

Universidad Nacional de Rosario
Facultad de Ciencias Exactas, Ingeniería y Agrimensura
Escuela de Ingeniería Electrónica



Sistemas y Señales I

Trabajo práctico N°1 : 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
------------------------------	------------

Introducción:

Para el problema 1 se presenta un circuito RLC que, como se dio en teoría de circuitos, se encuentra caracterizado por las relaciones V-A de sus componentes. Debido a que trabajaremos en tiempo discreto, se busca la ecuación en diferencias que caracteriza al sistema para poder resolverla mediante algún método de integración, en este caso, el de Euler. Es importante en estos términos tener en cuenta el paso de integración numérica a utilizar en la iteración ya que este puede causar divergencia si utiliza un paso mayor al adecuado.

Como resultado de la iteración se obtienen las respuestas temporales de carga y corriente del circuito. Se utilizan las mismas para encontrar la salida del sistema (Tensión en inductor) y así poder hallar la respuesta en frecuencia del sistema.

Para el problema 2 se presenta un par de micrófonos pivotantes que realizan un seguimiento de señales a través de la diferencia de tiempo que tendrá la señal en llegar a uno o al otro. La diferencia de tiempo de la señal, puede calcularse a partir de la correlación entre las señales recibidas por cada micrófono.

Luego, a partir de la diferencia de tiempo (retardo) entre la llegada de la señal a un micrófono y al otro, mediante recursos trigonométricos y físicos, se puede hallar en ángulo de la señal emisora respecto del eje medio entre los micrófonos.

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.

a) 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:

$$U_R(t) = i(t) \cdot R \quad U_L(t) = L \cdot \frac{di(t)}{dt} \quad i_c(t) = C \cdot \frac{du_c(t)}{dt}$$

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

$$\begin{aligned} U(t) &= U_R(t) + U_L(t) + U_C(t) \\ U(t) &= i(t) \cdot R + L \frac{di(t)}{dt} + \frac{q(t)}{C} \\ U(t) &= \frac{dq(t)}{dt} \cdot R + L \frac{d^2q(t)}{dt^2} + \frac{q(t)}{C} \quad (*) \end{aligned}$$

b) 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:

$$\begin{cases} x_1(t) = q(t) \\ x_2(t) = q'(t) = i(t) \end{cases}$$

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

$$x_1'(t) = x_2(t)$$

$$x_2'(t) = \frac{U(t) - \frac{x_1(t)}{C} - R x_2(t)}{L}$$

Utilizando el método de Euler encontramos las aproximaciones:

$$\begin{cases} x_1'(t) = x_2(t) \approx \frac{x_1((n+1)T) - x_1(nT)}{T} \\ x_2'(t) = \frac{U(t) - \frac{x_1(nT)}{C} - R x_2(nT)}{L} \approx \frac{x_2((n+1)T) - x_2(nT)}{T} \end{cases}$$

Despejando las variables de interés x_1 y x_2 :

$$\begin{cases} x_1((n+1)T) \approx x_2(nT)T + x_1(nT) \\ x_2((n+1)T) \approx \frac{T}{L} \left[U(nT) - \frac{x_1(nT)}{C} - R x_2(nT) \right] + x_2(nT) \end{cases}$$

Cambiando el índice en la discretización:

$$\begin{cases} x_1(nT) \approx x_2((n-1)T) + x_1((n-1)T) \\ x_2(nT) \approx \frac{T}{L} \left[U((n-1)T) - \frac{x_1(nT)}{C} - R x_2(nT) \right] + x_2((n-1)T) \end{cases}$$

