# oscilacionesForzadas_20sF2adap

March 23, 2021

## 1 Oscilaciones Forzadas

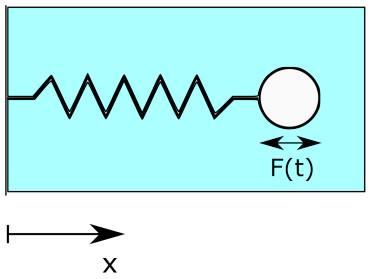
Adaptación del notebook de Paulina Knees y Milton Katz, F2 2.o cuat. 2020, teórica de Ricardo Depine


---

Vamos a repasar el oscilador amortiguado y el forzado. Además vamos a usar algunas funciones de python para acostumbrarnos a esta nueva (para algunxs) herramienta.

Tenemos un problema unidimensional de una pesa de masa $m$ engarzada a un resorte de constante elástica $k$ y longitud natural $l_0$. Podemos suponer que está inmerso en un fluido viscoso con constante $c$. El enunciado de la guía nos sugiere que utilicemos $k = m\omega_0^2$ y $\Gamma = c/m$.

En física 1 solían poner el eje x con el origen en la pared.




Lo primero que tenemos que hacer entonces es escribir la ecuación de Newton:

$$m\ddot{x} = F_{elas} + F_{visc} \tag{1}$$

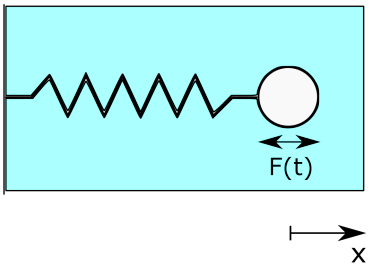
Las únicas fuerzas presentes son la elástica y la viscosa, las escribimos explícitamente, recuerden que ambas son restitutivas.

$$m\ddot{x} = -k(x - l_0) - c\dot{x} \tag{2}$$

$$\implies \ddot{x} = -\omega_0^2 x - \Gamma\dot{x} + l_0\omega_0^2 \tag{3}$$

En el último paso reescribimos la ecuación de Newton con las constantes propuestas por la guía.

Para hacerlo más sencillo, les propongo que este ejercicio lo encaremos con el eje en la posición en la que el resorte está relajado (posición de equilibrio). Además es lo que vamos a hacer durante




toda la materia.

Considerando también la fuerza $F(t)$, pueden ver que la ecuación que queda es

$$\ddot{x} + \Gamma\dot{x} + \omega_0^2 x = \frac{F_0}{m}\cos\Omega t$$

¿Se acuerdan cómo resolver estas ecuaciones? Al ser lineal, la solución más general es la suma de la solución de la ecuación homogénea y una solución particular de esta ecuación tal como la tenemos acá.

Primero resolvamos el sistema homogeneo, ¿cuál es?

$$\ddot{x}_h + \omega_0^2 x_h + \Gamma\dot{x}_h = 0 \qquad (1) \qquad (4)$$

Esta ecuación aparece en la guía y el problema nos propone la solución

$$x_h(t) = Ce^{-t/2\tau}\cos(\omega_1 t + \theta).$$

Las constantes $\omega_1$ y $\tau$ no las conocemos. Para hallarlas, tenemos que encontrar las derivadas (primera y segunda) de $x_h$ y reemplazar en (1). Las constantes $C$ y $\theta$ van a quedar definidas una vez que explicitemos las condiciones iniciales. Les voy a dejar hechas las derivadas, los pasos intermedios se los dejo a ustedes (es importante que los hagan).

$$\dot{x}_h = -Ce^{t/2\tau}\left(\frac{1}{2\tau}\cos(\omega_1 t + \theta) + \omega_1\sin(\omega_1 t + \theta)\right) \qquad (2)$$

$$\ddot{x}_h = Ce^{t/2\tau}\left[\frac{1}{2\tau}\left(\frac{\cos(\omega_1 t + \theta)}{2\tau} + \omega_1\sin(\omega_1 t + \theta)\right) + \frac{\omega_1}{2\tau}\sin(\omega_1 t + \theta) - \omega_1^2\cos(\omega_1 t + \theta)\right] \qquad (3)$$

Ahora tenemos que reemplazar todo en (1). De nuevo, varios pasos intermedios les quedan a ustedes.

$$\begin{aligned} 0 =& Ce^{-t/2\tau}\left[\frac{1}{2\tau}\left(\frac{\cos(\omega_1 t + \theta)}{2\tau} + \omega_1\sin(\omega_1 t + \theta)\right) + \frac{\omega_1}{2\tau}\sin(\omega_1 t + \theta) - \omega_1^2\cos(\omega_1 t + \theta)\right] \\ & - \Gamma Ce^{-t/2\tau}\left(\frac{1}{2\tau}\cos(\omega_1 t + \theta) + \omega_1\sin(\omega_1 t + \theta)\right) + \omega_0^2 Ce^{-t/2\tau}\cos(\omega_1 t + \theta) \end{aligned}$$

Los términos que están en color los podemos cancelar dado que toda la ecuación está igualada a 0. *¿Por qué estos términos no pueden ser igual a cero*?

