

UNIVERSIDAD POLITECNICA DE VALENCIA

ESCUELA POLITECNICA SUPERIOR DE GANDIA

Master en Ingeniería Acústica



UNIVERSIDAD
POLITECNICA
DE VALENCIA



ESCUELA POLITECNICA
SUPERIOR DE GANDIA

**“Simulador de recintos
tridimensionales para el análisis del
campo acústico en algoritmos de
procesado de la señal”**

TESIS DE MASTER

Autor:

Jorge Machín Mindán

Director:

D. Miguel Ferrer Contreras

GANDIA, 2010

ÍNDICE

RESUMEN	3
I. OBJETIVOS.....	4
II. INTRODUCCIÓN	4
III. DECISIONES	5
III.1. SIMULACION FDTD.....	5
III.2. CONDICIONES DE CONTORNO	6
III.3. PASO TEMPORAL.....	6
III.4. CREACIÓN DE LAS VARIABLES Y MATRICES	7
III.5. SEÑALES DE EXCITACIÓN.....	7
III.6. FUENTES DE RUIDO	8
III.7. ALGORITMO PARA EL CAR	8
III.8. RESPUESTA AL IMPULSO	9
IV. CREACIÓN DEL INTERFAZ.....	10
IV.1. INTERFAZ 2 D	10
IV.2. INTERFAZ 3 D	11
IV.3. RESTRICCIONES	11
V. SIMULACIONES EN DOS DIMENSIONES.....	13
V.1. ESTUDIO DEL CORRECTO FUNCIONAMIENTO DEL SISTEMA.....	13
V.2. ESTUDIO DE LA INFLUENCIA DEL NÚMERO DE FUENTES Y SENSORES	15
V.3. EXPRESIÓN DE LOS RESULTADOS	19
V.4. VARIACIÓN DEL ALFA (ABSORCIÓN DEL PML).....	20
V.5. IDEA: ENCERRAR LA FUENTE DE RUIDO	22
V.6. SIMULACIÓN MULTIFUENTE.....	24
VI. SIMULACIONES EN TRES DIMENSIONES	26
VI.1. COMPARACIÓN CON 2 D	26
VI.2. CAR EN TRES DIMENSIONES	27
VII. CONCLUSIONES	35
VIII. LÍNEAS FUTURAS.....	36
IX. BIBLIOGRAFÍA	37
ANEXO I. DEFINICIÓN DE VARIABLES.....	38
ANEXO II. EJEMPLO.....	40
ANEXO III. GUÍA PARA USUARIO.....	44

Simulador de recintos tridimensionales para el análisis del campo acústico en algoritmos de procesamiento de la señal.

Autor: Jorge Machín Mindán

Director: Miguel Ferrer Contreras

RESUMEN

Por todos es conocido que el uso de Finite-Difference Time-Domain (FDTD) es cada vez más usado para las simulaciones acústicas, tanto en el caso de dos como en el tres dimensiones. Por lo que, en este proyecto se pretende diseñar una herramienta para poder utilizar esta técnica, y comprobar el correcto funcionamiento de diferentes algoritmos de procesamiento de la señal, como pueden ser el control activo de ruido (CAR) y/o Wave Field Synthesis (WFS). La técnica FDTD, al tratarse de una simulación en tiempo, nos da información del campo acústico resultante en cada instante, por lo que nos proporciona una idea del comportamiento de los algoritmos y de la sala tanto en el transitorio como en el estacionario (del filtro adaptativo).

Las comprobaciones que se va a llevar a cabo en este trabajo van a ser:

- Robustez y facilidad de manejo de las herramientas diseñadas.
- El Estudio del comportamiento de los algoritmos en el estacionario, el cual nos da información del campo acústico resultante, y así poder ser comparado con la simulación sin el algoritmo.

ABSTRACT

Everyone knows that the use of Finite-Difference Time-Domain (FDTD) is increasingly used for acoustic simulations both in the case of two as in three dimensions. Therefore, the objectives of this project is the realization of an interface for the use of this technique and verify the correct operation of different signal processing algorithms, such as active noise control (ANC) and Wave Field Synthesis (WFS). FDTD technique give us information of the acoustics field at any moment, by giving us information about the behavior of the algorithms and the room in the transient and in the steady-state (filter).

Checks to be carried out in this work will be:

- Robustness and ease of management tools designed.
- The Study of the behavior of algorithms in the steady-state, which gives us the sound field information, so they can be compared to the simulation without the algorithm.

I. OBJETIVOS

El objetivo de esta tesina es doble: por un lado se pretende diseñar una herramienta que permita simular la propagación del campo acústico en un recinto tridimensional, una vez definida la geometría del recinto y las características de las discontinuidades (paredes), y por el otro lado incluir el procesado discreto de diferentes algoritmos de Control activo de ruido y/o Wave Field Synthesis que gobiernen la generación de señales acústicas en cualquier punto del recinto simulado.

De esta forma, se podrá analizar el campo acústico existente en cualquier plano del recinto (o incluso en cualquier región tridimensional), no sólo ante determinadas señales que se propagan de forma libre en el recinto sino también cuando estos algoritmos están en funcionamiento.

II. INTRODUCCIÓN

Esta tesina tenía como objetivo la creación de un interfaz para la simulación de la propagación acústica, y fue por ello por lo que la técnica elegida para esta tipo de simulación fue Finite Difference Time Domine (FDTD), una técnica cada vez más utilizada en acústica.

Elegimos esta técnica porque nos permite visionar como actúan los diferentes algoritmos aplicados con el paso del tiempo. Es interesante ver como varía la presión sonora en cada instante de tiempo para comprender mejor los algoritmos de control activo de ruido.

Sin embargo esta técnica también tiene un gran inconveniente, y es el gran coste computacional que conlleva este tipo de simulación, sobretodo en el caso de tres dimensiones. Es por esto, por lo que en este proyecto se ha decidido crear un interfaz para la simulación en dos dimensiones como una primer aproximación al problema.

Por otro lado tenemos el control activo de ruido (CAR), el cual consiste en reducir el ruido, presente en un punto o zona, a partir de una señal (en contrafase al ruido) generada electrónicamente. Se basa en el principio de superposición de campos sonoros, más concretamente en el de interferencia destructiva.

III. DECISIONES

A lo largo de todo el desarrollo del proyecto se ha visto la necesidad de tomar varias decisiones, a continuación se explican alguna de las más importantes.

III.1. SIMULACION FDTD

Al tratarse de una simulación acústica debemos utilizar la ecuación de ondas para obtener el nivel de presión en cada punto. A continuación vamos a explicar brevemente las ecuaciones y simplificaciones elegidas para llevar a cabo nuestras simulaciones.

Ecuación de ondas en dos dimensiones[1]:

$$\begin{aligned}\frac{\partial p}{\partial x} + \rho \frac{\partial u_x}{\partial t} &= 0 \\ \frac{\partial p}{\partial y} + \rho \frac{\partial u_y}{\partial t} &= 0 \\ \frac{\partial p}{\partial t} + \rho c^2 \left(\frac{\partial u_x}{\partial x} + \frac{\partial u_y}{\partial y} \right) &= 0\end{aligned}\tag{1}$$

Donde:

- p, presión (escalar)
- x, coordenada espacial x
- y, coordenada espacial y
- u_x, velocidad de la partícula en el eje x (vector)
- u_y, velocidad de la partícula en el eje y (vector)
- t, es el tiempo
- c, velocidad de propagación del sonido en el aire
- ρ, es la densidad del aire

Para la resolución de las ecuaciones anteriores aproximamos las primeras derivadas tanto en tiempo como en espacio, por el cociente de la diferencia entre muestras espaciales y el incremento espacial (derivada primera en espacio). La derivada en tiempo se aproxima de la misma manera, pero el incremento del denominador es el temporal y la diferencia es entre 2 instantes de tiempo consecutivos en el mismo punto.

Expresión de la aproximación utilizada para la solución de las derivadas parciales:

$$\text{Derivada en espacio} \quad \left. \frac{\partial f}{\partial x} \right|_{x=x_0} = \frac{f(x_0 + \frac{1}{2} \Delta x) - f(x_0 - \frac{1}{2} \Delta x)}{\Delta x} \quad (2)$$

$$\text{Derivada en tiempo} \quad \left. \frac{\partial f}{\partial t} \right|_{t=t_0} = \frac{f(t_0 + \frac{1}{2} \Delta t) - f(t_0 - \frac{1}{2} \Delta t)}{\Delta t} \quad (3)$$

El método elegido, FDTD, calcula los valores en instantes alternados; calculamos u_x y u_y en el mismo instante mientras que “p” la calculamos en otro instante, con la finalidad de conseguir un error de segundo orden debido a las aproximaciones utilizadas en las derivadas temporales. Este cálculo se realiza dentro de un bucle, en cada interacción aumentamos el paso correspondiente a un incremento temporal.

Expresiones a resolver en cada iteración [2]:

$$\text{Presión:} \quad p(i, t + \frac{\Delta t}{2}) = p(i, t - \frac{\Delta t}{2}) - \Delta t \rho c^2 \frac{u(i + \frac{1}{2}, t) - u(i - \frac{1}{2}, t)}{\Delta x} \quad (4)$$

$$\text{Velocidad:} \quad u(i + \frac{1}{2}, t + \Delta t) = u(i + \frac{1}{2}, t) - \frac{\Delta t}{\rho} \frac{p(i + 1, t + \frac{\Delta t}{2}) - p(i, t + \frac{\Delta t}{2})}{\Delta h} \quad (5)$$

Este vector velocidad se resuelve dos veces, una para cada eje (x e y).

III.2. CONDICIONES DE CONTORNO

Se decidió introducir al final de nuestro dominio de simulación unas condiciones de contorno absorbentes, para ver las diferencias entre distintas absorciones en las paredes. Es por lo que se eligió que la caracterización de las discontinuidades (paredes) fueran definidas mediante Perfectly Matched Layer (PML) [3] [4].

A su vez también se decidió tomar la condición de contorno, en la paredes, en la que la velocidad de las partículas en la superficie sea nula ($u_i = 0$), donde \bar{i} es perpendicular a la dirección del contorno [5].

III.3. PASO TEMPORAL

En nuestras simulaciones también se decidió utilizar un número de Courant (CFL) igual a 1 (si se desea utilizar un incremento temporal distinto, sólo hay que rellenar la casilla de Incremento Temporal con el valor deseado).

El número de Courant (CFL) debe ser menor o igual a la unidad para que la simulación sea estable, ya que se trata de una condición de convergencia de ecuaciones diferenciales en derivadas parciales. Como consecuencia de esta condición, el paso temporal debe ser inferior a un cierto valor, de lo contrario la simulación producirá resultados incorrectos.

III.4. CREACIÓN DE LAS VARIABLES Y MATRICES

Una vez tomadas las decisiones correspondientes al PML y a las condiciones de contorno pasamos a definir las variables y las matrices.

El primer paso es definir el contorno con los puntos cargados en la GUI, dicha interface será diseñada en Matlab (más adelante se explicará como esta diseñada). Con estos puntos definiremos una matriz que guarde los contornos. Será esta matriz la que aplicaremos como máscara a la hora de la simulación, la cual nos marcará las condiciones de contorno (velocidad de las partículas nula).

A la vez que definimos esta máscara también haremos lo propio con las discontinuidades (las paredes), las cuales serán definidas mediante PMLs.

Otra decisión importante es la de utilizar, por defecto, un incremento temporal tal que el número de Courant sea 1. No obstante, como se ha comentado antes, si se desea utilizar un incremento temporal menor, este se puede utilizar rellenando el hueco correspondiente a "Incremento Temporal" en el interfaz. La explicación de cómo se realizan los pasos anteriores esta explicado en el anexo I.

III.5. SEÑALES DE EXCITACIÓN

Con el fin de poder realizar diferentes simulaciones se ha decidido elegir varios tipos de señales de excitación, estas señales son aplicables a cada una de las fuentes de ruido:

- Señal senoidal. Tono puro (frecuencia fundamental definida por el usuario).
- Señal ricker. Excitación Ricker, frecuencia de corte definida por el usuario.
- Ruido filtrado. El ruido generado tiene una función de probabilidad Gaussiana con una densidad de potencia constante hasta la frecuencias de corte (W_n) definida por el usuario, la media de esta señal es cero y su desviación típica uno.

Todas las fuentes de ruido blanco que emitan a la vez tienen la misma señal. Esto nos permite aproximar las superficies que generen sonido mediante fuentes puntuales emitiendo en fase.

- Archivo wav. Para este paso hay que realizar un remuestreo de la señal, para que la frecuencia de muestreo de dicha señal coincida con la inversa del incremento temporal de la simulación.

III.6. FUENTES DE RUIDO

En primer lugar debemos comentar que en nuestras simulaciones usamos fuentes puntuales. Por todos es conocido que estas fuentes se caracterizan por comportarse como fuentes omnidireccionales. Sabemos que esto no es real, pero lo podemos considerar como una aproximación.

Este “problema” se podría solucionar definiendo las fuentes como pistones planos (formados por un conjunto de fuentes puntuales).

III.7. ALGORITMO PARA EL CAR

En esta tesina se decidió implementar el sistema de Control Activo de Ruido basado en el algoritmo LMS con la estructura de filtrado-x, ya que se trata de uno de los más usados para este fin [6] [7] [8].

Este algoritmo nos permite utilizar filtros adaptativos, es decir, que se recalculen sus coeficientes en cada interacción del algoritmo. Dichos coeficientes cambian conforme varía la señal de error (los sensores en nuestro caso) intentando que la potencia de la señal de error sea mínima, en nuestro caso la señal de error es la señal en los sensores.

Una vez decidido ya el algoritmo a utilizar se consideró la posibilidad de aplicarlo con varias fuentes secundarias y varios sensores a la vez, pero sólo con una señal de referencia (una única fuente de ruido). Después de diseñar este algoritmo se vieron otras posibles aplicaciones, como se verá más adelante, por lo que se decidió realizar también el algoritmo con varias señales de referencia.

Para el correcto funcionamiento de la simulación deben calcularse tantos filtros adaptativos como pares de señal de referencia – fuentes secundarias haya en la simulación.

Hay varios métodos de calcular los nuevos coeficientes del filtro adaptativo, pero en nuestro caso se ha elegido el cálculo de los nuevos coeficientes como la media de todas las señales de error (e_k) por su correspondiente señal de referencia filtrada (xf_{ijk}) [6].

Los nuevos coeficientes del filtro se calculan conforme a la siguiente expresión:

$$w_{ij} = w_{ij} - 2 * \mu * (xf_{ij1} * e_1 + xf_{ij2} * e_2 + \dots + xf_{ijK} * e_k) \quad (6)$$

A su vez se generarán tantas señales anti-ruido como fuentes secundarias haya (j):

$$y_j = x_1 * w_{1j} + x_2 * w_{2j} + \dots + x_I * w_{Ij} \quad (7)$$

Tendremos tantas señales de referencia como fuentes de ruido haya (i):

$$x_i$$

A su vez estas señales de referencia serán filtradas por las plantas acústicas $B_{i(j,:)}$:

$$xf_{ijk} = x_i * B_{ij} \quad (8)$$

Y se monitorizará la señal de error (señal a minimizar) en tantos puntos como sensores haya (k):

$$e_k$$

Otra decisión, casi obligatoria, es que el usuario pueda elegir el paso de convergencia (μ), que como hemos visto anteriormente es un parámetro que regula la velocidad de convergencia del filtro adaptativo. La elección de este parámetro tiene especial interés ya que es el que puede hacer que el sistema converja lentamente o que incluso se vuelva inestable.

La señal (o señales) de referencia que obtenemos para el control activo de ruido suponemos que se toma de la fuente de ruido de una forma mecánica (acelerómetro o similar), por lo que no se trata de un sensor acústico (micrófono). Esto se hace así para evitar retroalimentaciones acústicas en nuestro sistema.

III.8. RESPUESTA AL IMPULSO

Una vez ya elegido el algoritmo para el CAR, ya conocemos los cálculos que debemos realizar, y uno de estos cálculos es el de las plantas, o caminos acústicos.

