



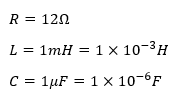
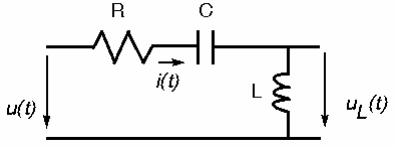
**Trabajo práctico Nº1 : Introducción a Matlab**

|  |
| --- |
| **Docentes** |
| **Dr. Juan Carlos Gómez**  **Ing. Franco Del Colle**  **Dr. Ariel Bayá**  **Dr. Gonzalo Sad** |

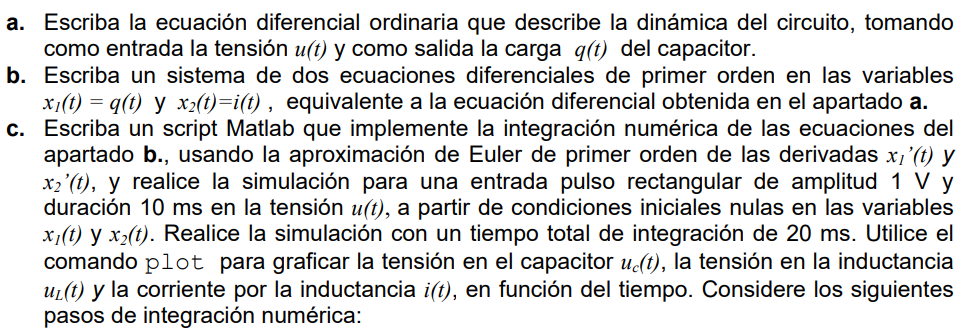
|  |  |
| --- | --- |
| **Integrantes del grupo** | |
| **BELLINI, Valentin** | **B-6127/1** |
| **GASPOZ, Gastón Leandro** | **G-5580/8** |

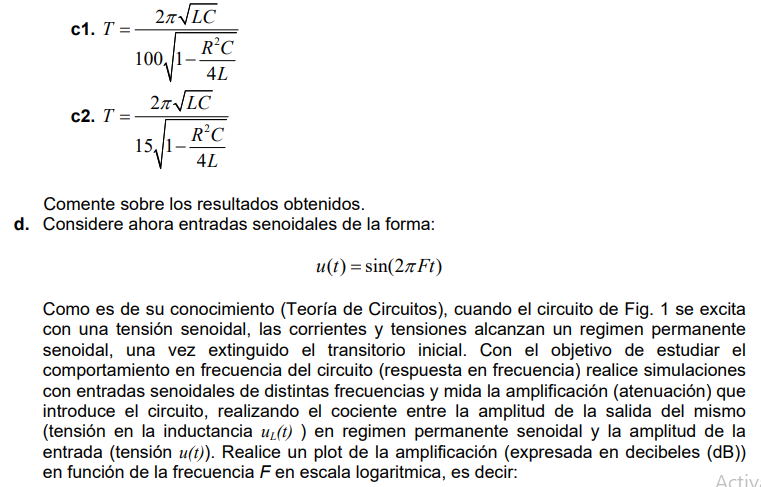
|  |  |
| --- | --- |
| **Fecha de entrega del trabajo** | **10/05/2021** |

