



UNIVERSIDAD NACIONAL
AUTÓNOMA DE
MÉXICO

UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

PROGRAMA DE MAESTRÍA Y DOCTORADO EN
INGENIERÍA

**Medición del aislamiento sonoro de materiales.
Comparación con varias técnicas.**

T E S I S

Que para obtener el título de:
MAESTRO EN INGENIERÍA ELÉCTRICA
INSTRUMENTACIÓN ACÚSTICA

P R E S E N T A

ODÍN MORENO AMEZCUA

TUTOR

M. EN C. SANTIAGO JESÚS PÉREZ RUIZ



2007



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

Jurado asignado

PRESIDENTE: DRA. ROSALBA CASTAÑEDA GUZMÁN.

SECRETARIO: DR. PABLO LUIS RENDÓN GARRIDO.

VOCAL: M. EN C. SANTIAGO JESÚS PÉREZ RUIZ.

1^{er} SUPLENTE: DR. ANDRÉS ESTEBAN PÉREZ MATZUMOTO.

2° SUPLENTE: M. EN I. ANTONIO PÉREZ LÓPEZ.

LUGAR DONDE SE REALIZÓ LA TESIS:

**CENTRO DE CIENCIAS APLICADAS Y DESARROLLO TECNOLÓGICO, UNAM, CIUDAD
UNIVERSITARIA.**

CIUDAD DE MÉXICO, DISTRITO FEDERAL, MÉXICO.

TUTOR DE TESIS:

M. EN C. SANTIAGO JESÚS PÉREZ RUIZ.



FIRMA

Agradecimientos

A mis papás Edith y Gonzalo por su infinito cariño, apoyo, orientación, comprensión y motivación incondicionales que me han dado a lo largo de toda mi vida tanto en mis éxitos como fracasos y que han hecho de mi una persona de bien.

A mis hermanas Kóchill y Cillali que siempre con su ternura, cariño, compañía y consejo he vivido con ellas grandes momentos en todas las etapas de nuestras vidas.

A mi Tutor el M. en C. Santiago Jesús Pérez Ruiz por su apoyo, orientación y paciencia en el desarrollo de esta importante etapa académica de mi vida.

A mi novia Perlita que siempre ha estado a mi lado brindándome su cariño, apoyo y confianza que me llena de fortaleza para seguir adelante en nuestras vidas y proyectos.

A mis profesores sinodales la atención, disponibilidad y orientación para conmigo en la realización y mejora de este trabajo.

A mis profesores de la maestría: M. Santiago, M. Antonio, Dra. Rosalba, Dr. Pablo, Dr. Felipe, Dra. Graciela y M. Ricardo, que siempre me dieron su apoyo y orientación para la realización de mis estudios.

Agradecimientos

Al CCADET y su Laboratorio de Acústica por las facilidades en la adecuación de las instalaciones y montaje de materiales para su evaluación sonora.

Al CONACYT y la DGEP por la beca otorgada para la realización de estos estudios.

A CEMEX y al CENAM por la confianza y apoyo en el aporte de materiales e instrumental para la realización de mediciones y comparación de resultados que han nutrido positivamente este trabajo.

A mis amigos de la maestría: Héctor, Alejandro, Vinicio, Vicente, César, Laura, Violeta, Javier y Felipe, con los que he pasado momentos muy agradables dentro y fuera de nuestros estudios.

A mis amigos del CENAM: Andrés, Noé y Adolfo que me orientaron en cada una de sus visitas al CCADET en cuanto a su experiencia para la realización de mis mediciones.

Al Ing. Anastacio Villalobos Castrejón que me ha compartido su gran experiencia profesional en el terreno de la acústica y la docencia, brindándome su apoyo y consejo para mi desarrollo profesional.

Gracias

Odín

ÍNDICE

| | Pág. |
|--|-----------|
| Jurado asignado..... | i |
| Agradecimientos..... | ii |
| Índice..... | iv |
| INTRODUCCIÓN..... | 1 |
| 1.1 Importancia de la medición del aislamiento sonoro..... | 1 |
| 1.2 La necesidad de métodos alternativos de medición..... | 2 |
| ESTRUCTURACIÓN..... | 3 |
| Justificación..... | 3 |
| Objetivo de la tesis..... | 3 |
| Programación de capítulos..... | 4 |
| I. FUNDAMENTOS TEÓRICOS DEL AISLAMIENTO SONORO..... | 5 |
| 1.1. Concepto de ondas mecánicas..... | 5 |
| 1.2. Concepto de aislamiento..... | 5 |
| 1.3. Incidencia de ondas mecánicas sobre materiales..... | 6 |
| 1.4. Transmisión de ondas mecánicas en materiales..... | 7 |
| 1.5. Expresión del aislamiento sonoro..... | 11 |
| 1.6. Categoría de transmisión sonora (Sound Transmission Class, STC)..... | 13 |
| 1.7. Ley de masas y efecto de coincidencia..... | 15 |
| 1.8. Normativa existente..... | 20 |
| II. TÉCNICAS DE MEDICIÓN DEL AISLAMIENTO SONORO..... | 21 |
| 2.1. Técnicas de laboratorio..... | 21 |
| 2.1.1. Cámaras de transmisión..... | 21 |
| 2.1.2. Técnica por medio de Nivel de Presión Sonora..... | 22 |
| 2.2. Técnicas “ <i>in situ</i> ” alternativas..... | 22 |
| 2.2.1. Técnica Cepstrum..... | 23 |
| 2.2.2. Técnica MLS..... | 25 |
| III. DESARROLLO EXPERIMENTAL DE MEDICIÓN CON LA TÉCNICA DE NIVELES DE PRESIÓN SONORA NPS..... | 28 |
| 3.1. Técnica NPS..... | 28 |
| 3.2. Planteamiento del problema..... | 28 |
| 3.3. Caracterización de la Cámara de transmisión..... | 29 |
| 3.4. Programación en la PC con Matlab 6.5..... | 30 |
| 3.5. Ajuste de instrumental de medición..... | 32 |
| 3.6. Procedimiento de medición..... | 35 |
| 3.7. Resultados..... | 42 |

ÍNDICE (cont)

| | Pág. |
|--|-----------|
| IV. DESARROLLO EXPERIMENTAL DE MEDICIÓN CON LA TÉCNICA DE CEPSTRUM..... | 52 |
| 4.1. Técnica Cepstrum..... | 52 |
| 4.2. Planteamiento del problema..... | 52 |
| 4.3. Programación en la PC con Matlab 6.5..... | 58 |
| 4.4. Ajuste de instrumental de medición..... | 59 |
| 4.5. Procedimiento de medición..... | 61 |
| 4.6. Resultados..... | 62 |
| V. DESARROLLO EXPERIMENTAL DE MEDICIÓN CON LA TÉCNICA DE SECUENCIAS DE MÁXIMA LONGITUD MLS..... | 63 |
| 5.1. Técnica MLS..... | 63 |
| 5.2. Planteamiento del problema..... | 63 |
| 5.3. Programación en la PC con Matlab 6.5..... | 64 |
| 5.4. Ajuste de instrumental de medición..... | 64 |
| 5.5. Procedimiento de medición..... | 64 |
| 5.6. Resultados..... | 66 |
| VI. COMPARACIÓN DE TÉCNICAS DE MEDICIÓN DE AISLAMIENTO SONORO..... | 68 |
| 6.1. Gráficas comparativas..... | 68 |
| CONCLUSIONES..... | 77 |
| BIBLIOGRAFÍA..... | 78 |
| ANEXOS Programas en Matlab (ver CD)..... | 80 |

INTRODUCCIÓN

Una de las características de los materiales empleados en la construcción es la capacidad para evitar la transmisión de energía sonora (Aislamiento sonoro). Esta propiedad es esencial para el confort en el interior de una vivienda. La medición normalizada de esta característica, método de Presión Sonora, requiere de una cámara especializada (cámara de transmisión), que afortunadamente es parte de la infraestructura del *CCADET*¹ perteneciente a la UNAM. Existen otros métodos de medición *in situ*² como el de Intensidad acústica, sin embargo, su instrumentación es elaborada y toma mucho tiempo realizarla. Por esta razón se exploran otros métodos alternativos.

El presente trabajo muestra un método de medición establecido y otros dos métodos *in situ* para medir aislamiento sin la necesidad de utilizar cámara de transmisión.

I.1. Importancia de la medición del aislamiento sonoro.

Desde que el hombre ha interactuado con la naturaleza, siempre ha estado sumergido en un sinfín de sonidos que produce su entorno, que van desde su comunidad hasta los producidos por la naturaleza.

Para comunicarse con sus semejantes en la vida diaria, siempre se ha percatado que para transmitir sus mensajes sonoros, éste debe sobresalir de todos los demás sonidos que se encuentran de lo contrario no se lograría la comunicación.

Dentro de las fuentes de ruido³ naturales se encuentran: la lluvia, los relámpagos y ruidos de animales que hacen que se tenga la necesidad de emitir un mensaje con mayor intensidad lo cual es incómodo; de ahí la necesidad de evitar que esos ruidos no lleguen a nosotros, dando lugar a la necesidad de aislar un ambiente de sonidos indeseables que perturban la actividad humana.

La manera más sencilla y primitiva de aislar acústicamente un ambiente seguramente ha sido colocar barreras sólidas y gruesas como rocas o muros las cuales impiden el paso del sonido hasta cierto grado, y esta solución es la que de manera empírica se ha manejado por mucho tiempo.

Si se desea aislar un recinto de los sonidos del exterior es necesario establecer en qué proporción se desea el no percibir lo que sucede afuera. Esto es importante ya que a medida que se requiera más aislamiento, mayor será la masa y sofisticación de los materiales y se debe procurar que se garantice las condiciones deseadas sin llegar a diseños que sobrepasen inútilmente las necesidades.

¹ CCADET Centro de Ciencias Aplicadas y Desarrollo Tecnológico.

² In situ se refiere a condiciones de medida dentro de un recinto no especializado bajo condiciones conocidas y controladas, que pueden ser comparadas con técnicas de laboratorio.

³ Ruido Señal no deseada que interfiere negativamente a los propósitos del envío y recepción de información.

Por lo anterior, se hace necesaria una forma de establecer la capacidad que tienen los materiales de evitar el paso del sonido y así determinar el más adecuado a la aplicación.

Puesto que los materiales existentes tienen propiedades universales se tienen que definir parámetros normalizados en base a sus propiedades mecánicas que permitan interpretar su comportamiento en cualquier lugar que se requieran. La forma básica de indicar el aislamiento sonoro es saber el nivel de ruido que se tiene en el exterior y cuanto logró pasar por la barrera. Haciendo la diferencia aritmética de estos dos niveles establece la capacidad de aislamiento independientemente del nivel que se utilice como se aprecia en la figura 1.

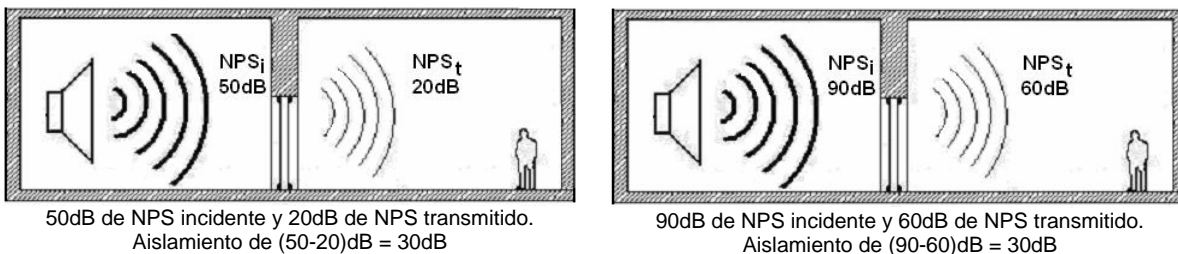


Figura 1. Visualización de aislamiento sonoro en función de los niveles incidente y transmitido a través de un muro en común que separa al recinto emisor del recinto receptor.

De aquí se puede ver que el muro que divide a estos dos recintos tiene la capacidad de aislar en 30dB el sonido que incide en él, por lo que se puede decir que en aplicaciones que requieran de 30dB de aislamiento es posible usar este material. Desde luego existe una normatividad para describir esta propiedad que se mencionará más adelante.

Con esta idea se puede catalogar un material que sea utilizado para los muros de una construcción, sabiendo de antemano esta propiedad mecánica.

1.2. La necesidad de métodos alternativos de medición.

Los métodos normalizados para medir el aislamiento sonoro de los materiales, permiten establecer las condiciones necesarias para obtener una lectura tal que, solo tome en cuenta el comportamiento del material minimizando cualquier perturbación que contamine los resultados, como son los ruidos ambientales y los propios, producidos y transmitidos por el experimento, debido a los muros en donde se encuentra (transmisión por flanqueo).

Para lograr lo anterior, es necesario contar con una cámara de transmisión, que es un conjunto de instalaciones especializadas, que evita el paso de ruidos externos y permite medir exclusivamente el sonido que pasa por la muestra, minimizando la posibilidad de tomar en cuenta sonidos que hayan pasado por rutas indirectas. Este tipo de cámara es muy costosa, pero afortunadamente el CCADET cuenta con una. Posteriormente se debe montar el material a medir lo cual resulta muy laborioso y lleva de dos a tres días para que esté listo con las condiciones que indican las normas.

El conocer la capacidad de aislar de un material es muy útil, pero es muy laborioso y costoso llegar a saber con certeza el comportamiento del mismo, por lo que surge la necesidad de instrumentar métodos alternativos que permitan llegar al mismo resultado, sin la necesidad de contar con instalaciones como esas y que requiera solamente equipo de laboratorio básico que, inclusive, se pueda transportar al lugar donde se quiera realizar la medición. Para lograrlo que propone utilizar herramientas de cómputo, que reciban las señales incidente y transmitida para procesarse y así obtener un resultado de forma más ágil.

Desde luego se tienen que hacer varias consideraciones para poder establecer estos métodos como válidos y compararse con las técnicas establecidas.

ESTRUCTURACIÓN.

El presente trabajo aborda el tema de la medición del aislamiento sonoro indicando: los principios en los que se basa este fenómeno, la manera en que se mide, la técnica de medición normalizada y las alternativas de medición para este propósito.

I.3. Justificación.

Es necesario medir la capacidad de aislamiento de los materiales para permitir el confort acústico adecuado en función de la actividad que se quiera realizar, ya que hay algunas que requerirán un mayor o menor grado de esta propiedad.

Por lo anterior se han llevado a cabo diferentes técnicas y procedimientos estandarizados para cuantificar, lo más fielmente posible, el aislamiento de sonido bajo condiciones controladas. Para alcanzar este objetivo, se comparan mediciones en cámara contra condiciones “*in situ*” con instrumental básico de laboratorio y una computadora personal que devuelvan resultados útiles y que sean comparables con los métodos que hasta ahora han sido establecidos.

Al aplicar técnicas alternativas de medición, se podrá determinar el aislamiento sonoro de materiales en un recinto convencional semireverberante y no especializado, inclusive se podrá transportar el instrumental al lugar físico en donde se requiera saber cuánto aísla un muro.

I.4. Objetivo de la tesis.

El objetivo principal de esta Tesis consiste en comparar diferentes técnicas de medición de aislamiento sonoro como son las desarrolladas bajo condiciones de laboratorio (Técnica por *NPS*) y otros métodos como son los “*in situ*” (*Cepstrum* y *MLS*), utilizando instrumental básico como micrófonos, amplificador, altavoces, y una computadora personal programada para adquirir y procesar los datos, indicando el funcionamiento de las técnicas para la su adecuada aplicación.

I.5. Programación de capítulos.

La Tesis se divide en seis capítulos que desarrollan los fundamentos teóricos, las técnicas de medición, el desarrollo experimental, los resultados comparativos y conclusiones.

En el primer capítulo se habla de Los fundamentos del aislamiento sonoro que permiten dar a conocer el análisis de la incidencia sonora y transmisión en los materiales, la manera en que éste opera, cómo se expresa y la normativa para hacerlo.

En el segundo capítulo se dan a conocer las técnicas que se desarrollarán en el presente trabajo como son las de laboratorio y las "*in situ*".

El tercer capítulo muestra el desarrollo experimental que se realizó en la técnica *NPS*, en el cuarto con la técnica *Cepstrum* y en el quinto con *MLS*.

En estos últimos tres se indica cómo se plantea el problema, la programación, los ajustes del instrumental, procedimiento de medida y los resultados obtenidos.

En el sexto capítulo se muestra la comparación de resultados obtenidos con las diferentes técnicas de medición con gráficas de aislamiento sonoro.

Más adelante se dan las conclusiones generales del trabajo, en donde se dan a conocer observaciones, errores de medición y propuestas de mejora.

Finalmente se presentan las referencias bibliográficas utilizadas para este trabajo de investigación.

I. FUNDAMENTOS TEÓRICOS DEL AISLAMIENTO SONORO

1.1. Concepto de ondas mecánicas.

Son señales que viajan a través de un medio elástico como pueden ser los gases, líquidos o sólidos y pueden perturbar el medio en el que viajan experimentando una resistencia a su paso. En el caso particular de los sólidos y gases (que son los medios que se abordan en este trabajo), las moléculas tienen una fuerza elástica restauradora similar a la que tiene un resorte cuando se le estira y suelta; esta fuerza, acoplada con la inercia del sistema, permite que la materia participe en vibraciones oscilatorias.

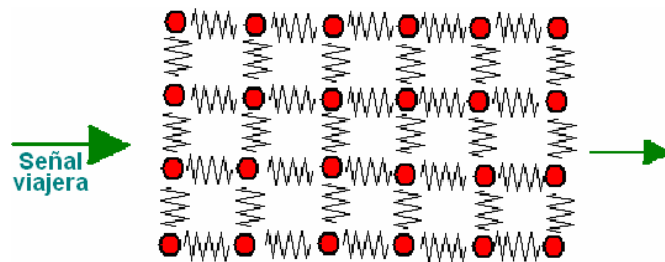


Figura 1.1 Moléculas de un medio elástico por el cual viajará una señal y será modificada en amplitud a su paso debido a la impedancia sonora.

Para el caso de este trabajo, las ondas cuya frecuencia se encuentre dentro del ancho de banda audible, que va de 20Hz a 20KHz, se les llama de ondas sonoras.

1.2. Concepto de aislamiento.

Cuando las ondas sonoras inciden en un obstáculo, las presiones que actúan sobre él hacen que éste vibre. Una parte de la energía vibratoria transportada por las ondas se transmite a través del obstáculo y pone en movimiento el aire situado del otro lado, generando sonido. Parte de la energía es absorbida, otra reflejada y otra transmitida. El aislamiento, expresado en función de distintas magnitudes físicas, informará sobre la proporción de la energía transmitida respecto a la incidente.

El aislamiento sonoro es la propiedad mecánica que tienen los materiales de evitar en cierta medida el paso de ondas acústicas a través de él.

La energía del sonido que incide (E_i) se descompone en la energía reflejada al medio emisor (E_r) y la energía absorbida (E_a), es decir, la no reflejada se distribuye en energía disipada en el material (E_d) y energía transmitida al medio receptor (E_t).

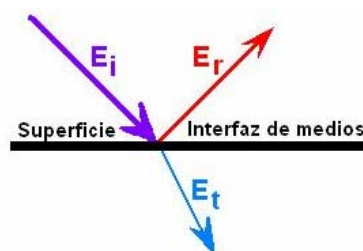


Figura 1.2. Onda incidente reflejada y transmitida en una interfase de dos medios.

1.3. Incidencia de ondas sonoras sobre materiales.

Cuando una señal sonora viaja en un medio elástico e incide en la frontera con otro que tiene diferentes cualidades en su densidad, velocidad de propagación e impedancia acústica¹ $r=\rho c$, se experimenta una modificación en la propagación de la señal debido al cambio de medio a otro.

La señal original incidente tiene una amplitud compleja de presión P_i que se verá afectada de manera considerable cuando incida en la frontera de los dos medios. Una parte de la señal será transmitida y otra parte será reflejada. Para cuantificar estas proporciones se pueden definir los coeficientes de transmisión de presión T y de reflexión de presión R [11].

$$T = P_t/P_i \quad (1-1)$$

$$R = P_r/P_i \quad (1-2)$$

Donde:

P_t es la amplitud compleja de presión transmitida.

P_r es la amplitud compleja de presión reflejada.

P_i es la amplitud compleja de presión incidente.

Además se ha de cumplir la condición de continuidad de presión en la frontera así que se tiene:

$$1 + R = T \quad (1-3)$$

Considerando que la señal reflejada va en sentido contrario a la señal transmitida. Viéndolo de otra forma, la expresión se puede tener así:

$$1 + (-R) = T \quad (1-4)$$

Lo que significa que al coeficiente original se le resta la parte reflejada y se obtiene la porción que se transmite. Desde luego existe la posibilidad de diferentes formas en que la señal sonora incida en la frontera.

Los modos básicos son: incidencia normal² y oblicua³. En estos casos siempre se tendrá un ángulo de incidencia que será igual al ángulo de reflexión respecto a la normal en el primer medio, y un ángulo de transmisión de la señal en el segundo medio, que dependerá de las propiedades antes mencionadas de los dos medios.

¹ Cuando se habla en términos de acústica se hace referencia a las vibraciones en medios elásticos dentro y fuera del intervalo audible de frecuencias, mientras que, al hablar de fenómenos sonoros, solo se toman en cuenta las frecuencias que el oído humano puede percibir.

² La incidencia normal se considera como un caso particular de la incidencia oblicua cuando $\theta_i=0^\circ$.

³ Existe un tercer caso que es el de incidencia aleatoria, en donde se da la combinación de incidencia normal y oblicua de manera simultánea.

De acuerdo al desarrollo hecho por Kinsler [11] el valor de R (llamado coeficiente de reflexión de Rayleigh) para incidencia oblicua, se calcula con la fórmula 1-5.

$$R = \frac{(r_2/r_1) - (\cos\theta_t/\cos\theta_i)}{(r_2/r_1) + (\cos\theta_t/\cos\theta_i)} \quad (1-5)$$

Donde:

θ_i es el ángulo de incidencia

θ_t es el ángulo de transmisión.

r_1 es la impedancia acústica en el medio 1.

r_2 es la impedancia acústica en el medio 2.

Y el coeficiente de reflexión de potencia es:

$$R_\pi = |R|^2 \quad (1-6)$$

Aplicando la ley de la conservación de energía, la potencia de la onda incidente debe repartirse entre la onda reflejada y la transmitida de tal manera que

$$R_\pi + T_\pi = 1 \quad (1-7)$$

Donde:

R_π es el coeficiente de reflexión de potencia.

T_π es el coeficiente de transmisión de potencia.

De lo anterior se tiene que T_π en función de las propiedades de los medios y los ángulos involucrados, queda finalmente como:

$$T_\pi = \frac{4(r_2/r_1)\cos\theta_t/\cos\theta_i}{(r_2/r_1 + \cos\theta_t/\cos\theta_i)^2} \quad \theta_t \text{ real} \quad (1-8)$$

De esta expresión se puede ver que existen muchas combinaciones de ρ y c de las cuales se analizará y calcularán valores, para el caso de un incidencia oblicua de ondas en una frontera de gas-sólido (aire-muro). Véase la figuras 1.3 -1.5.

1.4. Transmisión de ondas sonoras en materiales.

Para estudiar la manera en la que se transmiten las ondas de un medio a otro existen varias formas de incidencia en una frontera de dos medios, por lo que solamente se desarrollarán los casos prácticos que se necesitan conocer como es un material sólido en contacto con la atmósfera. Para entender este fenómeno se tomará en cuenta la velocidad del sonido c y la impedancia del medio r en los dos siguientes casos.

A) Variación de coeficiente de reflexión y transmisión bajo incidencia oblicua⁴ con $c_2 > c_1$ y $r_2 > r_1$.

Este es el caso que se da cuando una onda que viaja por el aire incide en una superficie sólida haciendo que la señal pase de un medio de menor densidad a uno de mayor densidad, así que la velocidad aumentará y la impedancia acústica también.

De la *ley de Snell* para ondas que pasan de un medio con velocidad de propagación c_1 a otro con velocidad de propagación c_2 , la variación del ángulo de transmisión relacionado con el de incidencia está gobernada por la siguiente ecuación.

$$\frac{\text{sen}\theta_i}{c_1} = \frac{\text{sen}\theta_t}{c_2} \quad (1-9)$$

Donde:

θ_i es el ángulo de incidencia

θ_t es el ángulo de transmisión.

c_1 es la velocidad de propagación de las ondas en el medio 1.

c_2 es la velocidad de propagación de las ondas en el medio 2.

Cuando $c_1 < c_2$, el haz transmitido se aleja de la normal, de tal manera que va llegar un punto en el que el ángulo del haz transmitido llegue a ser de 90° con su correspondiente ángulo de incidencia, cuando esto sucede, existe un ángulo crítico de incidencia θ_c que se da cuando $\theta_t = 90^\circ$ con la fórmula 1-10.

$$\text{sen}\theta_c = \frac{c_1}{c_2} \quad (1-10)$$

Para mostrar cómo se comporta la señal que incide en un medio, se presenta un arreglo de dos medios con sus respectivos valores de c y r de tal manera que se aprecie los comportamientos de las ondas bajo diferentes ángulos de incidencia como se puede ver en las figuras 1.3 a 1.5.

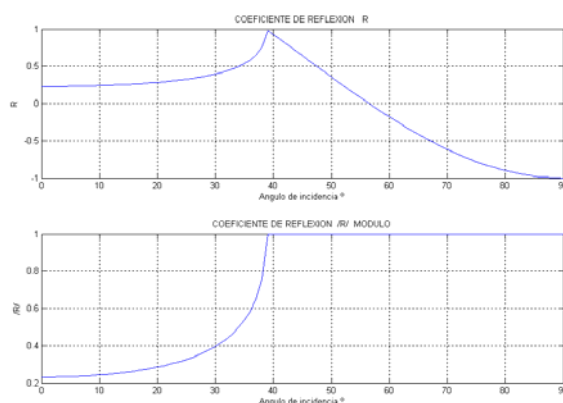


Figura 1.3. Comportamiento del coeficiente de reflexión.

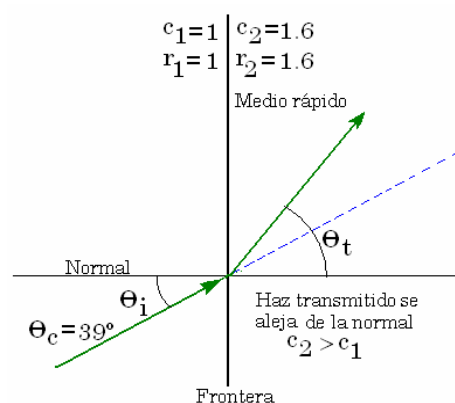
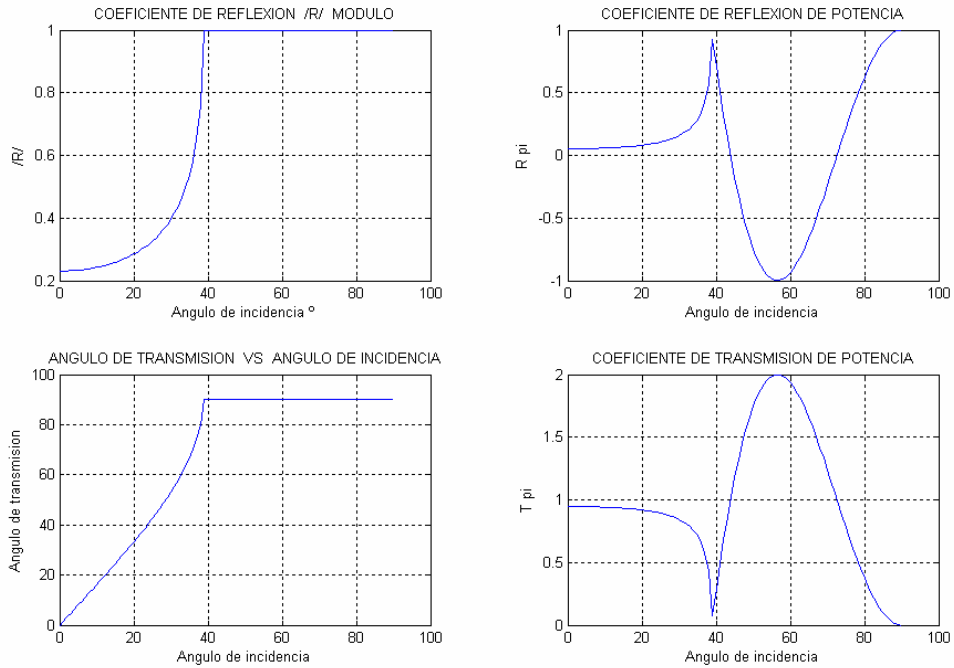


Figura 1.4. Incidencia oblicua en un medio rápido y de mayor impedancia sonora.
 $\theta_c = 39^\circ \rightarrow \theta_t \rightarrow 90^\circ$

⁴ El caso de incidencia normal se da particularmente cuando el ángulo correspondiente es igual a cero.

Bajo estas condiciones de r y c , siempre se tiene $\theta_t > \theta_i$ por lo que el haz transmitido siempre se aleja de la normal. A medida que θ_i aumenta se acerca al ángulo crítico $\theta_c = 39^\circ$ (calculado de la ecuación 1-6) en el que $\theta_t \rightarrow 90^\circ$, a partir de este punto la onda transmitida propaga paralelamente a la frontera y tiene una amplitud que decae perpendicularmente a la dirección de propagación; θ_t es puramente imaginario, la onda incidente es totalmente reflejada y no hay transmisión, como se puede ver en las figuras 1.5.



Figuras 1.5. Visualización del comportamiento de la reflexión, la transmisión y su ángulo correspondiente en función de los ángulos de incidencia bajo las condiciones $c_2 > c_1$ y $r_2 > r_1$.

B) Variación de coeficiente de reflexión y transmisión bajo incidencia oblicua con $c_2 < c_1$ y $r_2 < r_1$.

Ahora en este es el caso sucede lo contrario de lo anterior cuando la onda sale del material para salir nuevamente a aire. De esta manera, la señal pasa de un medio de mayor densidad a uno de menor densidad, así que la velocidad disminuirá y también la impedancia acústica. Véase figuras 1.6 - 1.8.

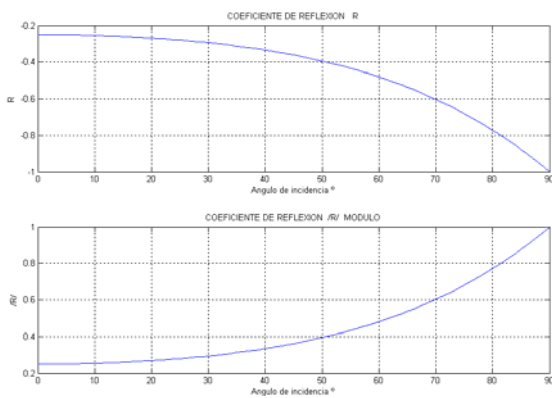


Figura 1.6. Comportamiento del coeficiente de reflexión.

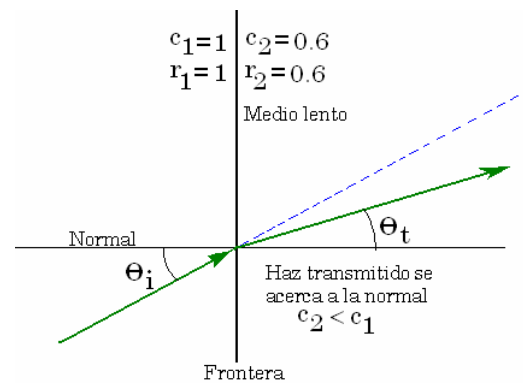
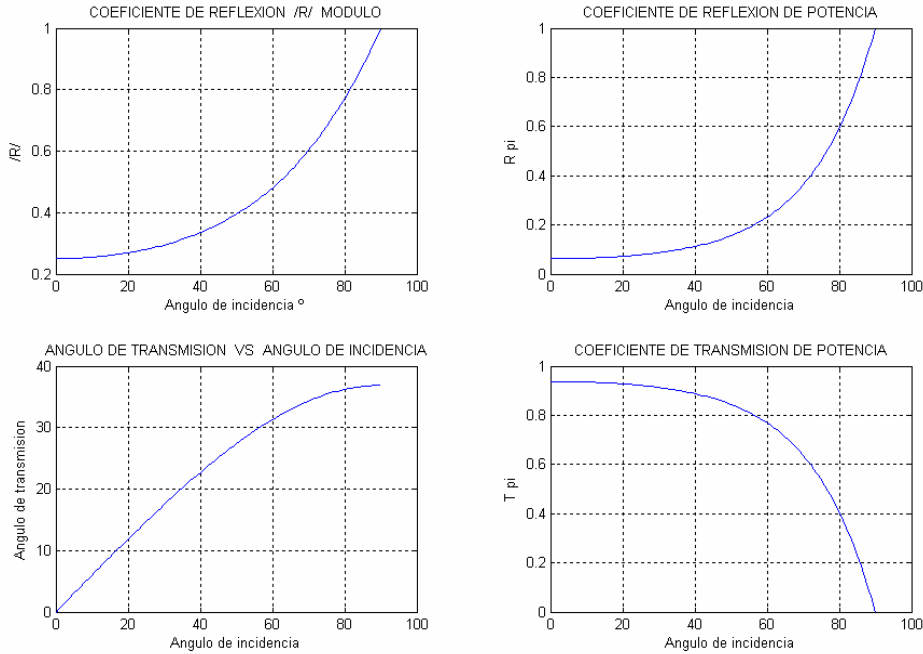


Figura 1.7. Incidencia oblicua en un medio lento y menor impedancia.
 $\theta_i \rightarrow 90^\circ$ ——— $\theta_t \rightarrow 37^\circ$

Bajo estas condiciones de r y c , siempre se tiene $\theta_t < \theta_i$ porque el haz transmitido se acerca a la normal siempre presentándose el caso A a la inversa. Cuando $\theta_i \rightarrow 90^\circ$, se da el máximo ángulo de incidencia, presentándose el mayor ángulo de transmisión posible que para este caso sería $\theta_t \approx 37^\circ$ ⁵. Llegando a esta condición se presenta la incidencia razante donde la onda incidente sería totalmente reflejada $R=1$.



Figuras 1.8. Visualización del comportamiento de la reflexión la transmisión y su ángulo correspondiente en función de los ángulos de incidencia bajo las condiciones $c_2 < c_1$ y $r_2 < r_1$.

Si se juntan los casos A y B para un muro real la trayectoria de la señal sería como se muestra en la figura 1.9.

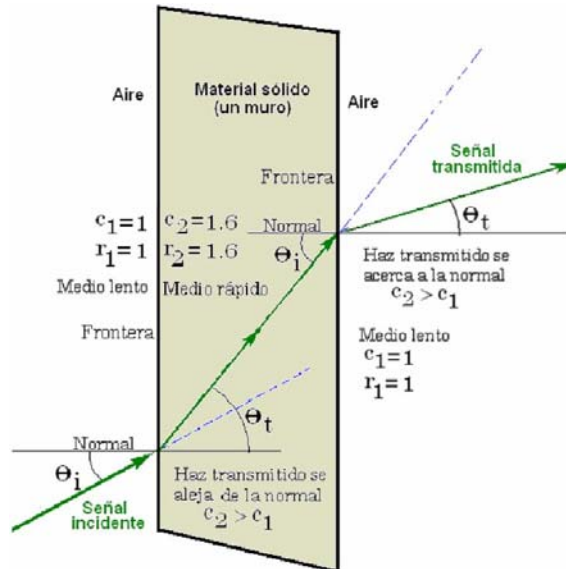


Figura 1.9. Transmisión del sonido a través de un muro sólido.⁶

⁵ Esto quiere decir que siempre se transmitirá la onda al segundo medio por lo que no existe un ángulo crítico como el caso A.

⁶ Esta figura muestra de manera exagerada la variación de los ángulos en el proceso con el propósito de hacer más visible el fenómeno de transmisión.

Se tiene que la onda sonora incidente sufre un cambio en su trayectoria al transmitirse al muro con un ángulo θ_i ; primero tiende a alejarse de la normal y posteriormente a salir del muro vuelve a acercarse a la normal, en decir, sucede el fenómeno a la inversa, por lo que el ángulo de salida será el mismo que el de entrada respecto a la normal.

Nota importante: De las gráficas 1.3 a 1.9 se ve que: para una onda que viaja por el aire e incide en un muro, cuando $\theta_i > \theta_c$, la señal incidente se reflejará totalmente, por lo que las únicas ondas que lograrán transmitirse y salir nuevamente del material al aire, serán aquellas que lleguen a incidir en el muro con $\theta_i < \theta_c$.

Esta es una observación importante porque muestra que gran parte de la señal es reflejada y una pequeña porción es transmitida sobre todo cuando se trata de muros altamente masivos.

En el caso que se presenta en la figura 1.9 se aprecia que las bases teóricas están desarrolladas para un medio homogéneo (aire y/o agua), una incidencia normal de la onda al muro divisorio y para muros homogéneos de un sólo material con sus correspondientes casos de incidencia oblicua. Lo que se analiza es el gran cambio de impedancia acústica ρc que se da en la trayectoria de transmisión a través el medio de propagación aire-material-aire, en donde se aprecia que la mayor parte de la energía acústica se refleja y otra de menor dimensión se transmite.

Como se veía en la figura 1 de la introducción, se tiene un recinto que tiene un muro en común con otro recinto el cual vibra debido a las señales incidentes, esta perturbación permite que el aire circundante en el segundo recinto reciba las vibraciones correspondientes y de esta manera el muro se convierte en una nueva fuente sonora que reproduce las frecuencias que se generaron en el recinto emisor pero con menor amplitud dependiendo de las propiedades físicas del muro.

Partiendo de lo anterior, se puede obtener la base para medir el aislamiento de un material conociendo la magnitud de la señal incidente y la transmitida, y en la medida de que esta última sea más pequeña, el aislamiento será mayor y viceversa. Con estos dos datos se puede establecer una cuantificación, que puede establecerse por medio de un cociente o bien expresarlo en dB^7 .

1.5. Expresión del aislamiento sonoro

Para dar a conocer la capacidad de aislamiento sonoro de un material, se debe indicar su magnitud en función de la frecuencia, ya que se presenta diferente comportamiento del material para cada longitud de onda, es decir, para diferentes frecuencias se tendrá diferente capacidad de aislamiento.

⁷ El decibel (dB) es una unidad de medición que permite expresar una magnitud física como el logaritmo del cociente de la misma y una referencia

El aislamiento sonoro, índice de reducción sonora o pérdida por transmisión R , se define con “diez veces el logaritmo de base diez del cociente entre la potencia sonora que incide sobre el objeto de prueba, W_1 , y la potencia sonora transmitida a través de éste, W_2 ”

$$R = 10 \log_{10} \frac{W_1}{W_2} \quad (1-11)$$

La Pérdida por Transmisión es definida como:

$$R = 10 \log_{10}(1/\tau) \text{ dB} \quad (1-12)$$

El coeficiente de transmisión τ para energía acústica que pasa de un volumen a otro es definido como:

$$\tau = \text{Potencia Acústica transmitida/Potencia Acústica incidente.} \quad (1-13)$$

Para incidencia normal, el coeficiente de transmisión τ esta dado teóricamente por:

$$\tau = \frac{4n}{[(\omega m - s/\omega)/\rho_2 c_2]^2 + (\omega_0 m \eta / \rho_2 c_2 + n + 1)^2} \quad (1-14)$$

Donde:

$$n = \rho_1 c_1 / \rho_2 c_2$$

η = factor de pérdida en el vacío. $\eta < 1$.

$\omega = 2\pi f$ = frecuencia angular.

ω_0 = frecuencia angular natural.

m = masa por unidad de área.

s = rigidez por unidad de área.

La forma práctica de expresar el aislamiento por cada banda de frecuencia es determinar el Nivel de Presión Sonora (NPS) en el lado incidente en dB y restar aritméticamente el nivel presentado en el otro lado.

$$\text{Aislamiento (R)} = NPS_{\text{incidente}} - NPS_{\text{transmitido}} \quad (1-15)$$

Donde cada nivel es resultado de la siguiente fórmula:

$$NPS = 20 \log_{10} \frac{P_1}{P_0} \quad (1-16)$$

Donde:

P_0 = presión sonora de referencia = $20\mu Pa$ @ $1KHz$.

P_1 = presión sonora experimental.

1.6. Categoría transmisión sonora (Sound Transmission Class, STC)

Este indicador está basado en estudios hechos con fuentes de ruido típicas de viviendas multifamiliares, constituye una especificación de un solo número, muy efectiva para indicar las características de aislamiento sonoro de un muro de manera global tomando en cuenta las bandas de frecuencia.

Desde luego este valor no deja ver toda la información, por lo que no se debe asumir que en todas las bandas de frecuencia presentará el mismo comportamiento.

Por ejemplo se tiene gráfica 1.10.

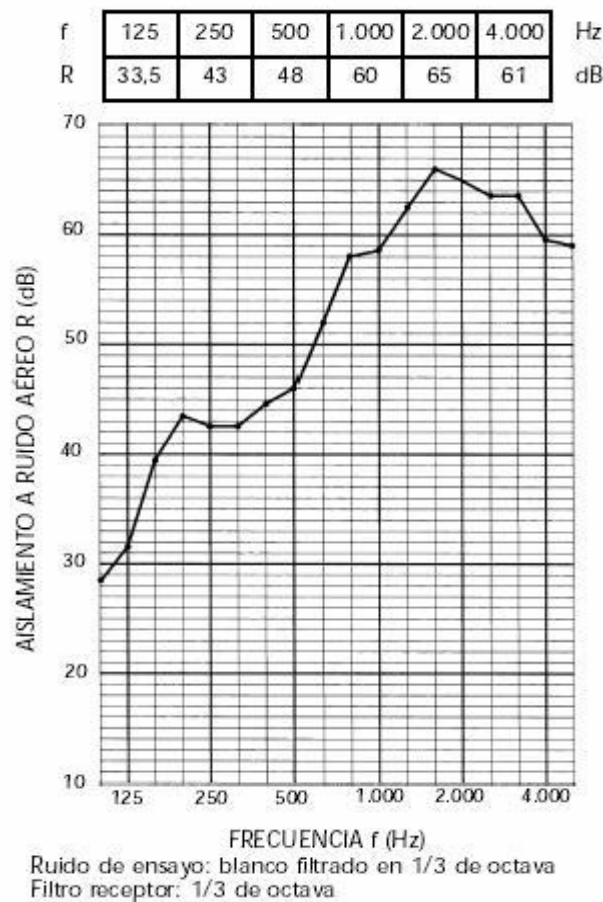


Figura 1.10 Gráfica de aislamiento sonoro de un muro por bandas de tercios de octava con su STC.

El STC para este material es de $51.5dB$, pero observe que en bajas frecuencias el aislamiento es menor y en altas frecuencias es mayor⁸, pero se establece este número para indicar en que categoría se ubica. Este dato es a menudo utilizado por arquitectos que desean saber de manera general que material usar para cierto propósito sin tener que poner atención a todas las bandas de frecuencia.

⁸ Esta diferencia de aislamiento se debe a la magnitud de la longitud de onda que se trata, así pues, en bajas frecuencias, la longitud de onda tiene de dimensiones comparables con la muestra a medir, por lo que, el muro no representa un obstáculo de consideración, en cambio, cuando la longitud de onda es pequeña, sucede lo contrario: el muro tiene dimensiones mayores y evita con facilidad que las ondas pasen.

El número de *STC* toma en cuenta la compensación que tiene que darse al aislamiento debido a la gran pérdida de su parte en bajas frecuencias por lo que su valor será representativo en función de las cualidades del oído humano promedio.

Existen dos métodos para calcular el *STC*, el método gráfico y el numérico.

Para el método gráfico se hace lo siguiente:

Se mide el aislamiento en 18 bandas continuas de frecuencia de $1/3$ de octava entre 100 y 5000Hz . Estos valores se comparan con una familia de contornos de referencia cuya curva de respuesta es similar a la de *ponderación A*, cada una de las cuales consta de tres líneas rectas que abarcan respectivamente bajas medias y altas frecuencias como se ve en la figura 1.11.

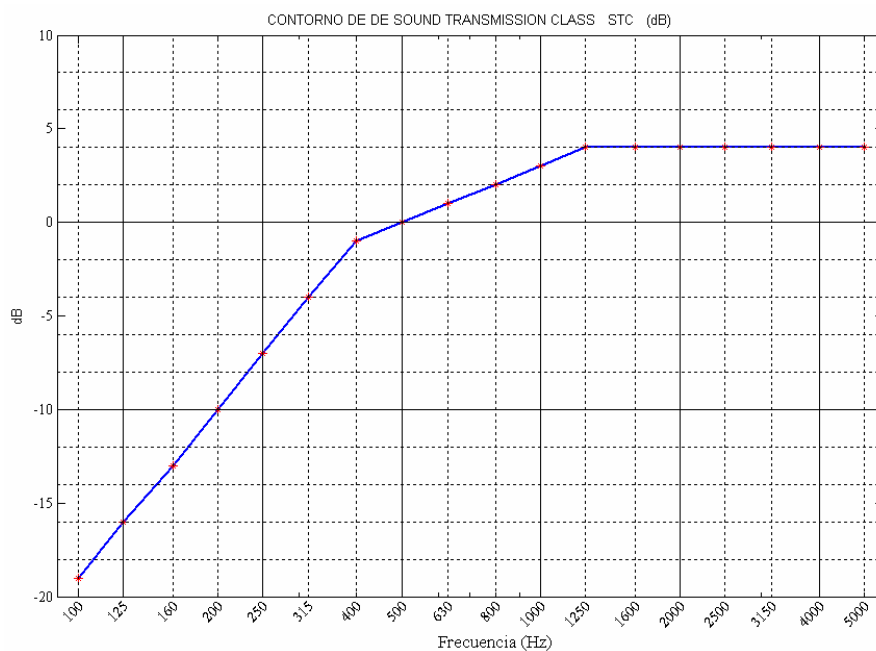


Figura 1.11 Contornos STC en bandas de frecuencia de tercios de octava para calcular el STC de un material a partir de los niveles de aislamiento por banda.

Después se escoge el contorno de referencia de tal manera que la deficiencia máxima (desviación de los datos abajo del contorno) a cualquier frecuencia no exceda de 8 dB y la diferencia total a todas las frecuencias no exceda de 32 dB .

El *STC* será entonces el valor de aislamiento correspondiente a la intersección del contorno de referencia elegido con la ordenada a 500Hz .

El método numérico fue desarrollado mediante un programa en *Matlab 6.5* en la *PC* con el que se ingresan los valores de aislamiento por banda de $1/3$ de octava en las bandas de frecuencia de estudio según la norma *ISO* que van de 100 Hz a 5KHz y devuelve el *STC* correspondiente. Este programa ha sido corroborado con la verificación de que el resultado dado por el método gráfico coincida con el método numérico.

1.7. Ley de masas y efecto de coincidencia

Este apartado permite estudiar y comprender el comportamiento de sistemas físicos más complejos y cercanos al hecho real.

Dado un material que es sometido a la incidencia de ondas sonoras, la energía atravesará el muro a través de sus poros en forma de una nueva onda generada al vibrar el muro. La atenuación de la onda sonora incidente depende de la frecuencia del sonido, de la masa del muro y de sus condiciones de contorno. En el caso de una pared simple, las propiedades aislantes varían considerablemente con la frecuencia, por lo que se suelen distinguir tres zonas de comportamiento:

Zona1 A bajas frecuencias, la pared se comporta como un muelle o placa vibrante, el aislamiento decae con la frecuencia y se ve afectado así mismo por las frecuencias propias o de resonancia, que están relacionadas con la rigidez y las condiciones de contorno de la pared. La existencia de estas frecuencias de resonancia disminuye drásticamente el aislamiento por debajo de los 100Hz.

Debajo de la frecuencia natural (de resonancia). $\omega \ll \omega_0 = (s/m)^{1/2}$ se tiene lo siguiente

$$R = 20 \log_{10} s - 20 \log_{10} (\omega/2\pi) - 20 \log_{10} (4\pi\rho_0c) \text{ dB} \quad (1-17)$$

ρ_0c representa la impedancia acústica del aire. En este caso, R se ve determinado primariamente por la rigidez elástica s del montaje y es insensitivo a la masa por unidad de área m y a la amortiguación ($r = \omega_0 m \eta$). R se decreta en frecuencia a una razón de 6 dB por octava.

La ley de masas se puede representar por medio de la figura 1.12.

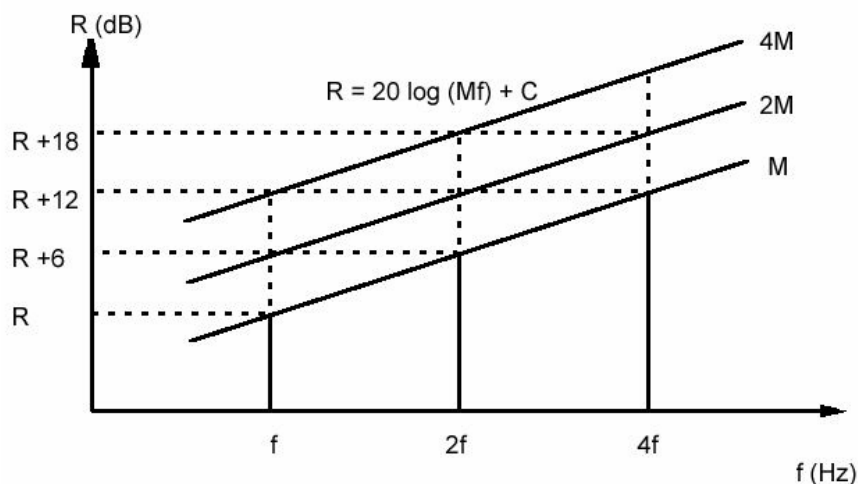


Figura 1.12 Ley de masas para un muro simple en función de la frecuencia.

En otras palabras R se determina por la masa por unidad de área y es muy insensible al amortiguamiento y rigidez. Se incrementa con la frecuencia por 6 dB/octava y por 6 dB cada vez que se dobla la masa.

Esta expresión es conocida en la literatura de acústica como la “Ley de Masas de Incidencia Normal” o “Ley de Masas”.

En la práctica, los materiales no responden exactamente a esta ley pero es una buena aproximación para entender su comportamiento.

Zona 2. Corresponde a la mayor parte de interés en las construcciones ya que la frecuencia de excitación se iguala a la frecuencia natural del sistema. En este intervalo de frecuencias el comportamiento de la pared está gobernado por su masa y se la puede considerar como una gran cantidad de pequeñas masas deslizando unas sobre otras. Para este rango de frecuencias se encuentran expresiones teóricas para el aislamiento en función de la masa y de la frecuencia llamada la ley de masas de la forma:

$$R = 20 \log_{10}(Mf) + C \quad (1-18)$$

Donde:

M = la masa por unidad de superficie

f = frecuencia

C = constante que depende de la impedancia del material.

El aislamiento R aumenta con una pendiente de $6dB/octava$, si se duplica la masa del espécimen o la frecuencia, aunque esto es así sólo hasta un cierto límite superior de frecuencias.

Así de esta forma en la frecuencia natural de resonancia $\omega = \omega_0$ se tiene

$$R = 20 \log_{10}(\omega_0/2\pi) + 20 \log_{10} m + 20 \log_{10} \eta - 20 \log_{10}(\rho_0 c/\pi) \text{ dB} \quad (1-19)$$

Esta difiere de la ley de masas en $\omega = \omega_0$ por $20 \log_{10}(\eta)$ dB

En incidencia oblicua, a un ángulo de incidencia Φ y el mismo medio en ambos lados del muro (aire-muro-aire) es importante la condición en la cual la onda incidente es coincidente con la onda de flexión del muro. En este caso, el coeficiente de transmisión esta dado por:

$$\tau = \frac{(2\rho_0 c / \omega m)^2 \sec^2 \phi}{[(2\rho_0 c / \omega m) \sec \phi + (k/k_b)^4 \eta \text{ sen}^4 \phi]^2 + [1 - (k/k_b)^4 \text{ sen}^4 \phi]^2} \quad (1-20)$$

Y la condición de coincidencia es:

$$k \text{ sen } \phi = k_b = (\omega^2 m / D)^{1/4} \quad (1-21)$$

Donde se asume un amortiguamiento estructural dado por la rigidez de flexión compleja (*complex bending stiffness*) $D' = D(1 + j\eta)$.

Este fenómeno es conocido como coincidencia, en la cual la parte reactiva del denominador del coeficiente de transmisión desaparece. Para cada ángulo de incidencia ϕ existe una frecuencia única de coincidencia ω_{co} y viceversa. Sin embargo como $\text{sen}\phi$ no puede exceder uno, existe un límite bajo de frecuencia a la cual puede ocurrir, dado por:

$$\omega_c = c^2(m/D)^{1/2} \quad (1-22)$$

Donde ω_c es conocida como la “frecuencia crítica” o la frecuencia de coincidencia mas baja. Debajo de esta frecuencia la transmisión es controlada por la masa.

El efecto de coincidencia se da cuando las ondas de flexión⁹ (λ_F) que se generan en la pared al vibrar, pueden entrar en resonancia con la onda sonora incidente (λ). Esto se puede representar con la siguiente figura.

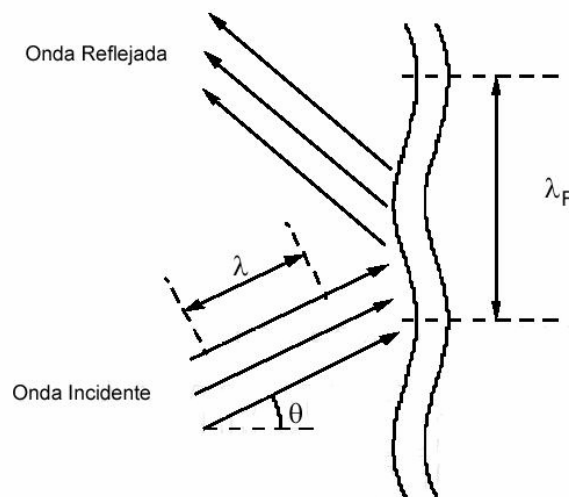


Figura 1.13 . Onda Sonora de flexión. Fenómeno de coincidencia

La menor de las frecuencias de coincidencia posibles se llama frecuencia crítica y marca el límite superior a la zona 2. Debido al fenómeno de resonancia, el aislamiento se reduce bruscamente a esta frecuencia. Por esta razón es que cada material puede llegar a presentar una deficiencia de aislamiento en una banda de frecuencia determinada ya que se crea una onda estacionaria en el interior de la partición y esto hace que el material radie sonido extra a otro lado haciendo que el NPS transmitido se incremente.

Para sonidos incidentes cuya longitud de onda cumpla con la relación

$$\lambda = \lambda_F \text{sen}\theta \quad (1-23)$$

Donde λ_F es la longitud de onda de flexión originada en la partición y θ el ángulo de incidencia, se producirá la resonancia. La frecuencia crítica viene dada por:

⁹ Las ondas de flexión son denominadas así puesto que al incidir en un medio, estas cambian de dirección doblándose en la interfaz.

$$f_c = \frac{6.14 \times 10^4}{d} \sqrt{\frac{\rho(1-\sigma^2)}{E}} \quad (1-24)$$

Donde:

d = el espesor del espécimen.

ρ = la densidad del material.

σ = es el coeficiente de Poisson.

E = es el módulo de Young.

λ = la longitud de onda incidente.

Zona 3.- Por encima de la frecuencia crítica, la pared no obedece a la ley de masas comportándose nuevamente como un resorte porque la longitud de onda tiende a ser menor. En esta zona el factor que gobierna las variaciones del aislamiento es el amortiguamiento interno del material, es decir, la capacidad del material para absorber la energía acústica a las ondas de flexión. Existen sucesivas frecuencias de coincidencia muy distanciadas. La pendiente de la curva de aislamiento a frecuencias justo superiores a la de coincidencia para ser de 10dB/octava en vez de los 6dB correspondientes a la ley de masas.

Las tres zonas descritas pueden apreciarse en la siguiente figura:

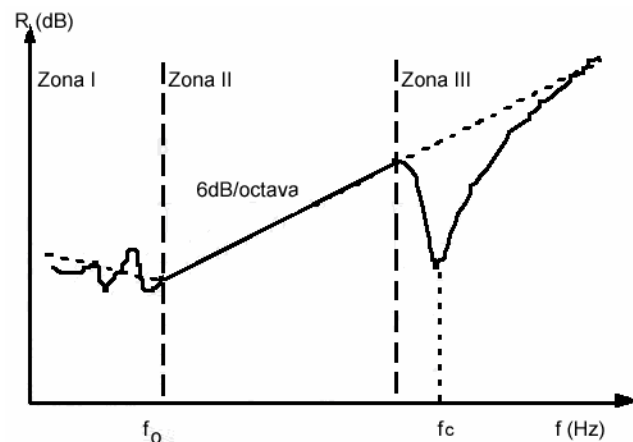


Figura 3.13 Zonas de aislamiento de una pared simple.

Cabe notar que, aunque en la zona 2 es la Ley de masas quien gobierna el aislamiento, existe una dependencia con el ángulo de incidencia. La variación del aislamiento con el ángulo de incidencia Φ viene dado por la ecuación:

$$R_\Phi = 10 \log_{10} \left[\frac{M\pi f \cos \Phi}{\rho_0 c} \right]^2 = 20 \log_{10} (M\pi f \cos \Phi) - 42 \quad (1-25)$$

Donde:

ρ_0 = la densidad del aire

c = la velocidad del sonido en el aire.

Por lo anterior se tiene que el aislamiento disminuye en aquellas situaciones en que la onda incide rasante con el muro, lo cual se observa frecuentemente en el caso de fuentes direccionales, tales como el ruido del tráfico al incidir sobre las fachadas de pisos altos.

En el caso particular de recintos cerrados con ondas incidiendo en todas direcciones, si se hace un promedio para todos los ángulos de incidencia entre 0° y 180° , se obtiene un aislamiento $5dB$ menor que en el caso de incidencia normal.

Otra manera de expresar lo anterior de forma analítica se puede escribir como:

$$R = 20 \log_{10} m + 20 \log_{10} (\omega/2\pi) - 20 \log_{10} (\rho_0 c / \pi) \text{ dB} \quad (1-26)$$

Y en aire se puede aproximar con:

$$R \approx 20 \log_{10} (m\omega/2\pi) - 42 \text{ dB} \quad (1-27)$$

En la práctica, las ondas sonoras inciden en el muro divisorio desde muchos ángulos simultáneamente y la distribución direccional de la intensidad incidente rara vez se conoce, por lo que un modelo de campo difuso (incidencia aleatoria) es el que se asume. Este modelo supone ondas planas no-correlacionadas que inciden en todas direcciones con la misma probabilidad y con fase aleatoria.

La expresión para el coeficiente de transmisión de campo difuso es:

$$\tau_d = \frac{\int_0^{\pi/2} \tau(\phi) \text{sen}(\phi) \cos(\phi) d\phi}{\int_0^{\pi/2} \text{sen}(\phi) \cos(\phi) d\phi} = \int_0^{\pi/2} \tau(\phi) \text{sen}(2\phi) d\phi \quad (1-28)$$

La expresión no es muy amigable para su integración analítica por lo que se prefiere usar la expresión para frecuencias por debajo de la frecuencia crítica. Quedando la expresión:

$$R_d = R(0) - 10 \log_{10} [0.23 R(0)] \text{ dB} \quad (1-29)$$

Es sabido que los resultados experimentales no coinciden bien con esta ecuación. Por este motivo se suele emplear en la literatura una expresión empírica con valores un poco mayores:

$$R_f = R(0) - 5 \text{ dB} \quad (1-30)$$

O bien

$$R_f = 20 \log_{10} (m \omega / 2\pi) - 47 \text{ dB} \quad (1-31)$$

Estas expresiones son conocidas como la “Ley de masas para incidencia de campo difuso”.

Como destaca Fahy [12], una expresión para el Índice de Reducción de Sonido en Campo Difuso a frecuencias por encima de la frecuencia crítica fue derivada por Cremer [14].

$$R_d = R(0) + 10 \log_{10} (f/f_c - 1) + 10 \log_{10} \eta - 2 \text{ dB} \quad (1-32)$$

Para el método de medición con Intensidad Acústica, que implica la medición del flujo medio de energía sonora a través de una superficie perpendicular a la dirección de propagación, su desarrollo, instrumentación y mediciones están ampliamente cubiertas en Fahy [12] aun que en este trabajo no se desarrollará el procedimiento de esta técnica, simplemente se darán algunos resultados que se obtuvieron por parte del CENAM.

Para este caso el *Sound Transmission Loss*, tiene la ecuación:

$$TL = R = 10 \log_{10} [\langle \overline{p_1^2} \rangle S / 4\rho_0 c W_t] \text{ dB} \quad (1-33)$$

Donde:

$\langle \overline{p_1^2} \rangle$ = es el promedio espacial de la presión cuadrática media en el recinto de la fuente.

S = la superficie del muro

W_t = la potencia acústica transmitida a través de una superficie de medición de intensidad que encierra la cara de transmisión del muro divisorio.

1.8. Normativa existente.

Para la medición del aislamiento sonoro se tiene la siguiente norma:

UNE-EN ISO 140-3:1995 ACÚSTICA. MEDICIÓN DEL AISLAMIENTO ACÚSTICO EN LOS EDIFICIOS Y DE LOS ELEMENTOS DE CONSTRUCCIÓN. PARTE 3: MEDICIÓN EN LABORATORIO DEL AISLAMIENTO ACÚSTICO AL RUIDO AÉREO DE LOS ELEMENTOS DE CONSTRUCCIÓN [1]

En esta norma se citan todas las condiciones, procedimientos y cálculos para lograr una medición confiable, creando un campo difuso de ruido aleatorio en un recinto emisor con un *NPS* determinado y tanto el nivel en el recinto emisor como el receptor se tienen que colocar el instrumental bajo condiciones específicas.

Para el desarrollo de estas mediciones se consultó este documento [1] del que se elige las condiciones de medida en función de la dimensión de las instalaciones. La ubicación de los elementos fue condensada de acuerdo con el personal del CENAM y los académicos del CCADET para tener condiciones iguales de medición.

