta + t \omega)])}{\left(\sqrt{\left(a_0^2 / \left(4 \beta^2 \omega^2 + (\omega^2 - \omega_0^2)^2 \right) \right)} \right)} \right) \right) / \left(2 \left(4 \beta^2 \omega^2 + (\omega^2 - \omega_0^2)^2 \right) \right)$$

$$- \frac{2 a_0^2 \beta \omega^2}{4 \beta^2 \omega^2 + (\omega^2 - \omega_0^2)^2} - a_0 \omega \sqrt{\left(a_0^2 / \left(4 \beta^2 \omega^2 + (\omega^2 - \omega_0^2)^2 \right) \right)} \sin[\delta]$$

FullSimplify[c /. {Sin[δ] → Sinδ}]

0

(*Los términos que son proporcionales a t no están porque se cancelan. Se extraen.*)

parte4 = FullSimplify[

TrigExpand[FullSimplify[parte3 - c * t]] /. {Cos[δ] → Cosδ, Sin[δ] → Sinδ}]

$$\left(a_0^2 \left((\omega^2 + \omega_0^2) \left(4 \beta^2 \omega^2 + (\omega^2 - \omega_0^2)^2 \right) + \right. \right.$$

$$\left. \left. (\omega - \omega_0) (\omega + \omega_0) \left(\left(-4 \beta^2 \omega^2 + (\omega^2 - \omega_0^2)^2 \right) \cos[2 t \omega] + 4 \beta \omega (-\omega^2 + \omega_0^2) \sin[2 t \omega] \right) \right) \right) /$$

$$\left(2 \left(4 \beta^2 \omega^2 + (\omega^2 - \omega_0^2)^2 \right)^2 \right)$$

(*Sumamos (ω^2-ω0^2)*x[t]^2 a la ecuación*)

parte4A = (ω^2 - ω0^2) * FullSimplify[TrigExpand[x[t]^2]

/. {Cos[δ] → Cosδ, Sin[δ] → Sinδ, Cos[δ]^2 → Cosδ^2, Sin[δ]^2 → Sinδ^2}]

$$\left((\omega^2 - \omega_0^2) \left(2 a_0 \beta \omega \cos[t \omega] + a_0 (\omega - \omega_0) (\omega + \omega_0) \sin[t \omega] \right)^2 \right) / \left(4 \beta^2 \omega^2 + (\omega^2 - \omega_0^2)^2 \right)^2$$

parte5 = FullSimplify[parte4 + parte4A]

$$\frac{a\theta^2 \omega^2}{4 \beta^2 \omega^2 + (\omega^2 - \omega\theta^2)^2}$$

FullSimplify[At^2]

FullSimplify[At^2 \omega^2 - parte5]

$$\frac{a\theta^2}{4 \beta^2 \omega^2 + (\omega^2 - \omega\theta^2)^2}$$

0

(*Conservación de la energía*)

eq[x1_, v1_] = -v1^2 - \omega^2 * x1^2 + parte5

$$-v1^2 - x1^2 \omega^2 + \frac{a\theta^2 \omega^2}{4 \beta^2 \omega^2 + (\omega^2 - \omega\theta^2)^2}$$

(*Esta es la ecuación de la elipse que vemos en el plano de fase*)

Capítulo 7

Apéndice: Estudio de la estabilidad

En esta memoria hemos observado que sistemas sometidos a amortiguación tienden hacia un punto fijo, como se puede inferir a partir de sus respectivos planos de fase (por ejemplo, ver la figura 2.8), mientras que otros, como en el caso de sistemas forzados como el representado en la figura 2.11 tienden hacia una órbita cerrada.

Podríamos pensar que la tendencia que presentan estas trayectorias está relacionada con la estabilidad del sistema. De hecho, el sistema tiende a regresar a dicho estado (el punto fijo u órbita cerrada) tras sufrir una perturbación. La perturbación se obtiene variando las condiciones iniciales entorno al punto de equilibrio.

Además, también cabe resaltar que un punto fijo no sólo funciona como sumidero. Si el sistema es estable, es lógico pensar que las trayectorias del plano de fase tenderán hacia él, pero también puede ocurrir que en caso de que el sistema sea inestable, el sentido de las trayectorias se invierta y se alejen del punto fijo. En cualquier caso, nos será de interés para concluir la memoria describir los cálculos necesarios para poder definir si el sistema estudiado es estable, y determinar el tipo de estabilidad que presentará.