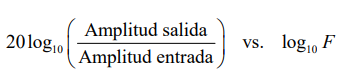
**Problema 1. Circuito RLC.**

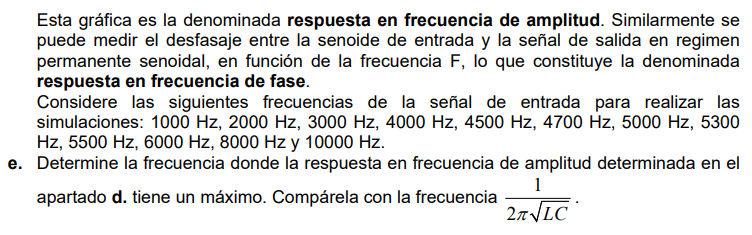


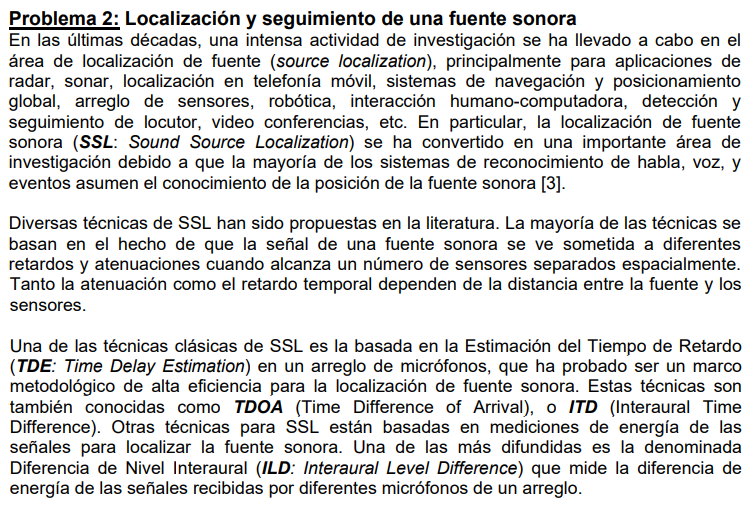
*Figura 1. Circuito RLC serie.*

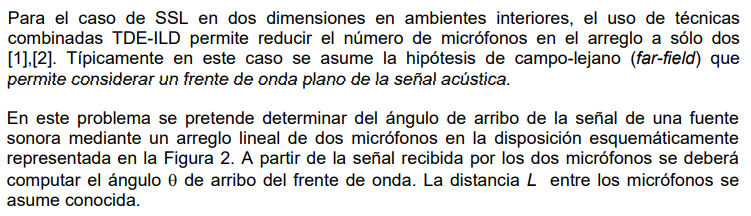


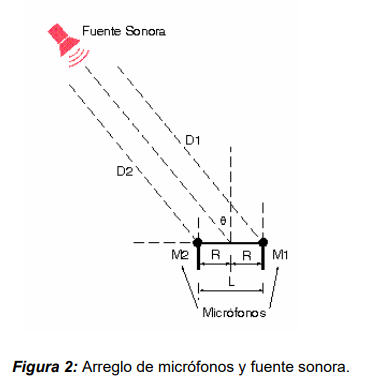


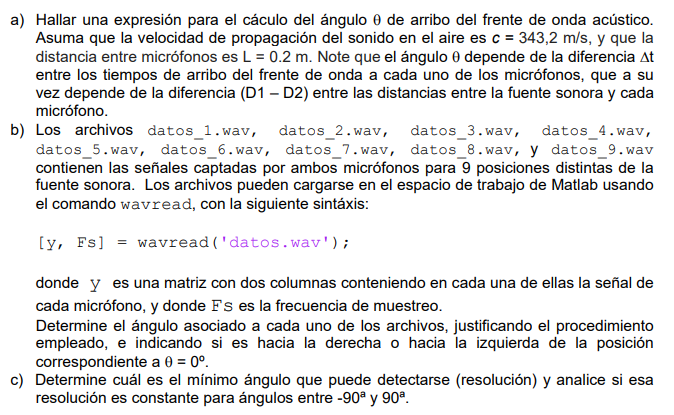












**Problema 1. Circuito RLC.**

En este problema se presenta un circuito RLC serie con una fuente de tensión. Se aplica el método de Euler y se analiza la respuesta de la salida con distintos pasos de integración numérica para una entrada definida como un pulso rectangular. En ítem posteriores se calcula la respuesta en amplitud y frecuencia de fase para luego comparar el valor de frecuencia máxima de amplitud obtenido con el valor máximo teórico conocido en teoría de circuitos.

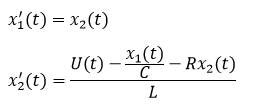
1. Para el circuito RLC serie, supondremos características de capacitor e inductor lineales para encontrar una relación mediante la tensión de entrada U(t) y la carga del capacitor q(t).