II. TÉCNICAS DE MEDICIÓN DEL AISLAMIENTO SONORO

2.1. Técnicas de laboratorio.

Para las técnicas de laboratorio se debe contar con instalaciones especiales como es la cámara de transmisión que involucra gran infraestructura que solamente algunos centros de medición pueden contar.

Las mediciones en general llevan la mayor carga de trabajo en el planteamiento de colocación de los dispositivos de medida para que se tengan lecturas confiables y repetibles.

2.1.1. Cámaras de transmisión.

Las cámaras de transmisión consisten básicamente en un recinto aislado del exterior con paredes altamente masivas que alberga dos recintos más pequeños de forma irregular y de volúmenes diferentes interconectados con una cara en común que será la muestra a medir.

Uno de los cuartos será el emisor y el otro el receptor en donde se debe lograr en la mayor medida posible que el sonido sólo pase por la muestra evitando la transmisión por flanqueo o sea por las aristas u orillas.

Una vista superior de una cámara de transmisión es la siguiente:

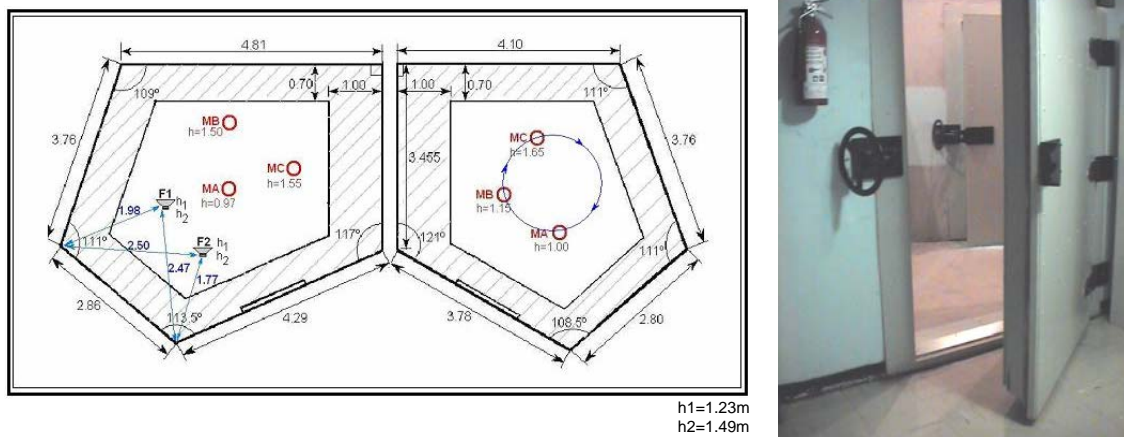


Figura 2.1 Cámara de transmisión con elementos de medición. Vista superior y entrada.

En la figura 2.1 se puede ver que el recinto emisor de la izquierda tiene las posiciones de fuentes sonoras y de micrófonos indicados por círculos rojos a diferentes alturas del piso h para captar el sonido de campo difuso en diferentes puntos. En el otro recinto se ve que están los micrófonos para medir el sonido transmitido en diferentes posiciones o bien se puede hacer rotar un micrófono y hacer un promedio espacial.

Las dimensiones están dadas en metros y corresponden a las que tiene la del laboratorio de acústica del CCADET. Las superficies rayadas corresponden a áreas restringidas en donde según la norma, no se puede colocar instrumental de medición.

2.1.2. Técnica por medio de Nivel de Presión Sonora.

Para medir con esta técnica ya establecida, es necesario acondicionar acústicamente los recintos de tal manera que los tiempos de reverberación sean los adecuados, según las normas, para crear un campo difuso en el recinto emisor. Se mide el NPS cuando menos bajo cinco diferentes combinaciones de micrófono-altavoz, según las normas y teniendo estos datos se puede calcular el aislamiento por cada banda de frecuencia de interés con la siguiente fórmula:

$$R = NPS_1 - NPS_2 + 10 \log_{10} \frac{S}{A} \quad (2-1)$$

Donde:

NPS_1 = el NPS del recinto emisor.

NPS_2 = el NPS del recinto receptor.

S = el área del espécimen a medir.

A = el área de absorción equivalente o también llamada de absorción sonora.

El tercer sumando de la fórmula 2-1 incrementa la magnitud del aislamiento que se mide, debido a que se tiene que tomar en cuenta que, debido a la cara del lado del recinto receptor, existe una porción de sonido que está de más y eso hace que el NPS detectado sea más elevado de lo que debería ser. Para solucionar esto, se debe tomar en cuenta el área del muro para tener una compensación que permita expresar acertadamente cuanto cuánto del sonido se aisló, tomando en cuenta que sólo el efecto que tiene el sonido al pasar de un lado a otro sin añadir los efectos debidos a las reflexiones dentro del cuarto.

El área de absorción equivalente se calcula con la fórmula:

$$A = 0.16 \frac{V}{T} \quad (2-2)$$

Donde:

V = volumen del recinto receptor en m^3 .

T = tiempo de reverberación de la banda de frecuencia en estudio en segundos

Nota: como se puede apreciar, sustitución de unidades no arroja unidades de superficie, pero la constante ya involucra otras variables para obtener el valor deseado en m^2 con la sustitución numérica.

En el desarrollo experimental se darán a conocer los detalles para realizar esta medición

2.2. Técnicas “in situ” alternativas.

Este tipo de técnicas están basadas en realizar mediciones dentro de recintos semireverberantes cerrados con condiciones no muy exigentes a las que se plantean en los métodos de laboratorio; por supuesto deben controlarse en la más que se pueda el ruido de fondo, con la ventaja de que los montajes son rápidos y ofrecen una alternativa que puede ser muy útil para la realización de este tipo de mediciones.

2.2.1. Técnica Cepstrum.

Esta técnica consiste en utilizar una señal de prueba como ruido aleatorio que se mida y se le haga incidir en el material para que sufra una pérdida de transmisión midiendo del otro lado la cantidad de sonido que logró pasar y así se puede evaluar R .

La justificación del uso de esta técnica es: que puede manejar ruido aleatorio que contiene todas las frecuencias en un ancho de banda, dependiente de la frecuencia de muestreo.

Además, la *técnica Cepstrum* permite la posibilidad de eliminar toda perturbación no deseada como puede ser las reflexiones del recinto y las difracciones que pueden generar los bordes del espécimen que alteran la lectura.

Debido a que al monitorear la señal transmitida es prácticamente imposible saber cuándo llegan las señales no deseadas se utiliza el espectro *Ceptrum* de la señal transmitida el cual indica la presencia de repeticiones captadas. Esto se puede ver de otra manera como la presencia de reflejos debido a las paredes próximas del recinto. Desde luego estas señales de ruido para la medición deben ser eliminadas con técnicas de ventaneo¹.

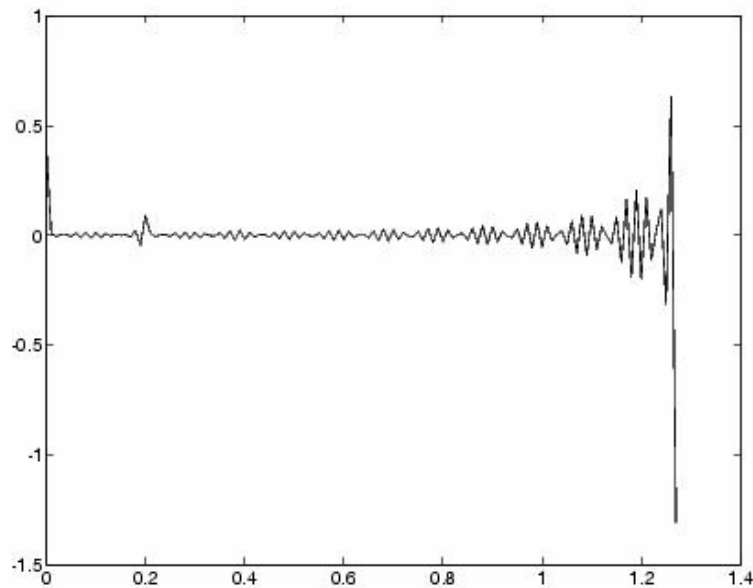


Figura 2.2. Cepstrum de una señal que indica los reflejos presentes.

En la figura se puede ver el *Cepstrum* como una señal en función del tiempo en donde en el $t=0$ se presenta la señal que primeramente se recibe. Note que en $t=0.2s$ se presenta una perturbación muy notable “espiga” que indica la presencia de un reflejo o sea que se está detectando una señal igual a la que se detectó en el $t=0$ pero con menor amplitud (un reflejo).

¹ La técnica de ventaneo consiste en visualizar y tomar en cuenta una porción de una señal, de tal manera que la forma de la ventana, sobrepuesta en la señal original, le de un peso a cada valor de la señal en un intervalo de 0 a 1, para procesarla más adelante. Así, cada dato de la señal se multiplicará por el correspondiente valor de la ventana que esté en la misma ordenada.

Para el propósito de esta medición se evalúa el *Cepstrum real* de una señal en el tiempo x , y se calcula determinando el logaritmo natural de la magnitud de la transformada de Fourier de la señal x , entonces se obtiene la transformada inversa de Fourier de la secuencia anterior [18]. Lo anterior queda expresado matemáticamente en la fórmula 2-3.

$$C_x = \frac{1}{2\pi} \int_{-\infty}^{+\infty} \log |X(e^{j\omega})| e^{j\omega n} d\omega \quad (2-3)$$

Para determinar el aislamiento sonoro se parte de un trabajo titulado: "A Quick Method of Estimating Sound Insulation" de Vladimir D. Smirenov [9] en donde propone un esquema de conexión de dispositivos así:

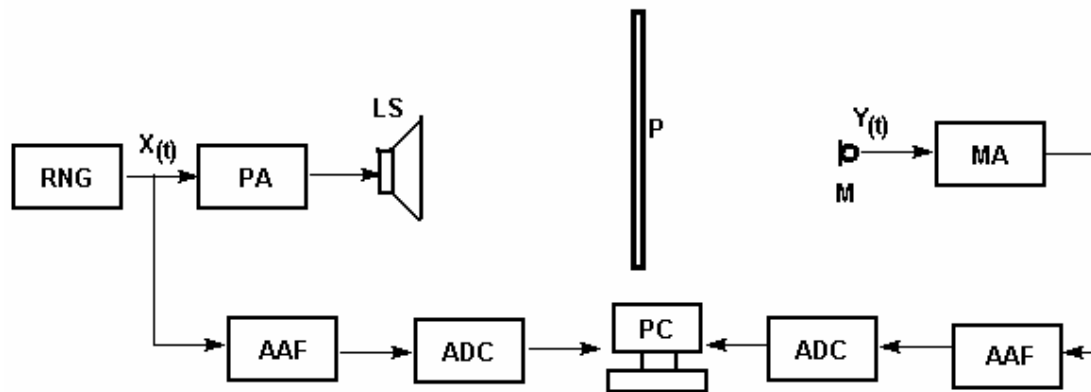


Figura 2.3. Método de medición de aislamiento sonoro propuesto por Smirenov.

En su método propone utilizar ruido aleatorio y registrarlo en la *PC* por medio de convertidores *A/D*. Se ve que por un lado registra el ruido que se obtiene del generador $x(t)$ y por el otro lado captura la señal que se transmitió $y(t)$ al otro lado de la muestra a examinar P . De esta manera él se propone extraer el ruido reverberante del recinto en el que se encuentra y así estimar el aislamiento en diferentes bandas de frecuencia.

Esta técnica permitió tener las bases para proponer este método para poder medir sin la necesidad de una cámara de transmisión.

Para conocer el funcionamiento del *Cepstrum* se recurrió al estudio del capítulo 9. *Cepstrum, Liftered Spectrum Sixth Guided Tour* [7] del manual del Analizador 2034 de B&K. La operación de esta función es la siguiente:

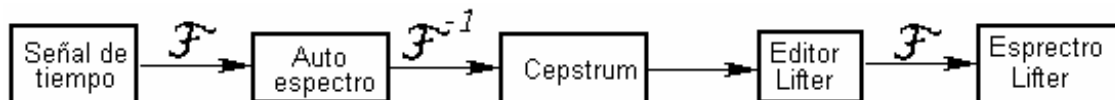


Figura 2.4. Secuencia de cálculo simplificada para la obtención del *Cepstrum*

Se observó que cada reflejo presentado se manifiesta por medio de una "espiga" cuyo tamaño es casi proporcional al de la amplitud con la que ésta se presenta.

A partir del *Cepstrum* se obtiene el *Liftered Spectrum* el cual da un espectro en función de la frecuencia cuya forma cambia dependiendo del número de datos que se tomen en cuenta del *Cepstrum*.

En la medida en la que se tomen mayor cantidad de datos, el *Liftered* se parecerá más al espectro de la señal. Si se seleccionan los datos que están exclusivamente antes del primer reflejo (primera espiga), se mostrará el espectro de la señal directa y de ahí se puede obtener la señal correspondiente en tiempo aplicando la transformada inversa de Fourier.

Los ejercicios planteados en el manual descrito [7], se desarrollaron para adquirir habilidad en el manejo de la técnica.

La implementación de este modelo en la *PC* con *Matlab 6.5* permite evaluar el *Cepstrum* real de una señal y con base a su forma se procesa la información para obtener el aislamiento sonoro.

2.2.2. Técnica MLS.

Para la medición de *R* se pueden utilizar las Secuencias de Máxima Longitud (Maximum Length Sequences, *MLS*) que son básicamente una secuencia de pulsos que se generan de manera pseudoaleatoria² que permiten obtener la respuesta al impulso del sistema que se mide con el mismo montaje de dispositivos que en la técnica *Cepstrum*.

La justificación de porqué se puede ocupar esta técnica es: que la señal de prueba consiste en aplicar *deltas* que contienen una gama de frecuencias en función de la frecuencia de muestreo que se trabaje; entonces al obtener la respuesta impulsiva de la señal incidente y la transmitida, se puede evaluar su contenido de frecuencia y calcular el aislamiento por diferencia de nivel.

Las *MLS* se generan bajo una frecuencia de muestreo y se presentan con un periodo que contiene $(2^N)-1$ eventos, donde *N* es el número de orden de la secuencia. Pueden existir diferentes *MLS* del mismo orden y se pueden repetir cuantas veces sea necesario para que se obtenga un promedio de todas esas respuestas al impulso que se repiten cada vez a una determinada frecuencia de muestreo.

Estas señales de excitación tienen como característica principal que la autocorrelación es aproximadamente una señal delta. Si tomamos en cuenta esta propiedad y de acuerdo con la teoría de los sistemas dinámicos lineales, la correlación de entrada salida $\Phi_{xy}[n]$ de un sistema lineal invariante en el tiempo [6] se define como:

$$\Phi_{xy}[n] = \Phi_{xx}[n] * h[n] \equiv \sum_{k=0}^{N-1} x(k-n)_y(k) \quad (2-4)$$

Donde:

$\Phi_{xx}[n]$ corresponde a la autocorrelación de la señal de entrada $x[n]$,
 $h[n]$ es la respuesta al impulso y * indica convolución.

² El término pseudoaleatorio se refiere a la reproducción de una señal aleatoria de cierta duración y contenido de frecuencia, que se repite un número determinado de veces.

Ahora, si $\Phi_{xx}[n]$ es igual a una señal delta, se tiene que la respuesta al impulso es proporcional a la función de correlación entre la entrada y la salida $\Phi_{xy}[n]$. En la figura 2.5 se presenta lo mencionado.

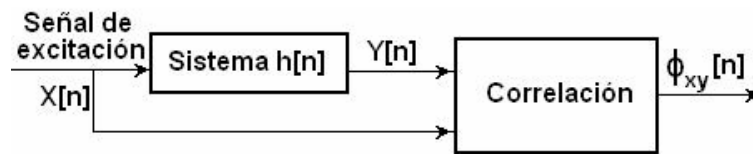


Figura 2.5. Método para medir la respuesta al impulso a partir de la función de correlación entrada salida

La respuesta al impulso de un sistema puede ser obtenida calculando la correlación cruzada entre las señales de entrada y salida, después se puede procesar la señal en tiempo con un ventaneo mediante una *ventana tukey* que puede acotar al intervalo de tiempo de interés.

En la siguiente figura se aprecia un acercamiento a una respuesta impulsiva de un sistema en donde se nota que casi al final hay una repetición del impulso o sea una reflexión debido al recinto en donde se mide misma que debe ser eliminada con técnicas de ventaneo³ para que de esta manera se trabaje con el sonido directo.

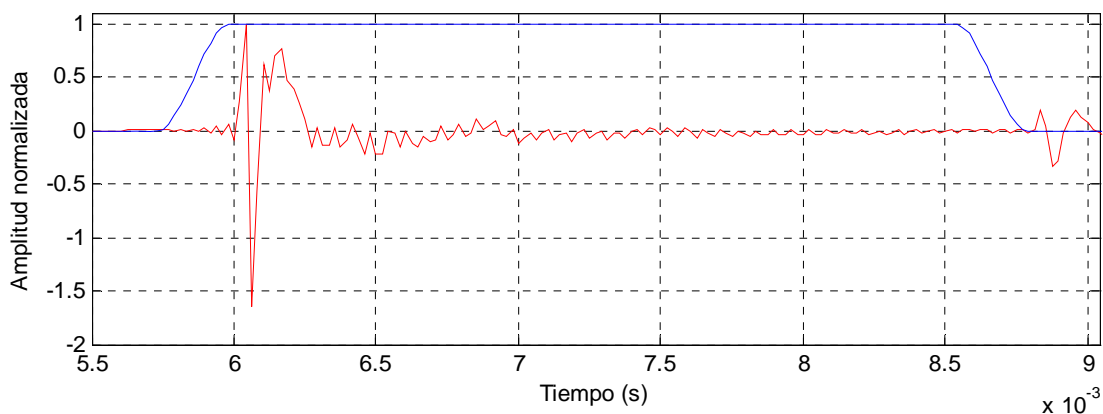


Figura 2.6. Respuesta impulsiva normalizada presentando el impulso directo y una repetición de menor amplitud, con una *ventana tukey* de meseta plana.

Para esta técnica se debe medir una *MLS* incidente y otra transmitida las cuales deberán ser evaluadas y comparadas en magnitud para determinar el aislamiento evaluando el área bajo la curva de cada una.

Las técnicas *MLS* tienen ventajas comparadas con otras formas de medición, como son:

- Presenta una potencia del espectro prácticamente plana en donde puede caer la respuesta en unos $1.6dB$ en un tercio de la frecuencia de muestreo.
- Desecha las componentes de *DC* de la señal muestreada.
- La relación señal a ruido es muy alta.

³ La técnica de ventaneo consiste en visualizar y tomar en cuenta una porción de una señal, de tal manera que la forma de la ventana, sobrepuesta en la señal original, le de un peso a cada valor de la señal en un intervalo de 0 a 1, para procesarla más adelante. Así, cada dato de la señal se multiplicará por el correspondiente valor de la ventana que esté en la misma ordenada.

- La correlación cruzada usada en el cálculo de la respuesta al impulso reduce todo el ruido de fondo (no correlacionado con *MLS*) así que las medidas pueden ser realizadas también en ambientes relativamente ruidosos.
- El uso de técnicas de promedio ayudan a incrementar la relación señal a ruido.

Dentro de las precauciones que se deben tomar al utilizar *MLS* son:

- La duración de la *MLS* debe ser mayor que la respuesta al impulso del sistema por analizar o cuando menos el mismo tamaño. Si estas condiciones no se satisfacen, algunas de las partes no serán consideradas generando una deficiente medición.
- El sistema bajo estudio debe ser invariante en el tiempo, al menos durante el intervalo de medida, o sea que se busca lograr un régimen estacionario.

Las *MLS* de manera ideal lo que hacen es aplicarle una señal que contenga toda una gama de frecuencias en un ancho de banda, que está en función de la frecuencia de muestreo de adquisición, para posteriormente analizar cual es el nivel registrado por banda de frecuencia.

Esta técnica resulta muy práctica ya que solamente se trabaja con dos respuestas impulsivas, lo cual facilita el procesamiento de señal.

La técnica *MLS* se logra generando las secuencias con un programa en lenguaje *C* que es producto de la tesis: Medición y cálculo de los índices acústicos en recintos [6] con el que se pueden reproducir y capturar con la tarjeta de sonido de la *PC* para obtener los dos archivos de datos que contienen las respuestas al impulso.

III DESARROLLO EXPERIMENTAL DE MEDICIÓN CON LA TÉCNICA DE NIVELES DE PRESIÓN SONORA NPS

En este capítulo se describen cada una de las técnicas de medición utilizadas como son *NPS*, *Cepstrum* y *MLS*. Cada una de ellas tiene un modo de funcionamiento en función del tipo de señal de prueba que se usa y del montaje del material a medir. Cabe aclarar que, se está proponiendo una forma de medición que disminuya costos y tiempos de ejecución por lo que se está instrumentando el uso de la computadora como herramienta de análisis, en lugar de un analizador de espectro en tiempo real, que es cuando menos diez veces más caro que un ordenador.

Cada técnica obtiene y graba los datos del experimento en su memoria, para posteriormente procesarlos, lo que implica no tener un análisis en tiempo real como lo hacen los instrumentos especializados, pero para nuestro caso no es necesario, ya que los resultados se dan unos minutos después. Precisamente gran parte del trabajo, consistió en que los resultados entregados por los programas realizados, dieran los mismos valores que daría un analizador por lo que se instrumentaron una serie de procedimientos de calibración y ajuste del instrumental para obtener resultados verdaderos y presentables.

3.1. Técnica NPS.

La técnica *NPS* (Técnica por medio de medición de Nivel de Presión Sonora), consiste básicamente en un método establecido por la norma *ISO-140 parte III*; en la que establece el uso de una cámara de transmisión bajo condiciones bien definidas [1]. Lo que se realiza básicamente es: aplicar ruido blanco en un ancho de banda de 100 a 5KHz en el recinto emisor y medir los niveles en tercios de octava, posteriormente se mide de la misma manera el nivel registrado en el recinto receptor; así pues se comparan banda por banda de NPS para conocer el nivel que logró aislar el material restando el nivel transmitido al nivel incidente.

3.2. Planteamiento del problema.

Se debe medir el Nivel de Presión Sonora en ambos recintos por medio de micrófonos que se conectan a un analizador que plasmará en pantalla los niveles por tercios de octava incidentes y transmitidos mismos que se pueden procesar para determinar el aislamiento. Este método ya establecido se pretende implementar usando una computadora que utilice programas en *Matlab 6.5* que procese los datos en lugar del analizador de espectro en tiempo real.

Para que la computadora pueda procesar la información es necesario ingresar la señal eléctrica de los micrófonos a su entrada de audio e la tarjeta de sonido en dos canales. La señal de los micrófonos se debe acondicionar usando un amplificador de medición que indique en su carátula los niveles que se están registrando y la salida será conectada a uno de los canales de entrada de la tarjeta de sonido. La señal recibida debe llegar a la computadora con un nivel que esté en el intervalo dinámico de la tarjeta.

El mejor nivel de señal será aquel que esté lejos del nivel de ruido, pero que no llegue a saturar el intervalo dinámico de la tarjeta de sonido de la PC, por lo que se deben determinar estos niveles.

Se requiere medir los tiempos de reverberación por bandas de tercio de octava, en ambas cámaras, a partir de las curvas de decaimiento, verificando que cada una de ellas esté bajo la norma. Para ello, se medirá esta magnitud en más de cinco posiciones en la cámara.

Se medirán los *NPS* cuando menos en cinco combinaciones diferentes de posición de la fuente y micrófono en ambas cámaras, por lo que cada caso tendrá un archivo de datos.

Con la solución de estas premisas se contará con un sistema de medición que no requerirá un analizador de espectros, bastará con el uso de micrófonos con amplificador de medición o simplemente sonómetros que se conecten a la PC. El uso de este procedimiento se desarrolla en los siguientes puntos.

3.3. Caracterización de la Cámara de transmisión.

Uno de los puntos básicos para medir con esta cámara, es comenzar con un evaluación que permita establecer cuales son los límites de medición y los posibles errores que pudiera arrojar este método de medida. Para hacer esto, se debe instalar un material a medir, que suponga un espécimen que evite al máximo la transmisión de sonido de una cámara a la otra, así se tendrán condiciones de aislamiento máximo posible, donde la única vía por la que pasará sonido será por flanqueo, es decir, por medio de los bordes de la muestra, las esquinas de la construcción y cualquier otra vía indirecta que no sea el material de prueba. Así es que, se podrá saber si la transmisión por vías indirectas influirá en la evaluación de materiales y permitirá conocer los límites de medición de las instalaciones.

En la práctica se construyó un muro doble de ladrillo rojo con un espesor de 25cm el cual permitió determinar el aislamiento máximo de la cámara de transmisión y la transmisión por flanqueo.



Foto 3.1 Espesor del muro de 25 cm. Imagen tomada al derribar el muro de prueba.



Foto 3.2. Muro construido para la caracterización para generar condiciones de aislamiento máximo

Nota: Los siguientes puntos pretenden mostrar cómo es que se mide el aislamiento sonoro con los programas elaborados. Este procedimiento de medición es el mismo que se aplicará para medir tanto la caracterización de la cámara de transmisión, como cada una de las muestras que se quieran medir.

3.4. Programación en la PC con Matlab 6.5.

Los programas realizados en *Matlab* 6.5 permiten adquirir las señales de los niveles detectados por los micrófonos y fueron ajustados de tal manera que los resultados que arrojan fueran confiables. Se llevaron a cabo numerosas pruebas de adquisición de datos para saber cuáles eran las mejores condiciones de los controles de nivel de entrada y salida de los instrumentos ya que no existe un manual que indique los ajustes para este tipo de conexión que se propone.

Programas elaborados para medición y cálculo de aislamiento

El siguiente listado menciona los programas que se realizaron y se da una breve explicación de su función. El primer programa que se emplea para esta técnica es:

RMS94dBcal2CHs.m.- Este programa permite medir el voltaje *RMS* de los dos canales de entrada de línea (izquierdo y derecho), y permite visualizar la forma de onda de la señal de prueba. Puede usarse para tener diferentes tipos de información, por ejemplo:

a).- Saber el nivel *RMS* introducido a la *PC* correspondiente a un *NPS* de calibración, aplicado al micrófono para que sirva de referencia, es decir que se tendrá un voltaje detectado que corresponderá a *93.8dB* de *NPS*. Con ello se podrá referenciar cualquier señal en el futuro y así saber, con base en este voltaje, cual es la magnitud en *dB* de una señal de la que solo conocemos su valor *RMS*.

b).- Se puede conocer cual es el voltaje de saturación de la tarjeta, simplemente incrementando la señal aplicada hasta que se empiece a recortar la señal. Así se sabrá cuál es el voltaje máximo que puede soportar la tarjeta sin tener distorsión armónica y así evitar que se dañe.

c).- Permite saber cual es el voltaje de ruido simplemente cortocircuitando la entrada o conectando los aparatos sin aplicar señal. Así se puede saber el intervalo dinámico de entrada conociendo el voltaje antes de saturación y el de ruido.

Nota: el programa graba simultáneamente los dos canales, así que si se desea trabajar con uno a la vez, simplemente se ignora el otro sin problema alguno.

RevBdeF.m.- Este programa permite evaluar el tiempo de reverberación, en bandas de frecuencia de tercio de octava, de *100* a *5KHz* a partir del archivo de la respuesta impulsiva por medio de *MLS*, en un solo punto de medida. Permite visualizar rápidamente si el archivo de la respuesta impulsiva no tiene errores, si no presenta saturación y si tiene suficiente margen de caída, cada vez que se toma un punto de medición.

RevBdeFtodo.m.- En éste se calculan los tiempos de reverberación promedio, con desviación estándar de doce archivos (doce posiciones *B1, B2, ... , B12*) de respuestas impulsivas de la cámara receptora y genera otro archivo *.mat* que servirá para en cálculo del aislamiento.

RevBdeFCamaraA.- Hace lo mismo que el anterior, sólo que este sirve para aplicarse a la cámara emisora para constatar que los tiempos de reverberación están bajo norma.

RdeF2CHyRdBOFF.m.- Este programa permite medir el ruido de fondo de ambas cámaras teniendo en cuenta que todos los instrumentos que pudieran generar algún ruido acústico estén apagados (medición de ruido exclusivo de las cámaras).

RdeF2CHyRdBON.m.- realiza lo mismo que el anterior, pero ahora puede añadir el ruido eléctrico que generan los propios instrumentos como el amplificador al mínimo, con la fuente omnidireccional, y la jirafa girando. (Este es el ruido de fondo medido que realmente estará presente en las mediciones de aislamiento).

RdeFProcesamientoON.m y *RdeFProcesamientoOFF.m.*- Estos dos programas permiten procesar los datos generados por los dos anteriores respectivamente.

La idea de separar los archivos es que uno de ellos adquiera los datos y el otro los procese, ya que en la práctica se ha visto que es mejor y más seguro porque se puede grabar exclusivamente con un programa y con el otro se puede, exclusivamente, procesar los datos sin riesgo de perder la información. De lo contrario, teniendo todo en un solo archivo, se tendría el riesgo de correr el programa y borrar la adquisición. Si solamente se quisiera procesar los datos se tendría que copiar y pegar la parte correspondiente al procesamiento de datos, lo cual es incómodo.

AislamientoPreliminar.m.- Con éste puede saber en qué orden o magnitud estará el aislamiento del material a medir para saber cuales son los ajustes más adecuados del instrumental a las condiciones reales que se esperan.

AislamientoPreparativo.m.- en este se ingresan todos los parámetros de ajuste fino como son los voltajes de referencia, frecuencia de muestreo, límites de graficación, etc. con lo que esté listo para hacer las mediciones en cada uno de los puntos.

AislamientoPA.m, AislamientoPB.m, ...'AislamientoPF.m, 'AislamientoPA2.m, ... AislamientoPF2.m.- son doce archivos uno para cada combinación de posición de micrófono y fuente. Permite adquirir los datos en ambas cámaras al mismo tiempo, generando el archivo de datos y, si el usuario lo quiere, puede ver cual es el aislamiento sonoro en ese punto específico o bien terminar con el programa y seguir con el siguiente punto de medición.

CalculoPABCDEF2.m.- recopila todas las adquisiciones de los doce puntos anteriores y evalúa el aislamiento final desplegando tablas y gráficas con desviación estándar.

Todos estos programas fueron probados a detalle para que entregaran resultados confiables. El detalle de cómo se utilizan se da a conocer en el procedimiento de medición y la codificación se encuentra a final de este trabajo.

3.5. Ajuste de instrumental de medición.

El ajuste de los dispositivos se lleva a cabo bajo las siguientes condiciones de conexión que se ilustran en la figura 3.1.

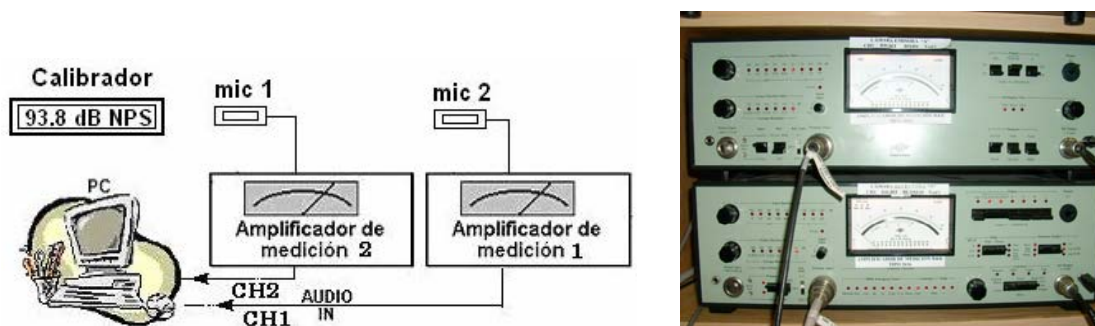


Figura 3.1. Conexión de dispositivos para la adquisición de niveles de presión sonora con la tarjeta de sonido de una PC para la calibración del instrumental.

Para que los datos que se obtengan y procesen tengan resultados calibrados, lo primero será establecer qué niveles de señal es capaz de recibir la tarjeta de sonido. Primero se ajustan los amplificadores de medición utilizando un calibrador que entrega a los micrófonos un Nivel de Presión Sonora de 93.8dB girando el potenciómetro *adjust* con un desarmador.

El amplificador de medición entrega a la salida un voltaje o nivel de señal desconocido que será recibido por la PC, entonces se usa el programa: *RMS94dBcal2CHs.m.* (descrito anteriormente) procediendo a captar la señal del un calibrador por un tiempo de 5 segundos, de esta forma se puede ver la forma de onda capturada y su valor *RMS*.

Nota importante: Cabe mencionar que el voltaje de salida del amplificador de medición está en función de sus controles de sensibilidad y un indicativo de variación de nivel es la posición de la aguja de la carátula, así que en caso de que los controles estén muy altos, es seguro que la aguja no se moverá ya que la escala es muy alta para indicar el *NPS* de 93.8dB que está recibiendo el micrófono, de cualquier manera a la salida se tiene una señal muy pequeña de valor desconocido por el momento pero que no se da uno cuenta de que existe por lo que se recomienda ajustar el equipo de tal manera que la aguja muestre desplazamiento a media escala.

El control de ganancia de la línea de entrada es el que gobierna el intervalo dinámico de entrada de la señal que se graba y lo ideal es que se posicione hasta el máximo.

Para simular diferentes niveles de entrada se puede girar la perilla de descalibración para aplicar mayor voltaje a la PC. En la figura se ilustra con una pantalla del programa *Cool Edit Ver. 2.0.*, cómo se apreciaría esta variación de nivel, en control de entrada (*Wave in*) y una carátula del amplificador de medición.

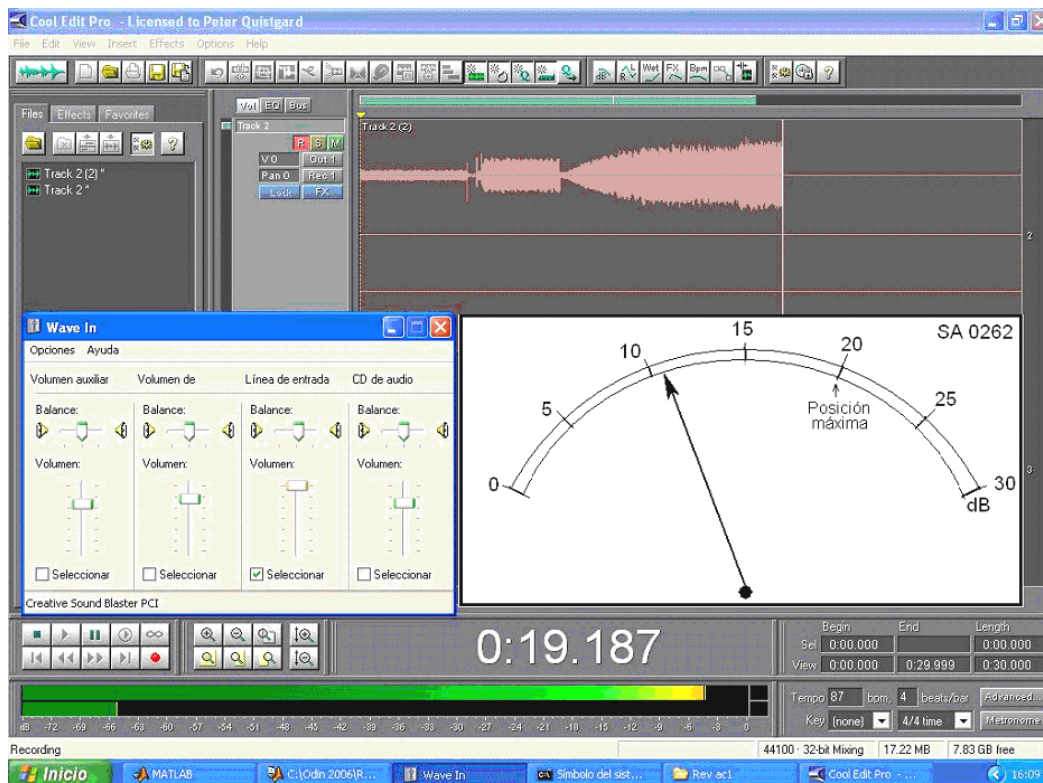


Figura 3.2. Representación de variación de nivel, control de nivel de entrada y carátula de amplificador de medición bajo condiciones adecuadas de nivel para adquirir una señal aleatoria.

Con los controles del amplificador de medición haciendo que el indicador rojo de escala esté en 80 con la aguja posicionada a nivel 13.8 indica que tenemos en el *NPS* de 93.8dB , lo cual se traduce en un voltaje para la tarjeta de sonido.

Si se aumenta la deflexión de la aguja con la perilla de descalibración¹, se puede conocer cual sería la posición máxima de la aguja para que se obtuviera un voltaje de saturación. Experimentalmente se encontró que si la aguja llegaba al número 20 con un tono puro de 1KHz , se tendría un voltaje de salida que comenzaría a saturar la tarjeta de sonido, por lo que se puede determinar que para cualquier medición se debe procurar que la aguja no rebase ese nivel para evitar mediciones distorsionadas y daños a la *PC*. El valor pico de la señal de tono continuo que puede ser captada sin distorsión fue de 0.81 , lo cual equivale a un valor *RMS* de 0.57 .

¹ No olvide regresar la perilla a la posición original hasta que escuche un clip cuando termine sus pruebas de experimentación de nivel, de lo contrario sus lecturas serán erróneas.

Una situación importante es: cuando se trata de señales de ruido aleatorio, con el mismo nivel de aguja se presenta saturación, por lo que se bajó en nivel de aguja 10dB por debajo del nivel de saturación de tono continuo, de esta forma se garantizaba que ningún pico del ruido rebasara en nivel de 0.81 .

Por supuesto el *RMS* para esa nueva señal es 10dB más pequeño y aunque la aguja muestre una posición más baja que el tono continuo, no puede manifestar en una fracción de segundo que existen variaciones que casi saturan la lectura y esto se debe a la inercia del mecanismo de resorte, sin embargo eléctricamente si se detecta esa variación con la computadora.

Para propósitos de calibración con el tono continuo se procede a tomar el *RMS* entregado ante un *NPS* de 93.8dB , que se refleja con una posición de aguja de 13.8 , con lo cual se está tomando el mayor *RMS* sin saturar; se observa que si la aguja estuviera en 3.8 , se registrarían 10dB menos de voltaje y de *RMS* a la tarjeta de sonido, por lo que el error tiende a ser ligeramente mayor. Así que se recomienda tomar el nivel más alto que se pueda.

Finalmente los datos importantes que se toman son:

- Un *RMS* que indica un posicionamiento de aguja de 13.8 correspondiente a 93.8dB de *NPS*
- Una señal de saturación para la tarjeta de sonido con un posicionamiento de aguja de 20 , para tono continuo de 1KHz con valor pico de 0.81 y $\text{RMS}=0.57$ y para ruido aleatorio con una posición de aguja 10dB menor al anterior.

Estos valores son la clave del ajuste del equipo utilizado y para cualquier otro que se utilice, como por ejemplo un sonómetro, que entrega una señal de *AC*.

Para mostrar la flexibilidad que aporta esta información se muestra el siguiente ejemplo:

Suponga que quiere medir *NPS* de ruido aleatorio que están en el orden de 58dB y se quiere ajustar el equipo para que se recabe la información con el menor error posible.

El primer paso es ajustar los controles de sensibilidad del amplificador de medición, de tal manera que la aguja no rebase el número 10 y esto se logra haciendo que en el indicador rojo de escala de la carátula se ilumine en la posición 50 , así que la aguja se encontrará alrededor del número 8 (indicando 58dB).

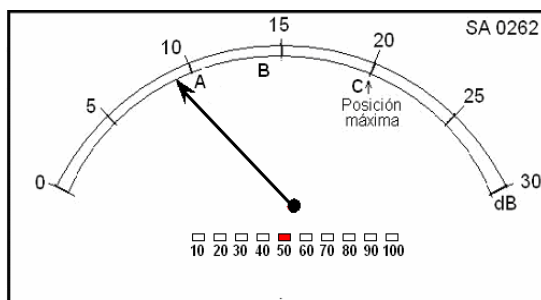


Figura 3.3. Carátula del amplificador de medición indicando un *NPS* de 58dB con la escala correspondiente y las posiciones críticas de aguja.

El *RMS* que se debe considerar es el que se obtiene al tener la aguja en 13.8 (posición *B*) o sea 0.285 que equivale a un *NPS* de 63.8dB, estas dos últimas cifras son las que se deben introducir en los programas subsecuentes de medición.

Con este ejemplo se sabe que la adquisición no tendrá valores saturados porque la aguja no rebasa la posición *A*.

Los valores importantes son los que se dan de la posición *B* que dicen: que se tendrá un *RMS* de 0.285 cuando se tenga un *NPS* de 63.8dB.

En la posición *C* se sabe que si se rebasa este límite en tono continuo, habrá saturación; mientras que en señales aleatorias, no se debe rebasar la posición *A*.

El ajuste del instrumental de medición, es en la sección más tardada en el proceso de medición del aislamiento sonoro, ya que se deben investigar los niveles que se pueden manejar, dependiendo de las características del equipo disponible, que no siempre será el mismo.

3.6 Procedimiento de medición.

Para evaluar el aislamiento sonoro con la técnica de *NPS* primeramente debe acondicionarse las cámaras emisora y receptora de tal manera que los tiempos de reverberación cumplan con la norma *ISO 140-3 [1]* de tal manera que se cumpla:

$$1s \leq T_{rev} \leq 2 \cdot (V/50)^{2/3} \quad (3-1)$$

donde *V* es el volumen del recinto que se quiera evaluar de tal manera que se tiene:

Para la cámara receptora, $V = 68.3m^3$ Por lo tanto $1s \leq T_{rev} \leq 2.46s$
Para la cámara emisora, $V = 77.6m^3$ Por lo tanto $1s \leq T_{rev} \leq 2.68s$

Para acondicionar el recinto se dispuso a colocar material absorbente en las esquinas para tratar de aminorar los efectos de las ondas estacionarias de los modos de vibración con la práctica de ensayo y error.



Foto 3.3 Acondicionamiento acústico de la cámara de transmisión con material absorbente.

Para medir los tiempos de reverberación se utilizaron Secuencias de Máxima Longitud (MLS^2) donde la respuesta al impulso del recinto se utiliza para obtener la curva de decaimiento. Para ello se monta el equipo como se muestra a continuación.

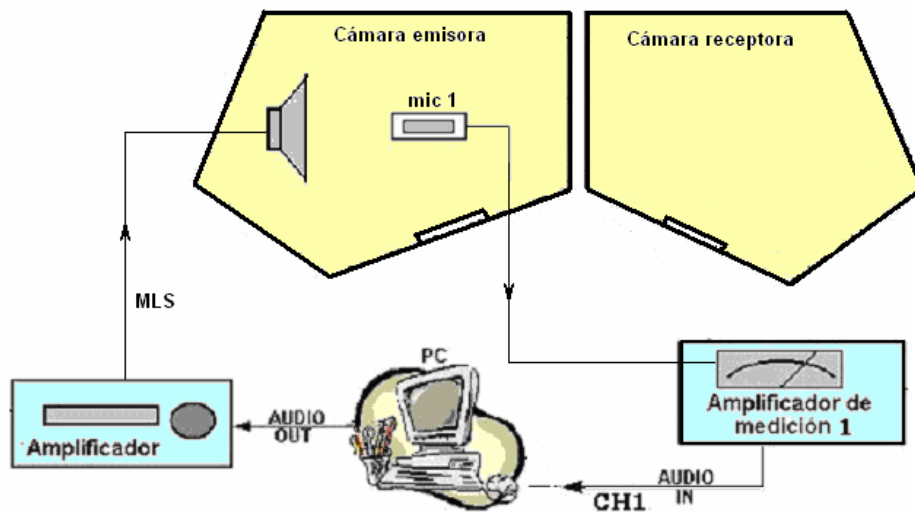


Figura 3.4. Conexión de instrumental para medir el tiempo de reverberación con MLS en la cámara emisora y la cámara receptora

La sección del amplificador consta de un filtro pasa banda, un preamplificador y un ecualizador.



Foto 3.4 Instrumental empleado en el exterior de la cámara de transmisión para adquirir y procesar las señales de las cámaras emisora y receptora.

Se debe medir por separado cada una de las cámaras utilizando solamente el canal 1 (izquierdo).

El NPS que registre el amplificador de medición deberá estar alrededor de $100dB$ con el propósito de que la curva de decaimiento sea lo más grande posible.

² Vease sección 2.2.2.

Por tratarse de una señal pseudoaleatoria³ se debe procurar que la aguja en la carátula no rebase el número 10.

Para la generación de la secuencia *MLS* estos son los ajustes recomendados y es producto de múltiples ensayos en donde se puede el porcentaje del intervalo dinámico debe estar entre el 70 y 90%. Fuera de estos límites, se tiene una curva de decaimiento más corta.

```

C:\reverb>mlsloop -r 24000 -c 2 -o 0 -m 18 -n 10 -g 35 -f test2418a
mlsloop: Block 1, dynamic range 73%.
mlsloop: Block 2, dynamic range 73%.
mlsloop: Block 3, dynamic range 73%.
mlsloop: Block 4, dynamic range 73%.
mlsloop: Block 5, dynamic range 73%.
mlsloop: Block 6, dynamic range 73%.
mlsloop: Block 7, dynamic range 73%.
mlsloop: Block 8, dynamic range 73%.
mlsloop: Block 9, dynamic range 73%.
mlsloop: Block 10, dynamic range 73%.

C:\reverb>mlsloop -r 24000 -c 2 -o 0 -m 18 -n 10 -g 35 -f test2418b
mlsloop: Block 1, dynamic range 73%.
mlsloop: Block 2, dynamic range 73%.
mlsloop: Block 3, dynamic range 73%.
mlsloop: Block 4, dynamic range 73%.
mlsloop: Block 5, dynamic range 74%.
mlsloop: Block 6, dynamic range 74%.
mlsloop: Block 7, dynamic range 73%.
mlsloop: Block 8, dynamic range 73%.
mlsloop: Block 9, dynamic range 73%.
mlsloop: Block 10, dynamic range 73%.

C:\reverb>

```

Figura 3.5. Ventana de MS DOS que muestra los comandos para la generación de las MLS con los parámetros de ajuste.

Para esta ventana se aprecia que se genera el archivo *test2418b.dat* con una frecuencia de muestreo de *24000Hz*, la respuesta impulsiva se procesa con el archivo *RevBdeF.m* introduciendo el nombre del archivo que se quiere evaluar. Las gráficas correspondientes se muestran a continuación.

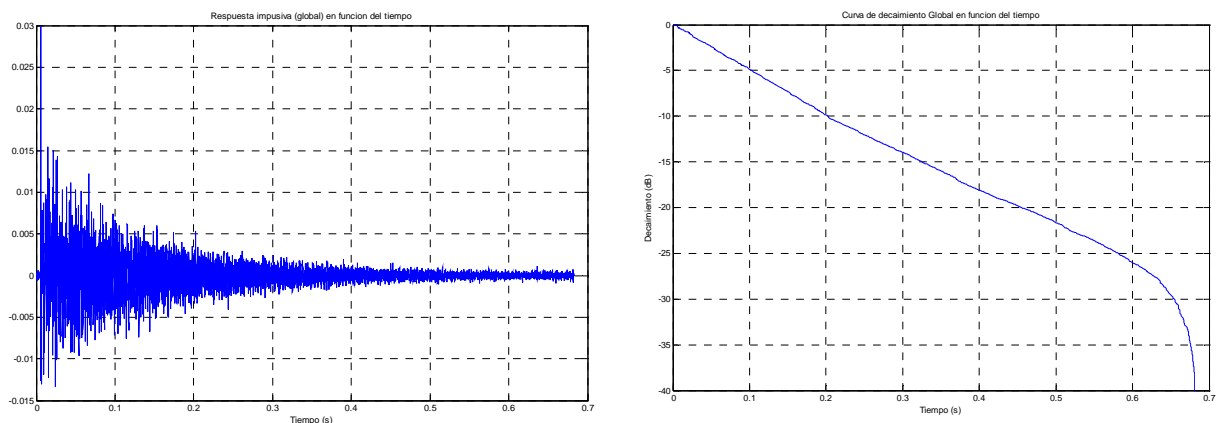


Figura 3.6. Respuesta impulsiva de un recinto y su curva de decaimiento.

A partir de la curva de decaimiento, se obtiene el tiempo de reverberación correspondiente, indicando el nivel en *dB* más bajo que se pueda antes que la curva empiece a doblarse y este punto se tomará como base para extrapolar la gráfica y calcular en qué tiempo hubiera caído la señal hasta *60dB* en línea recta.

³ El término pseudoaleatoria se refiere a la reproducción de una señal aleatoria de cierta duración y contenido de frecuencia, que se repite un número determinado de veces.

En este caso particular el nivel adecuado para seleccionar es de 25 debido a que antes de llegar a este punto, la caída es casi recta o pareja, después de este punto la información no es confiable.

Este caso es para evaluar el tiempo de reverberación de manera global pero lo que se requiere es evaluar los tiempos de reverberación por tercios de octava, por lo que se procede es filtrar la respuesta impulsiva, desde la banda de 100Hz hasta la de 5KHz; obteniendo un total de 18 bandas y cada respuesta impulsiva filtrada tiene su respectivo nivel en dB de curva de decaimiento, por lo que es necesario que se anote cada valor con la frecuencia correspondiente.

El programa dispone de una matriz en la cual se introducen estos valores y aquí se muestra una ventana que ilustra lo anterior.

```

2 %la curva de decaimiento para que posteriormente se ocupe el programa rt_xr
3 %y se mida el tiempo de reverberacion por babdas de frecuencia
4 close all
5 clear
6 clc
7
8 disp('ESTE PROGRAMA CALCULA EL TIEMPO DE REVERBERACION EN UN SOLO LUGAR')
9 disp('CARGA EL ARCHIVO QUE INDIQUE *.DAT Y LO ANALIZA EN BANDAS DE FRECUENCIA')
10 disp('*')
11 pause(1)
12 disp('ESPERE CARGANDO Y PROCESANDO DATOS ...')
13
14 load B1.dat %Introduzca el archivo a examinar <nombredearchivo.dat>
15 y=B1; %Vuelva a escribir el nombre <y=nombredearchivo>
16
17 Tmax=4;
18 ValorDeca=-40;%Limite inferior de graficacion para cada respuesta impulsiva
19 fs=24000;
20 h=20;% h= limite inferior seleccionado para la curva de decaimiento global en dB
21 % 100 125 160 200 250 315 400 500 630 800 1000 1250 1600 2000 2500 3150 4000 5000
22 hs=[10 15 15 15 20 20 20 20 20 20 20 20 20 20 20 20 20 20];%Limites inferior
23
24 ts=1/fs;
25 t=0:ts:length(y)/fs;
26 tmen1=t(1:length(t)-1);%Vector tiempo con una muestra menos para graficar la respuesta impulsiva
27 tmen2=t(1:length(t)-2);%Vector tiempo con dos muestras menos para poder graficar mas adelante la curv

```

Figura 3.7. Ventana de programa en Matlab que muestra la sección de carga de archivo *B1.dat* y la introducción de variables *Tmax*, *ValorDeca*, *fs*, *h* y *hs*

La variable *h* tiene el valor de decaimiento global y la variable *hs* debe tener los valores que uno considera por cada banda de frecuencia en orden ascendente. Como guía se pone en la línea superior un desplegado de frecuencias en modo de comentario (verde) para que no falte o sobre algún valor. Observe que se está cargando el archivo *B1.dat* y la variable *y* debe llevar el mismo nombre. La variable *ValorDeca* puede cambiarla con propósitos de graficación y facilitar la rapidez de visualización.

Una vez terminado ésto se tiene que correr nuevamente el programa y así tendrá los tiempos de reverberación por bandas de tercios de octava. De esta manera puede verificar que cada respuesta impulsiva se está detectando correctamente. Si las gráficas salen muy distintas a lo normal, entonces vuelva a emitir la MLS del ensayo, para evitar el efecto relativo de circularidad de la convolución.

Se debe usar una MLS de orden suficiente para ver el decaimiento pero también, no debe ser de orden muy grande, en donde se vea que transcurre mucho tiempo después del punto de elección del decaimiento ya que el resto de la señal se requiere y solamente se desperdicia memoria, además se pierde facilidad de visualización para seleccionar el punto de decaimiento.

Para la evaluación del aislamiento sonoro, se debe calcular el tiempo de reverberación promedio en la cámara receptora, por lo que se procede a medir bajo 12 condiciones o combinaciones de micrófono y fuente sonora, lo cual implica tener 12 archivos de datos .dat que tendrán los nombres $B1$, $B2$, $B3$, ... , $B12$.

Para el caso de la cámara emisora no se requiere tener los valores de tiempos de reverberación para propósitos de cálculo de aislamiento, pero se debe evaluar, para garantizar que los tiempos estén bajo norma. A los archivos generados para el recinto emisor tendrán los siguientes nombres: $A1$, $A2$, $A3$, ... , $A12$.

Una vez obtenidos los archivos de la cámara receptora, se procede a procesarlos con el programa *RevBdeFtodo.m* que funciona de la misma manera que el *RevBdeF.m*, con la diferencia de que las curvas de decaimiento se presentarán de manera conjunta por frecuencia, en otras palabras, se presentarán en una sola gráfica las 12 curvas de decaimiento, correspondientes a 100Hz, en otra gráfica se visualizarán las 12 correspondientes a 125Hz, etcétera, hasta llegar 5KHz. En cada una de estas gráficas se debe seleccionar un punto en común que compartan las doce curvas de una frecuencia en específico.

En la figura 3.8 se muestran un ejemplo de 12 curvas de decaimiento para la frecuencia de 1KHz y por extrapolación se obtiene el tiempo de reverberación en esa banda de frecuencia.

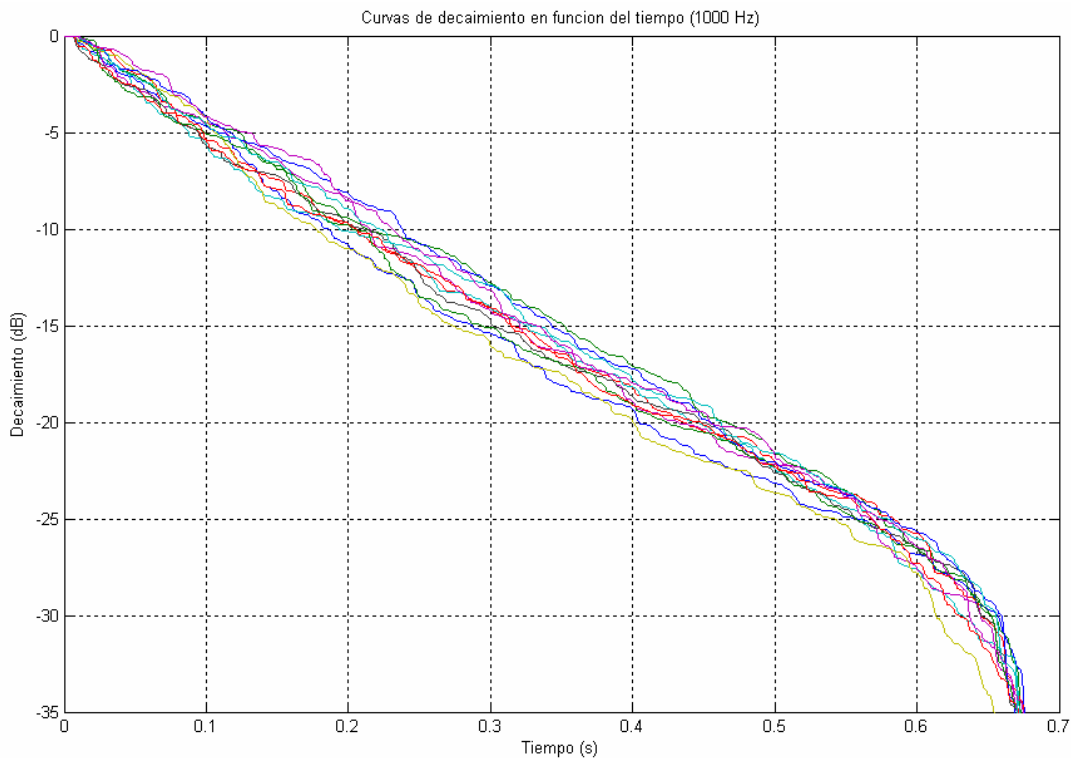


Figura 3.8. Curvas de decaimiento de respuestas impulsivas filtradas en la banda de 1KHz con un punto de decaimiento común de 25dB.

Los valores se deben vaciar en la variable *hs*, de la misma manera antes descrita y, así, se tendrá un promedio de los tiempos de reverberación, por cada cámara indicando la desviación estándar. Para el caso de la cámara receptora, se genera automáticamente un archivo llamado *rtprom.mat*, que contiene los tiempos de de reverberación promedio, que se requieren para la evaluación del aislamiento, el mismo que se carga automáticamente cuando se inicia los preparativos de esta medición.

Evaluación del aislamiento sonoro

Para evaluar finalmente el aislamiento sonoro, se procede a anotar los valores RMS de cada canal, estableciendo el NPS correspondiente a la posición de la aguja.

En el programa *AislamientoPreparativo1.m* deberá introducirse los valores que se solicitan, como se muestra en la figura 3.9.

En este caso se observa que para el canal 1 (transmitido) se trabajará con 53.8dB con un *RMS* de 0.286 y para el canal 2 (incidente) con 103.8dB con un *RMS* de 0.276 . Además se graficará el aislamiento con una escala vertical que va de los 10 a 70dB y los *NPS* incidentes y transmitidos, se graficarán con una escala de 10 a 90dB . Se introduce el área de espécimen, el volumen de los dos recintos, la frecuencia de muestreo y el tiempo de la adquisición de los datos, que se ha establecido en dos minutos por cada ensayo.

```

18 - disp('AislamientoPF2.m')
19 - disp(' TAMBIEN SE CARGA EL VECTOR CON LOS TIEMPOS DE REVERBERACION QUE SE CALCULARON CON EL PROGRAMA: R
20 - disp('SE ACONSEJA VERIFICAR CADA UNA DE LAS VARIABLES IGRESADAS AL PROGRAMA')
21 - %*****INDICAR dB DEL SONOMETRO Y VrefRMS MEDIDO PARA CH1 Y CH2*****
22 - dBref1=53.8;%Nivel que indico el sonometro al realizar la calibracion en CH1
23 - Vref1=0.286;%Voltaje medido para el NPS verificado con con el sonometro CH1
24 - dBref2=103.8;%Nivel que indico el sonometro al realizar la calibracion en CH2
25 - Vref2=0.276;%Voltaje medido para el NPS verificado con con el sonometro en CH2
26 - ymin=10;%Valor limite inferior en dB para que sea graficar el aislamiento.
27 - ymax=70;
28 - NPSymin=10;%Valor limite inferior en dB para que sea graficar los niveles de ruido.
29 - NPSymax=90;
30 - V2=77.6;%Volumen de recinto emisor (m3)
31 - V1=68.3;%Volumen del recinto receptor (m3)
32 - Sesp=9.75;%Area de especimen (m2)
33 - fs=22050;%Frecuencia de muestreo
34 - duracion=10;%Duracion en segundos de los disparos
35 - minutos=2;%Tiempo en minutos de la duracion de la prueba de adquisicion por cada posicion
36 - Notriggers=minutos*60/duracion;%Numero de disparos realizados en la prueba
37 - %*****
38 - %*****
39 - load rtprom.mat
40 - %rtprom=[1; 2; 3; 4; 5; 6; 7; 8; 9; 10; 11; 12; 13; 14; 15; 16; 17; 18];
41 -
42 - for z=1:18
43 - AABCDEF(z)=0.163*V1/rtpromCENAM(z);%Absorcion equivalente promedio que sera aplicada a cada ensayo par
44 - NABCDEF(z)=10*log10(Sesp/AABCDEF(z));%Componente global para aplicarse a la STL final

```

Figura 3.9. Ventana con el programa de Matlab *AislamientoPreparativo1.m* en donde se introducen los valores que utilizarán todos los programas subsecuentes en la medición de aislamiento.

Se puede correr el programa *AislamientoPreliminar.m* para saber en qué orden se encontrarán los resultados finales, rastreando en una sola posición de elementos, sin olvidar el introducir los valores de las variables antes mencionadas.

Ahora mida el aislamiento en 12 posiciones o combinaciones diferentes de micrófono y fuente según la norma *ISO 140-3* [1], sin olvidar aplicar ruido aleatorio en la cámara emisora con *100dB* de *NPS* aproximadamente.

Corra los 12 archivos: *AislamientoPA.m*, *AislamientoPB.m*, ...'*AislamientoPF.m*', '*AislamientoPA2.m*, ... *AislamientoPF2.m*., en donde cada uno es para cada combinación, no olvidando correr el programa *AislamientoPreparativo.m* antes de cada ensayo y siga las instrucciones del programa pulsando "Enter" cuando se le pida.

Para recabar toda la información de todos los ensayos, se corre el programa: *CalculoPABCDEF2.m* y se espera a que se desarrolle el procesamiento y así se verán las graficas y tablas finales.

De esta manera se tiene el aislamiento medido con esta técnica, sin la necesidad de un analizador de espectro.