Así, podemos comenzar con la ecuación del oscilador armónico simple, donde:

$$\frac{dx}{dt} = v; \quad \frac{dv}{dt} = -\omega_0^2 x; \quad (7.1)$$

que a su vez se puede representar como un sistema de ecuaciones en forma matricial:

$$\begin{bmatrix} \frac{dx}{dt} \\ \frac{dv}{dt} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\omega_0^2 & 0 \end{bmatrix} \begin{bmatrix} x \\ v \end{bmatrix}.$$

Este método también será aplicable en el caso de sistemas con ecuaciones

no lineales, y será necesario realizar una linealización de cada una de las funciones que lo componen en torno al punto de equilibrio. Por ejemplo, para el caso del péndulo simple, cuando las amplitudes no son lo suficientemente pequeñas:

$$\begin{aligned}\frac{dx}{dt} &= v = f_1(x, v); \\ \frac{dv}{dt} &= -\omega_0^2 \operatorname{sen}(x) = f_2(x, v).\end{aligned}\quad (7.2)$$

Los puntos fijos del sistema serían $v_n = 0$ y $x_n = n\pi$, $n = 0, \pm 1, \dots$. Trabajaríamos con el sistema linealizado en torno a cada punto de equilibrio¹. Sustituimos $x = x_n + \Delta x$ y $v = v_n + \Delta v$ en las ecuaciones anteriores y despreciamos todos los términos de orden dos o mayor en Δx y Δv

$$\begin{bmatrix} \frac{d\Delta x}{dt} \\ \frac{d\Delta v}{dt} \end{bmatrix} = \begin{bmatrix} \frac{df_1}{dx}(x_n, v_n) & \frac{df_1}{dv}(x_n, v_n) \\ \frac{df_2}{dx}(x_n, v_n) & \frac{df_2}{dv}(x_n, v_n) \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta v \end{bmatrix}.$$

Siendo Δx y Δv conocidas como variables de desviación del sistema en torno al punto de equilibrio o de funcionamiento. El método descrito a continuación será aplicable entonces a este mismo sistema linealizado.

Continuando con el sistema original del oscilador lineal, más sencillo, necesitaremos obtener el polinomio característico de la matriz A , también conocida como matriz de estado:

$$A = \begin{bmatrix} 0 & 1 \\ -\omega_0^2 & 0 \end{bmatrix}.$$

Así, el polinomio característico se obtiene a partir del cálculo de la ecuación que nos permitirá obtener los autovalores de la matriz A . Partiendo de $A - \lambda I = 0$:

$$\begin{bmatrix} 0 & 1 \\ -\omega_0^2 & 0 \end{bmatrix} - \begin{bmatrix} \lambda & 0 \\ 0 & \lambda \end{bmatrix} = 0;$$

$$\begin{vmatrix} -\lambda & 1 \\ -\omega_0^2 & -\lambda \end{vmatrix} = 0,$$

obtenemos el polinomio característico:

$$\lambda^2 + \omega_0^2 = 0, \Rightarrow \lambda_{1,2} = \pm i\omega_0; \quad (7.3)$$

Los autovalores obtenidos serán imaginarios, y por tanto volvemos a obtener los valores obtenidos anteriormente en la resolución de (2.8), como podemos

¹Como el sistema es periódico se estudian sólo dos puntos de equilibrio, $n = 0$ y $n = 1$

observar en la ecuación (2.12).

En el caso de las oscilaciones amortiguadas, realizando el mismo análisis podemos obtener que los autovalores del polinomio característico son:

$$\lambda_{1,2} = -d \pm i\omega_d = 0, \quad (7.4)$$

siendo $\omega_d = \sqrt{\omega_0^2 - d^2}$, lo cual coincide a su vez con la ecuación (2.19).

En este caso, si $d < \omega_0$ entonces los autovalores serán complejos conjugados y el plano de fase resultará ser una espiral estable dirigida hacia el punto fijo, mientras que si $d > \omega_0$ (lo que se conocerá como régimen sobreamortiguado), entonces los dos autovalores serán reales y negativos, y el punto fijo será un nodo estable.