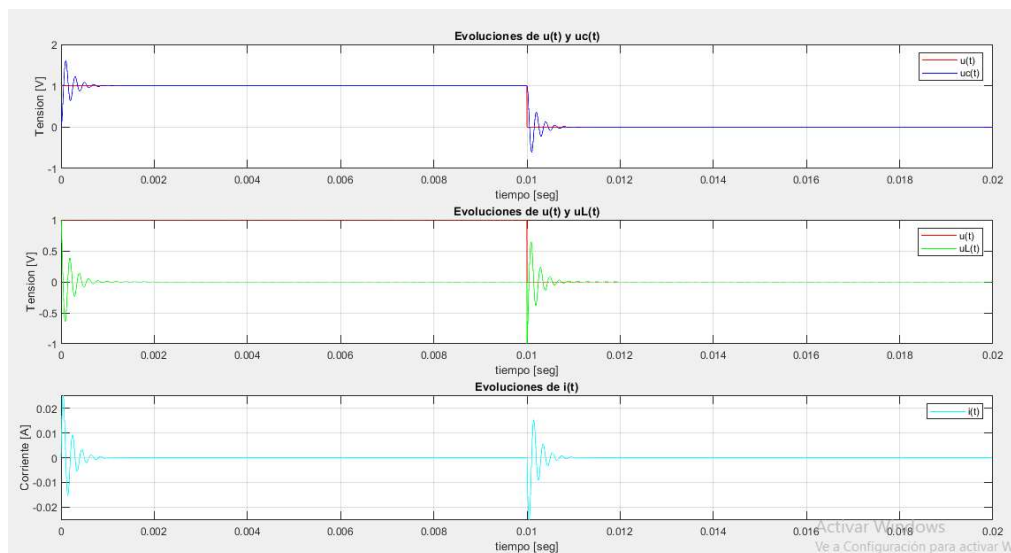
Finalmente se obtiene:

$$\begin{cases} x_1(nT) \approx x_2((n-1)T) + x_1((n-1)T) \\ x_2(nT) \approx \frac{T}{L} \left[U((n-1)T) - \frac{x_1((n-1)T)}{C} - R x_2((n-1)T) \right] + x_2((n-1)T) \end{cases}$$

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.

Considerando el paso de integración numérica T1:

Figura 3.

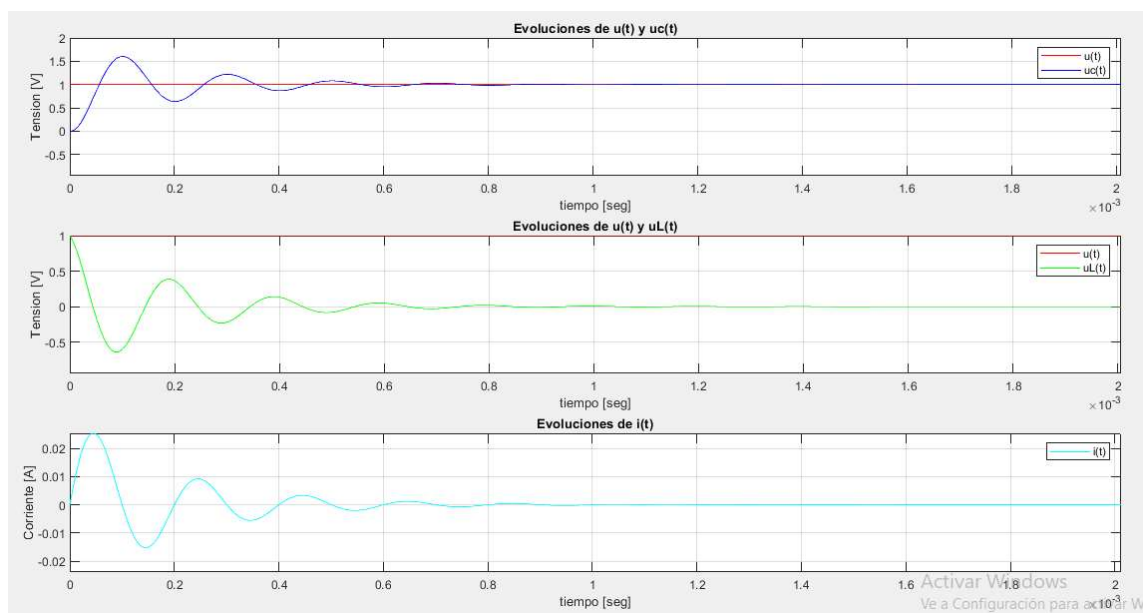


Evoluciones temporales del circuito para un tiempo total de integración de 2ms [$T1=1.9869e-06$]

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

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:



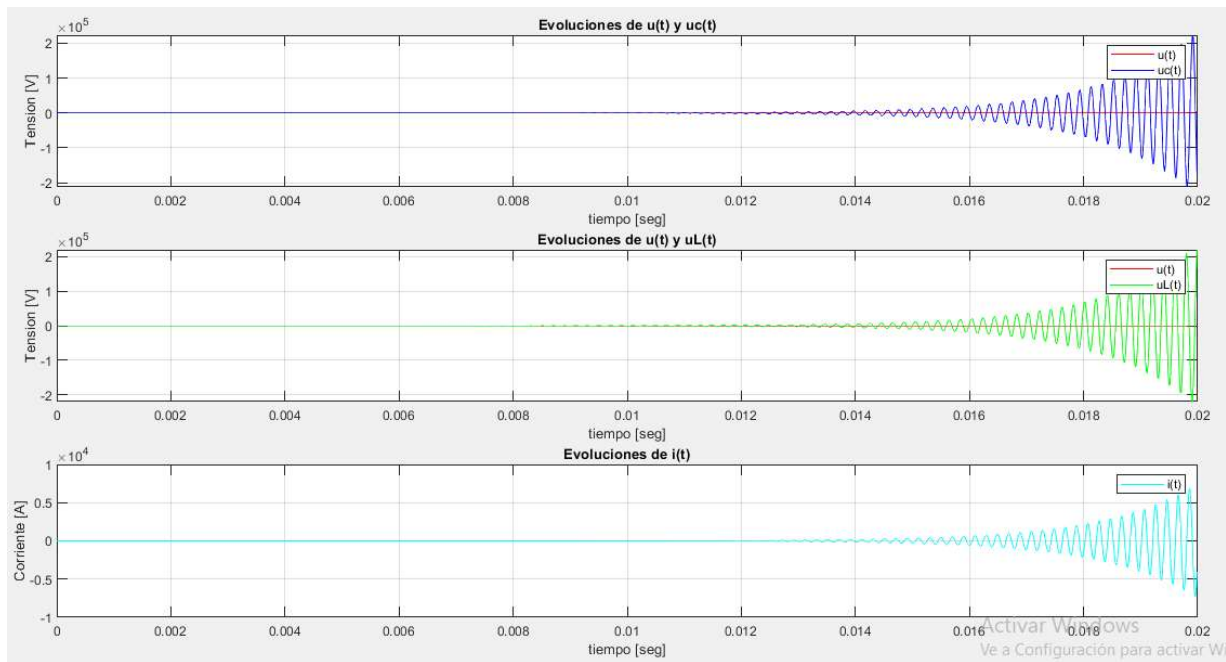


Figura 4. Evoluciones temporales del circuito para un tiempo total de integración de 20ms [$T_2 = 1.3246e-05$]

Observamos como en este caso el sistema se desestabiliza teniendo valores crecientes en su respuesta. Esta respuesta no representa las evoluciones temporales de nuestro circuito RLC, 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”

```

R = 12; C = 1e-6; L = 1e-3; %Valores de L, R y C dados.
x1(1) = 0; %condicion inicial de carga
x2(1) = 0; % condicion inicial de corriente
t10 = 10e-3; %tiempo pedido
valor = 100; %valor variable para paso de integración
T=(2*pi*sqrt(L*C)/(valor*sqrt(1-R^2*C/4*L))); %paso de integración
Amplitud = 1; %Amplitud del pulso
u = Amplitud * ones(1,round(t10/T)); %Entrada pulso de 10ms
for n = round(t10/T)+1:1:round(2*t10/T)
    u(n) = 0; %definimos 0 el vector desde 10ms a 20ms
end
for n = 1:1:round(2*t10/T)-1 %Método de aproximación de Euler
    x2(n+1) = u(n)* T/L + x2(n)*(1-R*T/L) - x1(n)*T/(C*L);
    x1(n+1) = T*x2(n) + x1(n);
end
tm = [0:1:round(2*t10/T)-1];
t = tm*T; % Escala del tiempo
uc = x1/C; %Tension en el capacitor
uL = u-uc-x2*R; %Tensión en inductor
subplot(3,1,1), plot(t,u,'r',t,uc,'b'), grid on;
xlabel('tiempo [seg]'); ylabel('Tension [V]'); title('Evoluciones de u(t) y uc(t)'); legend('u(t)','uc(t)');
subplot(3,1,2), plot(t,u,'r',t,uL,'g'), grid on;
xlabel('tiempo [seg]'); ylabel('Tension [V]'); title('Evoluciones de u(t) y uL(t)'); legend('u(t)','uL(t)');
subplot(3,1,3), plot(t,x2,'c'), grid on;
xlabel('tiempo [seg]'); ylabel('Corriente [A]'); title('Evoluciones de i(t)'); legend('i(t)');
    
```

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

A modo de presentación mostramos las respuestas de $U_c(t)$ y $U_L(t)$ para otros pasos de integración.