NIVELES DE PRESIÓN SONORA INCIDENTES Y TRANSMITIDOS CON STD

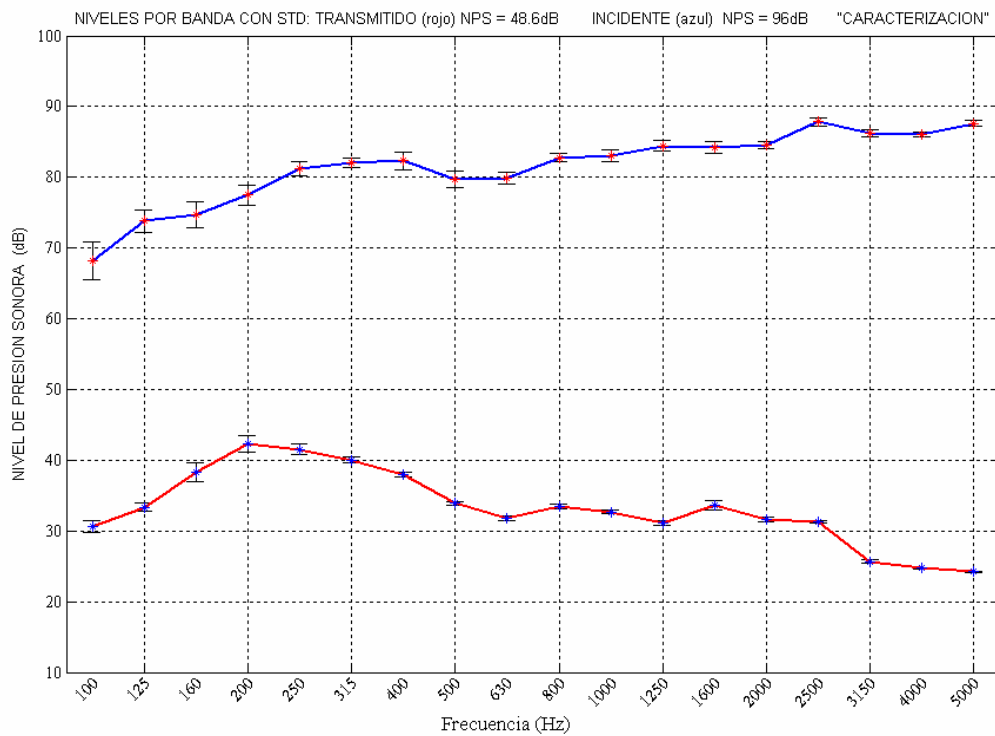


Figura 3.11. NPS calculados con desviación estándar para el recinto emisor y receptor por bandas de frecuencia de tercios de octava en la caracterización de la cámara de transmisión.

GRÁFICA DE AISLAMIENTO CON STD

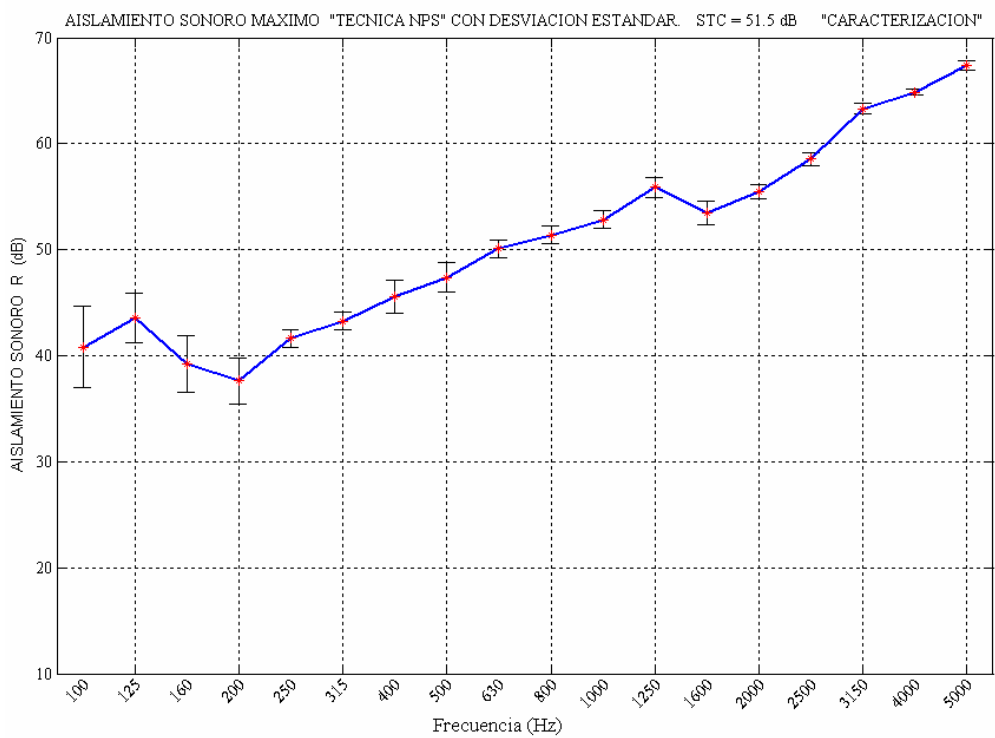


Figura 3.12 Curva de aislamiento sonoro con desviación estándar y STC del muro en la caracterización de la cámara de transmisión.

*
 TABLA DE RESULTADOS AISLAMIENTO SONORO MAXIMO "CARACTERIZACION"

| Frecuencia | NPS(dB)CH1 | NPS(dB)CH2 | 10log(S/A) | Aislamiento (dB) |
|------------|------------|------------|------------|------------------|
| 100.0 | 30.6 | 68.2 | 3.2 | 40.8 |
| 125.0 | 33.4 | 73.8 | 3.1 | 43.6 |
| 160.0 | 38.3 | 74.7 | 2.8 | 39.3 |
| 200.0 | 42.4 | 77.5 | 2.6 | 37.6 |
| 250.0 | 41.5 | 81.3 | 1.9 | 41.6 |
| 315.0 | 40.0 | 82.0 | 1.3 | 43.3 |
| 400.0 | 38.0 | 82.3 | 1.2 | 45.5 |
| 500.0 | 33.9 | 79.7 | 1.6 | 47.4 |
| 630.0 | 31.7 | 79.9 | 1.9 | 50.1 |
| 800.0 | 33.4 | 82.7 | 2.1 | 51.4 |
| 1000.0 | 32.7 | 83.1 | 2.5 | 52.8 |
| 1250.0 | 31.1 | 84.4 | 2.6 | 55.9 |
| 1600.0 | 33.6 | 84.2 | 2.9 | 53.4 |
| 2000.0 | 31.6 | 84.6 | 2.5 | 55.5 |
| 2500.0 | 31.3 | 87.8 | 2.1 | 58.6 |
| 3150.0 | 25.7 | 86.2 | 2.8 | 63.3 |
| 4000.0 | 24.7 | 86.0 | 3.6 | 64.9 |
| 5000.0 | 24.3 | 87.6 | 4.0 | 67.4 |

*

Tabla 3.2. Tabla con valores calculados con el *NPS CH1* (recinto receptor), el *NPS CH2* (recinto emisor), factor de compensación $10\log(S/A)$ y el aislamiento final.

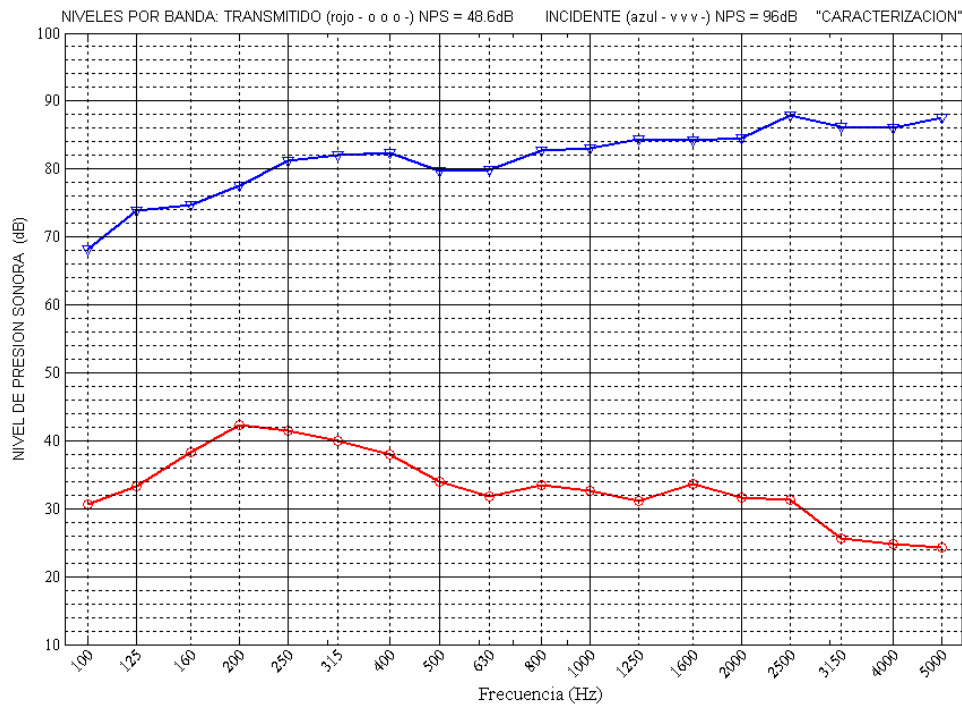


Figura 3.13. NPS calculados para el recinto emisor y receptor por bandas de frecuencia de tercios de octava en la caracterización de la cámara de transmisión.

```

PROGRAMA TERMINADO
IMPORTANTE: SE CREARON LOS ARCHIVOS AislamientodBNPS.mat y STDaislamientosNPS.mat
Los tiene que copiar a la carpeta de Recopilacion
*
* * * * *
IMPORTANTE 2: Y TAMBIEN SE CREARON LOS ARCHIVOS ValoresdBltodo.mat , ValoresdB2todo.mat , BdeF.mat
STDNPS1.mat Y STDNPS2.mat & dB1005KCH1.mat Y dB1005KCH2.mat
PARA QUE LOS PUEDA CARGAR Y GRAFICAR DE NUEVO POR SI SE QUERE DARLE OTRA PRESENTACION O AGREGAR INFORMACION
*
Tambien se grabaron los archivos ValoresdBltodo.mat y ValoresdB2todo.mat
Por si quiere utilizarlos para graficar VS BdeF en otro momento
  
```

Figura 3.14. Mensaje desplegado al final del programa que informa sobre los archivos generados y las instrucciones para ser utilizados en otro procesamiento.

TEMPOS DE REVERBERACIÓN CÁMARA RECEPTORA

El tiempo de reverberación se debe encontrar en estos intervalos:

$$1s \leq T_{rev} \leq 2*(V/50)^{2/3} \quad (3-2)$$

Para la cámara receptora, $V = 68.3m^3$

Por lo tanto $1s \leq T_{rev} \leq 2.46s$

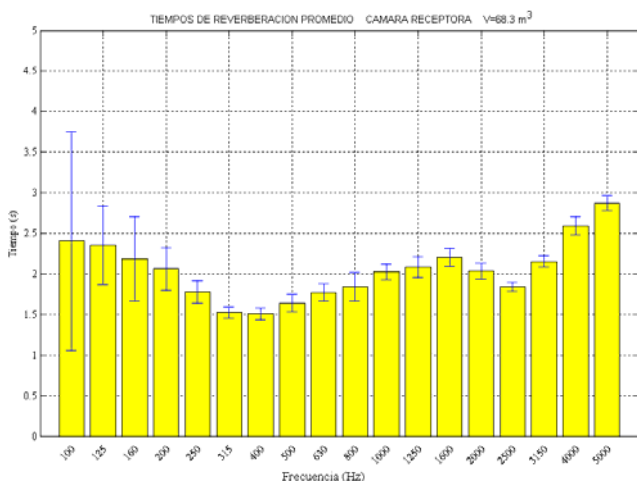


Figura 3.15 Tiempos de reverberación calculados con desviación estándar de la cámara receptora en la caracterización de la cámara de transmisión

"TABLA DE RESULTADOS CON PROMEDIOS"

| Frecuencia | Trev(seg) | STD |
|------------|-----------|------|
| 100.00 | 2.41 | 1.35 |
| 125.00 | 2.35 | 0.48 |
| 160.00 | 2.18 | 0.52 |
| 200.00 | 2.06 | 0.26 |
| 250.00 | 1.78 | 0.14 |
| 315.00 | 1.52 | 0.07 |
| 400.00 | 1.50 | 0.07 |
| 500.00 | 1.64 | 0.10 |
| 630.00 | 1.77 | 0.10 |
| 800.00 | 1.84 | 0.17 |
| 1000.00 | 2.03 | 0.10 |
| 1250.00 | 2.08 | 0.13 |
| 1600.00 | 2.20 | 0.11 |
| 2000.00 | 2.03 | 0.10 |
| 2500.00 | 1.84 | 0.06 |
| 3150.00 | 2.15 | 0.07 |
| 4000.00 | 2.59 | 0.11 |
| 5000.00 | 2.87 | 0.09 |

Tabla 3.3. Tabla con resultados de los tiempos de reverberación medidos en la cámara receptora por bandas de frecuencia de tercios de octava con desviación estándar.

Nota: estos tiempos son los que se utilizan para calcular el área de absorción equivalente.

Para la cámara emisora, $V = 77.6m^3$ Por lo tanto $1s \leq T_{rev} \leq 2.68s$

TIEMPOS DE REVERBERACIÓN CÁMARA EMISORA

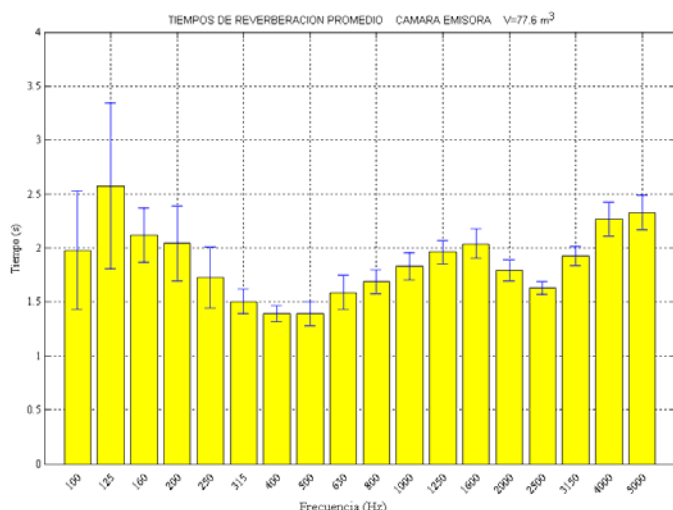


Figura 3.16 Tiempos de reverberación calculados con desviación estándar de la cámara emisora en la caracterización de la cámara de transmisión

"TABLA DE RESULTADOS CON PROMEDIOS"

| Frecuencia | Trev(seg) | STD |
|------------|-----------|------|
| 100.00 | 1.98 | 0.55 |
| 125.00 | 2.57 | 0.77 |
| 160.00 | 2.12 | 0.25 |
| 200.00 | 2.04 | 0.35 |
| 250.00 | 1.73 | 0.28 |
| 315.00 | 1.51 | 0.11 |
| 400.00 | 1.39 | 0.07 |
| 500.00 | 1.39 | 0.11 |
| 630.00 | 1.59 | 0.16 |
| 800.00 | 1.69 | 0.11 |
| 1000.00 | 1.83 | 0.13 |
| 1250.00 | 1.96 | 0.11 |
| 1600.00 | 2.04 | 0.14 |
| 2000.00 | 1.79 | 0.10 |
| 2500.00 | 1.63 | 0.06 |
| 3150.00 | 1.92 | 0.09 |
| 4000.00 | 2.27 | 0.15 |
| 5000.00 | 2.33 | 0.16 |

Tabla 3.4. Tabla con resultados de los tiempos de reverberación medidos en la cámara emisora por bandas de frecuencia de tercios de octava con desviación estándar.

RUIDO DE FONDO CON LOS APARATOS APAGADOS

Este es el ruido de fondo que tienen las cámaras en ausencia de aparatos o bien sin energizar.

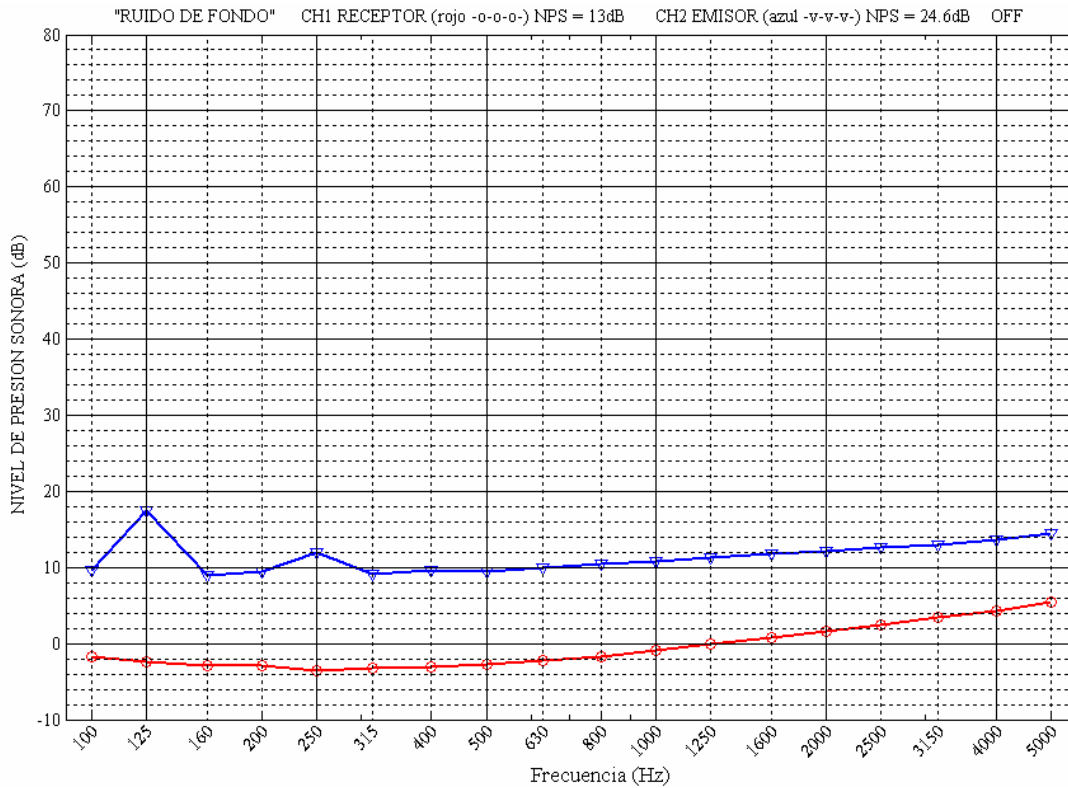


Figura 3.17. NPS de ruido de fondo calculados para el recinto emisor y receptor por bandas de frecuencia de tercios de octava en la caracterización de la cámara de transmisión con todos los dispositivos de medición apagados.

NPS Global en CH1 = 19.2 dB
NPS Global en CH2 = 26.5 dB.

NPS Banda 100-5KHz CH1 = 13.0 dB.
NPS Banda 100-5KHz CH2 = 24.6 dB.

```

*****RUIDO MEDIDO DE CH1 Y CH2*****
          APARATOS APAGADOS
Frecuencia  NPS(dB)CH1  NPS(dB)CH2  Aislamiento (dB)
100.0       -1.7        9.7         11.4
125.0       -2.5       17.4        19.9
160.0       -2.9        8.9        11.8
200.0       -2.9        9.4        12.3
250.0       -3.5       12.0        15.5
315.0       -3.3        9.1        12.4
400.0       -3.0        9.6        12.6
500.0       -2.8        9.5        12.3
630.0       -2.3        9.9        12.2
800.0       -1.7       10.4        12.0
1000.0      -0.9       10.8        11.7
1250.0      -0.1       11.3        11.4
1600.0       0.8       11.8        11.0
2000.0       1.7       12.2        10.5
2500.0       2.5       12.6        10.1
3150.0       3.4       13.0         9.6
4000.0       4.4       13.6         9.3
5000.0       5.4       14.4         9.0
    
```

Tabla 3.5. Tabla con valores de los NPS por bandas de frecuencia de tercios de octava para el recinto receptor (CH1) y el emisor (CH2) y el aislamiento que se tiene entre estos dos canales con todos los dispositivos de medición apagados.

RUIDO DE FONDO CON LOS APARATOS APAGADOS

Se puede tener otra presentación de los resultados.

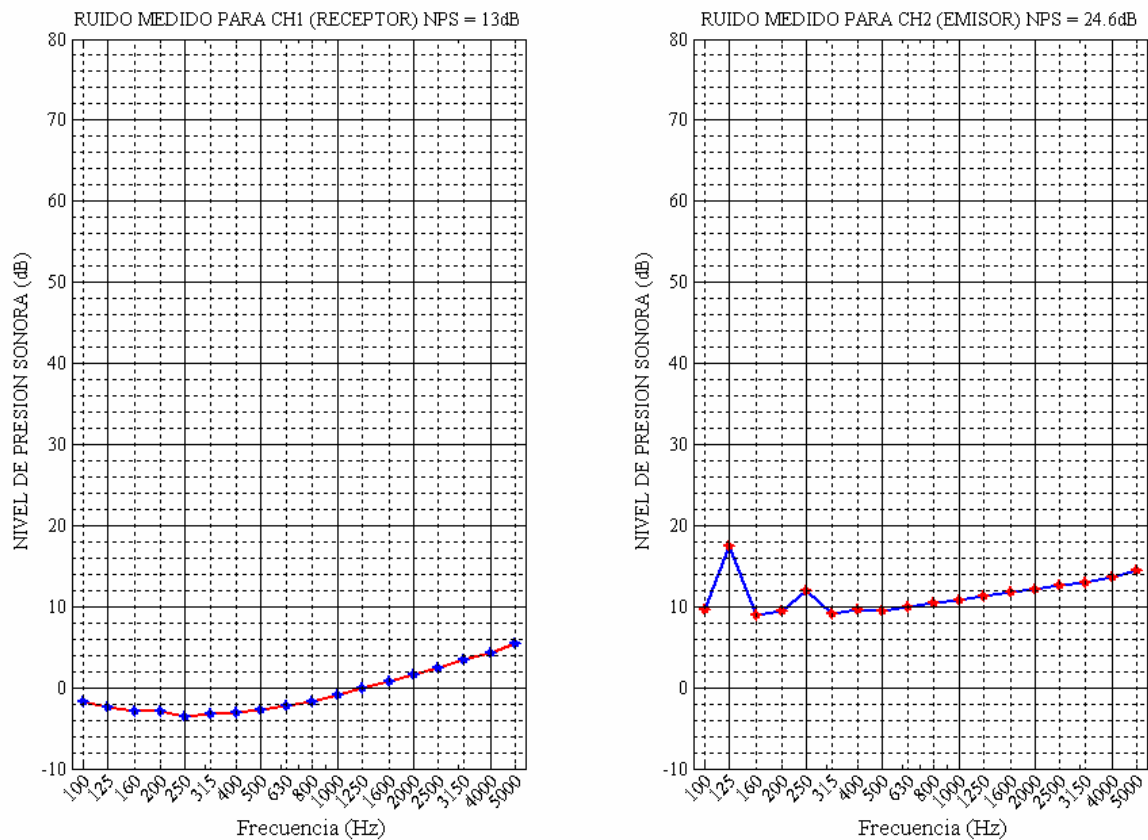


Figura 3.18. NPS de ruido de fondo calculados para el recinto receptor y emisor en gráficas separadas por bandas de frecuencia de tercios de octava en la caracterización de la cámara de transmisión con todos los dispositivos de medición apagados.

NPS Global en CH1 = 19.2 dB.
NPS Global en CH2 = 26.5 dB.

NPS Banda 100-5KHz CH1 = 13.0 dB.
NPS Banda 100-5KHz CH2 = 24.6 dB.

```

*****RUIDO MEDIDO DE CH1 Y CH2*****
          APARATOS APAGADOS
Frecuencia  NPS (dB)CH1  NPS (dB)CH2  Aislamiento (dB)
100.0      -1.7          9.7          11.4
125.0      -2.5          17.4         19.9
160.0      -2.9          8.9          11.8
200.0      -2.9          9.4          12.3
250.0      -3.5          12.0         15.5
315.0      -3.3          9.1          12.4
400.0      -3.0          9.6          12.6
500.0      -2.8          9.5          12.3
630.0      -2.3          9.9          12.2
800.0      -1.7          10.4         12.0
1000.0     -0.9          10.8         11.7
1250.0     -0.1          11.3         11.4
1600.0     0.8           11.8         11.0
2000.0     1.7           12.2         10.5
2500.0     2.5           12.6         10.1
3150.0     3.4           13.0          9.6
4000.0     4.4           13.6          9.3
5000.0     5.4           14.4          9.0
    
```

Tabla 3.6. Tabla con valores de los NPS por bandas de frecuencia de tercios de octava para el recinto receptor (CH1) y el emisor (CH2) y el aislamiento que se tiene entre estos dos canales con todos los dispositivos de medición apagados.

RUIDO DE FONDO CON LOS APARATOS ENCENDIDOS

Este es el ruido de fondo que tienen las cámaras con los aparatos encendidos pero con los controles de ganancia al mínimo emitiéndose exclusivamente el ruido intrínseco de los dispositivos.

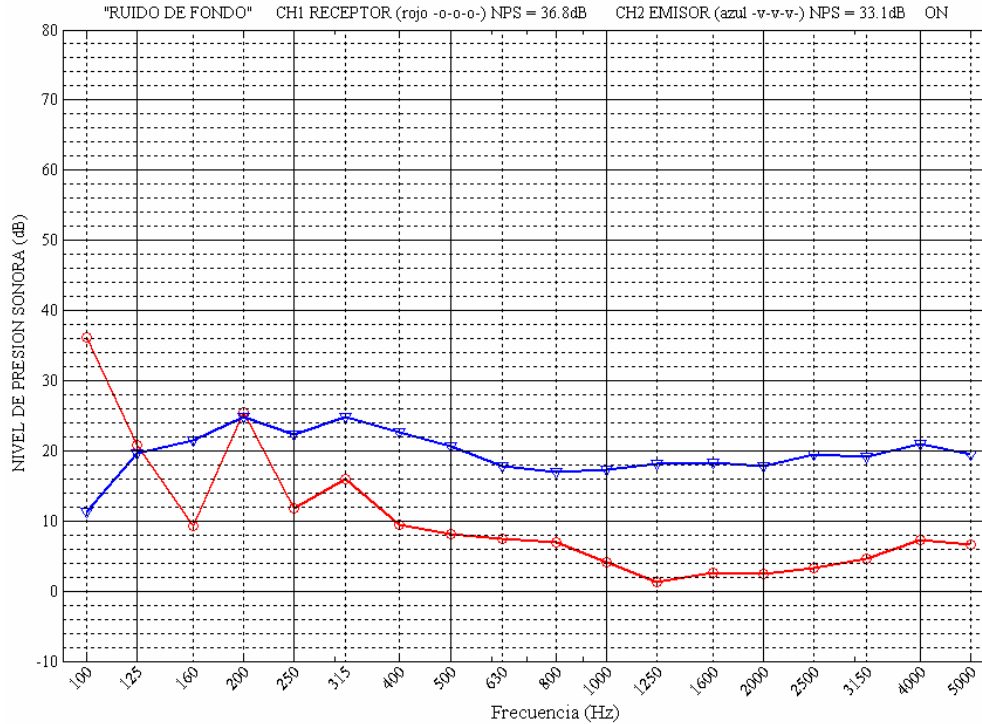


Figura 3.19. NPS de ruido de fondo calculados para el recinto emisor y receptor por bandas de frecuencia de tercios de octava en la caracterización de la cámara de transmisión con todos los dispositivos de medición encendidos.

Nota: En el recinto receptor la jirafa está rotando con 128 segundos/revolución.

NPS Global en CH1 = 37.3 dB.
NPS Global en CH2 = 33.5 dB..

NPS Banda 100-5KHz CH1 = 36.8 dB.
NPS Banda 100-5KHz CH2 = 33.1 dB.

```

*****RUIDO MEDIDO DE CH1 Y CH2*****
          APARATOS ENCENDIDOS
Frecuencia  NPS (dB) CH1  NPS (dB) CH2  Aislamiento (dB)
100.0      36.2         11.3         -25.0
125.0      20.8         19.6          -1.2
160.0       9.3         21.4         12.1
200.0      25.6         24.7          -0.8
250.0      11.8         22.4         10.6
315.0      16.0         24.8           8.8
400.0       9.5         22.6         13.1
500.0       8.2         20.7         12.5
630.0       7.4         17.8         10.4
800.0       7.0         17.0         10.0
1000.0      4.1         17.4         13.3
1250.0      1.2         18.1         16.9
1600.0      2.6         18.3         15.7
2000.0      2.4         17.7         15.3
2500.0      3.3         19.4         16.2
3150.0      4.5         19.2         14.6
4000.0      7.2         20.9         13.7
5000.0      6.7         19.5         12.9
*****
    
```

Tabla 3.7. Tabla con valores de los NPS por bandas de frecuencia de tercios de octava para el recinto receptor (CH1) y el emisor (CH2) y el aislamiento que se tiene entre estos dos canales con todos los dispositivos de medición encendidos.

RUIDO DE FONDO CON LOS APARATOS ENCENDIDOS

Se tiene otra presentación de resultados.

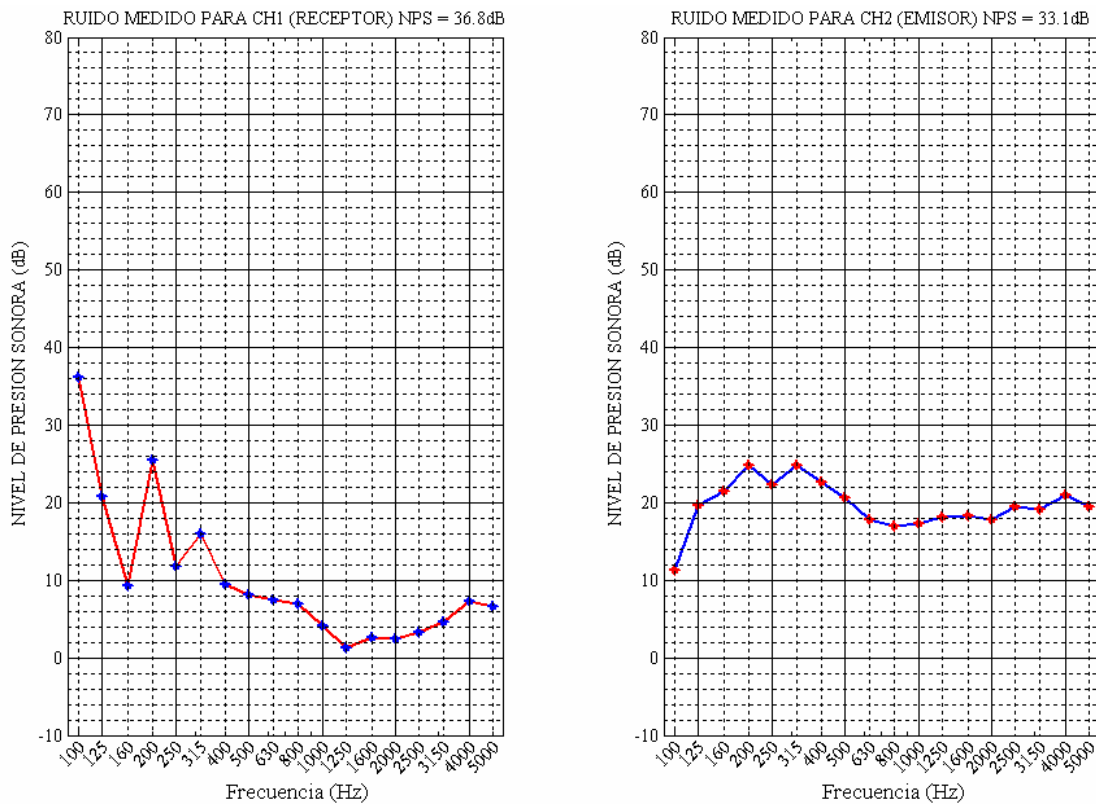


Figura 3.20. NPS de ruido de fondo calculados para el recinto receptor y emisor en gráficas separadas por bandas de frecuencia de tercios de octava en la caracterización de la cámara de transmisión con todos los dispositivos de medición encendidos.

NPS Global en CH1 = 37.3 dB.
NPS Global en CH2 = 33.5 dB.

NPS Banda 100-5KHz CH1 = 36.8 dB.
NPS Banda 100-5KHz CH2 = 33.1 dB.

```

*****RUIDO MEDIDO DE CH1 Y CH2*****
                APARATOS ENCENDIDOS
Frecuencia    NPS (dB) CH1    NPS (dB) CH2    Aislamiento (dB)
100.0         36.2           11.3             -25.0
125.0         20.8           19.6              -1.2
160.0          9.3           21.4              12.1
200.0         25.6           24.7              -0.8
250.0         11.8           22.4              10.6
315.0         16.0           24.8               8.8
400.0          9.5           22.6              13.1
500.0          8.2           20.7              12.5
630.0          7.4           17.8              10.4
800.0          7.0           17.0              10.0
1000.0         4.1           17.4              13.3
1250.0         1.2           18.1              16.9
1600.0         2.6           18.3              15.7
2000.0         2.4           17.7              15.3
2500.0         3.3           19.4              16.2
3150.0         4.5           19.2              14.6
4000.0         7.2           20.9              13.7
5000.0         6.7           19.5              12.9
    
```

Tabla 3.8. Tabla con valores de los NPS por bandas de frecuencia de tercios de octava para el recinto receptor (CH1) y el emisor (CH2) y el aislamiento que se tiene entre estos dos canales con todos los dispositivos de medición encendidos.

RUIDO DE FONDO CÁMARA RECEPTORA CON EL BOOM ENCENDIDO CON DIFERENTES VELOCIDADES

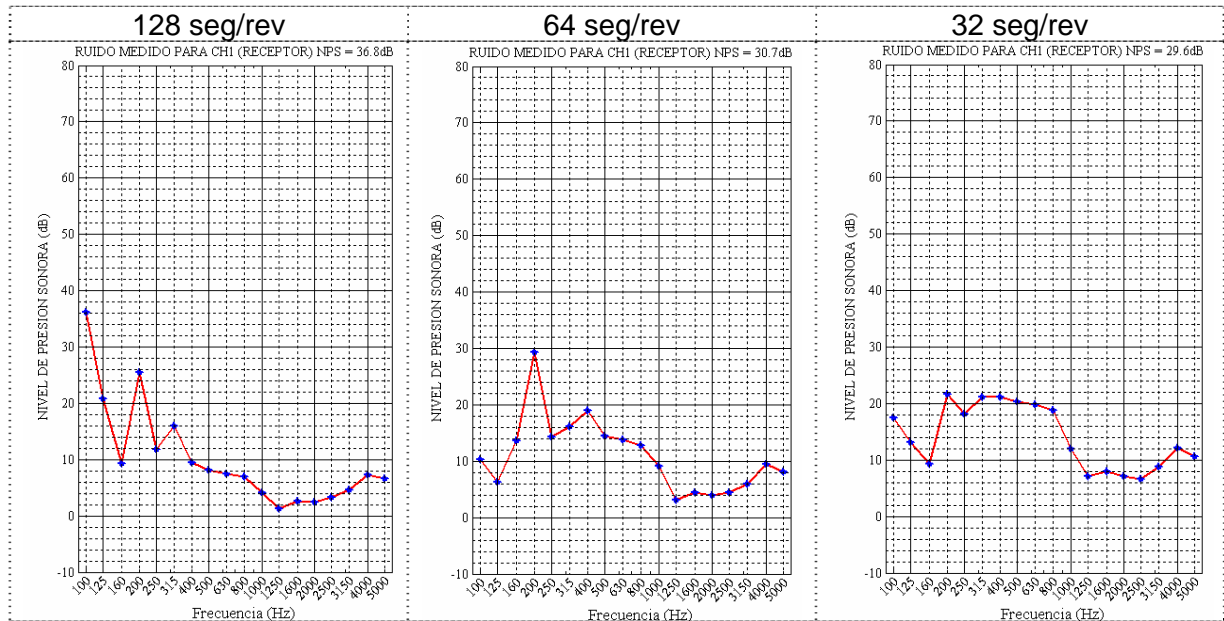


Figura 3.21. Curvas de ruido de fondo en el recinto receptor bajo tres velocidades diferentes de rotación del boom.

| NPS Global en CH1 = 37.3 dB NPS Banda 100-5KHz CH1 = 36.8 dB | | NPS Global en CH1 = 34.9 dB. NPS Banda 100-5KHz CH1 = 30.7 dB | | NPS Global en CH1 = 36.8 dB. NPS Banda 100-5KHz CH1 = 29.6 dB | |
|---|---------|--|---------|--|---------|
| Frecuencia | NPS(dB) | Frecuencia | NPS(dB) | Frecuencia | NPS(dB) |
| 100 | 36.2 | 100 | 10.4 | 100 | 17.5 |
| 125 | 20.8 | 125 | 6.3 | 125 | 13.2 |
| 160 | 9.3 | 160 | 13.6 | 160 | 9.4 |
| 200 | 25.6 | 200 | 29.4 | 200 | 21.6 |
| 250 | 11.8 | 250 | 14.3 | 250 | 18.1 |
| 315 | 16.0 | 315 | 16.2 | 315 | 21.2 |
| 400 | 9.5 | 400 | 18.9 | 400 | 21.2 |
| 500 | 8.2 | 500 | 14.4 | 500 | 20.4 |
| 630 | 7.4 | 630 | 13.8 | 630 | 19.9 |
| 800 | 7.0 | 800 | 12.8 | 800 | 18.9 |
| 1000 | 4.1 | 1000 | 9.1 | 1000 | 12.0 |
| 1250 | 1.2 | 1250 | 3.0 | 1250 | 7.1 |
| 1600 | 2.6 | 1600 | 4.4 | 1600 | 8.0 |
| 2000 | 2.4 | 2000 | 3.9 | 2000 | 7.1 |
| 2500 | 3.3 | 2500 | 4.4 | 2500 | 6.7 |
| 3150 | 4.5 | 3150 | 6.0 | 3150 | 8.7 |
| 4000 | 7.2 | 4000 | 9.4 | 4000 | 12.1 |
| 5000 | 6.7 | 5000 | 8.2 | 5000 | 10.6 |

Tabla 3.9. Tabla con valores de los NPS de ruido de fondo por tercios de octava del recinto receptor bajo tres velocidades diferentes de rotación del boom.

COMPARACIÓN DE CURVAS DE RUIDO DE FONDO

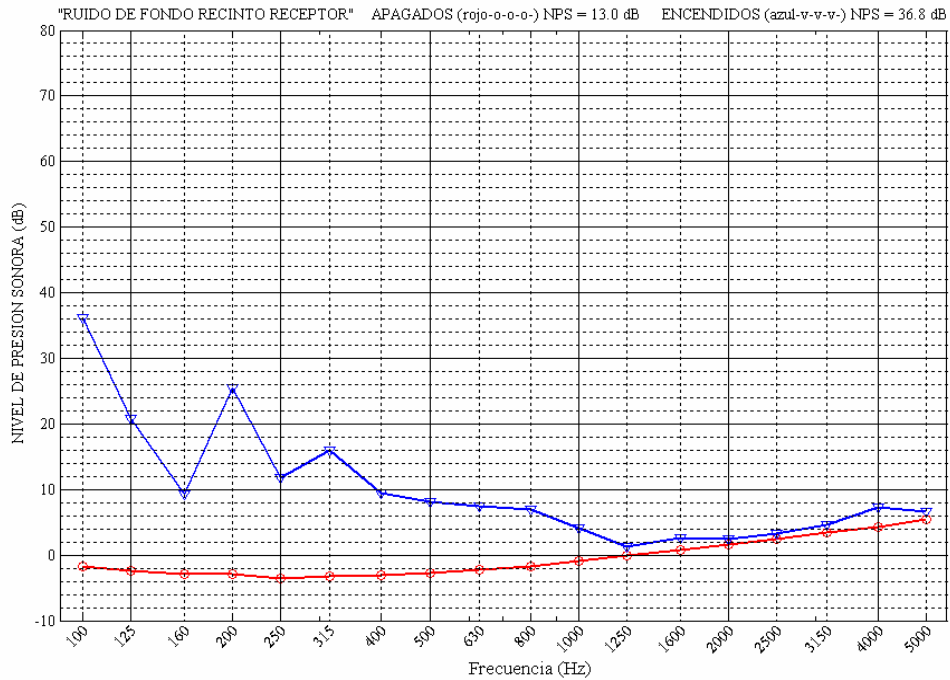


Figura 3.22. Curvas de ruido de fondo calculado para el recinto receptor donde se observa los niveles de ruido con los aparatos apagados y el aumento de ruido cuando los aparatos son encendidos.

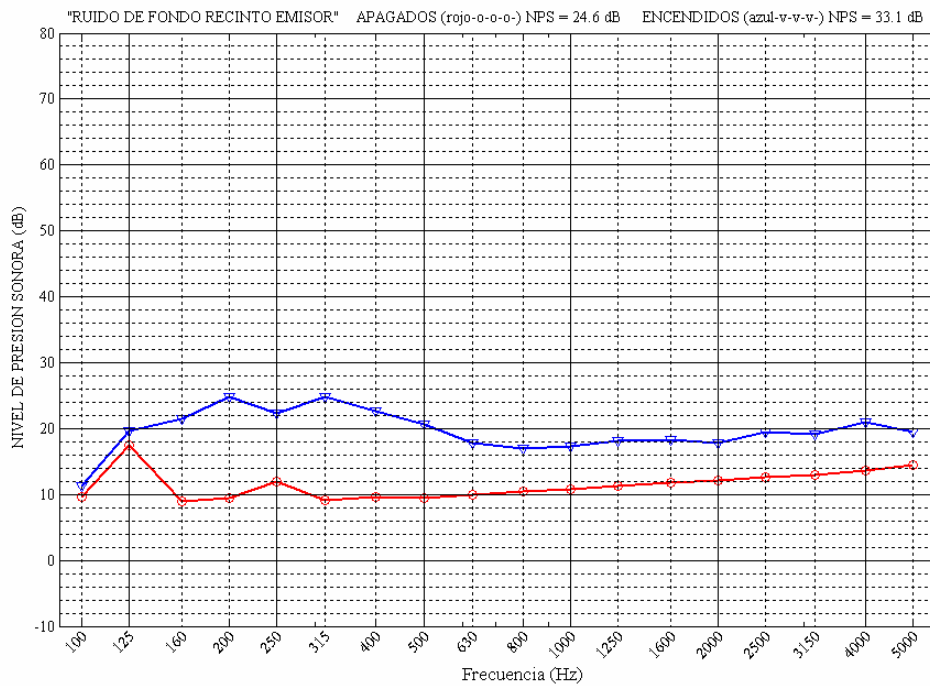


Figura 3.23. Curvas de ruido de fondo calculado para el recinto emisor donde se observa los niveles de ruido con los aparatos apagados y el aumento de ruido cuando los aparatos son encendidos.

De estos resultados se puede ver que la transmisión pro flanqueo no contribuye a las mediciones de aislamiento, por lo que las mediciones con futuros materiales pueden ser confiables.

IV DESARROLLO EXPERIMENTAL DE MEDICIÓN CON LA TÉCNICA DE CEPSTRUM

4.1. Técnica Cesptrum.

Esta técnica de medición *"in situ"*, permitirá medir el aislamiento sonoro de un material sin la necesidad de instalaciones especializadas como lo es la cámara de transmisión. En vez de ello, se usa un montaje en un recinto semireverberante en donde se coloca la muestra verticalmente, para hacer incidir una señal de prueba que se transmitirá por ella y en función de las magnitudes incidente y transmitida, se evaluará el aislamiento en donde se eliminarán en efecto de los reflejos provenientes de las superficies más cercanas, logrando medir valores libres de ecos que contaminar la medición. Para lograrlo se recurre al cálculo del *Cepstrum*, que básicamente permite saber en qué momento la señal capturada presenta repetición de la señal o sea que se detectó un reflejo, así se puede eliminar aquella parte de la señal contaminada por ecos.

4.2. Planteamiento del problema

Para la medición con esta técnica se propone un montaje como el que se muestra en la siguiente figura.

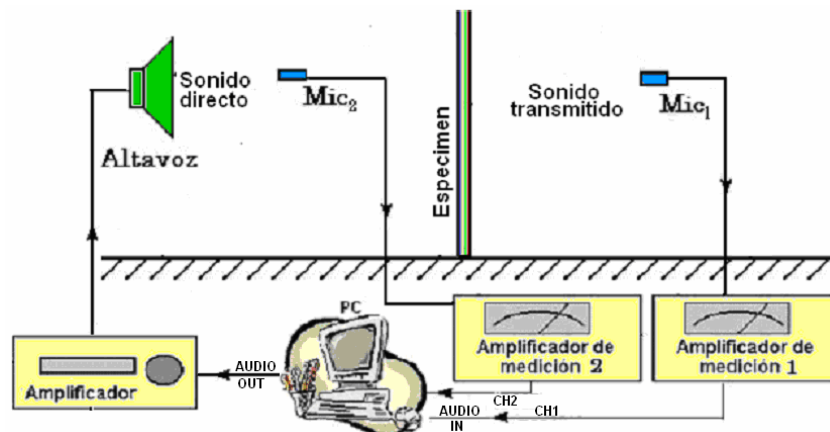


Figura 4.1. Diagrama de conexión para la medición de aislamiento sonoro con técnica *Cepstrum*.



Foto 4.1. Montaje de la fuente sonora y el micrófono para la medición del sonido incidente en el espécimen.



Foto 4.2. Montaje del micrófono para la medición del sonido transmitido por el espécimen.



Foto 4.3. Vista superior del montaje de dispositivos para la medición in situ.



Foto 4.4. Vista lateral del montaje de dispositivos con acercamiento al espécimen.

En la figura 4.1 y las fotografías 4.1 a 4.4 se ve una serie de dispositivos en donde se ocupará una señal de excitación generada por la computadora, que se hará incidir en el espécimen y así cada uno de los dos micrófonos detecten los sonidos incidentes y transmitidos respectivamente que serán adquiridos en los dos canales de entrada de audio de la PC.

Se planteó usar diferentes señales de excitación que pudieran generarse por la computadora, dentro de las señales ocupadas se experimentó con ráfagas de tonos puros, señales *chirp* que hacen un barrido en un intervalo lineal de frecuencia en un breve tiempo y ruido aleatorio. De las tres opciones se optó por utilizar la última ya que presentaba menores dificultades para procesar además de que el *Cepstrum* se manifestaba de manera más clara. Con las otras señales se llegan a presentar frecuentemente problemas de fase en donde se tenían casos de cancelación de amplitudes por lo que la detección de la repetición de las señales no se lograba captar correctamente.

Con *Matlab 6.5*, se hicieron numerosas pruebas, en las que se simulaba la presencia de una señal directa y sus reflejos a un tiempo previamente programado; por lo que al evaluar el *Cepstrum*, se debería apreciar con claridad la existencia de la primera reflexión en el tiempo programado y la ráfaga que presentó menor cantidad de problemas fue la de ruido aleatorio ya que es menos probable que se llegue a presentar un problema de fase.

En una de las múltiples pruebas realizadas, para saber como manipular las ráfagas, se obtuvo información crucial para realizar correctamente la programación, montaje y colocación de dispositivos de medida.

Se simuló una ráfaga de ruido aleatorio con una duración de *1ms* seguida de cuatro reflexiones con tiempos y amplitudes respecto a la incidente como se indica.

tretraso1 = 0.0008s;
tretraso2 = 0.0015s;
tretraso3 = 0.0020s;
tretraso4 = 0.0025s;

Amplitud de retraso 1 = 0.7; primer retraso
 Amplitud de retraso 2 = 0.5; segundo retraso
 Amplitud de retraso 3 = 0.4; tercer retraso
 Amplitud de retraso 4 = 0.3; cuarto retraso

Estas señales se presentan en el siguiente cuadro de figuras:

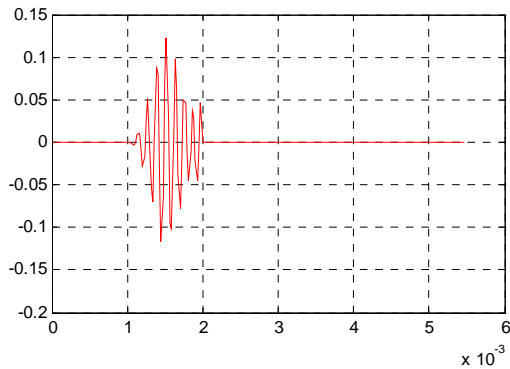


Figura 4.2-A. Ráfaga incidente simulada 1ms después del comienzo de la adquisición.

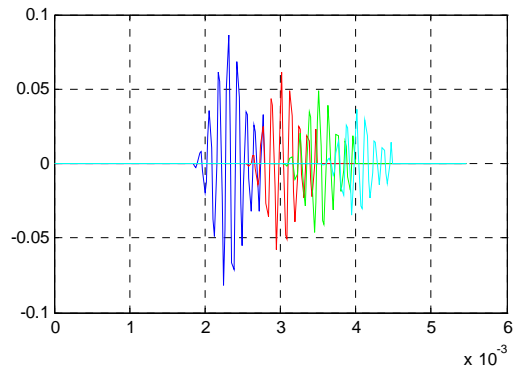


Figura 4.2-B. Repetición de ráfagas con diferentes tiempos de retardo y amplitud decreciente.

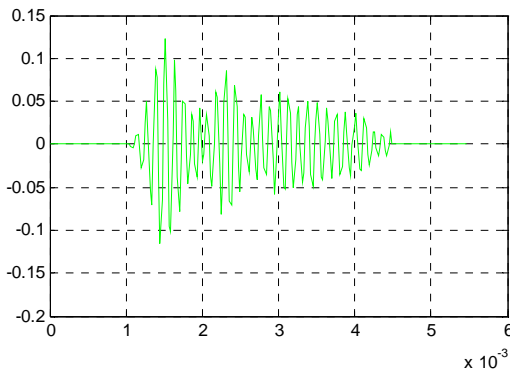


Figura 4.2-C. Suma de la ráfaga incidente y sus repeticiones (señal a procesar).

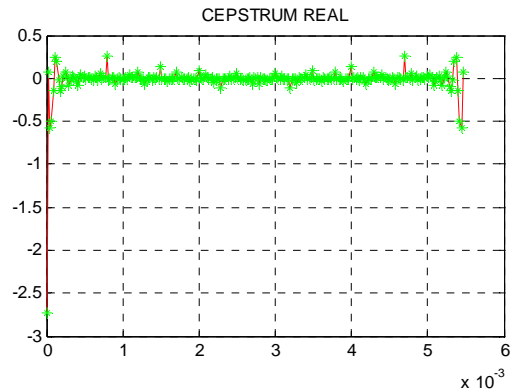


Figura 4.2-D. Gráfica del *Cepstrum* calculado de la señal simulada indicando la existencia de repeticiones de la ráfaga incidente.

En el conjunto de figuras 4.2-A - 4.2-D se presenta lo siguiente:

-En la figura A se encuentra la ráfaga incidente que comienza 1ms después, la idea de no colocar señal desde $t=0$ es que se está previendo que a la señal le tomará un tiempo para ir del altavoz hasta el micrófono.

-En la figura B se encuentran las cuatro reflexiones con sus retrasos y amplitudes correspondientes que se programaron.

-En la figura C se presenta la suma de las cinco señales anteriores tal como se presentaría en un caso real en donde no se aprecia en dónde exactamente están los reflejos.

-En la figura D se encuentra el *Cepstrum* que se obtuvo de la gráfica C y en él se ve una espiga en la parte positiva después $t=0$ en $t=0.0008$ con una amplitud que se puede reconocer lo cual coincide con el tiempo esperado que se programó.

Como consecuencia de estas simulaciones, se llegó a establecer los criterios para la duración y manejo de las ráfagas, así como del *Cepstrum* y son los siguientes:

- 1.- La ráfaga debe tener una duración mayor que el tiempo en el que pudiera llegar el primer reflejo, de lo contrario existen problemas, y eso se puede estimar dependiendo de la superficie más cercana de la que pudiera provenir el primer eco¹.
- 2.- No importa cuánto tiempo tarde en llegar la señal directa de cualquier manera en el *Cepstrum* se presentará en $t=0$, la existencia del sonido directo con una gran espiga y los reflejos con otras espigas de menor tamaño.
- 3.- El *Cepstrum* se muestra a manera de reflejo al final de su duración, que es igual al tiempo de captura de la señal que se analiza, por lo que se recomienda, que el tiempo de captura debe ser cuando menos del doble de la duración de la ráfaga, mas el tiempo que tarda en llegar al micrófono.
- 4.- Si el tiempo de adquisición es muy largo, no importa, el único inconveniente es que es más difícil visualizar las repeticiones sin acercamientos, además de que se consume más memoria.
- 5.- Debido a la presentación simétrica del *Cepstrum*, es necesario procesar solo la primera mitad para detectar la presencia de reflejos. Si se llega a considerar toda la gráfica se tendrán problemas para determinar acertadamente las repeticiones.

Este es un resumen de las pruebas y observaciones de los múltiples ensayos que se realizaron con las diferentes ráfagas de prueba.

Montaje de dispositivos

El montaje para esta técnica quedó como se muestra en la figura 4.3.

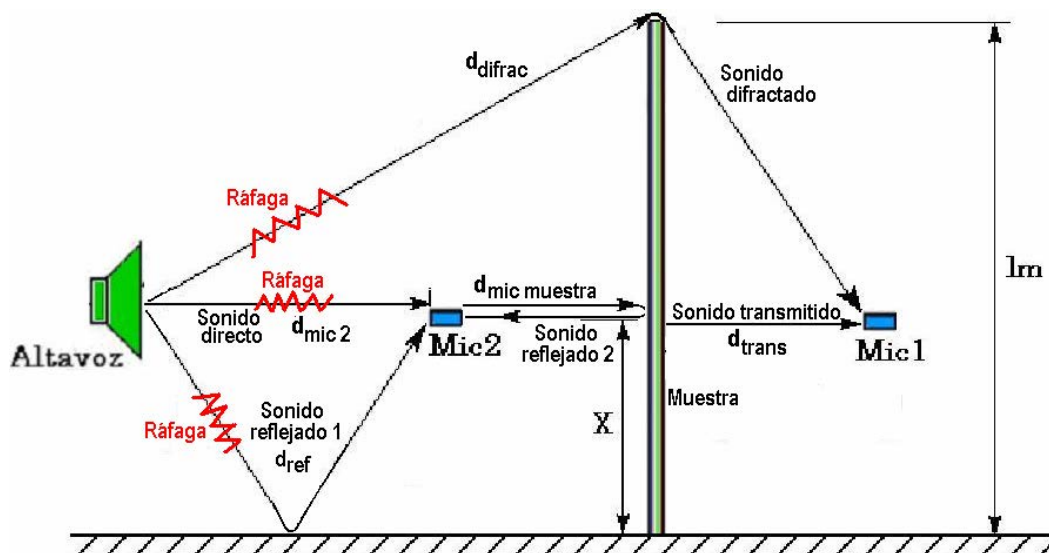


Figura 4.3. Montaje de dispositivos y trayectorias principales de la ráfagas para la medición de aislamiento sonoro con la técnica *Cepstrum*.

¹ Para más detalles sobre el cálculo del tamaño de la ráfaga consulte la sección sobre optimización de distancias.

La clave para el funcionamiento adecuado de la técnica, es que el montaje del instrumental permita la medición con las condiciones óptimas, de tal manera que, se aproveche al máximo la duración de las ráfagas y que estén libres de reflejos por el mayor tiempo posible.

En este montaje de la figura 4.3 se puede apreciar que: considerando la velocidad del sonido como c , que el *mic2* recibe una ráfaga de duración t_{raf} que tarda en llegar d_{mic2}/c segundos; pasando este tiempo, comienza a captarse la ráfaga con su duración correspondiente. La ráfaga reflejada por el piso tarda en llegar al *mic2* un tiempo de d_{ref}/c segundos, o sea que llega un poco después de ráfaga directa, este tiempo de diferencia es $(d_{ref}/c - d_{mic2}/c)=A$.

La ráfaga directa pasa por el *mic2* y finalmente llega a la muestra, reflejándose parte de ella, para nuevamente incidir en el *mic2* un tiempo $2d_{micmuestra}/c=B$ segundos después (porque es el doble de distancia, ida y vuelta).

Cualquiera de las dos ráfagas reflejadas que lleguen al *mic2*, será suficiente para contaminar la lectura del sonido directo, por lo que es necesario hacer que ambas lleguen al mismo tiempo, es decir, con el mismo retraso. De aquí se ve que el tiempo de recorrido de la señal reflejada por el piso, es el que dictará la distancia del *mic2* a la muestra, por las siguientes condiciones:

-La trayectoria del sonido reflejado por la muestra puede ser tan grande como pueda desplazarse horizontalmente el instrumental alejándose unos de otros. La trayectoria del sonido reflejado por el piso está limitada por la altura de la muestra, por lo que no se puede colocar el instrumental más allá de 1m, lo cual implica sacar los dispositivos del experimento.

Basados en estos dos puntos, se establece que el reflejo del piso es el que dictará las distancias óptimas a calcular, ya que el posicionamiento vertical es el que está limitado.

Se observa que mientras más grande es d_{mic2} , el retraso del reflejo disminuye, debido a que d_{mic2} y d_{ref} tienden a ser iguales.

Para el caso de la señal transmitida, ésta llegará teóricamente en $(d_{micmuestra} + d_{mic2} + d_{trans})/c$ segundos después y la ráfaga difractada llegará en d_{difrac}/c segundos o sea que tendrá un retraso de $C= d_{difrac}/c - (d_{micmuestra} + d_{mic2} + d_{trans})/c$ segundos. Este tiempo es el que se tarda en llegar la difracción una vez que llegó la ráfaga transmitida al *mic1*.

Se puede decir que si los tiempos A , B y C llegan a ser iguales, las condiciones de montaje serán óptimas ya que para el caso del *mic2*, los reflejos llegarán al mismo tiempo y se aprovecha al máximo la ráfaga. Para el *mic1* colocado a la misma altura, es necesario que la difracción llegue con el mismo retraso respecto al sonido transmitido. Para facilitar las lecturas se colocan todos los dispositivos a la misma altura.

Casos extremos.

En esta sección se muestran cuales serían los montajes llevados al extremo para que se entienda el proceso de optimización de distancias.

Siempre hay que procurar que las ráfagas reflejadas por el piso y muestra lleguen con el mismo retraso independientemente de la posición de los elementos.

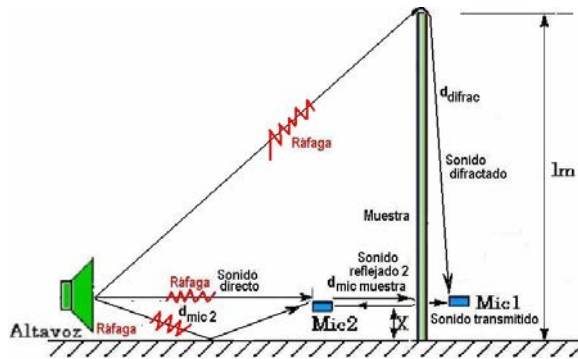


Figura 4.4. Montaje de dispositivos en donde el instrumental se encuentra casi sobre el piso con $X=0$.

Aquí se aprecia que el retraso de las ráfagas reflejadas por el piso y la muestra es casi nulo, o sea que la ráfaga directa se contamina de inmediato con reflexiones por lo que no se puede medir adecuadamente el sonido incidente.

En cambio el retraso de la ráfaga difractada es muy grande lo cual da un margen muy amplio para grabar la ráfaga transmitida.

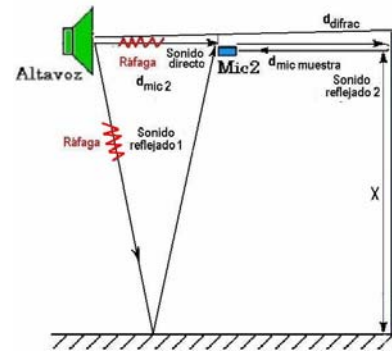


Figura 4.5. Montaje de los dispositivos en donde el instrumental se encuentra colocado a una altura de la muestra $X < l_m$.

Aquí se aprecia todo lo contrario, la ráfaga difractada llega casi al mismo tiempo que la transmitida, las ráfagas reflejadas tardan mucho más que el sonido incidente.

De estas condiciones extremas se puede inferir que existe una posición intermedia óptima que permite aprovechar la ráfaga incidente transmitida con la máxima fidelidad posible libre de de reflexiones y difracciones que contaminen la medición.

Observe que el micrófono receptor está muy cerca de la muestra, para que mida, exactamente, qué magnitud de señal logra transmitirse, además para el otro micrófono se prevé que la señal que capte de modo directo se atenuará por la distancia recorrida hacia la muestra (ley inversa de la distancia), así que se tendrá el valor de señal que incide en la muestra y cuánto se logra transmitir, de manera fiel.

La distancia que va a dictar las condiciones óptimas de montaje es d_{mic2} , ya que al hacer esta distancia más grande, la ráfaga incidente y reflejada por el piso tenderán a llegar al mismo tiempo; entonces, mientras d_{mic2} sea más pequeña, más lejano en tiempo se encontrará de llegar el sonido reflejado.

Pero sucede que al emitirse la ráfaga, la grabación conlleva un retraso de tiempo inherente al equipo de cómputo por lo que no se puede colocar el *mic2* pegado al altavoz ya que la ráfaga se grabará a destiempo por lo que hay que buscar la distancia d_{mic2} tal que el *mic2* comience a grabar justo cuando llegue la ráfaga; para ello hay que ir rastreando distancias, hasta encontrar la adecuada en la que se garantice que se va a captar la ráfaga, cuando ya esté reproducida y emitida plenamente; basado en esto, se tendrá el montaje óptimo, obteniendo un aprovechamiento máximo de la ráfaga, limpia libre de reflexiones y difracciones.

4.3 Programación en la PC con Matlab 6.5.

Para la adquisición de datos y ajustes del equipo se realizaron programas en Matlab que se mencionan a continuación, dando una breve explicación de lo que hace cada uno.

calculohoptim.m.- Este programa permite calcular las posiciones óptimas del equipo, introduciendo los datos de la *dmic2*, y temperatura ambiente. Entrega el valor de la altura de los dispositivos, duración mínima de la ráfaga y la ubicación de los dos micrófonos.

calniv2mics.m .- Este programa permite ajustar los amplificadores de medición, de tal forma que entreguen el mismo nivel de señal en los dos canales de entrada de audio bajo condiciones iguales de medición.