Las relaciones V-A de cada elemento serán las siguientes:

Planteamos ley de Kirchhoff de tensión en la malla y reemplazamos las tensiones por las relaciones V-A halladas

1. Obtuvimos una ecuación diferencial ordinaria (EDO) de 2º orden que describe la dinámica del sistema. Buscamos expresarla como un sistema con dos ecuaciones diferenciales ordinarias de 1º orden con un cambio de variables de la forma:

A partir de estas dos ecuaciones y despejando la derivada segunda de la carga en la ecuación (\*), se expresan las siguientes relaciones diferenciales:



Utilizando el método de Euler encontramos las aproximaciones:

Despejando las variables de interés :

Cambiando el índice en la discretización:

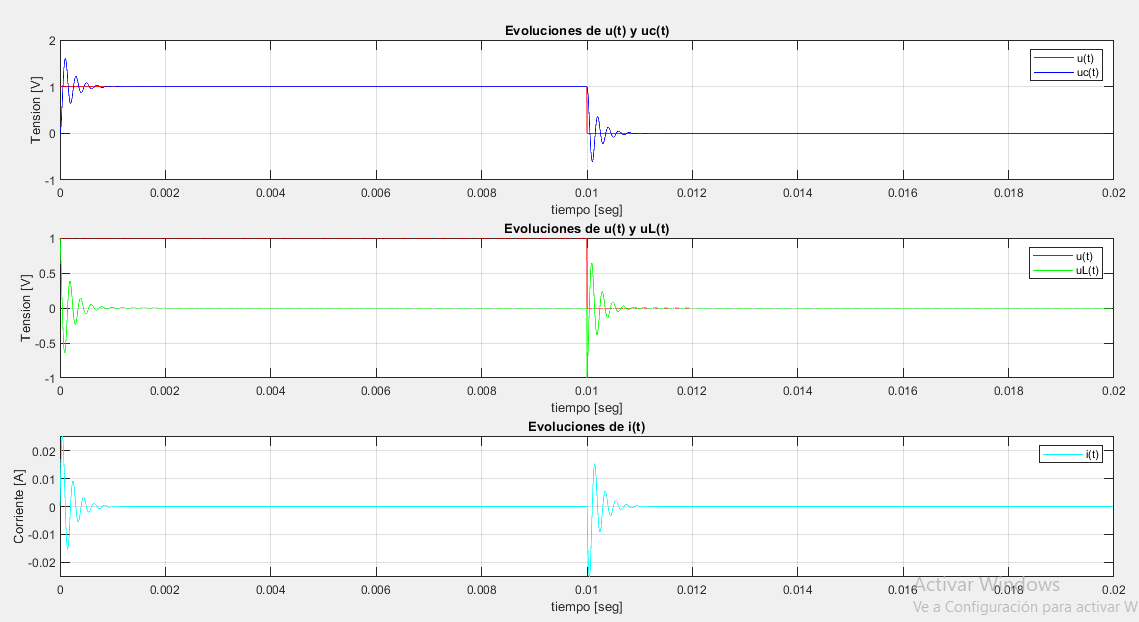
Finalmente se obtiene:

c) En el script de Matlab consideramos las condiciones iniciales de capacitor e inductor nulas, el pulso rectangular de 10ms y amplitud 1V. Se grafica las respuestas para un tiempo total de integración de 20ms y para un tiempo de 2ms con el fin de apreciar mejor el amortiguamiento.