Finalmente, para $d = \omega_0$, obtenemos un único autovalor negativo igual a $\lambda_{1,2} = -d$ y el punto fijo será un nodo estable. Otro análisis (equivalente) a realizar será a partir del determinante de la matriz de estado y la traza, donde:

$$A = \begin{bmatrix} a & b \\ c & d \end{bmatrix}.$$

El determinante será

$$D = ad - bc; \quad (7.5)$$

Y la traza:

$$\zeta = a + d. \quad (7.6)$$

Para valores positivos de la traza el sistema será inestable, y además podemos calcular los autovalores como:

$$\lambda_{1,2} = \zeta \pm \frac{\sqrt{\zeta^2 - 4D}}{2}. \quad (7.7)$$

Y además, en función de si el sistema es subamortiguado, sobreamortiguado, críticamente amortiguado, estable o inestable, podremos definir las regiones de estabilidad representadas en la figura 7.1 donde se clasifican en función de los valores de sus trazas y determinantes.

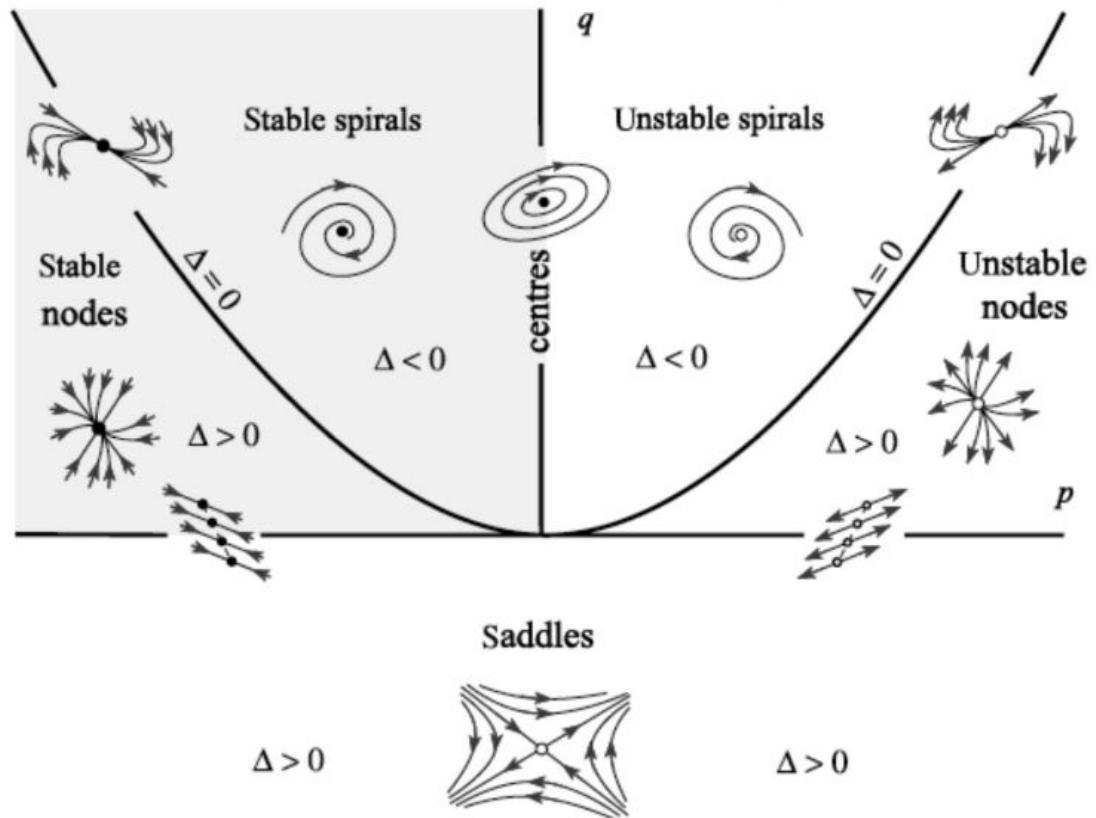


Figura 7.1: Clasificación de los planos de fase de los distintos tipos de sistema estudiados en función de sus determinantes y sus trazas, como aparece representado en las diapositivas de la referencia bibliográfica [16], siendo q el determinante y p la traza. $\Delta = \zeta^2 - 4D$.