Nos queda entonces la siguiente ecuación

$$\cos(\omega_1 t+\theta)\left(\frac{1}{(2\tau)^2}-\omega_1^2-\frac{\Gamma}{2\tau}+\omega_0^2\right)+\sin(\omega_1 t+\theta)\left(\frac{\omega_1}{2\tau}+\frac{\omega_1}{2\tau}-\Gamma\omega_1\right)=0 \tag{5}$$

¿Qué podemos hacer para hallar $\tau$ y $\omega_1$? Escribí de esa forma la ecuación por el siguiente motivo. Las funciones sin y cos forman base completa del espacio de funciones (ya vamos a volver con esto). Esto significa que puedo describir lo que quiera como combinación de senos y cosenos. Además, como forman una base completa y lo que tenemos está igualado a cero, la única opción que nos queda es que lo que acompaña a cada una de las funciones sea cero. Podemos hacer una analogía con las bases de vectores. En ese caso, ya sabemos que podemos escribir cualquier vector como combinación lineal de vectores que llamamos linealmente independientes. En este caso pasa lo mismo. Estamos escribiendo el resultado en base de los "autovectores" seno y coseno.

Entonces ahora nos queda un sistema de dos ecuaciones con dos incógnitas bastante sencillo para resolver

$$\left.\begin{aligned}\frac{1}{(2\tau)^2}-\omega_1^2-\frac{\Gamma}{2\tau}+\omega_0^2=0\\ \frac{\omega_1}{2\tau}+\frac{\omega_1}{2\tau}-\Gamma\omega_1=0\end{aligned}\right\}\implies \tau=\frac{1}{\Gamma},\ \omega_1^2=\omega_0^2-\frac{\Gamma^2}{4}$$

Ahora viene la parte más interesante. Podemos ver que $\omega_1$ depende de dos parámetros del problema. El tipo de solución del problema va a depender de la relación entre esos parámetros. Tenemos tres casos posibles, $\omega_1^2>0$, $\omega_1^2<0$ y $\omega_1^2=0$. Estudiemos los tres casos y grafiquemos las tres soluciones posibles.

Recordamos la forma funcional de la posición de la pesa como función del tiempo $x_h(t)=Ce^{-\Gamma t/2}\cos(\omega_1 t+\theta)$.

## 1.1 Caso subamortiguado

En este caso con $\omega_0^2>\frac{\Gamma^2}{4}$ tenemos el producto de una exponencial que decae y una oscilación Esperamos que hayan osilaciones en el sistema, pero cada vez más atenuadas por la exponencial hasta que decaiga completamente. Vamos a graficarlo.

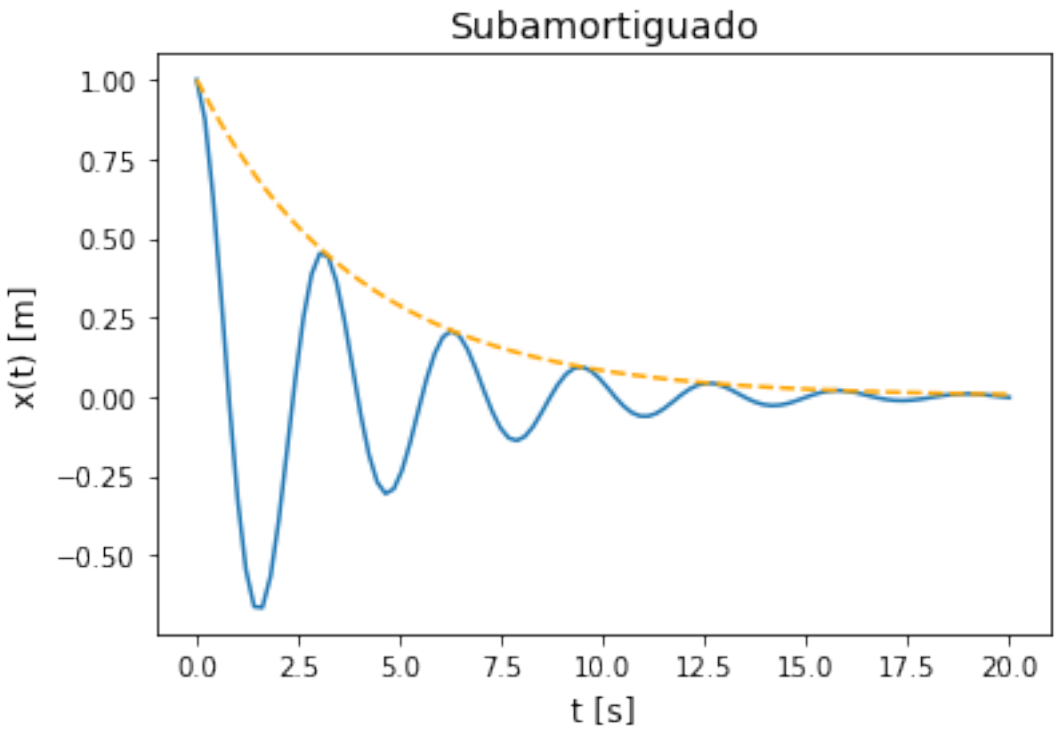
[16]:
```
import numpy as np
import matplotlib.pyplot as plt
```