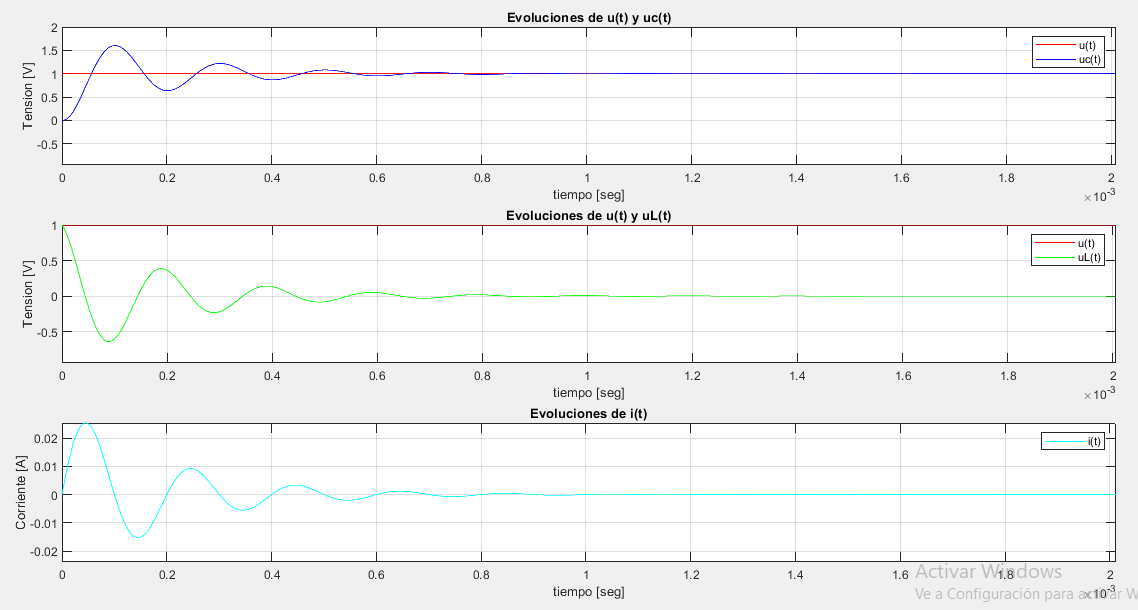
*Figura 2. Evoluciones temporales de las variables. Matlab.*

Considerando el paso de integración numérica T1:

*Figura 3. Evoluciones temporales del circuito para un tiempo total de integración de 20ms [T1=1.9869e-06]*



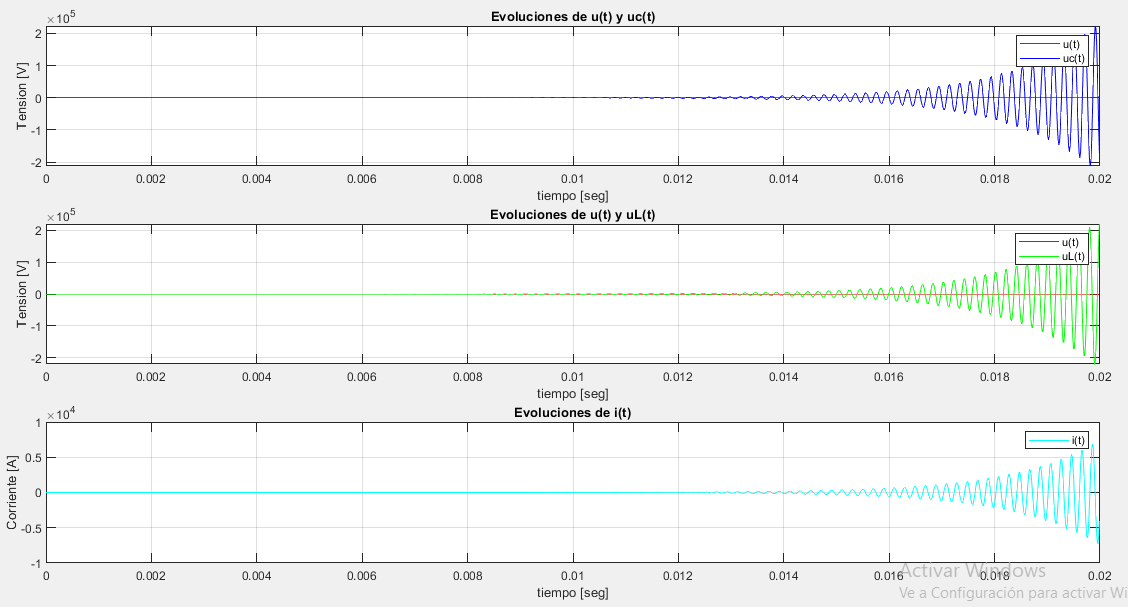
*Evoluciones temporales del circuito para un tiempo total de integración de 20ms*



*Figura 3. Evoluciones temporales del circuito para un tiempo total de integración de 2ms [T1=1.9869e-06]*

Podemos observar cómo, luego de un valor pico, como las respuestas comienza a oscilar estabilizándose mientras transcurre el tiempo. La respuesta que se presenta es de tipo subamortiguada para este paso de integración numérica.