Capítulo 8

Conclusiones

En el primer capítulo estudiamos la naturaleza de los principales movimientos oscilatorios, para el oscilador armónico simple, el amortiguado, y el amortiguado y forzado, siguiendo una metodología constructiva a partir de la cual se han ido describiendo cada uno de los elementos que caracterizan sus comportamientos.

Así, para el oscilador armónico simple utilizamos el sistema masa-resorte, donde la partícula está sometida a una fuerza elástica lineal. Esto determinaba el comportamiento oscilatorio de la posición de la partícula y la frecuencia natural ω_0 del oscilador armónico simple, que es independiente de las condiciones iniciales. La amplitud de las oscilaciones queda determinada por las condiciones iniciales.

Para el oscilador armónico amortiguado introducimos una fuerza de fricción proporcional a la velocidad de la partícula (fricción de Stokes), originada a partir de la acción del rozamiento entre las partículas y el medio. Estudiamos tres regímenes, dependiendo de la relación entre el coeficiente de disipación y la frecuencia propia del sistema. En todos los casos, la partícula termina en su posición de equilibrio.

Finalmente, para el oscilador armónico amortiguado y forzado, estudiamos la influencia de una fuerza externa constante. La dinámica de la partícula en este caso es similar al caso del oscilador armónico amortiguado. La diferencia principal es que la fuerza constante, fija otra posición de equilibrio.

Si la fuerza es periódica, entonces la evolución de la partícula consta de dos partes claramente diferenciadas, el régimen transitorio que desaparece después de un cierto tiempo $t \gg 1/\beta$ y un régimen estacionario, donde la partícula oscila con la frecuencia de la fuerza externa. Curiosamente, para determinadas condiciones iniciales, el régimen transitorio no existe y el movimiento de la partícula es periódico desde el instante inicial. Esto hace que en el plano de fase, las curvas sean también elipses cerradas, como en el caso del oscilador armónico simple.

Una vez encontradas las ecuaciones de movimiento de cada uno de los osciladores, obteníamos una ecuación diferencial ordinaria, que junto con las condiciones iniciales, determinan la aceleración, la velocidad y la posición de la partícula en cualquier instante de tiempo. Debido al carácter lineal de la dinámica, estas ecuaciones se resolvieron analíticamente. También, hemos realizado un estudio numérico de las mismas, hemos representado los planos de fase de cada uno de los osciladores, y hemos demostrado que las curvas observadas en el plano de fase, representan la conservación de la energía total del sistema.

Obteníamos no ya una representación en dos dimensiones, sino en tres, del comportamiento de los mismos y que podemos observar en las figuras 8.1-8.3. En la memoria también estudiamos en detalle la influencia de la fricción de