[17]:
```
t = np.linspace(0,20,100)
Gamma = 0.5 #viscosidad del fluido donde está inmerso el sistema.
w0 = 2 #w0 depende de la pesa y de la constante del resorte.
omega1 = np.sqrt(w0**2-Gamma**2/4)
C = 1 #theta y C dependen de las condiciones iniciales. Supongamos que son␣
 ↪tales que C=1 y theta=0.
theta = 0
```

[22]:
```
sub_t = np.exp(-Gamma* t/ 2)* np.cos(omega1* t)
exp_t = np.exp(-Gamma* t/ 2)

fig, ax = plt.subplots()
ax.set_title('Subamortiguado', fontsize=14)
ax.set_xlabel('t [s]', fontsize=12)
ax.set_ylabel('x(t) [m]', fontsize=12)
ax.plot(t, sub_t)
ax.plot(t, exp_t,'--',color = 'orange') #graficamos en línea punteada la␣
↪exponencial sola.
```

[22]:
```
[<matplotlib.lines.Line2D at 0x7f4bc5a1dbe0>]
```



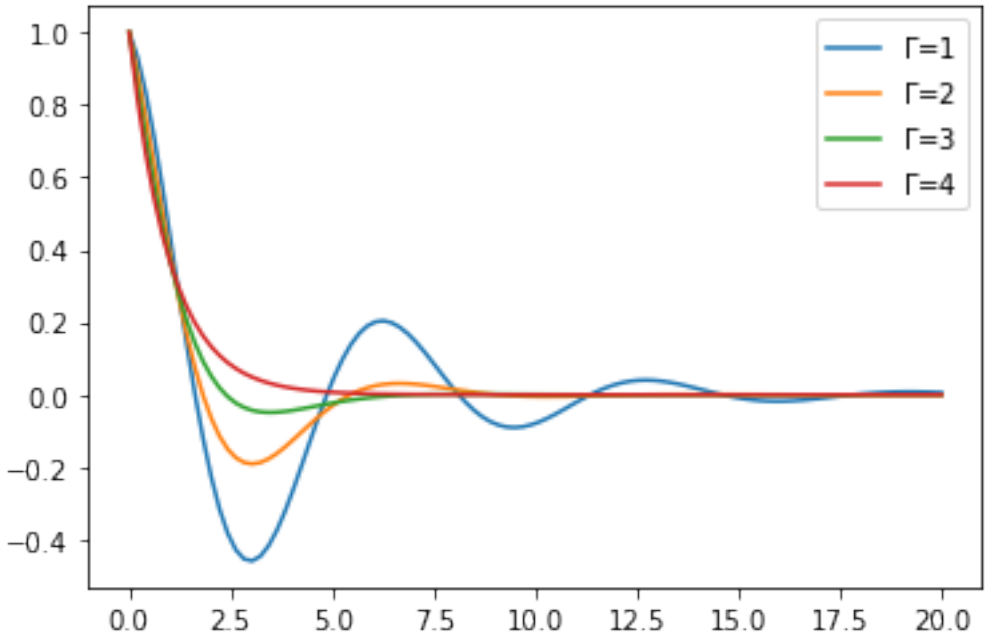

Como vemos, tenemos una oscilación que con el tiempo decae a causa de la exponencial.

Supongamos que estamos haciendo un experimento y podemos cambiar el fluido donde está inmerso el resorte con la pesa, es decir, podemos cambiar el $\Gamma$. Vamos a ver cómo cambia $x(t)$ a medida que nos aceramos al caso crítico.