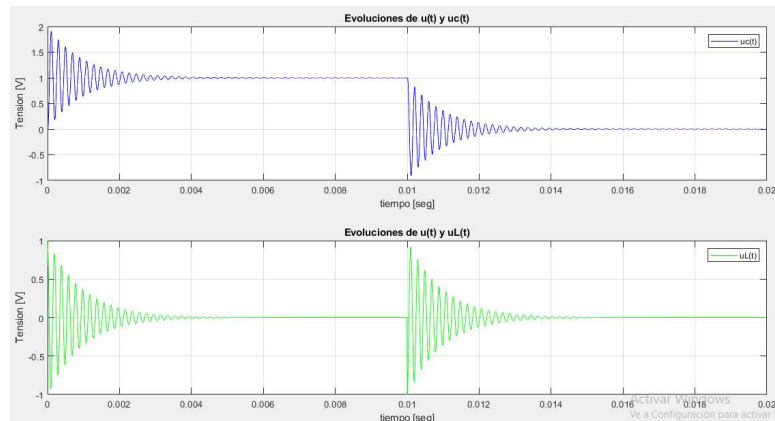


Figura 5a. Respuestas de tensiones para $T = 9.9346e-06$ ("valor" = 20)

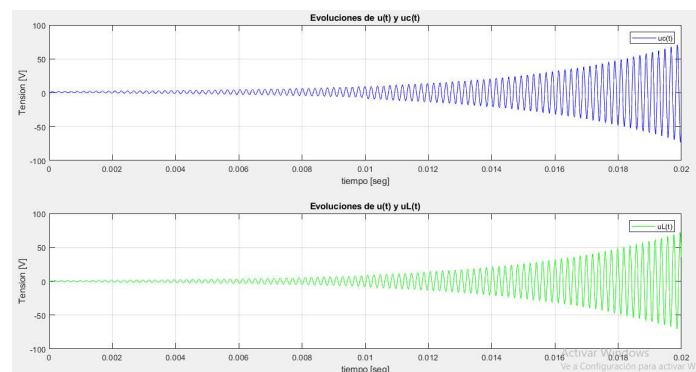


Figura 5b. Respuesta de tensiones para $T = 1.2418e-05$ ("valor" = 16)

*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. Se propone una entrada senoidal con frecuencia variable de la forma:

$$u(t) = \sin(2\pi Ft)$$

Se utiliza el mismo método de cálculo que para la entrada del escalón unitario, solo que con la nueva entrada definida. A partir de un vector de frecuencias dado, se aproximan las soluciones de $x_1(n)$ y $x_2(n)$ utilizando el método de Euler para cada frecuencia. Luego, a partir de la carga y la corriente, junto con las relaciones volt-ampere de los elementos dinámicos, calculamos la tensión en bornes del capacitor e inductor.

A partir de esto, se busca el máximo de la tensión en el inductor para cada frecuencia a fin de obtener la respuesta en frecuencia del circuito. Luego se aplica escala logarítmica a la función para poder graficarla.

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

Script: "TP1_Ej1parteII.m"

```

i = 1;
R = 12; C = 1e-6; L = 1e-3; %Valores de L, R y C dados.
x1(1) = 0; %condicion inicial de carga
x2(1) = 0; % condicion inicial de corriente
Tau = 2*L/R;
valor = 100; %valor variable para paso de integración
T=(2*pi*sqrt(L*C)/(valor*sqrt(1-R^2*C/4*L))); %paso de integración
F = [1000,2000,3000,4000,4500,4700,5000,5050,5300,5500,6000,8000,10000]; %Vector de frecuencias dadas
for n = 1:length(F)
    t = [0:T:20e-3];
    A = 1; %Amplitud del pulso
    u = A*sin(2*pi*F(n).*t); %Entrada senoidal definida
    for n = 2:length(t) %Método de aproximación de Euler
        x1(n) = x1(n-1) + T*x2(n-1);
        x2(n) = (1-T*R/L)*x2(n-1) - (T/(L*C))*x1(n-1) + (T/L)*u(n-1);
    end
    uL = u-(x1/C)-x2*R; %Tensión en inductor
    [fila, columna] = find(uL == max(uL(round(5*Tau/T):end)),1);
    Amp(i) = uL(columna);
    i = i+1;
end
RF = 20*log10(Amp); %Respuesta en frecuencia en db
semilogx(F,RF), grid on;
xlabel('Frecuencia [Rad/s]'); ylabel('Ganancia [dB]'); title('Respuesta en frecuencia de amplitud');
[f,c] = find(RF == max(RF),1); %Buscamos cuando el maximo coincide con RF
fteorica = 1./(2*3.14*sqrt(L*C)); %Frecuencia teórica máxima
fexperimental = F(c); %frecuencia a la cual se da el maximo segun nuestros cálculos
Comparacion = fteorica / fexperimental; %comparacion
    
```