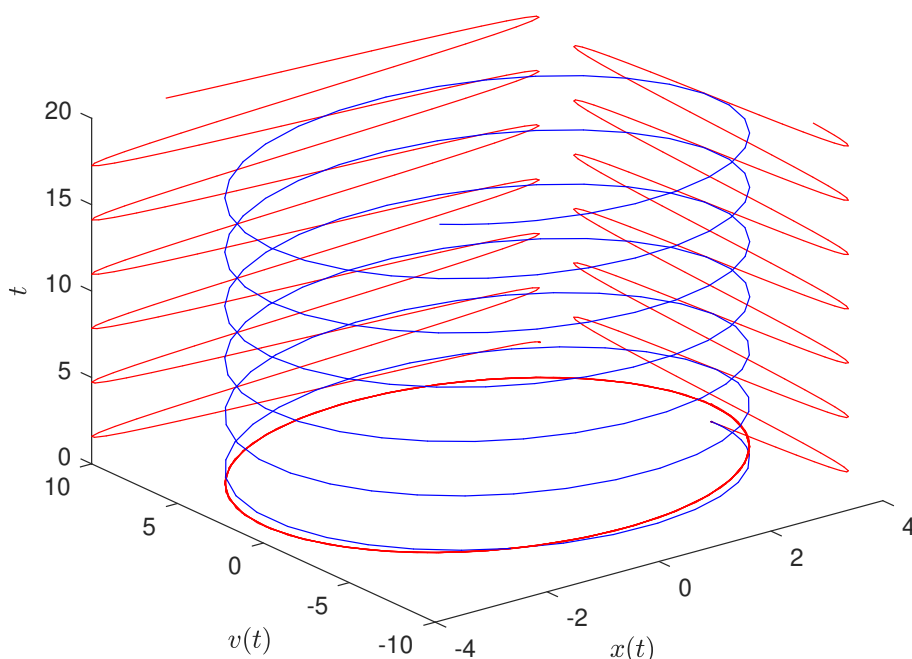


Figura 8.1: Oscilador armónico simple. Condiciones iniciales: $x(0) = 4$ y $v(0) = 0$. Parámetros: $\omega_0 = 2$.

Coulomb en el sistema masa-resorte. Aunque el sistema es no lineal, se puede resolver exactamente porque en intervalos de tiempo es lineal a trozos. Es de destacar que en cada intervalo de tiempo hay una magnitud que se conserva [3]. Este tipo de disipación también lleva al sistema a su estado de equilibrio. Hay tres diferencias fundamentales con la fricción de Stokes, (i) la aceleración es una función discontinua en los extremos de cada intervalo, (ii) la amplitud de las oscilaciones no decae exponencialmente, y (iii) la posición de equilibrio aquí, generalmente, queda desplazada del punto $x = 0$. Como podemos ver en la

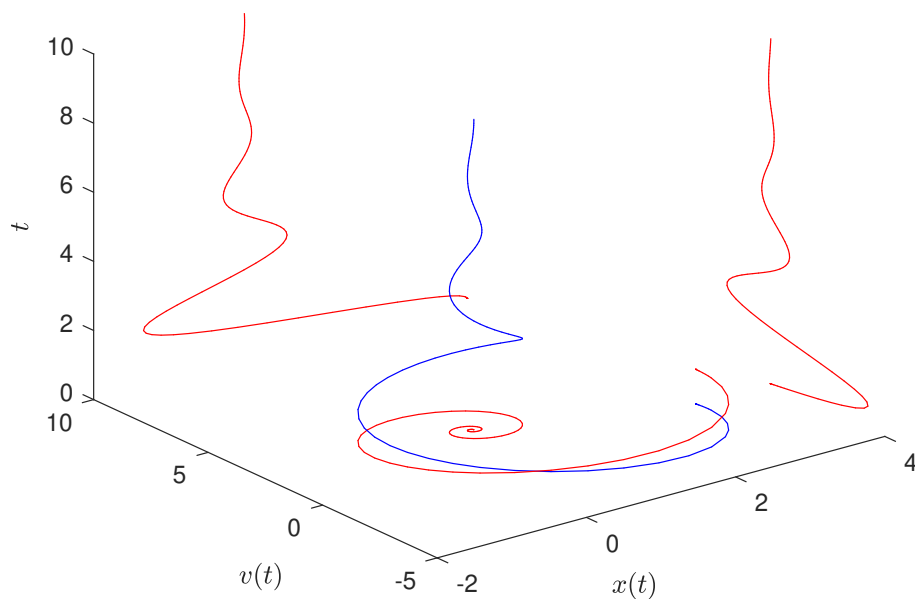


Figura 8.2: Oscilador armónico amortiguado. Condiciones iniciales: $x(0) = 3$, $v(0) = 0$. Parámetros: $\omega_0 = 2$ y $\beta = 0,5$.