[20]:
```
t1 = np.linspace(0,10,100)
Gamma1 = 1
Gamma2 = 2
Gamma3 = 3
Gamma4 = 4
```

```
omega_1 = np.sqrt(w0**2-Gamma1**2/4)
omega_2 = np.sqrt(w0**2-Gamma2**2/4)
omega_3 = np.sqrt(w0**2-Gamma3**2/4)
omega_4 = np.sqrt(w0**2-Gamma4**2/4)
sub_t1 = np.exp(-Gamma1*t1/2)*np.cos(omega_1*t1)
sub_t2 = np.exp(-Gamma2*t1/2)*np.cos(omega_2*t1)
sub_t3 = np.exp(-Gamma3*t1/2)*np.cos(omega_3*t1)
sub_t4 = np.exp(-Gamma4*t1/2)*np.cos(omega_4*t1)

fig, ax = plt.subplots()
ax.plot(t, sub_t1, label='$\Gamma$=1')
ax.plot(t, sub_t2, label='$\Gamma$=2')
ax.plot(t, sub_t3, label='$\Gamma$=3')
ax.plot(t, sub_t4, label='$\Gamma$=4')
ax.legend()
```




Vemos que a medida que $\Gamma$ está más cerca del valor que tomamos de $\omega_0$, el gráfico más se aproxima al de una exponencial decreciente, como lo es en el caso crítico.

## 1.2 Caso de amortiguamiento crítico

Supongamos que a partir del caso anterior, cambiamos el fluido y llegamos al caso crítico donde $\omega_0^2 = \frac{\Gamma^2}{4}$. Es sencillo ver que sólo tenemos la exponencial, dado que el $\cos(0) = 1$.

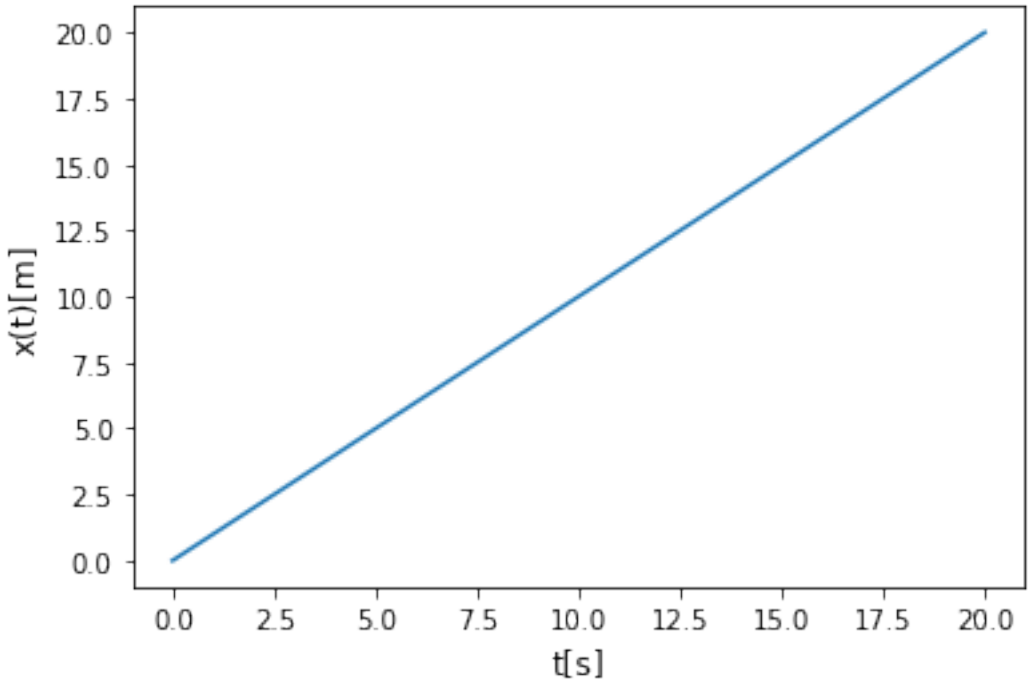
¿Se animan a graficarlo ustedes? Les dejamos el esqueleto del código para que piensen el resto ustedes. Fíjense en los gráficos anteriores y hagan lo mismo

[27]:
```
#¿Hace falta importar las bibliotecas de nuevo?
#¿Hace falta definir de nuevo t?

Gamma_c=2*w0
crit_t= 0 # Acá la exponencial

fig, ax = plt.subplots()
#Ponele un título al gráfico
ax.set_xlabel('t[s]', fontsize=12)
ax.set_ylabel('x(t)[m]', fontsize=12)
ax.plot(t, t) #Completá
```

[27]: [<matplotlib.lines.Line2D at 0x7f4bc5991358>]