ApreparativoCepstrum.m.- en él se introducen los valores comunes de ajuste como son las distancias de elementos, temperatura ambiente, tamaño de la ráfaga, etc.

ACapturaCepstrum.m.- Permite emitir y capturar las ráfagas de ruido aleatorio con un número programado de ensayos.

AProcesamientoCepstrum.m.- sirve para procesar los datos obtenidos con el programa anterior y mostrar las fases de procesamiento y tablas y gráficas finales de aislamiento.

El propósito de que estén por separado es evitar el borrado de los datos adquiridos por error; cuando se quieran procesar en varios ensayos, basta con que se carguen los datos de preparación y luego se procesen los datos sin tocar el archivo que los captura.

Estos programas son producto de una simplificación de ensayos que se realizaron para determinar cómo se podía medir el aislamiento con mayor facilidad.

4.4 Ajuste de instrumental de medición.

A diferencia de la técnica *NPS*, en éste método no es necesario calibrar los amplificadores de medición con el calibrador de $93.8dB$ de *NPS*, lo que se tiene que hacer es colocar los dos micrófonos a una misma distancia, lo más lejanos posible de piso techo y paredes y emitir una ráfaga de ruido aleatorio, de tal forma que, los dos canales de la tarjeta de sonido presenten el mismo nivel de *RMS*, con los controles de sensibilidad iguales en ambos amplificadores de medición.

Con este ajuste, se comparan los niveles de las ráfagas detectadas y se evalúa el aislamiento sonoro en bandas de frecuencia.

El primer paso es determinar a que distancia deben estar colocados los micrófonos desde altavoz, para que la grabación comience exactamente cuando la ráfaga llegue. Para lograr esto, debe montar el equipo como se muestra en la figura 4.6.

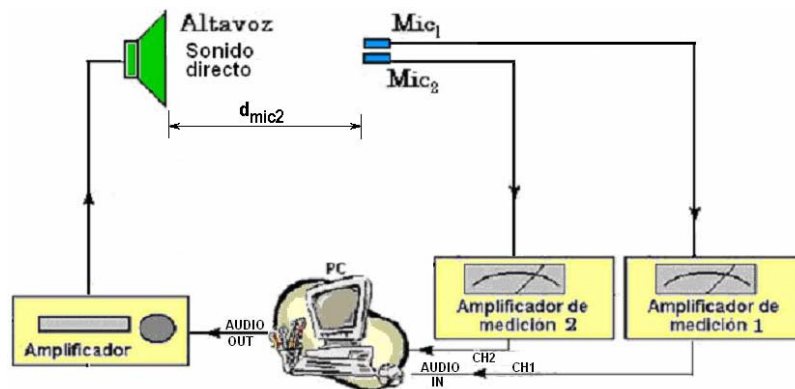


Figura 4.6. Montaje para ajustar los niveles de entrada de los canales de audio de la tarjeta de sonido

Se propone inicialmente que la distancia de micrófonos y altavoz sea de 1 metro y emitir la ráfaga de tono continuo y de ruido aleatorio con duración de $0.01s$ con el programa *calniv2mics.m*, la captura de la ráfaga se vería de esta forma.

Para el micrófono que grabará la señal transmitida se tienen las figuras 4.7-A – 4.7-C.

Vista normal

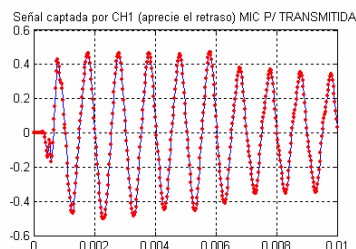


Figura 4.7-A. Señal transmitida y adquirida por el *mic1* a un metro de distancia de la fuente sonora.

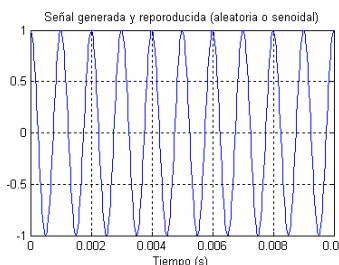


Figura 4.7-B. Señal generada por la PC que fue reproducida.

Acercamiento

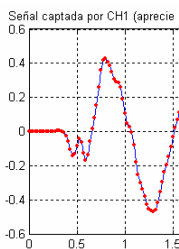


Figura 4.7-C. Acercamiento horizontal de la figura 4.7-A de la señal transmitida y adquirida por el *mic1*.

Para el micrófono que grabará la señal incidente se tienen las figuras 4.8-A – 4.8-C.

Vista normal

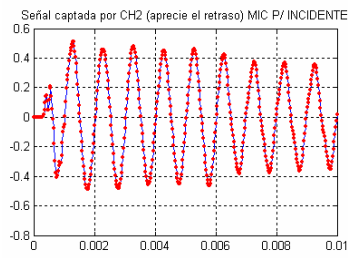


Figura 4.8-A. Señal incidente y adquirida por el mic2 a un metro de distancia de la fuente sonora.

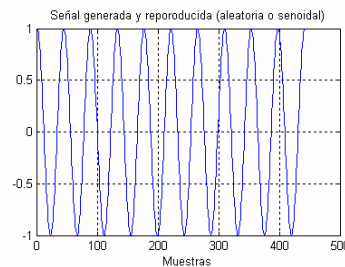


Figura 4.8-B. Señal generada por la PC que fue reproducida.

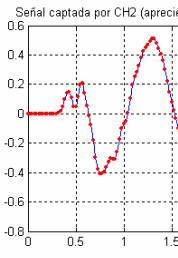


Figura 4.8-C. Acercamiento horizontal de la figura 4.8-A de la señal transmitida y adquirida por el mic2.

En esta gráficas se ve cómo es que la grabación ya se inició y la ráfaga emitida todavía no llega y lo hace hasta transcurridos $0.5ms$ después, eso significa que si se acercan los dispositivos un poco más, se logrará que la grabación comience justo cuando llegue la ráfaga y esto se puede apreciar en las siguientes gráficas.

Haciendo poco a poco el acercamiento, se logra el objetivo de la distancia, en este caso se trató de una distancia de $0.45m$ pero puede variar dependiendo del número de ventanas abiertas en la computadora. Para el micrófono que grabará la señal transmitida con este ajuste se tienen las figuras 4.9-A – 4.9-C.

Vista normal

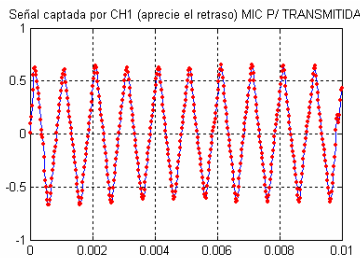


Figura 4.9-A. Señal transmitida y adquirida por el mic1 a $0.45m$ de distancia de la fuente sonora.

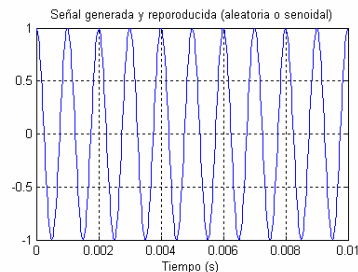


Figura 4.9-B. Señal generada por la PC que fue reproducida.

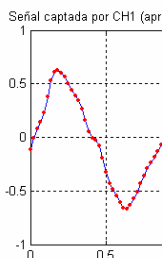


Figura 4.9-C. Acercamiento horizontal de la figura 4.9-A de la señal transmitida y adquirida por el mic1.

Para el micrófono que grabará la señal incidente con el ajuste de $0.45m$, se tienen las figuras 4.10-A – 4.10-C.

Vista normal

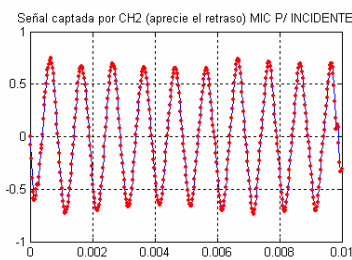


Figura 4.10-A. Señal incidente y adquirida por el mic2 a $0.45m$ de

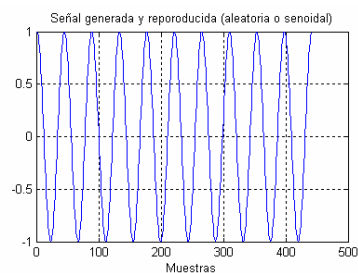


Figura 4.10-B. Señal generada por la PC que fue reproducida.

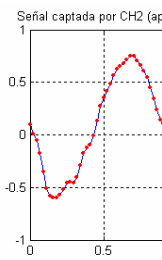


Figura 4.10-C. Acercamiento horizontal de la figura 4.10-A de

distancia de la fuente sonora.

la señal transmitida y adquirida
por el *mic2*.

Este mismo proceso sirve para ver que la señal capturada no esté saturada y al mismo tiempo se despliega en pantalla el *RMS* de cada canal y la relación en *dB* que guardan entre sí. El ajuste de nivel adecuado se da cuando se tienen cerca de $0\pm 0.1\text{dB}$ este ajuste se hace con la perilla de descalibración que permite compensar los niveles de voltaje entregados a la salida del amplificador de medición.

El proceso de acercamiento de elementos se debe hacer gradualmente hasta alcanzar la distancia adecuada, misma que servirá para calcular las posiciones óptimas de los elementos, cuando se termine con esta actividad, ya no se debe mover ningún control fino, de lo contrario, se deberá comenzar desde el principio.

Una vez hecho lo anterior, se deben colocar los dispositivos de acuerdo con los cálculos y volver a correr el programa para ver que la ráfaga incidente y transmitida se detecten sin saturación y con nivel suficientemente alto. Para el caso de la señal transmitida, es muy probable que necesite aumentar el voltaje de salida, ya que el nivel será menor, para incrementarlo se debe girar el control de sensibilidad de entrada a la izquierda por pasos de 10dB hasta encontrar la posición que mayor nivel entregue sin saturar, por lo tanto habrá una diferencia de sensibilidades, entre amplificadores de medición en *dB*, esa cantidad anótela y tómela en cuenta para el proceso de medición.

El proceso de ajuste ya está listo por lo que se puede comenzar las mediciones.

4.5 Procedimiento de medición.

Con el programa *ApreparativoCepstrum.m*, se deben ingresar los valores comunes de ajuste como son las distancias de elementos, temperatura ambiente, tamaño de la ráfaga, frecuencia de muestreo, número de ráfagas a emitir, la diferencia en *dB* de los dos amplificadores de medición, etc. Posteriormente se debe correr el programa para que se carguen las variables y sean utilizadas por el siguiente programa.

Para capturar las ráfagas de *n* ensayos, se debe ejecutar el programa *ACapturaCepstrum.m* y pulsar enter cada vez que se pida, se recomienda por experiencia que cada vez que la computadora esté en estado de espera para emitir la ráfaga se debe esperar unos 5 segundos para darle oportunidad a la PC que se estabilice. El usuario puede ver inmediatamente el aspecto que tienen las ráfagas, si alguna de ellas no se ve de manera similar a las que se vieron en el proceso de ajuste, es necesario que vuelva a empezar la captura esperando un poco más de tiempo entre ensayos.

Finalmente, se debe correr el programa *AProcesamientoCepstrum.m*- con el que se podrá ver, paso a paso, el procesamiento de las señales pulsando enter cuando se solicite. Al terminar el programa, se verán desplegadas las tablas y gráficas del aislamiento sonoro final.

4.6. Resultados.

Dentro de los resultados que arroja este procedimiento de medición, se muestra la figura 4.11 y su tabla de datos correspondiente 4.1 obtenidos para el panel B.

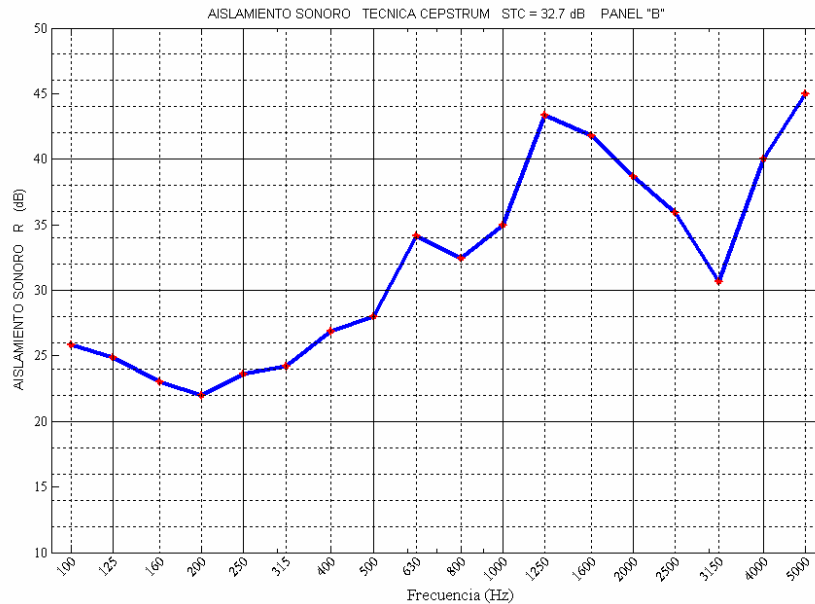


Figura 4.11. Curva de aislamiento sonoro medida con la técnica *Cepstrum* del panel B con *STC* correspondiente.

```

STC Banda 100-5KHz= 32.7 dB.
*****
La caída del mic2 a la muestra es de -6.2 dB
*****

```

| Frecuencia | RMSprom1 | RMSprom2 | RMSprom2caido | Aislamiento dB (20log(RMSprom2caido/RMSprom1)) |
|------------|----------|----------|---------------|--|
| 100.0 | 0.0 | 0.0 | 0.0 | 17.4 |
| 125.0 | 0.0 | 0.0 | 0.0 | 18.7 |
| 160.0 | 0.0 | 0.0 | 0.0 | 20.2 |
| 200.0 | 0.0 | 0.0 | 0.0 | 20.7 |
| 250.0 | 0.0 | 0.0 | 0.0 | 21.2 |
| 315.0 | 0.0 | 0.0 | 0.0 | 23.8 |
| 400.0 | 0.0 | 0.0 | 0.0 | 23.9 |
| 500.0 | 0.0 | 0.0 | 0.0 | 27.5 |
| 630.0 | 0.0 | 0.0 | 0.0 | 29.4 |
| 800.0 | 0.1 | 0.1 | 0.0 | 27.4 |
| 1000.0 | 0.1 | 0.1 | 0.0 | 27.1 |
| 1250.0 | 0.0 | 0.1 | 0.1 | 22.7 |
| 1600.0 | 0.1 | 0.1 | 0.1 | 19.5 |
| 2000.0 | 0.1 | 0.1 | 0.1 | 18.0 |
| 2500.0 | 0.1 | 0.1 | 0.1 | 18.9 |
| 3150.0 | 0.2 | 0.1 | 0.1 | 19.5 |
| 4000.0 | 0.2 | 0.4 | 0.2 | 19.9 |
| 5000.0 | 0.1 | 0.4 | 0.2 | 18.0 |

Tabla 4.1. Tabla de valores de aislamiento sonoro medido por bandas de frecuencia de tercios de octava del panel B con *STC* y caída de nivel del *mic2* a la muestra medida.

En el capítulo IV se mostrará la comparación de gráficas para saber el grado de confiabilidad de esta técnica.

V DESARROLLO EXPERIMENTAL DE MEDICIÓN CON LA TÉCNICA DE SECUENCIAS DE MÁXIMA LONGITUD MLS

5.1. Técnica MLS.

Esta técnica de medición *in situ*, tendrá ajustes iniciales muy similares a la técnica *Cepstrum*, con la diferencia de que se trabajará con las respuestas impulsivas incidente y transmitida ventaneadas, a las que se le calculará el área bajo la curva, obteniendo así, un indicativo de la energía que posee cada señal. La evaluación de nivel de las respuestas impulsivas se calcula con la fórmula:

$$n^2_{(t)} = N \int_t^{\infty} h^2(u) du \quad (5-1)$$

Donde:

h es la respuesta impulsiva.

N es un factor de ganancia de la función h .

Para el caso de una respuesta impulsiva de duración finita T_{MLS} , la ecuación se transforma en:

$$n^2_{(t)} = N \int_{t=0}^{T_{MLS}} h^2(u) du \quad (5-2)$$

5.2. Planteamiento del problema.

En la medición *MLS* el montaje de dispositivos es el mismo que se usa en la técnica *Cepstrum* y se muestra en la figura 5.1.

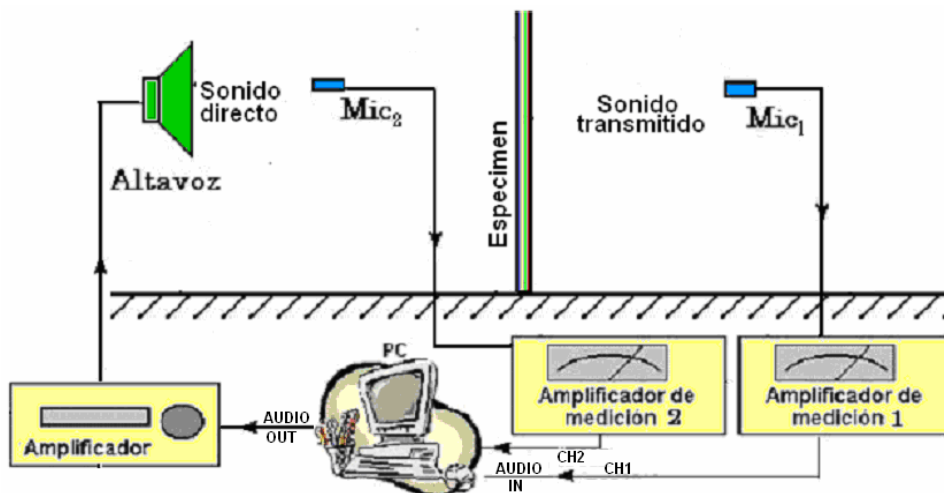


Figura 5.1. Diagrama de conexión para la medición de aislamiento sonoro con técnica *MLS*.

De la misma forma se requiere una optimización de distancias, que ahora servirá para tener una respuesta impulsiva con la mayor duración posible antes que se presente el primer reflejo.

Una vez establecido el montaje se tiene que evaluar el indicativo de energía de las señales incidentes y transmitidas para calcular el aislamiento.

5.3. Programación en la PC con Matlab 6.5.

Los programas desarrollados para este método fueron los siguientes:

calculohoptim.m y *calniv2mics.m* .- que son los mismos que en la técnica Cepstrum.

RastreoMLS2CH.m.- con este programa se puede visualizar las respuestas al impulso incidente y transmitida que se adquirieron simultáneamente en los dos canales y haciendo acercamientos se puede determinar en qué muestra comienza la ráfaga a manifestarse.

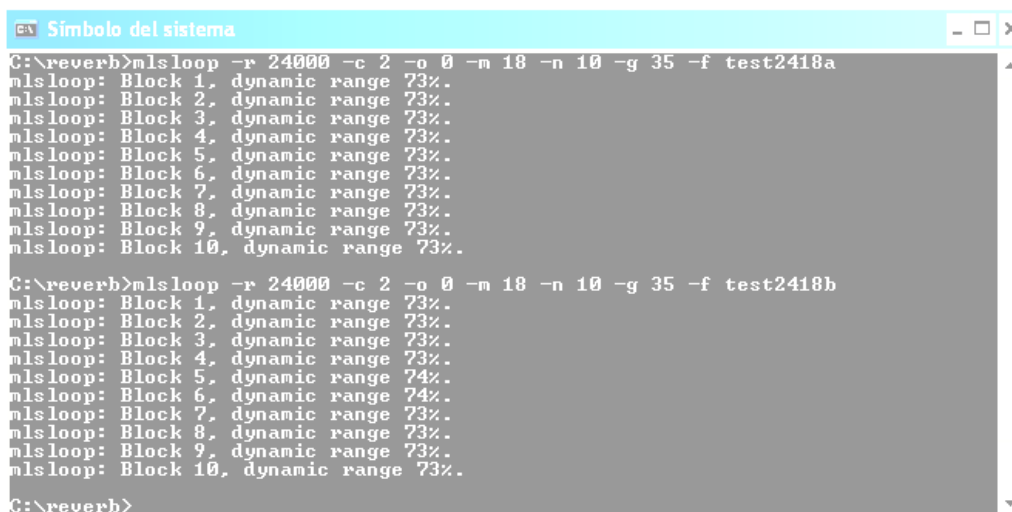
AislamientoMLSsincamara1.m.- en este programa se introducen los valores de las distancias del montaje, el número de muestra a partir de la cual se comenzará a tomar en cuenta la respuesta impulsiva incidente, los límites de graficación, etc. finalmente presenta los resultados con tablas y gráficas de aislamiento sonoro.

5.4. Ajuste de instrumental de medición.

El ajuste de los canales de adquisición, sigue el mismo procedimiento que con la técnica *Cepstrum* y de la misma manera se usa el criterio para la colocación de los dispositivos.¹

5.5. Procedimiento de medición.

Con los aparatos ya montados se procede a emitir las MLS, abriendo la ventana siguiente en el directorio en donde se encuentre trabajando, su procesamiento e indique que va a capturar *MLS* en dos canales simultáneamente como se ve en la siguiente ventana.



```
C:\reverh>mlsloop -r 24000 -c 2 -o 0 -n 18 -n 10 -g 35 -f test2418a
mlsloop: Block 1, dynamic range 73%.
mlsloop: Block 2, dynamic range 73%.
mlsloop: Block 3, dynamic range 73%.
mlsloop: Block 4, dynamic range 73%.
mlsloop: Block 5, dynamic range 73%.
mlsloop: Block 6, dynamic range 73%.
mlsloop: Block 7, dynamic range 73%.
mlsloop: Block 8, dynamic range 73%.
mlsloop: Block 9, dynamic range 73%.
mlsloop: Block 10, dynamic range 73%.

C:\reverh>mlsloop -r 24000 -c 2 -o 0 -n 18 -n 10 -g 35 -f test2418b
mlsloop: Block 1, dynamic range 73%.
mlsloop: Block 2, dynamic range 73%.
mlsloop: Block 3, dynamic range 73%.
mlsloop: Block 4, dynamic range 73%.
mlsloop: Block 5, dynamic range 74%.
mlsloop: Block 6, dynamic range 74%.
mlsloop: Block 7, dynamic range 73%.
mlsloop: Block 8, dynamic range 73%.
mlsloop: Block 9, dynamic range 73%.
mlsloop: Block 10, dynamic range 73%.

C:\reverh>
```

Figura 5.2. Ventana de *MS DOS* donde se ejecuta el comando *mlsloop* con los parámetros correspondientes para que la *PC* genere y emita las *MLS*.

¹ Consulte la sección correspondiente en el apartado de manejo de la técnica Cepstrum.

Se debe asegurar que la aguja de los dos amplificadores de medición no rebase el número 10 y también corrobore que el intervalo dinámico se encuentre entre 70 y 90%. También tome nota de cuántos decibeles de diferencia existen entre los dos indicadores de escala iluminados en rojo, para que los ingrese en el siguiente paso del procedimiento

Corra el programa *RastreoMLS2CH.m* ingresando el nombre del archivo que grabó y verifique en qué punto puede comenzar a considerarse la respuesta impulsiva incidente. Como ejemplo vea las figuras 5.3 y 5.4.

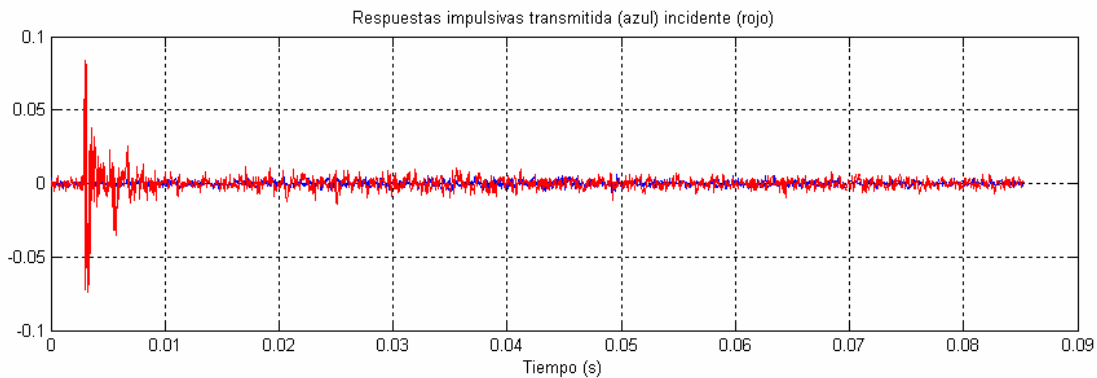


Figura 5.3. Respuestas impulsivas adquiridas por ambos canales con una duración de 0.085s.

Haciendo un acercamiento de la figura 5.3 se aprecia lo siguiente.

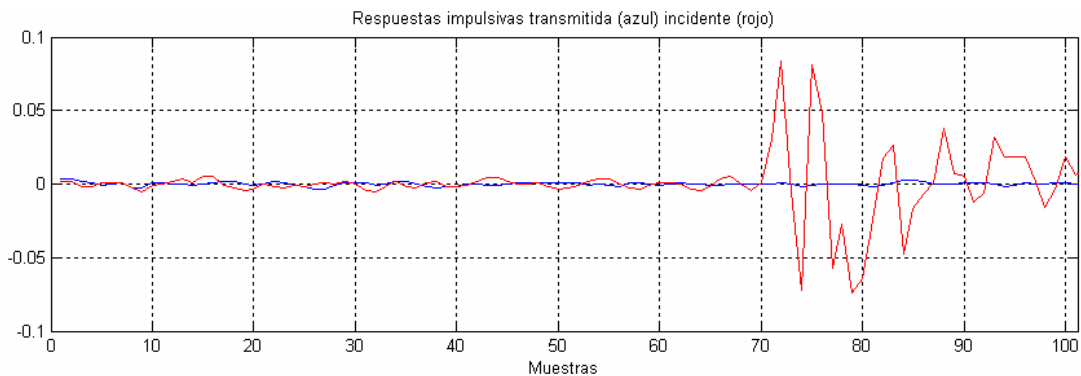


Figura 5.4. Acercamiento horizontal de las respuestas impulsivas de la figura 5.4 en la sección de mayor amplitud registrada.

Las figuras 5.3 y 5.4 corresponden al mismo ensayo, solo que en la 5.3 se presentan las respuestas impulsivas en función del tiempo en su totalidad y no se puede distinguir una de otra, y en la 5.4 se hace un acercamiento horizontal, en función de las muestras, lográndose ver ambas señales. Se aprecia que a partir de la muestra 70 comienza a manifestarse la ráfaga, lo que sucede antes simplemente es el retraso inherente al procesamiento de la computadora y no debe ser considerado. Considere el número 70 para procesar los datos como se ve en la siguiente ventana de programa.

```

C:\Odin 2006\Programas transmision\PANEL B\MLS SIN camara\AislamientoMLSsincamara1.m
File Edit View Text Debug Breakpoints Web Window Help
Stack: Base
1 Este programa calcula el aislamiento con MLS SIN camara de transmision
2 %utilizando el archivo test11.dat generado
3
4 clc
5 clear
6 close all
7 disp('ESTE PROGRAMA CALCULA EL AISLAMIENTO SONORO CON MLS SIN CAMARA DE TRANSMISION')
8 disp('SE UTILIZA EL ARCHIVO RMLS.dat CON LAS RESPUESTAS IMPULSIVAS CAPTURADAS')
9 disp('*****FILTRA LAS SEÑALES VENTANEADAS*****')
10 disp('*')
11 disp('Para comenzar a realizar la medicion pulse ENTER')
12 pause
13
14 load test11.dat
15 MuesQuitarInicio=70;%indicar a partir de cual muestra se va a considerar la respuesta impulsiva
16 MuesalMax=3;%contar el numero de muestras que hay sin cotar la primera, hasta el maximo
17 dBmas=15;%Incerte valores multiples de 10 que indiquen la diferencia de sensibilidad en los canales
18 CompensaciondB=0+dBmas;%Incerte valores multiples de 10 que indiquen la diferencia de sensibilidad en los canales
19
20 y1=test11(:,1);%Este vector contiene los datos del sonido transmitido
21 y2=test11(:,2);%Este vector contiene los datos del sonido incidente
22 y1=y1(MuesQuitarInicio:length(y1));
23 y2=y2(MuesQuitarInicio:length(y2));

```

Figura 5.5. Ventana de programa de *Matlab* 6.5 donde se carga el archivo *test11.dat* (secuencia *MLS*) y el valor de *70* para eliminar los datos generados por retraso de adquisición de la *PC*.

Ingrese también los valores del posicionamiento de elementos, corra el programa y visualice los resultados.

5.6. Resultados.

Los resultados que entrega el programa son una tabla de valores de aislamiento y una gráfica de aislamiento sonoro final. Para visualizar los resultados vea las siguientes figuras correspondientes a los resultados del panel *B*.

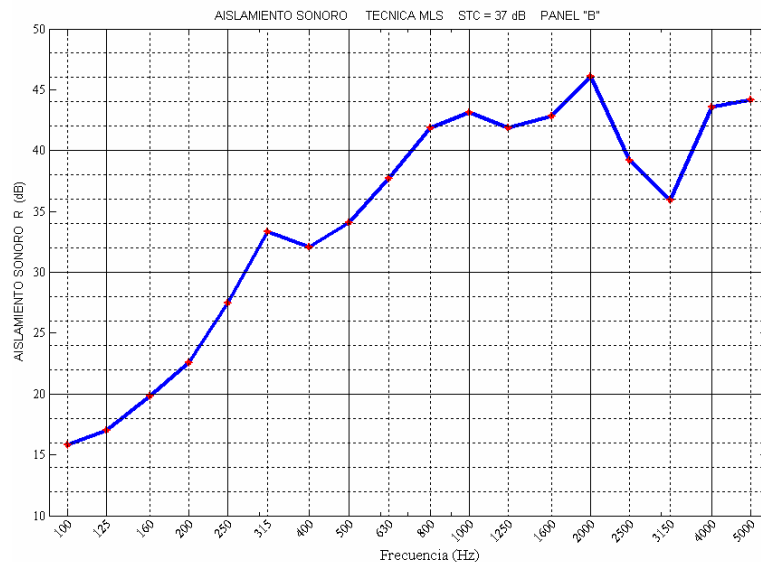


Figura 5.6. Curva de aislamiento sonoro medida con la técnica *MLS* del panel *B* con *STC* correspondiente.

STC Banda 100-5KHz= 37.0 dB.

 La caída del mic2 a la muestra es de -6.2 dB

| Frecuencia | Emic1 | Emic2 | Emic2caido | Aislamiento dB (10log(Emic1/Emic2caido)) |
|------------|-------|-------|------------|--|
| 100.0 | 0.0 | 0.0 | 0.0 | 15.8 |
| 125.0 | 0.0 | 0.0 | 0.0 | 17.0 |
| 160.0 | 0.0 | 0.0 | 0.0 | 19.8 |
| 200.0 | 0.0 | 0.0 | 0.0 | 22.6 |
| 250.0 | 0.0 | 0.0 | 0.0 | 27.5 |
| 315.0 | 0.0 | 0.0 | 0.0 | 33.3 |
| 400.0 | 0.0 | 0.0 | 0.0 | 32.1 |
| 500.0 | 0.0 | 0.0 | 0.0 | 34.0 |
| 630.0 | 0.0 | 0.0 | 0.0 | 37.7 |
| 800.0 | 0.0 | 0.0 | 0.0 | 41.8 |
| 1000.0 | 0.0 | 0.0 | 0.0 | 43.1 |
| 1250.0 | 0.0 | 0.0 | 0.0 | 41.9 |
| 1600.0 | 0.0 | 0.0 | 0.0 | 42.9 |
| 2000.0 | 0.0 | 0.0 | 0.0 | 46.1 |
| 2500.0 | 0.0 | 0.0 | 0.0 | 39.2 |
| 3150.0 | 0.0 | 0.0 | 0.0 | 36.0 |
| 4000.0 | 0.0 | 0.0 | 0.0 | 43.6 |
| 5000.0 | 0.0 | 0.0 | 0.0 | 44.2 |

Tabla 5.1. Tabla de valores de aislamiento sonoro medido por bandas de frecuencia de tercios de octava del panel B con STC y caída de nivel del mic2 a la muestra medida.

En el capítulo IV se mostrará la comparación de gráficas para saber el grado de confiabilidad de esta técnica.

VI COMPARACIÓN DE TÉCNICAS DE MEDICIÓN DE AISLAMIENTO SONORO

Esta es sin duda una de las partes interesantes de la tesis ya que se muestran los resultados finales de todos los procedimientos y permiten establecer un criterio para usar estos métodos.

Desde luego para llegar a este paso se debió hacer un exhaustivo número de ajustes para que la computadora desplegara resultados lo más cercanos posibles a los procedimientos estándar para que fueran útiles, de esta manera es que se lograron los objetivos del presente trabajo.

6.1. Gráficas comparativas.

Se midieron de manera completa un total de cuatro muestras de paneles dobles de tablarroca de 13mm de espesor y una separación de 6cm resultando las siguientes curvas de comparativas.

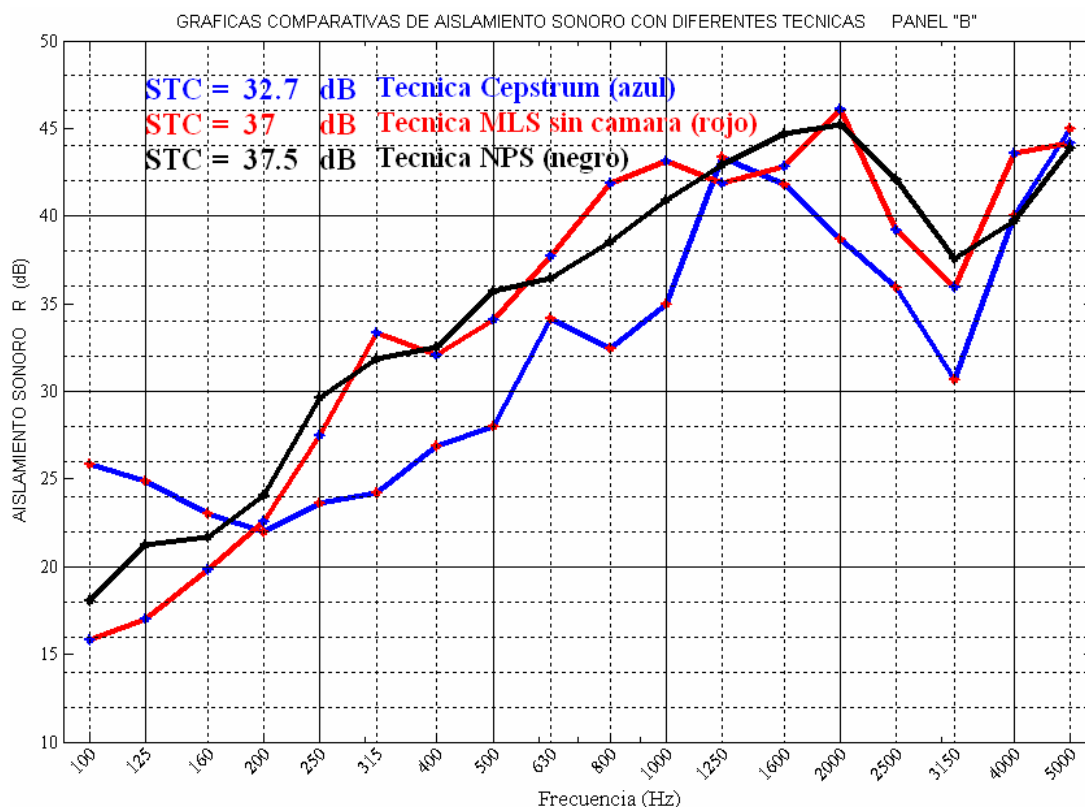


Figura 6.1. Curvas de aislamiento sonoro obtenido con diferentes técnicas Panel B.

De aquí se ve que la curva que se debe llegar a obtener es la de NPS ya que es la que se obtuvo bajo los lineamientos de la norma ISO 140-3 [1] y la que más se parece es la de técnica MLS.

En todas las gráficas se nota un decremento del aislamiento sonoro en la banda de 3.15KHz que se debe a la onda estacionaria que se genera entre los paneles que se encuentran separados por media longitud de onda. Eso hace que resuene y por lo tanto se pierde aislamiento al ser favorecida esa banda de frecuencias por las dimensiones que tiene el panel.

Otra curva comparativa es la que se tiene del método de intensidad¹ contra el método de presión sonora y es la siguiente para el panel C:

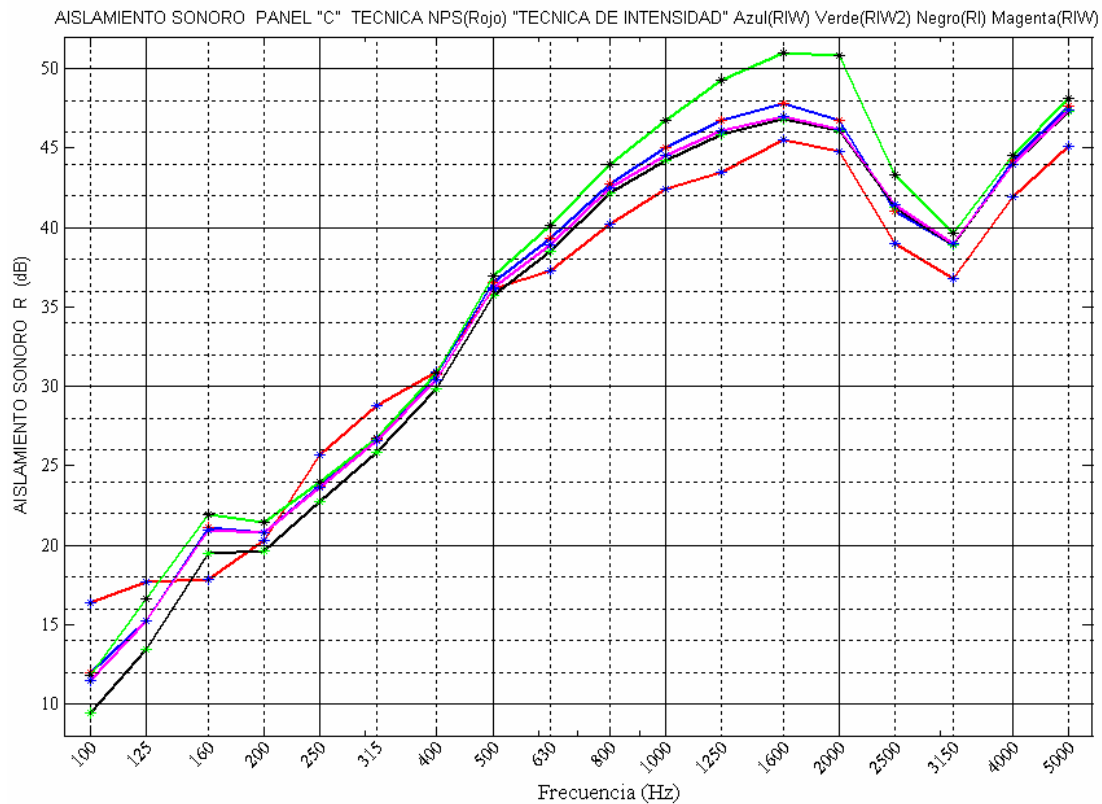


Figura 6.2. Comparación de gráficas de aislamiento sonoro con la técnica NPS y la técnica de intensidad proporcionada por el CENAM.

| STC | AISLAMIENTO (dB) | | | | |
|-----------------|------------------|-----------|------------|------------|--------------|
| | 36.2 | 35.6 | 35.7 | 34.4 | 35.5 |
| Frecuencia (Hz) | ROJO (NPS) | AZUL (NI) | VERDE (NI) | NEGRO (NI) | MAGENTA (NI) |
| 100 | 16.3 | 11.9 | 11.8 | 9.4 | 11.5 |
| 125 | 17.6 | 15.2 | 16.6 | 13.4 | 15.2 |
| 160 | 17.8 | 21.1 | 21.9 | 19.5 | 20.9 |
| 200 | 20.2 | 20.7 | 21.4 | 19.6 | 20.8 |
| 250 | 25.6 | 23.7 | 24.0 | 22.7 | 23.6 |
| 315 | 28.7 | 26.6 | 26.7 | 25.8 | 26.6 |
| 400 | 30.8 | 30.7 | 30.8 | 29.8 | 30.4 |
| 500 | 36.1 | 36.5 | 36.9 | 35.7 | 36.2 |
| 630 | 37.2 | 39.3 | 40.1 | 38.5 | 38.9 |
| 800 | 40.2 | 42.7 | 44.0 | 42.2 | 42.5 |
| 1000 | 42.4 | 44.9 | 46.7 | 44.2 | 44.5 |
| 1250 | 43.5 | 46.7 | 49.3 | 45.8 | 46.1 |
| 1600 | 45.5 | 47.8 | 51.0 | 46.8 | 47.0 |
| 2000 | 44.7 | 46.7 | 50.8 | 46.1 | 46.2 |
| 2500 | 39.0 | 41.0 | 43.3 | 41.3 | 41.4 |
| 3150 | 36.7 | 38.9 | 39.6 | 38.9 | 39.0 |
| 4000 | 41.9 | 44.2 | 44.5 | 44.0 | 44.0 |
| 5000 | 45.1 | 47.6 | 48.1 | 47.3 | 47.4 |

Tabla 6.1. Valores niveles de aislamiento sonoro con la técnica de NPS y los ensayos con la técnica de intensidad de la figura 6.2.

La curva roja de técnica NPS es muy parecida a las curvas de intensidad y de hecho debe tener mayor nivel de aislamiento en la técnica de intensidad.

¹ Los datos de aislamiento del método de intensidad fueron medidos y proporcionados por el CENAM para propósitos de comparación y verificación de validación del presente trabajo de Tesis.

El personal del *CENAM* comentó una situación interesante respecto a la curva de intensidad de color verde en relación a las otras que se obtuvieron mencionando que *“este resultado de aislamiento tan marcado es producto de eliminar las mediciones de flujo de energía sonora en aquellos puntos donde se presentaban fugas (en las orillas de la muestra) por lo que al final de cuentas se tiene mayor aislamiento en ciertas frecuencias. Las otras curvas de intensidad se parecen más entre si y con la curva de presión sonora, ya que se están considerando las fugas que se presentaron en la medición considerando todos los puntos de exploración”*.



Foto 6.1. Personal del *CENAM* haciendo ajustes del instrumental para las mediciones.

Estas son las gráficas comparativas de los diferentes paneles medidos.

PANEL A

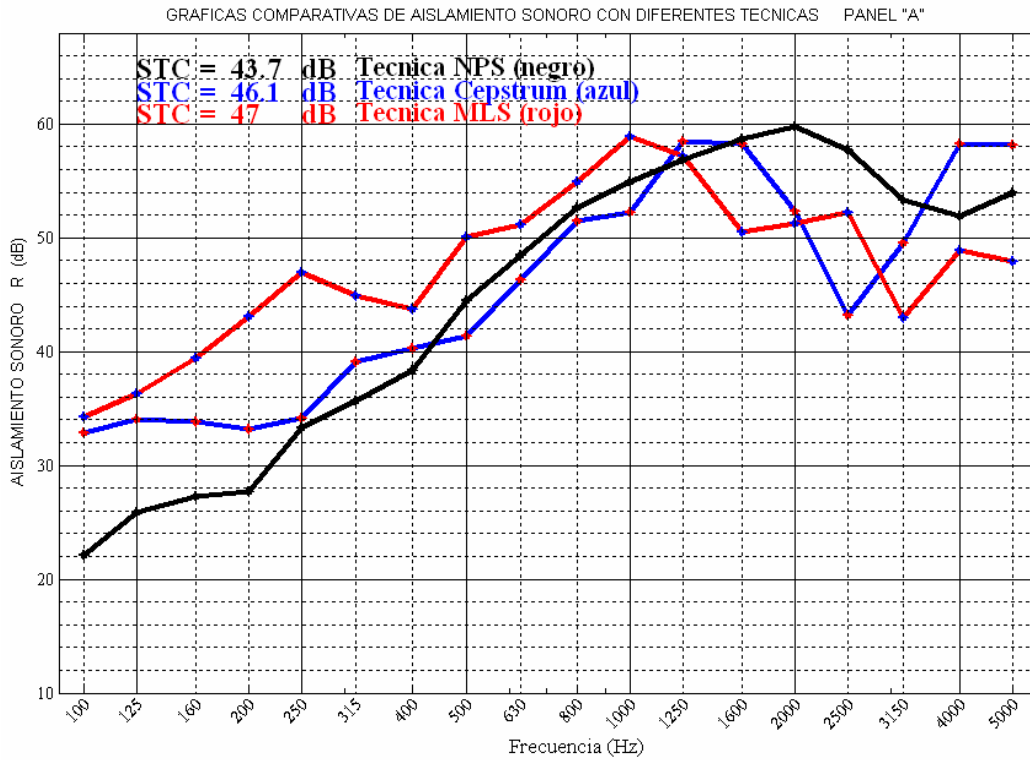


Figura 6.3-A. Comparación de gráficas de aislamiento sonoro obtenidas con las técnicas *NPS*, *Cepstrum* y *MLS* para el panel A. (Densidad superficial = 12.30Kg/m²)

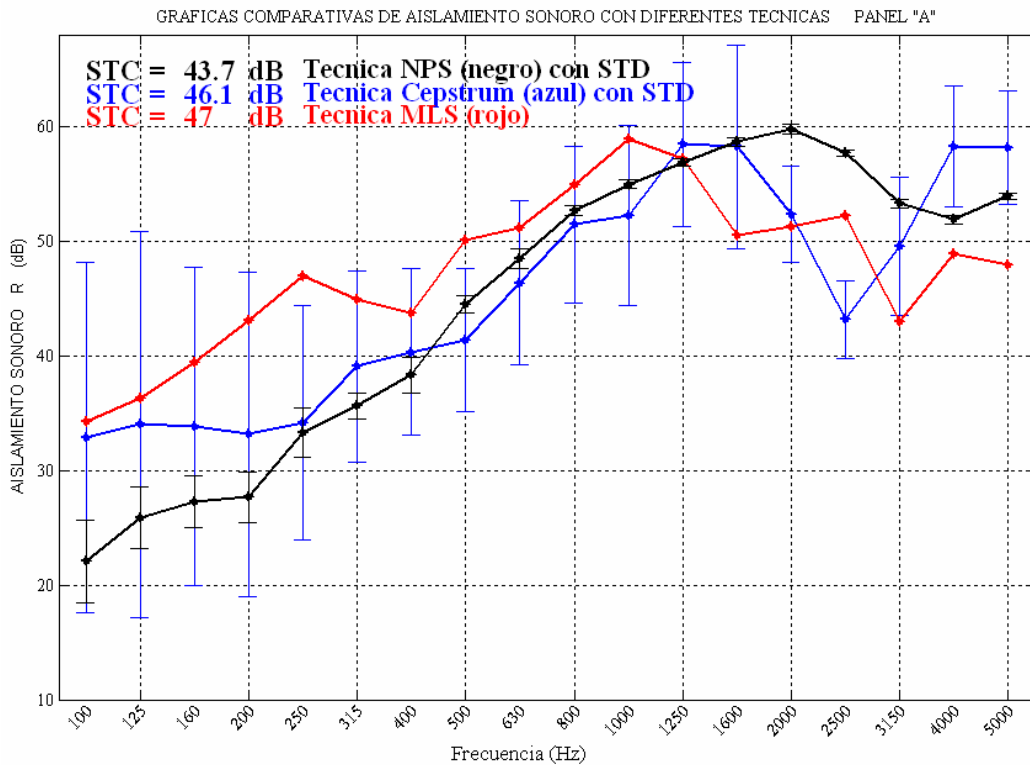


Figura 6.3-B. Comparación de gráficas de aislamiento sonoro obtenidas con las técnicas *NPS* y *Cepstrum* con desviación estándar y técnica *MLS* para el panel A.

PANEL B

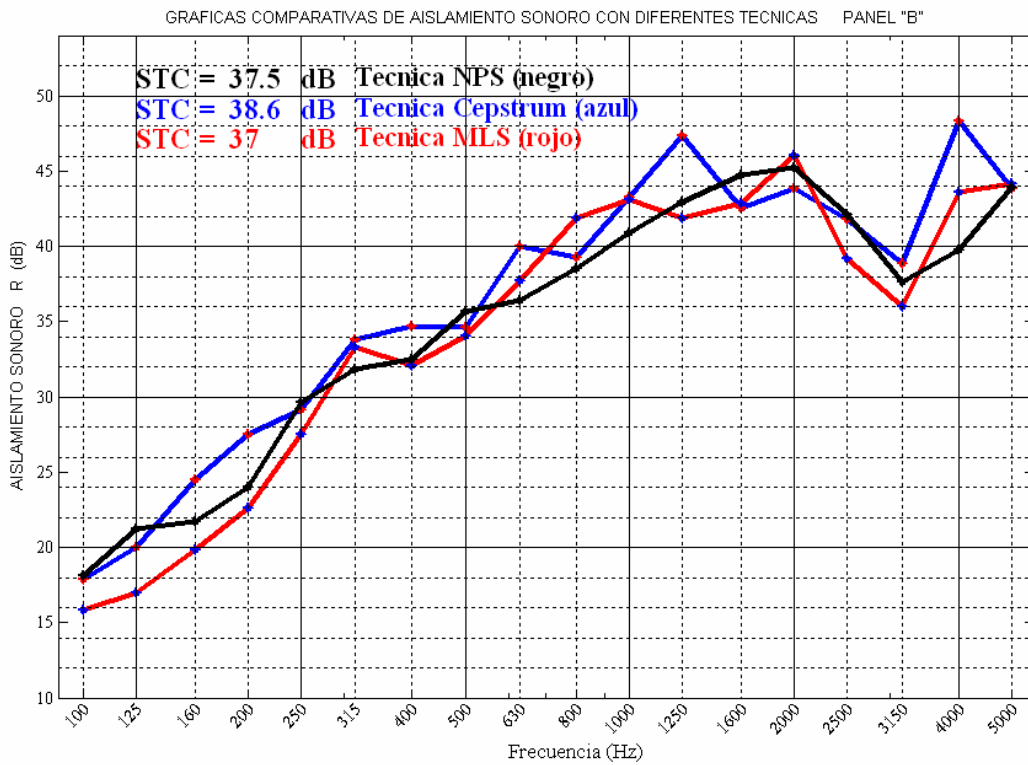


Figura 6.4. Comparación de gráficas de aislamiento sonoro obtenidas con las técnicas NPS, Cepstrum y MLS para el panel B. (Densidad superficial = 12.23Kg/m²)

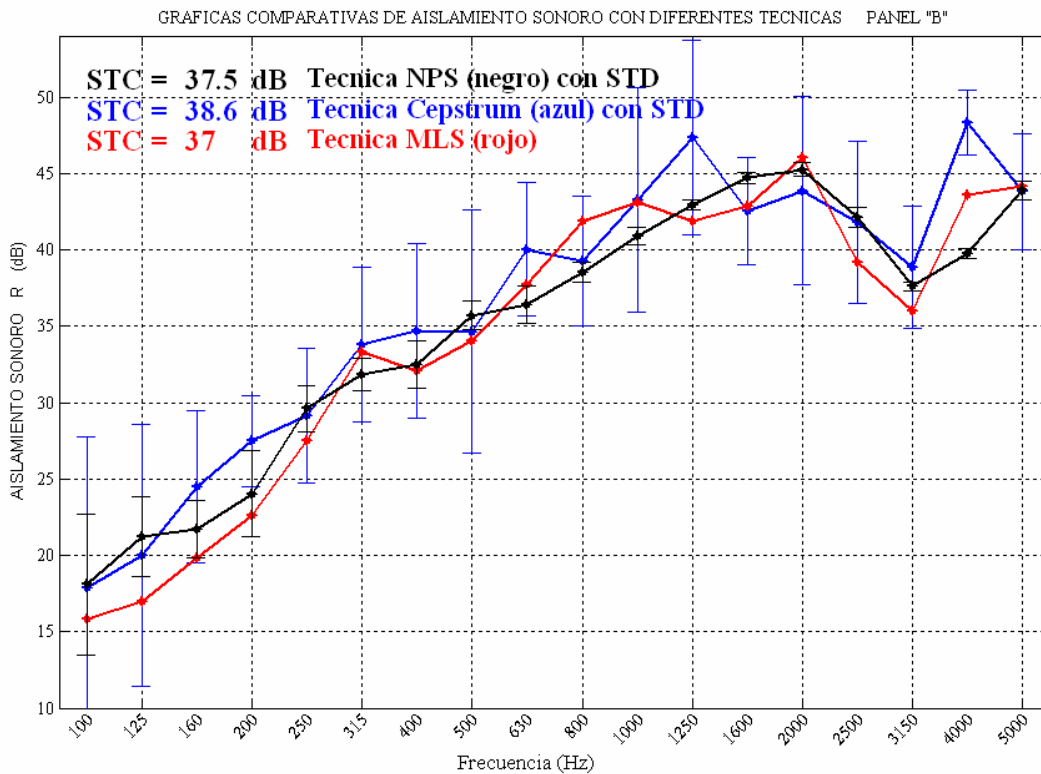


Figura 6.4-B. Comparación de gráficas de aislamiento sonoro obtenidas con las técnicas NPS y Cepstrum con desviación estándar y técnica MLS para el panel B.

PANEL C

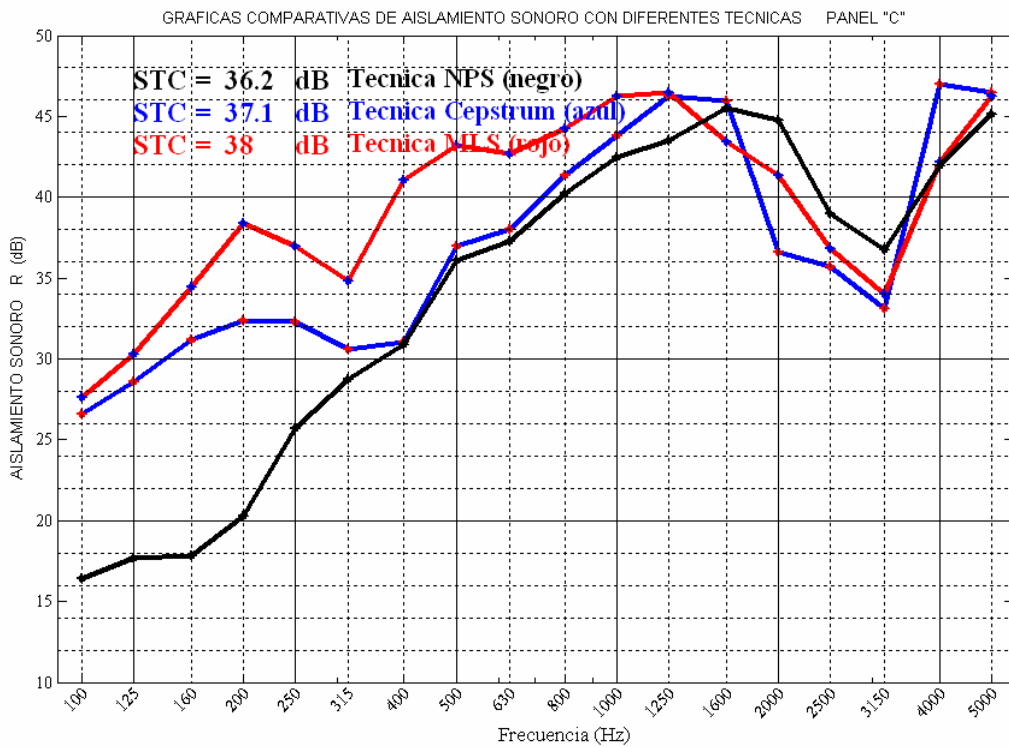


Figura 6.5. Comparación de gráficas de aislamiento sonoro obtenidas con las técnicas NPS, Cepstrum y MLS para el panel C. (Densidad superficial = 7.74Kg/m²)

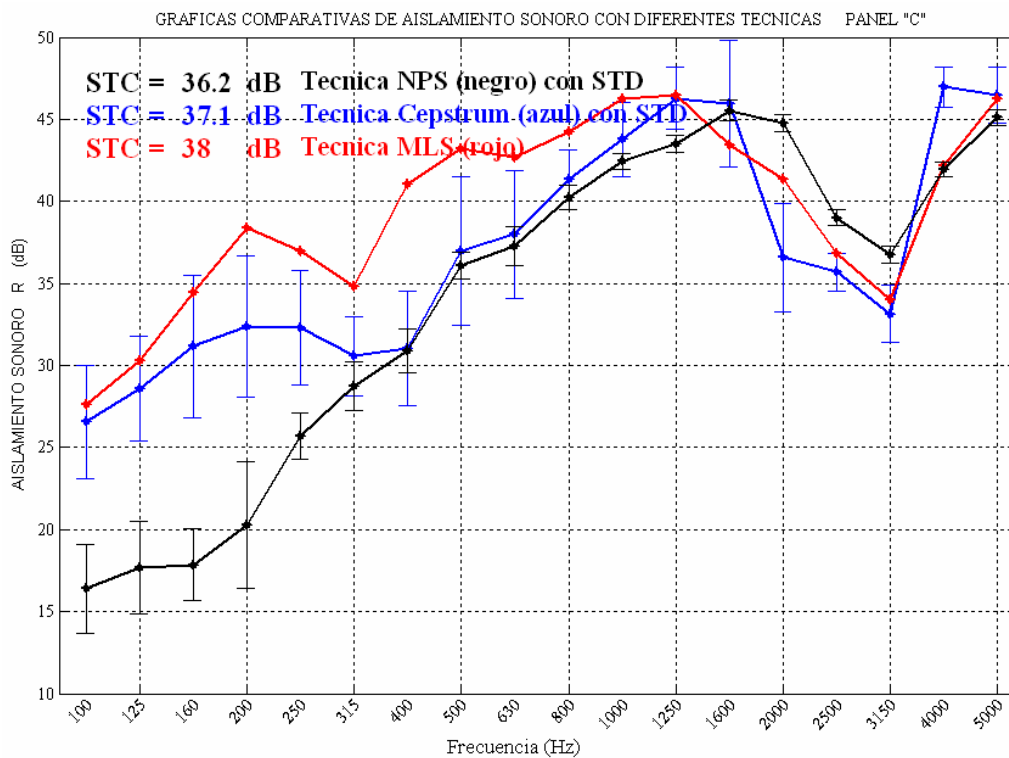


Figura 6.5-B. Comparación de gráficas de aislamiento sonoro obtenidas con las técnicas NPS y Cepstrum con desviación estándar y técnica MLS para el panel C.

PANEL D

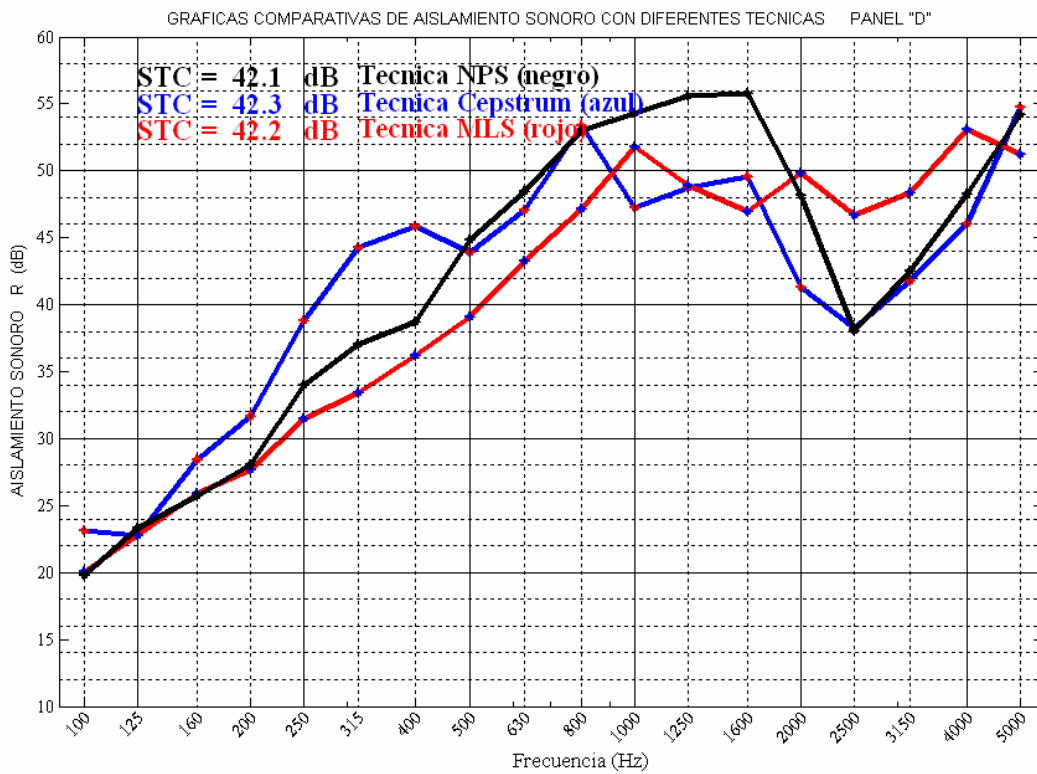


Figura 6.6. Comparación de gráficas de aislamiento sonoro obtenidas con las técnicas *NPS*, *Cepstrum* y *MLS* para el panel *D*. (Densidad superficial = 15.21Kg/m²)

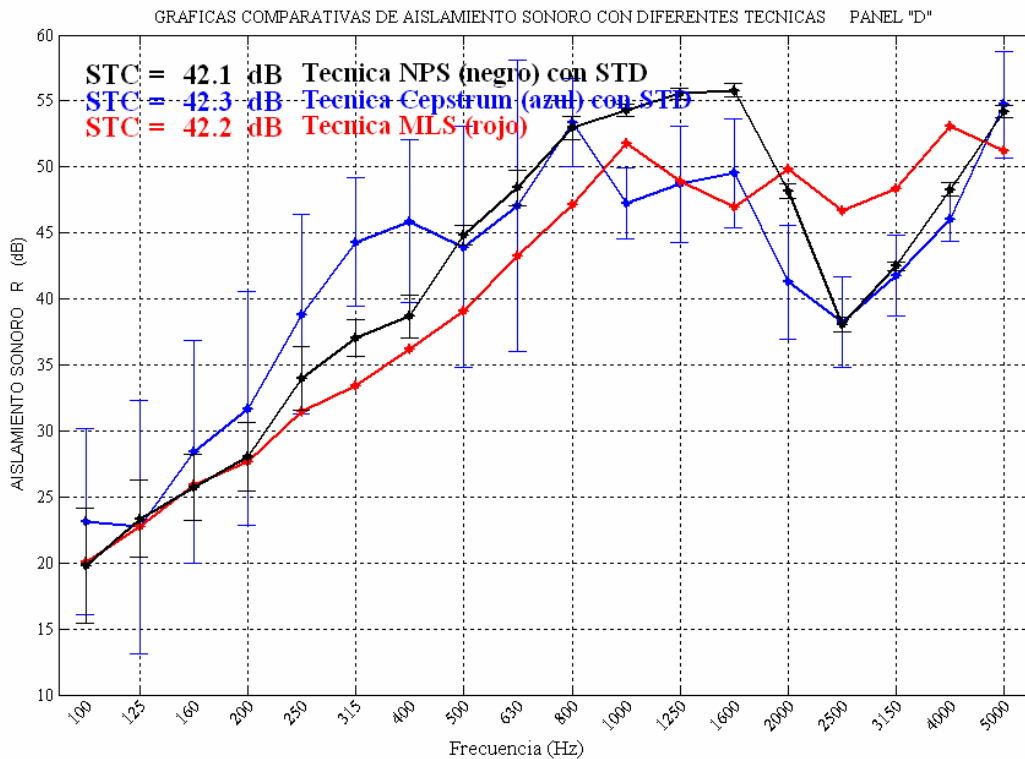


Figura 6.6-B. Comparación de gráficas de aislamiento sonoro obtenidas con las técnicas *NPS* y *Cepstrum* con desviación estándar y técnica *MLS* para el panel *D*.