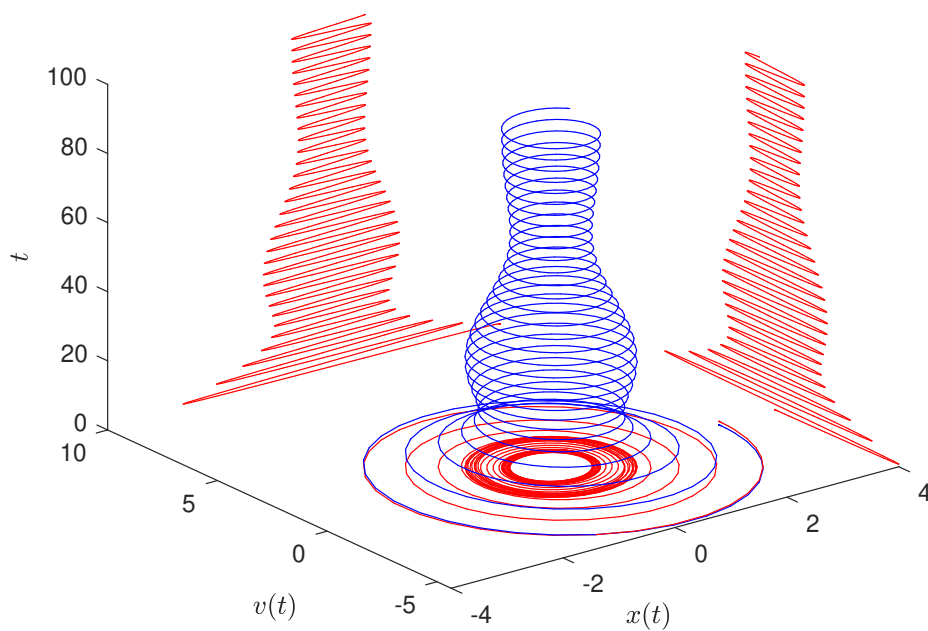


Figura 8.3: Oscilador armónico amortiguado y forzado. Condiciones iniciales: $x(0) = 3$, $v(0) = 0$. Parámetros: $\omega_0 = 2$, $\beta = 0,05$, $a_0 = 0,3$, $b_0 = 0$, y $\omega = 1,9$.

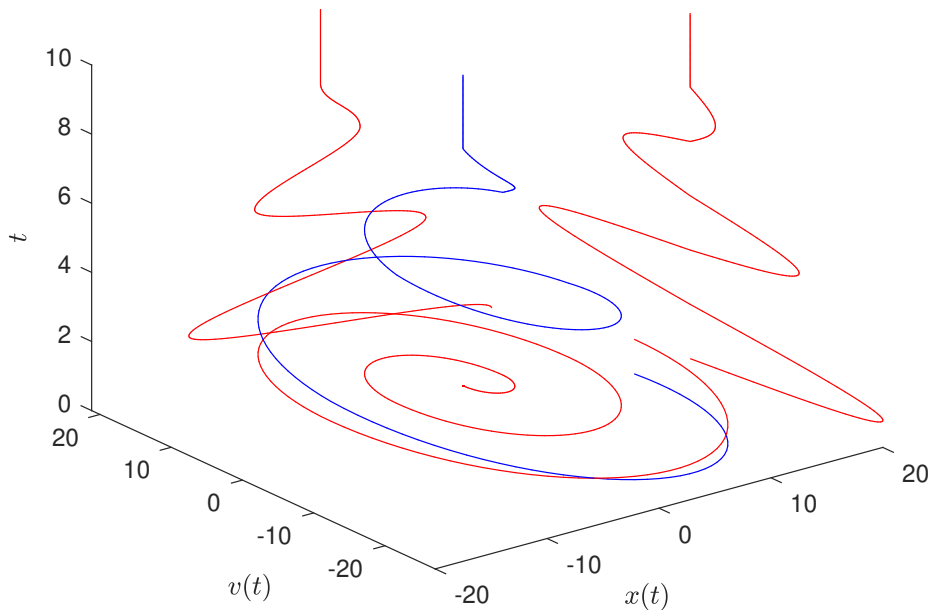


Figura 8.4: Oscilador armónico amortiguado sometido a fricción dinámica y estática. Condiciones iniciales: $x(0) = 15$, $v(0) = 0$. Parámetros: $\omega_0 = 2$, $\alpha = 5,88$ y $\mu_s/\mu_k = 1,2$.

figura 8.4, un tratamiento análogo aplicado a los resultados obtenidos a partir del estudio del efecto de la fricción estática y dinámica produce un resultado similar al del oscilador armónico amortiguado.