### 1.3 Caso sobreamortiguado

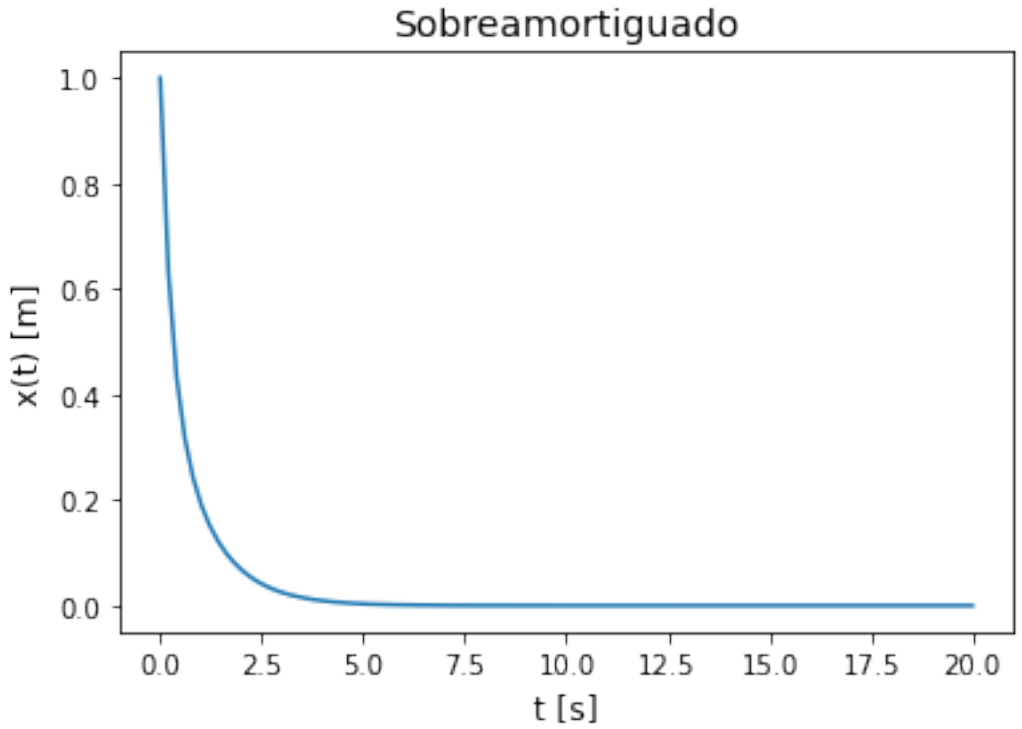
En este caso $\omega_0^2 < \frac{\Gamma^2}{4}$ y nuestro problema cambia de régimen. Tenemos un problema (que no es tan problemático en realidad). ¡Ahora $\omega_1$ es un número complejo! Escribimos $\omega_1 = i|\omega_1|$. Les dejo a ustedes comprobar que $\cos(ix) = \cosh(x)$ (probar esto es ejercicio de ustedes).

Para esto recuerden que el coseno hiperbólico se puede escribir como suma de exponenciales (¿y el coseno?). Para convencerse pueden fijarse en el plot qué pasa si en vez de escribir $np.\cos$, escriben $np.\cosh$

[28]:
```
Gamma5 = 5
omega1_sob = np.sqrt(Gamma5**2 /4 - w0**2)
sobre_t = np.exp(- Gamma5* t/ 2)* np.cos(1j* omega1_sob* t)

fig, ax = plt.subplots()
ax.set_title('Sobreamortiguado', fontsize=14)
ax.set_xlabel('t [s]', fontsize=12)
ax.set_ylabel('x(t) [m]', fontsize=12)
ax.plot(t, sobre_t.real)
```

[28]:
```
[<matplotlib.lines.Line2D at 0x7f4bc57adf98>]
```




Fijense que a $t \to \infty$, todas las soluciones van a cero, es decir que no importa en qué caso estemos, la parte homogenea de la solución va a desaparecer con el tiempo, esto le da el nombre a esta parte de la solución de *transitorio*.

### 1.4 Forzado

Concentrémonos ahora en el término que perdura en el tiempo que lo llamaremos *estacionario* y lo obtendremos en la solución particular. Hallémosla

$$\ddot{x}_p + \Gamma \dot{x}_p + \omega_0^2 x_p = \frac{F_0}{m} \cos \Omega t$$

¿Cómo resolvemos esto? Haciendole caso a la guía, en este caso al ejercicio 9. El eunciado nos dice

que nos da la solución para tiempos suficientemente grandes donde solo existe la parte estacionaria, exactamente lo que estamos buscando.

$$x_p(t) = A\sin\Omega t + B\cos\Omega t$$

Al igual que lo que hicieron antes, deriven y reemplacen en la ecuación.

$$(\omega_0^2 A - A\Omega^2 - \Gamma\Omega B)\sin\Omega t + (\omega_0^2 B + \Gamma\Omega A - B\Omega^2 - \frac{F_0}{m})\cos\Omega t = 0$$

Al igual que antes agrupo las cosas para que me queden dos factores que tienen que ser 0 independientemente.

$$(\omega_0^2 - \Omega^2)A - \Gamma\Omega B = 0$$
$$\Gamma\Omega A + (\omega_0^2 - \Omega^2)B - \frac{F_0}{m} = 0$$

Al igual que en la parte anterior nos queda un sistema de 2x2 que podemos resolver.

$$A = \frac{F_0}{m}\frac{\Gamma\Omega}{\Gamma^2\Omega^2 + (\omega_0^2 - \Omega^2)^2}$$

$$B = \frac{F_0}{m}\frac{(\omega_0^2 - \Omega^2)}{\Gamma^2\Omega^2 + (\omega_0^2 - \Omega^2)^2}$$