RUIDO DE FONDO MEDIDO PARA LAS TÉCNICAS *IN SITU*

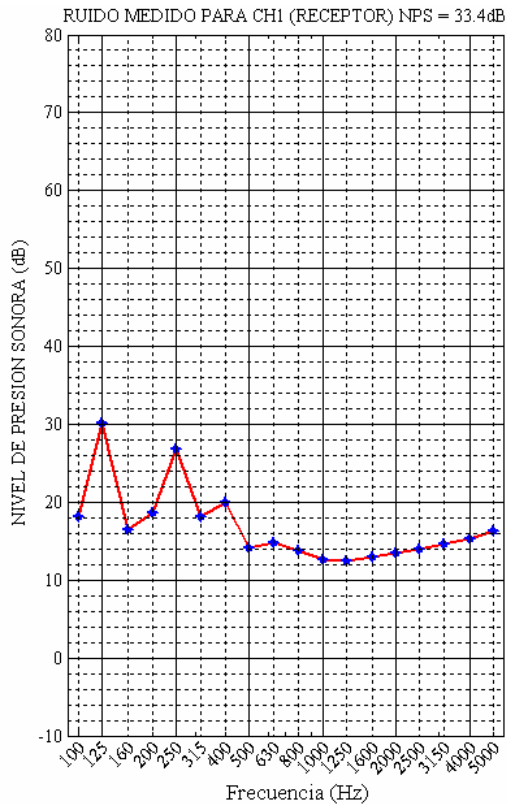


Figura 6.7. Ruido de fondo medido por el *mic1* ubicado del lado receptor de la muestra.

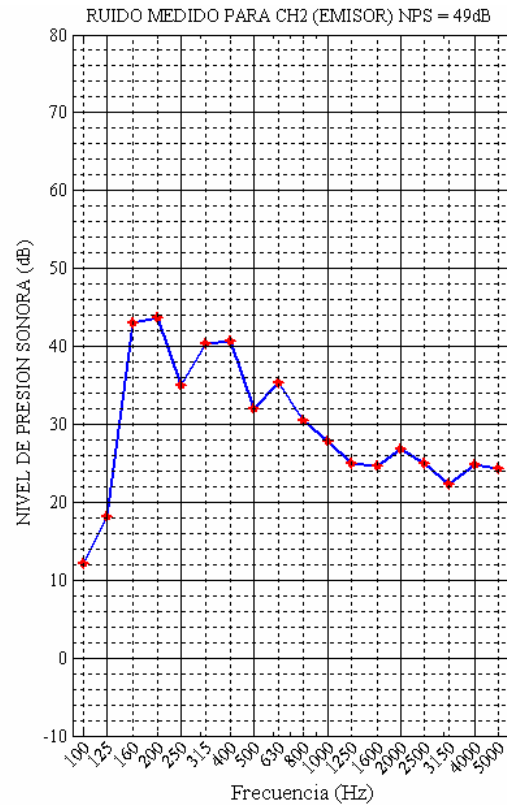


Figura 6.8. Ruido de fondo medido por el *mic2* ubicado del lado emisor de la muestra.

Para tener otra visualización se puede graficar simultáneamente los niveles de ruido en ambos lados y así se puede ver cuánto influye el hecho de que un micrófono reciba directamente el ruido de la fuente sonora y el otro que se encuentra del otro lado de la muestra.

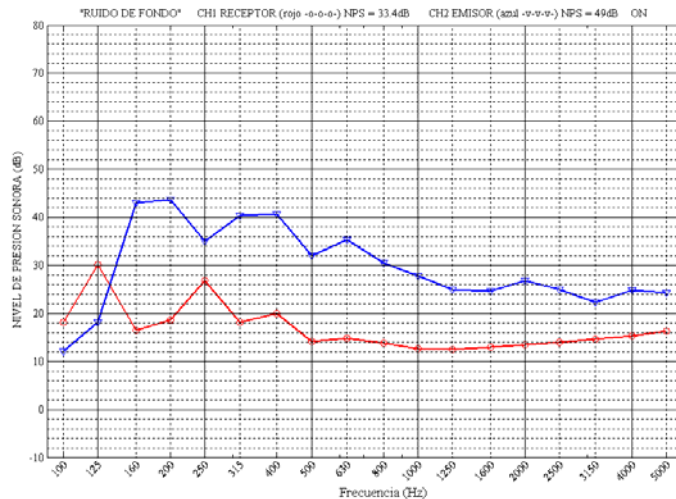


Figura 6.9. Curvas comparativas del ruido de fondo medido por el *mic1* (lado receptor) y *mic2* (lado emisor) para la medición *in situ*.

| TABLA DE RESULTADOS | | | |
|---|-------------|-------------|------------------|
| *****RUIDO MEDIDO DE CH1 Y CH2***** | | | |
| APARATOS ENCENDIDOS MEDICIONES <i>IN SITU</i> | | | |
| Frecuencia | NPS (dB)CH1 | NPS (dB)CH2 | Aislamiento (dB) |
| 100.0 | 18.2 | 12.1 | -6.1 |
| 125.0 | 30.2 | 18.2 | -12.0 |
| 160.0 | 16.5 | 43.1 | 26.6 |
| 200.0 | 18.7 | 43.8 | 25.1 |
| 250.0 | 26.8 | 35.0 | 8.2 |
| 315.0 | 18.2 | 40.3 | 22.2 |
| 400.0 | 20.0 | 40.7 | 20.7 |
| 500.0 | 14.2 | 32.1 | 17.9 |
| 630.0 | 14.8 | 35.3 | 20.5 |
| 800.0 | 13.8 | 30.5 | 16.7 |
| 1000.0 | 12.7 | 27.9 | 15.2 |
| 1250.0 | 12.5 | 25.0 | 12.5 |
| 1600.0 | 13.0 | 24.6 | 11.6 |
| 2000.0 | 13.5 | 26.9 | 13.3 |
| 2500.0 | 14.0 | 25.0 | 11.0 |
| 3150.0 | 14.6 | 22.3 | 7.7 |
| 4000.0 | 15.3 | 24.8 | 9.5 |
| 5000.0 | 16.2 | 24.4 | 8.2 |

Tabla 6.2. Niveles de ruido de fondo medidos de la figura 6.9 con aislamiento bajo condiciones *in situ*.

TIEMPOS DE REVERBERACIÓN DE LOS RECINTOS *IN SITU*

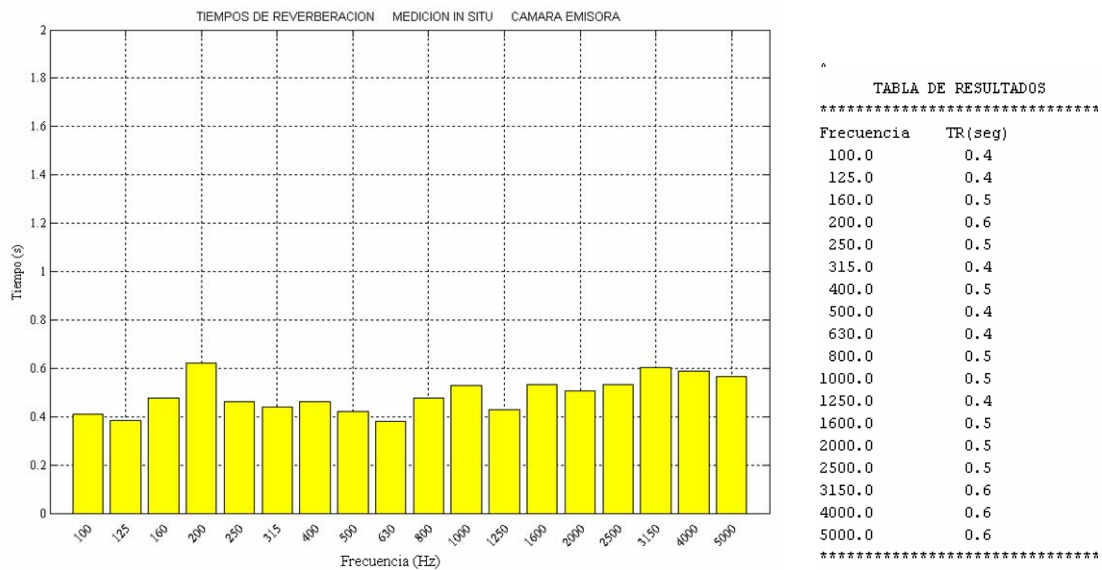


Figura 6.10. Tiempos de reverberación medidos en el recinto semireverberante para las mediciones *in situ* en la cámara emisora.

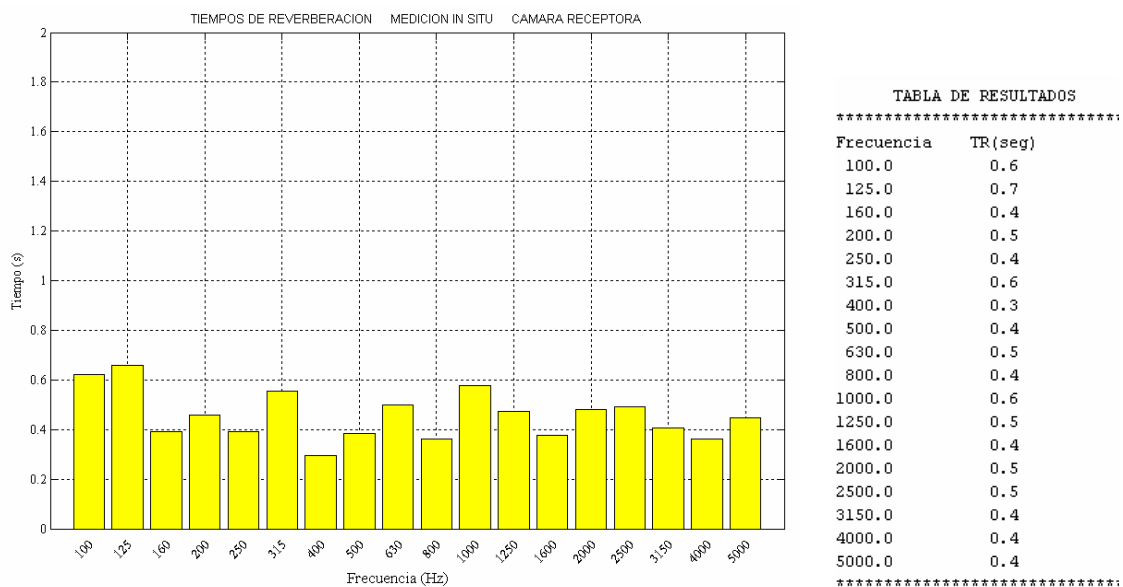


Figura 6.11. Tiempos de reverberación medidos en el recinto receptor para las mediciones *in situ* de sonido transmitido en la cámara receptora.

De los resultados que se tienen se observa que la desviación estándar en la técnica *Cepstrum* es muy grande en bajas frecuencias debido a que el tamaño de la ráfaga para ser evaluada, apenas llega a tener dos ciclos por lo que no puede considerarse confiable la medición cuando menos en frecuencias debajo de los 500Hz, a menos que se hagan un número de ensayos más grande, pero eso conlleva mayor tiempo en las mediciones.

Las técnicas *in situ* presentan resultados más aceptables en medias y altas frecuencias debido a que las señales a procesar contienen un número mayor de ciclos por lo que el error disminuye. En todos los casos se aprecia que se logra calcular un *STC* de manera eficiente, así que de manera confiable se puede estimar el aislamiento de manera general.

CONCLUSIONES

En este trabajo se han alcanzado muchos resultados importantes ya que se ha abarcado desde la calibración y los ajustes del instrumental de medición pasando por la elaboración de programas para la adecuación y procesamiento de señal hasta las tablas y gráficas finales de aislamiento.

La técnica de niveles de presión sonora *NPS* se ha establecido como la base para ver si las otras técnicas funcionan y mientras esté en la etapa de prueba es necesario hacer la comparación antes de utilizar una medición *in situ* definitiva.

Si se requiere una medición aproximada se puede hacer con los métodos alternativos pero no debe considerarse como definitiva la medición.

De acuerdo a las gráficas comparativas se aprecia que las técnica *MLS* tiende a dar resultados mas confiables en bajas frecuencias hasta los *500Hz* aproximadamente mientras que la técnica *Cepstrum* tiende a ser más confiable en altas frecuencias llegando siendo la mas baja de ellas en promedio de *2000Hz*.

Sin embargo hay una situación que hay que resaltar de los métodos alternativos: midiendo de esta forma hay una estimado muy aceptable en la búsqueda del *STC* lo cual permite siempre dar una idea de qué orden de aislamiento se está tratando. Si bien falta mucho para que estas técnicas sustituyan al método de *NPS*, se pueden realizar mediciones que no requieran de resultados de la precisión de un laboratorio certificado.

Dentro de los errores que se pueden cometer al realizar mediciones se encuentra que se tomen valores de voltaje elevados cercanos al nivel de saturación como referencia o bien que los controles de nivel de entrada de audio no estén al máximo lo cual limita la capacidad de entrada de señal.

La inercia del cono de la fuente sonora puede introducir un error debido a que en ciertas frecuencias no logra responder tan rápido para desplazarse a una amplitud estable.

Basados en los modelos planteados se puede realizar una interfaz gráfica que facilite en ingreso de datos, sin embargo el arduo trabajo de programación bastó para tener resultados preliminares.

Se puede tener altavoces de mejor calidad que respondan más rápido a las señales breves que se intentan reproducir, disminuyendo el efecto de la inercia.

Para llevar a cabo exitosamente la aplicación de estas técnicas se recomienda consultar detalles con el autor de la tesis ya que en caso de tener alguna dificultad se puede resolver rápidamente, desde luego es básico estudiar el presente documento, ya que en la medida que se entienda el razonamiento planteado será más fácil el desarrollo experimental.

Recomiendo que para utilizar este método exista una capacitación que haga más rápido el aprendizaje de las técnicas de medición y de esta forma se podrán hacer en un futuro mejoras aplicando nuevos enfoques distintos al del autor.

BIBLIOGRAFÍA

- [1] UNE- ISO 140-3:1995 Acoustics - Measurement of sound insulation in buildings and of building elements. Part 3: Laboratory measurements of airborne sound insulation of building elements.
- [2] C. J. Struck and S. F. Temme, "Simulated free field measurements," *J. Audio Eng. Soc.* 42, 467-482 (1994).
- [3] PÉREZ Ruiz Santiago Jesús, ALCÁNTARA Iniestra Salvador. Sistemas acústicos de alto desempeño. Benemérita Universidad Autónoma de Puebla. Textos Científicos, 2003,101-105.
- [4] JAMES Glyn. Matemáticas avanzadas para ingeniería. Ed. Prentice may. 2da. Ed. 2002, 388-391.
- [5] Cyril M. Harris. Manual de medidas acústicas y control de ruido. Ed. McGraw Hill
- [6] V. Garrido, Medición y cálculo de los índices acústicos en recintos. Tesis de Licenciatura. (Ing. Mecánica Eléctrica, Facultad de Ingeniería, UNAM, México 1999.
- [7] Manual del Analizador 2034 de B&K. *Cap 9. Cepstrum, Liftered Spectrum Sixth Guided Tour. Pp115-121.*
- [8] R. N. S. Hammad, A. A. Mustafa, M. K. Abdelazeez Technical Note. Sound Insulation of different types of Shutters Used in Jordan. *Applied Acoustics* 40 pp 373-381, Faculty of Engineering and Technology, University of Jordan, Amman, Jordan. 1994.
- [9] SMIRENOV Vladimir D. Technical Note. A quick method of estimating sound insulation. *Applied Acoustics* 40 pp 79-87, Laboratory of Building Acoustics, Moscow Research Institute of Standard and Experimental Design, 1992
- [10] E. R. Green, P. M. Tathavadekar, D. Poliquin, and A. L. Zent. Measurement system for acoustic absorption using the cepstrum technique. *Memorias. Inter.noise* 2002.
- [11] KINSLER Lawrence E. *Fundamentos de Acústica* p.p175-194. Ed Limusa.1999
- [12] Fahy, F., *Sound and Structural Vibration*. Academic Press INC. Gran Bretaña (1987).
- [13] Porges, G., *Applied Acoustics*. Edward Arnold Publishers LTD. Gran Bretaña (1977).
- [14] Cremer L., Heckl M., Ungar E., *Structure Borne Sound*, Springer-Verlag, Berlin, Heidelberg, New York (1973).
- [15] RUIZ Boullosa Ricardo PÉREZ Ruiz Santiago. Estudio sobre el laboratorio de acústica del Centro de Instrumentos. 46 pp. 1982.
- [16] Instruction Manual. Measuring Amplifier 2636 B&K. June 1985.
- [17] Instruction Manual. Measuring Amplifier 2610 B&K. June 1985.
- [18] Data Acquisition Toolbox. For use with MATLAB. User's Guide Versión 1 The Math Works Inc. 1998-1999.
- [19] B. G. Van Zyl, P.J. Erasmus and G. J. J. Van der Merwe. Determination of sound reduction indices in the presence of flanking transmission. *Applied acoustics*. Vol. 19 pp 25-39 1986.
- [20] Nota técnica. Sound Flanking in residential Dwellings.

- [21] R. W. Guy and P. Sauer. The influence of sills and reveals on sound transmission loss. *Applied acoustics* Vol. 17 pp 453-476 1984.
- [22] TADEU Antonio. Mateus Diogo. Sound insulation provided by single and double panel walls – a comparison of analytical solutions versus experimental results. *Applied acoustics* Vol. 65 pp 15-29 2004.
- [23] BRAVO José María. SINISTERRA Jaime. Influence of air layers and damping layers between gypsum boards on sound transmission. *Applied acoustics* Vol. 63 pp 1051-1059 2002.
- [24] CROCKER Malcolm J. P. K. Raju. Bjorn Forssen. Measurement of transmission loss of panels by the direct determination of transmitted acoustic intensity. *Noise control Engineering*. pp 6-11 July-August 1981
- [25] URIS Antonio. CERVERA Francisco. Sound transmission loss of lightweight double walls filled with polyurethane foam. *Noise control Engineering*. Vol. 48 pp 5-7 Jan-Feb 2000.
- [26] COPS André. MINTEN Mark. Comparative study between the sound intensity method and the conventional two-room method to calculate the sound transmission loss of wall constructions. *Noise control Engineering Journal* pp 104-111 May-June 1984.
- [27] PAPADOPOULOS Christos I. Development of an optimised, standard-compliant procedure to calculate sound transmission loss: design of transmission rooms. *Applied acoustics* Vol. 63 pp 1003-1029. 2002.
- [28] Standard test method for measurement of airborne sound insulation in buildings. ASTM designation: E 336 – 90 pp 727-736.
- [29] ISO 11957:1996(E) Acoustics- Determination of sound insulation performance of cabins – Laboratory and in situ measurements.
- [30] WARNOCK A. C. C. FASOLD Wolfgang. Sound insulation: Airborne and impact. *Encyclopedia of Acoustics*. pp 1129-1160.
- [31] BJOR Ole-Herman. Evaluation of decay curves for determination of reverberation time and non-linearity. *Acta Acustica United with Acustica* Vol. 90. pp 788-789. 2004.
- [32] Odín Moreno Amezcua; S. J. Pérez-Ruíz; A.E. Pérez Matzumoto Measurement of Sound Insulation using CEPSTRUM. *Congreso Internacional en Innovación y Desarrollo Tecnológico. Memorias* 2006.

ANEXOS

Programas realizados en Matlab 6.5 que pueden ser consultados en el CD.

TÉCNICA NPS

RMS94dBcal2CHs.m.
RevBdeF.m.
RevBdeFtodo.m.
RevBdeFCamaraA.
RdeF2CHyRdBOFF.m
RdeF2CHyRdBON.m.
RdeFProcesamientoON.m
RdeFProcesamientoOFF.
AislamientoPreliminar.m.
AislamientoPreparativo.m.
AislamientoPA.m, AislamientoPB.m, ...'AislamientoPF.m, 'AislamientoPA2.m,
... AislamientoPF2.m.
CalculoPABCDEF2.m.

TÉCNICA CEPSTRUM

calculohoptim.m.
calniv2mics.m.
ApreparativoCepstrum.m.
ACapturaCepstrum.m.
AProcesamientoCepstrum.m.

TÉCNICA MLS

calculohoptim.m y calniv2mics.m .- los mismos que en la técnica Cepstrum.
RastreoMLS2CH.m.
AislamientoMLSSincamara1.m.

TÉCNICA NPS

RMS94dBcal2CHs.m.

```
%Este programa mide el RMS en dos canales simultáneamente para obtener
los
%Vref 1 y 2 de la escala seleccionada y muestra la grafica de la señal
%capturada
clear
clc %Borrar pantalla
close all
d = 10;%Tiempo en segundos de la grabacion
fs=22050;%Frecuencia de muestreo
tmuestra=1/fs;%Tiempo por muestra
t=0:tmuestra:d;%Vector de tiempo con el numero de muestras exacto para cubrir
el tiempo de grabacion
tx=t(1:length(t)-1);
disp('Este programa mide el RMS en dos canales simultaneamente para
obtener los')
disp('Vref 1 y 2 de la escala seleccionada y muestra la grafica de la señal
capturada')
disp('Para comenzar a grabar, presione ENTER')
pause
disp('Grabando señal de adquisicion en dos canales, ESPERE ....')
%INICIA SECCION DE GRABACION CON WINSOUND
ai = analoginput('winsound');% Se prepara el elemento ai
CH=addchannel(ai,[1,2]);%Se adiciona un canal para grabar
set(ai,'TriggerChannel',CH(1),'TriggerChannel',CH(2));
set(ai,'SampleRate',fs);%Se ajusta el elemento ai a una frecuencia de muestreo
TasaReal=get(ai,'SampleRate'); %Se le indica que se debe conseguir
informacion con una f de muestreo.
set(ai,'SamplesPerTrigger',d*TasaReal);%Se ajusta ai, el numero de muestras
para ser grabadas
set(ai,'TriggerType','Immediate');%Se indica que el inicio de grabacion seria
inmediato

%
start(ai) %Que comienze a operar el elemento ai.
while strcmp(ai.Running,'On');%Mientras corra 'ai', que este prendido
end %Que termine

data = getdata(ai); %Almacene en el vector data lo que haya capturado 'ai'
delete(ai); %Que borre lo que tiene almacenado 'ai' para prepararse de nuevo
%TERMINA SECCION DE GRABACION CON WINSOUND

%data1=[data1;0];

RMSdata1 = RMSScf(data(:,1));%Se calcula el RMS de la señal del CH1
RMSdata2 = RMSScf(data(:,2));%Se calcula el RMS de la señal del CH2
```

```

disp('El RMS (Vref1 CH1) para la escala seleccionada para 94dB es:')
disp(RMSdata1)
disp('El RMS (Vref2 CH2) para la escala seleccionada para 94dB es:')
disp(RMSdata2)
disp('*')
disp('Verifique que cada nivel RMS sea menor al nivel de saturacion de lo
contrario existe saturacion')
disp('Verifique las curvas de las señales capturadas no esten recortadas de lo
contrario existe saturacion')
disp('Generalmente el nivel pico de saturacion se encuentra entre 0.5 y 0.81
dependiendo de su tarjeta de sonido')

```

```

figure
subplot(2,1,1)
plot(data(:,1))
title(['Señal capturada por Mic 1 CH1 -TRANSMITIDA- RMS Vref1 = '
,num2str(RMSdata1)])
xlabel('Muestras')
grid

```

```

subplot(2,1,2)
plot(data(:,2))
title(['Señal capturada por Mic 2 CH2 -INCIDENTE- RMS Vref2 = '
,num2str(RMSdata2)])
xlabel('Muestras')
grid

```

```

pause

```

```

figure
subplot(2,1,1)
plot(tx,data(:,1))
title(['Señal capturada por Mic 1 CH1 -TRANSMITIDA- RMS Vref1 = '
,num2str(RMSdata1)])
xlabel('Tiempo (s)')
grid

```

```

subplot(2,1,2)
plot(tx,data(:,2))
title(['Señal capturada por Mic 2 CH2 -INCIDENTE- RMS Vref2 = '
,num2str(RMSdata2)])
xlabel('Tiempo (s)')
grid

```

RevBdeF.m.

```
%Este programa grafica la respuesta impulsiva de la reverberacion y muestra
%la curva de decaimiento para que posteriormente se ocupe el programa rt_xr
%y se mida el tiempo de reverberacion por babdas de frecuencia
close all
clear
clc

disp('ESTE PROGRAMA CALCULA EL TIEMPO DE REVERBERACION EN
UN SOLO LUGAR')
disp('CARGA EL ARCHIVO QUE INDIQUE *.DAT Y LO ANALIZA EN BANDAS
DE FRECUENCIA')
disp('*')
pause(1)
disp('ESPERE CARGANDO Y PROCESANDO DATOS ...')

load A12.dat %Introduzca el archivo a examinar <nombredearchivo.dat>
y=A12; %Vuelva a escribir el nombre <y=nombredearchivo>

Tmax=4;
ValorDeca=-35;%Limite inferior de graficacion para cada respuesta impulsiva
fs=24000;
h=20;% h= limite inferior seleccionado para la curva de decaimiento global en
dB
% 100 125 160 200 250 315 400 500 630 800 1000 1250 1600 2000 2500
3150 4000 5000
hs=[10 15 15 15 20 20 20 20 20 20 20 20 20 20 20 20 20 20
20];%Limites inferiores por banda de frecuencia

ts=1/fs;
t=0:ts:length(y)/fs;
tmen1=t(1:length(t)-1);%Vector tiempo con una muestra menos para graficar la
respuesta impulsiva
tmen2=t(1:length(t)-2);%Vector tiempo con dos muestras menos para poder
graficar mas adelante la curva de decaimiento ya que tiene una muestra menos
que la respuesta impulsiva y dos meno del vector tiempo generado ya que este
ultimo tiene una muestra mas por el cero que lleva al principio

x=cdecay(y,fs);%curva de decaimiento global
z=rt_xe(x,h,fs);%Tiempo de reverberacion global

BdeF=[100 125 160 200 250 315 400 500 630 800 1000 1250 1600 2000 2500
3150 4000 5000];

nt=length(BdeF);%Se van a tener 19 tercios de octava de 125Hz hasta 8KHz

for n = 1:nt %Se va a procesar la muestra de disparo q en tercios de octava
    f = BdeF(n);
```

```

    wn = [round((f)*2^(-1/6)) round((f)*2^(1/6))];%Se establecen las
frecuencias de corte
    [b,a] = butter(3,wn/fs*2);%filtro fabricado
    filtrada = filter(b,a,y);
    yfilt(:,n)=filtrada;%En la columna n, todos los datos (filas) se almacena la
señal filtrada del tercio n evaluado.

```

```

    %Calculando curva de decaimiento
    cd(n,:)=cdecay(filtrada,fs);%Tomo el vector filtrada,ya que es el vector
filtrado actual en el ciclo for
    rt(n)=rt_xe(cd(n,:),hs(n),fs);

```

```

end

```

```

%////////////////////////////////////
%////////SECCION PARA DARLE NOMBRE AL ARCHIVO////////////////////////////////
%////////////////////////////////CON TIEMPOS DE REVERBERACION////////////////////////////////
% rttest1=rt;%Asignacion a una nueva variable por bandas
% rtGtest1=z;%Tiempo de reverberacion global
% save 'rttest1' rttest1;%Salvando archivo con timpo de reverberacion por
bandas
% save 'rtGtest1' rtGtest1;%Salvando Tiempo de rverberacion global un valor
%////////////////////////////////////
%////////////////////////////////////

```

```

TablaRT=[BdeF' rt'];

```

```

clc

```

```

TRtexto=sprintf('Tiempo de reverberacion global = %3.1f segundos.',z);
disp(TRtexto)

```

```

%////////////////////////////////////
disp('*')
disp('  TABLA DE RESULTADOS')
disp('*****')
disp('Frecuencia  TR(seg)')
TF=num2str(TablaRT,'%13.1f');
disp(TF)
disp('*****')
disp('*')
%////////////////////////////////////

```

```

figure
%subplot(2,1,1)
plot(tmen1,y)
title('Respuesta impusiva (global) en funcion del tiempo')
xlabel('Tiempo (s)')
grid

```

```

pause

```

```

%subplot(2,1,2)
figure
plot(tmen2,x)
grid
title('Curva de decaimiento Global en funcion del tiempo')
xlabel('Tiempo (s)')
ylabel('Decaimiento (dB)')
ylim([ValorDeca 0])

%*/**/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*
%*/**/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*
%*/**/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*
disp('Para ver la grafica de tiempos de reverberacion pulse ENTER')
pause
figure
bar(rt)
title('TIEMPOS DE REVERBERACION')
Grarevbar(Tmax)

disp('Para ver las curvas de repuestas impulsivas y de decaimiento por banda
de frecuencia pulse ENTER')
pause

for m=1:nt
disp('Para ver la siguiente curva pulse ENTER')
pause

figure
subplot(2,1,1)
plot(tmen1,yfilt(:,m))
grid
title(['Respuesta impusiva en funcion del tiempo (' ,num2str(BdeF(m)), ' Hz)'])
xlabel('Tiempo (s)')

subplot(2,1,2)
plot(tmen2,cd(m,:))
grid
title(['Curva de decaimiento en funcion del tiempo (' ,num2str(BdeF(m)), ' Hz)'])
xlabel('Tiempo (s)')
ylabel('Decaimiento (dB)')
ylim([ValorDeca 0])

end

disp('TERMINADO DE PRESENTACION DE LAS GRAFICAS')

```

RevBdeFtodo.m.

%Este programa obtiene el tiempo de reverberacion apartir de 12 archivos
%de datos de respuestas impulsivas que se deben indicar al principio y en
%la variable "ys". Saca los promedios t grafica los resultados finales. El
%vector que tiene los tiempos de rev. es "rtprom". finalmente se ven las
%curvas de decaimiento por frecuencia para seleccionar los puntos en el
%vector hs y volver a correr el programa.

```
close all
clear
clc
Tmax=4;
dBmin=-30;
dBmax=0;
disp('ESTE PROGRAMA CALCULA EL TIEMPO DE REVERBERACION PARA
LA CAMARA RECEPTORA')
disp('CARGA LOS ARCHIVOS B1.dat ... B12.dat')
disp('PULSE ENTER PARA COMENZAR LOS CALCULOS ...')
pause
disp('ESPERE CARGANDO Y PROCESANDO DATOS ...')
```

```
load B1.dat
load B2.dat
load B3.dat
load B4.dat
load B5.dat
load B6.dat
load B7.dat
load B8.dat
load B9.dat
load B10.dat
load B11.dat
load B12.dat
```

```
ys=[B1 B2 B3 B4 B5 B6 B7 B8 B9 B10 B11 B12];
```

```
y=B1;%Iguacion con una respuesta impulsiva para tomarlo como referencia
para medir su tamaño
```

```
fs=24000;
```

```
h=15;% h= limite inferior en dB
```

```
% 100 125 160 200 250 315 400 500 630 800 1000 1250 1600 2000 2500
3150 4000 5000
```

```
%hs=[12 17 20 20 25 25 25 25 25 25 20 20 15 15 15 15 20
20];%Limites inferiores por banda de frecuencia
```

```
%hs=[10 15 15 15 20 20 20 20 20 20 20 20 20 20 20 20 20
20];%Limites inferiores por banda de frecuencia
```

```
hs=[16 14 14 20 20 20 20 20 20 20 20 25 25 20 20 20 20
20];%Limites inferiores por banda de frecuencia
```

```

ts=1/fs;
t=0:ts:length(y)/fs;
tmen1=t(1:length(t)-1);%Vector tiempo con una muestra menos para graficar la
respuesta impulsiva
tmen2=t(1:length(t)-2);%Vector tiempo con dos muestras menos para poder
graficar mas adelante la curva de decaimiento ya que tiene una muestra menos
que la respuesta impulsiva y dos menos del vector tiempo generado ya que este
ultimo tiene una muestra mas por el cero que lleva al principio

x=cdecay(y,fs);%curva de decaimiento global
z=rt_xe(x,h,fs);%Tiempo de reverberacion global

BdeF=[100 125 160 200 250 315 400 500 630 800 1000 1250 1600 2000 2500
3150 4000 5000];
nt=length(BdeF);%Se van a tener 19 tercios de octava de 125Hz hasta 8KHz

k=size(ys,2);

for j = 1:k%Se va a procesar desde la columna j hasta 12 que es el numero de
archivos

    for n = 1:nt %Se va a procesar la muestra de disparo q en tercios de octava
        f = BdeF(n);
        wn = [round((f)*2^(-1/6)) round((f)*2^(1/6))];%Se establecen las
frecuencias de corte
        [b,a] = butter(3,wn/fs*2);%filtro fabricado
        filtrada = filter(b,a,ys(:,j));
        yfilt(:,n)=filtrada;%En la columna n, todos los datos (filas) se almacena la
señal filtrada del tercio n evaluado.

        %Calculando curva de decaimiento
        cd(n,:)=cdecay(filtrada,fs);%Tomo el vector filtrada,ya que es el vector
filtrado actual en el ciclo for
        rt(n)=rt_xe(cd(n,:),hs(n),fs);%Se calcula el tiempo de reverberacion para
esta columna

    end
    cd1(j,:)=cd(1,:);%Curvas de decaimiento 1
    cd2(j,:)=cd(2,:);%Curvas de decaimiento 2
    cd3(j,:)=cd(3,:);%Curvas de decaimiento 3
    cd4(j,:)=cd(4,:);%Curvas de decaimiento 4
    cd5(j,:)=cd(5,:);%Curvas de decaimiento 5
    cd6(j,:)=cd(6,:);%Curvas de decaimiento 6
    cd7(j,:)=cd(7,:);%Curvas de decaimiento 7
    cd8(j,:)=cd(8,:);%Curvas de decaimiento 8
    cd9(j,:)=cd(9,:);%Curvas de decaimiento 9
    cd10(j,:)=cd(10,:);%Curvas de decaimiento 10
    cd11(j,:)=cd(11,:);%Curvas de decaimiento 11

```

```

cd12(j,:)=cd(12,:);%Curvas de decaimiento 12
cd13(j,:)=cd(13,:);%Curvas de decaimiento 13
cd14(j,:)=cd(14,:);%Curvas de decaimiento 14
cd15(j,:)=cd(15,:);%Curvas de decaimiento 15
cd16(j,:)=cd(16,:);%Curvas de decaimiento 16
cd17(j,:)=cd(17,:);%Curvas de decaimiento 17
cd18(j,:)=cd(18,:);%Curvas de decaimiento 18

rtall(:,j)=rt';%Se crea un vector con 12 columnas con los tiempos de rev de
cada ensayo

end

unos=ones(k,1);
NoBarra=1:nt;
rtprom=rtall*unos/k;
save 'rtprom' rtprom;
STDrt=std(rtall);%Crea un vector fila con cada STD por frecuencia

TablaRT=[BdeF' rtall];%Creando las tablas para desplegar

TablaRTProm=[BdeF' rtprom STDrt'];%Creando las tablas para desplegar

disp('*')
disp(' "TABLA DE RESULTADOS POR ENSAYO (12 POSICIONES)')
disp('*****')
disp('Frecuencia      Trev(seg) ')
TFA=num2str(TablaRT,'%13.2f');
disp(TFA)
disp('*****')
disp('*')
disp('*')
disp('Para ver la tabla de tiempos de reverberacion con promedios pulse
ENTER')
pause

disp("TABLA DE RESULTADOS CON PROMEDIOS")
disp('*****')
disp('Frecuencia Trev(seg)  STD')
TFB=num2str(TablaRTProm,'%13.2f');
disp(TFB)
disp('*****')
disp('*')
disp('Para ver las graficas de los tiempos de reverberacion promedio pulse
ENTER')
pause

figure
bar(rtprom)

```



```
title('TIEMPOS DE REVERBERACION PROMEDIO CAMARA RECEPTORA
V=68.3 m')
Grarevbar(Tmax)
disp('Para ver las siguiente grafica pulse ENTER')
pause
```

```
figure
errorbar(NoBarra,rtprom,STDrt,'r')
hold on
plot(NoBarra,rtprom)
title('TIEMPOS DE REVERBERACION PROMEDIO CAMARA RECEPTORA
V=68.3 m')
Grarevbar(Tmax)
disp('Para ver la ultima grafica pulse ENTER')
pause
```

```
figure
bar(rtprom,'y')
hold on
errorbar2(NoBarra,rtprom,STDrt)
title('TIEMPOS DE REVERBERACION PROMEDIO CAMARA RECEPTORA
V=68.3 m')
Grarevbar(Tmax)
```

```
disp('PROGRAMA TERMINADO. SE GENERO EL ARCHIVO rtprom.mat QUE
CONTIENE LOS Trev PARA EL AISLAMIENTO')
disp('SI DESEA VER LAS CURVAS DE DECAIMIENTO PARA
SELECCUIONAR LOS PUNTOS DE DECAIMIENTO HS PUSE ENTER')
disp('DE LO CONTRARIO PULSE CTRL+C')
pause
```

```
figure,plot(tmen2,cd1'),grid
title('Curvas de decaimiento en funcion del tiempo (100 Hz)')
xlabel('Tiempo (s)')
ylabel('Decaimiento (dB)')
ylim([dBmin dBmax])
pause
```

```
figure,plot(tmen2,cd2'),grid
title('Curvas de decaimiento en funcion del tiempo (125 Hz)')
xlabel('Tiempo (s)')
ylabel('Decaimiento (dB)')
ylim([dBmin dBmax])
pause
```

```
figure,plot(tmen2,cd3'),grid
title('Curvas de decaimiento en funcion del tiempo (160 Hz)')
```

```
xlabel('Tiempo (s)')
ylabel('Decaimiento (dB)')
ylim([dBmin dBmax])
pause
```

```
figure,plot(tmen2,cd4'),grid
title('Curvas de decaimiento en funcion del tiempo (200 Hz)')
xlabel('Tiempo (s)')
ylabel('Decaimiento (dB)')
ylim([dBmin dBmax])
pause
```

```
figure,plot(tmen2,cd5'),grid
title('Curvas de decaimiento en funcion del tiempo (250 Hz)')
xlabel('Tiempo (s)')
ylabel('Decaimiento (dB)')
ylim([dBmin dBmax])
pause
```

```
figure,plot(tmen2,cd6'),grid
title('Curvas de decaimiento en funcion del tiempo (315 Hz)')
xlabel('Tiempo (s)')
ylabel('Decaimiento (dB)')
ylim([dBmin dBmax])
pause
```

```
figure,plot(tmen2,cd7'),grid
title('Curvas de decaimiento en funcion del tiempo (400 Hz)')
xlabel('Tiempo (s)')
ylabel('Decaimiento (dB)')
ylim([dBmin dBmax])
pause
```

```
figure,plot(tmen2,cd8'),grid
title('Curvas de decaimiento en funcion del tiempo (500 Hz)')
xlabel('Tiempo (s)')
ylabel('Decaimiento (dB)')
ylim([dBmin dBmax])
pause
```

```
figure,plot(tmen2,cd9'),grid
title('Curvas de decaimiento en funcion del tiempo (630 Hz)')
xlabel('Tiempo (s)')
ylabel('Decaimiento (dB)')
ylim([dBmin dBmax])
pause
```

```
figure,plot(tmen2,cd10'),grid
title('Curvas de decaimiento en funcion del tiempo (800 Hz)')
xlabel('Tiempo (s)')
```

```
ylabel('Decaimiento (dB)')
ylim([dBmin dBmax])
pause
```

```
figure,plot(tmen2,cd11'),grid
title('Curvas de decaimiento en funcion del tiempo (1000 Hz)')
xlabel('Tiempo (s)')
ylabel('Decaimiento (dB)')
ylim([dBmin dBmax])
pause
```

```
figure,plot(tmen2,cd12'),grid
title('Curvas de decaimiento en funcion del tiempo (1250 Hz)')
xlabel('Tiempo (s)')
ylabel('Decaimiento (dB)')
ylim([dBmin dBmax])
pause
```

```
figure,plot(tmen2,cd13'),grid
title('Curvas de decaimiento en funcion del tiempo (1600 Hz)')
xlabel('Tiempo (s)')
ylabel('Decaimiento (dB)')
ylim([dBmin dBmax])
pause
```

```
figure,plot(tmen2,cd14'),grid
title('Curvas de decaimiento en funcion del tiempo (2000 Hz)')
xlabel('Tiempo (s)')
ylabel('Decaimiento (dB)')
ylim([dBmin dBmax])
pause
```

```
figure,plot(tmen2,cd15'),grid
title('Curvas de decaimiento en funcion del tiempo (2500 Hz)')
xlabel('Tiempo (s)')
ylabel('Decaimiento (dB)')
ylim([dBmin dBmax])
pause
```

```
figure,plot(tmen2,cd16'),grid
title('Curvas de decaimiento en funcion del tiempo (3150 Hz)')
xlabel('Tiempo (s)')
ylabel('Decaimiento (dB)')
ylim([dBmin dBmax])
pause
```

```
figure,plot(tmen2,cd17'),grid
title('Curvas de decaimiento en funcion del tiempo (4000 Hz)')
xlabel('Tiempo (s)')
ylabel('Decaimiento (dB)')
```

```
ylim([dBmin dBmax])  
pause
```

```
figure,plot(tmen2,cd18'),grid  
title('Curvas de decaimiento en funcion del tiempo (5000 Hz)')  
xlabel('Tiempo (s)')  
ylim([dBmin dBmax])  
ylabel('Decaimiento (dB)')
```

```
disp('PROGRAMA TERMINADO')
```

RevBdeFtodoCamaraA.m

%Este programa obtiene el tiempo de reverberacion apartir de 12 archivos
%de datos de respuestas impulsivas que se deben indicar al principio y en
%la variable "ys". Saca los promedios t grafica los resultados finales. El
%vector que tiene los tiempos de rev. es "rtpromCamaraA". finalmente se ven
las
%curvas de decaimiento por frecuencia para seleccionar los puntos en el
%vector hs y volver a correr el programa.

```
close all
```

```
clear
```

```
clc
```

```
Tmax=4;
```

```
dBmin=-30;
```

```
dBmax=0;
```

```
disp('ESPERE CARGANDO Y PROCESANDO DATOS ...')
```

```
load A1.dat
```

```
load A2.dat
```

```
load A3.dat
```

```
load A4.dat
```

```
load A5.dat
```

```
load A6.dat
```

```
load A7.dat
```

```
load A8.dat
```

```
load A9.dat
```

```
load A10.dat
```

```
load A11.dat
```

```
load A12.dat
```

```
ys=[A1 A2 A3 A4 A5 A6 A7 A8 A9 A10 A11 A12];
```

```
y=A1;%Iguacion con una respuesta impulsiva para tomarlo como referencia  
para medir su tamaño
```

```
fs=24000;
```

```
h=10;% h= limite inferior en dB
```

```
% 100 125 160 200 250 315 400 500 630 800 1000 1250 1600 2000 2500  
3150 4000 5000
```

```
hs=[12 14 15 23 23 18 20 20 20 20 20 23 23 18 18 18 20  
20];%Limites inferiores por banda de frecuencia
```

```
ts=1/fs;
```

```
t=0:ts:length(y)/fs;
```

```
tmen1=t(1:length(t)-1);%Vector tiempo con una muestra menos para graficar la  
respuesta impulsiva
```

```
tmen2=t(1:length(t)-2);%Vector tiempo con dos muestras menos para poder  
graficar mas adelante la curva de decaimiento ya que tiene una muestra menos
```

que la respuesta impulsiva y dos meno del vector tiempo generado ya que este ultimo tiene una muestra mas por el cero que lleva al principio

```
x=cdecay(y,fs);%curva de decaimiento global  
z=rt_xe(x,h,fs);%Tiempo de reverberacion global
```

```
BdeF=[100 125 160 200 250 315 400 500 630 800 1000 1250 1600 2000 2500  
3150 4000 5000];  
nt=length(BdeF);%Se van a tener 19 tercios de octava de 125Hz hasta 8KHz
```

```
k=size(ys,2);
```

```
for j = 1:k%Se va a procesar desde la columna j hasta 12 que es el numero de  
archivos
```

```
    for n = 1:nt %Se va a procesar la muestra de disparo q en tercios de octava  
        f = BdeF(n);  
        wn = [round((f)*2^(-1/6)) round((f)*2^(1/6))];%Se establecen las  
frecuencias de corte  
        [b,a] = butter(3,wn/fs*2);%filtro fabricado  
        filtrada = filter(b,a,ys(:,j));  
        yfilt(:,n)=filtrada;%En la columna n, todos los datos (filas) se almacena la  
señal filtrada del tercio n evaluado.
```

```
        %Calculando curva de decaimiento  
        cd(n,:)=cdecay(filtrada,fs);%Tomo el vector filtrada,ya que es el vector  
filtrado actual en el ciclo for  
        rt(n)=rt_xe(cd(n,:),hs(n),fs);%Se calcula el tiempo de reverberacion para  
esta columna
```

```
    end  
    cd1(j,:)=cd(1,:);%Curvas de decaimiento 1  
    cd2(j,:)=cd(2,:);%Curvas de decaimiento 2  
    cd3(j,:)=cd(3,:);%Curvas de decaimiento 3  
    cd4(j,:)=cd(4,:);%Curvas de decaimiento 4  
    cd5(j,:)=cd(5,:);%Curvas de decaimiento 5  
    cd6(j,:)=cd(6,:);%Curvas de decaimiento 6  
    cd7(j,:)=cd(7,:);%Curvas de decaimiento 7  
    cd8(j,:)=cd(8,:);%Curvas de decaimiento 8  
    cd9(j,:)=cd(9,:);%Curvas de decaimiento 9  
    cd10(j,:)=cd(10,:);%Curvas de decaimiento 10  
    cd11(j,:)=cd(11,:);%Curvas de decaimiento 11  
    cd12(j,:)=cd(12,:);%Curvas de decaimiento 12  
    cd13(j,:)=cd(13,:);%Curvas de decaimiento 13  
    cd14(j,:)=cd(14,:);%Curvas de decaimiento 14  
    cd15(j,:)=cd(15,:);%Curvas de decaimiento 15  
    cd16(j,:)=cd(16,:);%Curvas de decaimiento 16  
    cd17(j,:)=cd(17,:);%Curvas de decaimiento 17
```

```

cd18(j,:)=cd(18,:);%Curvas de decaimiento 18

rtall(:,j)=rt';%Se crea un vector con 12 columnas con los tiempos de rev de
cada ensayo

end

unos=ones(k,1);
NoBarra=1:nt;
rtpromCamaraA=rtall*unos/k;
save 'rtpromCamaraA' rtpromCamaraA;
STDrt=std(rtall);%Crea un vector fila con cada STD por frecuencia

TablaRT=[BdeF' rtall];%Creando las tablas para desplegar

TablaRTprom=[BdeF' rtpromCamaraA STDrt'];%Creando las tablas para
desplegar

disp('*')
disp('  "TABLA DE RESULTADOS POR ENSAYO (12 POSICIONES)"')
disp('*****')
disp('Frecuencia      Trev(seg)  ')
TFA=num2str(TablaRT,'%13.2f');
disp(TFA)
disp('*****')
disp('*')
disp('*')
disp('Para ver la tabla de tiempos de reverberacion con promedios pulse
ENTER')
pause

disp(' "TABLA DE RESULTADOS CON PROMEDIOS"')
disp('*****')
disp('Frecuencia Trev(seg)  STD')
TFB=num2str(TablaRTprom,'%13.2f');
disp(TFB)
disp('*****')
disp('*')
disp('Para ver las graficas de los tiempos de reverberacion promedio pulse
ENTER')
pause

figure
bar(rtpromCamaraA)
title('TIEMPOS DE REVERBERACION PROMEDIO      CAMARA EMISORA
V=77.6 m')
Grarevbar(Tmax)
disp('Para ver las siguiente grafica pulse ENTER')
pause

```

```

figure
errorbar(NoBarra,rtpromCamaraA,STDrt,'r')
hold on
plot(NoBarra,rtpromCamaraA)
title('TIEMPOS DE REVERBERACION PROMEDIO          CAMARA EMISORA
V=77.6 m')
Grarevbar(Tmax)
disp('Para ver la ultima grafica pulse ENTER')
pause

```

```

figure
bar(rtpromCamaraA,'y')
hold on
errorbar2(NoBarra,rtpromCamaraA,STDrt)
title('TIEMPOS DE REVERBERACION PROMEDIO          CAMARA EMISORA
V=77.6 m')
Grarevbar(Tmax)

```

```

disp('PROGRAMA TERMINADO. SE GENERO EL ARCHIVO
rtpromCamaraA.mat QUE CONTIENE LOS Trev PARA EL AISLAMIENTO')
disp('SI DESEA VER LAS CURVAS DE DECAIMIENTO PARA
SELECCUIONAR LOS PUNTOS DE DECAIMIENTO HS PUSE ENTER')
disp('DE LO CONTRARIO PULSE CTRL+C')
pause

```

```

figure,plot(tmen2,cd1'),grid
title('Curvas de decaimiento en funcion del tiempo (100 Hz)')
xlabel('Tiempo (s)')
ylabel('Decaimiento (dB)')
ylim([dBmin dBmax])
pause

```

```

figure,plot(tmen2,cd2'),grid
title('Curvas de decaimiento en funcion del tiempo (125 Hz)')
xlabel('Tiempo (s)')
ylabel('Decaimiento (dB)')
ylim([dBmin dBmax])
pause

```

```

figure,plot(tmen2,cd3'),grid
title('Curvas de decaimiento en funcion del tiempo (160 Hz)')
xlabel('Tiempo (s)')
ylabel('Decaimiento (dB)')
ylim([dBmin dBmax])
pause

```

```

figure,plot(tmen2,cd4'),grid

```



```
title('Curvas de decaimiento en funcion del tiempo (200 Hz)')
xlabel('Tiempo (s)')
ylabel('Decaimiento (dB)')
ylim([dBmin dBmax])
pause
```

```
figure,plot(tmen2,cd5'),grid
title('Curvas de decaimiento en funcion del tiempo (250 Hz)')
xlabel('Tiempo (s)')
ylabel('Decaimiento (dB)')
ylim([dBmin dBmax])
pause
```

```
figure,plot(tmen2,cd6'),grid
title('Curvas de decaimiento en funcion del tiempo (315 Hz)')
xlabel('Tiempo (s)')
ylabel('Decaimiento (dB)')
ylim([dBmin dBmax])
pause
```

```
figure,plot(tmen2,cd7'),grid
title('Curvas de decaimiento en funcion del tiempo (400 Hz)')
xlabel('Tiempo (s)')
ylabel('Decaimiento (dB)')
ylim([dBmin dBmax])
pause
```

```
figure,plot(tmen2,cd8'),grid
title('Curvas de decaimiento en funcion del tiempo (500 Hz)')
xlabel('Tiempo (s)')
ylabel('Decaimiento (dB)')
ylim([dBmin dBmax])
pause
```

```
figure,plot(tmen2,cd9'),grid
title('Curvas de decaimiento en funcion del tiempo (630 Hz)')
xlabel('Tiempo (s)')
ylabel('Decaimiento (dB)')
ylim([dBmin dBmax])
pause
```

```
figure,plot(tmen2,cd10'),grid
title('Curvas de decaimiento en funcion del tiempo (800 Hz)')
xlabel('Tiempo (s)')
ylabel('Decaimiento (dB)')
ylim([dBmin dBmax])
pause
```

```
figure,plot(tmen2,cd11'),grid
title('Curvas de decaimiento en funcion del tiempo (1000 Hz)')
```

```
xlabel('Tiempo (s)')
ylabel('Decaimiento (dB)')
ylim([dBmin dBmax])
pause
```

```
figure,plot(tmen2,cd12'),grid
title('Curvas de decaimiento en funcion del tiempo (1250 Hz)')
xlabel('Tiempo (s)')
ylabel('Decaimiento (dB)')
ylim([dBmin dBmax])
pause
```

```
figure,plot(tmen2,cd13'),grid
title('Curvas de decaimiento en funcion del tiempo (1600 Hz)')
xlabel('Tiempo (s)')
ylabel('Decaimiento (dB)')
ylim([dBmin dBmax])
pause
```

```
figure,plot(tmen2,cd14'),grid
title('Curvas de decaimiento en funcion del tiempo (2000 Hz)')
xlabel('Tiempo (s)')
ylabel('Decaimiento (dB)')
ylim([dBmin dBmax])
pause
```

```
figure,plot(tmen2,cd15'),grid
title('Curvas de decaimiento en funcion del tiempo (2500 Hz)')
xlabel('Tiempo (s)')
ylabel('Decaimiento (dB)')
ylim([dBmin dBmax])
pause
```

```
figure,plot(tmen2,cd16'),grid
title('Curvas de decaimiento en funcion del tiempo (3150 Hz)')
xlabel('Tiempo (s)')
ylabel('Decaimiento (dB)')
ylim([dBmin dBmax])
pause
```

```
figure,plot(tmen2,cd17'),grid
title('Curvas de decaimiento en funcion del tiempo (4000 Hz)')
xlabel('Tiempo (s)')
ylabel('Decaimiento (dB)')
ylim([dBmin dBmax])
pause
```

```
figure,plot(tmen2,cd18'),grid
title('Curvas de decaimiento en funcion del tiempo (5000 Hz)')
xlabel('Tiempo (s)')
```

```
ylabel('Decaimiento (dB)')  
ylim([dBmin dBmax])  
  
disp('PROGRAMA TERMINADO')
```

RdeF2CHyRdBOFF.m

%PROGRAMA PARA MEDIR EL RUIDO DE FONDO USANDO DOS
CANALES

```
clear
clc
close all
%*****INDICAR dB DEL SONOMETRO AL CALIBRAR Y VrefRMS MEDIDO
PARA CH1 Y CH2*****
dBref1=23.8;%Nivel que indico el sonometro al realizar la calibracion en CH1
Vref1=0.255;%Voltaje medido para el NPS verificado con con el sonometro
CH1
dBref2=33.8;%Nivel que indico el sonometro al realizar la calibracion en CH2
Vref2=0.275;%Voltaje medido para el NPS verificado con con el sonometro en
CH2
fs=22050;%Frecuencia de muestreo
duration=10;%Duracion en segundos de los disparos
minutos=3;%Tiempo en minutos de la duracion de la prueba de adquisicion
Notriggers=minutos*60/duration;%Numero de disparos realizados en la prueba
ymin=20;
ymax=90;
NPSymin=-10;
NPSymax=80;
%*****
%*****

%INICIA SECCION DE GRABACION CON WINSOUND
disp('SE MEDIRA EL RUIDO EN DOS CANALES PUDIENDO CALCULAR EL
AISLAMIENTO QUE EXISTE ENTRE ELLOS')
disp('O BIEN PUEDE MEDIR EL AISLAMIENTO MAXIMO EN LA
CARACTERIZACION DE LA CAMARA EN UN SOLO PUNTO')
disp('*****CON LOS INSTRUMENTOS APAGADOS*****')
disp('*')
Comentario1=sprintf('El grabado de los datos tomara %2.0f minutos.',minutos);
disp(Comentario1)

Comentario2=sprintf('Estan programados %3d disparos.',Notriggers');
disp(Comentario2)
disp('*')

disp('Para comenzar a grabar, presione ENTER')
pause
disp('*')
disp('GRABANDO DATOS AL ARCHIVO RdeF2CHOFF.daq, ESPERE.....')
disp('*')
```

```

ai = analoginput('winsound'); %Las vías y los canales de entrada y salida
se establecen -
ch= addchannel(ai,1:2);

set(ai,'SampleRate',fs);

ActualRate=get(ai,'SampleRate');
set(ai,'SamplesPerTrigger',duration*ActualRate);%

set(ai,'TriggerRepeat', Notriggers);%Cantidad de disparos
set(ai,'LogFileName','RdeF2CHOFF.daq');%Se crea el aisla2ch.daq
set(ai,'LogToDiskMode','Overwrite');
set(ai,'LoggingMode','Disk&Memory');

start(ai); %Comienza la reproducción / grabación.
while (strcmp(ai.Running,'On'));
end
delete(ai); %Que borre lo que tiene almacenado 'ai' para prepararse de nuevo
%TERMINA SECCION DE GRABACION CON WINSOUND

disp('*****')
disp('ARCHIVO RdeF2CHOFF.daq GENERADO')
disp('GRABADO DE DATOS TERMINADO')
disp('AHORA EJECUTE EL PROGRAMA RdeFProcesamientoOFF.m')
disp('*****')

```

RdeF2CHyRdBON.m.