Como ya se ha comentado, el algoritmo LMS con filtrado x necesita, para su correcto funcionamiento, de tantas plantas acústicas como pares fuente-sensor haya:

Las plantas relacionadas con cada fuente secundaria son guardadas como una variable nueva (B_i), mientras que cada sensor se guarda como una fila diferente $B_{i(j,:)}$.

La señal elegida para obtener estas plantas acústicas es un impulso. Una vez emitido este impulso y guardado las señales recibidas en los sensores durante la simulación, ya tenemos la información necesaria para describir el camino acústico.

El número de muestras de la planta es definido por el usuario para buscar una buena "relación" entre fiabilidad y tiempo de cálculo. El tamaño de la planta también determina cuando el CAR empieza a funcionar, es decir, el CAR tiene que almacenar tantas muestras como tamaño tengan los filtros para poder realizar sus cálculos.

Todas las decisiones anteriores son extrapolables para el caso de simulaciones en tres dimensiones.

IV. CREACIÓN DEL INTERFAZ

Teniendo presente el objetivo de esta tesina, se planteo la creación de dos interfaces diferentes, una para las simulaciones en dos dimensiones, que nos serviría para obtener una primera idea a cerca de los resultados, y la otra para las simulaciones en tres dimensiones.

Como hemos comentado antes, estas herramientas serán dos interfaces gráficas diseñadas en Matlab, por lo que los diferentes algoritmos y la FDTD también están programadas en dicho lenguaje.

IV.1. INTERFAZ 2 D

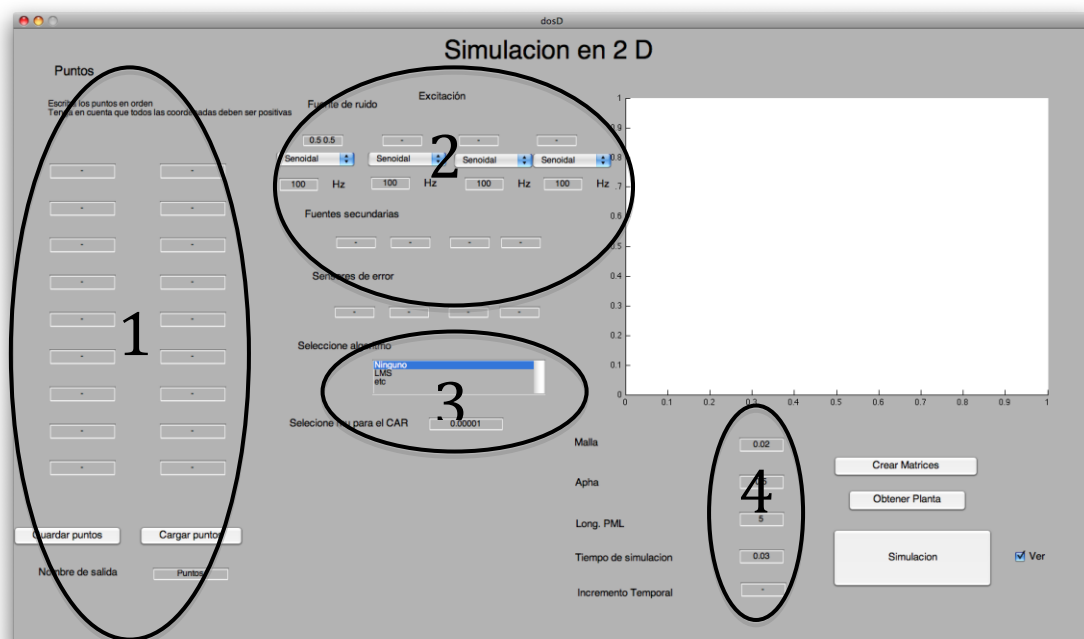


Fig.1. Vista de la GUI para la simulación en dos dimensiones.

Esquema del interfaz:

- 1 Definición de los puntos del contorno.
- 2 Elección de la posición de las fuentes y de los sensores.
- 3 Elección del algoritmo (tipo de simulación que se va a realizar: si se va aplicar o no algún tipo de procesado).
- 4 Elección de parámetros para la construcción de las matrices para la simulación (Alpha, longitud del PML, tamaño de la malla...).

IV.2. INTERFAZ 3 D

La interfaz para la simulación de tres dimensiones sigue la misma estructura que la de dos dimensiones, la única variación es que la definición de las diferentes variables se realizan en varias pestañas, dentro de la misma GUI.

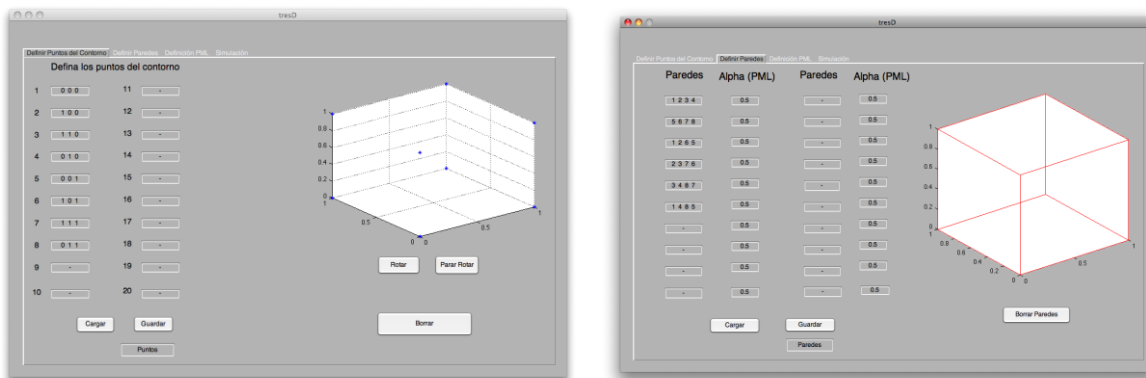


Fig. 2 y 3. Pantallas de la interfaz de tres dimensiones para la definición del contorno (puntos del contorno y definición de las paredes).

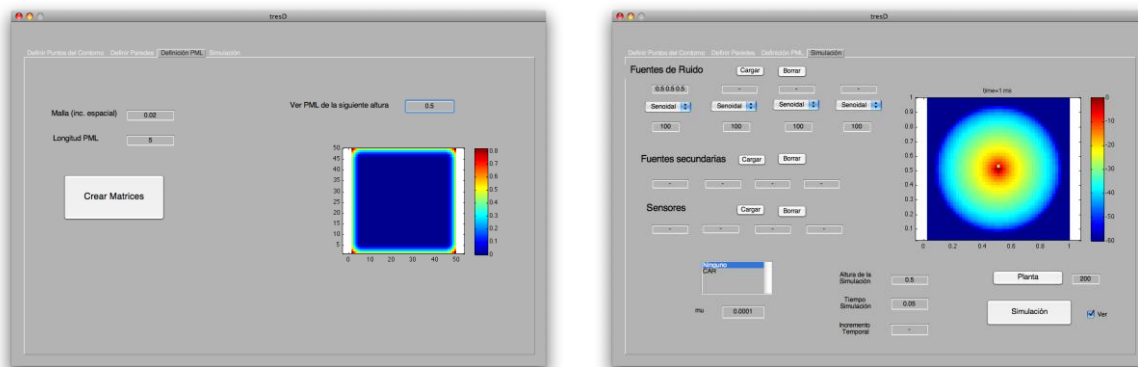


Fig. 4 y 5. Pantallas de la interfaz de tres dimensiones. Fig.3 para la definición del PML y su posible visualización. Fig.4. Elección del algoritmo, posicionamiento de fuentes y sensores y posible visualización.

IV.3. RESTRICCIONES

A continuación vamos a comentar las restricciones con las que cuentan nuestras interfaces actualmente. Estas restricciones son debidas a las decisiones que se han tomado a lo largo del proyecto, y muchas de ellas son fácilmente solucionables.

Definición en orden de los puntos de contorno (creación del perímetro en orden sólo en el caso 2D, se debe a que a la vez que se definen los puntos, también se definen las paredes). Otra restricción a la hora de definir los puntos del contorno es que sólo podemos definir puntos con coordenadas positivas.

Autor: Jorge Machín Mindán, email: jormacmi@epsg.upv.es

Fecha de entrega: 16-10-2010

Creación sólo de paredes cuadradas (no paredes inclinadas, ni más de cuatro lados). → Se puede solucionar creando una función de interpolación para la definición de este tipo de paredes.

Definición, en orden, de los vértices de las paredes, es decir, todos los vértices tienen que estar contiguos. Esta restricción está sólo presente en el caso de tres dimensiones.

Definición de un número limitado de puntos del contorno mediante la GUI. → Mediante la carga de un fichero se pueden definir más puntos que los huecos de la GUI.

Creación de paredes internas. → Este propósito no ha sido contemplado a la hora de diseñar la GUI, pero se soluciona, fácilmente, creando una pantalla más dentro de la GUI para este propósito (de forma similar a la ya existente).

V. SIMULACIONES EN DOS DIMENSIONES

A continuación se muestran varios estudios que se realizaron mediante simulaciones en dos dimensiones, para ver el correcto funcionamiento tanto de la interfaz como de los algoritmos de CAR.

V.1. ESTUDIO DEL CORRECTO FUNCIONAMIENTO DEL SISTEMA

El primer paso para comprobar si el funcionamiento del algoritmo del CAR es correcto es observar las señales presentes en los sensores, y si realmente se está reduciendo el ruido en la sala o no.

Otro estudio importante es el de la convergencia de los filtros, para ello observamos el comportamiento de los coeficientes con el paso del tiempo (habrá tantos filtros adaptativos como pares fuentes secundarias – señales de referencia haya).

Esta primera simulación se llevó a cabo en una sala cuadrada de escasas dimensiones, con una fuente de ruido emitiendo una señal senoidal de 100 Hz, el sistema CAR utilizado consta de cuatro fuentes secundarias cuatro sensores de error.

En primer lugar observamos que al emitir una señal senoidal en nuestro espacio de simulación, el sistema CAR actúa y obtenemos las siguientes grabaciones (en la posición de los sensores).

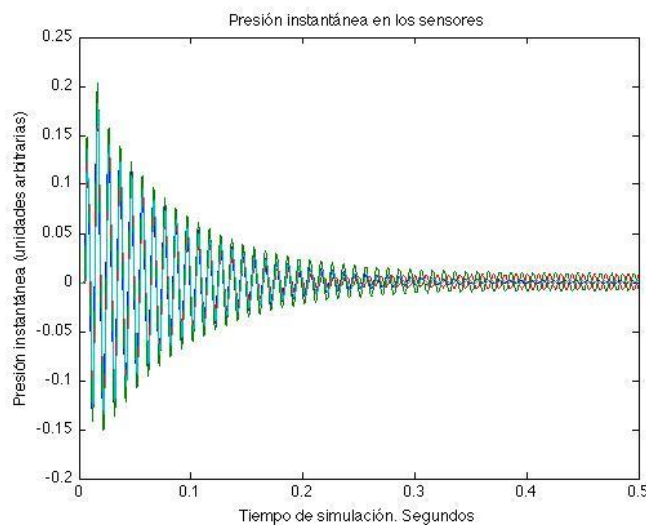


Fig.6. Señal grabada en los sensores.

Como se observa, al comienzo la señal es elevada y rápidamente el nivel de ruido baja. Esto es debido a que el CAR funciona correctamente.

Para comprobar el funcionamiento se decidió mostrar la diferencia entre las potencias instantáneas de la simulación con el sistema y sin él (en un de los sensores).

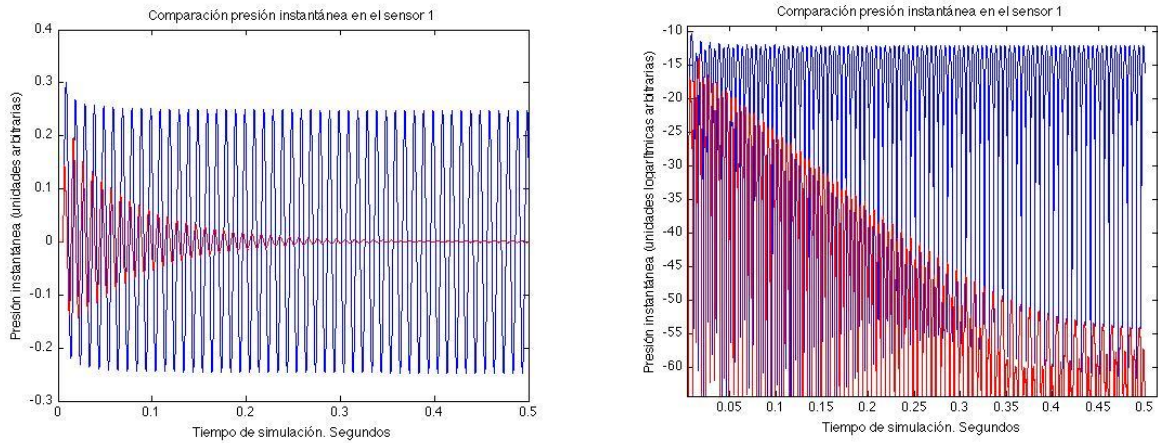


Fig.7 y 8. Comparación entre las señales grabadas en el CAR y sin él, en los sensores de error.

Se observa como la señal en el sensor es muy inferior a la señal que habría en ese mismo punto de no estar conectado el CAR.

Este hecho se aprecia mejor en la imagen de la derecha, la cual representa la misma señal pero con la presión instantánea en escala logarítmica (arbitraria). En ella podemos ver como la potencia instantánea de la simulación, en los sensores, disminuye notablemente, pasando de valores de -10 dB a valores de -45 dB en 300 ms, y este decaimiento sigue por encima del medio segundo.

Otra parte del estudio consiste en observar las señales que son emitidas para anular al ruido, estas señales deben tener la misma frecuencia que la fuente de ruido y además también nos ayudan a ver cuando se llega al estacionario, que llegará cuando esta señal se estabilice.

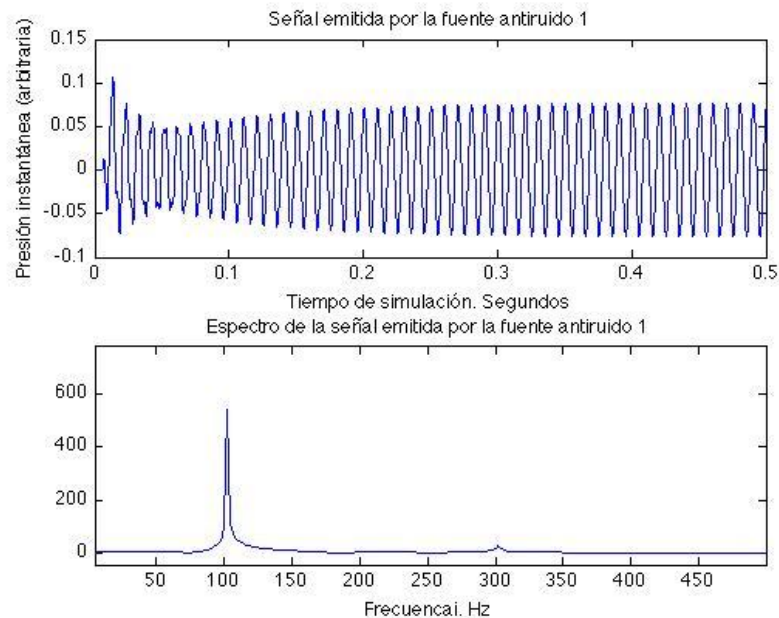


Fig.9. Información de la señal antiruido emitida por el CAR.

Se puede observar, que la señal que emitimos es de la misma frecuencia que el ruido a eliminar.

Y por último, para comprobar el correcto funcionamiento del CAR debemos observar que los coeficientes de sus filtros adaptativos convergen hacia la solución.