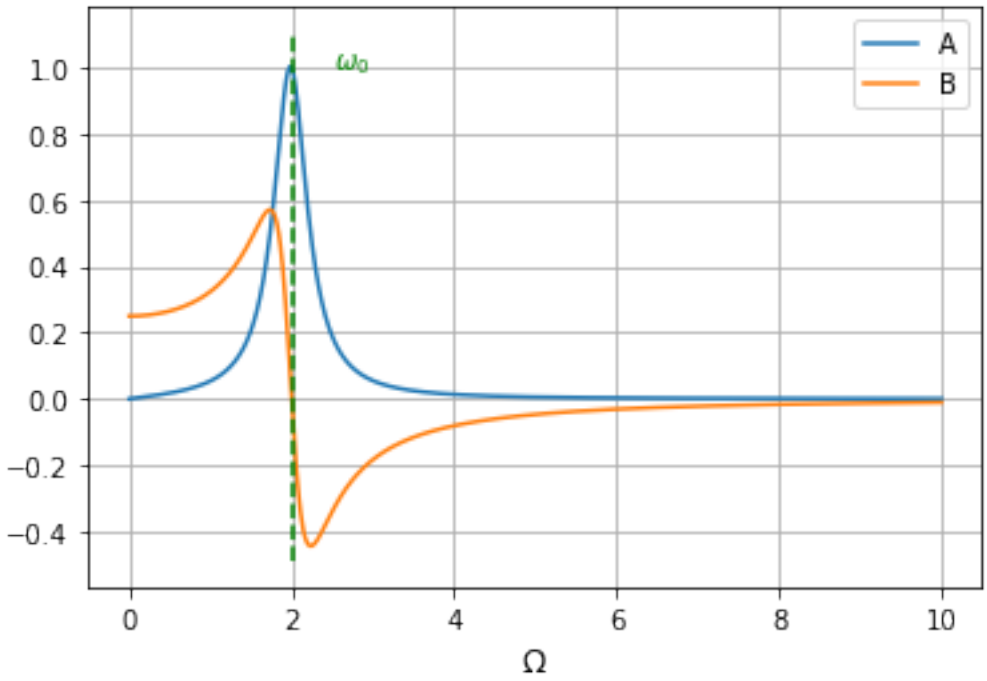
Lo primero que llama la atención de estas ecuaciones es la resonancia cuando $\Omega = \omega_0$, veamos los gráficos.

[33]:
```
m = 1
F = 1
omega0 = 2
gamma = 0.5

omega = np.linspace(0,10,1000)
A = (F/m)* gamma*omega/ (gamma**2 * omega**2 + (omega0**2 - omega**2)**2)
B = (F/m)* (omega0**2 - omega**2)/ (gamma**2 * omega**2 + (omega0**2 -␣
 ↪omega**2)**2)

fig, ax = plt.subplots()
ax.set_xlabel('$\Omega$', fontsize=12)
ax.plot(omega, A, label = 'A')
ax.plot(omega, B, label = 'B')
ax.plot([omega0]*2,[1.1*min(B),1.1*max(A)],'--',color = 'green') #Ploteo una␣
 ↪linea punteada en omega0 para ver la resonancia, hablemos de esta sintaxis,␣
 ↪acuerdense que sumar dos listas devuelve una nueva lista con la unión de las␣
 ↪dos. Entonces multiplicar es como sumarla con sí misma, hubiese sido lo␣
 ↪mismo escribir [omega0,omega0] pero quería mostrarles esto que suele ser␣
 ↪útil cuando queremos una lista de algún elemento repetida muchas veces
```

```
ax.text(2.5,1,'$\omega_0$',color = 'green') #Estás son cosas que agrego para␣
↪mostrarles que se pueden ir haciendo gráficos cada vez más lindos y para que␣
↪sepan que estas herramientas existen. Si les confunde, no les den bola
ax.legend()
ax.grid()
```