%PROGRAMA PARA MEDIR EL RUIDO DE FONDO USANDO DOS CANALES

```
clear
clc
close all
%*****INDICAR dB DEL SONOMETRO AL CALIBRAR Y VrefRMS MEDIDO
PARA CH1 Y CH2*****
dBref1=43.8;%Nivel que indico el sonometro al realizar la calibracion en CH1
Vref1=0.255;%Voltaje medido para el NPS verificado con con el sonometro
CH1
dBref2=43.8;%Nivel que indico el sonometro al realizar la calibracion en CH2
Vref2=0.275;%Voltaje medido para el NPS verificado con con el sonometro en
CH2
fs=22050;%Frecuencia de muestreo
duration=10;%Duracion en segundos de los disparos
minutos=3;%Tiempo en minutos de la duracion de la prueba de adquisicion
Notriggers=minutos*60/duration;%Numero de disparos realizados en la prueba
ymin=20;
ymax=90;
NPSymin=-10;
NPSymax=80;
%*****
%*****

%INICIA SECCION DE GRABACION CON WINSOUND
disp('SE MEDIRA EL RUIDO EN DOS CANALES PUDIENDO CALCULAR EL
AISLAMIENTO QUE EXISTE ENTRE ELLOS')
disp('O BIEN PUEDE MEDIR EL AISLAMIENTO MAXIMO EN LA
CARACTERIZACION DE LA CAMARA EN UN SOLO PUNTO')
disp('*****CON LOS INSTRUMENTOS ENCENDIDOS*****')
disp('*')
Comentario1=sprintf('El grabado de los datos tomara %2.0f minutos.',minutos);
disp(Comentario1)

Comentario2=sprintf('Estan programados %3d disparos.',Notriggers');
disp(Comentario2)
disp('*')

disp('Para comenzar a grabar, presione ENTER')
pause
disp('*')
disp('GRABANDO DATOS AL ARCHIVO RdeF2CHON.daq, ESPERE.....')
disp('*')
```

```

ai = analoginput('winsound'); %Las vías y los canales de entrada y salida
se establecen -
ch= addchannel(ai,1:2);

set(ai,'SampleRate',fs);

ActualRate=get(ai,'SampleRate');
set(ai,'SamplesPerTrigger',duration*ActualRate);%

set(ai,'TriggerRepeat', Notriggers);%Cantidad de disparos
set(ai,'LogFileName','RdeF2CHON.daq');%Se crea el aisla2ch.daq
set(ai,'LogToDiskMode','Overwrite');
set(ai,'LoggingMode','Disk&Memory');

start(ai); %Comienza la reproducción / grabación.
while (strcmp(ai.Running,'On'));
end
delete(ai); %Que borre lo que tiene almacenado 'ai' para prepararse de nuevo
%TERMINA SECCION DE GRABACION CON WINSOUND

disp('*****')
disp('ARCHIVO RdeF2CHON.daq GENERADO')
disp('GRABADO DE DATOS TERMINADO')
disp('AHORA EJECUTE EL PROGRAMA RdeFProcesamientoON.m')
disp('*****')

```

RdeFProcesamientoON.m

```
disp('*****')
disp('ESTE PROGRAMA PROCESA LOS DATOS DEL ARCHIVO
RdeF2CHON.dat')
disp('*****CON LOS INSTRUMENTOS ENCENDIDOS*****')
disp('Para procesar los datos del archivo y medir el Ruido de Fondo con
aislamiento entre canales, presione ENTER')
disp('*****')
pause

clc

for q=1:Notriggers

disp('LLAMANDO DATOS, ESPERE')
Cdisparo=sprintf('PROCESANDO DATOS DEL DISPARO NUMERO %2d',q);
disp(Cdisparo)
disp('*')

[data,time]=daqread('RdeF2CHON.daq','Triggers',[q]);%CREA UN DATA con el
disparo q

data1=data(:,1);%Se crea un vector que tenga solo los datos del CH1
data2=data(:,2);%Se crea un vector que tenga solo los datos del CH2

RMSrfondo1(q) = sqrt(sum(data1.*data1)/size(data1,1));%Se calcula el RMS de
la señal CH1 capturada
RMSrfondo2(q) = sqrt(sum(data2.*data2)/size(data2,1));%Se calcula el RMS de
la señal CH2 capturada

BdeF=[100 125 160 200 250 315 400 500 630 800 1000 1250 1600 2000 2500
3150 4000 5000];
nt=length(BdeF);%Se van a tener 19 tercios de octava de 125Hz hasta 8KHz

for n = 1:nt %Se va a procesar la muestra de disparo q en tercios de octava
f = BdeF(n);
wn = [round((f)*2^(-1/6)) round((f)*2^(1/6))];%Se establecen las
frecuencias de corte
[b,a] = butter(3,wn/fs*2);%filtro fabricado

filtrada1 = filter(b,a,data1);
filtrada2 = filter(b,a,data2);
yfilt1(:,n)=filtrada1;%En la columna n, todos los datos (filas) se almacena
la señal filtrada del tercio n evaluado.
yfilt2(:,n)=filtrada2;%En la columna n, todos los datos (filas) se almacena
la señal filtrada del tercio n evaluado.
```



```

    niv1(n) = sqrt(sum(filtrada1.*filtrada1)/size(filtrada1,1));%Se calcula el
RMS de la señal filtrada
    niv2(n) = sqrt(sum(filtrada2.*filtrada2)/size(filtrada2,1));%Se calcula el
RMS de la señal filtrada
end

```

```

    tabladenivacumulada1(:,q)=niv1';%niv' porque debo acumular una columna
de 18 valores
    tabladenivacumulada2(:,q)=niv2';%niv' porque debo acumular una columna
de 18 valores

```

```

TablaacumuladadBprov1(:,q)=20*log10(tabladenivacumulada1(:,q)/Vref1)+dBref1;

```

```

TablaacumuladadBprov2(:,q)=20*log10(tabladenivacumulada2(:,q)/Vref2)+dBref2;

```

```

    AislamientoProv=TablaacumuladadBprov2(:,q)-TablaacumuladadBprov1(:,q);

```

```

    TablaRecienCalculado=[BdeF'                               TablaacumuladadBprov1(:,q)
TablaacumuladadBprov2(:,q) AislamientoProv];%Se prepara una tabla con el
    %ultimo calculo de dB realizado hasta el momento

```

```

    clc
%////////////////////////////////////
disp('          NIVELES CALCULADOS HASTA EL MOMENTO')
disp('          RUIDO MEDIDO EN 2 CANALES Y AISLAMIENTO')
disp('          APARATOS ENCENDIDOS')
disp('*****')
disp('Frecuencia  NPS(dB)CH1  NPS(dB)CH2  Aislamiento (dB) Trigger')
TRC=num2str(TablaRecienCalculado,'%15.1f');
disp(TRC)
disp('*****')
disp('*')
%////////////////////////////////////
end

```

```

%CALCULO DE LOS RMS PROMEDIO POR TERCIOS DE OCTAVA Y EN dB
Identidad=ones(Notriggers,1);%Ntriggers=3

```

```

promediosdeRMS1=(tabladenivacumulada1*Identidad)/Notriggers;
promediosdeRMS2=(tabladenivacumulada2*Identidad)/Notriggers;

```

```

ValoresdB1=20*log10(promediosdeRMS1/Vref1)+dBref1;
ValoresdB2=20*log10(promediosdeRMS2/Vref2)+dBref2;
AislamientodBtercios=ValoresdB2-ValoresdB1;
%*****

```

```

%/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*
%/*/*/*/*CALCULO DE NIVELES GLOBALES POR CANAL /*/*/*/*/*/*
%/*/*/*/*/*/*EN LAS BANDAS DE ESTUDIO/*/*/*/*/*/*/*/*/*
dB1005KCH1=SumadBf(ValoresdB1);
dB1005KCH2=SumadBf(ValoresdB2);
Aisla1005K=dB1005KCH2-dB1005KCH1;
STC=STCf(AislamientoBtercios);%Se calcula el STC medido en ese punto
%/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*
%/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*

%CALCULO DEL RMS *GLOBAL* POR TODOS LOS ENSAYOS
RMSpromedio1=(RMSfondo1*Identidad)/Notriggers;
RMSpromedio2=(RMSfondo2*Identidad)/Notriggers;

RdeFglobaldB1=20*log10(RMSpromedio1/Vref1)+dBref1;
RdeFglobaldB2=20*log10(RMSpromedio2/Vref2)+dBref2;
AislamientoGlobal=RdeFglobaldB2-RdeFglobaldB1;%Calculo del aislamiento
global
%*****

Tablafinal=[BdeF' ValoresdB1 ValoresdB2 AislamientoBtercios]; %tABLA DE
PAREJAS BANDA-NIVEL

disp('*')
disp('*****')
disp('NIVELES YA CALCULADOS CON EXITO, para ver los resultados finales
pulse ENTER')
pause
clc

disp('*')
disp('*****')
NPSCH1=sprintf('NPS Global en CH1 = %3.1f dB.',RdeFglobaldB1);
disp(NPSCH1)
NPSCH2=sprintf('NPS Global en CH2 = %3.1f dB.',RdeFglobaldB2);
disp(NPSCH2)
disp('*****')

AG=sprintf('Aislamiento Global = %3.1f dB.',AislamientoGlobal);
disp(AG)
disp('*****')

%/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*
disp('*')
disp('*****')
dBCH1=sprintf('NPS Banda 100-5KHz CH1 = %3.1f dB.',dB1005KCH1);
disp(dBCH1)
dBCH2=sprintf('NPS Banda 100-5KHz CH2 = %3.1f dB.',dB1005KCH2);
disp(dBCH2)
disp('*****')

```



```

subplot(2,2,3)
bar(Aislamiento dB tercios)
grid
title('AISLAMIENTO ENTRE CANALES CH1 Y CH2 TERCIOS DE OCTAVA')
xlabel('Numero de banda')
ylabel('R(dB)')

subplot(2,2,4)
plot(BdeF,Aislamiento dB tercios)
hold on
plot(BdeF,Aislamiento dB tercios,'*r')
grid

title('AISLAMIENTO ENTRE CANALES CH1 Y CH2 TERCIOS DE OCTAVA')
xlabel('Frecuencia (Hz)')
ylabel('R(dB)')

%*****GRAFICADO FORMAL DE
RESULTADOS*****
disp('Para ver la graficas FORMALES pulse ENTER repetidamente')
pause

figure
subplot(1,2,1)
semilogx(BdeF,Valores dB1,'LineWidth',1)
hold on
semilogx(BdeF,Valores dB1,'+r','LineWidth',2)
hold on
semilogx(BdeF,Valores dB2,'LineWidth',1)
hold on
semilogx(BdeF,Valores dB2,'*b','LineWidth',2)
hold on
GraticulaGeneral(NPSymin,NPSymax)
title(['RUIDO DE FONDO CH1 (++) NPS =
',num2str(round(dB1005KCH1*10)/10),'dB Y CH2 (***) NPS =
',num2str(round(dB1005KCH2*10)/10),'dB'])
ylabel('NIVEL DE PRESION SONORA (dB)')

subplot(1,2,2)
semilogx(BdeF,Aislamiento dB tercios,'LineWidth',3)
hold on
semilogx(BdeF,Aislamiento dB tercios,'*r','LineWidth',2)
GraticulaGeneral(ymin,ymax)
title(['AISLAMIENTO SONORO ENTRE CH1 Y CH2 STC =
',num2str(round(10*STC)/10),' dB'])
ylabel('AISLAMIENTO R (dB)')

figure
subplot(1,2,1)

```

```

semilogx(BdeF,ValoresdB1,'r','LineWidth',2)
hold on
semilogx(BdeF,ValoresdB1,'*b','LineWidth',2)
GraticulaGeneral(NPSymin,NPSymax)
title(['RUIDO MEDIDO PARA CH1 (RECEPTOR) NPS =
',num2str(round(dB1005KCH1*10)/10),'dB'])
ylabel('NIVEL DE PRESION SONORA (dB)')

```

```

subplot(1,2,2)
semilogx(BdeF,ValoresdB2,'LineWidth',2)
hold on
semilogx(BdeF,ValoresdB2,'*r','LineWidth',2)
GraticulaGeneral(NPSymin,NPSymax)
title(['RUIDO MEDIDO PARA CH2 (EMISOR) NPS =
',num2str(round(dB1005KCH2*10)/10),'dB'])
ylabel('NIVEL DE PRESION SONORA (dB)')

```

```

pause
figure
semilogx(BdeF,ValoresdB1,'r','LineWidth',2)
hold on
semilogx(BdeF,ValoresdB1,'or','LineWidth',1)
hold on
semilogx(BdeF,ValoresdB2,'LineWidth',2)
hold on
semilogx(BdeF,ValoresdB2,'vb','LineWidth',1)
hold on
GraticulaGeneral(NPSymin,NPSymax)
title(['"RUIDO DE FONDO" CH1 RECEPTOR (rojo -o-o-o-) NPS =
',num2str(round(dB1005KCH1*10)/10),'dB CH2 EMISOR (azul -v-v-v-) NPS
= ',num2str(round(dB1005KCH2*10)/10),'dB ON'])
ylabel('NIVEL DE PRESION SONORA (dB)')

```

```

pause
figure
semilogx(BdeF,AislamientoBtercios,'LineWidth',3)
hold on
semilogx(BdeF,AislamientoBtercios,'*r','LineWidth',2)
GraticulaGeneral(ymin,ymax)
title("")
title(['AISLAMIENTO SONORO ENTRE CH1 Y CH2 STC =
',num2str(round(10*STC)/10),' dB ON'])

ylabel('AISLAMIENTO R (dB)')

```

```

disp('PROGRAMA TERMINADO')

```

```

ValoresdB1ON=ValoresdB1;

```

```
ValoresdB2ON=ValoresdB2;
```

```
save ValoresdB1ON.mat ValoresdB1ON  
save ValoresdB2ON.mat ValoresdB2ON  
save BdeF.mat BdeF  
save dB1005KCH1ON.mat dB1005KCH1  
save dB1005KCH2ON.mat dB1005KCH2
```

```
disp('Se han generado los archivos: save ValoresdB1ON.mat  
ValoresdB2ON.mat y BdeF.mat')  
disp('Si ya evaluo el aislamiento con el programa CalculoPABCDEF2.mat')  
disp('ahora corra el programa TFlanqueo.m para evaluar la transmision por  
flanqueo')
```

RdeFProcesamientoOFF.

```
disp('*****')
disp('ESTE PROGRAMA PROCESA LOS DATOS DEL ARCHIVO
RdeF2CHON.dat')
disp('*****CON LOS INSTRUMENTOS APAGADOS*****')
disp('Para procesar los datos del archivo y medir el Ruido de Fondo con
aislamiento entre canales, presione ENTER')
disp('*****')
pause

clc

for q=1:Notriggers

disp('LLAMANDO DATOS, ESPERE')
Cdisparo=sprintf('PROCESANDO DATOS DEL DISPARO NUMERO %2d',q);
disp(Cdisparo)
disp('*')

[data,time]=daqread('RdeF2CHOFF.daq','Triggers',[q]);%CREA UN DATA con
el disparo q

data1=data(:,1);%Se crea un vector que tenga solo los datos del CH1
data2=data(:,2);%Se crea un vector que tenga solo los datos del CH2

RMSrfondo1(q) = sqrt(sum(data1.*data1)/size(data1,1));%Se calcula el RMS de
la señal CH1 capturada
RMSrfondo2(q) = sqrt(sum(data2.*data2)/size(data2,1));%Se calcula el RMS de
la señal CH2 capturada

BdeF=[100 125 160 200 250 315 400 500 630 800 1000 1250 1600 2000 2500
3150 4000 5000];
nt=length(BdeF);%Se van a tener 19 tercios de octava de 125Hz hasta 8KHz

for n = 1:nt %Se va a procesar la muestra de disparo q en tercios de octava
f = BdeF(n);
wn = [round((f)*2^(-1/6)) round((f)*2^(1/6))];%Se establecen las
frecuencias de corte
[b,a] = butter(3,wn/fs*2);%filtro fabricado

filtrada1 = filter(b,a,data1);
filtrada2 = filter(b,a,data2);
yfilt1(:,n)=filtrada1;%En la columna n, todos los datos (filas) se almacena
la señal filtrada del tercio n evaluado.
yfilt2(:,n)=filtrada2;%En la columna n, todos los datos (filas) se almacena
la señal filtrada del tercio n evaluado.
```

```

        niv1(n) = sqrt(sum(filtrada1.*filtrada1)/size(filtrada1,1));%Se calcula el
RMS de la señal filtrada
        niv2(n) = sqrt(sum(filtrada2.*filtrada2)/size(filtrada2,1));%Se calcula el
RMS de la señal filtrada
    end

```

```

        tabladenivacumulada1(:,q)=niv1';%niv' porque debo acumular una columna
de 18 valores
        tabladenivacumulada2(:,q)=niv2';%niv' porque debo acumular una columna
de 18 valores

```

```

TablaacumuladadBprov1(:,q)=20*log10(tabladenivacumulada1(:,q)/Vref1)+dBref1;

```

```

TablaacumuladadBprov2(:,q)=20*log10(tabladenivacumulada2(:,q)/Vref2)+dBref2;

```

```

    AislamientoProv=TablaacumuladadBprov2(:,q)-TablaacumuladadBprov1(:,q);

```

```

    TablaRecienCalculado=[BdeF'
        TablaacumuladadBprov1(:,q)
TablaacumuladadBprov2(:,q) AislamientoProv];%Se prepara una tabla con el
    %ultimo calculo de dB realizado hasta el momento

```

```

        clc
        %////////////////////////////////////
disp('          NIVELES CALCULADOS HASTA EL MOMENTO')
disp('          RUIDO MEDIDO EN 2 CANALES Y AISLAMIENTO')
disp('          APARATOS APAGADOS')
disp('*****')
disp('Frecuencia  NPS(dB)CH1  NPS(dB)CH2  Aislamiento (dB) Trigger')
TRC=num2str(TablaRecienCalculado,'%15.1f');
disp(TRC)
disp('*****')
disp('*')
        %////////////////////////////////////
    end

```

```

%CALCULO DE LOS RMS PROMEDIO POR TERCIOS DE OCTAVA Y EN dB
Identidad=ones(Notriggers,1);%Ntriggers=3

```

```

promediosdeRMS1=(tabladenivacumulada1*Identidad)/Notriggers;
promediosdeRMS2=(tabladenivacumulada2*Identidad)/Notriggers;

```

```

ValoresdB1=20*log10(promediosdeRMS1/Vref1)+dBref1;
ValoresdB2=20*log10(promediosdeRMS2/Vref2)+dBref2;
AislamientodBtercios=ValoresdB2-ValoresdB1;
%*****

```

```

%/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*

```



```

%/*/*/*/*CALCULO DE NIVELES GLOBALES POR CANAL */*/*/*/*/*/
%/*/*/*/*/*/*EN LAS BANDAS DE ESTUDIO/*/*/*/*/*/*/*/*/*/*/*/
dB1005KCH1=SumadBf(ValoresdB1);
dB1005KCH2=SumadBf(ValoresdB2);
Aisla1005K=dB1005KCH2-dB1005KCH1;
STC=STCf(AislamientoBtercios);%Se calcula el STC medido en ese punto
%/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/
%/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/

```

```

%CALCULO DEL RMS *GLOBAL* POR TODOS LOS ENSAYOS
RMSpromedio1=(RMSfondo1*Identidad)/Notriggers;
RMSpromedio2=(RMSfondo2*Identidad)/Notriggers;

RdeFglobaldB1=20*log10(RMSpromedio1/Vref1)+dBref1;
RdeFglobaldB2=20*log10(RMSpromedio2/Vref2)+dBref2;
AislamientoGlobal=RdeFglobaldB2-RdeFglobaldB1;%Calculo del aislamiento
global
%*****

```

Tablafinal=[BdeF' ValoresdB1 ValoresdB2 AislamientoBtercios]; %tABLA DE PAREJAS BANDA-NIVEL

```

disp('*')
disp('*****')
disp('NIVELES YA CALCULADOS CON EXITO, para ver los resultados finales
pulse ENTER')
pause
clc

```

```

disp('*')
disp('*****')
NPSCH1=sprintf('NPS Global en CH1 = %3.1f dB.',RdeFglobaldB1);
disp(NPSCH1)
NPSCH2=sprintf('NPS Global en CH2 = %3.1f dB.',RdeFglobaldB2);
disp(NPSCH2)
disp('*****')

```

```

AG=sprintf('Aislamiento Global = %3.1f dB.',AislamientoGlobal);
disp(AG)
disp('*****')

```

```

%/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/
disp('*')
disp('*****')
dBCH1=sprintf('NPS Banda 100-5KHz CH1 = %3.1f dB.',dB1005KCH1);
disp(dBCH1)
dBCH2=sprintf('NPS Banda 100-5KHz CH2 = %3.1f dB.',dB1005KCH2);
disp(dBCH2)
disp('*****')

```



```

subplot(2,2,3)
bar(AislamientoBtercios)
grid
title('AISLAMIENTO ENTRE CANALES CH1 Y CH2 TERCIOS DE OCTAVA')
xlabel('Numero de banda')
ylabel('R(dB)')

subplot(2,2,4)
plot(BdeF,AislamientoBtercios)
hold on
plot(BdeF,AislamientoBtercios,'*r')
grid

title('AISLAMIENTO ENTRE CANALES CH1 Y CH2 TERCIOS DE OCTAVA')
xlabel('Frecuencia (Hz)')
ylabel('R(dB)')

%*****GRAFICADO FORMAL DE
RESULTADOS*****
disp('Para ver la graficas FORMALES pulse ENTER repetidamente')
pause

figure
subplot(1,2,1)
semilogx(BdeF,ValoresdB1,'LineWidth',1)
hold on
semilogx(BdeF,ValoresdB1,'+r','LineWidth',2)
hold on
semilogx(BdeF,ValoresdB2,'LineWidth',1)
hold on
semilogx(BdeF,ValoresdB2,'*b','LineWidth',2)
hold on
GraticulaGeneral(NPSymin,NPSymax)
title(['RUIDO DE FONDO CH1 (++) NPS =
',num2str(round(dB1005KCH1*10)/10),'dB Y CH2 (***) NPS =
',num2str(round(dB1005KCH2*10)/10),'dB'])
ylabel('NIVEL DE PRESION SONORA (dB)')

subplot(1,2,2)
semilogx(BdeF,AislamientoBtercios,'LineWidth',3)
hold on
semilogx(BdeF,AislamientoBtercios,'*r','LineWidth',2)
GraticulaGeneral(ymin,ymax)
title(['AISLAMIENTO SONORO ENTRE CH1 Y CH2 STC =
',num2str(round(10*STC)/10),' dB'])
ylabel('AISLAMIENTO R (dB)')

pause
figure
subplot(1,2,1)

```

```

semilogx(BdeF,ValoresdB1,'r','LineWidth',2)
hold on
semilogx(BdeF,ValoresdB1,'*b','LineWidth',2)
GraticulaGeneral(NPSymin,NPSymax)
title(['RUIDO MEDIDO PARA CH1 (RECEPTOR) NPS =
',num2str(round(dB1005KCH1*10)/10),'dB'])
ylabel('NIVEL DE PRESION SONORA (dB)')

subplot(1,2,2)
semilogx(BdeF,ValoresdB2,'LineWidth',2)
hold on
semilogx(BdeF,ValoresdB2,'*r','LineWidth',2)
GraticulaGeneral(NPSymin,NPSymax)
title(['RUIDO MEDIDO PARA CH2 (EMISOR) NPS =
',num2str(round(dB1005KCH2*10)/10),'dB'])
ylabel('NIVEL DE PRESION SONORA (dB)')

pause
figure
semilogx(BdeF,ValoresdB1,'r','LineWidth',2)
hold on
semilogx(BdeF,ValoresdB1,'or','LineWidth',1)
hold on
semilogx(BdeF,ValoresdB2,'LineWidth',2)
hold on
semilogx(BdeF,ValoresdB2,'vb','LineWidth',1)
hold on
GraticulaGeneral(NPSymin,NPSymax)
title(['"RUIDO DE FONDO" CH1 RECEPTOR (rojo -o-o-o-) NPS =
',num2str(round(dB1005KCH1*10)/10),'dB CH2 EMISOR (azul -v-v-v-) NPS
= ',num2str(round(dB1005KCH2*10)/10),'dB OFF'])
ylabel('NIVEL DE PRESION SONORA (dB)')

pause
figure
semilogx(BdeF,AislamientoBtercios,'LineWidth',3)
hold on
semilogx(BdeF,AislamientoBtercios,'*r','LineWidth',2)
GraticulaGeneral(ymin,ymax)
title('')
title(['AISLAMIENTO SONORO ENTRE CH1 Y CH2 STC =
',num2str(round(10*STC)/10),' dB OFF'])

ylabel('AISLAMIENTO R (dB)')

disp('PROGRAMA TERMINADO')

ValoresdB1OFF=ValoresdB1;
ValoresdB2OFF=ValoresdB2;

```

```
save ValoresdB1OFF.mat ValoresdB1OFF
save ValoresdB2OFF.mat ValoresdB2OFF
```

```
save BdeF.mat BdeF
save dB1005KCH1OFF.mat dB1005KCH1
save dB1005KCH2OFF.mat dB1005KCH2
```

```
disp('Se han generado los archivos: save ValoresdB1OFF.mat
ValoresdB2OFF.mat y BdeF.mat')
```

```
disp('Si ya evaluo el aislamiento con el programa CalculoPABCDEF2.mat')
```

```
disp('ahora corra el programa TFlanqueo.m para evaluar la transmision por
flanqueo')
```

AislamientoPreliminar.m.

```
clear
clc
close all
%*****INDICAR dB DEL SONOMETRO AL CALIBRAR Y VrefRMS MEDIDO
PARA CH1 Y CH2*****
dBref1=73.8;%Nivel que indico el sonometro al realizar la calibracion en CH1
Vref1=0.255;%Voltaje medido para el NPS verificado con con el sonometro
CH1
dBref2=113.8;%Nivel que indico el sonometro al realizar la calibracion en CH2
Vref2=0.275;%Voltaje medido para el NPS verificado con con el sonometro en
CH2

fs=22050;%Frecuencia de muestreo
duration=5;%Duracion en segundos de los disparos
minutos=0.25;%Tiempo en minutos de la duracion de la prueba de adquisicion
Notriggers=minutos*60/duration;%Numero de disparos realizados en la prueba
ymin=10;
ymax=60;
NPSymin=10;
NPSymax=100;
%*****
%*****

%INICIA SECCION DE GRABACION CON WINSOUND
disp('SE MEDIRA EL AISLAMIENTO PRELIMINAR EN DOS CANALES EN UN
SOLO PUNTO')
disp('ESTO LE SERVIRA PARA ASEGURARSE QUE EN LA CAMARA
RECEPTORA TIENE CUANDO MENO 15dB SOBRE')
disp('EL NIVEL DE RUIDO POR BANDA DE FRECUENCIA Y PUEDA
COMENZAR SUS MEDICIONES DEFINITIVAS')
disp('*')
Comentario1=sprintf('El grabado de los datos tomara %2.0f minutos.',minutos);
disp(Comentario1)

Comentario2=sprintf('Estan programados %3d disparos.',Notriggers');
disp(Comentario2)
disp('*')

disp('Para comezar a grabar, presione ENTER')
pause
disp('*')
disp('GRABANDO DATOS AL ARCHIVO RPrelim2CH.daq, ESPERE.....')
disp('*')
```

```

ai = analoginput('winsound'); %Las vías y los canales de entrada y salida
se establecen -
ch= addchannel(ai,1:2);

set(ai,'SampleRate',fs);

ActualRate=get(ai,'SampleRate');
set(ai,'SamplesPerTrigger',duration*ActualRate);%

set(ai,'TriggerRepeat', Notriggers);%Cantidad de disparos
set(ai,'LogFileName','RPrelim2CH.daq');%Se crea el aisa2ch.daq
set(ai,'LogToDiskMode','Overwrite');
set(ai,'LoggingMode','Disk&Memory');

start(ai); %Comienza la reproducción / grabación.
while (strcmp(ai.Running,'On'));
end
delete(ai); %Que borre lo que tiene almacenado 'ai' para prepararse de nuevo
%TERMINA SECCION DE GRABACION CON WINSOUND

disp('*****')
disp('ARCHIVO RPrelim2CH.daq GENERADO')
disp('GRABADO DE DATOS TERMINADO')
disp('Para procesar los datos del archivo y medir el Ruido de Fondo con
aislamiento entre canales, presione ENTER')
disp('*****')

pause
clc

for q=1:Notriggers

disp('LLAMANDO DATOS, ESPERE')
Cdisparo=sprintf('PROCESANDO DATOS DEL DISPARO NUMERO %2d',q);
disp(Cdisparo)
disp('*')

[data,time]=daqread('RPrelim2CH.daq','Triggers',[q]);%CREA UN DATA con el
disparo q

data1=data(:,1);%Se crea un vector que tenga solo los datos del CH1
data2=data(:,2);%Se crea un vector que tenga solo los datos del CH2

RMSrfondo1(q) = sqrt(sum(data1.*data1)/size(data1,1));%Se calcula el RMS de
la señal CH1 capturada
RMSrfondo2(q) = sqrt(sum(data2.*data2)/size(data2,1));%Se calcula el RMS de
la señal CH2 capturada

BdeF=[100 125 160 200 250 315 400 500 630 800 1000 1250 1600 2000 2500
3150 4000 5000];

```

```

nt=length(BdeF);%Se van a tener 19 tercios de octava de 125Hz hasta 8KHz

for n = 1:nt %Se va a procesar la muestra de disparo q en tercios de octava
    f = BdeF(n);
    wn = [round((f)*2^(-1/6)) round((f)*2^(1/6))];%Se establecen las
frecuencias de corte
    [b,a] = butter(3,wn/fs*2);%filtro fabricado

    filtrada1 = filter(b,a,data1);
    filtrada2 = filter(b,a,data2);
    yfilt1(:,n)=filtrada1;%En la columna n, todos los datos (filas) se almacena
la señal filtrada del tercio n evaluado.
    yfilt2(:,n)=filtrada2;%En la columna n, todos los datos (filas) se almacena
la señal filtrada del tercio n evaluado.
    niv1(n) = sqrt(sum(filtrada1.*filtrada1)/size(filtrada1,1));%Se calcula el
RMS de la señal filtrada
    niv2(n) = sqrt(sum(filtrada2.*filtrada2)/size(filtrada2,1));%Se calcula el
RMS de la señal filtrada
end

    tabladenivacumulada1(:,q)=niv1';%niv' porque debo acumular una columna
de 18 valores
    tabladenivacumulada2(:,q)=niv2';%niv' porque debo acumular una columna
de 18 valores

TablaacumuladadBprov1(:,q)=20*log10(tabladenivacumulada1(:,q)/Vref1)+dBref1;

TablaacumuladadBprov2(:,q)=20*log10(tabladenivacumulada2(:,q)/Vref2)+dBref2;
AislamientoProv=TablaacumuladadBprov2(:,q)-TablaacumuladadBprov1(:,q);

    TablaRecienCalculado=[BdeF' TablaacumuladadBprov1(:,q)
TablaacumuladadBprov2(:,q) AislamientoProv];%Se prepara una tabla con el
%ultimo calculo de dB realizado hasta el momento

    clc
%////////////////////////////////////
disp('          NIVELES CALCULADOS HASTA EL MOMENTO')
disp('          RUIDO MEDIDO EN 2 CANALES Y AISLAMIENTO')
disp('*****')
disp('Frecuencia  NPS(dB)CH1  NPS(dB)CH2  Aislamiento (dB) Trigger')
TRC=num2str(TablaRecienCalculado,'%15.1f');
disp(TRC)
disp('*****')
disp('*')
%////////////////////////////////////
end

```



```

% plot(BdeF,ValoresdB1,'r')
% hold on
% plot(BdeF,ValoresdB1,'*r')
% hold on
% plot(BdeF,ValoresdB2,'b')
% hold on
% plot(BdeF,ValoresdB2,'.b')
% hold on
% grid
% title('NIVEL DE RUIDO DE CH1(***) CH2(...) TERCIOS DE OCTAVA')
% xlabel('Frecuencia (Hz)')
% ylabel('Nivel de ruido(dB)')
%
% subplot(2,2,3)
% bar(AislamientoBtercios)
% grid
% title('AISLAMIENTO ENTRE CANALES CHI Y CH2 TERCIOS DE OCTAVA')
% xlabel('Numero de banda')
% ylabel('R(dB)')
%
% subplot(2,2,4)
% plot(BdeF,AislamientoBtercios)
% hold on
% plot(BdeF,AislamientoBtercios,'*r')
% grid
%
% title('AISLAMIENTO ENTRE CANALES CHI Y CH2 TERCIOS DE OCTAVA')
% xlabel('Frecuencia (Hz)')
% ylabel('R(dB)')

```

```

%*****GRAFICADO FORMAL DE
RESULTADOS*****
disp('Para ver la graficas FORMALES pulse ENTER repetidamente')
pause

```

```

figure
subplot(1,2,1)
semilogx(BdeF,ValoresdB1,'LineWidth',1)
hold on
semilogx(BdeF,ValoresdB1,'+r','LineWidth',2)
hold on
semilogx(BdeF,ValoresdB2,'LineWidth',1)
hold on
semilogx(BdeF,ValoresdB2,'*b','LineWidth',2)
hold on
GraticulaGeneral(NPSymin,NPSymax)
title(['RUIDO EN CAMARAS CH1 RECEPTORA(+++) NPS =
',num2str(round(dB1005KCH1*10)/10),'dB Y CH2 FUENTE (***) NPS =
',num2str(round(dB1005KCH2*10)/10),'dB'])
ylabel('NIVEL DE PRESION SONORA (dB)')

```

```

subplot(1,2,2)
semilogx(BdeF,AislamientoBtercios,'LineWidth',3)
hold on
semilogx(BdeF,AislamientoBtercios,'*r','LineWidth',2)
GraticulaGeneral(ymin,ymax)
title(['AISLAMIENTO SONORO ENTRE CH1 Y CH2      STC =
',num2str(round(10*STC)/10),' dB'])
ylabel('AISLAMIENTO R (dB)')

```

```

pause
figure
subplot(1,2,1)
semilogx(BdeF,ValoresdB1,'LineWidth',1)
hold on
semilogx(BdeF,ValoresdB1,'+r','LineWidth',2)
GraticulaGeneral(NPSymin,NPSymax)
title(['RUIDO MEDIDO PARA CH1      (RECEPTORA)      NPS =
',num2str(round(dB1005KCH1*10)/10),'dB'])
ylabel('NIVEL DE PRESION SONORA (dB)')

```

```

subplot(1,2,2)
semilogx(BdeF,ValoresdB2,'LineWidth',1)
hold on
semilogx(BdeF,ValoresdB2,'*b','LineWidth',2)
GraticulaGeneral(NPSymin,NPSymax)
title(['RUIDO MEDIDO PARA CH2      (FUENTE)      NPS =
',num2str(round(dB1005KCH2*10)/10),'dB'])
ylabel('NIVEL DE PRESION SONORA (dB)')

```

```

figure
semilogx(BdeF,ValoresdB1,'r','LineWidth',2)
hold on
semilogx(BdeF,ValoresdB1,'or','LineWidth',1)
hold on
semilogx(BdeF,ValoresdB2,'b','LineWidth',2)
hold on
semilogx(BdeF,ValoresdB2,'vb','LineWidth',1)
ylabel('NIVEL DE PRESION SONORA (dB)')
%title(['NIVELES POR BANDA: TRANSMITIDO (ooo)rojo NPS = 48.6dB,
INCIDENTE (vvv)azul NPS = 96.0dB'])
title(['NIVELES POR BANDA: TRANSMITIDO (o-o-o) NPS =
',num2str(round(dB1005KCH1*10)/10),'dB      INCIDENTE (v-v-v)azul NPS =
',num2str(round(dB1005KCH2*10)/10),'dB      PROBETA PS08'])
GraticulaGeneral(NPSymin,NPSymax)

```

```

pause

```

```

figure

```

```

semilogx(BdeF,AislamientoBtercios,'LineWidth',3)
hold on
semilogx(BdeF,AislamientoBtercios,'*r','LineWidth',2)
ylabel('AISLAMINETO SONORO R (dB)')
title(['AISLAMIENTO SONORO PANEL "A" STC =
',num2str(round(10*STC)/10),' dB'])
GraticulaGeneral(ymin,ymax)

% pause
% figure
% semilogx(BdeF,ValoresdB1,'LineWidth',1)
% hold on
% semilogx(BdeF,ValoresdB1,'+r','LineWidth',2)
% hold on
% semilogx(BdeF,ValoresdB2,'LineWidth',1)
% hold on
% semilogx(BdeF,ValoresdB2,'*b','LineWidth',2)
% hold on
% GraticulaGeneral(ymin,ymax)
% title(['RUIDO MEDIDO CH1 RECEPTORA (+++) NPS =
',num2str(round(dB1005KCH1*10)/10),'dB Y CH2 FUENTE (***) NPS =
',num2str(round(dB1005KCH2*10)/10),'dB'])
% ylabel('NIVEL DE PRESION SONORA (dB)')
%
% pause
% figure
% semilogx(BdeF,AislamientoBtercios,'LineWidth',3)
% hold on
% semilogx(BdeF,AislamientoBtercios,'*r','LineWidth',2)
% GraticulaGeneral(ymin,ymax)
% title('')
% title(['AISLAMIENTO SONORO ENTRE CH1 Y CH2 STC =
',num2str(round(10*STC)/10),' dB'])
%
% ylabel('AISLAMIENTO R (dB)')

disp('PROGRAMA TERMINADO')

```

AislamientoPreparativo.m.

```
%PROGRAMA DE PREPARATIVOS Y DATOS COMUNES PARA MEDIR AISLAMIENTO
```

```
clear
clc %Borrar pantalla
close all
disp('ESTE PROGRAMA SIRVE PARA DAR DE ALTA TODAS LAS VARIABLES EN COMUN QUE TIENEN LOS SIGUIENTES PROGRAMAS:')
disp('AislamientoPA.m')
disp('AislamientoPB.m')
disp('AislamientoPC.m')
disp('AislamientoPD.m')
disp('AislamientoPE.m')
disp('AislamientoPF.m')
disp('AislamientoPA2.m')
disp('AislamientoPB2.m')
disp('AislamientoPC2.m')
disp('AislamientoPD2.m')
disp('AislamientoPE2.m')
disp('AislamientoPF2.m')
disp('TAMBIEN SE CARGA EL VECTOR CON LOS TIEMPOS DE REVERBERACION QUE SE CALCULARON CON EL PROGRAMA: RevBdeFtodo')
disp('SE ACONSEJA VERIFICAR CADA UNA DE LAS VARIABLES INGRESADAS AL PROGRAMA')
%*****INDICAR dB DEL SONOMETRO Y VrefRMS MEDIDO PARA CH1 Y CH2*****
dBref1=73.8;%Nivel que indico el sonometro al realizar la calibracion en CH1
Vref1=0.255;%Voltaje medido para el NPS verificado con con el sonometro CH1
dBref2=113.8;%Nivel que indico el sonometro al realizar la calibracion en CH2
Vref2=0.275;%Voltaje medido para el NPS verificado con con el sonometro en CH2

ymin=10;%Valor limite inferior en dB para que sea graficar el aislamiento.
ymax=60;%Valor limite superior en dB para que sea graficar el aislamiento.
NPSymin=10;%Valor limite inferior en dB para que sea graficar los niveles de ruido.
NPSymax=100;

%*****
%*****
load rtprom.mat%se carga el archivo que tiene los tiempos de reverberacion
fs=22050;%Frecuencia de muestreo
duration=10;%Duracion en segundos de los disparos
minutos=2;%Tiempo en minutos de la duracion de la prueba de adquisicion por cada posicion
```

Notriggers=minutos*60/duration;%Numero de disparos realizados en la prueba

V2=77.6;%Volumen de recinto emisor (m3)

V1=68.3;%Volumen del recinto receptor (m3)

Sesp=9.75;%Area de especimen (m2)

for z=1:18

AABCDEF(z)=0.163*V1/rtprom(z);%Absorcion equivalente promedio que sera aplicada a cada ensayo para realizar la compensacion por aislamiento necesaria

NABCDEF(z)=10*log10(Sesp/AABCDEF(z));%Componente global para aplicarse a la STL final

end

pause(2)

disp('*')

disp('*')

disp('LOS AJUSTES DE LAS VARIABLES COMUNES YA ESTAN CARGADOS Y CALCULADOS')

disp('AHORA PUEDE COMENZAR LA ADQUISICION DE DATOS DE LAS 12 POSICIONES DE MEDIDA')

disp('O BIEN, SI YA LAS TERMINO, PUEDE PROCESAR LOS DATOS PARA FINALIZAR EJECUTANDO EL ARCHIVO CalculoPABCDEF2.m')

**AislamientoPA.m, AislamientoPB.m, ...'AislamientoPF.m,
'AislamientoPA2.m, ... AislamientoPF2.m.**

```
clc
close all
%INICIA SECCION DE GRABACION CON WINSOUND
disp('SE EVALUARA EL AISLAMIENTO ENTRE LOS DOS CANALES CH1 Y
CH2 POSICION DE MICS "A"')
disp('*')
Comentario1=sprintf('El grabado de los datos tomara %2.2f minutos.',minutos);
disp(Comentario1)

Comentario2=sprintf('Estan programados %3d disparos.',Notriggers');
disp(Comentario2)
disp('*')

disp('Para comenzar a grabar, presione ENTER')
pause
disp('*')
disp('GRABANDO DATOS AL ARCHIVO AislaPA.daq, ESPERE.....')
disp('*')

ai = analoginput('winsound'); %Las vías y los canales de entrada y salida
se establecen -
ch= addchannel(ai,1:2);

set(ai,'SampleRate',fs);

ActualRate=get(ai,'SampleRate');
set(ai,'SamplesPerTrigger',duration*ActualRate);%

set(ai,'TriggerRepeat', Notriggers);%Cantidad de disparos
set(ai,'LogFileName','AislaPA.daq');%Se crea el aisla2ch.daq
set(ai,'LogToDiskMode','Overwrite');
set(ai,'LoggingMode','Disk&Memory');

start(ai); %Comienza la reproducción / grabación.
while (strcmp(ai.Running,'On'));
end
delete(ai); %Que borre lo que tiene almacenado 'ai' para prepararse de nuevo
%TERMINA SECCION DE GRABACION CON WINSOUND

disp('*****')
disp('ARCHIVO AislaPA.daq GENERADO')
disp('GRABADO DE DATOS TERMINADO')
disp('Si desea procesar los datos del archivo para ver el aislamiento en este
punto, presione ENTER')
```



```

disp('De lo contrario pulse CTRL+C y cuninue evaluando en el siguiente punto
de medicion')
disp('*****')

```

```

pause
clc

```

```

for q=1:Notriggers

```

```

disp('LLAMANDO DATOS, ESPERE')
Cdisparo=sprintf('PROCESANDO DATOS DEL DISPARO NUMERO %2d',q);
disp(Cdisparo)
disp('*')

```

```

[data,time]=daqread('AislaPA.daq','Triggers',[q]);%CREA UN DATA con el
disparo q

```

```

data1=data(:,1);%Se crea un vector que tenga solo los datos del CH1
data2=data(:,2);%Se crea un vector que tenga solo los datos del CH2

```

```

RMSrfondo1(q) = sqrt(sum(data1.*data1)/size(data1,1));%Se calcula el RMS de
la señal CH1 capturada
RMSrfondo2(q) = sqrt(sum(data2.*data2)/size(data2,1));%Se calcula el RMS de
la señal CH2 capturada

```

```

BdeF=[100 125 160 200 250 315 400 500 630 800 1000 1250 1600 2000 2500
3150 4000 5000];

```

```

nt=length(BdeF);%Se van a tener 19 tercios de octava de 125Hz hasta 8KHz

```

```

for n = 1:nt %Se va a procesar la muestra de disparo q en tercios de octava
f = BdeF(n);
wn = [round((f)*2^(-1/6)) round((f)*2^(1/6))];%Se establecen las
frecuencias de corte
[b,a] = butter(3,wn/fs*2);%filtro fabricado

```

```

filtrada1 = filter(b,a,data1);
filtrada2 = filter(b,a,data2);
yfilt1(:,n)=filtrada1;%En la columna n, todos los datos (filas) se almacena
la señal filtrada del tercio n evaluado.
yfilt2(:,n)=filtrada2;%En la columna n, todos los datos (filas) se almacena
la señal filtrada del tercio n evaluado.
niv1(n) = sqrt(sum(filtrada1.*filtrada1)/size(filtrada1,1));%Se calcula el
RMS de la señal filtrada
niv2(n) = sqrt(sum(filtrada2.*filtrada2)/size(filtrada2,1));%Se calcula el
RMS de la señal filtrada
end

```

```

tabladenivacumulada1(:,q)=niv1';%niv' porque debo acumular una columna
de 18 valores

```

tabladenivacumulada2(:,q)=niv2';%niv' porque debo acumular una columna de 18 valores

TablaacumuladadBprov1(:,q)=20*log10(tabladenivacumulada1(:,q)/Vref1)+dBref1;

TablaacumuladadBprov2(:,q)=20*log10(tabladenivacumulada2(:,q)/Vref2)+dBref2;

AislamientoProv=TablaacumuladadBprov2(:,q)-
TablaacumuladadBprov1(:,q)+NABCDEF';

TablaRecienCalculado=[BdeF' TablaacumuladadBprov1(:,q)
TablaacumuladadBprov2(:,q) NABCDEF' AislamientoProv];%Se prepara una tabla con el
%ultimo calculo de dB realizado hasta el momento

```

clc
%////////////////////////////////////
disp('          NIVELES CALCULADOS HASTA EL MOMENTO')
disp('*****')
disp('Frecuencia  NPS(dB)CH1  NPS(dB)CH2    10log(S/A)  Aislamiento
(dB) Trigger')
TRC=num2str(TablaRecienCalculado,'%15.1f');
disp(TRC)
disp('*****')
disp('*')
%////////////////////////////////////
end

```

```

disp('*')
disp('*****')
disp('NIVELES YA CALCULADOS, para ver los resultados finales pulse
ENTER')
pause
clc

```

%CALCULO DE LOS RMS PROMEDIO POR TERCIOS DE OCTAVA Y EN dB
Identidad=ones(Notriggers,1);%Ntriggers=3

promediosdeRMS1=(tabladenivacumulada1*Identidad)/Notriggers;
promediosdeRMS2=(tabladenivacumulada2*Identidad)/Notriggers;

ValoresdB1=20*log10(promediosdeRMS1/Vref1)+dBref1;
ValoresdB2=20*log10(promediosdeRMS2/Vref2)+dBref2;
AislamientoBtercios=ValoresdB2-ValoresdB1+NABCDEF';%Le estoy
agregando el termino NA debido al especimen
%*****

```

%/**/**/**/**/**/**/**/**/**/**/**/**/**/**/**/**/**/**/**/**/**/**/
%/**/**/**CALCULO DE NIVELES GLOBALES POR CANAL **/**/**/**/
%/**/**/**/**EN LAS BANDAS DE ESTUDIO**/**/**/**/**/**/
dB1005KCH1=SumadBf(ValoresdB1);
dB1005KCH2=SumadBf(ValoresdB2);
Aisla1005K=dB1005KCH2-dB1005KCH1;
STC=STCf(AislamientodBtercios);%Se calcula el STC medido en ese punto
%/**/**/**/**/**/**/**/**/**/**/**/**/**/**/**/**/**/**/**/**/**/**/
%/**/**/**/**/**/**/**/**/**/**/**/**/**/**/**/**/**/**/**/**/**/**/

%CALCULO DEL RMS *GLOBAL* POR TODOS LOS ENSAYOS
RMSpromedio1=(RMSrfondo1*Identidad)/Notrigger;
RMSpromedio2=(RMSrfondo2*Identidad)/Notrigger;

RdeFglobaldB1=20*log10(RMSpromedio1/Vref1)+dBref1;
RdeFglobaldB2=20*log10(RMSpromedio2/Vref2)+dBref2;
AislamientoGlobal=RdeFglobaldB2-RdeFglobaldB1;%Calculo del aislamiento
global
%*****

Tablafinal=[BdeF' ValoresdB1 ValoresdB2 NABCDEF' AislamientodBtercios];
%tABLA DE PAREJAS BANDA-NIVEL

disp('*')
disp('*****')
NPSCH1=sprintf('NPS Global en CH1 = %3.1f dB.',RdeFglobaldB1);
disp(NPSCH1)
NPSCH2=sprintf('NPS Global en CH2 = %3.1f dB.',RdeFglobaldB2);
disp(NPSCH2)
disp('*****')

AG=sprintf('Aislamiento Global = %3.1f dB.',AislamientoGlobal);
disp(AG)
disp('*****')

%/**/**/**/**/**/**/**/**/**/**/**/**/**/**/**/**/**/**/**/**/**/**/
disp('*')
disp('*****')
dBCH1=sprintf('NPS Banda 100-5KHz CH1 = %3.1f dB.',dB1005KCH1);
disp(dBCH1)
dBCH2=sprintf('NPS Banda 100-5KHz CH2 = %3.1f dB.',dB1005KCH2);
disp(dBCH2)
disp('*****')

AdB=sprintf('Aislamiento Global Banda 100-5KHz= %3.1f dB.',Aisla1005K);
disp(AdB)
disp('*****')
STCtexto=sprintf('STC Banda 100-5KHz= %3.1f dB.',STC);
disp(STCtexto)
disp('*****')

```

```

%*/**/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*
%*/**/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*

%//////////////////////////////////////
disp('*')
disp('          TABLA DE RESULTADOS')
disp('*****')
disp('Frecuencia  NPS(dB)CH1  NPS(dB)CH2    10log(S/A)  Aislamiento
(dB) ')
TF=num2str(Tablafinal,'%15.1f');
disp(TF)
disp('*****')
disp('*')
%//////////////////////////////////////

disp('Para ver la graficas finales pulse ENTER')
pause

figure

subplot(2,2,1)
plot(BdeF,ValoresdB1)
hold on
plot(BdeF,ValoresdB1,'*r')
grid
title('NIVEL DE RUIDO DE CH1 TERCIOS DE OCTAVA')
xlabel('Frecuencia (Hz)')
ylabel('Nivel de ruido(dB)')

subplot(2,2,2)
plot(BdeF,ValoresdB1,'r')
hold on
plot(BdeF,ValoresdB1,'*r')
hold on
plot(BdeF,ValoresdB2,'b')
hold on
plot(BdeF,ValoresdB2,'.b')
hold on
grid
title('NIVEL DE RUIDO DE CH1(***) CH2(...) TERCIOS DE OCTAVA')
xlabel('Frecuencia (Hz)')
ylabel('Nivel de ruido(dB)')

subplot(2,2,3)
bar(Aislamiento dBtercios)
grid
title('AISLAMIENTO ENTRE CANALES CHI Y CH2 TERCIOS DE OCTAVA')
xlabel('Numero de banda')
ylabel('R (dB)')

```

```

subplot(2,2,4)
plot(BdeF,AislamientoBtercios)
hold on
plot(BdeF,AislamientoBtercios,'*r')
grid

title('AISLAMIENTO ENTRE CANALES CHI Y CH2 TERCIOS DE OCTAVA')
xlabel('Frecuencia (Hz)')
ylabel('R (dB)')

%*****GRAFICADO FORMAL DE
RESULTADOS*****
disp('PARA VER EL GRAFICADO FORMAL DE RESULTADOS PULSE
ENTER')
pause

figure
semilogx(BdeF,ValoresdB1,'r','LineWidth',1)
hold on
semilogx(BdeF,ValoresdB1,'+r','LineWidth',2)
hold on
semilogx(BdeF,ValoresdB2,'b','LineWidth',1)
hold on
semilogx(BdeF,ValoresdB2,'*b','LineWidth',2)
ylabel('NIVEL DE PRESION SONORA (dB)')
title('NIVELES CAPTURADOS POR AMBOS CANALES,
TRANSMITIDO(+++)rojo INCIDENTE(***)azul')
GraticulaGeneral(NPSymin,NPSymax)

disp('Para ver la tabla final pulse ENTER')
pause

figure
semilogx(BdeF,AislamientoBtercios,'LineWidth',3)
hold on
semilogx(BdeF,AislamientoBtercios,'*r','LineWidth',2)
ylabel('Aislamiento sonoro R (dB)')
title(['SOUND TRANSMISSION CLASS STC = ',num2str(round(10*STC)/10),'
dB. PUNTO "A"'])
GraticulaGeneral(ymin,ymax)

```

CalculoPABCDEF2.m.

```
clc
close all
disp('SE CALCULARA EL AISLAMIENTO ENTRE LAS CAMARAS
CARGANDO LAS ARCHIVOS AislaPA---PF.daq')
disp('Y TAMBIEN LOS ARCHIVOS AislaPA2---PF2.daq CON CON SEGUNDA
POSICION DE FUENTE')
disp('Pulse ENTER para comenzar a calcular los datos')
disp('* * * * *')
pause
S=0;

%*****
%***** INICIA PROCESAMIENTO 1 *****
%*****

%NO SE SI SEA NECESARIO COLOCAR q=1;
for q=1:Notriggers

disp('LLAMANDO DATOS DEL ARCHIVO AislaPA.daq, ESPERE')
Cdisparo=sprintf('PROCESANDO DATOS DEL DISPARO NUMERO %2d',q);
disp(Cdisparo)
disp('*')

[data,time]=daqread('AislaPA.daq','Triggers',[q]);%CREA UN DATA con el
disparo q

data1=data(:,1);%Se crea un vector que tenga solo los datos del CH1
data2=data(:,2);%Se crea un vector que tenga solo los datos del CH2

RMSrfondo1(q) = sqrt(sum(data1.*data1)/size(data1,1));%Se calcula el RMS de
la señal CH1 capturada
RMSrfondo2(q) = sqrt(sum(data2.*data2)/size(data2,1));%Se calcula el RMS de
la señal CH2 capturada

BdeF=[100 125 160 200 250 315 400 500 630 800 1000 1250 1600 2000 2500
3150 4000 5000];
nt=length(BdeF);%Se van a tener 19 tercios de octava de 125Hz hasta 8KHz

for n = 1:nt %Se va a procesar la muestra de disparo q en tercios de octava
f = BdeF(n);
wn = [round((f)*2^(-1/6)) round((f)*2^(1/6))];%Se establecen las
frecuencias de corte
[b,a] = butter(3,wn/fs*2);%filtro fabricado

filtrada1 = filter(b,a,data1);
filtrada2 = filter(b,a,data2);
```

```

        yfilt1(:,n)=filtrada1;%En la columna n, todos los datos (filas) se almacena
la señal filtrada del tercio n evaluado.
        yfilt2(:,n)=filtrada2;%En la columna n, todos los datos (filas) se almacena
la señal filtrada del tercio n evaluado.
        niv1(n) = sqrt(sum(filtrada1.*filtrada1)/size(filtrada1,1));%Se calcula el
RMS de la señal filtrada
        niv2(n) = sqrt(sum(filtrada2.*filtrada2)/size(filtrada2,1));%Se calcula el
RMS de la señal filtrada
    end

    tabladenivacumulada1(:,q)=niv1';%niv' porque debo acumular una columna
de 18 valores
    tabladenivacumulada2(:,q)=niv2';%niv' porque debo acumular una columna
de 18 valores

    TablaacumuladadBprov1(:,q)=20*log10(tabladenivacumulada1(:,q)/Vref1)+dBref1;

    TablaacumuladadBprov2(:,q)=20*log10(tabladenivacumulada2(:,q)/Vref2)+dBref2;
    AislamientoProv=TablaacumuladadBprov2(:,q)-
    TablaacumuladadBprov1(:,q)+NABCDEF';

%   %Preparando la matriz para sacar la STD al final
% ValorSTC=STCf(AislamientoProv);
% TablaAislamientos(S,:)=AislamientoProv';
% TablaSTC(L,1)=ValorSTC;

    TablaRecienCalculado=[BdeF'                               TablaacumuladadBprov1(:,q)
    TablaacumuladadBprov2(:,q) NABCDEF' AislamientoProv];%Se prepara una
tabla con el
    %ultimo calculo de dB realizado hasta el momento

    clc

%////////////////////////////////////
disp('          NIVELES CALCULADOS HASTA EL MOMENTO')
disp('*****')
disp('Frecuencia  NPS(dB)CH1  NPS(dB)CH2    10log(S/A) Aislamiento
(dB) Trigger')
TRC=num2str(TablaRecienCalculado,'%15.1f');
disp(TRC)
disp('*****')
disp('*')
%////////////////////////////////////

end

```

```
%CALCULO DE LOS RMS PROMEDIO POR TERCIOS DE OCTAVA Y EN dB
Identidad=ones(Notriggers,1);
```

```
promediosdeRMS1=(tabladenivacumulada1*Identidad)/Notriggers;
promediosdeRMS2=(tabladenivacumulada2*Identidad)/Notriggers;
```

```
ValoresdB1=20*log10(promediosdeRMS1/Vref1)+dBref1;
ValoresdB2=20*log10(promediosdeRMS2/Vref2)+dBref2;
AislamientoBtercios=ValoresdB2-ValoresdB1+NABCDEF';
%*****
```

```
%CALCULO DEL RMS *GLOBAL* POR TODOS LOS ENSAYOS
```

```
RMSpromedio1=(RMSrfondo1*Identidad)/Notriggers;
RMSpromedio2=(RMSrfondo2*Identidad)/Notriggers;
```

```
RdeFglobaldB1=20*log10(RMSpromedio1/Vref1)+dBref1;
RdeFglobaldB2=20*log10(RMSpromedio2/Vref2)+dBref2;
AislamientoGlobal=RdeFglobaldB2-RdeFglobaldB1;%Calculo del aislamiento
global
```

```
S=S+1;
```

```
PromsRMS1(:,S)=promediosdeRMS1;%Se va acumulando en la columna S el
promedio de RMS por tercios de octava
PromsRMS2(:,S)=promediosdeRMS2;
RMSglob1(S)=RMSpromedio1;%Se va acumulando en la posicion S el
promedio global de RMS
RMSglob2(S)=RMSpromedio2;
```

```
%Preparando la matriz para sacar la STD al final*****
ValorSTC=STCf(AislamientoProv);
TablaAislamientos(S,:)=AislamientoProv';
TablaSTC(S,1)=ValorSTC;
%*****
```

```
disp('*')
disp('*****')
disp('NIVELES YA CALCULADOS PARA ESTE ARCHIVO')
```

```
%*****
%***** TERMINA PROCESAMIENTO 1 *****
%*****
```

```
disp('Esperando 2 segundos para llamar al siguiente archivo, lea su
informacion')
```

```
pause(2)
```

```
clc
```

```
%*****
```



```

%***** INICIA PROCESAMIENTO 2 *****
%*****

%NO SE SI SEA NECESARIO COLOCAR q=1;
for q=1:Notriggers

disp('LLAMANDO DATOS DEL ARCHIVO AislaPB.daq, ESPERE')
Cdisparo=sprintf('PROCESANDO DATOS DEL DISPARO NUMERO %2d',q);
disp(Cdisparo)
disp('*')

[data,time]=daqread('AislaPB.daq','Triggers',[q]);%CREA UN DATA con el
disparo q

data1=data(:,1);%Se crea un vector que tenga solo los datos del CH1
data2=data(:,2);%Se crea un vector que tenga solo los datos del CH2

RMSrfondo1(q) = sqrt(sum(data1.*data1)/size(data1,1));%Se calcula el RMS de
la señal CH1 capturada
RMSrfondo2(q) = sqrt(sum(data2.*data2)/size(data2,1));%Se calcula el RMS de
la señal CH2 capturada

BdeF=[100 125 160 200 250 315 400 500 630 800 1000 1250 1600 2000 2500
3150 4000 5000];
nt=length(BdeF);%Se van a tener 19 tercios de octava de 125Hz hasta 8KHz

for n = 1:nt %Se va a procesar la muestra de disparo q en tercios de octava
f = BdeF(n);
wn = [round((f)*2^(-1/6)) round((f)*2^(1/6))];%Se establecen las
frecuencias de corte
[b,a] = butter(3,wn/fs*2);%filtro fabricado

filtrada1 = filter(b,a,data1);
filtrada2 = filter(b,a,data2);
yfilt1(:,n)=filtrada1;%En la columna n, todos los datos (filas) se almacena
la señal filtrada del tercio n evaluado.
yfilt2(:,n)=filtrada2;%En la columna n, todos los datos (filas) se almacena
la señal filtrada del tercio n evaluado.
niv1(n) = sqrt(sum(filtrada1.*filtrada1)/size(filtrada1,1));%Se calcula el
RMS de la señal filtrada
niv2(n) = sqrt(sum(filtrada2.*filtrada2)/size(filtrada2,1));%Se calcula el
RMS de la señal filtrada
end

tabladenivacumulada1(:,q)=niv1';%niv' porque debo acumular una columna
de 18 valores
tabladenivacumulada2(:,q)=niv2';%niv' porque debo acumular una columna
de 18 valores

```

```
TablaacumuladadBprov1(:,q)=20*log10(tabladenivacumulada1(:,q)/Vref1)+dBref1;
```

```
TablaacumuladadBprov2(:,q)=20*log10(tabladenivacumulada2(:,q)/Vref2)+dBref2;
```

```
AislamientoProv=TablaacumuladadBprov2(:,q)-  
TablaacumuladadBprov1(:,q)+NABCDEF';
```

```
TablaRecienCalculado=[BdeF' TablaacumuladadBprov1(:,q)  
TablaacumuladadBprov2(:,q) NABCDEF' AislamientoProv];%Se prepara una  
tabla con el
```

```
%ultimo calculo de dB realizado hasta el momento
```

```
clc
```

```
%////////////////////////////////////  
disp(' NIVELES CALCULADOS HASTA EL MOMENTO')  
disp('*****')  
disp('Frecuencia NPS(dB)CH1 NPS(dB)CH2 10log(S/A) Aislamiento  
(dB) Trigger')  
TRC=num2str(TablaRecienCalculado,'%15.1f');  
disp(TRC)  
disp('*****')  
disp('*')  
%////////////////////////////////////
```

```
end
```

```
%CALCULO DE LOS RMS PROMEDIO POR TERCIOS DE OCTAVA Y EN dB  
Identidad=ones(Notriggers,1);
```

```
promediosdeRMS1=(tabladenivacumulada1*Identidad)/Notriggers;  
promediosdeRMS2=(tabladenivacumulada2*Identidad)/Notriggers;
```

```
ValoresdB1=20*log10(promediosdeRMS1/Vref1)+dBref1;  
ValoresdB2=20*log10(promediosdeRMS2/Vref2)+dBref2;  
AislamientoBtercios=ValoresdB2-ValoresdB1+NABCDEF';  
%*****
```

```
%CALCULO DEL RMS *GLOBAL* POR TODOS LOS ENSAYOS
```

```
RMSpromedio1=(RMSrfondo1*Identidad)/Notriggers;  
RMSpromedio2=(RMSrfondo2*Identidad)/Notriggers;
```

```
RdeFglobaldB1=20*log10(RMSpromedio1/Vref1)+dBref1;  
RdeFglobaldB2=20*log10(RMSpromedio2/Vref2)+dBref2;  
AislamientoGlobal=RdeFglobaldB2-RdeFglobaldB1;%Calculo del aislamiento  
global
```

```

S=S+1;
PromsRMS1(:,S)=promediosdeRMS1;%Se va acumulando en la columna S el
promedio de RMS por tercios de octava
PromsRMS2(:,S)=promediosdeRMS2;
RMSglob1(S)=RMSpromedio1;%Se va acumulando en la posicion S el
promedio global de RMS
RMSglob2(S)=RMSpromedio2;

```

```

    %Preparando la matriz para sacar la STD al final*****
ValorSTC=STCf(AislamientoProv);
TablaAislamientos(S,:)=AislamientoProv';
TablaSTC(S,1)=ValorSTC;
    %*****

```

```

disp('*')
disp('*****')
disp('NIVELES YA CALCULADOS PARA ESTE ARCHIVO')

```

```

%*****
%***** TERMINA PROCESAMIENTO 2 *****
%*****

```

```

disp('Esperando 2 segundos para llamar al siguiente archivo, lea su
informacion')

```

```

pause(2)

```

```

clc

```

```

%*****
%***** INICIA PROCESAMIENTO 3 *****
%*****

```

```

%NO SE SI SEA NECESARIO COLOCAR q=1;
for q=1:Notriggers

```

```

disp('LLAMANDO DATOS DEL ARCHIVO AislaPC.daq, ESPERE')
Cdisparo=sprintf('PROCESANDO DATOS DEL DISPARO NUMERO %2d',q);
disp(Cdisparo)
disp('*')

```

```

[data,time]=daqread('AislaPC.daq','Triggers',[q]);%CREA UN DATA con el
disparo q

```

```

data1=data(:,1);%Se crea un vector que tenga solo los datos del CH1
data2=data(:,2);%Se crea un vector que tenga solo los datos del CH2

```

```

RMSrfondo1(q) = sqrt(sum(data1.*data1)/size(data1,1));%Se calcula el RMS de
la señal CH1 capturada

```

```
RMSrfondo2(q) = sqrt(sum(data2.*data2)/size(data2,1));%Se calcula el RMS de la señal CH2 capturada
```

```
BdeF=[100 125 160 200 250 315 400 500 630 800 1000 1250 1600 2000 2500 3150 4000 5000];
```

```
nt=length(BdeF);%Se van a tener 19 tercios de octava de 125Hz hasta 8KHz
```

```
for n = 1:nt %Se va a procesar la muestra de disparo q en tercios de octava  
f = BdeF(n);  
wn = [round((f)*2^(-1/6)) round((f)*2^(1/6))];%Se establecen las frecuencias de corte
```

```
[b,a] = butter(3,wn/fs*2);%filtro fabricado
```

```
filtrada1 = filter(b,a,data1);
```

```
filtrada2 = filter(b,a,data2);
```

```
yfilt1(:,n)=filtrada1;%En la columna n, todos los datos (filas) se almacena la señal filtrada del tercio n evaluado.
```

```
yfilt2(:,n)=filtrada2;%En la columna n, todos los datos (filas) se almacena la señal filtrada del tercio n evaluado.
```

```
niv1(n) = sqrt(sum(filtrada1.*filtrada1)/size(filtrada1,1));%Se calcula el RMS de la señal filtrada
```

```
niv2(n) = sqrt(sum(filtrada2.*filtrada2)/size(filtrada2,1));%Se calcula el RMS de la señal filtrada
```

```
end
```

```
tabladenivacumulada1(:,q)=niv1';%niv' porque debo acumular una columna de 18 valores
```

```
tabladenivacumulada2(:,q)=niv2';%niv' porque debo acumular una columna de 18 valores
```

```
TablaacumuladadBprov1(:,q)=20*log10(tabladenivacumulada1(:,q)/Vref1)+dBref1;
```

```
TablaacumuladadBprov2(:,q)=20*log10(tabladenivacumulada2(:,q)/Vref2)+dBref2;
```

```
AislamientoProv=TablaacumuladadBprov2(:,q)-  
TablaacumuladadBprov1(:,q)+NABCDEF';
```

```
TablaRecienCalculado=[BdeF' TablaacumuladadBprov1(:,q)  
TablaacumuladadBprov2(:,q) NABCDEF' AislamientoProv];%Se prepara una tabla con el
```

```
%ultimo calculo de dB realizado hasta el momento
```

```
clc
```

```
%////////////////////////////////////
```

```
disp(' NIVELES CALCULADOS HASTA EL MOMENTO')
```

```
disp('*****')
```

```

disp('Frecuencia   NPS(dB)CH1   NPS(dB)CH2   10log(S/A)  Aislamiento
(dB) Trigger')
TRC=num2str(TablaRecienCalculado,'%15.1f');
disp(TRC)
disp('*****')
disp('*')
%////////////////////////////////////

end

%CALCULO DE LOS RMS PROMEDIO POR TERCIOS DE OCTAVA Y EN dB
Identidad=ones(Notriggers,1);

promediosdeRMS1=(tabladenivacumulada1*Identidad)/Notriggers;
promediosdeRMS2=(tabladenivacumulada2*Identidad)/Notriggers;

ValoresdB1=20*log10(promediosdeRMS1/Vref1)+dBref1;
ValoresdB2=20*log10(promediosdeRMS2/Vref2)+dBref2;
AislamientoBtercios=ValoresdB2-ValoresdB1+NABCDEF';
%*****

%CALCULO DEL RMS *GLOBAL* POR TODOS LOS ENSAYOS
RMSpromedio1=(RMSfondo1*Identidad)/Notriggers;
RMSpromedio2=(RMSfondo2*Identidad)/Notriggers;

RdeFglobaldB1=20*log10(RMSpromedio1/Vref1)+dBref1;
RdeFglobaldB2=20*log10(RMSpromedio2/Vref2)+dBref2;
AislamientoGlobal=RdeFglobaldB2-RdeFglobaldB1;%Calculo del aislamiento
global

S=S+1;
PromsRMS1(:,S)=promediosdeRMS1;%Se va acumulando en la columna S el
promedio de RMS por tercios de octava
PromsRMS2(:,S)=promediosdeRMS2;
RMSglob1(S)=RMSpromedio1;%Se va acumulando en la posicion S el
promedio global de RMS
RMSglob2(S)=RMSpromedio2;

    %Preparando la matriz para sacar la STD al final*****
ValorSTC=STCf(AislamientoProv);
TablaAislamientos(S,:)=AislamientoProv';
TablaSTC(S,1)=ValorSTC;
    %*****

disp('*')
disp('*****')
disp('NIVELES YA CALCULADOS PARA ESTE ARCHIVO')

```

```

%*****
%***** TERMINA PROCESAMIENTO 3 *****
%*****

disp('Esperando 2 segundos para llamar al siguiente archivo, lea su
informacion')
pause(2)
clc
%*****
%***** INICIA PROCESAMIENTO 4 *****
%*****

%NO SE SI SEA NECESARIO COLOCAR q=1;
for q=1:Notriggers

disp('LLAMANDO DATOS DEL ARCHIVO AislaPD.daq, ESPERE')
Cdisparo=sprintf('PROCESANDO DATOS DEL DISPARO NUMERO %2d',q);
disp(Cdisparo)
disp('*')

[data,time]=daqread('AislaPD.daq','Triggers',[q]);%CREA UN DATA con el
disparo q

data1=data(:,1);%Se crea un vector que tenga solo los datos del CH1
data2=data(:,2);%Se crea un vector que tenga solo los datos del CH2

RMSrfondo1(q) = sqrt(sum(data1.*data1)/size(data1,1));%Se calcula el RMS de
la señal CH1 capturada
RMSrfondo2(q) = sqrt(sum(data2.*data2)/size(data2,1));%Se calcula el RMS de
la señal CH2 capturada

BdeF=[100 125 160 200 250 315 400 500 630 800 1000 1250 1600 2000 2500
3150 4000 5000];
nt=length(BdeF);%Se van a tener 19 tercios de octava de 125Hz hasta 8KHz

for n = 1:nt %Se va a procesar la muestra de disparo q en tercios de octava
f = BdeF(n);
wn = [round((f)*2^(-1/6)) round((f)*2^(1/6))];%Se establecen las
frecuencias de corte
[b,a] = butter(3,wn/fs*2);%filtro fabricado

filtrada1 = filter(b,a,data1);
filtrada2 = filter(b,a,data2);
yfilt1(:,n)=filtrada1;%En la columna n, todos los datos (filas) se almacena
la señal filtrada del tercio n evaluado.
yfilt2(:,n)=filtrada2;%En la columna n, todos los datos (filas) se almacena
la señal filtrada del tercio n evaluado.