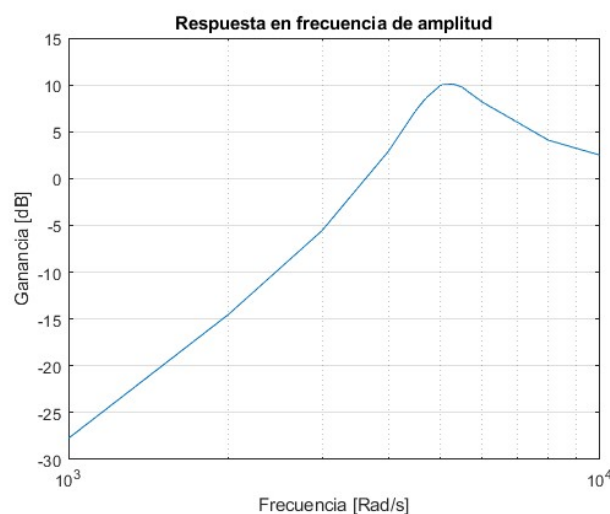


Figura 5a. Respuesta en frecuencia de amplitud

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 $F_{\text{experimental}}$ mientras que, al teórico, calculado con conocimientos de teoría de circuitos, lo llamamos $F_{\text{teórico}}$. Aquí podemos observar los resultados:

$$F_{\text{Experimental}} = 5300 \text{ Hz}$$
$$F_{\text{Teórica}} = \frac{1}{2\pi\sqrt{LC}} = 5035.5 \text{ Hz}$$

Luego se comparan ambas:

$$\frac{F_{\text{Teórica}}}{F_{\text{Experimental}}} = 0,9501$$

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

$$\varepsilon_{\text{abs}} = |F_{\text{Teórica}} - F_{\text{Experimental}}| = 264.5 \text{ Hz}$$

$$\varepsilon_{\text{relativo}} = \frac{264.5 \text{ Hz}}{5035.5 \text{ Hz}} = 0.0525 \rightarrow \text{Error relativo porcentual del 5.2\%}$$

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

- a. 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.

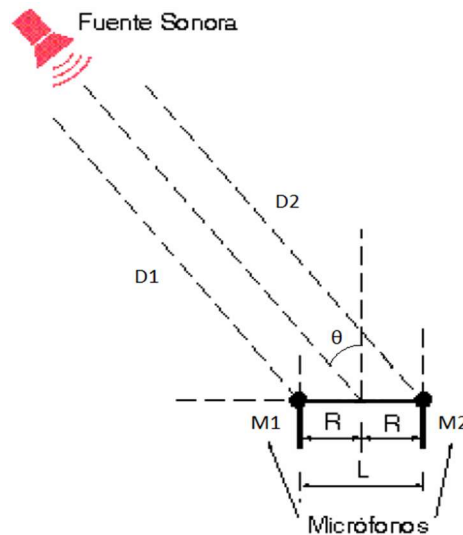


Figura 6. Esquema del modelo.

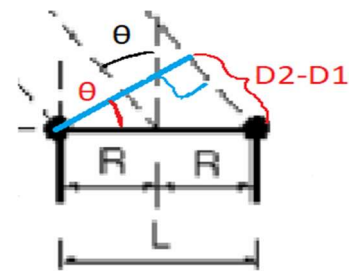


Figura 7. Esquema del modelo ampliado.