Así, los resultados fundamentales de este trabajo han sido:

1. La creación de programas de cálculo numérico en *Matlab* para obtener la posición y la velocidad de los osciladores estudiados, así como la comparación de los resultados analíticos con la solución aproximada calculada por *Matlab*. Estos códigos serán utilizados en un futuro en el análisis de sistemas no lineales, donde se desconocen las soluciones exactas.
2. Cabe destacar que en el caso del oscilador sometido a fricción estática y dinámica, la solución numérica programada (utilizando los resultados teóricos) se resolvía sin errores, mientras que la solución aproximada obtenida por *Matlab* desembocaba en muchas ocasiones en una fuga de memoria. Esto es debido a que la función *ode* tomaba un paso entre puntos a calcular demasiado pequeño, y generaba resultados de manera indiscriminada y sin criterio de parada, que terminaban con el fallo del programa o el bloqueo al alcanzar el límite de memoria disponible. Además, la solución numérica

se obtenía en cuestión de milisegundos, mientras el cálculo de la solución aproximada (si funcionaba) podía alcanzar los siete minutos.

3. La creación de un modelo sólido para la simulación y representación gráfica de los resultados obtenidos, así como la caracterización tridimensional de los planos de fase de cada uno de los osciladores. El diseño de animaciones a través de SolidWorks. Todo ello nos permite estudiar en detalle el comportamiento del sistema tanto en términos de energía, como de velocidad, desplazamiento y aceleración para cualquier momento dado.
4. El estudio de la conservación de la energía de los osciladores, así como la representación gráfica del fenómeno a través de los planos de fase, lo cual es de especial interés en el caso de los osciladores amortiguados, forzado y sometido a fricción estática y dinámica.
5. El estudio de la solución particular del oscilador armónico amortiguado y forzado para el cual se anula el régimen transitorio, donde pudimos ver que se anulaba la solución homogénea del sistema y el oscilador “olvidaba” su frecuencia natural y las condiciones iniciales, oscilando a la frecuencia de la fuerza externa a la que se encuentra sometido desde el instante inicial.

A través del trabajo, hemos ofrecido una visión global y detallada del comportamiento de un oscilador armónico, y de la influencia de la fricción de Stokes y de las fuerzas externas sobre el mismo. Esto nos ha permitido estudiar el efecto de la fricción no lineal, reduciendo el problema al caso del oscilador armónico simple.

Existen otros problemas y cuestiones que han surgido en el desarrollo de la memoria y que nos gustaría abordar en un futuro, como por ejemplo:

1. En el cálculo del oscilador sometido a fricción no lineal, por simplicidad, partíamos del supuesto de que la velocidad inicial de la partícula era nula. Mientras que para los casos anteriores, la velocidad inicial era una de las condiciones iniciales que podíamos definir. Por tanto, el próximo objetivo será resolver el sistema dinámico cuando la velocidad inicial sea diferente de cero.
2. El oscilador no lineal forzado y con disipación, para determinados valores de los parámetros presenta comportamiento caótico [12]. Este fenómeno, sólo puede darse en sistemas no lineales muy complejos, los cuales no son del todo comprendidos. El estudio del oscilador armónico sometido a

fricción no lineal y forzado es un sistema perfecto para investigar la aparición del caos, puesto que se trata de un sistema relativamente sencillo de analizar, pero al mismo tiempo suficientemente complejo como para que el fenómeno aparezca.

3. Los planos de fase obtenidos representan el comportamiento de partículas puntuales. El mismo estudio se aplicará a ondas viajeras localizadas, de manera que podamos establecer un cierto paralelismo, encontrando similitudes entre el comportamiento de un oscilador y la propagación del centro de masa de la onda (o de la energía) en dichos sistemas. Otra cuestión importante será investigar el uso de dichos planos de fase en la estabilidad de estos sistemas.

Finalmente, cabe destacar de nuevo que las gráficas y recursos multimedia obtenidos a partir de las simulaciones y animaciones con el SolidWorks nos ayudan a presentar este tipo de problemas, con un enfoque didáctico.

Es de tener en cuenta que en apenas tres capítulos, se ha realizado un completo recorrido desde principios básicos conocidos por los estudiantes más novatos, como pueden ser la ley de Hooke y el comportamiento de un oscilador armónico simple, hasta los casos más complicados que involucran comportamientos no lineales. Aunque las primeras ecuaciones diferenciales estudiadas, las lineales, suponen realmente una minoría de todas aquellas con las que físicos, matemáticos e ingenieros trabajan a diario (aunque sean obtenidas a través de aproximaciones), su estudio nos proporciona una herramienta fundamental a la hora de investigar los problemas no lineales.