```

```

    niv1(n) = sqrt(sum(filtrada1.*filtrada1)/size(filtrada1,1));%Se calcula el
RMS de la señal filtrada
    niv2(n) = sqrt(sum(filtrada2.*filtrada2)/size(filtrada2,1));%Se calcula el
RMS de la señal filtrada
end

```

```

    tabladenivacumulada1(:,q)=niv1';%niv' porque debo acumular una columna
de 18 valores
    tabladenivacumulada2(:,q)=niv2';%niv' porque debo acumular una columna
de 18 valores

```

```

TablaacumuladadBprov1(:,q)=20*log10(tabladenivacumulada1(:,q)/Vref1)+dBref1;

```

```

TablaacumuladadBprov2(:,q)=20*log10(tabladenivacumulada2(:,q)/Vref2)+dBref2;

```

```

AislamientoProv=TablaacumuladadBprov2(:,q)-
TablaacumuladadBprov1(:,q)+NABCDEF';

```

```

    TablaRecienCalculado=[BdeF'                               TablaacumuladadBprov1(:,q)
TablaacumuladadBprov2(:,q) NABCDEF' AislamientoProv];%Se prepara una
tabla con el
    %ultimo calculo de dB realizado hasta el momento

```

```

clc

```

```

%////////////////////////////////////
disp('          NIVELES CALCULADOS HASTA EL MOMENTO')
disp('*****')
disp('Frecuencia  NPS(dB)CH1  NPS(dB)CH2    10log(S/A)  Aislamiento
(dB) Trigger')
TRC=num2str(TablaRecienCalculado,'%15.1f');
disp(TRC)
disp('*****')
disp('*')
%////////////////////////////////////

```

```

end

```

```

%CALCULO DE LOS RMS PROMEDIO POR TERCIOS DE OCTAVA Y EN dB
Identidad=ones(Notriggers,1);

```

```

promediosdeRMS1=(tabladenivacumulada1*Identidad)/Notriggers;
promediosdeRMS2=(tabladenivacumulada2*Identidad)/Notriggers;

```

```

ValoresdB1=20*log10(promediosdeRMS1/Vref1)+dBref1;
ValoresdB2=20*log10(promediosdeRMS2/Vref2)+dBref2;

```

```

AislamientoBtercios=ValoresdB2-ValoresdB1+NABCEDEF';
%*****

%CALCULO DEL RMS *GLOBAL* POR TODOS LOS ENSAYOS
RMSpromedio1=(RMSfondo1*Identidad)/Notriggers;
RMSpromedio2=(RMSfondo2*Identidad)/Notriggers;

RdeFglobaldB1=20*log10(RMSpromedio1/Vref1)+dBref1;
RdeFglobaldB2=20*log10(RMSpromedio2/Vref2)+dBref2;
AislamientoGlobal=RdeFglobaldB2-RdeFglobaldB1;%Calculo del aislamiento
global

S=S+1;
PromsRMS1(:,S)=promediosdeRMS1;%Se va acumulando en la columna S el
promedio de RMS por tercios de octava
PromsRMS2(:,S)=promediosdeRMS2;
RMSglob1(S)=RMSpromedio1;%Se va acumulando en la posicion S el
promedio global de RMS
RMSglob2(S)=RMSpromedio2;

    %Preparando la matriz para sacar la STD al final*****
ValorSTC=STCf(AislamientoProv);
TablaAislamientos(S,:)=AislamientoProv';
TablaSTC(S,1)=ValorSTC;
    %*****

disp('*')
disp('*****')
disp('NIVELES YA CALCULADOS PARA ESTE ARCHIVO')

%*****
%***** TERMINA PROCESAMIENTO 4 *****
%*****

disp('Esperando 2 segundos para llamar al siguiente archivo, lea su
informacion')
pause(2)
clc
%*****
%***** INICIA PROCESAMIENTO 5 *****
%*****

%NO SE SI SEA NECESARIO COLOCAR q=1;
for q=1:Notriggers

disp('LLAMANDO DATOS DEL ARCHIVO AislaPE.daq, ESPERE')
Cdisparo=sprintf('PROCESANDO DATOS DEL DISPARO NUMERO %2d',q);

```



```

disp(Cdisparo)
disp('*')

[data,time]=daqread('AislaPE.daq','Triggers',[q]);%CREA UN DATA con el
disparo q

data1=data(:,1);%Se crea un vector que tenga solo los datos del CH1
data2=data(:,2);%Se crea un vector que tenga solo los datos del CH2

RMSrfondo1(q) = sqrt(sum(data1.*data1)/size(data1,1));%Se calcula el RMS de
la señal CH1 capturada
RMSrfondo2(q) = sqrt(sum(data2.*data2)/size(data2,1));%Se calcula el RMS de
la señal CH2 capturada

BdeF=[100 125 160 200 250 315 400 500 630 800 1000 1250 1600 2000 2500
3150 4000 5000];
nt=length(BdeF);%Se van a tener 19 tercios de octava de 125Hz hasta 8KHz

for n = 1:nt %Se va a procesar la muestra de disparo q en tercios de octava
    f = BdeF(n);
    wn = [round((f)*2^(-1/6)) round((f)*2^(1/6))];%Se establecen las
frecuencias de corte
    [b,a] = butter(3,wn/fs*2);%filtro fabricado

    filtrada1 = filter(b,a,data1);
    filtrada2 = filter(b,a,data2);
    yfilt1(:,n)=filtrada1;%En la columna n, todos los datos (filas) se almacena
la señal filtrada del tercio n evaluado.
    yfilt2(:,n)=filtrada2;%En la columna n, todos los datos (filas) se almacena
la señal filtrada del tercio n evaluado.
    niv1(n) = sqrt(sum(filtrada1.*filtrada1)/size(filtrada1,1));%Se calcula el
RMS de la señal filtrada
    niv2(n) = sqrt(sum(filtrada2.*filtrada2)/size(filtrada2,1));%Se calcula el
RMS de la señal filtrada
end

    tabladenivacumulada1(:,q)=niv1';%niv' porque debo acumular una columna
de 18 valores
    tabladenivacumulada2(:,q)=niv2';%niv' porque debo acumular una columna
de 18 valores

TablaacumuladadBprov1(:,q)=20*log10(tabladenivacumulada1(:,q)/Vref1)+dBref1;

TablaacumuladadBprov2(:,q)=20*log10(tabladenivacumulada2(:,q)/Vref2)+dBref2;

AislamientoProv=TablaacumuladadBprov2(:,q)-
TablaacumuladadBprov1(:,q)+NABCDEF';

```

```

    TablaRecienCalculado=[BdeF'
    TablaacumuladadBprov2(:,q) NABCDEF' AislamientoProv];%Se prepara una
    tabla con el
    %ultimo calculo de dB realizado hasta el momento

```

```

    clc

```

```

    %////////////////////////////////////
    disp('          NIVELES CALCULADOS HASTA EL MOMENTO')
    disp('*****')
    disp('Frecuencia  NPS(dB)CH1  NPS(dB)CH2    10log(S/A)  Aislamiento
    (dB) Trigger')
    TRC=num2str(TablaRecienCalculado,'%15.1f');
    disp(TRC)
    disp('*****')
    disp('*')
    %////////////////////////////////////

```

```

    end

    %CALCULO DE LOS RMS PROMEDIO POR TERCIOS DE OCTAVA Y EN dB
    Identidad=ones(Notriggers,1);

```

```

    promediosdeRMS1=(tabladenivacumulada1*Identidad)/Notriggers;
    promediosdeRMS2=(tabladenivacumulada2*Identidad)/Notriggers;

```

```

    ValoresdB1=20*log10(promediosdeRMS1/Vref1)+dBref1;
    ValoresdB2=20*log10(promediosdeRMS2/Vref2)+dBref2;
    AislamientodBtercios=ValoresdB2-ValoresdB1+NABCDEF';
    %*****

```

```

    %CALCULO DEL RMS *GLOBAL* POR TODOS LOS ENSAYOS
    RMSpromedio1=(RMSfondo1*Identidad)/Notriggers;
    RMSpromedio2=(RMSfondo2*Identidad)/Notriggers;

```

```

    RdeFglobaldB1=20*log10(RMSpromedio1/Vref1)+dBref1;
    RdeFglobaldB2=20*log10(RMSpromedio2/Vref2)+dBref2;
    AislamientoGlobal=RdeFglobaldB2-RdeFglobaldB1;%Calculo del aislamiento
    global

```

```

    S=S+1;
    PromsRMS1(:,S)=promediosdeRMS1;%Se va acumulando en la columna S el
    promedio de RMS por tercios de octava
    PromsRMS2(:,S)=promediosdeRMS2;
    RMSglob1(S)=RMSpromedio1;%Se va acumulando en la posicion S el
    promedio global de RMS
    RMSglob2(S)=RMSpromedio2;

```

```

    %Preparando la matriz para sacar la STD al final*****
ValorSTC=STCf(AislamientoProv);
TablaAislamientos(S,:)=AislamientoProv';
TablaSTC(S,1)=ValorSTC;
    %*****

disp('*')
disp('*****')
disp('NIVELES YA CALCULADOS PARA ESTE ARCHIVO')
disp('*')

%*****
%***** TERMINA PROCESAMIENTO 5 *****
%*****
disp('Esperando 2 segundos para llamar al siguiente archivo, lea su
informacion')
pause(2)
clc
%*****
%***** INICIA PROCESAMIENTO 6 *****
%*****

%NO SE SI SEA NECESARIO COLOCAR q=1;
for q=1:Notriggers

disp('LLAMANDO DATOS DEL ARCHIVO AislaPF.daq, ESPERE')
Cdisparo=sprintf('PROCESANDO DATOS DEL DISPARO NUMERO %2d',q);
disp(Cdisparo)
disp('*')

[data,time]=daqread('AislaPF.daq','Triggers',[q]);%CREA UN DATA con el
disparo q

data1=data(:,1);%Se crea un vector que tenga solo los datos del CH1
data2=data(:,2);%Se crea un vector que tenga solo los datos del CH2

RMSrfondo1(q) = sqrt(sum(data1.*data1)/size(data1,1));%Se calcula el RMS de
la señal CH1 capturada
RMSrfondo2(q) = sqrt(sum(data2.*data2)/size(data2,1));%Se calcula el RMS de
la señal CH2 capturada

BdeF=[100 125 160 200 250 315 400 500 630 800 1000 1250 1600 2000 2500
3150 4000 5000];
nt=length(BdeF);%Se van a tener 19 tercios de octava de 125Hz hasta 8KHz

for n = 1:nt %Se va a procesar la muestra de disparo q en tercios de octava
    f = BdeF(n);
    wn = [round((f)*2^(-1/6)) round((f)*2^(1/6))];%Se establecen las
frecuencias de corte

```

```

[b,a] = butter(3,wn/fs*2);%filtro fabricado

filtrada1 = filter(b,a,data1);
filtrada2 = filter(b,a,data2);
yfilt1(:,n)=filtrada1;%En la columna n, todos los datos (filas) se almacena
la señal filtrada del tercio n evaluado.
yfilt2(:,n)=filtrada2;%En la columna n, todos los datos (filas) se almacena
la señal filtrada del tercio n evaluado.
niv1(n) = sqrt(sum(filtrada1.*filtrada1)/size(filtrada1,1));%Se calcula el
RMS de la señal filtrada
niv2(n) = sqrt(sum(filtrada2.*filtrada2)/size(filtrada2,1));%Se calcula el
RMS de la señal filtrada
end

tabladenivacumulada1(:,q)=niv1';%niv' porque debo acumular una columna
de 18 valores
tabladenivacumulada2(:,q)=niv2';%niv' porque debo acumular una columna
de 18 valores

TablaacumuladadBprov1(:,q)=20*log10(tabladenivacumulada1(:,q)/Vref1)+dBref1;

TablaacumuladadBprov2(:,q)=20*log10(tabladenivacumulada2(:,q)/Vref2)+dBref2;
AislamientoProv=TablaacumuladadBprov2(:,q)-
TablaacumuladadBprov1(:,q)+NABCDEF';

TablaRecienCalculado=[BdeF' TablaacumuladadBprov1(:,q)
TablaacumuladadBprov2(:,q) NABCDEF' AislamientoProv];%Se prepara una
tabla con el
%ultimo calculo de dB realizado hasta el momento

clc

%////////////////////////////////////
disp('          NIVELES CALCULADOS HASTA EL MOMENTO')
disp('*****')
disp('Frecuencia  NPS(dB)CH1  NPS(dB)CH2    10log(S/A)  Aislamiento
(dB) Trigger')
TRC=num2str(TablaRecienCalculado,'%15.1f');
disp(TRC)
disp('*****')
disp('*')
%////////////////////////////////////

end

%CALCULO DE LOS RMS PROMEDIO POR TERCIOS DE OCTAVA Y EN dB

```

```

Identidad=ones(Notriggers,1);

promediosdeRMS1=(tabladenivacumulada1*Identidad)/Notriggers;
promediosdeRMS2=(tabladenivacumulada2*Identidad)/Notriggers;

ValoresdB1=20*log10(promediosdeRMS1/Vref1)+dBref1;
ValoresdB2=20*log10(promediosdeRMS2/Vref2)+dBref2;
AislamientoBtercios=ValoresdB2-ValoresdB1+NABCDEF';
%*****

%CALCULO DEL RMS *GLOBAL* POR TODOS LOS ENSAYOS
RMSpromedio1=(RMSfondo1*Identidad)/Notriggers;
RMSpromedio2=(RMSfondo2*Identidad)/Notriggers;

RdeFglobaldB1=20*log10(RMSpromedio1/Vref1)+dBref1;
RdeFglobaldB2=20*log10(RMSpromedio2/Vref2)+dBref2;
AislamientoGlobal=RdeFglobaldB2-RdeFglobaldB1;%Calculo del aislamiento
global

S=S+1;
PromsRMS1(:,S)=promediosdeRMS1;%Se va acumulando en la columna S el
promedio de RMS por tercios de octava
PromsRMS2(:,S)=promediosdeRMS2;
RMSglob1(S)=RMSpromedio1;%Se va acumulando en la posicion S el
promedio global de RMS
RMSglob2(S)=RMSpromedio2;

    %Preparando la matriz para sacar la STD al final*****
ValorSTC=STCf(AislamientoProv);
TablaAislamientos(S,:)=AislamientoProv';
TablaSTC(S,1)=ValorSTC;
    %*****

disp('*')
disp('*****')
disp('NIVELES YA CALCULADOS PARA ESTE ARCHIVO')
disp('*')
disp('***NIVELES CALCULADOS EXITOSAMENTE PARA LOS PRIMEROS
SEIS ARCHIVOS ***')
% disp('***PARA LA PRIMERA POSICION DE FUENTE SONORA')
% disp('PULSE ENTER PARA CALCULAR LOS 6 NIVELES PARA LA
SIGUIENTE POSICION DE FUENTE')

%*****
%***** TERMINA PROCESAMIENTO 6 *****
%*****

```

```

disp('Esperando 2 segundos para llamar al siguiente archivo, lea su
informacion')
pause(2)
clc
%*****
%***** INICIA PROCESAMIENTO 7 *****
%*****

%NO SE SI SEA NECESARIO COLOCAR q=1;
for q=1:Notriggers

disp('LLAMANDO DATOS DEL ARCHIVO AislaPA2.daq, ESPERE')
Cdisparo=sprintf('PROCESANDO DATOS DEL DISPARO NUMERO %2d',q);
disp(Cdisparo)
disp('*')

[data,time]=daqread('AislaPA2.daq','Triggers',[q]);%CREA UN DATA con el
disparo q

data1=data(:,1);%Se crea un vector que tenga solo los datos del CH1
data2=data(:,2);%Se crea un vector que tenga solo los datos del CH2

RMSrfondo1(q) = sqrt(sum(data1.*data1)/size(data1,1));%Se calcula el RMS de
la señal CH1 capturada
RMSrfondo2(q) = sqrt(sum(data2.*data2)/size(data2,1));%Se calcula el RMS de
la señal CH2 capturada

BdeF=[100 125 160 200 250 315 400 500 630 800 1000 1250 1600 2000 2500
3150 4000 5000];
nt=length(BdeF);%Se van a tener 19 tercios de octava de 125Hz hasta 8KHz

for n = 1:nt %Se va a procesar la muestra de disparo q en tercios de octava
f = BdeF(n);
wn = [round((f)*2^(-1/6)) round((f)*2^(1/6))];%Se establecen las
frecuencias de corte
[b,a] = butter(3,wn/fs*2);%filtro fabricado

filtrada1 = filter(b,a,data1);
filtrada2 = filter(b,a,data2);
yfilt1(:,n)=filtrada1;%En la columna n, todos los datos (filas) se almacena
la señal filtrada del tercio n evaluado.
yfilt2(:,n)=filtrada2;%En la columna n, todos los datos (filas) se almacena
la señal filtrada del tercio n evaluado.
niv1(n) = sqrt(sum(filtrada1.*filtrada1)/size(filtrada1,1));%Se calcula el
RMS de la señal filtrada
niv2(n) = sqrt(sum(filtrada2.*filtrada2)/size(filtrada2,1));%Se calcula el
RMS de la señal filtrada
end

```

tabladenivacumulada1(:,q)=niv1';%niv' porque debo acumular una columna de 18 valores

tabladenivacumulada2(:,q)=niv2';%niv' porque debo acumular una columna de 18 valores

TablaacumuladadBprov1(:,q)=20*log10(tabladenivacumulada1(:,q)/Vref1)+dBref1;

TablaacumuladadBprov2(:,q)=20*log10(tabladenivacumulada2(:,q)/Vref2)+dBref2;

AislamientoProv=TablaacumuladadBprov2(:,q)-TablaacumuladadBprov1(:,q)+NABCDEF';

TablaRecienCalculado=[BdeF' TablaacumuladadBprov1(:,q)
TablaacumuladadBprov2(:,q) NABCDEF' AislamientoProv];%Se prepara una tabla con el

%ultimo calculo de dB realizado hasta el momento

clc

```
%////////////////////////////////////  
disp('          NIVELES CALCULADOS HASTA EL MOMENTO')  
disp('*****')  
disp('Frecuencia  NPS(dB)CH1  NPS(dB)CH2    10log(S/A)  Aislamiento  
(dB) Trigger')  
TRC=num2str(TablaRecienCalculado,'%15.1f');  
disp(TRC)  
disp('*****')  
disp('*')  
%////////////////////////////////////
```

end

%CALCULO DE LOS RMS PROMEDIO POR TERCIOS DE OCTAVA Y EN dB
Identidad=ones(Notriggers,1);

promediosdeRMS1=(tabladenivacumulada1*Identidad)/Notriggers;
promediosdeRMS2=(tabladenivacumulada2*Identidad)/Notriggers;

ValoresdB1=20*log10(promediosdeRMS1/Vref1)+dBref1;
ValoresdB2=20*log10(promediosdeRMS2/Vref2)+dBref2;
AislamientoBtercios=ValoresdB2-ValoresdB1+NABCDEF';
%*****

%CALCULO DEL RMS *GLOBAL* POR TODOS LOS ENSAYOS
RMSpromedio1=(RMSfondo1*Identidad)/Notriggers;
RMSpromedio2=(RMSfondo2*Identidad)/Notriggers;

```
RdeFglobaldB1=20*log10(RMSpromedio1/Vref1)+dBref1;
RdeFglobaldB2=20*log10(RMSpromedio2/Vref2)+dBref2;
AislamientoGlobal=RdeFglobaldB2-RdeFglobaldB1;%Calculo del aislamiento
global
```

```
S=S+1;
PromsRMS1(:,S)=promediosdeRMS1;%Se va acumulando en la columna S el
promedio de RMS por tercios de octava
PromsRMS2(:,S)=promediosdeRMS2;
RMSglob1(S)=RMSpromedio1;%Se va acumulando en la posicion S el
promedio global de RMS
RMSglob2(S)=RMSpromedio2;
```

```
%Preparando la matriz para sacar la STD al final*****
ValorSTC=STCf(AislamientoProv);
TablaAislamientos(S,:)=AislamientoProv';
TablaSTC(S,1)=ValorSTC;
%*****
```

```
disp('*')
disp('*****')
disp('NIVELES YA CALCULADOS PARA ESTE ARCHIVO')
```

```
%*****
%***** TERMINA PROCESAMIENTO 7 *****
%*****
```

```
disp('Esperando 2 segundos para llamar al siguiente archivo, lea su
informacion')
```

```
pause(2)
clc
%*****
%***** INICIA PROCESAMIENTO 8 *****
%*****
```

```
%NO SE SI SEA NECESARIO COLOCAR q=1;
for q=1:Notriggers
```

```
disp('LLAMANDO DATOS DEL ARCHIVO AislaPB2.daq, ESPERE')
Cdisparo=sprintf('PROCESANDO DATOS DEL DISPARO NUMERO %2d',q);
disp(Cdisparo)
disp('*')
```

```
[data,time]=daqread('AislaPB2.daq','Triggers',[q]);%CREA UN DATA con el
disparo q
```



```

data1=data(:,1);%Se crea un vector que tenga solo los datos del CH1
data2=data(:,2);%Se crea un vector que tenga solo los datos del CH2

RMSrfondo1(q) = sqrt(sum(data1.*data1)/size(data1,1));%Se calcula el RMS de
la señal CH1 capturada
RMSrfondo2(q) = sqrt(sum(data2.*data2)/size(data2,1));%Se calcula el RMS de
la señal CH2 capturada

BdeF=[100 125 160 200 250 315 400 500 630 800 1000 1250 1600 2000 2500
3150 4000 5000];
nt=length(BdeF);%Se van a tener 19 tercios de octava de 125Hz hasta 8KHz

for n = 1:nt %Se va a procesar la muestra de disparo q en tercios de octava
    f = BdeF(n);
    wn = [round((f)*2^(-1/6)) round((f)*2^(1/6))];%Se establecen las
frecuencias de corte
    [b,a] = butter(3,wn/fs*2);%filtro fabricado

    filtrada1 = filter(b,a,data1);
    filtrada2 = filter(b,a,data2);
    yfilt1(:,n)=filtrada1;%En la columna n, todos los datos (filas) se almacena
la señal filtrada del tercio n evaluado.
    yfilt2(:,n)=filtrada2;%En la columna n, todos los datos (filas) se almacena
la señal filtrada del tercio n evaluado.
    niv1(n) = sqrt(sum(filtrada1.*filtrada1)/size(filtrada1,1));%Se calcula el
RMS de la señal filtrada
    niv2(n) = sqrt(sum(filtrada2.*filtrada2)/size(filtrada2,1));%Se calcula el
RMS de la señal filtrada
end

    tabladenivacumulada1(:,q)=niv1';%niv' porque debo acumular una columna
de 18 valores
    tabladenivacumulada2(:,q)=niv2';%niv' porque debo acumular una columna
de 18 valores

TablaacumuladadBprov1(:,q)=20*log10(tabladenivacumulada1(:,q)/Vref1)+dBref1;

TablaacumuladadBprov2(:,q)=20*log10(tabladenivacumulada2(:,q)/Vref2)+dBref2;
AislamientoProv=TablaacumuladadBprov2(:,q)-
TablaacumuladadBprov1(:,q)+NABCDEF';

    TablaRecienCalculado=[BdeF' TablaacumuladadBprov1(:,q)
TablaacumuladadBprov2(:,q) NABCDEF' AislamientoProv];%Se prepara una
tabla con el
    %ultimo calculo de dB realizado hasta el momento

clc

```

```

%////////////////////////////////////
disp('          NIVELES CALCULADOS HASTA EL MOMENTO')
disp('*****')
disp('Frecuencia  NPS(dB)CH1  NPS(dB)CH2    10log(S/A)  Aislamiento
(dB) Trigger')
TRC=num2str(TablaRecienCalculado,'%15.1f');
disp(TRC)
disp('*****')
disp('*')
%////////////////////////////////////

```

```
end
```

```
%CALCULO DE LOS RMS PROMEDIO POR TERCIOS DE OCTAVA Y EN dB
Identidad=ones(Notriggers,1);
```

```
promediosdeRMS1=(tabladenivacumulada1*Identidad)/Notriggers;
promediosdeRMS2=(tabladenivacumulada2*Identidad)/Notriggers;
```

```
ValoresdB1=20*log10(promediosdeRMS1/Vref1)+dBref1;
ValoresdB2=20*log10(promediosdeRMS2/Vref2)+dBref2;
AislamientoBtercios=ValoresdB2-ValoresdB1+NABCDEF';
%*****
```

```
%CALCULO DEL RMS *GLOBAL* POR TODOS LOS ENSAYOS
```

```
RMSpromedio1=(RMSrfondo1*Identidad)/Notriggers;
RMSpromedio2=(RMSrfondo2*Identidad)/Notriggers;
```

```
RdeFglobaldB1=20*log10(RMSpromedio1/Vref1)+dBref1;
RdeFglobaldB2=20*log10(RMSpromedio2/Vref2)+dBref2;
AislamientoGlobal=RdeFglobaldB2-RdeFglobaldB1;%Calculo del aislamiento
global
```

```
S=S+1;
```

```
PromsRMS1(:,S)=promediosdeRMS1;%Se va acumulando en la columna S el
promedio de RMS por tercios de octava
```

```
PromsRMS2(:,S)=promediosdeRMS2;
```

```
RMSglob1(S)=RMSpromedio1;%Se va acumulando en la posicion S el
promedio global de RMS
```

```
RMSglob2(S)=RMSpromedio2;
```

```
%Preparando la matriz para sacar la STD al final*****
```

```
ValorSTC=STCf(AislamientoProv);
```

```
TablaAislamientos(S,:)=AislamientoProv';
```

```
TablaSTC(S,1)=ValorSTC;
```

```
%*****
```

```

disp('*')
disp('*****')
disp('NIVELES YA CALCULADOS PARA ESTE ARCHIVO')

%*****
%***** TERMINA PROCESAMIENTO 8 *****
%*****

disp('Esperando 2 segundos para llamar al siguiente archivo, lea su
informacion')
pause(2)
clc
%*****
%***** INICIA PROCESAMIENTO 9 *****
%*****

%NO SE SI SEA NECESARIO COLOCAR q=1;
for q=1:Notriggers

disp('LLAMANDO DATOS DEL ARCHIVO AislaPC2.daq, ESPERE')
Cdisparo=sprintf('PROCESANDO DATOS DEL DISPARO NUMERO %2d',q);
disp(Cdisparo)
disp('*')

[data,time]=daqread('AislaPC2.daq','Triggers',[q]);%CREA UN DATA con el
disparo q

data1=data(:,1);%Se crea un vector que tenga solo los datos del CH1
data2=data(:,2);%Se crea un vector que tenga solo los datos del CH2

RMSrfondo1(q) = sqrt(sum(data1.*data1)/size(data1,1));%Se calcula el RMS de
la señal CH1 capturada
RMSrfondo2(q) = sqrt(sum(data2.*data2)/size(data2,1));%Se calcula el RMS de
la señal CH2 capturada

BdeF=[100 125 160 200 250 315 400 500 630 800 1000 1250 1600 2000 2500
3150 4000 5000];
nt=length(BdeF);%Se van a tener 19 tercios de octava de 125Hz hasta 8KHz

for n = 1:nt %Se va a procesar la muestra de disparo q en tercios de octava
f = BdeF(n);
wn = [round((f)*2^(-1/6)) round((f)*2^(1/6))];%Se establecen las
frecuencias de corte
[b,a] = butter(3,wn/fs*2);%filtro fabricado

filtrada1 = filter(b,a,data1);
filtrada2 = filter(b,a,data2);

```

```

        yfilt1(:,n)=filtrada1;%En la columna n, todos los datos (filas) se almacena
la señal filtrada del tercio n evaluado.
        yfilt2(:,n)=filtrada2;%En la columna n, todos los datos (filas) se almacena
la señal filtrada del tercio n evaluado.
        niv1(n) = sqrt(sum(filtrada1.*filtrada1)/size(filtrada1,1));%Se calcula el
RMS de la señal filtrada
        niv2(n) = sqrt(sum(filtrada2.*filtrada2)/size(filtrada2,1));%Se calcula el
RMS de la señal filtrada
    end

    tabladenivacumulada1(:,q)=niv1';%niv' porque debo acumular una columna
de 18 valores
    tabladenivacumulada2(:,q)=niv2';%niv' porque debo acumular una columna
de 18 valores

    TablaacumuladadBprov1(:,q)=20*log10(tabladenivacumulada1(:,q)/Vref1)+dBref1;

    TablaacumuladadBprov2(:,q)=20*log10(tabladenivacumulada2(:,q)/Vref2)+dBref2;
    AislamientoProv=TablaacumuladadBprov2(:,q)-
    TablaacumuladadBprov1(:,q)+NABCDEF';

    TablaRecienCalculado=[BdeF'                TablaacumuladadBprov1(:,q)
    TablaacumuladadBprov2(:,q) NABCDEF' AislamientoProv];%Se prepara una
tabla con el
    %ultimo calculo de dB realizado hasta el momento

    clc

    %////////////////////////////////////
    disp('                NIVELES CALCULADOS HASTA EL MOMENTO')
    disp('*****')
    disp('Frecuencia  NPS(dB)CH1  NPS(dB)CH2    10log(S/A)  Aislamiento
(dB) Trigger')
    TRC=num2str(TablaRecienCalculado,'%15.1f');
    disp(TRC)
    disp('*****')
    disp('*')
    %////////////////////////////////////

end

%CALCULO DE LOS RMS PROMEDIO POR TERCIOS DE OCTAVA Y EN dB
Identidad=ones(Notriggers,1);

promediosdeRMS1=(tabladenivacumulada1*Identidad)/Notriggers;
promediosdeRMS2=(tabladenivacumulada2*Identidad)/Notriggers;

```

```

ValoresdB1=20*log10(promediosdeRMS1/Vref1)+dBref1;
ValoresdB2=20*log10(promediosdeRMS2/Vref2)+dBref2;
AislamientoBtercios=ValoresdB2-ValoresdB1+NABCDEF';
%*****

%CALCULO DEL RMS *GLOBAL* POR TODOS LOS ENSAYOS
RMSpromedio1=(RMSrfondo1*Identidad)/Notriggers;
RMSpromedio2=(RMSrfondo2*Identidad)/Notriggers;

RdeFglobaldB1=20*log10(RMSpromedio1/Vref1)+dBref1;
RdeFglobaldB2=20*log10(RMSpromedio2/Vref2)+dBref2;
AislamientoGlobal=RdeFglobaldB2-RdeFglobaldB1;%Calculo del aislamiento
global

S=S+1;
PromsRMS1(:,S)=promediosdeRMS1;%Se va acumulando en la columna S el
promedio de RMS por tercios de octava
PromsRMS2(:,S)=promediosdeRMS2;
RMSglob1(S)=RMSpromedio1;%Se va acumulando en la posicion S el
promedio global de RMS
RMSglob2(S)=RMSpromedio2;

    %Preparando la matriz para sacar la STD al final*****
ValorSTC=STCf(AislamientoProv);
TablaAislamientos(S,:)=AislamientoProv';
TablaSTC(S,1)=ValorSTC;
    %*****

disp('*')
disp('*****')
disp('NIVELES YA CALCULADOS PARA ESTE ARCHIVO')

%*****
%***** TERMINA PROCESAMIENTO 9 *****
%*****

disp('Esperando 2 segundos para llamar al siguiente archivo, lea su
informacion')
pause(2)
clc
%*****
%***** INICIA PROCESAMIENTO 10 *****
%*****

%NO SE SI SEA NECESARIO COLOCAR q=1;

```

```

for q=1:Notriggers

disp('LLAMANDO DATOS DEL ARCHIVO AislaPD2.daq, ESPERE')
Cdisparo=sprintf('PROCESANDO DATOS DEL DISPARO NUMERO %2d',q);
disp(Cdisparo)
disp('*')

[data,time]=daqread('AislaPD2.daq','Triggers',[q]);%CREA UN DATA con el
disparo q

data1=data(:,1);%Se crea un vector que tenga solo los datos del CH1
data2=data(:,2);%Se crea un vector que tenga solo los datos del CH2

RMSrfondo1(q) = sqrt(sum(data1.*data1)/size(data1,1));%Se calcula el RMS de
la señal CH1 capturada
RMSrfondo2(q) = sqrt(sum(data2.*data2)/size(data2,1));%Se calcula el RMS de
la señal CH2 capturada

BdeF=[100 125 160 200 250 315 400 500 630 800 1000 1250 1600 2000 2500
3150 4000 5000];
nt=length(BdeF);%Se van a tener 19 tercios de octava de 125Hz hasta 8KHz

for n = 1:nt %Se va a procesar la muestra de disparo q en tercios de octava
f = BdeF(n);
wn = [round((f)*2^(-1/6)) round((f)*2^(1/6))];%Se establecen las
frecuencias de corte
[b,a] = butter(3,wn/fs*2);%filtro fabricado

filtrada1 = filter(b,a,data1);
filtrada2 = filter(b,a,data2);
yf1(:,n)=filtrada1;%En la columna n, todos los datos (filas) se almacena
la señal filtrada del tercio n evaluado.
yf2(:,n)=filtrada2;%En la columna n, todos los datos (filas) se almacena
la señal filtrada del tercio n evaluado.
niv1(n) = sqrt(sum(filtrada1.*filtrada1)/size(filtrada1,1));%Se calcula el
RMS de la señal filtrada
niv2(n) = sqrt(sum(filtrada2.*filtrada2)/size(filtrada2,1));%Se calcula el
RMS de la señal filtrada
end

tabladenivacumulada1(:,q)=niv1';%niv' porque debo acumular una columna
de 18 valores
tabladenivacumulada2(:,q)=niv2';%niv' porque debo acumular una columna
de 18 valores

TablaacumuladadBprov1(:,q)=20*log10(tabladenivacumulada1(:,q)/Vref1)+dBref1;

```

```

TablaacumuladadBprov2(:,q)=20*log10(tabladenivacumulada2(:,q)/Vref2)+dBref2;
AislamientoProv=TablaacumuladadBprov2(:,q)-
TablaacumuladadBprov1(:,q)+NABCDEF';

TablaRecienCalculado=[BdeF' TablaacumuladadBprov1(:,q)
TablaacumuladadBprov2(:,q) NABCDEF' AislamientoProv];%Se prepara una
tabla con el
%ultimo calculo de dB realizado hasta el momento

clc

%////////////////////////////////////
disp(' NIVELES CALCULADOS HASTA EL MOMENTO')
disp('*****')
disp('Frecuencia NPS(dB)CH1 NPS(dB)CH2 10log(S/A) Aislamiento
(dB) Trigger')
TRC=num2str(TablaRecienCalculado,'%15.1f');
disp(TRC)
disp('*****')
disp('*')
%////////////////////////////////////

end

%CALCULO DE LOS RMS PROMEDIO POR TERCIOS DE OCTAVA Y EN dB
Identidad=ones(Notriggers,1);

promediosdeRMS1=(tabladenivacumulada1*Identidad)/Notriggers;
promediosdeRMS2=(tabladenivacumulada2*Identidad)/Notriggers;

ValoresdB1=20*log10(promediosdeRMS1/Vref1)+dBref1;
ValoresdB2=20*log10(promediosdeRMS2/Vref2)+dBref2;
AislamientoBtercios=ValoresdB2-ValoresdB1+NABCDEF';
%*****

%CALCULO DEL RMS *GLOBAL* POR TODOS LOS ENSAYOS
RMSpromedio1=(RMSfondo1*Identidad)/Notriggers;
RMSpromedio2=(RMSfondo2*Identidad)/Notriggers;

RdeFglobaldB1=20*log10(RMSpromedio1/Vref1)+dBref1;
RdeFglobaldB2=20*log10(RMSpromedio2/Vref2)+dBref2;
AislamientoGlobal=RdeFglobaldB2-RdeFglobaldB1;%Calculo del aislamiento
global

S=S+1;

```

```
PromsRMS1(:,S)=promediosdeRMS1;%Se va acumulando en la columna S el
promedio de RMS por tercios de octava
PromsRMS2(:,S)=promediosdeRMS2;
RMSglob1(S)=RMSpromedio1;%Se va acumulando en la posicion S el
promedio global de RMS
RMSglob2(S)=RMSpromedio2;
```

```
%Preparando la matriz para sacar la STD al final*****
ValorSTC=STCf(AislamientoProv);
TablaAislamientos(S,:)=AislamientoProv';
TablaSTC(S,1)=ValorSTC;
%*****
```

```
disp('*')
disp('*****')
disp('NIVELES YA CALCULADOS PARA ESTE ARCHIVO')
```

```
%*****
%***** TERMINA PROCESAMIENTO 10 *****
%*****
```

```
disp('Esperando 2 segundos para llamar al siguiente archivo, lea su
informacion')
pause(2)
clc
%*****
%***** INICIA PROCESAMIENTO 11 *****
%*****
```

```
%NO SE SI SEA NECESARIO COLOCAR q=1;
for q=1:Notriggers
```

```
disp('LLAMANDO DATOS DEL ARCHIVO AislaPE2.daq, ESPERE')
Cdisparo=sprintf('PROCESANDO DATOS DEL DISPARO NUMERO %2d',q);
disp(Cdisparo)
disp('*')
```

```
[data,time]=daqread('AislaPE2.daq','Triggers',[q]);%CREA UN DATA con el
disparo q
```

```
data1=data(:,1);%Se crea un vector que tenga solo los datos del CH1
data2=data(:,2);%Se crea un vector que tenga solo los datos del CH2
```

```
RMSrfondo1(q) = sqrt(sum(data1.*data1)/size(data1,1));%Se calcula el RMS de
la señal CH1 capturada
RMSrfondo2(q) = sqrt(sum(data2.*data2)/size(data2,1));%Se calcula el RMS de
la señal CH2 capturada
```



```

BdeF=[100 125 160 200 250 315 400 500 630 800 1000 1250 1600 2000 2500
3150 4000 5000];
nt=length(BdeF);%Se van a tener 19 tercios de octava de 125Hz hasta 8KHz

for n = 1:nt %Se va a procesar la muestra de disparo q en tercios de octava
    f = BdeF(n);
    wn = [round((f)*2^(-1/6)) round((f)*2^(1/6))];%Se establecen las
frecuencias de corte
    [b,a] = butter(3,wn/fs*2);%filtro fabricado

    filtrada1 = filter(b,a,data1);
    filtrada2 = filter(b,a,data2);
    yfilt1(:,n)=filtrada1;%En la columna n, todos los datos (filas) se almacena
la señal filtrada del tercio n evaluado.
    yfilt2(:,n)=filtrada2;%En la columna n, todos los datos (filas) se almacena
la señal filtrada del tercio n evaluado.
    niv1(n) = sqrt(sum(filtrada1.*filtrada1)/size(filtrada1,1));%Se calcula el
RMS de la señal filtrada
    niv2(n) = sqrt(sum(filtrada2.*filtrada2)/size(filtrada2,1));%Se calcula el
RMS de la señal filtrada
end

    tabladenivacumulada1(:,q)=niv1';%niv' porque debo acumular una columna
de 18 valores
    tabladenivacumulada2(:,q)=niv2';%niv' porque debo acumular una columna
de 18 valores

TablaacumuladadBprov1(:,q)=20*log10(tabladenivacumulada1(:,q)/Vref1)+dBref1;

TablaacumuladadBprov2(:,q)=20*log10(tabladenivacumulada2(:,q)/Vref2)+dBref2;
AislamientoProv=TablaacumuladadBprov2(:,q)-
TablaacumuladadBprov1(:,q)+NABCDEF';

    TablaRecienCalculado=[BdeF' TablaacumuladadBprov1(:,q)
TablaacumuladadBprov2(:,q) NABCDEF' AislamientoProv];%Se prepara una
tabla con el
    %ultimo calculo de dB realizado hasta el momento

    clc

%////////////////////////////////////
disp('          NIVELES CALCULADOS HASTA EL MOMENTO')
disp('*****')
disp('Frecuencia  NPS(dB)CH1  NPS(dB)CH2  10log(S/A) Aislamiento
(dB) Trigger')
TRC=num2str(TablaRecienCalculado,'%15.1f');

```

```

disp(TRC)
disp('*****')
disp('*')
%//////////////////////////////////////////////////////////////////

end

%CALCULO DE LOS RMS PROMEDIO POR TERCIOS DE OCTAVA Y EN dB
Identidad=ones(Notriggers,1);

promediosdeRMS1=(tabladenivacumulada1*Identidad)/Notriggers;
promediosdeRMS2=(tabladenivacumulada2*Identidad)/Notriggers;

ValoresdB1=20*log10(promediosdeRMS1/Vref1)+dBref1;
ValoresdB2=20*log10(promediosdeRMS2/Vref2)+dBref2;
AislamientoBtercios=ValoresdB2-ValoresdB1+NABCDEF';
%*****

%CALCULO DEL RMS *GLOBAL* POR TODOS LOS ENSAYOS
RMSpromedio1=(RMSrfondo1*Identidad)/Notriggers;
RMSpromedio2=(RMSrfondo2*Identidad)/Notriggers;

RdeFglobaldB1=20*log10(RMSpromedio1/Vref1)+dBref1;
RdeFglobaldB2=20*log10(RMSpromedio2/Vref2)+dBref2;
AislamientoGlobal=RdeFglobaldB2-RdeFglobaldB1;%Calculo del aislamiento
global

S=S+1;
PromsRMS1(:,S)=promediosdeRMS1;%Se va acumulando en la columna S el
promedio de RMS por tercios de octava
PromsRMS2(:,S)=promediosdeRMS2;
RMSglob1(S)=RMSpromedio1;%Se va acumulando en la posicion S el
promedio global de RMS
RMSglob2(S)=RMSpromedio2;

    %Preparando la matriz para sacar la STD al final*****
ValorSTC=STCf(AislamientoProv);
TablaAislamientos(S,:)=AislamientoProv';
TablaSTC(S,1)=ValorSTC;
    %*****

disp('*')
disp('*****')
disp('NIVELES YA CALCULADOS PARA ESTE ARCHIVO')

```

```

%*****
%***** TERMINA PROCESAMIENTO 11 *****
%*****

disp('Esperando 2 segundos para llamar al siguiente archivo, lea su
informacion')
pause(2)
clc
%*****
%***** INICIA PROCESAMIENTO 12 *****
%*****

%NO SE SI SEA NECESARIO COLOCAR q=1;
for q=1:Notriggers

disp('LLAMANDO DATOS DEL ARCHIVO AislaPF2.daq, ESPERE')
Cdisparo=sprintf('PROCESANDO DATOS DEL DISPARO NUMERO %2d',q);
disp(Cdisparo)
disp('*')

[data,time]=daqread('AislaPF2.daq','Triggers',[q]);%CREA UN DATA con el
disparo q

data1=data(:,1);%Se crea un vector que tenga solo los datos del CH1
data2=data(:,2);%Se crea un vector que tenga solo los datos del CH2

RMSrfondo1(q) = sqrt(sum(data1.*data1)/size(data1,1));%Se calcula el RMS de
la señal CH1 capturada
RMSrfondo2(q) = sqrt(sum(data2.*data2)/size(data2,1));%Se calcula el RMS de
la señal CH2 capturada

BdeF=[100 125 160 200 250 315 400 500 630 800 1000 1250 1600 2000 2500
3150 4000 5000];
nt=length(BdeF);%Se van a tener 19 tercios de octava de 125Hz hasta 8KHz

for n = 1:nt %Se va a procesar la muestra de disparo q en tercios de octava
f = BdeF(n);
wn = [round((f)*2^(-1/6)) round((f)*2^(1/6))];%Se establecen las
frecuencias de corte
[b,a] = butter(3,wn/fs*2);%filtro fabricado

filtrada1 = filter(b,a,data1);
filtrada2 = filter(b,a,data2);
yfilt1(:,n)=filtrada1;%En la columna n, todos los datos (filas) se almacena
la señal filtrada del tercio n evaluado.
yfilt2(:,n)=filtrada2;%En la columna n, todos los datos (filas) se almacena
la señal filtrada del tercio n evaluado.
niv1(n) = sqrt(sum(filtrada1.*filtrada1)/size(filtrada1,1));%Se calcula el
RMS de la señal filtrada

```

```

    niv2(n) = sqrt(sum(filtrada2.*filtrada2)/size(filtrada2,1));%Se calcula el
    RMS de la señal filtrada
end

```

```

    tabladenivacumulada1(:,q)=niv1';%niv' porque debo acumular una columna
    de 18 valores
    tabladenivacumulada2(:,q)=niv2';%niv' porque debo acumular una columna
    de 18 valores

```

```

TablaacumuladadBprov1(:,q)=20*log10(tabladenivacumulada1(:,q)/Vref1)+dBref1;

```

```

TablaacumuladadBprov2(:,q)=20*log10(tabladenivacumulada2(:,q)/Vref2)+dBref2;

```

```

    AislamientoProv=TablaacumuladadBprov2(:,q)-
    TablaacumuladadBprov1(:,q)+NABCDEF';

```

```

    TablaRecienCalculado=[BdeF'                               TablaacumuladadBprov1(:,q)
    TablaacumuladadBprov2(:,q) NABCDEF' AislamientoProv];%Se prepara una
    tabla con el
    %ultimo calculo de dB realizado hasta el momento

```

```

    clc

```

```

%////////////////////////////////////
disp('          NIVELES CALCULADOS HASTA EL MOMENTO')
disp('*****')
disp('Frecuencia  NPS(dB)CH1  NPS(dB)CH2    10log(S/A) Aislamiento
(dB) Trigger')
TRC=num2str(TablaRecienCalculado,'%15.1f');
disp(TRC)
disp('*****')
disp('*')
%////////////////////////////////////

```

```

end

```

```

%CALCULO DE LOS RMS PROMEDIO POR TERCIOS DE OCTAVA Y EN dB
Identidad=ones(Notriggers,1);

```

```

promediosdeRMS1=(tabladenivacumulada1*Identidad)/Notriggers;
promediosdeRMS2=(tabladenivacumulada2*Identidad)/Notriggers;

```

```

ValoresdB1=20*log10(promediosdeRMS1/Vref1)+dBref1;
ValoresdB2=20*log10(promediosdeRMS2/Vref2)+dBref2;
AislamientodBtercios=ValoresdB2-ValoresdB1+NABCDEF';
%*****

```

```

%CALCULO DEL RMS *GLOBAL* POR TODOS LOS ENSAYOS
RMSpromedio1=(RMSrfondo1*Identidad)/Notrigger;
RMSpromedio2=(RMSrfondo2*Identidad)/Notrigger;

RdeFglobaldB1=20*log10(RMSpromedio1/Vref1)+dBref1;
RdeFglobaldB2=20*log10(RMSpromedio2/Vref2)+dBref2;
AislamientoGlobal=RdeFglobaldB2-RdeFglobaldB1;%Calculo del aislamiento
global

S=S+1;
PromsRMS1(:,S)=promediosdeRMS1;%Se va acumulando en la columna S el
promedio de RMS por tercios de octava
PromsRMS2(:,S)=promediosdeRMS2;
RMSglob1(S)=RMSpromedio1;%Se va acumulando en la posicion S el
promedio global de RMS
RMSglob2(S)=RMSpromedio2;

    %Preparando la matriz para sacar la STD al final*****
ValorSTC=STCf(AislamientoProv);
TablaAislamientos(S,:)=AislamientoProv';
TablaSTC(S,1)=ValorSTC;
    %*****

disp('*')
disp('*****')
disp('NIVELES YA CALCULADOS PARA ESTE ARCHIVO')

%*****
%***** TERMINA PROCESAMIENTO 12 *****
%*****
disp('*')
disp('*****')
disp('***NIVELES CALCULADOS EXITOSAMENTE PARA LOS DOCE
ARCHIVOS***')
disp('*****')
disp('PULSE ENTER PARA VER LOS RESULTADOS FINALES')
pause
clc

%////////////////////////////////////
%En esta seccion obtendremos la DESVIACION STANDART de debido a cada
uno
%de los ensayos
STDAislamientosNPS=std(TablaAislamientos);

```

```
%////////////////////////////////////
```

```
%CALCULO DE LOS RMS PROMEDIO FINALES CONSIDERANDO TODOS
LOS ARCHIVOS POR TERCIOS DE OCTAVA Y EN dB
```

```
Identidad=ones(S,1);
```

```
promediosdeRMS1todo=(PromsRMS1*Identidad)/S;%Promedia todas la
columnas
```

```
promediosdeRMS2todo=(PromsRMS2*Identidad)/S;
```

```
ValoresdB1todo=20*log10(promediosdeRMS1todo/Vref1)+dBref1;%%Evalua
en dB los promedios de RMS
```

```
ValoresdB2todo=20*log10(promediosdeRMS2todo/Vref2)+dBref2;
```

```
AislamientoBterciostodo=ValoresdB2todo-
```

```
ValoresdB1todo+NABCEFF';%Calcula aislamiento en dB cosiderando la
influencia de la muestra
```

```
%*****
```

```
%////////////////////////////////////
```

```
%/////////////////////////////////Tabla de valores en dB de todos los ensayos////////
```

```
PromsRMS1dB=20*log10(PromsRMS1/Vref1)+dBref1;
```

```
PromsRMS2dB=20*log10(PromsRMS2/Vref2)+dBref2;
```

```
STDNPS1=std(PromsRMS1dB');
```

```
STDNPS2=std(PromsRMS2dB');
```

```
%////////////////////////////////////
```

```
*/**/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*
```

```
*/**/*/*CALCULO DE NIVELES GLOBALES POR CANAL */**/*/*/*
```

```
*/**/*/*/*/*/*EN LAS BANDAS DE ESTUDIO*/**/*/*/*/*/*/*
```

```
dB1005KCH1=SumadBf(ValoresdB1todo);
```

```
dB1005KCH2=SumadBf(ValoresdB2todo);
```

```
Aisla1005K=dB1005KCH2-dB1005KCH1;%Contribucion por el material de
manera global
```

```
STC=STCf(AislamientoBterciostodo);%Se calcula el STC medido en ese
punto
```

```
*/**/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*
```

```
%CALCULO DEL RMS *GLOBAL* POR TODOS LOS ENSAYOS
```

```
RMSpromedio1global=sum(RMSglob1)/S;%Promedia los RMS globales
```

```
RMSpromedio2global=sum(RMSglob2)/S;
```

```
RdeFdB1global=20*log10(RMSpromedio1global/Vref1)+dBref1;%Evalua en dB
los RMS globales
```

```
RdeFdB2global=20*log10(RMSpromedio2global/Vref2)+dBref2;
```

```
AislamientoBGlobal=RdeFdB2global-RdeFdB1global;%Calculo del
aislamiento global en dB
```

```

TabladBfinal=[BdeF' ValoresdB1todo STDNPS1' ValoresdB2todo STDNPS2'
NABCDEF' AislamientoBterciostodo STDAislamientosNPS']; %tABLA DE
PAREJAS BANDA-NIVEL
TabladBfinalResumen=[BdeF' ValoresdB1todo ValoresdB2todo NABCDEF'
AislamientoBterciostodo]; %tABLA DE PAREJAS BANDA-NIVEL

```

```

disp('*')
disp('*****')
NPSdBCH1=sprintf('NPS Global Final en CH1 = %3.1f dB.',RdeFdB1global);
disp(NPSdBCH1)
NPSdBCH2=sprintf('NPS Global Final en CH2 = %3.1f dB.',RdeFdB2global);
disp(NPSdBCH2)
disp('*****')

```

```

AdBG=sprintf('Aislamiento Global Final = %3.1f dB.',AislamientoBGlobal);
disp(AdBG)
disp('*****')

```

```

%*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/
disp('*')
disp('*****')
dBCH1=sprintf('NPS Banda 100-5KHz CH1 = %3.1f dB.',dB1005KCH1);
disp(dBCH1)
dBCH2=sprintf('NPS Banda 100-5KHz CH2 = %3.1f dB.',dB1005KCH2);
disp(dBCH2)
disp('*****')

```

```

AdB=sprintf('Aislamiento Global Banda 100-5KHz= %3.1f dB.',Aisla1005K);
disp(AdB)
disp('*****')
STCtexto=sprintf('STC Banda 100-5KHz= %3.1f dB.',STC);
disp(STCtexto)
disp('*****')
%*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/
%*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/

```

```

%////////////////////////////////////
disp('*')
disp('          TABLA DE RESULTADOS  PANEL "D"')
disp('*****')
disp('Frecuencia  NPS(dB)CH1  NPS(dB)CH2  10log(S/A) Aislamiento
(dB)')
TdBFResumen=num2str(TabladBfinalResumen,'%15.1f');
disp(TdBFResumen)
disp('*****')
disp('*')
%////////////////////////////////////

```

```

%////////////////////////////////////
disp('*')
disp('                                TABLA DE RESULTADOS CON DESVIACIONES
ESTANDAR')
disp('*****
*****')
disp('Frecuencia NPS(dB)CH1   STD NPS1   NPS(dB)CH2   STD NPS2
10log(S/A) Aislamiento (dB) STD AISLAMIENTO')
TdBF=num2str(TabladBfinal,'%15.1f');
disp(TdBF)
disp('*****
*****')
disp('*')
%////////////////////////////////////

% disp('Para ver la graficas finales pulse ENTER')
% pause

% Graficas con detalles
% figure
%
% subplot(2,2,1)
% plot(BdeF,ValoresdB1todo)
% hold on
% plot(BdeF,ValoresdB1todo,'*r')
% grid
% title('NIVEL DE RUIDO DE CH1 TERCIOS DE OCTAVA')
% xlabel('Frecuencia (Hz)')
% ylabel('Nivel de ruido(dB)')
%
% subplot(2,2,2)
% plot(BdeF,ValoresdB1todo,'r')
% hold on
% plot(BdeF,ValoresdB1todo,'*r')
% hold on
% plot(BdeF,ValoresdB2todo,'b')
% hold on
% plot(BdeF,ValoresdB2todo,'.b')
% hold on
% grid
% title('NIVEL DE RUIDO DE CH1(***) CH2(...) TERCIOS DE OCTAVA')
% xlabel('Frecuencia (Hz)')
% ylabel('Nivel de ruido(dB)')
%
% subplot(2,2,3)
% bar(AislamientodBterciostodo)
% grid
% title('AISLAMIENTO ENTRE CANALES CHI Y CH2 TERCIOS DE OCTAVA')
% xlabel('Numero de banda')

```



```

% ylabel('R (dB)')
%
% subplot(2,2,4)
% plot(BdeF,AislamientoBterciostodo)
% hold on
% plot(BdeF,AislamientoBterciostodo,'*r')
% grid
%
% title('AISLAMIENTO ENTRE CANALES CHI Y CH2 TERCIOS DE OCTAVA')
% xlabel('Frecuencia (Hz)')
% ylabel('R (dB)')

%*****GRAFICADO FORMAL DE
RESULTADOS*****
%*****GRAFICADO FORMAL DE
RESULTADOS*****
%*****GRAFICADO FORMAL DE
RESULTADOS*****
disp('PARA VER EL GRAFICADO FORMAL DE RESULTADOS PULSE
ENTER')
pause

figure
semilogx(BdeF,ValoresdB1todo,'r','LineWidth',2)
hold on
semilogx(BdeF,ValoresdB1todo,'or','LineWidth',1)
hold on
semilogx(BdeF,ValoresdB2todo,'b','LineWidth',2)
hold on
semilogx(BdeF,ValoresdB2todo,'vb','LineWidth',1)
ylabel('NIVEL DE PRESION SONORA (dB)')
%title(['NIVELES POR BANDA: TRANSMITIDO (ooo)rojo NPS = 48.6dB,
INCIDENTE (vvv)azul NPS = 96.0dB'])
title(['NIVELES POR BANDA: TRANSMITIDO (rojo - o o o -) NPS =
',num2str(round(dB1005KCH1*10)/10),'dB INCIDENTE (azul - v v v -) NPS
= ',num2str(round(dB1005KCH2*10)/10),'dB PANEL "D"'])
GraticulaGeneral(NPSymin,NPSymax)

pause

figure
semilogx(BdeF,AislamientoBterciostodo,'LineWidth',3)
hold on
semilogx(BdeF,AislamientoBterciostodo,'*r','LineWidth',2)
ylabel('AISLAMIENTO SONORO R (dB)')
title(['AISLAMIENTO SONORO PANEL "D" STC =
',num2str(round(10*STC)/10),' dB'])
GraticulaGeneral(ymin,ymax)

```

```
disp('PULSE ENTER PARA VER LA GRAFICA DE VALORES FINALES CON  
DESVIACION ESTANDAR')
```

```
pause
```

```
NoBarra=1:18;
```

```
%Graficado de los niveles de presion sonora con desviacion estandar
```

```
figure
```

```
errorbar(NoBarra,ValoresdB1todo,STDNPS1,'k')
```

```
hold on
```

```
plot(NoBarra,ValoresdB1todo,'r','LineWidth',2)
```

```
hold on
```

```
plot(NoBarra,ValoresdB1todo,'*b','LineWidth',1)
```

```
hold on
```

```
errorbar(NoBarra,ValoresdB2todo,STDNPS2,'k')
```

```
hold on
```

```
plot(NoBarra,ValoresdB2todo,'b','LineWidth',2)
```

```
hold on
```

```
plot(NoBarra,ValoresdB2todo,'*r','LineWidth',1)
```

```
grid
```

```
ylabel('NIVEL DE PRESION SONORA (dB)')
```

```
title(['NIVELES POR BANDA CON STD: TRANSMITIDO (rojo) NPS =  
,num2str(round(dB1005KCH1*10)/10),'dB INCIDENTE (azul) NPS =  
,num2str(round(dB1005KCH2*10)/10),'dB PANEL "D"'])
```

```
GratNPSSTD(NPSymin,NPSymax)
```

```
pause
```

```
figure
```

```
errorbar(NoBarra,AislamientoBterciostodo,STDAislamientosNPS,'k')
```

```
hold on
```

```
plot(NoBarra,AislamientoBterciostodo,'LineWidth',2)
```

```
hold on
```

```
plot(NoBarra,AislamientoBterciostodo,'*r','LineWidth',1)
```

```
grid
```

```
ylabel('AISLAMIENTO SONORO R (dB)')
```

```
GratNPSSTD(ymin,ymax)
```

```
title(['AISLAMIENTO "TECNICA NPS" CON DESVIACION ESTANDAR. STC  
= ',num2str(round(10*STC)/10),' dB PANEL "D"'])
```

```
AislamientoBNPS=AislamientoBterciostodo;%Iguualado para salvarlo sepues  
con otro nombre
```

```
save AislamientoBNPS.mat AislamientoBNPS
```

```
save STDAislamientosNPS.mat STDAislamientosNPS
```

```
save ValoresdB1todo.mat ValoresdB1todo
```

```
save ValoresdB2todo.mat ValoresdB2todo
```

```
save STDNPS1.mat STDNPS1
```

```
save STDNPS2.mat STDNPS2
```

```
save dB1005KCH1.mat dB1005KCH1
```

```
save dB1005KCH2.mat dB1005KCH2
```

```
save BdeF.mat BdeF
```

```
disp('*****')
disp('PROGRAMA TERMINADO')
disp('IMPORTANTE: SE CREARON LOS ARCHIVOS AislamientoBNPS.mat y
STDAislamientosNPS.mat')

disp('Los tiene que copiar a la carpeta de Recopilacion')
disp('*')
disp('* * * * *')
disp('IMPORTANTE 2: Y TAMBIEN SE CREARON LOS ARCHIVOS
ValoresdB1todo.mat , ValoresdB2todo.mat , BdeF.mat')
disp('STDNPS1.mat Y STDNPS2.mat & dB1005KCH1.mat Y
dB1005KCH2.mat')
disp('PARA QUE LOS PUEDA CARGAR Y GRAFICAR DE NUEVO POR SI SE
QUERE DARLE OTRA PRESENTACION O AGREGAR INFORMACION')
disp('*')
disp('Tambien se grabaron los archivos ValoresdB1todo.mat y
ValoresdB2todo.mat')
disp('Por si quiere utilizarlos para graficar VS BdeF en otro momento')
```

TÉCNICA CEPSTRUM

calculooptim.m.