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:

$$\sin(\theta) = \frac{D2 - D1}{L} = \frac{c * (t_2 - t_1)}{L} = \frac{c * (\text{tiempo de retardo})}{L}$$

- b. 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 $D2=x(n \pm K)$), la autocorrelación presentará un máximo en dicho retardo.

$$r_{D1D2}(l) = \sum_{n=-\infty}^{\infty} x(n \pm K + l) * x(n) = r_{xx}(l \pm K)$$

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 Δ_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:

$$\Delta t = \frac{\text{muestra} - Y}{F_s}$$

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.

$$\theta = \arcsen\left(\frac{c * \Delta t}{L}\right)$$

- c. 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 $\Delta t = \frac{\text{muestra} - Y}{F_s}$ y, como tanto “muestra” e Y son enteros, el valor absoluto del menor valor que podamos calcular sería $1/F_s$. Ésta sería nuestra resolución para el Δt , por lo que la resolución para el ángulo sería:

$$\theta = \arcsen\left(\frac{c}{L * F_s}\right)$$

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"

```
L = 0.2; %distancia entre microfonos
velocidad_sonido = 342.2; %velocidad del sonido dada
n = [0:88199.9:10];
%Leemos los datos de cada audio y los almacenamos en una matriz
[y1, Fs1] = audioread('datos_1.wav');
[y2, Fs2] = audioread('datos_2.wav');
[y3, Fs3] = audioread('datos_3.wav');
[y4, Fs4] = audioread('datos_4.wav');
[y5, Fs5] = audioread('datos_5.wav');
[y6, Fs6] = audioread('datos_6.wav');
[y7, Fs7] = audioread('datos_7.wav');
[y8, Fs8] = audioread('datos_8.wav');
[y9, Fs9] = audioread('datos_9.wav');
Y = [y1,y2,y3,y4,y5,y6,y7,y8,y9];
Fs = [Fs1, Fs2, Fs3, Fs4, Fs5, Fs6, Fs7, Fs8, Fs9]; %vector frec de muestreo
for i = 1:1:9
    rxy = xcorr(y(:,2*i-1),y(:,2*i)); %correlacion cruzada
    muestra = find(rxy == max(rxy)); %Busca cuando la correlacion cruzada sea máxima.
    delta_t(i) = (muestra-length(y(:,2*i-1))) / Fs(i); %determina el delta_t para cada señal.
    angrad(i) = asin (((velocidad_sonido)*(delta_t(i)))/(L)); %vector de angulos para cada señal.
    ang(i) = (angrad(i) *180) / pi; %expresion de el angulo en grados.
    if (ang(i)<0)
        orientacion = 'izquierda';
    end
    if (ang(i)>0)
        orientacion = 'derecha'; %Condiciones para determinar la orientacion del ángulo.
    end
    if (ang(i)==0)
        orientacion = 'nula';
    end
    fprintf ('paquete %d: angulo en grados sexagesimales: %f, con orientación: %s \n',i,abs(ang(i)),orientacion)
end
%El ángulo mínimo (resolución) corresponde a una muestra y se
% encuentra cuando: [muestra-length(y(:,2*i-1))] = 1 y así
% Ángulo = asin(velocidad_sonido/L*Fs)
angmin = asin(velocidad_sonido/(L*Fs(1))) %angulo minimo en radianes
angminsec = (angmin * 180) / pi
% La resolución es constante debido a que depende de parámetros invariantes
% con el desplazamiento angular.
```

Y los resultados obtenidos para las señales de entrada dadas:

```
paquete 1: angulo en grados sexagesimales: 0.000000, con orientación: nula
paquete 2: angulo en grados sexagesimales: 15.758742, con orientación: derecha
paquete 3: angulo en grados sexagesimales: 30.289966, con orientación: derecha
paquete 4: angulo en grados sexagesimales: 44.296162, con orientación: derecha
paquete 5: angulo en grados sexagesimales: 58.601022, con orientación: derecha
paquete 6: angulo en grados sexagesimales: 13.461334, con orientación: izquierda
paquete 7: angulo en grados sexagesimales: 30.289966, con orientación: izquierda
paquete 8: angulo en grados sexagesimales: 44.296162, con orientación: izquierda
paquete 9: angulo en grados sexagesimales: 58.601022, con orientación: izquierda
```

```
El angulo mínimo en grados sexagesimales es: 2.223530
```

En dichos resultados observamos la orientación de la señal respecto de los micrófonos. Cuando el ángulo es positivo se indica que la señal se encuentra a la derecha de los micrófonos mientras que cuando es negativa se encuentra a la izquierda.

Por otro lado, obtuvimos un ángulo mínimo razonable, ya que el mismo representa el ángulo mínimo que percibe el arreglo de micrófonos dada su frecuencia de muestreo.