Los sistemas no lineales y el movimiento caótico tienen sus propias características (tales como la sensibilidad a las condiciones iniciales, de la que se puede leer en más detalle en [12]) que complican el estudio de los mismos, y es por ello que un tratamiento apropiado de los resultados puede permitir una comprensión mucho más intuitiva de los mismos. La interrelación de conceptos, la representación gráfica de resultados y la simulación suponen una punta de lanza para el estudio desde el punto de vista teórico de estos sistemas, allanando el camino al mismo tiempo que se avanza y permitiendo una asimilación más sencilla de los nuevos conocimientos adquiridos.

Porque, si bien en un principio puede resultar imponente la idea de que todo lo estudiado en lo referente a sistemas lineales y no lineales es una parte ínfima, y que la frontera entre sistemas no lineales que podemos tratar y el comportamiento caótico es difusa, facilitar la accesibilidad y comprensión ha sido una de las

tareas principales realizadas por expertos de renombre, como Richard Feynman y John R. Taylor.

Bibliografía

- [1] A. Bedford and W. Fowler, *Mecánica para ingeniería, Dinámica*, 5ª edición. Pearson Educación, México 2008.
- [2] I. I. Blekhman, *Selected Topics in Vibrational Mechanics, Vol. 11, Series A*, World Scientific Publishing Co., 2004.
- [3] L. M. Burko, *A piecewise-conserved constant of motion for a dissipative system*, Eur. J. Phys. **20**, 281 (1999).
- [4] R. P. Feynman y R. B. Leighton, *Física-Volumen I: Mecánica, radiación y calor*, Addison Wesley Longman de México S.A., México, 1998.
- [5] R. Hauko, D. Andreevski, D. Paul, M. Sterk, and R. Repnik, *Teaching of the harmonic oscillator damped by a constant force: The use of analogy and experiments*, Am. J. Phys. **86**, 657 (2018)
- [6] Software *Wolfram Mathematica*. Versión 9.0.
- [7] R. D. Peters and T. Pritchett, *The not-so-simple harmonic oscillator*, Am. J. Phys. **65** 1067, (1997).
- [8] G. D. Quiroga, and P. A. Ospina-Henao, *Dynamics of damped oscillations: physical pendulum*, Eur. J. Phys. **38**, 065005 (2017).
- [9] R. D. Peters, and T. Pritchett, *The not-so-simple harmonic oscillator*, Am. J. Phys. **65**, 1067 (1997).
- [10] R. Resnick, D. Halliday and K. S. Krane *Physics, Vol. 1*, 4ª edición, Compañía Editorial Continental, México 2001.
- [11] P. T. Squire, *Pendulum damping*, Am. J. Phys. **54** 984, (1986).
- [12] J. R. Taylor, *Mecánica clásica*, Editorial Reverté S.A., España, 2013.
- [13] P. A. Tipler y G. Mosca, *Física para la ciencia y la tecnología, Vol. 1*, 5ª edición. Editorial Reverté S.A., España, 2005.

- [14] B. Carter and M. S. Carter, *Latitude, How american astronomers solved the mystery of variation*, Naval Institute Press, Annapolis, 2003.
- [15] C. C. Cortés-García, *Identificación de una bifurcación de Hopf con o sin parámetros*, Revista de Ciencias Vol.21, Cali July/Dec. 2017.
- [16] E. Vizcaya Xilotl, *Sistemas dinámicos*, Programa de Ciencia y Humanismo, FC-UNAM / CEFPSVLT, 2017.
- [17] V. Farrell, *SOLIDWORKS Simulation: Introduction to Motion Analysis*, Hawk Ridge Systems, 2017, recuperado de: <https://hawkridgesys.com/blog/solidworks-simulation-introduction-motion-analysis>.
- [18] M.C. Casado Fernández, *Manual básico de matlab*, Servicios Informáticos U.C.M, Apoyo a Investigación y Docencia, 2004.
- [19] M. A. López-Mariño, J. A. Hernández-Olvera, L. A. Barroso, J. C. Trujillo Caballero, *Cómputo simbólico y gráfico: estudio del sistema masa-resorte*, Rev. Bras. Ensino Fís. vol.39 no.2, São Paulo 2017.