Consideramos el paso de integración numérica T2:



*Figura 4. Evoluciones temporales del circuito para un tiempo total de integración de 20ms [T2 = 1.3246e-05]*

Observamos como en este caso el sistema se desestabiliza teniendo valores crecientes en su respuesta. Este tipo de sistema no es realizable físicamente por lo que deducimos que, al tratarse de un paso de integración numérica relativamente grande, el mismo induce error en los cálculos del software (Matlab) para el método de Euler, que son arrastrados dando una respuesta equivocada.

Observación: puede cambiarse el paso de integración variando el parámetro **“valor”** en el script.

A continuación, se puede ver el script de Matlab realizado para obtener los resultados mostrados.

|  |
| --- |
| Script: **“TP1\_Ej1parteI.m”** |
|  |

Observación: Podría haberse definido el pulso de 10ms como una diferencia de funciones de Heaviside.

A modo de presentación mostramos las respuestas de Uc(t) y UL(t) para otros pasos de integración.

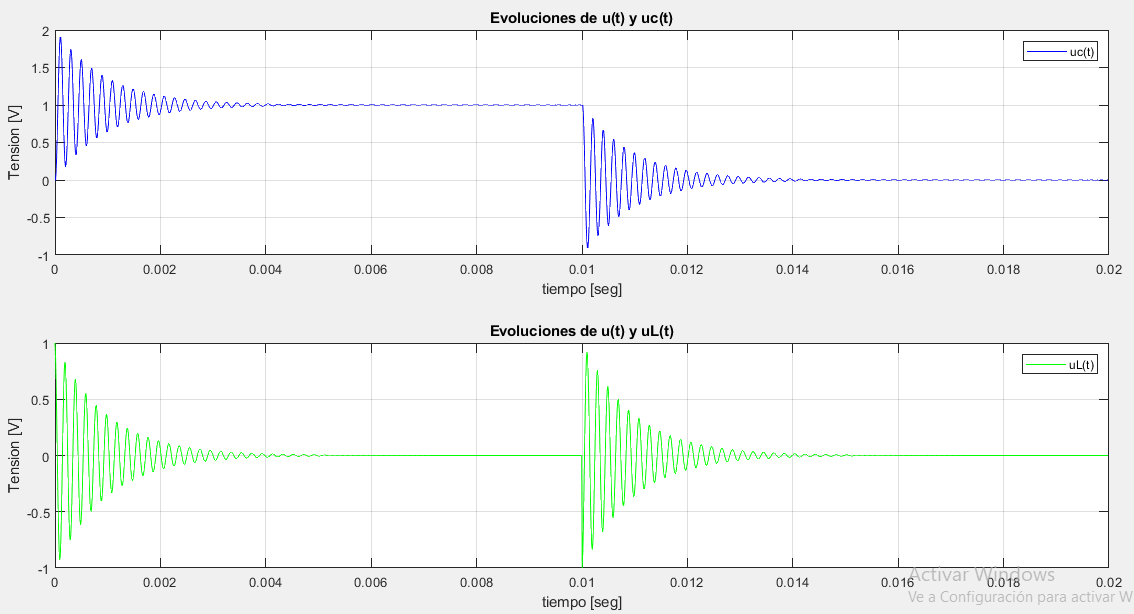


Figura 5a. Respuestas de tensiones para T = 9.9346e-06 (“valor” = 20)