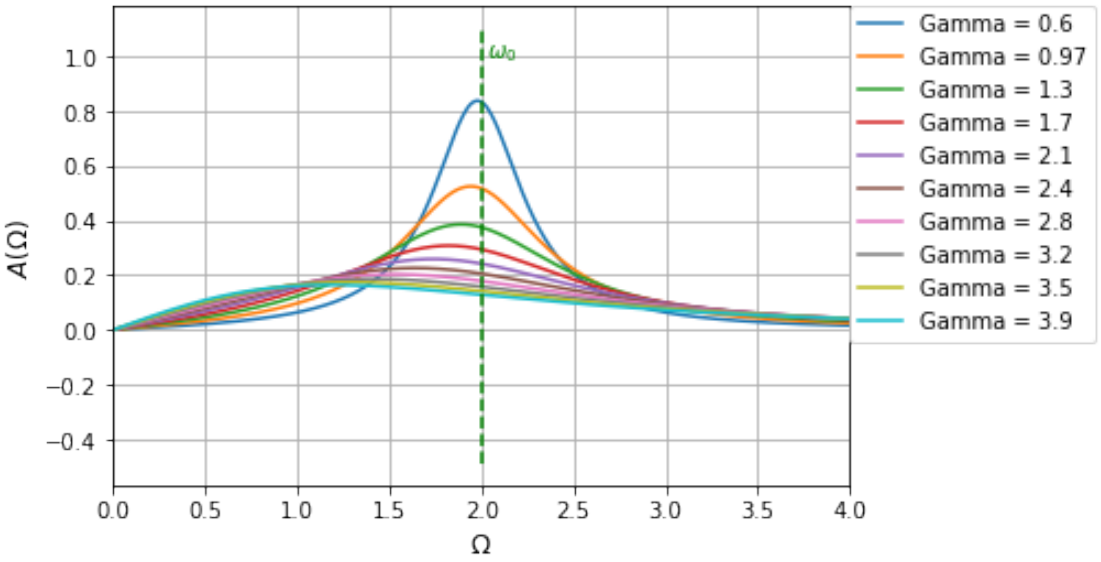

Pero qué es esto? una amplitud es negativa? Se hace 0 en $\omega_0$? ¡Pero a mí me dijeron que la resonancia es cuando la amplitud crece y se rompe todo! Bueno, no, eso está pasando con A, pero no con B, es por esto que A que, en la resonancia, absorbe toda la potencia que le damos, se llama **amplitud absorbente**. Por otro lado, a B se le llama **amplitud elástica (o dispersiva)** y se la asocia con absorción de potencia instantánea pero la potencia absorbida da cero si se la promedia en un ciclo completo (como en un resorte que absorbe y devuelve).

Otra cosa que podemos ver en el gráfico es que la resonancia no está exactamente en $\omega_0$, la amortiguación $\Gamma$ desplaza un poco el punto de máxima intensidad (prueben buscar el $\Omega$ que maximiza la amplitud $A$). Acá la parte analítica se las dejo a ustedes, yo les muestro lo que pasa en el gráfico cuando de a poco cambio gamma, vean que el pico se va desplazando

[40]:
```
gamVar = np.linspace(0.6,3.9,10)

fig, ax = plt.subplots()
for i in range(10):
  Avar = (F/m)* gamVar[i]* omega/ (gamVar[i]**2 * omega**2 + (omega0**2 -␣
↪omega**2)**2)
```

```
  ax.plot(omega,Avar, label = f'Gamma = {gamVar[i]:.2}') #Se dan cuenta cómo␣
 ↪formaté el texto?
ax.plot([omega0]*2,[1.1*min(B),1.1*max(A)],'--',color = 'green')
ax.text(2.02,1,'$\omega_0$',color = 'green')
ax.set_xlabel('$\Omega$', fontsize=12)
ax.set_ylabel('$A(\Omega)$', fontsize=12)
ax.set_xlim([0,4])
ax.legend(loc = [1,0.3])
ax.grid()
```




Podemos ver lo mismo pero animado, esto también lo hice con un código de python pero en otras IDEs, se trata solamente de agregar dos lineas dentro de ese for pero Colab no me lo reproducía como yo quería. Por eso, lo hice en otra IDE y lo subo como gif.

Volvamos a la potencia pero pongamosle números. La potencia instantánea está dada por $P(t) = F(t)\dot{x}_p$. Pero lo que más nos va a interesar es su promedio temporal durante un período

$$< P >= \frac{1}{T}\int_0^T P(t)dt = \frac{1}{T}\int_0^T F(t)\dot{x}_p(t)dt,$$

donde $T = \frac{2\pi}{\Omega}$ es el período de nuestra onda.

Para poder seguir les cuento que hay unas integrales que las vamos a usar durante toda la materia. Yo les cuento cuanto dan pero ustedes tienen la tarea de hacer las cuentas y entenderlas, son cuentas en una variable que no les deberían generar problemas pero como siempre si hay dudas consulten.

$$\int_0^T \sin\left(\frac{2\pi}{T}t\right)dt = 0$$

$$\int_0^T \cos\left(\frac{2\pi}{T}t\right)dt = 0$$

$$\int_0^T \sin^2\left(\frac{2\pi}{T}t\right) dt = \frac{T}{2}$$
$$\int_0^T \cos^2\left(\frac{2\pi}{T}t\right) dt = \frac{T}{2}$$
$$\int_0^T \sin\left(\frac{2\pi}{T}t\right)\cos\left(\frac{2\pi}{T}t\right) dt = 0$$

Volvamos a lo que estabamos haciendo.