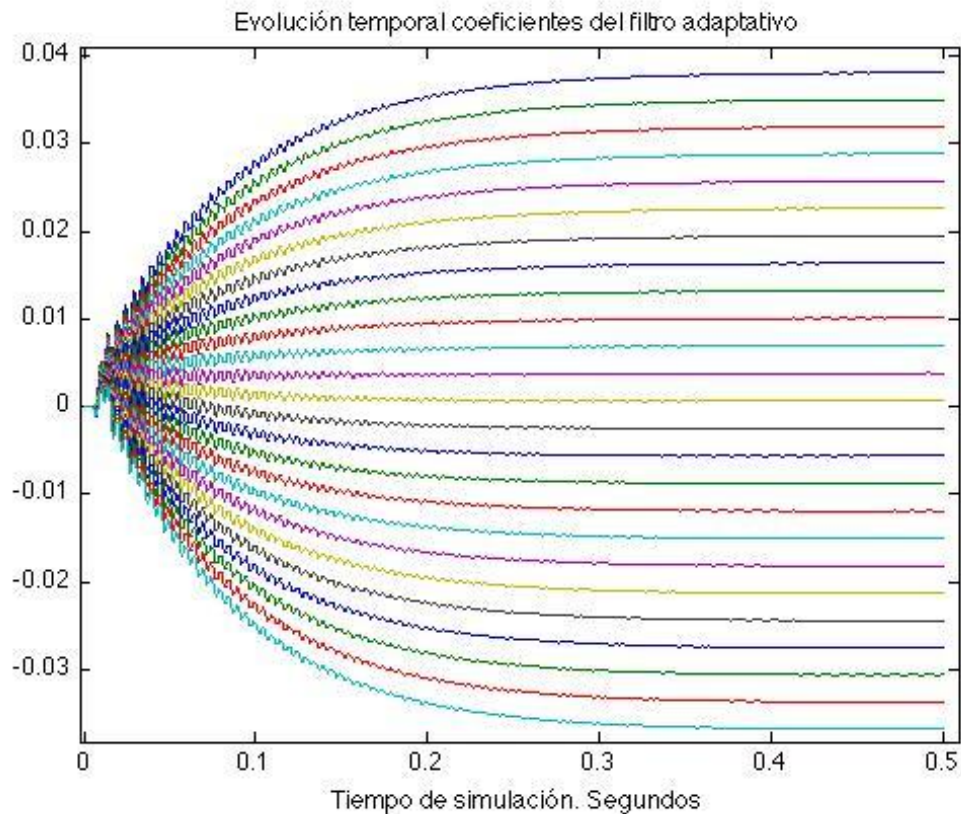


Fig.10. Evolución temporal de los coeficientes del filtro adaptativo (CAR).

Podemos apreciar que el filtro ha encontrado su solución, vemos como a partir de los 400 ms, los coeficientes se estabilizan siendo casi constantes.

V.2. ESTUDIO DE LA INFLUENCIA DEL NÚMERO DE FUENTES Y SENSORES

Con objeto de estudiar el comportamiento de algoritmo de CAR multicanal, se han realizado diversas simulaciones correspondientes a diferentes configuraciones multicanal.

Diseño de la simulación:

1º caso. La fuente emite sola, con una señal senoidal de 100 Hz, durante 500 ms.

2º caso. La fuente anterior emitiendo. Y un sistema de CAR con una fuente antiruido y con un sensor.

3º caso. La fuente anterior emitiendo. Y un sistema de CAR con dos fuentes antiruido y con dos sensores.

4º caso. La fuente anterior emitiendo. Y un sistema de CAR con tres fuentes antiruido y con tres sensores.

5º caso. La fuente anterior emitiendo. Y un sistema de CAR con tres fuentes antiruido y con cuatro sensores.

6º caso. La fuente anterior emitiendo. Y un sistema de CAR con cuatro fuentes antiruido y con cuatro sensores.

En la siguiente tabla se muestra la posición de las fuentes y los sensores en la simulación:

	Caso 1 (posición)	Caso 2 (posición)	Caso 3 (posición)	Caso 4 (posición)	Caso 5 (posición)	Caso 6 (posición)
Fuente	1 (0.5 0.5)	1 (0.5 0.5)	1 (0.5 0.5)	1 (0.5 0.5)	1 (0.5 0.5)	1 (0.5 0.5)
Fuente Anti- ruido	0	1 (1.5 1.8)	2 (1.5 1.8) (1 1.4)	3 (1.5 1.8) (1 1.4) (1.7 0.8)	3 (1.5 1.8) (1 1.4) (1.7 0.8)	4 (1.5 1.8) (1 1.4) (1.7 0.8) (1 1)
Sensor	0	1 (2 1.5)	2 (2 1.5) (2.2 1.2)	3 (2 1.5) (2.2 1.2) (2.3 1.8)	4 (2 1.5) (2.2 1.2) (2.3 1.8) (2.5 1.5)	4 (2 1.5) (2.2 1.2) (2.3 1.8) (2.5 1.5)

Fig.11. Tabla que representa la posición de los sensores y fuentes secundarias en el sistema de CAR

La siguiente imagen muestra el esquema de la simulación:

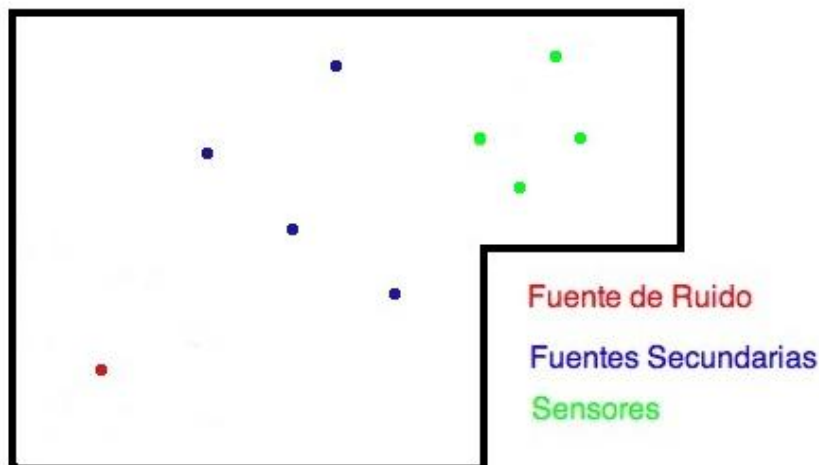


Fig.12. Esquema que representa el montaje de la simulación.

El primer paso para comprobar el comportamiento del algoritmo LMS con filtrado x multicanal es la comparación entre las señales grabadas en el mismo punto en distintas condiciones.

Con esta primera comparación queremos mostrar que sucede en el sensor uno (2 1.5) conforme aumentamos el número de estos y de fuentes de ruido.

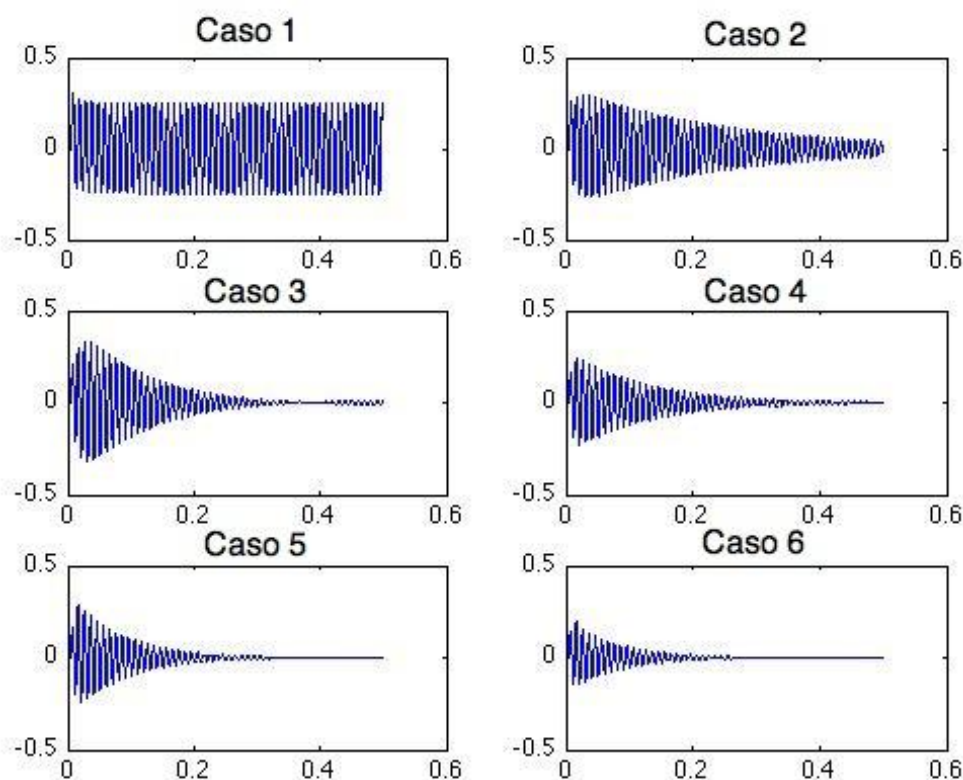


Fig.13. Representación de la señal grabada por un sensor en los diferentes casos. En el eje x tiempo de simulación (s), eje y presión instantánea (arbitraria).

Como vemos en la gráfica la presión en el punto disminuye conforme aumentamos el número de sensores y de fuentes, esto no siempre es así, hay casos en los que el sistema CAR da mayores atenuaciones (en los sensores) en los casos de una fuente secundaria y un único sensor de error. Pero el campo acústico resultante en los casos con más fuentes y/o sensores se observa una mayor superficie de actuación (atenuación). Un ejemplo de lo anteriormente comentado se puede ver en el anexo II.

Una vez que hemos visto las señales en los sensores de error queremos observar el campo acústico resultante de la simulación para conocer la atenuación en cada caso, ya que como hemos visto antes una mayor atenuación en un punto no tiene porque significar una mayor zona de actuación.

Para expresar los resultados utilizamos la potencia media de la presión acústica. Usamos esta potencia para hacernos una idea del campo acústico resultante (no queremos ver el campo del estacionario del CAR, es simplemente una forma de comparar los diferentes casos). Dado que todas las simulaciones tienen la misma duración y la fuente es la misma en todos los casos, las diferencias se deben únicamente al CAR.

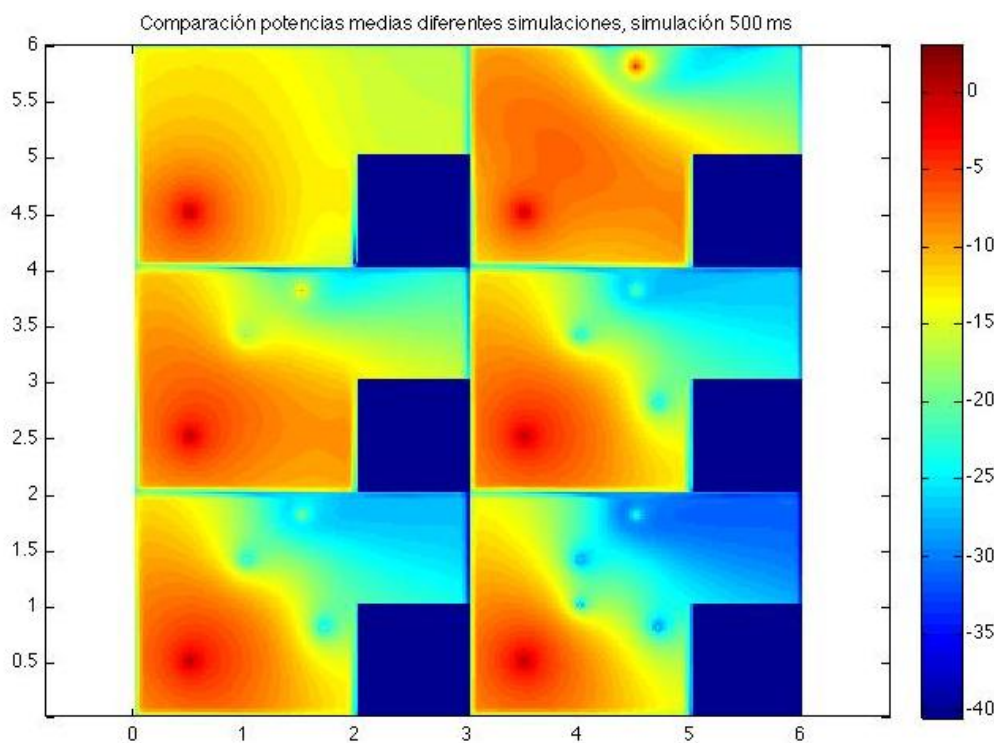


Fig. 14. Representación gráfica de las potencias medias obtenidas durante las simulaciones: de izquierda a derecha y de arriba abajo. Solo fuente, CAR 1 f y 1s, CAR 2f y 2s, CAR 3f y 3s, CAR 3 f y 4s, CAR 4f y 4s.

Tras ver la gráfica podemos observar que cuantos más pares de sensores – fuentes secundarias tenga nuestro sistema mayor será el área de influencia del CAR.

Como hemos comentado antes, debemos encontrar la mejor relación entre atenuación y superficie afectada dependiendo de cada caso. Así como la mejor posición posible de los transductores.

V.3. EXPRESIÓN DE LOS RESULTADOS

Una vez que vemos que el sistema funciona correctamente, intentamos averiguar la mejor forma de expresar los resultados para que se pueda mostrar la diferencia entre una sala con CAR y sin él.

Pensamos que la mejor forma de representar esta diferencia es la de mostrar los campos acústicos en las dos situaciones. En el caso sin CAR se realizará la potencia media de la presión en un intervalo concreto de la simulación, mientras que el caso del CAR el resultado será la potencia media de la presión del mismo intervalo de tiempo, pero con la condición de que se haya alcanzado el estacionario.

La siguiente gráfica muestra la comparación de potencias medias con y sin CAR, del mismo intervalo de tiempo una vez ya en el estacionario (llega a las 350 ms) durante un intervalo de 150 ms. La representación se realiza en escala logarítmica arbitraria.

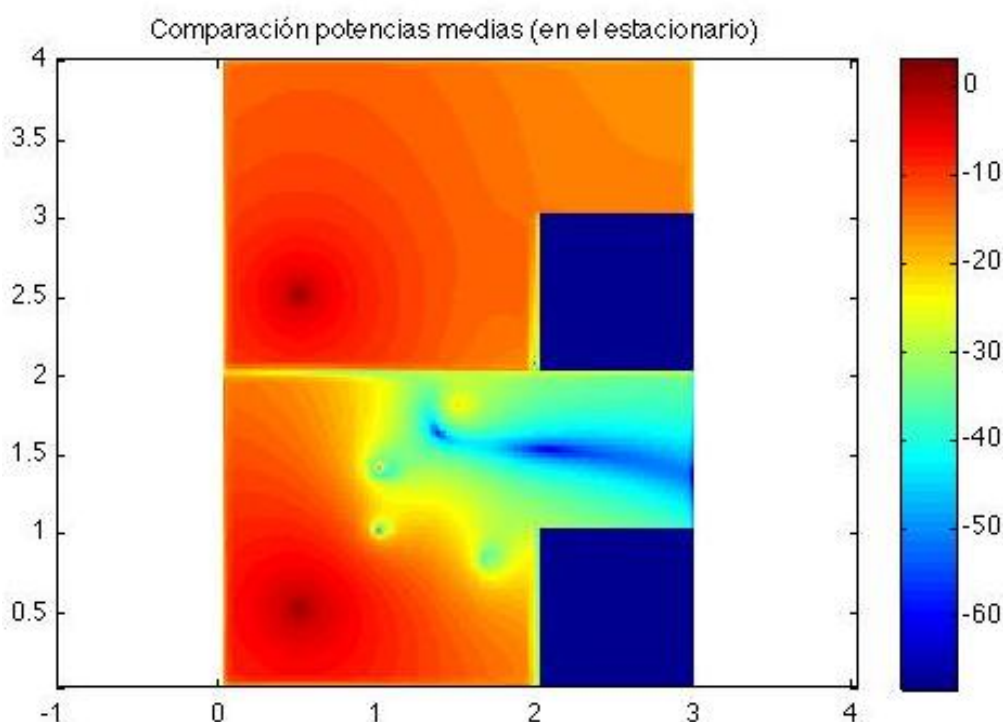


Fig.15. Comparación de potencias medias entre una sala con y sin CAR. Se muestran las dos salas juntas para tener la misma referencia a la hora de calcular la escala logarítmica. La sala sin CAR esta desde 2 hasta 4 en el eje y, mientras que la sala con CAR esta entre 0 y 2.