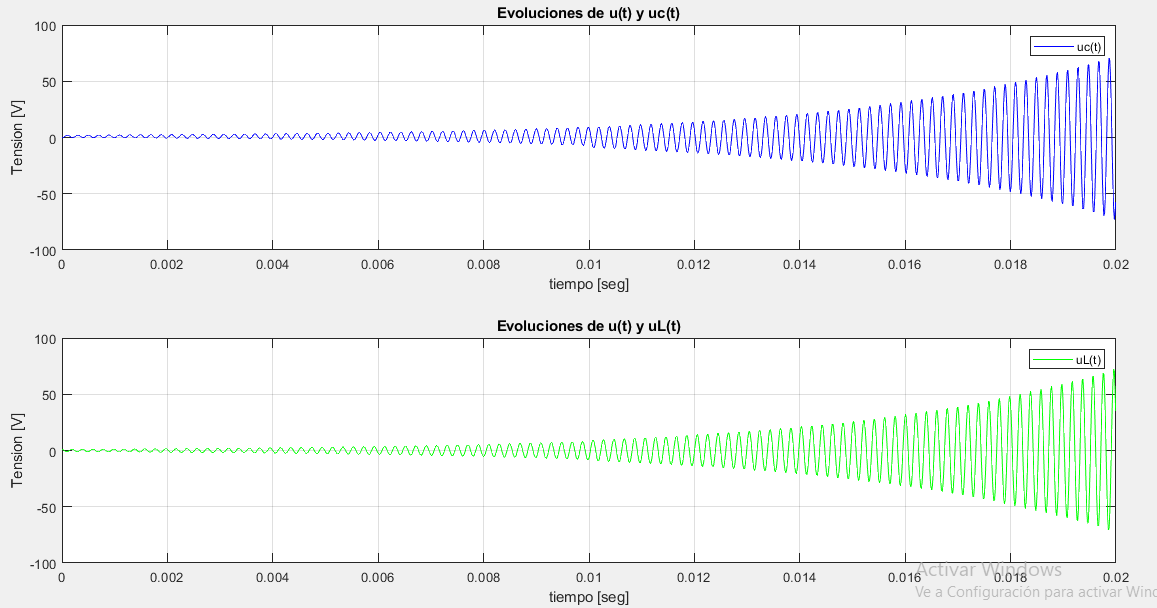


Figura 5b. Respuesta de tensiones para T = 1.2418e-05 (“valor” =16)

\*Observamos que, para este último, el paso de introducción ya induce un error en la aproximación visible en la gráfica.

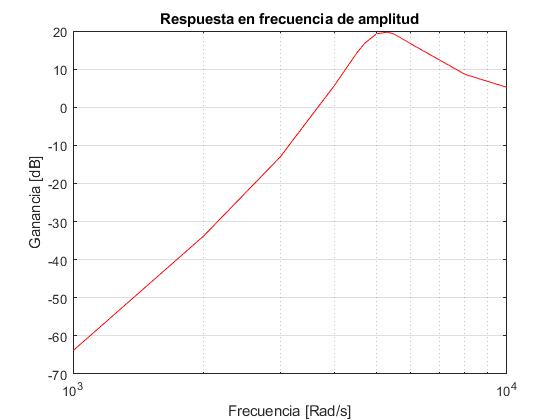
d) Una vez acabado el régimen transitorio, veremos cómo es la respuesta del circuito para distintos tipos de frecuencias dados. Trabajaremos con el método fasorial entonces:

Expresamos las variables del circuito en el sistema fasorial:

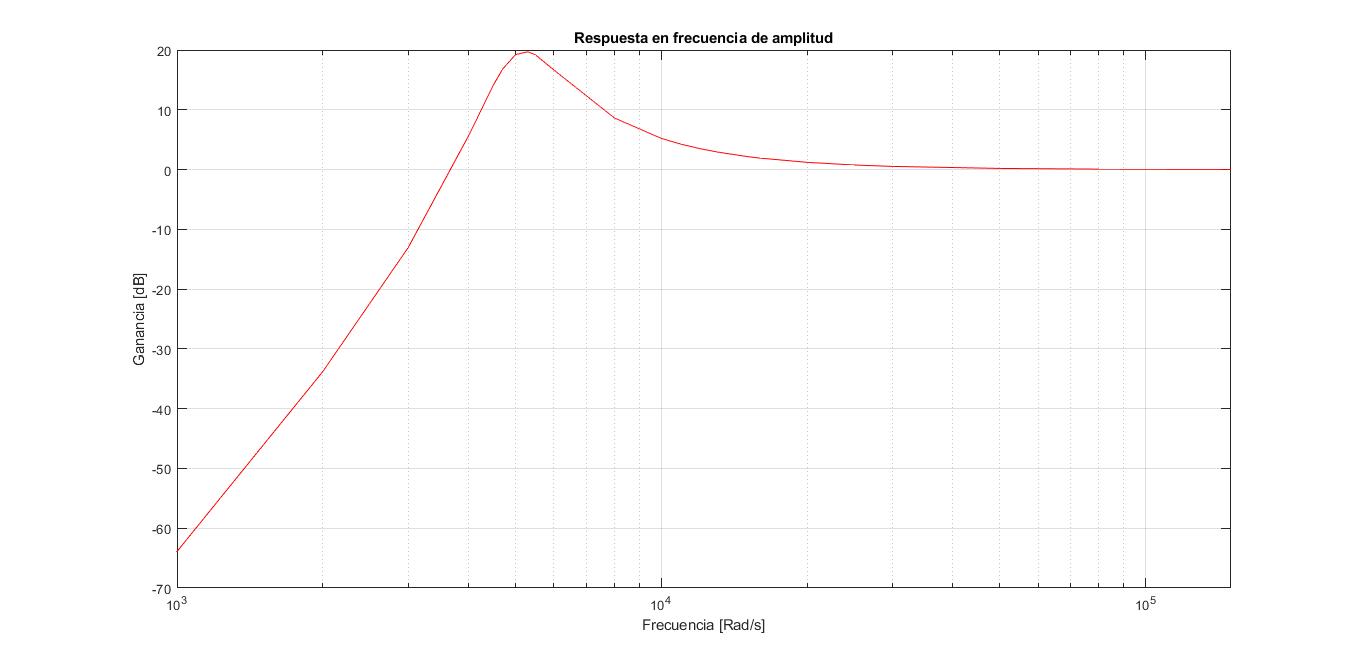
Luego la función transferencia del sistema será:

Utilizamos el siguiente script de Matlab desarrollado para graficar la respuesta de frecuencia de amplitud expresada en decibeles.

|  |
| --- |
| Script: **“TP1\_Ej1parteII.m”** |
|  |



*Figura 5a. Respuesta en frecuencia de amplitud*



*Figura 5b. Respuesta en frecuencia de amplitud con más datos en el vector frecuencias.*

Se observa como la ganancia máxima de la salida es 20db. Además, en la figura 5b, se adicionó frecuencias superiores para ver el comportamiento del circuito como filtro.

e) En el script de Matlab que vimos anteriormente, podemos observar cómo se calcula la frecuencia en la cual se produce el máximo de la respuesta. El valor por computo lo llamamos Fexperimental mientras que, al teórico, calculado con conocimientos de teoría de circuitos, lo llamamos Fteórico. Aquí podemos observar los resultados:

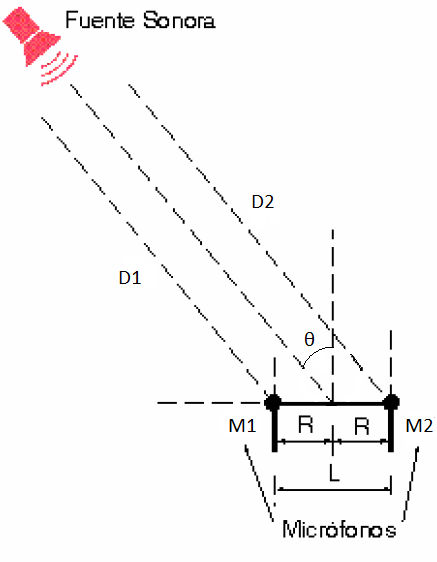
Luego se comparan ambas:

Si tomamos como certero el valor teórico de la frecuencia, calculamos el error absoluto y relativo respecto del experimental.