%Este programa calcula la altura de colocacion del bafle y los microfonos
%introduciendo los valores de la altura de la muestra(lm), la distancia al
%microfono 1 (dmic1), la distancia al microfono 2 (dmic2) t la temperatura
%ambiente (T) en °C.

```
function calculoX=calculooptim(lm,dmic2,dmic1,T)
```

```
% clear
```

```
% dmic1=0.01;
```

```
% dmic2=0.7;
```

```
% T=20;
```

```
% lm=1.22;
```

```
clc
```

```
c=331.5+0.607*T;
```

```
X=0;%Valores arbitrarios para inicializar los calculos
```

```
tretmic1=1; %Valores arbitrarios para inicializar los calculos
```

```
tretmic2=0;%Valores arbitrarios para inicializar los calculos
```

```
dmic2muestra=0.001;
```

```
n=0;
```

```
while tretmic2 <= tretmic1
```

```
    n=n+1;
```

```
    X=X+0.001;%Le incrementamos 1mm a la altura
```

```
    dREF=2*sqrt((dmic2/2)^2+X^2);%Distancia que recorre el sonido reflejado
```

```
    dDIF=sqrt((dmic2+dmic2muestra)^2+(lm-X)^2)+sqrt(dmic1^2+(lm-X)^2);%Distancia que recorre el sonido difractado
```

```
    dDIR=dmic2+dmic2muestra+dmic1;%Distancia total que recorre el sonido  
    directo hasta el mic1
```

```
    DIFERENCIAd=dREF-dmic2;%Diferencia de distancias entre lo reflejado y lo  
    directo al mic2.
```

```
    dmic2muestra=DIFERENCIAd/2;%Establecemos que la distancia del mic 2 a  
    la muestra debe ser la mitad.
```

```
    %Ya que la distancia del sonido reflejado de la muestra debe ir de ida
```

```
    %y de regreso. Esto se establece para que la distancia que recorre el
```

```
    %sonido reflejado por la muestra sea igual que la recorrida por la del
```

```
    %piso. Asi queda actualizada la nueva distancia del mic2 a la muestra
```

```
    %para el siguiente ciclo, y si ya es la optima, entonces ya se queda asi
```

```
    %LAS DISTANCIAS RECORRIDAS POR LOS SONIDOS REFLEJADOS  
    POR EL PISO Y LA
```

```
    %MUESTRA DEBEN SE IGUALES YA QUE ESTOS REFLEJOS DEBEN  
    LLEGAR AL MISMO
```

```
    %TIEMPO, MISMO EN EL QUE SE CERRARA LA VENTANA DE TIEMPO Y  
    NO SE
```

```

%REGISTRARA
tREF=dREF/c;%Tiempo de reflexion
tDIF=dDIF/c;%tiempo de difraccion
tDIR=dDIR/c;%Tiempo del sonido directo al mic1
tmic2=dmic2/c;%tiempo del sonido directo al mic2

tretmic2=tREF-tmic2;
tretmic1=tDIF-tDIR;
end
equis=num2str(X);
d2m=num2str(dmic2muestra);
d1m=num2str(dmic1);
DMIC2=num2str(dmic2);

TRETMIC2=num2str(tretmic2);
TRETMIC1=num2str(tretmic1);

disp('La altura de clocacion del altavoz y los microfonos es:');
disp(equis);
disp('metros');
disp('La distancia del mic2 a la muestra debe ser de');
disp(d2m);
disp('metros');

disp('La distancia del mic1 a la muestra (TRANSMITIDO) que propuso es
de');
disp(d1m);
disp('metros');
disp('La distancia del altavoz al mic2 que propuso es de');
disp(DMIC2);
disp('metros');

disp('LA duracion de la rafaga debe ser poco mas de:')
disp(TRETMIC2)
disp('segundos')

disp(' ')
disp('COMO DATO ADICIONAL PARA VISUALIZAR LOS RETRASOS SE
TIENE:')
disp('El tiempo de retraso del sonido REFLEJADO respecto al INCIDENTE
del segundo microfono es:')
disp(TRETMIC2)
disp('segundos')

disp('El tiempo de retraso del sonido DIFRACTADO respecto al
TRANSMITIDO del primer microfono es:')
disp(TRETMIC1)
disp('segundos')
disp('VEA QUE SON CASI IGUALES YA QUE LAS DISTANCIAS SON LAS
OPTIMAS')

```

calniv2mics.m.

```
%este programa permite verificar que los dos canales de entrada reciban el
mismo nivel colocados
%a la misma distancia. Ademas permite ajustar la distancia de los microfonos
para ver el retraso y asi
%se empiece a capturar cuando llegue la señal y no antes.
clc
clear
close all
Dur=0.01;%Duracion DE RAND de prueba
fs=44100;
Adirecta=1;%Amplitud de la señal directa
ts=1/fs;
t=0:ts:Dur;

%Calculando tiempo de llegada del primer reflejo
dmic1=1;%Distancia al mic1

f=1000;
%yrand=Adirecta*cos(2*pi*f*t);%Señal senoidal generable de frecuencia f o
bien se puede usar la expresion de abajo
yrand=Adirecta*2*(rand(1,floor(fs*Dur)+1)-0.5);%Señal aleatoria generadale
podemos agregar un dato mas
t=0:ts:(length(yrand)-1)/fs;

%*****
disp('Para comenzar a realizar la reproduccion y captura de señal en 2CHs pulse
ENTER')
pause
%Se genera la onda que será emitida.

ai1 = analoginput('winsound'); %Las vías y los canales de entrada y salida
se establecen -
canalid= addchannel(ai1,[1 2]);
set(ai1,'TriggerChannel',canalid(1),'TriggerChannel',canalid(2));
set(ai1,'SampleRate',fs);
fs=get(ai1,'SampleRate');
set(ai1,'SamplesPerTrigger',Dur*fs+1);%tamanorafaga/fs*fs
set(ai1,'TriggerType','Immediate');

ao = analogoutput('winsound');
chans = addchannel(ao,1);
set(ao,'SampleRate',fs);
set(ao,'TriggerType','Immediate');

putdata(ao,yrand);
```

```

start([ao,ai1]);          %Comienza la reproducción / grabación.

while (strcmp(ao.Running,'On'));
end

while (strcmp(ai1.Running,'On'));
end

data1 = getdata(ai1);%Se almacenan los datos de grabación.

delete(ai1);
delete(ao);

%yrms=sqrt( sum(y.*y)/ size(y,2) );

Smic1=sqrt( sum(data1(:,1).*data1(:,1))/ size(data1(:,1),1) );%RMS
mic1 (TODO JUNTO) se van acumulando en la posicion J.
Smic2=sqrt(sum(data1(:,2).*data1(:,2))/size(data1(:,2),1));%RMS mic2
Relmic1mic2=Smic1/Smic2;
Mic1mic2dB=20*log10(Relmic1mic2);

subplot(2,2,1)
plot(t,data1(:,1))
hold on
plot(t,data1(:,1),'.r')
title('Señal captada por CH1 (aprecie el retraso) MIC P/ TRANSMITIDA')
grid

subplot(2,2,2)
plot(t,yrand)
title('Señal generada y reproducida (aleatoria o senoidal)')
xlabel('Tiempo (s)')
grid

subplot(2,2,3)
plot(t,data1(:,2))
hold on
plot(t,data1(:,2),'.r')
title('Señal captada por CH2 (aprecie el retraso) MIC P/ INCIDENTE')
grid

subplot(2,2,4)
plot(yrand)
title('Señal generada y reproducida (aleatoria o senoidal)')
xlabel('Muestras')
grid

%CALCULO DEL AISLAMIENTO Y PRELIMINARES
disp('*')

```

```
disp('El nivel detectado por el mic1 es:')
disp(Smic1)
disp('El nivel detectado por el mic2 es:')
disp(Smic2)
disp('La relacion del mic1 con mic2 es:')
disp(Relmic1mic2)
disp('Su relacion en dB es:')
disp(Mic1mic2dB)
disp('*')
disp('Verifique que cada microfono capte el mismo nivel')
disp('Verifique que cada nivel RMS no tenga saturacion cuidando lo siguiente:')
disp('Verifique las curvas de las señales capturadas no esten recortadas (no
rebasen 0.5 a 0.81) de lo contrario existe saturacion')
disp('Cuando se tenga cerca de 0dB en la relacion, los mics estan recibiendo y
entregando el mismo RMS para condiciones iguales')
disp('Este valor refleja el error en dB que tendria su medicion. Mueva el control
Cal. hacia arriba')
disp('*')
disp('TERMINO DE PRUEBA')

end
```


ApreparativoCepstrum.m.

```
%Aislamientocepstrum aplicando una rafaga de ruido blanco de la que se
obtendra
%el Cepstrum para ser vetaneada y despues filtrada
clc
clear
close all
%function muestracapturada = nivtercioscap(tiempo,fmuestreo)
T=15;%TEMPERATURA AMBIENTE
c=331.5+0.607*T;
muestrascorte=1;
CompensaciondB=40;
Dur=0.01;%Duracion de la RAFAGA DE RAND FILTRADO
Duraleatoria=0.5;%Duracion del ruido blanco para filtrar y luego recortar
Tgrabacion=0.05;%Tiempo de grabacion de los microfonos en segundos
fs=44100;
ts=1/fs;
Adirecta=2;%Amplitud de la señal directa
t=0:ts:Dur;
razon=0.05;%Ratio para la ventana tukey para aplicar en el tiempo a la señal
transmitida
razon2=0.05;%Ratio2 para la ventana tukey2 para aplicar en el primer reflejo
porcentaje=10;%Introduzca aqui el porcentaje que ocupara la ventana Kaiser
de bajada
%de la señal directa para aplicarse en el CEPSTRUM y asi obtener el
%liftering
beta=10;%Aqui se introduce el valor BETA para la ventana Kaiser.
%Mientras mas alta, se veran mas las asintotas en los extremos.
Dretraso=0.40;%Distancia de retraso de la grabacion PARA EL MIC 2
(INCIDENTE)
Tretraso=Dretraso/c;%tiempo de retraso de la grabacion
Muestrasretraso=floor(Tretraso*fs);%Muestras de retraso de la grabacion
%MuestrasretrasoT1=Muestrasretraso-0;%Se hace esta compensacion debido
a que experimentalmente se tiene que posponer 182 muestras el ventaneo en
promedio
%Muestrasretraso=Muestrasretraso-0;%Se hace esta compensacion debido a
que experimentalmente se tiene que posponer 50 muestras el ventaneo en
promedio
CompensaciondBmic2=30;
NoPromedios=5;%NUMERO DE ENSAYOS A REALIZAR (NUMERO DE
PROMEDIOS)
ymin=10;%Limites inferior y superior para graficar el aislamiento
ymax=50;
CEROSMAS=1;%ESTO ES POR SI SE TIENEN QUE AGRAGAR MAS
CEROS A LA VENTANA DE SEÑAL TRANSMITIDA

%Calculando tiempo de llegada del primer reflejo
lm=1.23;%Altura de la muestra
```



```

% */**/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/.
% */**/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/.

%   for n = 1:nt
%       f = BdeF(n);
%           wn = [round((f)*2^(-1/6)) round((f)*2^(1/6))];%Se establecen las
frecuencias de corte
%       [b,a] = butter(3,wn/fs*2);%filtro fabricado
%       filtrada = filter(b,a,yrand);
%       yfilt(n,:)=filtrada;%En la fila n, todas las columnas, se almacena la señal
filtrada del tercio n evaluado.
%       niv(n) = sqrt(sum(filtrada.*filtrada)/size(filtrada,1));%Se calcula el RMS
de la señal filtrada
%   end
%
%
% figure
% bar(niv,'y')
% GraticulaRF(0,12,BdeF)
% title('Espectro de Ruido Blanco (Duraleatoria, 0.5s)')
% grid
%
% figure
% plot(BdeF,niv,'*')
% grid
%
% figure
% loglog(BdeF,niv,'*r')
% hold on
% bar(BdeF,niv)
% grid

disp('ESTE PROGRAMA CALCULA EL AISLAMIENTO SONORO CON
TECNICA CEPSTRUM EMITIENDO RAFAGAS DE RUIDO BLANCO')
disp('Aislamiento Cepstrum aplicando una rafaga de ruido blanco de la que se
obtendra')
disp('el Cepstrum para ser vetaneada y despues filtrada en N ensayos')
disp('*****')
disp('YA FUERON CARGADAS LAS VARIABLES PRELIMINARES; SI YA
TIENE LOS ARCHIVOS YSUMA')
disp('CORRA EL PROGRAMA AProcesamientoCepstrum.m, DE LO
CONTRARIO DEBE GRABAR LAS RAFAGAS')
disp('CON EL PROGRAMA ACapturaCepstrum.m')
disp('*****')
disp('*')

```

ACapturaCepstrum.m.

```
disp('*****  
*****')  
disp('YA FUERON CARGADAS LAS VARIABLES PRELIMINARES, AHORA  
GRABE LAS RAFAGAS Y CUANDO TERMINE')  
disp('CORRA EL PROGRAMA AProcesamientoCepstrum.m, CONTINUE  
PULSANDO ENTER...')  
disp('*****  
*****')  
disp('*')  
pause  
Ctexto0=sprintf('SE LLEVARAN A CABO%2d ENSAYOS',NoPromedios);  
disp(Ctexto0)  
disp('*')  
disp('Para comenzar a realizar la medicion pulse ENTER')  
pause  
close all  
  
%Inicia ciclo for para la EMISION DE RAFAGAS Y CAPTURA PARA UN  
NUMERO DE ENSAYOS PROGRAMADOS  
for P=1:NoPromedios  
    yrand=Adirecta*(rand(1,floor(fs*Duraleatoria)+1)-0.5);%Señal aleatoria  
    generadale podemos agregar un dato mas  
    senal=yrand(800:800+floor(Dur*fs));%Señal fabricada para ser emitida. Se  
    corta desde la muestra 800 hasta el numero de muestras de la rafaga  
    tamanorafaga=length(senal); %numero de muestras de la rafaga por emitida.  
  
    Ctexto1=sprintf('Emitiendo la rafaga %2d de %2d. ',P,NoPromedios);  
    disp(Ctexto1)  
  
    close all  
    ai1 = analoginput('winsound'); %Las vías y los canales de entrada y salida  
    se establecen -  
    canalid= addchannel(ai1,[1 2]);  
    set(ai1,'TriggerChannel',canalid(1),'TriggerChannel',canalid(2));  
    set(ai1,'SampleRate',fs);  
    fs=get(ai1,'SampleRate');  
    set(ai1,'SamplesPerTrigger',floor(Tgrabacion*fs)+1);%tamanorafaga/fs*fs  
    con un dato mas para que coincida con el vector de tiempo  
    set(ai1,'TriggerType','Immediate');  
  
    ao = analogoutput('winsound');  
    chans = addchannel(ao,1);  
    set(ao,'SampleRate',fs);  
    set(ao,'TriggerType','Immediate');  
    putdata(ao,senal);
```

```

start([ao,ai1]);          %Comienza la reproducción / grabación.

while (strcmp(ao.Running,'On'));
end

while (strcmp(ai1.Running,'On'));
end

data1 = getdata(ai1);%Se almacenan los datos de grabación.
delete(ai1);
delete(ao);

ysuma(:,P)=(data1(:,1)); %Los datos almacenados en data1 (señal
transmitida,columna P,mic1)
%o sea que esta es la señal a la que hay que aplicar el Cepstrum para analizar
ysuma2(:,P)=(data1(:,2)); %Los datos almacenados en data1 (señal directa con
reflejos, columna P,mic2)
%o sea que esta es la señal a la que hay que aplicar la ventana tukey2

%GRAFICADO DE SEÑALES GRABADAS*****
figure
subplot(3,1,1)
plot(vt,ysuma(:,P),'r')
title('SEÑAL CAPTURADA EN FUNCION DEL TIEMPO POR MIC1
(TRANSMITIDA)')
%xlabel('Tiempo (s)')
%ylim([-2 2])
grid

subplot(3,1,2)
plot(vt,ysuma2(:,P),'r')
title('SEÑAL CAPTURADA EN FUNCION DEL TIEMPO POR MIC2
(INCIDENTE)')
%xlabel('Tiempo (s)')
%ylim([-2 2])
grid

subplot(3,1,3)
plot(t,senal,'r')
%axis([0 t(length(t)) -2 2])
title('SEÑAL FABRICADA Y REPRODUCIDA EN FUNCION DEL TIEMPO
(RAFAGA)')
%xlabel('Tiempo (s)')
grid

Textodeprom=sprintf('El ensayo %2d de %2d ya fue
realizado',P,NoPromedios);
disp(Textodeprom)

```


AProcesamientoCepstrum.m.

```

%*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/
%*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/
%*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/
%AQUI YA TENEMOS LOS VECTORES ysuma Y ysuma2 que contienen las
rafagas
%capturadas se puede si se quiere correr el programa desde aqui si ya se
%tienen los dos archivos mencionados.
disp('****Se cargaran los archivos ysuma.mat y ysuma2.mat que se requieren
en este programa****')
disp('****a ejecutar llamado AProcesamientoCepstrum****')
disp('Para iniciar el procesamiento de las rafagas capturadas pulse ENTER')
disp('*****')
pause

load ysuma.mat
load ysuma2.mat

%FABRICACION DE VENTANA TUKEY2 PARA MEDIR EL REFLEJO
SIEMPRE SERA LA MISMA
%*****
tw2=tukeywin(muestrasretraso-1,razon2);%su tamano debe se el doble de
mic2-muestra
cerosfinales=zeros(1,floor(Tgrabacion*fs)-length(tw2)+1);%ojo AQUI SE LE
AGREGO UN VALOR MAS PARA QUE CUADRE
vfintukey2=[tw2'cerosfinales];

%Vamos a quitarle agunos elementos al principio si fuera necesario
%
ysuma=[ysuma(muestrascorte+1:length(ysuma));zeros(muestrascorte,1)];%ojo
ojo ojo LE ESTAMOS QUITANDO 100 MUESTRAS AL PRINCIPIO
%
ysuma2=[ysuma2(muestrascorte+1:length(ysuma2));zeros(muestrascorte,1)];%
ojo ojo ojo LE ESTAMOS QUITANDO 100 MUESTRAS AL PRINCIPIO

for L=1:NoPromedios

    %Vamos calcular el Cepstrum de la seÑaltransmitida global numero L
    %.....
    clc
    close all
    Textoderafaga1=sprintf('Se evaluara el CEPSTRUM GLOBAL del ensayo
%2d de %2d',L,NoPromedios);
    Textoderafaga2=sprintf('Para ver las seÑales %2d que seran analizadas,
pulse ENTER',L);
    disp(Textoderafaga1)
    disp(Textoderafaga2)

```

```

pause
figure
subplot(2,1,1)
plot(vt,ysuma(:,L),'r')
title('SEÑAL CAPTURADA POR MIC1 (TRANSMITIDA) f(t)')
xlabel('Tiempo (s)')
grid

subplot(2,1,2)
plot(vt,ysuma2(:,L),'r')
title('SEÑAL CAPTURADA POR MIC2 (INCIDENTE) f(t)')
xlabel('Tiempo (s)')
grid

disp('Para evaluar el Cepstrum Global de la señal TRANSMITIDA pulse
ENTER')
pause
[y ym] = rceps(ysuma(:,L));%calculando el Cepstrum real del vector ysuma
que es la señal transmitida
figure
subplot(2,1,1)
plot(y,'r')
hold on
plot(y,'.b')
title('')
title(['CEPSTRUM REAL DE LA SEÑAL' ,num2str(L),'TRANSMITIDA,
f(muestras)'])
xlabel('Muestras')
grid
subplot(2,1,2)
plot(vt,y,'r')
title(['CEPSTRUM REAL DE LA SEÑAL' ,num2str(L),'TRANSMITIDA, f(t)'])
xlabel('Tiempo (s)')
grid

%*****PROCESO DE RECTIFICADO DEL CEPSTRUM
CORTADO*****
ycepscortado=y(1:floor(Dur*fs)+1);%El corte debe ser del tamaño de lo que
dura una rafaga +1.este tiempo
%es el que nos interesa, lo que sigue adelante no nos interesa.
tcepsrec=vt(1:floor(Dur*fs)+1);%Vector de tiempo recortado con el mismo
numero de muestras que yceps recortado
for n=1:floor(Dur*fs)+1
    if ycepscortado(n) < 0
        ycepsrectificado(n)=0;
    else
        ycepsrectificado(n)=ycepscortado(n);%ycepsrectificado es el cepstrum
rectificado (lo dejamos igual)
    end
end
end

```



```

pedacitosceros=zeros(1,mretdifracCOM);
ycepsceros=[pedacitosceros
ycepsrectificado(mretdifracCOM+1:length(ycepsrectificado))];
%ycepsceros es el cepstrum con la parte inicial en ceros y servira para
%buscar el primer reflejo.

disp('Para ver el Cepstrum cortado y rectificado pulse ENTER')
pause
figure
subplot(2,1,1)
plot(ycepsrectificado,'.-b')
hold on
plot(ycepsceros,'.-r')
grid
title('CEPSTRUM REAL RECORTADO Y RECTIFICADO, ORIGINAL(AZUL)
SIN SEÑAL DIRECTA(ROJO) EN FUNCION DE LAS MUESTRAS')
xlabel('Muestras')
subplot(2,1,2)
plot(tcepsrec,ycepsrectificado,'b')
hold on
plot(tcepsrec,ycepsceros,'r')
grid
title('CEPSTRUM REAL RECORTADO Y RECTIFICADO, ORIGINAL(AZUL)
SIN SEÑAL DIRECTA(ROJO) EN FUNCION DEL TIEMPO')
xlabel('Tiempo (s)')

disp('Para fabricar la ventana para la señal TRANSMITIDA pulse ENTER')
pause
%oooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooo
oooooo
%***FABRICACION DE VENTANA TUKEY1 PARA APLICAR A LA SEÑAL
TRANSMITIDA CAPTURADA POR EL MIC1*****

[J,K]=max(ycepsceros);%Busco el el maximo con valor J y su posicion en K.
%Creando una ventana cuadrada de magnitud K-1 (sin tomar en cuenta el
reflejo

%Fabricacion de ventana TUKEY
tamañoventana=K-1;%Es de tamaño K-1 porque en la muestra K se presenta
la espiga por lo que si la ventana tiene una muestra menos, entonces se evitara
a toda costa la presencia de de ella.
%tamañoventana=muestrasretraso;
%*****
%tamañoventana=(muesdllegada-muestrasdirecto)*2;%OJO OJO OJO OJO
OJO OJO EST ELIMINA EL CEPSTRUM
%*****
cerosantes=zeros(1,muesbocinamuestramic1-
muestrasdirecto+CEROSMAS);%Estos ceros deben estar antes que la ventana
comience ya que es el tiempo que tardara en legar la transmision

```

```
tw=tukeywin(tamanoventana,razon);
cerosdespues=zeros(1,length(ysuma)-(tamanoventana)-
length(cerosantes));%Estos ceros son para contruir la ventana para aplicar ala
señal que tiene todos los reflejos
```

```
venttukeycerostukeyceros=[cerosantes tw' cerosdespues];
vfintukey=venttukeycerostukeyceros;%Ventana tukey final para aplicarse al la
señal original
%vfintukeyceps=[tw' cerosdespues2];%Ventana tukey final para aplicarse al
cepstrum
%filtrada=[filtrada(muestrascorte+1:length(filtrada));zeros(muestrascorte,1)];%oj
o ojo ojo LE ESTAMOS QUITANDO 100 MUESTRAS AL PRINCIPIO
```

```
%GRAFICADO DE LA VENTANA TUKEY SOBRE LA SEÑAL A VENTANEAR
```

```
figure
subplot(2,1,1)
plot(vfintukey,'.-r')
hold on
plot(ysuma(:,L),'b')
title('VENTANA TUKEY Y LA SEÑAL TRANSMITIDA GLOBAL, f(muestras)')
xlabel('Muestras')
grid
```

```
subplot(2,1,2)
plot(vt,vfintukey,'.-r')
hold on
plot(vt,ysuma(:,L)/max(ysuma(:,L)),'b')
title('VENTANA TUKEY Y LA SEÑAL TRANSMITIDA GLOBAL
NORMALIZADA, f(t)')
xlabel('Tiempo (s)')
grid
```

```
ysumaventtukey=ysuma(:,L)'.*vfintukey;%VENTANEO DE LA SEÑAL CON
REFLEJOS COMPLETO CON TUKEY
```

```
%GRAFICADO DE LA VENTANA TUKEY YSEÑAL CON REFLEJOS
VENTANEADA
```

```
% pause
% figure
% subplot(2,1,1)
% plot(vfintukey,'r')
% hold on
% plot(vfintukey,'.r')
% hold on
% plot(ysumaventtukey,'b')
% title('SEÑAL TRANSMITIDA YA VENTANEADA CON TUKEY EN FUNCION
DE LAS MUESTRAS')
% xlabel('Muestras')
% grid
%
```

```

% subplot(2,1,2)
% plot(vt,vfintukey,'r')
% hold on
% plot(vt,vfintukey,'.r')
% hold on
% plot(vt,ysumaventtukey,'b')
% title('SEÑAL TRANSMITIDA YA VENTANEADA CON TUKEY EN FUNCION
DEL TIEMPO')
% xlabel('Tiempo (s)')
% grid

```

```

%AQUI SE SE UTILIZA LA SEÑAL ORIGINAL CAPTURADA VENTANEADA
PARA OBTENER EL ESPECTRO
%DE LA SEÑAL RECUPERADA
%LA SEÑAL QUE ES TRANSFORMADA TIENE TODOS LOS CEROS DEL
INICIO Y DEL FINAL

```

```

fftabsysumavent=abs(fft(ysumaventtukey));
fftsumaventrec=fftabsysumavent(1:floor(length(fftabsysumavent)/2)+1);
vectorfrecysumavent=(0:fs/length(ysumaventtukey):fs/2);

```

```

% pause
% figure
% subplot(2,1,1)
% plot(vt,vfintukey,'r')
% hold on
% plot(vt,vfintukey,'.r')
% hold on
% plot(vt,ysumaventtukey,'b')
% title('SEÑAL TRANSMITIDA VENTANEADA CON TUKEY EN FUNCION
DEL TIEMPO')
% xlabel('Tiempo (segundos)')
% grid
%
% subplot(2,1,2)
% plot(vectorfrecysumavent,fftsumaventrec,'b')
% hold on
% plot(vectorfrecysumavent,fftsumaventrec,'.b')
% title('ESPECTRO DE LA SEÑAL TRANSMITIDA VENTANEADA DE ARRIBA
(CON CEROS AL PRINCIPIO Y AL FINAL)')
% xlabel('Frecuencia (Hz)')
% grid

```

```

%*****PROCESO PARA RECUPERAR CORTAR LA PARTE DE LA SEÑAL
TRANSMITIDA (extraerla)*****
ysventv=ysumaventtukey(CEROSMAS+muesbocinamuestramic1-
muestrasdirecto+1:CEROSMAS+muesbocinamuestramic1-
muestrasdirecto+tamanoventana);%Parte de la onda recuperada del vetaneo

```

```

tysventv=0:ts:(length(ysventv)-1)*ts;%Vector de tiempo construido para la señal
recuperada

% pause
% figure
% subplot(3,1,1)
% plot(vt,vfintukey,'r')
% hold on
% plot(vt,vfintukey,'.r')
% hold on
% plot(vt,ysumaventtukey,'b')
% title('SEÑAL TRANSMITIDA VENTANEADA CON TUKEY EN FUNCION
DEL TIEMPO')
% xlabel('Tiempo (segundos)')
% grid
%
% subplot(3,1,2)
% plot(ysventv)
% hold on
% plot(ysventv,'*')
% title('SEÑAL TRANSMITIDA RECUPERADA RECORTADA Y VENTANEADA
EN FUNCION DE LAS MUESTRAS')
% xlabel('Muestras')
% grid
%
% subplot(3,1,3)
% plot(tysventv,ysventv)
% hold on
% plot(tysventv,ysventv,'*')
% title('SEÑAL TRANSMITIDA RECUPERADA RECORTADA Y VENTANEADA
EN FUNCION DEL TIEMPO')
% xlabel('Tiempo (segundos)')
% grid
%LA TRANSFORMADA EN LA FRECUENCIA CORRECTA
fftyventv=fft(ysventv);
fftabsyventv=abs(fftyventv);
fftyventrec=fftabsyventv(1:floor(length(fftabsyventv)/2)+1);
vectorfrecysventv=(0:fs/length(ysventv):fs/2);

% SEÑAL RECUPERADA (transmitida) ysventv CON SU VECTOR DE
TIEMPO tysventv
%.....

%////////////////////////////////////
%GRAFICADO DE LA VENTANA TUKEY2 SOBRE LA SEÑAL INCIDENTE
DETECTADA POR EL MIC2
disp('Para ventanear la señal INCIDENTE pulse ENTER')
pause

```

```

figure
subplot(2,1,1)
plot(vfintukey2,'.-r')
hold on
plot(ysuma2(:,L),'b')
title('VENTANA TUKEY2 Y LA SEÑAL INCIDENTE PROPORCIONAL
f(muestras)')
xlabel('Muestras')
grid

```

```

subplot(2,1,2)
plot(vt,vfintukey2,'.-r')
hold on
plot(vt,ysuma2(:,L)/max(ysuma2(:,L)),'b')
title('VENTANA TUKEY2 Y LA SEÑAL INCIDENTE NORMALIZADA f(t)')
xlabel('Tiempo (s)')
grid

```

```

ysumaventtukey2=ysuma2(:,L).*vfintukey2;%VENTANEO DE LA SEÑAL CON
REFLEJOS COMPLETO CON TUKEY2

```

```

%CALCULO PARA LA OBTENCION DE LA SENAL DIRECTA (extraerla)

```

```

Sdirecta=ysumaventtukey2(1:length(tw2));
tSdirecta=vt(1:length(tw2));
%Sdirecta ES LA SEÑAL RECORTADA Y AHORA SE VA A FILTRAR junto
con ysventv

```

```

%GRAFICADO DE LAS DOS SEÑALES VENTANEADAS QUE ESTAN
LISTAS PARA SER
%FILTRADAS
disp('LAS DOS SEÑALES VENTANEADAS YA ESTAN LISTAS PARA SER
FILTRADAS, pulse ENTER PARA VERLAS')
pause

```

```

figure
subplot(2,1,1)
plot(tysventv,ysventv,'.-r')
title('SEÑAL TRANSMITIDA VENTANEADA LISTA PARA FILTRAR f(t)')
xlabel('Tiempo (s)')
grid

```

```

subplot(2,1,2)
plot(tSdirecta,Sdirecta,'.-r')
title('SEÑAL INCIDENTE VENTANEADA LISTA PARA FILTRAR f(t)')
xlabel('Tiempo (s)')
grid

```

```
%ESTA PARTE ES PARA EL VENTANER DEL CEPSTRUM CON KAISER, SI
FUERA NECESARIO SE HABILITARA *****
%*****
%*****
```

```
%*****KAISER PARA
LIFTER*****
```

```
%*****
%FABRICACION DE VENTANA KAISER PARA APLICARSELA AL
CEPSTRUM DE BAJADA CON UNOS AL PRINCIPIO
%tamanoventana=K-1;recuerda
%porcentaje=10;%Introduzca aqui el porcentaje que ocupara la ventana kaiser
de bajada de la señal directa
partekaiser=floor(tamanoventana*porcentaje/100);%Numero de muestras que
tendra la parte kaiser de bajada
ventkaiser=kaiser(partekaiser*2,beta);%Aqui se introduce el valor BETA de
kaiser
ventkaiserbajada=ventkaiser(floor(length(ventkaiser)/2)+1:length(ventkaiser));
unoskaiser=ones(1,tamanoventana-length(ventkaiserbajada));
ceroskaiser=zeros(1,length(y)-tamanoventana);
ventkaisercerosunos=[unoskaiser ventkaiserbajada' ceroskaiser];
vfinkaiser=ventkaisercerosunos;
```

```
%.....Zona de multiplicacion para ventaneo
cepsventkaiser=y.*vfinkaiser;%VENTANEO DEL CEPSTRUM COMPLETO
CON KAISER
```

```
%*****GRAFICADO DE LA VENTANA KAISER CONSTRUIDA SOBRE
EL CEPSTRUM A VENTANEAR*****
```

```
% pause
% figure
% subplot(2,1,1)
% plot(y,'b')
% hold on
% plot(y,'.b')
% hold on
% plot(vfinkaiser,'r')
% hold on
% plot(vfinkaiser,'.r')
% title('VENTANA KAISER FABRICADA PARA APLICARSE AL CEPSTRUM
EN FUNCION DE LAS MUESTRAS')
% xlabel('Muestras')
% grid
%
% subplot(2,1,2)
% plot(vt,y,'b')
% hold on
% plot(vt,y,'.b')
% hold on
```

```

% plot(vt,vfinkaiser,'r')
% hold on
% plot(vt,vfinkaiser,'.r')
% title('VENTANA KAISER FABRICADA PARA APLICARSE AL CEPSTRUM
EN FUNCION DEL TIEMPO')
% xlabel('Tiempo (segundos)')
% grid

%*****GRAFICADO DEL CEPSTRUM VENTANEADO CON KAISER
JUNTO CON LA VENTANA APLICADA*****
% pause
% figure
% subplot(2,1,1)
% plot(cepsventkaiser,'b')
% hold on
% plot(cepsventkaiser,'.b')
% hold on
% plot(vfinkaiser,'r')
% hold on
% plot(vfinkaiser,'.r')
% title('CEPSTRUM YA VENTANEADO CON KAISER EN FUNCION DE LAS
MUESTRAS')
% xlabel('Muestras')
% grid
%
% subplot(2,1,2)
% plot(vt,cepsventkaiser,'b')
% hold on
% plot(vt,cepsventkaiser,'.b')
% hold on
% plot(vt,vfinkaiser,'r')
% hold on
% plot(vt,vfinkaiser,'.r')
% title('CEPSTRUM YA VENTANEADO CON KAISER EN FUNCION DEL
TIEMPO')
% xlabel('Tiempo (segundos)')
% grid

%*****I M P O R T A N T E*****
%AQUI SE SE UTILIZA EL CEPSTUM VENTANEADO PARA OBTENER EL
ESPECTRO LIFTER
fftabscepskaiser=abs(fft(cepsventkaiser));
fftcepsreckaiser=fftabscepskaiser(1:floor(length(fftabscepskaiser)/2)+1);
vectorfreckaiser=(0:fs/length(y):fs/2);

% pause
% figure
% subplot(2,1,1)
% plot(vectorfreckaiser,fftcepsreckaiser)
% hold on

```

```

% plot(vectorfreckaiser,fftcepsreckaiser,'.r')
% title('ESPECTRO LIFTER DEL CEPSTRUM VENTANEADO CON KAISER')
% xlabel('Frecuencia (Hz)')
% grid
%
% subplot(2,1,2)
% fftcepskaiser=fft(cepsventkaiser);
% yreconstkaiser=ifft(fftcepskaiser);
% plot(vt,real(yreconstkaiser))
% title('TRANSFORMADA INVERSA DEL ESPECTRO LIFTER = CEPSTRUM
VENTANEADO (Comprobacion)')
% xlabel('Tiempo (segundos)')
% grid
%*****

%*****
%*****
% SEÑAL RECUPERADA (transmitida) ysventv CON SU VECTOR DE
TIEMPO tysventv
% SEÑAL RECUPERADA (incidente) Sdisrecta CON SU VECTOR DE
TIEMPO tSdirecta
disp('Las señales transmitida e incidente ya estan listas para ser filtradas, pulse
Enter para procesarlas')
pause
close all
clc
    for M=1:nt
        f = BdeF(M);
        wn = [round((f)*2^(-1/6)) round((f)*2^(1/6))];%Se establecen las
frecuencias de corte
        [b,a] = butter(3,wn/fs*2);%filtro fabricado
        filtrada = filter(b,a,ysventv);
        filtrada2= filter(b,a,Sdirecta);
        Ctexto1=sprintf('Analizando Frecuencia %2d Hz. Ensayo %2d de %2d.
Realizando...',f,L,NoPromedios);
        disp(Ctexto1)

figure
subplot(2,1,1)
plot(tysventv,filtrada,'r')
title(['SEÑAL FILTRADA POR MIC1 TRANSMITIDA (',num2str(f),' Hz)'])
xlabel('Tiempo (s)')
grid

subplot(2,1,2)
plot(tSdirecta,filtrada2,'r')
title(['SEÑAL FILTRADA POR MIC2 INCIDENTE (',num2str(f),' Hz)'])
xlabel('Tiempo (s)')
grid

```



```
disp('Pulse ENTER repetidamente para ver las graficas que muestran el
procesamiento de datos...')
disp('*')
```

```
RMSmic1(M,L)=sqrt( sum(filtrada.*filtrada)/ size(filtrada,1) );%RMS mic1
(TODO JUNTO) se van acumulando en la posicion J.
RMSmic2(M,L)=sqrt(sum(filtrada2.*filtrada2)/size(filtrada2,1));%RMS mic2
sonido incidente DIRECTO
```

```
disp('PULSE ENTER PARA SEGUIR PROCESANDO LA SIGUIENTE
FRECUENCIA DEL MISMO ENSAYO ')
```

```
disp('*****
****')
disp('*')
disp('*')
pause(0.5)
```

```
end%Este end corresponde al final de cada pormedio que se hace. Hay que
ver si es necesario cambiarlo en el futuro
```

```
dBcaida=20*log10(dpartida/dllegada);
FraccionCaida=10^(dBcaida/20);
RMSmic2Caido=RMSmic2(:,L)*FraccionCaida;
```

```
%CALCULO DEL AISLAMIENTO PROVISIONAL Y PRELIMINARES
```

```
AislamientodB=20*log10(RMSmic2Caido./RMSmic1(:,L))+CompensaciondB;
ValorSTC=STCf(AislamientodB);
TablaAislamientos(L,:)=AislamientodB';
TablaSTC(L,1)=ValorSTC;
```

```
% disp('El nivel detectado por el mic1 es:')
% disp(RMSmic1)
% disp('El nivel detectado por el mic2 es:')
% disp(RMSmic2)
% disp('El nivel que llega a la pared es:')
% disp(Nivpared)
% disp('El aislamiento es de:')
% disp(AislamientodB)
```

```
clc
```

```
TextoValoresRecien=sprintf('Los valores recien calculados para el ensayo%2d
son:',L);
disp(TextoValoresRecien)
STCtexto=sprintf('STC PROVISIONAL Banda 100-5KHz= %3.1f
dB.',ValorSTC);
disp(STCtexto)
disp('*****')
```

```

Textocaida=sprintf('La caida del mic2 a la muestra es de %2.1f dB',dBcaida);
disp(Textocaida)
disp('*****')
disp('Frecuencia   RMSmic1   RMSmic2   RMSmic2caido   Aislamiento dB
(20log(RMSmic2Caído/RMSmic1)')

tablavalores=[BdeF'      RMSmic1(:,L)      RMSmic2(:,L)      RMSmic2Caído
Aislamiento dB];
TV=num2str(tablavalores,'%13.1f');
disp(TV)
disp('*****')
disp('Para ver la grafica que se obtuvo para este ensayo, pulse ENTER')
pause

figure
semilogx(BdeF,Aislamiento dB,'LineWidth',3)
hold on
semilogx(BdeF,Aislamiento dB,'*r','LineWidth',2)
ylabel('AISLAMIENTO SONORO R (dB)')
title(['SOUND TRANSMISSION CLASS          STC          =
',num2str(round(10*ValorSTC)/10),' dB (PROVISIONAL)'])
GraficulaGeneral(ymin,ymax)
CFrec=sprintf('TERMINADO DE PROCESANDO DEL ENSAYO %2d',L);
disp(CFrec)
disp('PULSE ENTER PARA CONTINUAR')
disp('*')

pause
clc
end
%////////////////////////////////////
%En esta seccion obtendremos la desviacion estandar de debido a cada uno
%de los ensayos
STDAislamientosCepstrum=std(TablaAislamientos);

%////////////////////////////////////
identidad=ones(NoPromedios,1);
RMSprom1=(RMSmic1*identidad)/NoPromedios;%Calculando promedios de
RMS
RMSprom2=(RMSmic2*identidad)/NoPromedios;%Calculando promedios de
RMS

RMSprom2Caído=RMSprom2*FraccionCaída;
Aislamiento dB=20*log10(RMSprom2Caído./RMSprom1)+Compensacion dB;

%CALCULO DEL AISLAMIENTO Y PRELIMINARES

%*****CALCULANDO STC*****

```

```

ValorSTC=STCf(Aislamiento dB);
disp('LOS RESULTADOS FINALES SON:')
STCtexto=sprintf('STC Banda 100-5KHz= %3.1f dB.',ValorSTC);
disp(STCtexto)
disp('*****')
Textocaida=sprintf('La caída del mic2 a la muestra es de %2.1f dB',dBcaida);
disp(Textocaida)
disp('*****')
*****

tablavalores=[BdeF RMSprom1 RMSprom2 RMSprom2Caido Aislamiento dB
STDAislamiento Cepstrum'];
TV=num2str(tablavalores,'%15.1f');

disp('Frecuencia      RMSprom1      RMSprom2      RMSprom2caido
Aislamiento dB (20log(RMSprom2caido/RMSprom1)) STD Aislamiento')
disp(TV)
disp('*****')
*****

disp('TERMINO DE PROCESAMIENTO')
disp('*')
disp('PULSE ENTER PARA VER LA GRAFICA DE VALORES FINALES')
pause

figure
semilogx(BdeF,Aislamiento dB,'LineWidth',3)
hold on
semilogx(BdeF,Aislamiento dB,'*r','LineWidth',2)
ylabel('AISLAMIENTO SONORO R (dB)')
title(['AISLAMIENTO SONORO      TECNICA CEPSTRUM      STC =
',num2str(round(10*ValorSTC)/10),' dB  PANEL "B"'])
GraticulaGeneral(ymin,ymax)

disp('PULSE ENTER PARA VER LA GRAFICA DE VALORES FINALES CON
DESVIACION ESTANDAR')
pause
NoBarra=1:nt;

figure
errorbar(NoBarra,Aislamiento dB,STDAislamiento Cepstrum,'r')
hold on
plot(NoBarra,Aislamiento dB,'LineWidth',3)
hold on
plot(NoBarra,Aislamiento dB,'*r','LineWidth',2)
grid
ylabel('AISLAMIENTO SONORO R (dB)')
GratCepsSTD(ymin,ymax)
title(['AISLAMIENTO CEPSTRUM CON DESVIACION ESTANDAR. STC =
',num2str(round(10*ValorSTC)/10),' dB'])

```

```
AislamientoBCepstrum=AislamientoB;%Igualeado para salvarlo sepues con
otro nombre
save AislamientoBCepstrum.mat AislamientoBCepstrum
save STDAislamientosCepstrum.mat STDAislamientosCepstrum
```

```
disp('PULSE ENTER PARA VER LAS VENTANAS UTILIZADAS')
pause
```

```
%GRAFICANDO LAS VENTANAS KAISER POR PROCESAR
```

```
figure
subplot(2,2,1)
plot(ventkaiser)
title('VENTANA KAISER UTILIZADA')
ylim([-0.2 1.2])
grid
```

```
subplot(2,2,2)
plot(ventkaiserbajada)
title('VENTANA KAISER DE BAJADA UTILIZADA')
ylim([-0.2 1.2])
grid
```

```
subplot(2,2,3)
plot(tw)
title('VENTANA TUKEYWIN1 UTILIZADA')
ylim([-0.2 1.2])
grid
```

```
subplot(2,2,4)
plot(tw2)
title('VENTANA TUKEYWIN2 UTILIZADA')
ylim([-0.2 1.2])
grid
```

```
disp('PROGRAMA TERMINADO')
disp('IMPORTANTE: SE CREARON LOS ARCHIVOS
AislamientoBCepstrum.mat y STDAislamientosCepstrum.mat')
disp('Los tiene que copiar a la carpeta de Recopilacion')
pause
```

TÉCNICA MLS

calculohoptim.m y calniv2mics.m.- los mismos que en la técnica Cepstrum.

RastreoMLS2CH.m.

```
%Este programa muestra la respuesta impulsiva capturada en el archivo
test1.dat para
%que se aprecie cuantas muestras hay que recortar al principio de las señales
% y asi medir la energia de la secuencia sin retrasoscapturasion
%utilizando el archivo test1.dat generado
clc
clear
close all
load test11.dat
y1=test11(:,1);%Este vector contiene los datos del sonido transmitido
y2=test11(:,2);%Este vector contiene los datos del sonido incidente
y1=y1(1:length(y1));
y2=y2(1:length(y2));
fs=24000;
ts=1/fs;
ty=0:ts:(length(y2)-1)/fs;

Ddelay=0.35;%Distancia del altavoz al mic2 para compensar

subplot(2,1,1)
plot(y1)
hold on
plot(y2,'r'),grid
title('Respuestas impulsivas transmitida (azul) incidente (rojo)')
xlabel('Muestras')

subplot(2,1,2)
plot(ty,y1)
hold on
plot(ty,y2,'r'),grid
title('Respuestas impulsivas transmitida (azul) incidente (rojo)')
xlabel('Tiempo (s)')

%Calculando la energia total de la secuencia

Etot1=sum(y1.*y1)
Etot2=sum(y2.*y2)
```

AislamientoMLSsincamara1.m.

```
%Este programa calcula el aislamiento con MLS SIN camara de transmision
%utilizando el archivo test1.dat generado
clc
clear
close all
disp('ESTE PROGRAMA CALCULA EL AISLAMIENTO SONORO CON MLS
SIN CAMARA DE TRANSMISION')
disp('SE UTILIZA EL ARCHIVO RMLS.dat CON LAS RESPUESTAS
IMPULSIVAS CAPTURADAS')
disp('*****FILTRA LAS SEÑALES
VENTANEADAS*****')
disp('*')
disp('Para comenzar a realizar la medicion pulse ENTER')
pause

load test11.dat
y1=test11(:,1);%Este vector contiene los datos del sonido transmitido
y2=test11(:,2);%Este vector contiene los datos del sonido incidente
MuesQuitarInicio=115;%indicar a partir de cual muestra se va a considerar la
respuesta impulsiva
MuesalMax=0;%contar el numero de muestras que hay sin cotar la primera,
hasta el maximo
CerosMas=-66;%Ceros de mas o de menos al principio para la señal
transmitida para mover la ventana
UnosTW=-18;%agregar + o restar - unos a la ventana
dBmas=30;%Incerte valores multiples de 15 que indiquen la diferencia de
sensibilidad en los canales
CompensaciondB=10+dBmas;%Incerte valores multiples de 10 que indiquen la
diferencia de sensibilidad en los canales

y1=y1(MuesQuitarInicio:length(y1));
y2=y2(MuesQuitarInicio:length(y2));

fs=44100;
ts=1/fs;
ty=0:ts:(length(y2)-1)/fs;

T=27;%TEMPERATURA AMBIENTE
X=0.58;%Altura del mic1
lm=1.23;%Ato del especimen
c=331.5+0.607*T;

razon=0.01;%Ratio para la ventana tukey para aplicar en el tiempo a la señal
transmitida
razon2=0.01;%Ratio2 para la ventana tukey2 para aplicar en el primer reflejo
ymin=10;%Limites de dB para los resultados finales
ymax=64;
```



```

tmic2=dmic2/c;%Tiempo de llegaga del sonido directo
muestrasdirecto=floor(tmic2*fs);

dREF=2*sqrt((dmic2/2)^2+X^2);%Distancia recorrida por la reflexion del piso
tREF=dREF/c;%Tiempo de llegada del primer reflejo del piso
muestrasreflejo=floor(tREF*fs);%Muestras del primer reflejo del piso

tretmic2=tREF-tmic2;%Retraso del sonido reflejado
muestrasretraso=muestrasreflejo-muestrasdirecto;%Muestras que se tarda en
llegar el sonido reflejado

ddifrac=sqrt((dllegada+0.5*anchomuestra)^2+(lm-X)^2)+sqrt((lm-
X)^2+(dmic1+0.5*anchomuestra)^2);%Distancia recorrida por la difraccion
tdifrac=ddifrac/c;%Tiempo que ocupa en llegar la difraccion
mdifrac=floor(tdifrac*fs);%Muestras que se toma la difraccion en llegar
mretdifrac=mdifrac-muesbocinamuestramic1;%Muestras de retraso que tiene el
sonido difractado respecto al transmitido
mretdifracCOM=mretdifrac-10;%Le resto unas 10 muestras para que el
cepstrum rectificado tenga
%unos 10 ceros antes de que la primera difraccion se presente.
%unos 10 ceros antes de que la primera difraccion se presente.

BdeF=[100 125 160 200 250 315 400 500 630 800 1000 1250 1600 2000
2500 3150 4000 5000];
nt=length(BdeF);%Se van a tener 18 tercios de octava de 100Hz hasta 5KHz

%FABRICACION DE VENTANA TUKEY2 PARA LA SEÑAL INCIDENTE
%*****
disp('Para ventanear la señal Incidente, pulse ENTER')
pause
%tw2=tukeywin((muesbocinamuestramic1-muestrasdirecto)*2,razon2);%su
tamano debe se el doble de mic2-muestra
%tw2=tukeywin(muestrasretraso-MuesalMax-1,razon2);%su tamano debe se el
doble de mic2-muestra
tw2=tukeywin(muestrasretraso+UnosTW,razon2);%su tamano debe se el doble
de mic2-muestra
cerosfinales=zeros(1,length(y1)-length(tw2));%ojo AQUI SE LE AGREGO UN
VALOR MAS PARA QUE CUADRE
vfintukey2=[tw2' cerosfinales];

%GRAFICADO DE LA VENTANA TUKEY2 SOBRE LA SEÑAL INCIDENTE
DETECTADA POR EL MIC2
figure
subplot(2,1,1)
plot(vfintukey2,'r')
hold on
plot(vfintukey2,'.r')
hold on

```



```

plot(y2,'b')
title('VENTANA TUKEY2 Y LA SEÑAL MLS INCIDENTE PROPORCIONAL
f(muestras)')
xlabel('Muestras')
grid

```

```

subplot(2,1,2)
plot(ty,vfintukey2,'r')
hold on
plot(ty,vfintukey2,'.r')
hold on
plot(ty,y2/max(y2),'b')
title('VENTANA TUKEY2 Y LA SEÑAL MLS INCIDENTE NORMALIZADA f(t)')
xlabel('Tiempo (s)')
grid

```

```

ysumaventukey2=y2'.*vfintukey2;%VENTANEADO DE LA SEÑAL CON
REFLEJOS COMPLETO CON TUKEY2

```

```

%CALCULO PARA LA OBTENCION DE LA SENAL DIRECTA (extraerla)
Sdirecta=ysumaventukey2(1:length(tw2));
tSdirecta=0:ts:(length(Sdirecta)-1)*ts;
%CALCULO DEL RMS PARA LA SEÑAL VENTANEADA RECABADA POR EL
MIC2

```

```

Emic2Global=sum(Sdirecta.*Sdirecta);%E mic2 sonido incidente DIRECTO
Ventaneado

```

```

%TERMINO DEL VENTANEADO DE LA SEÑAL INCIDENTE
%////////////////////////////////////

```

```

%****FABRICACION DE VENTANA TUKEY1 PARA APLICAR A LA SEÑAL
TRANSMITIDA CAPTURADA POR EL MIC1*****
disp('*')
disp('Para ventanear la señal Transmitida, pulse ENTER')
pause

```

```

K=length(tw2);
tamanoventana=K;%Es de tamaño K-1 porque en la muestra K se presenta la
espiga por lo que si la ventana tiene una muestra menos, entonces se evitara a
toda costa la presencia de de ella.

```

```

cerosantes=zeros(1,muesbocinamuestramic1-
muestrasdirecto+MuesalMax+CerosMas);%Estos ceros deben estar antes que
la ventana comience ya que es el tiempo que tardara en llegar la transmision
tw=tukeywin(tamanoventana,razon);
cerosdespues=zeros(1,length(y1)-(tamanoventana)-length(cerosantes));%Estos
ceros son para contruir la ventana para aplicar ala señal que tiene todos los
reflejos

```

```

venttukeycerostukeyceros=[cerosantes tw' cerosdespues];
vfintukey=venttukeycerostukeyceros;%Ventana tukey final para aplicarse al la
señal original
%vfintukeyceps=[tw' cerosdespues2];%Ventana tukey final para aplicarse al
cepstrum

```

%GRAFICADO DE LA VENTANA TUKEY SOBRE LA SEÑAL A VENTANEAR

```

figure
subplot(2,1,1)
plot(vfintukey,'r')
hold on
plot(vfintukey,'.r')
hold on
plot(y1,'b')
title('VENTANA TUKEY Y LA SEÑAL MLS TRANSMITIDA f(muestras)')
xlabel('Muestras')
grid

```

```

subplot(2,1,2)
plot(ty,vfintukey,'r')
hold on
plot(ty,vfintukey,'.r')
hold on
plot(ty,y1/max(y1),'b')
title('VENTANA TUKEY Y LA SEÑAL TRANSMITIDA NORMALIZADA f(t)')
xlabel('Tiempo (s)')
grid

```

ysumaventtukey=y1'.*vfintukey;%VENTANEO DE LA SEÑAL TRANSMITIDA
CON REFLEJOS COMPLETO CON TUKEY

%*****PROCESO PARA CORTAR LA PARTE DE LA SEÑAL
TRANSMITIDA*****

```

ysventv=ysumaventtukey(muesbocinamuestramic1-
muestrasdirecto+MuesalMax+CerosMas+1:muesbocinamuestramic1-
muestrasdirecto+MuesalMax+CerosMas+tamanoventana);%Parte de la onda
recuperada del vetaneo
tysventv=0:ts:(length(ysventv)-1)*ts;%Vector de tiempo construido para la señal
recuperada

```

%CALCULANDO EL RMS DE LA SEÑAL RECUPERADA (transmitida) ysventv

Emic1Global=sum(ysventv.*ysventv);%E mic1 (TODO JUNTO) se van
acumulando en la posicion J.

Emic1Global=sum(ysumaventtukey.*ysumaventtukey);%E mic1 (TODO
JUNTO) se van acumulando en la posicion J.

%*****

%*****

%*****

```

%*****
%*****
%*****
%ysventv=señal recuperada transmitida
%Sdirecta=Señal recuperada incidente

%SE PROCEDE CON EL FILTRADO DE LAS SEÑALES RECUPERADAS
disp('Para filtrar las señales ventaneadas pulse ENTER')
pause

for M=1:nt
    f = BdeF(M);
    wn = [round((f)*2^(-1/6)) round((f)*2^(1/6))];%Se establecen las
frecuencias de corte
    [b,a] = butter(3,wn/fs*2);%filtro fabricado
    filtrada1= filter(b,a,ysventv);
    filtrada2= filter(b,a,Sdirecta);
    Ctexto1=sprintf('Analizando Frecuencia %2d Hz.',f);
    disp(Ctexto1)

figure
subplot(2,1,1)
plot(tysventv,filtrada1,'r')
title(['SEÑAL MLS FILTRADA POR MIC1 TRANSMITIDA (',num2str(f),' Hz)'])
xlabel('Tiempo (s)')
grid

subplot(2,1,2)
plot(tSdirecta,filtrada2,'r')
title(['SEÑAL CAPTURADA FILTRADA POR MIC2 INCIDENTE (',num2str(f),'
Hz)'])
xlabel('Tiempo (s)')
grid

%CALCULANDO EL RMS DE LA SEÑAL RECUPERADA (transmitida) ysventv

Emic1(M)=sum(filtrada1.*filtrada1);%E mic1 en la frecuencia M.
Emic2(M)=sum(filtrada2.*filtrada2);%E mic2 en la frecuencia M.

    disp('PULSE ENTER PARA SEGUIR PROCESANDO LA SIGUIENTE
FRECUENCIA')

disp('*****
****')
    disp('*')
    pause(0.1)

end%Este end corresponde al final de cada promedio que se hace. Hay que
ver si es necesario cambiarlo en el futuro

```

```

disp('EL FILTRADO DE LAS SEÑALES YA SE CONCLUYO PULSE ENTER
PARA MOSTAR LOS RESULTADOS FINALES')
disp('*')
pause
clc

dBcaida=20*log10(dpartida/dllegada);
FraccionCaida=10^(dBcaida/20);
Emic2Caido=Emic2*FraccionCaida;

%CALCULO DEL AISLAMIENTO Y PRELIMINARES
AislamientoB=10*log10(Emic2Caido./Emic1)+CompensaciondB;%Es energia
%clc

%*****CALCULANDO STC*****
ValorSTC=STCf(AislamientoB);
disp('LOS RESULTADOS FINALES SON:')
STCtexto=sprintf('STC Banda 100-5KHz= %3.1f dB.',ValorSTC);
disp(STCtexto)
disp('*****')
Textocaida=sprintf('La caida del mic2 a la muestra es de %2.1f dB',dBcaida);
disp(Textocaida)
disp('*****')

tablavalores=[BdeF' Emic1' Emic2' Emic2Caido' AislamientoB'];
TV=num2str(tablavalores,'%15.1f');

disp('Frecuencia      Emic1      Emic2      Emic2caido      Aislamiento dB
(10log(Emic1/Emic2caido))')
disp(TV)
disp('*****')
disp('TERMINO DE PROCESAMIENTO')
disp('*')
disp('PULSE ENTER PARA VER LA GRAFICA DE VALORES FINALES')
pause

figure
semilogx(BdeF,AislamientoB,'LineWidth',3)
hold on
semilogx(BdeF,AislamientoB,'*r','LineWidth',2)
ylabel('AISLAMIENTO SONORO R (dB)')
title(['AISLAMIENTO SONORO          TECNICA  MLS          STC  =
',num2str(round(10*ValorSTC)/10),' dB  PANEL "A"'])
GraticulaGeneral(ymin,ymax)

AislamientoBMLSsincamara=AislamientoB';%Igualeado para salvarlo sepues
con otro nombre
save AislamientoBMLSsincamara.mat AislamientoBMLSsincamara

```

```
disp('PULSE ENTER PARA VER LAS VENTANAS UTILIZADAS')
pause
```

```
disp('IMPORTANTE: SE CREO EL ARCHIVO
AislamientodBMLsincamara.mat')
disp('Lo tiene que copiar a la carpeta de Recopilacion')
pause
```

```
%GRAFICANDO LAS VENTANAS KAISER POR PROCESAR
```

```
figure
subplot(2,1,1)
plot(tw,'.-')
title('VENTANA TUKEYWIN1 UTILIZADA (PARA LA SEÑAL TRANSMITIDA)')
ylim([-0.2 1.2])
grid
```

```
subplot(2,1,2)
plot(tw2,'.-')
title('VENTANA TUKEYWIN2 UTILIZADA (PARA LA SEÑAL INCIDENTE)')
ylim([-0.2 1.2])
grid
```