$$P(t) = F_0 \cos\Omega t(A\Omega\cos\Omega t - B\Omega\sin\Omega t)$$

Así que cuando integremos y dividamos por $T$, usando las integrales que ahora ya conocemos, vamos a obtener

$$< P(t) >= \frac{F_0 A\Omega}{2}$$

Esta es la potencia que le estamos trasmitiendo a la masa, pero también hay una parte de la potencia que se disipa por fricción. ¿Cuánta potencia? Para calcularlo podemos hacer la misma integral de $P(t)$ en el tiempo pero en lugar de usar la potencia que usamos antes calculada con la fuerza aplicada a la masa, podemos usar la fuerza de fricción $F_r(t) = m\Gamma\dot{x}_p$. Entonces la potencia disipada queda

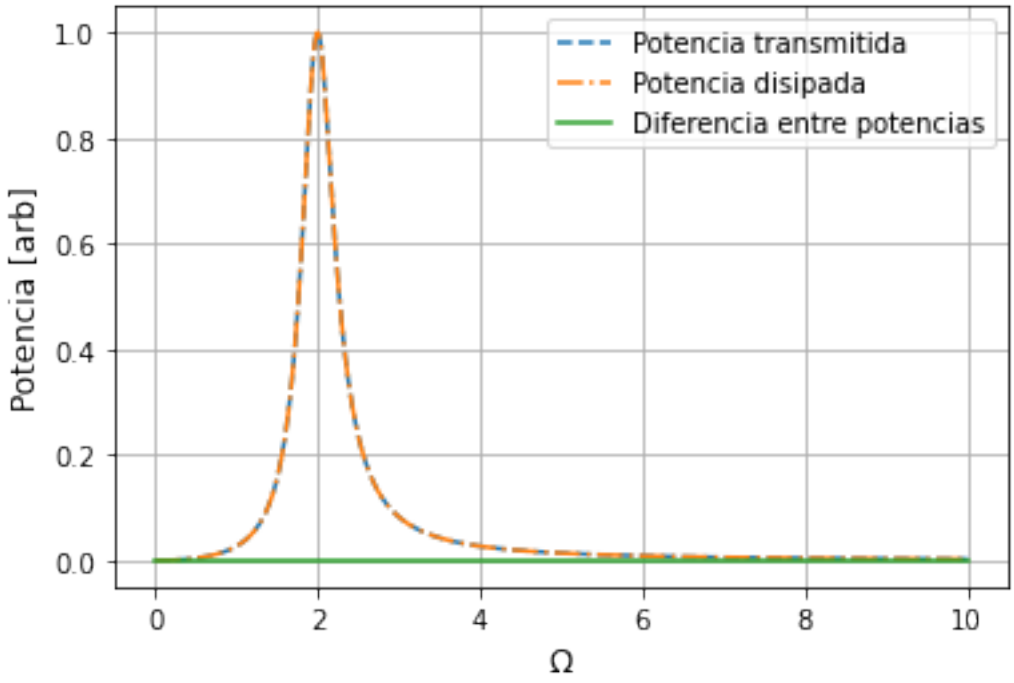
$$< P_{fr}(t) >= \frac{1}{T}\int_0^T m\Gamma\dot{x}_p^2\, dt$$

Igual que antes, integramos, dividimos por $T$ y obtenemos $< P_{fr}(t) >= \frac{1}{2}m\Gamma\Omega^2(A^2 + B^2)$.

La guía nos dice que verifiquemos la igualdad de estas dos ecuaciones pero antes pensemos, ¿por qué deberían ser iguales? En el régimen estacionario estamos haciendo una fuerza sobre la masa (realizando trabajo) y el movimiento es siempre el mismo. Si se disipara más de lo que se trasmite, la masa debería oscilar cada vez menos, si se trasmitiera más de lo que se disipa, la masa debería oscilar cada vez con más energía. Veamos qué nos dice Python.

[39]:
```
P = F*A*omega/2
Pfr = 0.5*m*gamma*omega**2*(A**2+B**2)

fig, ax = plt.subplots()
ax.plot(omega,P, '--',label = 'Potencia transmitida')
ax.plot(omega,Pfr,'-.', label = 'Potencia disipada')
ax.plot(omega, P - Pfr, label = 'Diferencia entre potencias')
ax.set_xlabel('$\Omega$', fontsize=12)
ax.set_ylabel('Potencia [arb]', fontsize=12)
plt.legend()
ax.grid()
```




¡Lo logramos! Fíjense que la curva azul quedó abajo de la naranja, eso se ve en la verde que es constantemente 0. Justo lo que esperabamos.

Seguramente ya les habrán contado muchas cosas y mostrado muchos videos de cosas que se rompen por resonancia, sino les dejo algunos videitos del puente de Tacoma y de un helicoptero que se rompe. La resonancia es un tema que va a aparecer un montón en la carrera y la idea es tan simple como la que acaban de ver (aunque las cuentas no tanto). ¿Se les ocurre porqué estas cosas se rompen? Piensenlo y nos lo cuentan por el campus.