Para poder visualizar mejor la diferencia se ha calculado la atenuación entre las dos simulaciones como el logaritmo del cociente de la potencia media sin CAR y la potencia media con CAR.

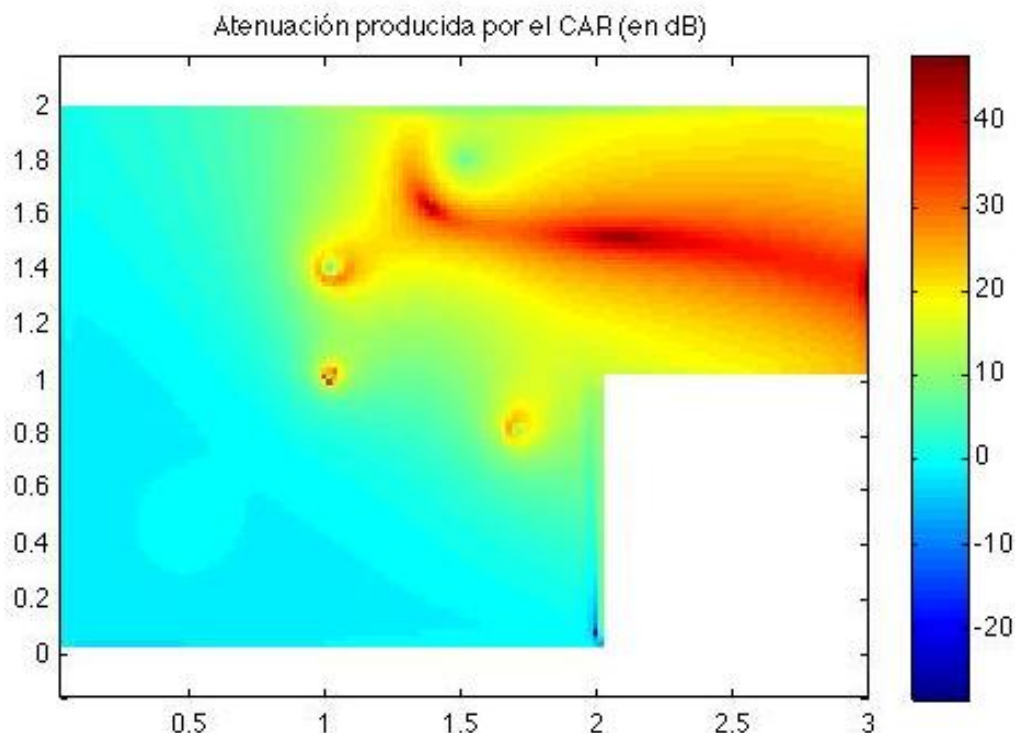


Fig.16. Diferencia entre las potencias medias con y sin CAR. Escala logarítmica referencia la potencia media de la sala sin el CAR.

Como podemos observar la zona de actuación producida por el CAR es bastante extensa y cubre la zona en la que queríamos atenuar el ruido. En la gráfica se observa que tenemos un máximo de atenuación de 48 dB, y la disminución del ruido es superior a los 25 dB en la zona de interés, zona en la que están situados los sensores.

V.4. VARIACIÓN DEL ALFA (ABSORCIÓN DEL PML)

En este apartado se pretende comprobar los resultados de las simulaciones con diferentes absorciones del PML. Dado que los ejemplos anteriores se han llevado a cabo con un alfa (absorción del PML) de 0.5, se ha querido comprobar como se comportaba el mismo algoritmo en condiciones más adversas, con un alfa menor.

Como nuevo coeficiente de absorción del PML se ha elegido 0.1.

A continuación podemos ver la comparación de potencias medias (una vez que el CAR está en el estacionario) entre la sala con y sin CAR.

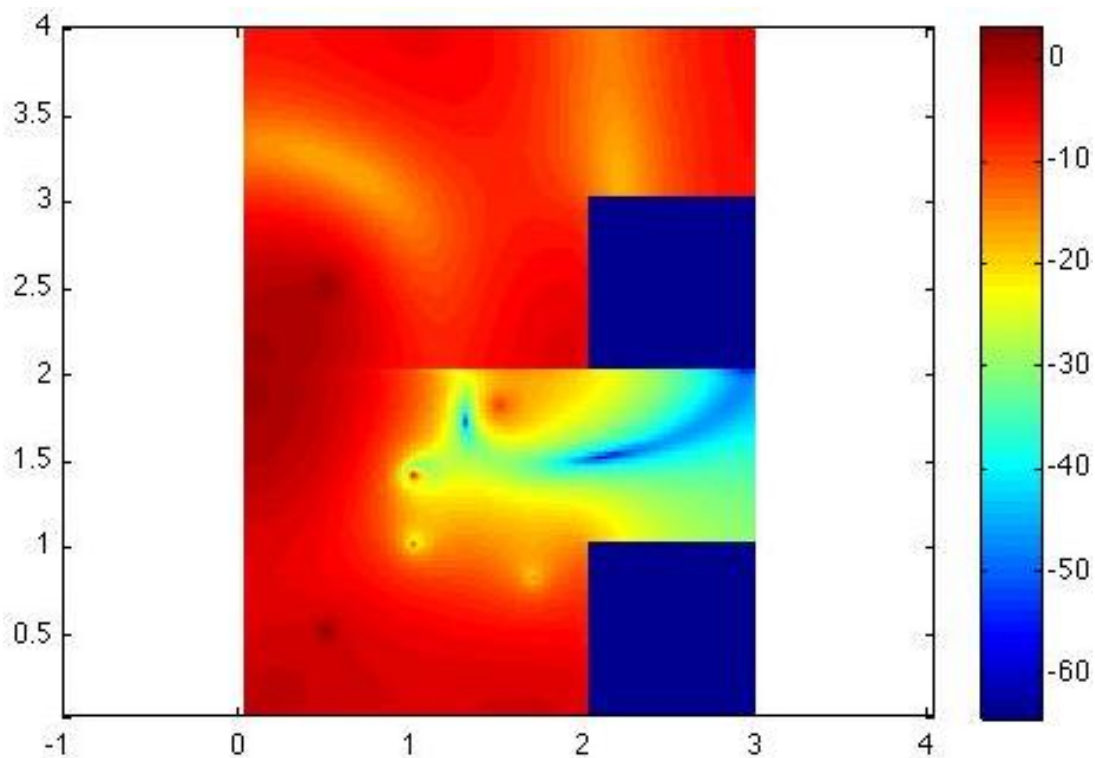


Fig. 17. Representación de las potencias medias con y sin CAR escala logarítmica arbitraria. La representación sigue la misma estructura que la Fig.11.

Cálculo de la atenuación en cada punto como la división, en dB, de la potencia media sin CAR entre la potencia media con CAR.

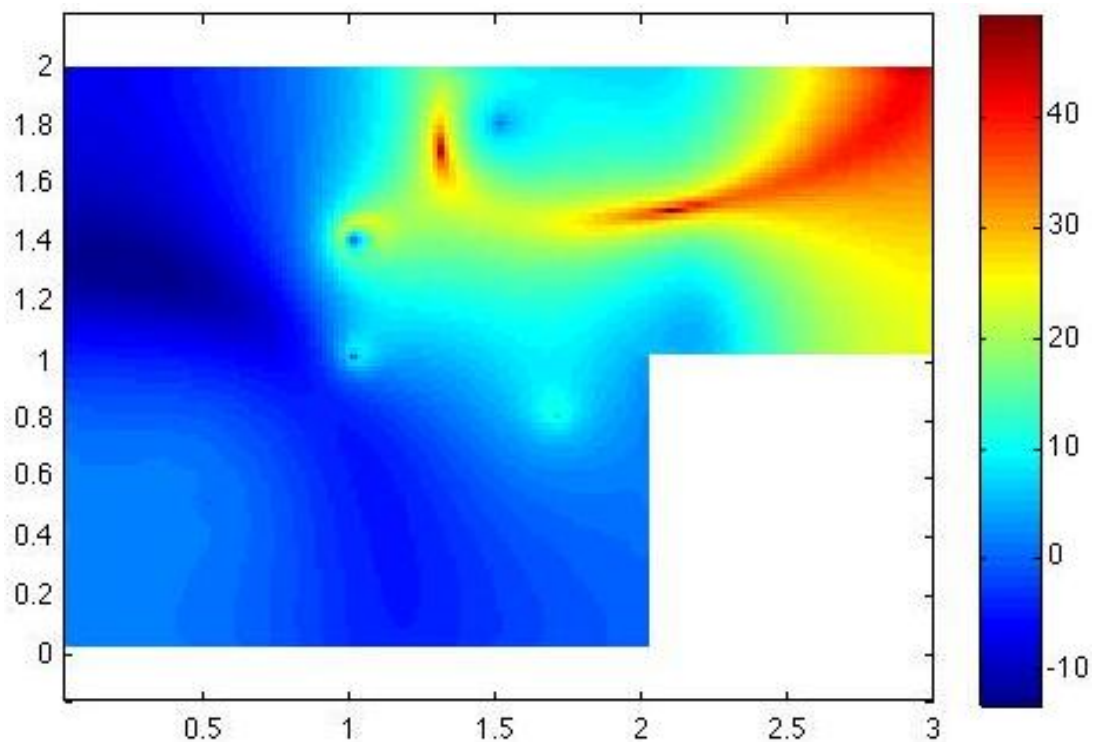


Fig.18. Diferencia entre las potencias medias con y sin CAR. Escala logarítmica referencia la potencia media de la sala sin el CAR.

En esta gráfica se observa que no sólo hay puntos en los que la señal es atenuada al activar el CAR, sino que también existen regiones de la sala en los que la presión aumenta (en torno a 10 dB), esto es mucho más real que el caso anterior en el que el CAR conseguía atenuaciones en toda la sala.

Todo ello se debe a que el anterior alfa del PML era muy elevado, es decir, absorbía mucha energía, consiguiendo condiciones casi anecoicas en la sala. Por el contrario el nuevo alfa no tiene este efecto, por lo que se asemeja mejor a una sala real.

En lo referente a la atenuación en este nuevo caso, vemos como el CAR sigue atenuando nuestra zona de interés, llegando a atenuaciones de 48 dB y siendo superior a 10 dB en la zona que nos interesa.

De todas estas observaciones podemos sacar la conclusión de que hay que tener especial cuidado al elegir el alfa del PML, para simular lo mejor posible nuestro caso a estudio. También hay que prestar atención en no confundir el alfa del PML con el coeficiente de absorción tal y como lo entendemos en acústica.

V.5. IDEA: ENCERRAR LA FUENTE DE RUIDO

Una vez que ya hemos visto que la herramienta diseñada funcionaba correctamente, sabemos como expresar los resultados y tras observar las simulaciones anteriores, pasamos al siguiente paso, buscar alguna utilidad a este sistema.

Viendo el ejemplo anterior se nos ocurrió intentar “encerrar” la fuente de ruido acústicamente, es decir, diseñar un sistema de control activo de ruido rodeando la fuente de ruido (senoidal de 100 Hz) con fuentes secundarias. Con la posición de los sensores de error no se pretende que el ruido sea mínimo en esos puntos, sino que el ruido sea también mínimo en todos los puntos de la sala.

A continuación se muestra un esquema de cómo se montó la simulación:



Fig.19. Esquema de la simulación.

En esta simulación se eligió un alfa (absorción PML) de 0.1, lo cual quiere decir que la absorción por parte del PML es baja, se diseñó así para intentar simular mejor las condiciones de un sala cualquiera.

En la siguiente gráfica se muestra la comparación de potencias medias durante 500 ms, ya llegado el estacionario del CAR. Con una señal de excitación de 100 Hz.

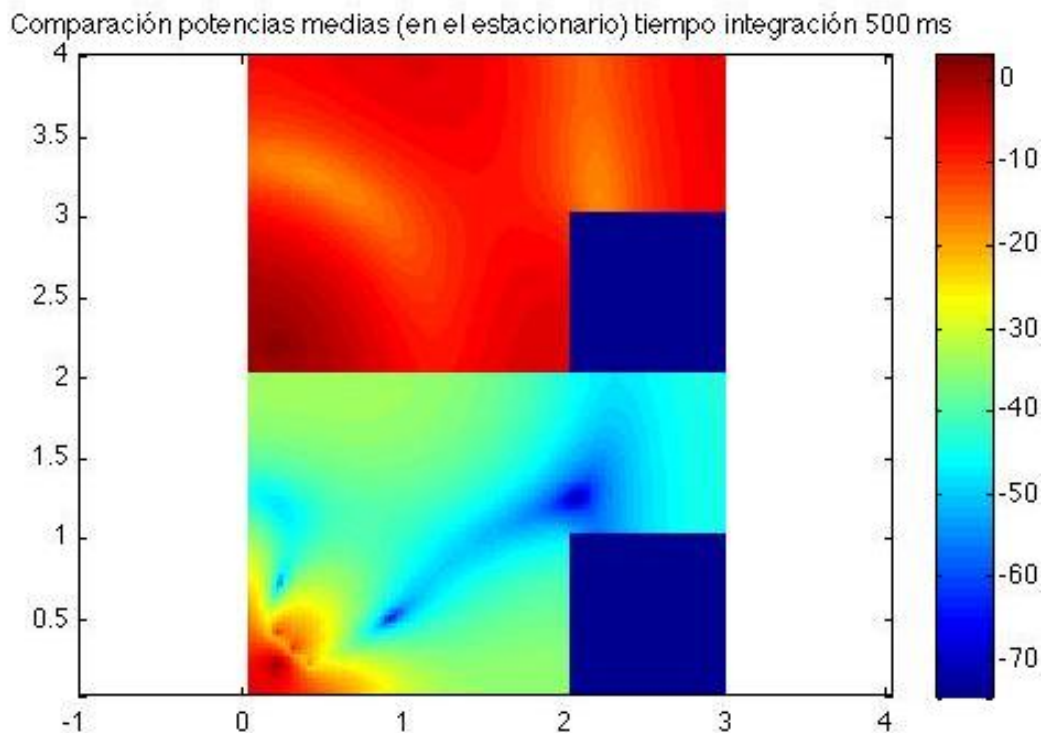


Fig. 20. Representación de las potencias medias con y sin CAR escala logarítmica arbitraria. La representación sigue la misma estructura que la Fig. 11.

En la imagen anterior observamos como este sistema consigue elevadas atenuaciones en toda la sala. También se puede observar que las atenuaciones más altas no se encuentran en los sensores de error, pero esto no es importante, ya que el objetivo de este montaje era atenuar el ruido en toda la sala.

Como hemos visto esta configuración es válida para baja frecuencia (100 Hz), pero que sucedería con frecuencias más altas, a continuación se muestra una imagen con el mismo diseño anterior, pero cambiando la frecuencia de emisión a 500 Hz, en ambos casos se eligen señales senoidales porque es más fácil ver su atenuación.

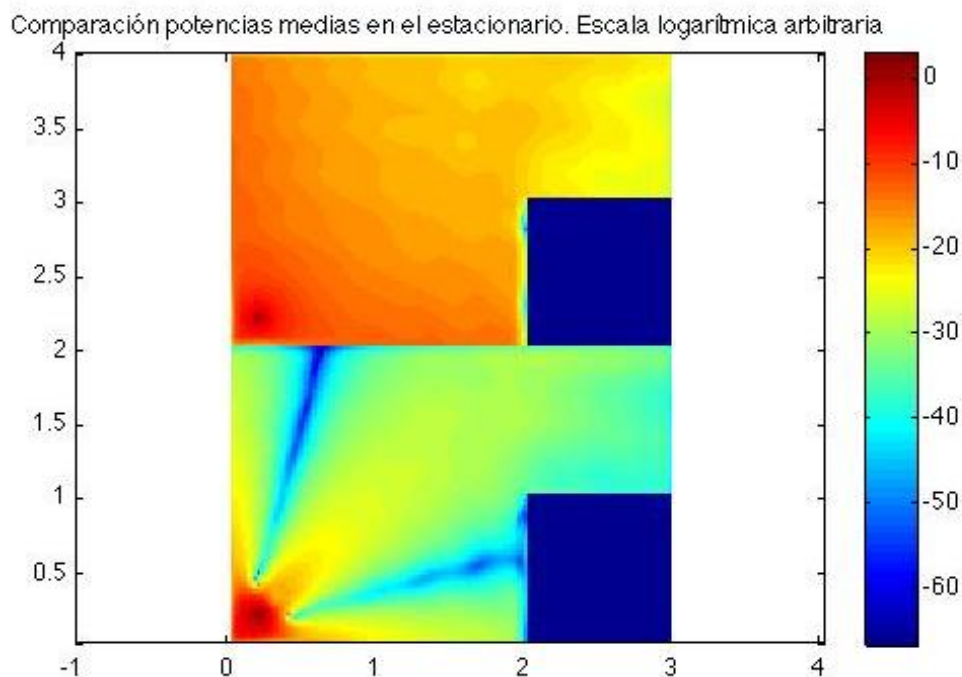


Fig. 21. Representación de las potencias medias con y sin CAR escala logarítmica arbitraria. Señal de excitación senoidal de 500 Hz. La representación sigue la misma estructura que la Fig.11.

Como podemos ver en la gráfica anterior, el ruido también se ha atenuado en la sala, sin llegar a conseguir atenuaciones tan elevadas como en el caso anterior. Como se puede ver este diseño no está preparado para frecuencias tan elevadas, pero este podría ser modificado para que actuase con mayor eficiencia: añadiendo más fuentes secundarias y colocándolas más cerca entre sí.

V.6. SIMULACIÓN MULTIFUENTE

De esta última aplicación surgió otra idea, la de intentar “encerrar” varias fuentes, para simular el ruido proveniente de una superficie vibrante, ventana o similar. Esta superficie emisora se simula con varias fuentes de ruido puntuales emitiendo en fase.

Para realizar esta simulación se decidió posicionar cinco fuentes de ruido cerca de la pared, y seis fuentes antiruido, todas ellas emitiendo una señal senoidal de 100 Hz en fase. También se decidió posicionar 5 sensores de error, estos sensores se posicionaron a medio metro de las fuentes secundarias formando una línea vertical, con la idea de que si el ruido es mínimo en estos puntos, aquellos puntos que estén detrás de ellos también se verán atenuados.

En las siguientes líneas se va a mostrar la diferencia de atenuaciones entre las dos simulaciones, con y sin CAR.

A continuación observamos la comparación de las señales grabadas en un sensor en las dos simulaciones, la gráfica se encuentra en unidades lineales arbitrarias.

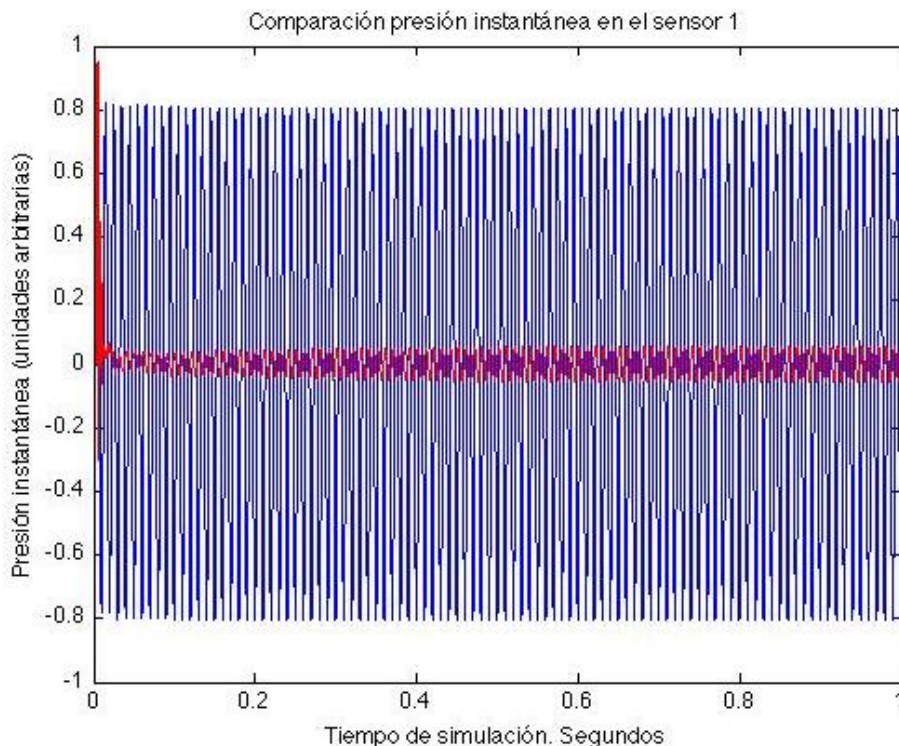


Fig.22. Comparación entre las señales grabadas con el CAR y sin él, en los sensores de error.

Para poder comprobar los efectos del CAR en toda la sala mostramos, en la siguiente imagen, los campos acústicos resultantes de las dos simulaciones.

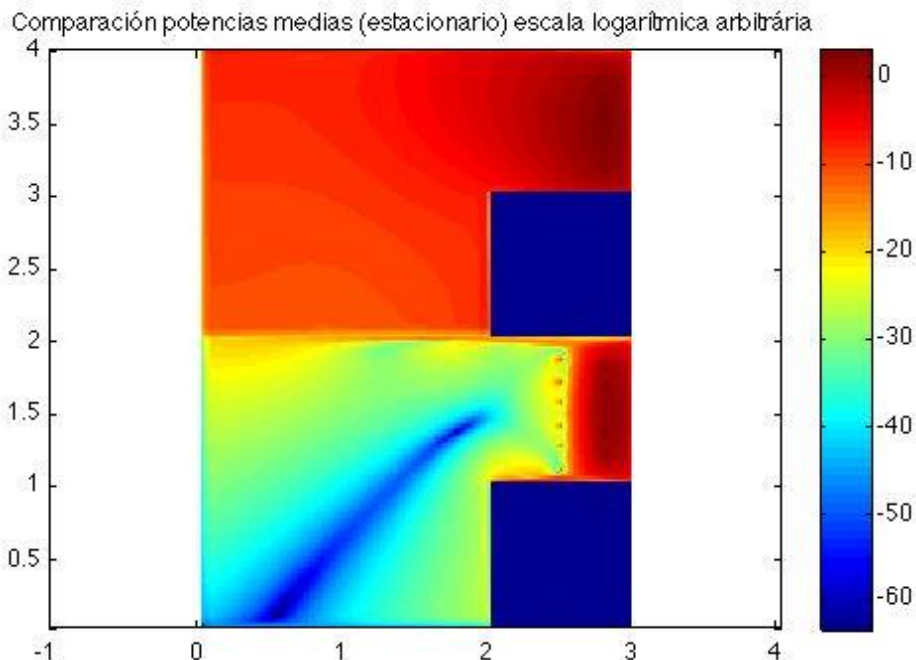


Fig. 23. Representación de las potencias medias con y sin CAR escala logarítmica arbitraria. La representación sigue la misma estructura que la Fig.11.

Como vemos se han conseguido atenuaciones en toda la sala, lo cual era nuestro objetivo.

VI. SIMULACIONES EN TRES DIMENSIONES

Tras desarrollar la herramienta para la simulación en dos dimensiones y comprobar que esta funcionaba como era esperado, se pasó al diseño del interfaz en tres dimensiones.

Una vez ya diseñada la nueva interface se quiso comparar los resultados de las simulaciones en dos y tres dimensiones. No olvidemos que la simulación en dos dimensiones no es real, es una aproximación que utilizamos para hacernos una idea del resultado y que nos ahorra mucho tiempo de cálculo.

A continuación se muestran algunas de las simulaciones que se pueden llevar a cabo con esta herramienta. Con ellas pretendemos entender un poco más el funcionamiento de los algoritmos de CAR, así como ver la robustez de la interfaz.

VI.1. COMPARACIÓN CON 2 D

El primer paso para comprobar el funcionamiento de los dos interfaces es la realización de la misma simulación en las dos herramientas. Se decidió elegir una sala con forma de "L" de altura constante, con una fuente en su interior (100 Hz). A su vez se colocaron varios micrófonos virtuales en la sala para tomar las señales grabadas como referencia.

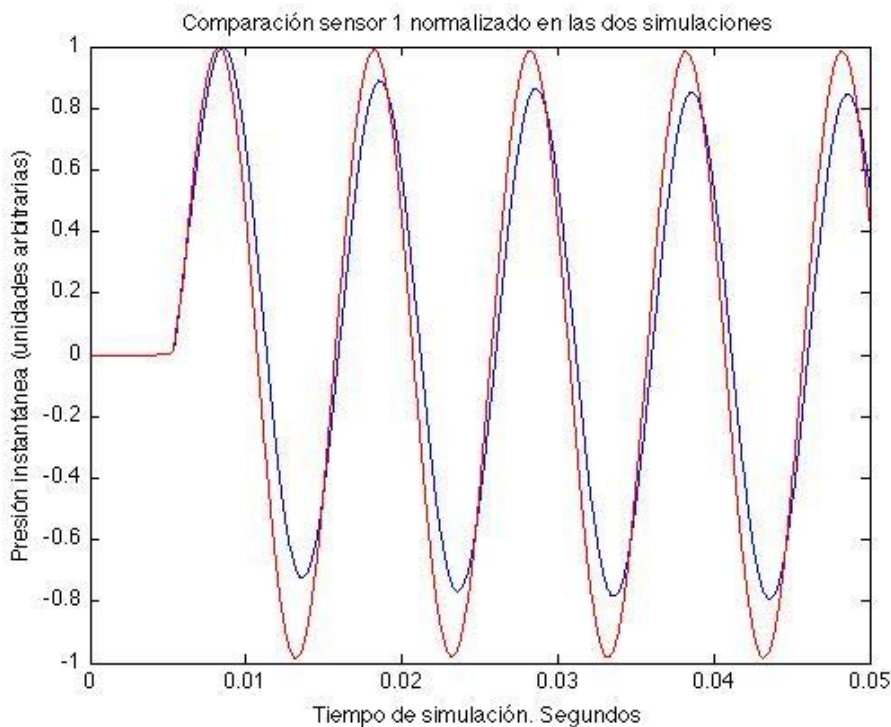


Fig.24. Comparación de la presión instantánea en los sensores, en los casos de simulación en dos y tres dimensiones.

Como se ve en el título de la gráfica, los valores han sido normalizados, esto es debido a la gran diferencia de amplitudes que presentaban las dos señales. En la imagen también se puede apreciar la similitud entre las señales, a excepción de un ligero

Autor: Jorge Machín Mindán, email: jormacmi@epsg.upv.es

Fecha de entrega: 16-10-2010

desfase. Pero si nos damos cuenta este desfase es constante, por lo que consideramos que las dos pequeñas diferencias entre las señales (amplitud y desfase) son debidas al cambio de tipo de propagación.

Otra curiosidad que se puede ver en la imagen anterior es que la señal medida en el recinto tridimensional mantiene la amplitud, mientras que en el caso de dos dimensiones se observa como la señal decae un poco. Consideramos que esto también es debido al cambio de propagación.

Otra comprobación que se también se realizó fue la simulación con el mismo montaje que en el caso anterior, pero cambiando la fuente de ruido por una fuente de ruido filtrado paso bajo, con una frecuencia de corte de 500 Hz (señal descrita en el apartado III.5.).

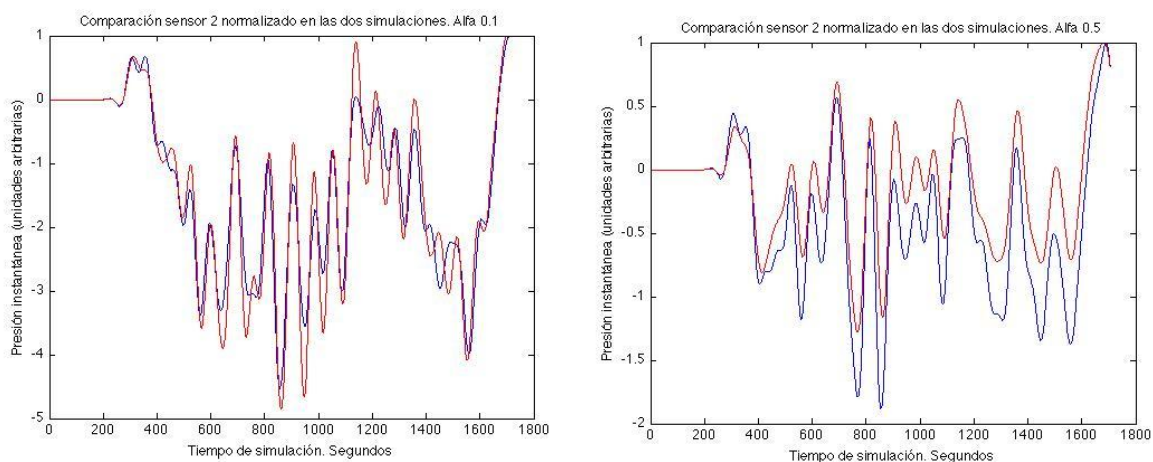


Fig.25 y 26. Comparación de las señales grabadas en los sensores en los casos de dos y tres dimensiones con diferentes absorciones del PML.

Como se puede observar en estas gráficas también han sido normalizadas. También se observan mayores diferencias que en el caso de la señal senoidal. Estas diferencias son variaciones en la amplitud de la señal, pero manteniendo la misma forma, por lo que consideramos que son las variaciones propias al cambio de propagación. Una curiosidad es que los resultados con una absorción baja son más parecidos que las simulaciones con más absorción.

VI.2. CAR EN TRES DIMENSIONES

Como ya se ha comentado antes, la primera prueba con el CAR en tres dimensiones se trata de la misma configuración que en dos dimensiones (Fig.7.), pero con una altura constante de 1.5 metros. La colocación de los sensores y fuentes también era la misma (en el mismo plano $z=0.5$).

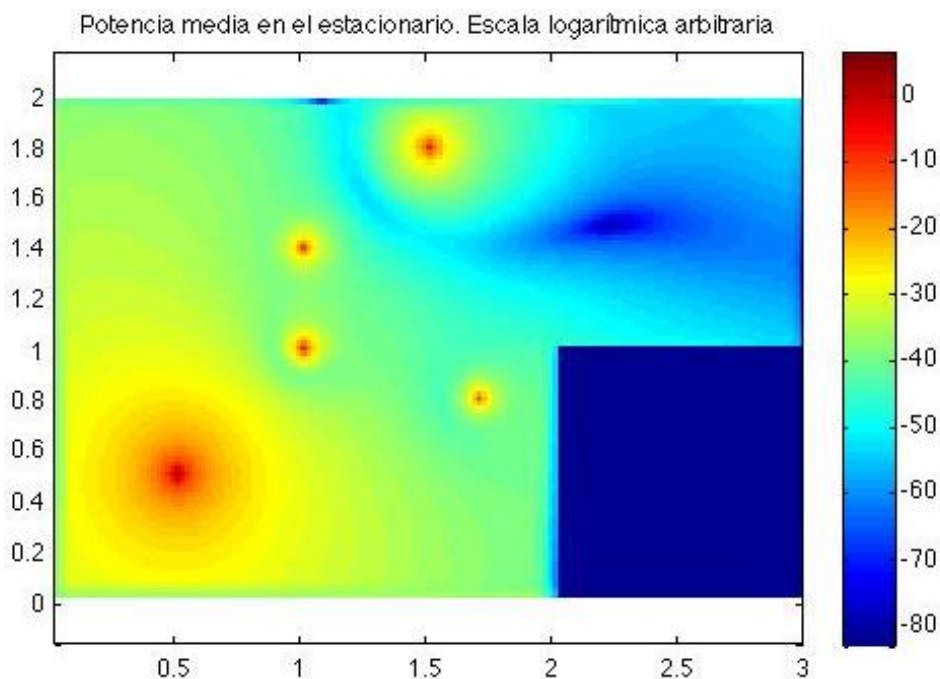


Fig.27. Expresa la potencia media en sala una vez en el transitorio del CAR a la altura de 0.5 m.

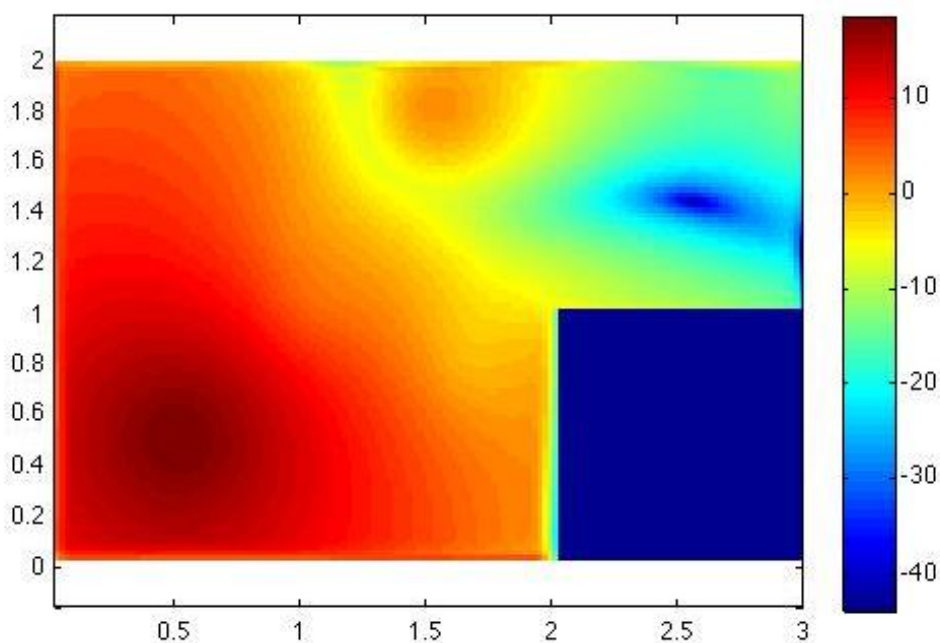


Fig.28. Expresa la potencia media de la presión en la sala una vez en el transitorio del CAR. Altura de 0.25 m.

Se puede observar que la potencia de ruido disminuye en la zona a estudio en las 2 alturas, no obstante se encuentra una gran diferencia en la zona baja de la sala donde las atenuaciones son muy inferiores. No olvidemos que los sensores se encuentran todos en el mismo plano, 0.5 m.

No olvidemos que con esta interface podemos diseñar fácilmente sistemas de CAR, así como la geometría de la sala que busquemos, por lo que se pueden realizar todas las simulaciones que nos imaginemos.

Tras ver la simulación anterior se diseñó un nuevo sistema para comprobar tanto el funcionamiento del algoritmo para la atenuación en zonas tridimensionales como la robustez de la herramienta.

Para esta nueva simulación se diseñó una sala y un sistema de CAR nuevo, como se puede ver a continuación:

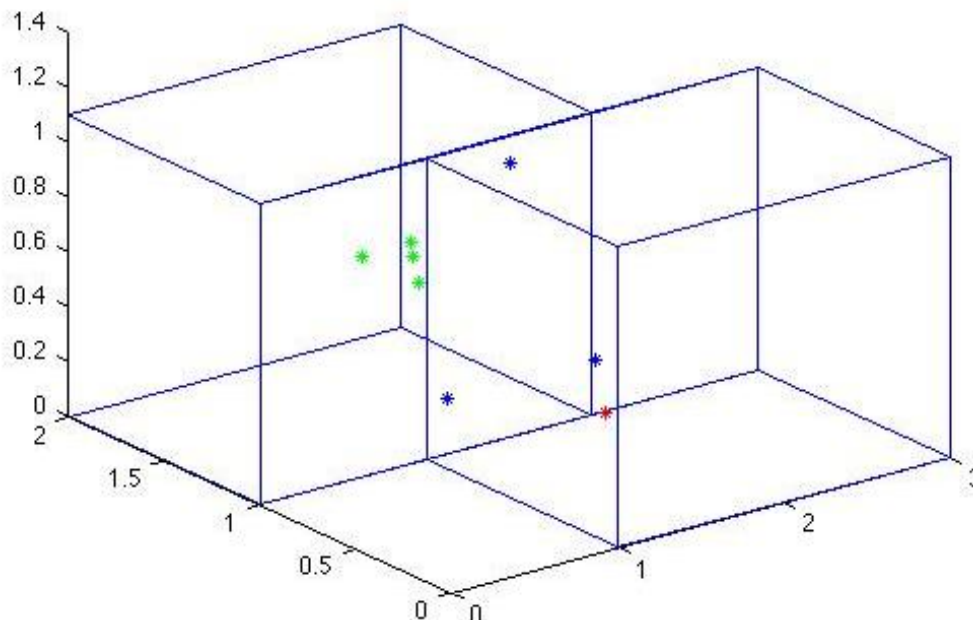


Fig.29. Esquema de la simulación. En rojo la fuente de ruido emitiendo una señal de ruido filtrada paso bajo a 300 Hz. En azul las fuentes secundarias y en verde los sensores de error.

Para ver el funcionamiento del CAR, como antes, mostramos a continuación la gráfica que compara las señales grabadas en el sensor uno en los dos casos (con y sin CAR).

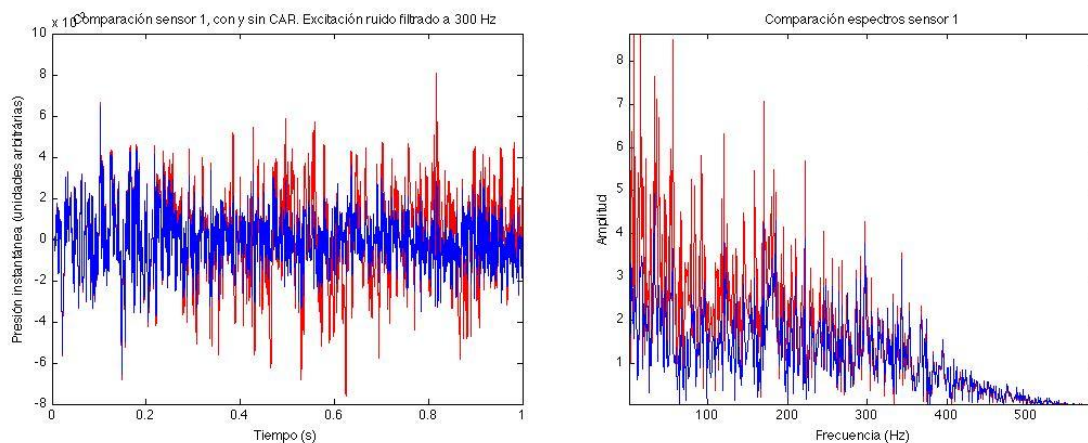


Fig. 30 y 31. Comparación entre el sensor 1 de la simulación con CAR y la simulación sin él. En la imagen de la izquierda vemos la amplitud de la señal con respecto al tiempo, mientras que la imagen de la derecha muestra el espectro de ambas señales.

Vemos como el control activo de ruido también funciona para señales de más alta frecuencia, no obstante las mayores atenuaciones se consiguen a baja frecuencia, como se puede ver en la imagen de la derecha. Por otro lado vemos que la excitación no tiene porque ser siempre una señal de ruido senoidal, sino que puede tratarse de cualquier tipo de señal, un ruido filtrado paso bajo a 300 Hz en este caso.

Siguiendo el objetivo de este proyecto nos disponemos a estudiar el campo acústico resultante tras las simulaciones. A continuación mostramos las potencias medias (en el estacionario del CAR) entre la sala con (a la derecha) y sin CAR a diferentes alturas.

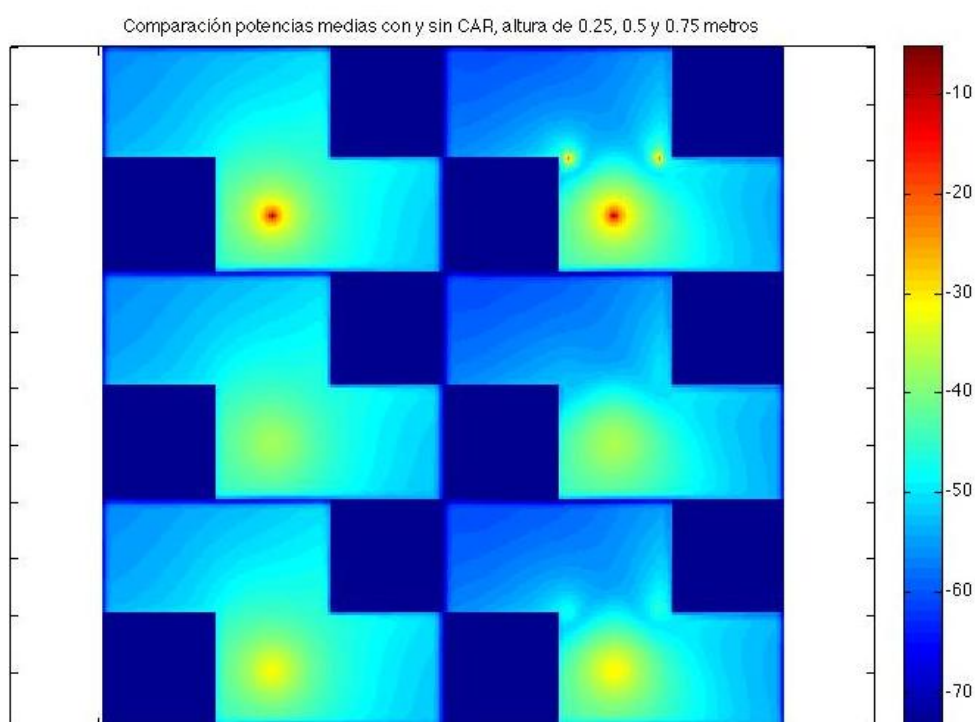


Fig.32. Comparación de las potencias medias (estacionario del CAR) entre la sala con (a la derecha) y sin CAR. De arriba abajo se muestra el resultado de la simulación a diferentes alturas, 0.25, 0.5 y 0.75 metros. Escala logarítmica arbitraria, se representan todos los resultados a la vez para tener la misma escala de referencia.

Para poder apreciar mejor la atenuación, se calculó la diferencia entre la potencia media sin CAR entre la potencia media con CAR.

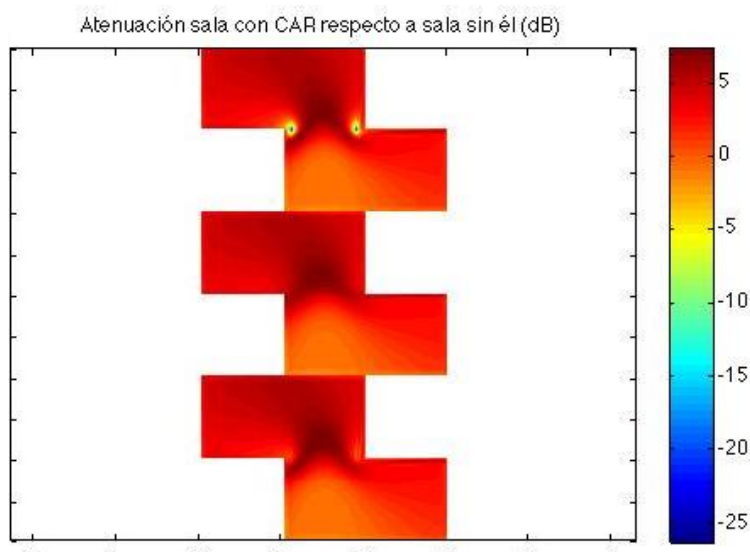


Fig.33. Atenuación producida por el CAR (dB ref potencia media sin CAR). De arriba abajo resultado de la simulación a 0.25, 0.5 y 0.75 metros.

Como se puede ver con el sistema de CAR diseñado se consiguen atenuaciones máximas de 7 dB y una media de unos 5 dB no sólo en la zona a estudio, sino en toda la mitad de la sala “superior” y en todas las alturas.

VI.3. TIEMPOS DE SIMULACIÓN

Como ya hemos mencionado con anteriormente un problema de este tipo de simulaciones es el gran coste computacional que conlleva. A continuación se muestra una tabla que recoge algunas de las simulaciones realizadas y los tiempos necesarios para llevarlas a cabo.

Descripción de la Simulación	Tiempo
Matriz: 100x100x100 50 ms	28 min.
Matriz: 100x100x100 50 ms Con CAR (4 F y 4 S)	36 min.
Matriz: 100x100x100 100 ms	55 min.

Matriz: 100x100x100 100 ms Con CAR (4 F y 4 S)	73 min.
---	---------

Fig.34. Tabla que expresa los tiempos de cálculo de diferentes simulaciones.

Viendo el problema que supone el coste computacional en tres dimensiones, sobretodo si aplicamos el CAR, se decidió intentar minimizar un poco este coste probando diferentes métodos de cálculo de los nuevos coeficientes del filtro adaptativo, sin dejar de lado el algoritmo LMS con filtrado x. Se propusieron dos posibles soluciones, la técnica minimax y la técnica scanning, que a continuación se explican.

De forma similar a la importancia de la elección del tipo de algoritmo a utilizar, la técnica de cálculo de los coeficientes es igual de importante, es decir, si conseguimos llegar al estacionario del CAR (que converja el filtro) tenemos que llegar al mismo resultado, independientemente del camino escogido. La misma idea puede ser aplicada a la estrategia para calcular los coeficientes del filtro, puede tardar más o menos, pero el resultado debe ser el mismo con las todas posibles técnicas [9].

- La estrategia “minimax” consiste en calcular los coeficientes del filtro adaptativo con el valor máximo de los sensores de error en valor absoluto.

$$e_kmax = \max(\text{abs}(e_1), \text{abs}(e_2), \dots, \text{abs}(e_k))$$

$$W_{ij} = W_{ij} - 2 * \mu * (x_{f_ijk} * e_kmax) \quad (9)$$

- La otra técnica, la de “scanning”, consiste en ir alternando la captura de datos de un sensor a otro en cada iteración.

$$W_{ij} = W_{ij} - 2 * \mu * (x_{f_ij1} * e_1) \quad (10)$$

$$\text{En la siguiente iteración: } w_{ij(n+1)} = w_{ij(n)} - 2 * \mu * (x_{f_ij2} * e_2)$$

Así sucesivamente.

Una vez definidas y programadas las técnicas a utilizar pasamos a las simulaciones. Utilizaremos la herramienta de dos dimensiones ya que el tiempo de cálculo se ve disminuido notablemente.

A continuación mostraremos la evolución temporal de los coeficientes de los filtros adaptativos para ver si las tres técnicas de cálculo nos dan los mismos resultados. Y cuanto tiempo tardan en converger cada una de ellas.

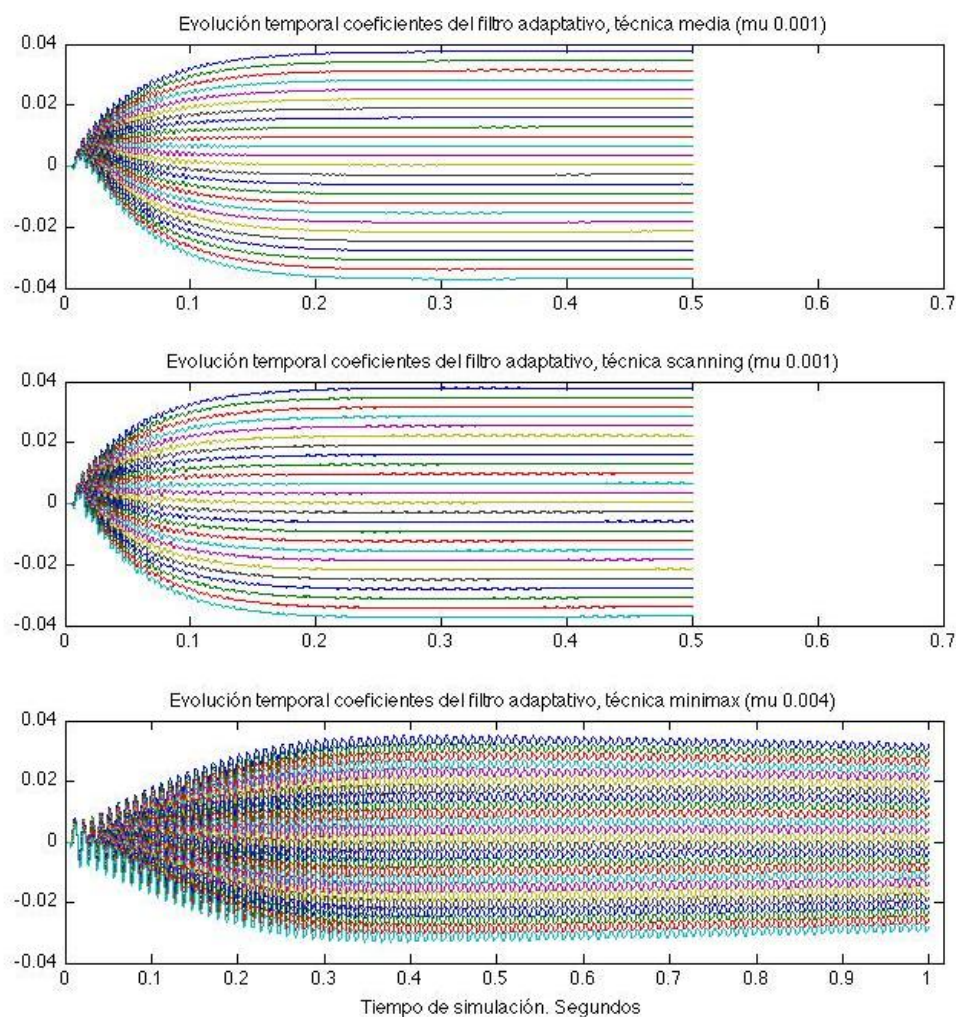


Fig.35. Comparación de la evolución temporal de los coeficientes del filtro adaptativo con las distintas técnicas.

Como podemos observar en la gráfica anterior, los dos primeros algoritmos convergen y llegan pronto a su solución, mientras que la tercera imagen parece que no llega a converger, siempre mantiene una oscilación alrededor de la solución.

A continuación mostraremos los campos acústicos resultantes de la misma simulación con las diferentes técnicas. De arriba abajo técnica de media, scanning y minimax.

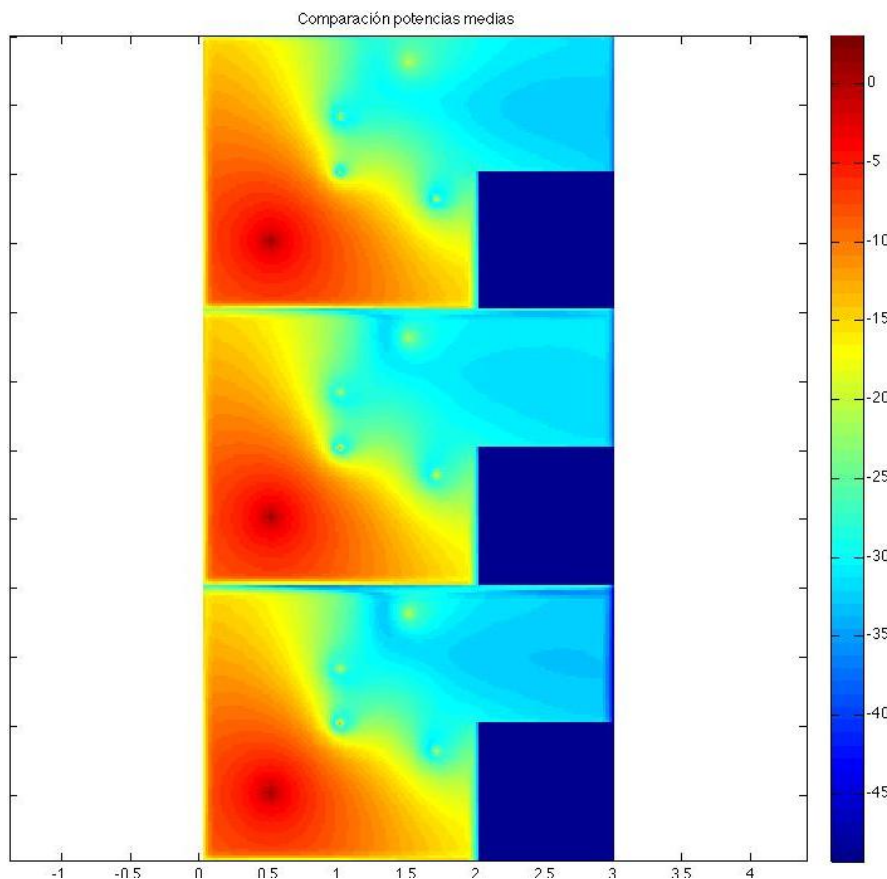


Fig.36. Comparación de potencias medias con distintas técnicas de cálculo. De arriba abajo, técnica media, scanning y minimax.

Tiempo de cálculo:

Tamaño matriz	Sistema CAR	Media	Scanning	Minimax
150 x 100 500 ms	4 F y 4 S	430''	415''	420''
100 x 100 x 100 50 ms	4 F y 4 S	36'	28'	35'

Fig.37. Tabla que expresa el tiempo de cálculo en las distintas simulaciones.

Tras ver los resultados de las tres técnicas y ver también el tiempo que lleva su procesado, recomendamos el uso de la técnica scanning para el cálculo de los nuevos coeficientes del filtro adaptativo, ya que las tres técnicas ofrecen resultados muy parecidos y es en el tiempo de cálculo en lo que se diferencian, siendo la técnica de scanning la que menos tiempo consume.

VII. CONCLUSIONES

Creemos que hemos conseguido cumplir con los objetivos marcados, desarrollar una herramienta para la simulación tridimensional de diferentes algoritmos de procesado de la señal, y con ella poder observar los campos acústicos resultantes. Y tras ver los resultados de las simulaciones anteriores, con diferentes configuraciones, creemos que se trata de una herramienta robusta y fácil de utilizar.

Podemos concluir que FDTD es una herramienta apropiada para ver como se comporta el CAR con el paso del tiempo, nos permite ver el campo acústico en cada momento, la evolución de los filtros del CAR, la señales en los sensores, el campo acústico en el transitorio, etc.

Tras ver la comparación podemos llegar a la conclusión de que el sistema CAR puede ser utilizado para intentar aislar las fuentes de ruido de una forma acústica. Si bien es cierto que debemos tener en cuenta la propagación del sonido para elegir la mejor colocación posible de nuestras fuentes secundarias y sensores de error. De todos modos un objetivo de este interfaz es la posibilidad de realizar distintas simulaciones para descubrir estas posiciones y así poder minimizar los efectos del ruido.

Hay que prestar atención al alfa del PML, ya que no se puede comparar con el coeficiente de absorción tal y como lo entendemos en acústica, sino que es la absorción del PML.

Mucho del tiempo invertido en la simulación es debido a las operaciones que tiene que realizar el CAR para funcionar, es por esto por lo que se propone el uso de la técnica scanning para ganar tiempo de cálculo. Que como hemos podido ver minimizan las operaciones a realizar en el algoritmo del CAR, si bien es cierto que pueden tardar más en converger los filtros adaptativos con estas técnicas, si el parámetro de convergencia “mu” no esta optimizado.

VIII. LÍNEAS FUTURAS

Es cierto que estas interfaces se han diseñado con objeto de simular diferentes algoritmos, pero también se le pueden dar otras aplicaciones, como puede ser la caracterización de salas mediante la obtención de la respuesta al impulso, sobretodo a baja frecuencia debido a la relación entre tamaño de la malla y frecuencia máxima a utilizar.

Otra aplicación puede ser didáctica con la explicación de fenómenos acústicos, difracción, superposición de ondas, efecto Doppler, etc.

Una de las líneas futuras más interesantes puede ser la de realizar distintas simulaciones con diferentes algoritmos de procesado de la señal, ya que al final sólo se puede realizar un algoritmo de Control Activo de Ruido en este trabajo, debido al poco tiempo disponible para la realización de este proyecto.

Otra línea interesante sería la de bajar el tiempo de cálculo de las simulaciones, para ello debería depurarse el código, optimizar el cálculo numérico, usar hardware basado en GPU's, etc. Por otro lado creo que una buena forma de bajar el tiempo de simulación es la de crear líneas de simetría para reducir el espacio de cálculo.

IX. BIBLIOGRAFÍA

- [1] Acoustics. L. L. Beranek. Acoustics Society of America, American Institute of Physics. 1993.
- [2] Adaptation of FDTD techniques to acoustics modeling. J.G. Maloney y K.E. Cummings. 11 th Annual Review of Progress in Applied Computational Electromagnetics. 1981.
- [3] Finite Difference Time Domain Tutorial. Dr Ian Drumm, The University of Salford.
- [4] Evaluation of the Perfectly Matched Layer for Computational Acoustics. Thomas L. Geers. 2002.
- [5] Wave Field Synthesis simulation by means of finite-difference time-domain technique. J. Escolano.
- [6] Multichannel adaptive filtering for signal enhancement. B. Widrow. 1981.
- [7] Analysis of the filtered-X LMS algorithm. E. Bjarson. 1995.
- [8] New block filtered-X LMS algorithms for active noise control Systems. S.M. Kuo. 2007.
- [9] Filtrado multicanal para control local sonoro basado en algoritmos de proyección afín. M. Ferrer. 2008

ANEXO I. Definición de Variables

DEFINICIÓN DE LA MATRIZ DE PUNTOS DE CONTORNO

El primer paso para la creación de las matrices que vamos a utilizar es definir una matriz donde guardaremos todos los puntos del contorno. Estos puntos son guardados en orden, cada punto, con sus respectivas coordenadas, en una fila.

Ejemplo de definición del primer punto del contorno:

```
datos.pos_puntos(1,:) = str2num(get(hObject,'String'));
```

Se elige esta forma de definir los puntos para poder guardar todos los puntos en la misma variable y así poder definir el número de puntos que se necesite, inclusive un número mayor de puntos de los que nos permite definir la GUI. Otra razón que nos lleva a esta decisión es la de evitar la utilización de bucles a la hora de leer esta información.

DEFINICIÓN DE LAS MATRICES DE CONTORNO, PMLs Y MATRICES DE SIMULACIÓN 2D

Como ya hemos comentado antes los puntos del contorno se definen en orden, por lo que ya conocemos las dimensiones máximas de nuestra matriz, la cual representa la superficie de simulación.

No sólo se crea la matriz en la que se desarrollará la simulación, también se crean otras matrices que utilizaremos como máscara a la hora del cálculo. Estas matrices son las encargadas de definir las condiciones de contorno (Pared rígida. Velocidad = 0).

De una forma similar serán definidas las matrices para definir las paredes mediante PML, en la creación de estas matrices podemos elegir el alfa máximo de la matriz así como la longitud del PML.

El primer paso para definir las matrices es calcular las dimensiones máximas de la matriz de simulación. Se calculan como los máximos en las dos dimensiones x e y. Con estos máximos se puede definir ya las matrices que guardarán los datos.

```
puntos= datos.pos_puntos / datos.malla;    % calculo de numero de nodos= tamaño/malla
dimensiones= max(puntos);                  % es solo para el máximo

%% Matrices para el PML
nx= dimensiones(1);    % En nx guardamos el máximo en el eje X
ny= dimensiones(2);    % En ny guardamos el máximo en el eje Y
alfaux=zeros(nx+1,ny); % Matriz para guardar información del contorno en el eje X
alfauy=zeros(nx,ny+1);
matrizX= ones(dimensiones(1)+1,dimensiones(2)); % Matriz con las dimensiones máximas para el eje X
matrizY= ones(dimensiones(1),dimensiones(2)+1);
```

Para realizar la matriz de contorno y de los PMLs, se cogen los puntos de contorno en parejas, de dos en dos, para posteriormente comprobar si los puntos están en la misma recta x o y , y sabiendo en que recta está el contorno podemos definir ya el PML de esa pared.

La matriz de contorno la definimos como una matriz de unos, menos los contornos que son 0, para poder multiplicarlos luego por la matriz de la velocidad (condición de contorno de velocidad nula en las paredes).

Para definir los PMLs, creamos los PMLs hacia el interior, si la pared esta cerca del mínimo o del máximo, ya sabemos hacia que lado se define el PML. En el caso en que la pared sea interior definimos el PML de una forma triangular, es decir, creamos un PML con el máximo en la pared y decreciente hacia ambos extremos.

Una vez realizado todo lo anterior, ya tenemos nuestras matrices definidas y preparadas para la simulación.

ANEXO II. EJEMPLO

En estas líneas se va a intentar demostrar un fenómeno que se da en algunos sistemas de control activo de ruido. Este fenómeno consiste en que en algunas ocasiones se obtienen mayores atenuaciones en los sistemas con una fuente antiruido y un sensor, pero dicha atenuación se da en una pequeña zona alrededor del sensor, mientras que si añadimos más sensores al sistema, la atenuación es menor pero se da en una mayor superficie.

Para llevar a cabo esta demostración se han realizado dos simulaciones, en la primera de ellas hay una fuente secundaria y un sensor de error, mientras que en la segunda hay una fuente y dos sensores de error.

Esquema de las simulaciones:

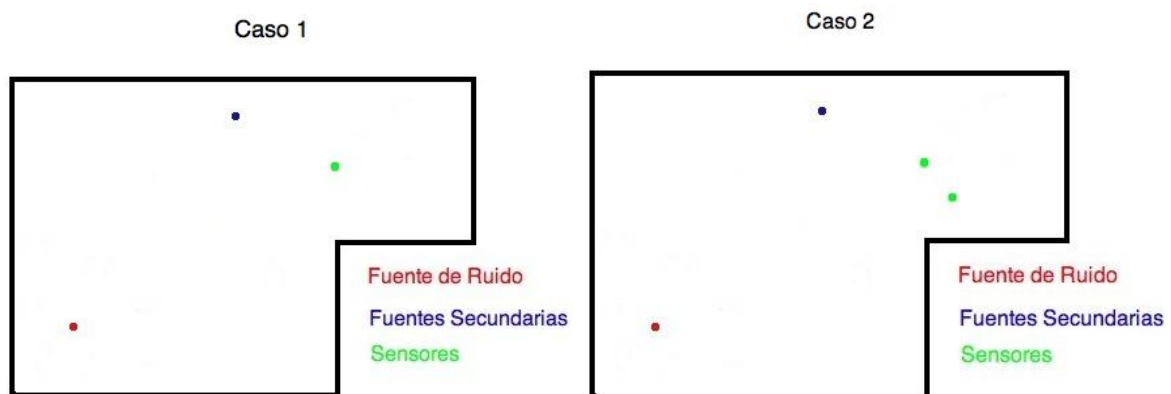


Fig.38y 39. Diseño de las simulaciones utilizadas para el ejemplo.

A continuación vamos a mostrar la señal grabada durante las dos simulaciones en el mismo punto.

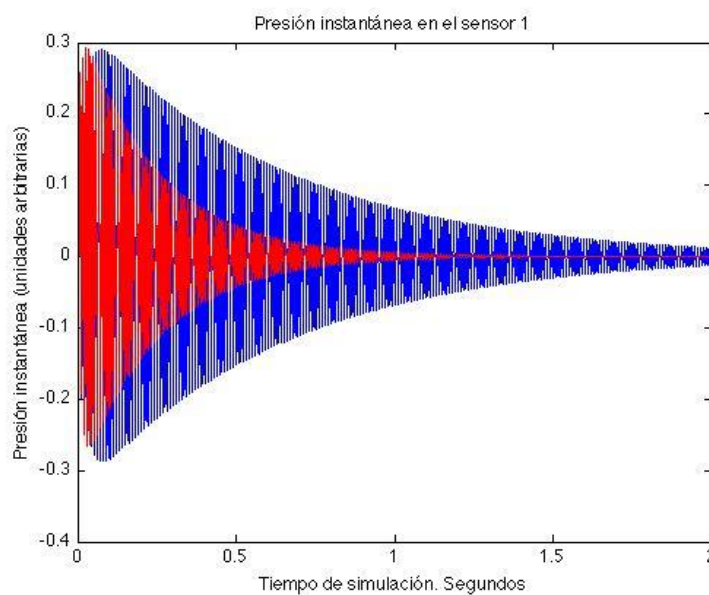


Fig.40. Comparación señal grabada en el sensor 1 durante las dos simulaciones.

La señal azul se corresponde con el sensor uno del caso 2 y la señal roja con el sensor del caso 1.

Como se observa la señal de error es menor en el primer caso, tal y como habíamos comentado, mientras que como se observa en la siguiente imagen, el CAR tiene una mayor superficie de actuación (consigue menores atenuaciones, pero en un espacio mayor).

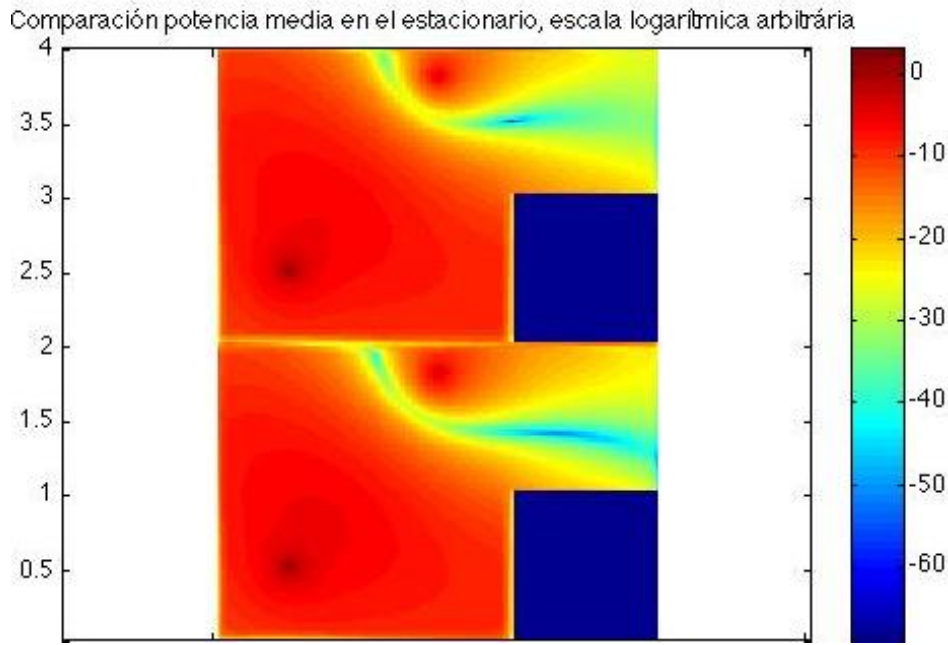


Fig.41. Comparación de la potencia media en los dos casos. Imagen superior caso 1, imagen inferior caso 2.

A continuación se procedió a calcular la atenuación conseguida al añadir un sensor, se calculó como el logaritmo de la división de la potencia media en el caso 1 entre la potencia media del caso 2:

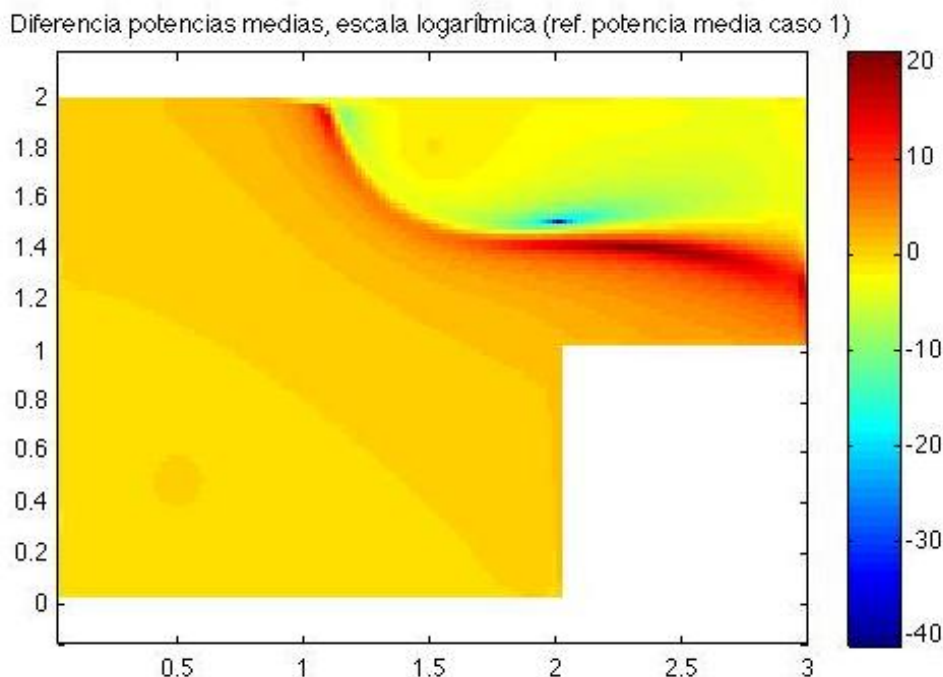


Fig.42. Atenuación producida por el CAR, en dB ref. potencia media sin CAR.

Se puede observar que en la mitad izquierda de la sala los valores no varían mucho, son cercanos a cero, por lo que podemos observar que la adición de un sensor no influye en esa parte de la sala.

No obstante vemos como la colocación de este sensor sí que influye en la zona a estudio. Esto se puede apreciar en la superficie roja, que nos indica que puntos de la sala han bajado el nivel de ruido al añadir el sensor, y en la zona azul la cual nos indica los puntos que han visto aumentado su nivel de ruido.

Tenemos que ser consciente de este “fenómeno” para diseñar el sistema CAR que más nos interese: debemos elegir entre atenuar una mayor zona de la sala entre 15 y 20 dB, o sacrificar esa zona por una atenuación mayor en un único punto (30 – 40 dB).

Como apuntábamos antes se puede llegar a la conclusión de que el CAR da mayores atenuaciones (en los sensores) en los casos de una fuente secundaria y un único sensor de error. Por el contrario, el campo acústico resultante en los casos con más fuentes y/o sensores se observe una mayor superficie de actuación (atenuación).

SOLAPAMIENTO DE ZONAS DE INFLUENCIA

Una idea relacionada con la expuesta anteriormente es la del solapamiento de zonas de influencia. Que a continuación vamos a intentar mostrar. El diseño de la simulación es similar al caso anterior, pero ahora vamos a conectar cuatro fuentes anti-ruido y un único sensor en cada una de las simulaciones (situación de las fuentes y sensores en la Fig.7).

Para apreciar este comportamiento del CAR debemos observar las zonas de actuación de las fuentes secundarias sobre cada uno de los sensores y sobre todos ellos a la vez.

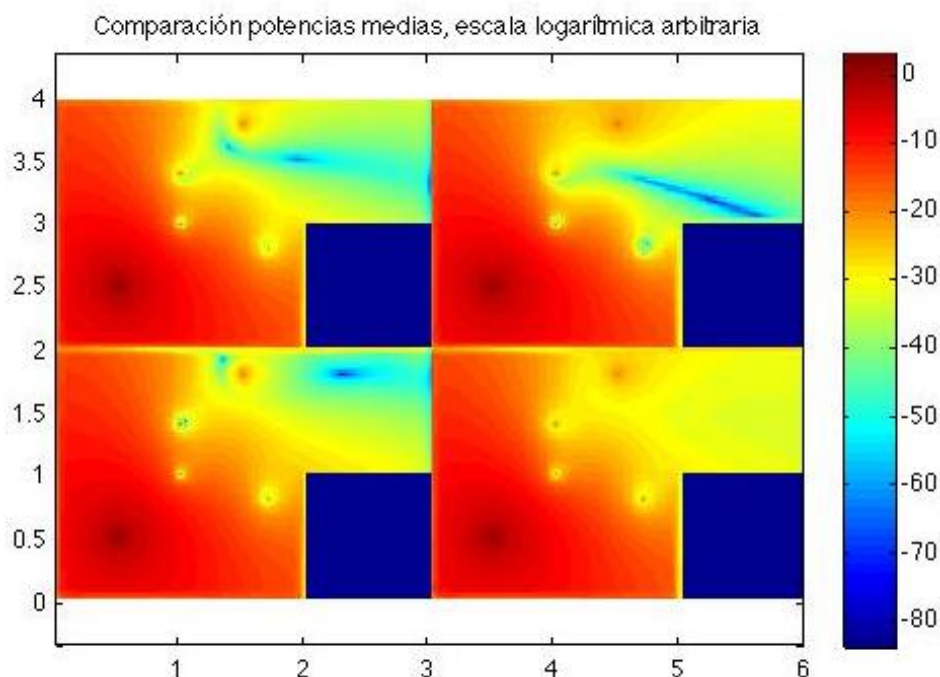


Fig.43. Potencias medias, área de influencia de cada uno de los sensores.

Como se ve en la imagen anterior, cada uno de los sensores hace que el CAR cree unas zonas de actuación distintas para cada sensor. Y por último podemos ver como todas estas zonas se superponen en el caso de utilizar los cuatro puntos como sensores de error.

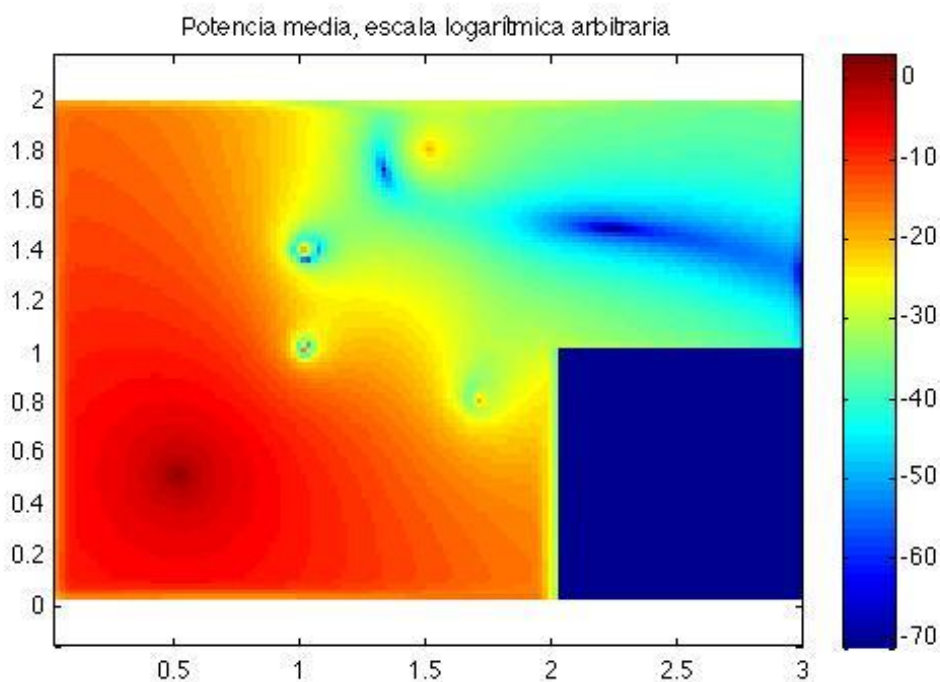


Fig.44. Potencia media, superposición de las zonas de actuación de cada uno de los sensores.

ANEXO III. GUÍA PARA USUARIO.

En este anexo se trata de explicar como se puede añadir a la plantilla un nuevo algoritmo. Para añadir un nuevo algoritmo debemos seguir una serie de pasos:

1º abrir la GUI, y crear una nueva opción en el listbox (el único que hay).

2º Debemos llamar a nuestro algoritmo desde el archivo simulación. Para ello debemos crear una condición en el archivo de que datos.algoritmo (variable que recoge el valor del listbox) coincida con la posición del nuevo algoritmo en el listbox.

Siguiendo los dos pasos anteriores ya podemos llamar a nuestro algoritmo. Pero debemos conocer cómo debemos recoger los datos, en que variables están y cómo debemos devolver los datos a la simulación (señales a emitir).

Datos de entrada:

FR:

Es la posición de las fuentes de ruido. En cada fila se encuentran las coordenadas de la fuente (dos o tres coordenadas dependiendo de la simulación elegida). Es una variable global.

F:

Es la posición de las fuentes anti ruido. En cada fila se encuentran las coordenadas de la fuente. Se trata de una variable global.

S:

Es la posición de los sensores. En cada fila se encuentran las coordenadas de cada uno de los sensores. Se trata Variable global.

datos.x:

Se trata de las señales en los sensores, la señal captada en cada sensor en una nueva fila. Se trata de una variable global. El vector se va llenando en cada interacción, es decir, mediante un sensor (mecánico) se adquieren las muestras una a una. La variable datos es una variable global.

contador:

Es una variable que guarda el número de iteración en la que nos encontramos, se incrementa después del paso por el procesado elegido (CAR, etc.). Es una variable local del archivo simulación.

p:

Es la matriz en la que se guardan los valores de la presión instantánea en cada instante de tiempo.

Datos de salida:

Debemos devolver al archivo simulación las señales a emitir como datos.y_i (siendo i cada una de las fuentes antiruido).

Para posteriores tratamientos se decidió guardar las señales anti-ruido enteras, es decir, deben tener la misma duración que la simulación. Es por esto por lo que debemos tener en cuenta en que interacción estamos para “colocar” la muestra en su sitio.

También hay que tener en cuenta cuando se actualiza el contador para emitir la señal que toca, es decir, en la iteración que estemos debemos emitir la muestra del vector datos.y_i en la posición contador menos uno.

Cargar ficheros:

Una opción de las interfaces es la de cargar diferentes ficheros. Estos deben ser archivos txt. Y deberán seguir una estructura concreta, dependiendo del tipo de información que contengan.

En el caso de fuentes secundarias, sensores, paredes y/o puntos del contorno las coordenadas de cada punto deberán estar en una línea diferente. Las coordenadas irán separadas por un espacio en blanco.

Mientras que en el caso de fuentes de ruido no se tendrá sólo la información de la posición de la fuente, sino que también se tendrá información de la señal de ruido. A continuación mostramos en un ejemplo como deberá ser escrita esta información:

Ejemplo de archivo para el almacenaje de fuentes de ruido en dos dimensiones:

```
0.5 0.5 1 100
```

Las dos primeras columnas están reservadas para las coordenadas x e y del punto, la siguiente columna indica el tipo de señal a emitir y por último la cuarta columna indica la frecuencia de la señal, frecuencia de corte, etc dependiendo del tipo de señal elegida.

A continuación mostramos los posibles tipos de señal con sus respectivos números asociados.

Senoidal	1
Ricker	2
Ruido filtrado	3
Archivo .wav	4