Deducimos que la diferencia entre ambas es debido a que experimentalmente se trabajó con aproximaciones del software y si bien el error podría ser menor, lo tomamos adecuado.

**Problema 2. Localización y seguimiento de una fuente sonora**

1. En este problema se presenta un par de micrófonos separados a una distancia L, los cuales pueden pivotear en el punto medio. Se les hace llegar una señal de sonido plana desde diferentes puntos y el sistema tiene que poder calcular su ángulo de incidencia para así poder rotar su eje y hacer un seguimiento de la señal.



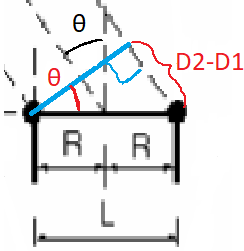


Figura 7. Esquema del modelo ampliado.

Figura 6. Esquema del modelo.

Como el frente de onda es plano se puede pensar que las señales D1 y D2 se mueven en líneas rectas paralelas y que comienzan un plano perpendicular a ambas. Por lo tanto, la diferencia de recorrido de ambas señales (por ende, el tiempo de retardo de la señal 1) se puede visualizar como en la figura 2 y se puede expresar como:

1. Ahora la única incógnita que nos resta encontrar es el tiempo de retardo para cada posición de la fuente sonora para así poder calcular su posición. Esto podemos realizarlo haciendo una correlación entre la señal recibida por el micrófono 1 y la recibida por el 2 ya que, al ser la misma señal x(n) pero solo retrasada o adelantada K unidades en el tiempo ( D1=x(n) y D2=x(n)), la autocorrelación presentará un máximo en dicho retardo.

Por lo tanto, si hacemos la correlación entre ambas señales de entrada y buscamos en qué punto es máximo (llamado “muestra” en el script) vamos a poder conseguir el valor de K (llamado delta\_t en el script). Hay que tener en cuenta que, como las señales D1 y D2 son finitas de longitud Y, la correlación nos va a dar un vector con l en el intervalo [-Y, Y] pero que Matlab lo representa como [0,2Y], por lo que se es necesario restar Y unidades a nuestra “muestra” para así poder calcular el Δt. Luego se divide la resta por la frecuencia a la cual se hizo el muestreo:

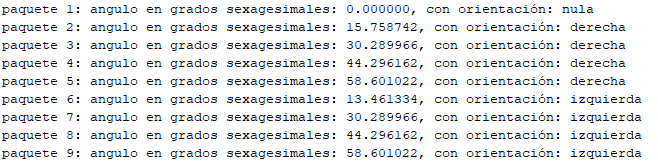
Con el tiempo de retardo calculado procedemos a calcular el ángulo con la fórmula presentada al principio. Hay que tener presente que si el Δt nos da negativo significa que la señal 1 llega después que la 2, por lo que el ángulo va a ser negativo, entonces la fuente sonora se encuentra a la derecha, mientras que si es positivo se encontraría a la izquierda. Por otro lado, si es 0°, la fuente sonora se encuentra en el centro, a igual distancia de ambos micrófonos.

1. Dicho ángulo theta está relacionado únicamente con la variación de tiempo que midamos, por lo que su resolución a su vez va a depender de la resolución con la cual podamos medirla. Como el Δt se calcula como y, como tanto “muestra” e Y son enteros, el valor absoluto del menor valor que podamos calcular sería 1/Fs. Ésta sería nuestra resolución para el Δt, por lo que la resolución para el ángulo sería:

Podemos observar que dicha resolución se mantiene constante siempre y cuando nuestra frecuencia de muestreo se mantenga constante, por lo cual no va a variar con la variación del ángulo relativo entre la señal y el eje de los micrófonos.

A continuación, se presenta el script de Matlab utilizado.

|  |
| --- |
| Script: **“TP1\_Ej2.m”** |
|  |

Y los resultados obtenidos para las señales de entrada dadas:

