



VI Congreso Iberoamericano de Acústica - FIA 2008
Buenos Aires, 5, 6 y 7 de noviembre de 2008

FIA2008-A154

Diseño de cajas acústicas optimizado mediante algoritmos genéticos

Ricardo Páez Toro^(a)
Patricio Olmos Duque^(b),

- (a) Departamento Ingeniería, Ingeniería en Control Acústico Ltda, Rogelio Ugarte 1817, Santiago, Chile. E-mail: Desarrollo2@contacus.cl.
- (b) Departamento Ingeniería, Ingeniería en Control Acústico Ltda, Rogelio Ugarte 1817, Santiago, Chile. E-mail: Ingenieria@contacus.cl.

Abstract

The process of design of acoustic enclosures has practically stayed inalterable from the development of the models of Thiele and Small. The process traditionally requires a high dominion on the part of the designer to obtain an reasonable frequency response according to the characteristics of the traditional electromechanical transducers. By means of the use of computational resources it is possible to reverse the design procedure, to initiate the design from an wished frequency response and to optimize the enclosure by means of the use of a genetic algorithm that evaluates the final performance of a series of parameters that influence in the polynomial solution that defines the characteristic response of each type of enclosure and alignment.

Resumen

El proceso de diseño de cajas acústicas se ha mantenido prácticamente inalterado desde el desarrollo de los modelos de Thiele y Small. El proceso tradicionalmente requiere un alto dominio por parte del diseñador para lograr una respuesta de frecuencia razonable de acuerdo a las características de los transductores electromecánicos tradicionales. Mediante el uso de recursos computacionales es posible invertir el procedimiento de diseño, iniciar el diseño desde una respuesta de frecuencia deseada y optimizar la caja mediante el uso de un algoritmo genético que evalúa el desempeño final de una serie de parámetros que influyen en la solución de polinomio que define las respuestas características de cada tipo de caja y alineamiento.

1 Introducción

El presente trabajo busca entregar una nueva técnica de diseño de cajas acústicas a partir de la respuesta de frecuencia deseada por el diseñador.

Para esto se implemento el software en Matlab 7, el cual mide los parámetros T-S de altavoz en cuestión y permite diseñar cajas del tipo cerrada y bass reflex a partir de la teoría clásica. La diferencia de este software en comparación con los de tipo comercial radica en que este software trabaja manipulando elemento del polinomio de la respuesta de frecuencia de cada tipo de caja, optimizando la respuesta mediante la reducción del error cuadrático medio por medio de la técnica de optimización por algoritmos genéticos.

2 Desarrollo

2.1 Adquisición de datos

Los parámetros T-S de los altavoces son obtenidos a partir de modulo de medición del software basado en el trabajo de C.J Stuck [1], el cual define el método computacional de obtención de los parámetros T-S de un altavoz electromecánico a partir de la medición de su respuesta de impedancia.

El funcionamiento del software en su conjunto, esta desarrollada en [2].



Ilustración 1: Kit de medición utilizado para la medición de parámetros T-S

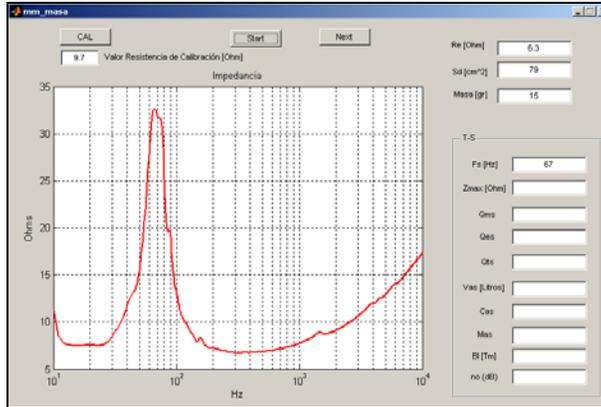


Ilustración 2: Respuesta de impedancia medida.

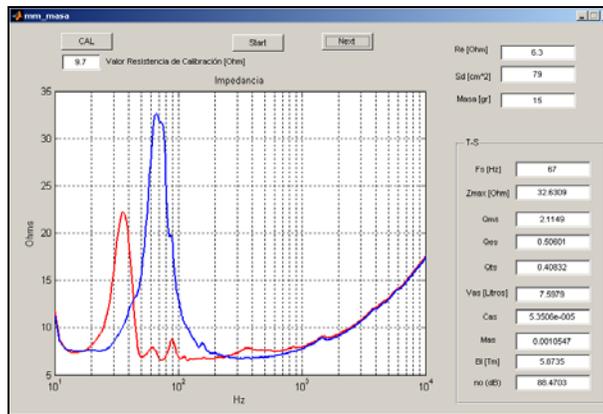


Ilustración 3: Medición de parámetros T-S

2.2 Diseño Clásico de cajas Acústicas

Para el desarrollo y solución del problema matemático del diseño de cajas acústicas se utilizaron las funciones de transferencia de los distintos tipos de cajas. El alineamiento de cada tipo de caja se realizó mediante el desarrollo del polinomio característico de cada tipo de caja [2], [3],[4],[5].

La ecuación 1 corresponde a la función de respuesta del tipo “cerrada”, en la cual los parámetros α y Q_L son valores de diseño a optimizar.

$$G(s) = \frac{s^4 T_B^2 T_S^2}{s^4 T_B^2 T_S^2 + s^3 \left(T_B^2 \frac{T_S}{Q_T} + T_B \frac{T_S^2}{Q_L} \right) + s^2 \left[(\alpha + 1) T_B^2 + \frac{T_B T_S}{Q_L Q_B} + T_S^2 \right] + s \left(\frac{T_B}{Q_L} + \frac{T_S}{Q_T} \right) + 1} \quad (1)$$

Las ecuaciones 2 a 6 corresponden a la función de respuesta del tipo “Bass reflex”, y sus parámetros de alineamiento del sistema, los cuales son valores de diseño a optimizar.

$$G(s) = \frac{s^4 T_0^4}{s^4 T_0^4 + a_1 s^3 T_0^3 + a_2 s^2 T_0^2 + a_3 s T_0 + 1} \quad (2)$$

$$T_0 = (T_B T_S)^{1/2} = \frac{T_S}{h^{1/2}} \quad (3)$$

$$a_1 = \frac{Q_L + h \cdot Q_T}{h^{1/2} \cdot Q_L \cdot Q_T} \quad (4)$$

$$a_2 = \frac{h + (\alpha + 1 + h^2) Q_L Q_T}{h \cdot Q_L \cdot Q_T} \quad (5)$$

$$a_3 = \frac{h \cdot Q_L + Q_T}{h^{1/2} \cdot Q_L \cdot Q_T} \quad (6)$$

2.3 Optimización - Algoritmos Genéticos (AG)[6]

El algoritmo genético es un método para los problemas obligados y no obligados de optimización, que se basa en la selección natural, en el proceso de encontrar soluciones optimizadas, conduce a la evolución biológica.

El algoritmo genético modifica en varias ocasiones una población de soluciones individuales. En cada paso, el algoritmo genético selecciona a individuos al azar de la población actual para ser padres y los utiliza para producir a los niños para la generación siguiente.

Sobre sucesivas generaciones la población “se desarrolla” hacia una solución optima.

Se puede aplicar el algoritmo genético para solucionar una variedad de problemas de la optimización que no estén bien satisfechos para los algoritmos estándares de la optimización, e incluir problemas en los cuales la función objetiva es discontinua, no diferenciable, estocástica, o altamente no lineal.

El algoritmo genético utiliza tres tipos principales de reglas en cada paso para crear la generación siguiente de la población actual:

- Las **reglas de la selección** seleccionan a individuos, llamados los **padres**, que contribuyen a la población en la generación siguiente.
- Las **reglas de la cruce** combinan a dos padres para formar a **niños** para la generación siguiente
- Las **reglas de la mutación** aplican cambios al azar a los padres individuales para formar niños.

El algoritmo genético a diferencia de un clásico, derivado-basado, algoritmo de la optimización de dos maneras principales, según lo resumido en la tabla siguiente

Algoritmo Clásico	Algoritmo Genético
Genera un solo punto en cada iteración. La secuencia de puntos acerca a una solución óptima	Genera una población de puntos en cada iteración. El mejor punto de la población acerca a una solución óptima.
Selecciona el punto siguiente en la secuencia por un cómputo determinista.	Selecciona a población siguiente por el cómputo que utiliza los generadores del número al azar.

Tabla 1: principales diferencias entre algoritmos Clásicos y Genéticos.

2.3.1 Codificación del Dominio

En la naturaleza las características de los seres vivos, incluso aquéllas que los hacen óptimos para habitar en su medio, están determinadas por las proteínas que producen. A su vez, estas proteínas (o más bien, los aminoácidos que las forman) se codifican en el material genético contenido en cada una de las células del individuo. Así pues, la naturaleza ha mapeado cada posible solución al problema de crear un individuo óptimo en una secuencia particular de bases que producirá ciertas proteínas, ha *codificado* el dominio del problema (todos los posibles individuos) mapeándolo al conjunto de todas las posibles secuencias de nucleótidos.

Así, para un algoritmo genético lo primero que se requiere es determinar en qué espacio se encuentran las posibles soluciones al problema que se pretende resolver. En caso de tener un problema de optimización de una función cuyo dominio es un subconjunto de los números reales, entonces este subconjunto es al que nos referimos. Pero el algoritmo opera sobre “*códigos genéticos*”, sobre genotipos que se deberán mapear al espacio de soluciones.

Es decir, es necesario *codificar* de alguna manera el dominio del problema para obtener estructuras manejables que puedan ser manipuladas por el AG. Cada una de estas estructuras constituye el equivalente al genotipo de un individuo en términos biológicos.

El elemento del dominio del problema al que se mapea este genotipo es el análogo al fenotipo. Es frecuente que el código de los elementos del dominio del problema utilice un alfabeto binario (0's y 1's).

Una vez que se ha definido la manera de codificar los elementos del dominio del problema y se conoce la forma de pasar de un elemento a su código y viceversa, es necesario fijar un punto de partida. Los algoritmos genéticos manipulan conjuntos de códigos en generaciones sucesivas. Nuevamente haciendo una analogía, manipulan poblaciones de códigos. En éstas un código puede aparecer más de una vez. El algoritmo se encargará de favorecer la aparición en la población de códigos que correspondan a elementos del dominio que estén próximos a resolver el problema. En resumen, el algoritmo recibirá como entrada una población de códigos y a partir de ésta generará nuevas poblaciones, donde algunos códigos desaparecerán mientras que otros, que se mapean en mejores soluciones posibles, aparecen con más frecuencia hasta que se encuentra una satisfactoria o hasta que se cumple alguna otra condición de terminación. Los códigos en una población, es decir, los elementos de ésta serán llamados *individuos* y a los códigos en general, ya no en el contexto exclusivo de una población, se les denominará indistintamente *cromosomas*, *genotipo*, *genoma* o *código genético*, por analogía con los términos biológicos de donde surgen.

2.3.2 Evaluación de la población

Para determinar cuáles individuos corresponden a buenas propuestas de solución y cuáles no, es necesario calificarlos de alguna manera. Cada individuo de cada generación de un algoritmo genético recibe una **calificación** o, para usar el término biológico, una medida de su **grado de adaptación (fitness)**. Éste es un número real no negativo tanto más grande cuanto mejor sea la solución propuesta por dicho individuo. El objetivo de este número es que permita distinguir propuestas de solución buenas de aquéllas que no lo son. Si el problema a resolver consiste en maximizar una función, entonces la calificación asignada a un individuo determinado debe indicar qué tan alto es el valor de la función en el elemento de su dominio codificado por el individuo. Si, en cambio, el problema es determinar la ruta más corta entre dos puntos, la calificación deberá ser tanto más alta cuanto más corto sea el camino codificado en el individuo que esté siendo calificado.

2.3.3 Selección

Una vez calificados todos los individuos de una generación, el algoritmo debe, al igual que lo hacen la naturaleza y el hombre, seleccionar a los individuos más calificados, mejor adaptados al medio, para que tengan mayor oportunidad de reproducción.

De esta forma se incrementa la probabilidad de tener individuos “buenos” (con alta calificación) en el futuro. Si de una determinada generación de individuos se seleccionaran sólo aquellos con una calificación mayor o igual que cierto número c para pasarlos a la siguiente generación, es claro que en ésta la calificación promedio superará c y por tanto al promedio de la generación anterior.

La selección ocasiona que haya más individuos buenos, explota el conocimiento que se ha obtenido hasta el momento, procurando elegir lo mejor que se haya encontrado, elevando así el nivel de adaptación de toda la población.

En principio podría parecer que es conveniente tener una estrategia de selección estricta para que mejore rápidamente la población y converja el algoritmo, es decir, que la población se acumule alrededor de un genotipo óptimo. Esto no es cierto. Lo que ocurrirá es que la población se acumulará rápidamente alrededor de algún individuo que sea bueno, comparativamente con el resto de los individuos considerados a lo largo de la ejecución del algoritmo, pero este individuo puede no ser el mejor posible. A esto se le suele llamar **convergencia prematura**. No se puede asegurar pero sí procurar que lo anterior no ocurra. Además de la explotación es necesario que exista exploración. El AG debe, no sólo seleccionar de entre lo mejor que ha encontrado, sino procurar encontrar mejores individuos. A esto se dedican los operadores que serán descritos a continuación, los que aseguran que en todo momento exista cierto grado de variedad en la población, procurando con ello que no se “vicie”.

En la estrategia de selección normalmente se incluye un elemento extra que sirve de “ancla”. Si sólo se hace selección forzando que sea más probable elegir al mejor individuo de la población pero sin asegurarlo, es posible que este individuo se pierda y no forme parte de la siguiente generación. Para evitar lo anterior se fuerza la selección de los mejores n individuos de la generación para pasar intactos a la siguiente. A esta estrategia se le denomina **elitismo** y puede ser generalizada especificando que permanezcan en la población los n mejores individuos de las pasadas k generaciones

2.3.4 Cruzamiento

Durante la meiosis ocurre el proceso de producción de gametos. El código genético de los padres de un individuo se mezcla para producir gametos cuyo contenido genético es híbrido, es decir, una mezcla. De esta manera es posible que un individuo herede a sus descendientes las características mezcladas de sus propios padres, por ejemplo: el color de ojos del padre y el de cabello de la madre o, para aprovechar el ejemplo mencionado antes, es posible que una gacela herede la velocidad de su abuelo paterno y la fuerza de su abuela paterna, la salud de su abuelo materno y la agudeza visual de su abuela materna. Si estas características le confirieron a sus ancestros una alta aptitud de sobre vivencia, entonces este individuo será, con alta probabilidad, un individuo exitoso en su manada. La cruce de los códigos genéticos de individuos exitosos favorece la aparición de nuevos individuos que hereden de sus ancestros características deseables.

En el contexto de los algoritmos genéticos reproducirse significa que, dados dos individuos seleccionados en función de su grado de adaptación, éstos pasen a formar parte de la siguiente generación o, al menos, mezclen sus códigos genéticos para generar *hijos* que posean un código híbrido. Es decir, los códigos genéticos de los individuos se *crucan*. Existen muchos mecanismos de cruzamiento.

En esta sección sólo se presenta uno de ellos pero todos tienen por objeto que el código de un individuo *A* y el de uno *B*, previamente seleccionados, se mezclen, es decir, se fragmenten y recombinen para formar nuevos individuos con la esperanza de que éstos hereden de sus progenitores las características deseables.

El mecanismo de cruzamiento más común es el llamado *crucamiento de un punto*. Por ahora considérese como una analogía directa del proceso de cruzamiento descrito en la sección anterior, a propósito de la generación de gametos, con la formación de un único quiasma entre los cromosomas (individuos). A este quiasma es al que se le denominará más adelante *punto de corte*.

2.3.5 Mutación

Algunas veces, muy pocas de hecho, la ADN-polimerasa (la enzima encargada de replicar el código genético), se equivoca y produce una mutación, una alteración accidental en el código genético de los seres vivos.

Ocasionalmente algunos elementos del código de ciertos individuos de un algoritmo genético se alteran a propósito. Éstos se seleccionan aleatoriamente en lo que constituye el símil de una *mutación*. El objetivo es generar nuevos individuos, que exploren regiones del dominio del problema que probablemente no se han visitado aún. Esta exploración no presupone conocimiento alguno, no es sesgada.

Aleatoriamente se buscan nuevas soluciones posibles que quizá superen las encontradas hasta el momento. Esta es una de las características que hacen aplicables los algoritmos genéticos a gran variedad de problemas: no presuponer conocimiento previo acerca del problema a resolver ni de su dominio, no sólo en la mutación sino en el proceso total. De hecho, el problema a resolver sólo determina la función de evaluación y la manera de codificar las soluciones posibles (la semántica de los códigos genéticos de los individuos). El resto de los subprocesos que constituyen el algoritmo son independientes y universalmente aplicables.

2.4 Región de Búsqueda y Minimización del error

Para lograr una representación visual de la respuesta de frecuencia deseada se implemento una rutina grafica que permite el dibujo de las respuestas “genéricas” de cada tipo da caja a partir de los parámetros T-S medidos. Esto se consigue utilizando dos parámetros de diseño de la respuesta deseada, “frecuencia de diseño” y “Peak de amplitud”[1].

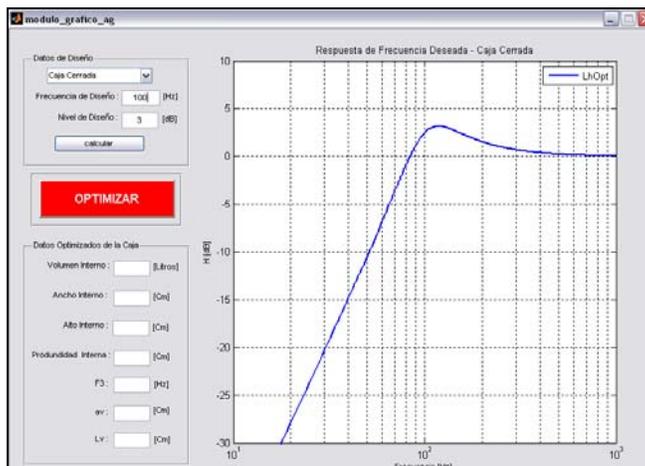


Ilustración 4: Respuesta de frecuencia deseada, diseño de caja cerrada.

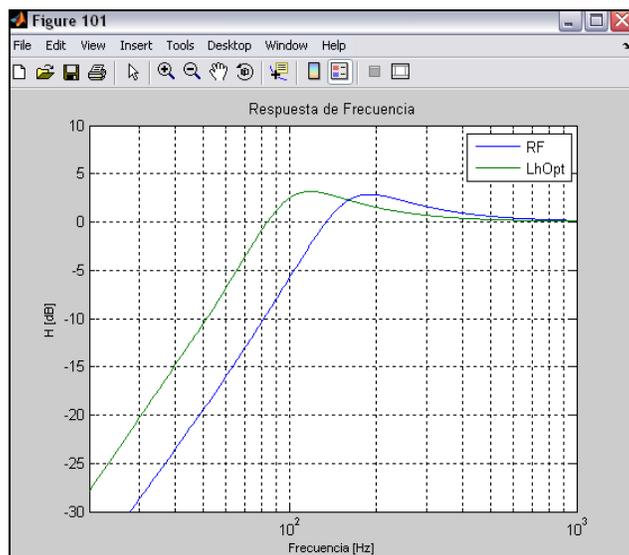


Ilustración 5: Iteración en la optimización de la curva.

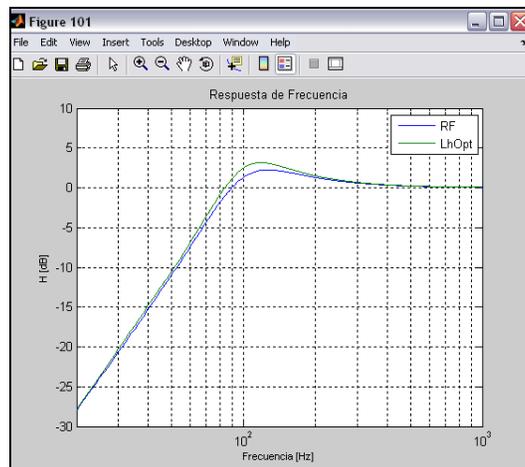


Ilustración 6: Iteración en la optimización de la curva.

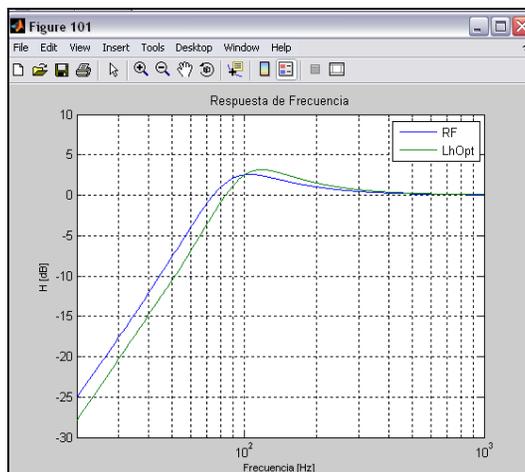


Ilustración 7: Iteración en la optimización de la curva.

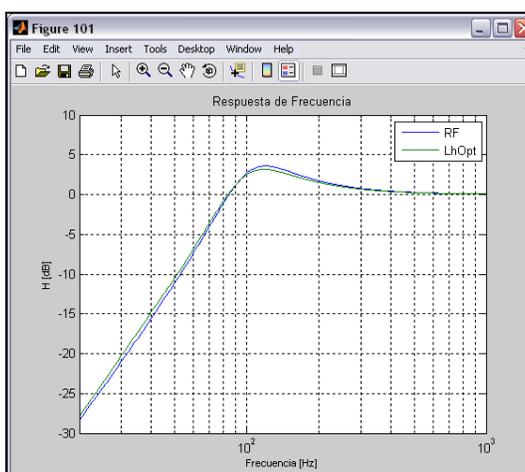


Ilustración 8: Iteración en la optimización de la curva.

En el caso de la caja cerrada, se definen 2 parámetros de diseño, α y $Q_L(1)$. Q_L esta determinado por el alineamiento deseado para la caja, por lo tanto queda fijo lo cual reduce la región de búsqueda para este parámetro, en el caso de α la región de búsqueda considerada toma como extremos de referencia, como mínimo la décima parte del volumen V_{AS} medido del altavoz y como máximo diez veces V_{AS} , de esta forma la diferencia entre la curva de diseño deseada es comparada en cada generación con la curva obtenida, calculando así el error cuadrático medio y almacenándolo de forma que se considere como solución valida aquella que reduzca el valor antes obtenido.

Para las cajas bass reflex, el procedimiento es similar, pero en este caso se consideran las variables a_1, a_2, a_3 del polinomio característico (2),(4),(5),(6). La obtención de los valores de a_1, a_2, a_3 son obtenidos a partir de la respuesta deseada y utilizados como extremos de la región de búsqueda del AG. La solución del alineamiento de la caja bass reflex se ha implementando de forma tal, que es alimentado directamente con los valores obtenidos en cada generación para a_1, a_2, a_3 controlándose el éxito del alineamiento mediante la comparación del valor de Q_T obtenido. Para garantizar que exista una población que entregue mutantes que permitan minimizar el error se toma como posible solución los individuos que entreguen un Q_T con una desviación máxima de un 5% de desviación.

3 RESULTADOS

Luego de realizar la búsqueda y optimización antes descritas, el modulo entrega las dimensiones internas de la caja a construir, es importante notar que la respuesta deseada y la respuesta obtenida son distintas pero aproximadas dependiendo de los parámetros T-S del altavoz utilizado.

Las ilustraciones siguientes dan cuenta del resultado tanto del diseño optimizado como de las mediciones de cada una de las cajas construidas.

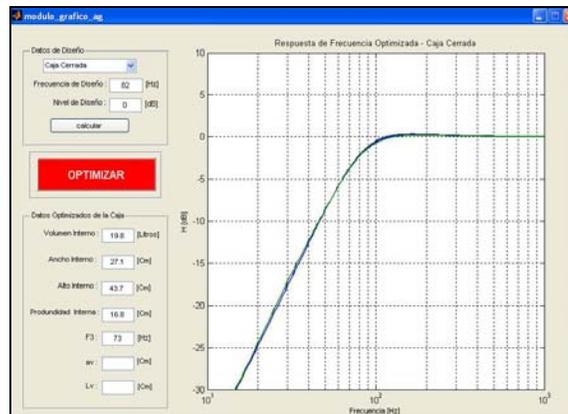


Ilustración 9: optimización y resultados para caja cerrada.

Como puede apreciarse el resultado es muy bueno en relación con la curva de diseño deseada, cabe destacar que las restricciones de diseño general de cajas se mantienen y que el altavoz utilizado en el calculo y optimización es la piedra angular ya que los todos los cálculos se realizan basándose en sus parámetros T-S.

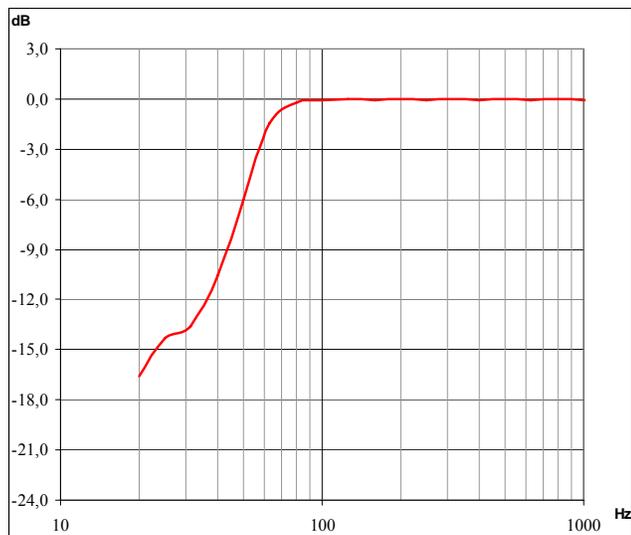


Ilustración 10: Respuesta medida, caja cerrada.

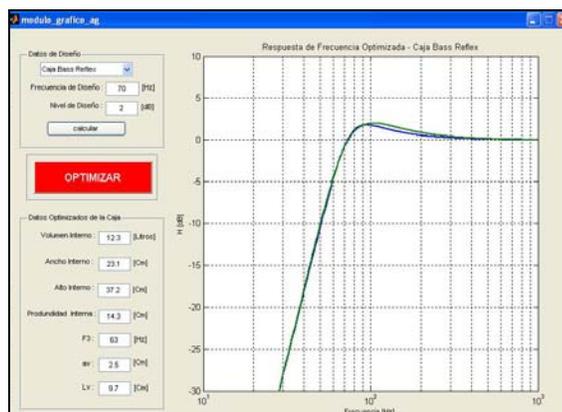


Ilustración 11: optimización y resultados para caja Bass reflex.

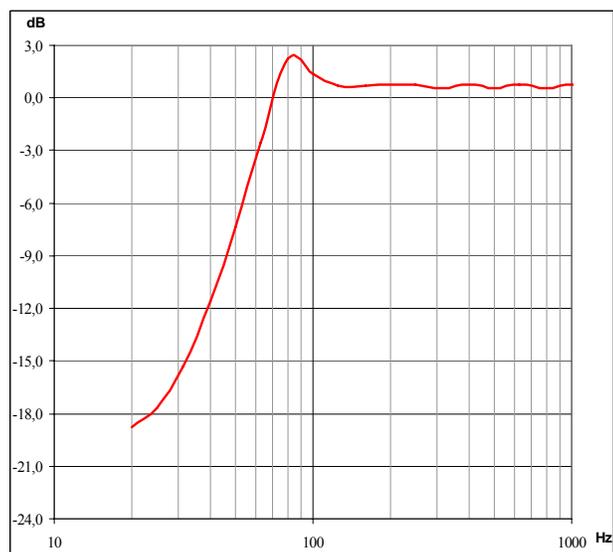


Ilustración 12: Respuesta medida, caja Bass reflex.

4 CONCLUSIONES

El proceso de diseño de cajas acústicas se ha realizado tradicionalmente tomado como datos de entrada ya sean volúmenes o razones de volúmenes, para obtener como resultado un valor discreto para la frecuencia de corte del sistema o una curva de la respuesta obtenida. El modelo de algoritmo genético presentado cambia esta forma “tradicional” de solucionar el problema de diseño, el cual se inicia con el diseño de la respuesta de frecuencia deseada y que termina con un volumen para los elementos de la caja y la respuesta “que es posible lograr” con el altavoz seleccionado. Desde ese punto de vista, el software presenta una gran herramienta de diseño con un potencial muy amplio y resultados concretos que avalan las predicciones realizadas.

Es posible hablar de una pequeña revolución del método de diseño ya que el sistema integra todas las etapas necesarias para lograr una caja funcional y optimizada desde la recopilación de datos del altavoz a utilizar, hasta el diseño y obtención de respuesta de frecuencia deseada.

5 REFERENCIAS

- [1] Olmos, Patricio; Páez, Ricardo; Barrera, Andres; Floody, Sergio (2008).“Diseño Geométrico Óptimo De Cajas Acústicas Basado En Técnicas De Algoritmo Genético”.
- [2] R. H. Small, (1972).“Direct-Radiator Loudspeaker System Analysis,” J. Audio Eng. Soc., vol. 20, pp. 383-395,.
- [3] R. H. Small,(1972-1973).“Closed-Box Loudspeaker Systems, Parts I and II” J. Audio Eng. Soc., vol. 20, p. 798-808 ,vol. 21, pp. 11-18.
- [4] R. H. Small,(1973).“Vented-Box Loudspeaker Systems, Part I, II, III IV” J. Audio Eng. Soc., vol. 21, pp. 363-372, 438-444, 549- 554, 635-639.
- [5] A. N. Thiele,(1971) "Loudspeakers in Vented-Boxes, Parts I and II" J. Audio Eng. Soc.,vol. 19, pp. 382- 392, pp. 471-483.
- [6] D. E. Goldberg,(1989).Genetic Algorithms in Search\ Optimization and Machine Learning, Cambridge, MA, Addison Wesley.
- [7] A. J. Keane, (1995).“Passive vibration control via unusual geometries the application of genetic algorithm optimization to structural design”, Journal of Sound and Vibration, Vol 185 (3), pp 441 – 453.
- [8] C.J Stuck (1987).“Determination of the Thielle-Small parameters using two-channel FFT analysis”, nd J. Audio Eng Soc., Preprint 2446 Present at the 82 Convention.
- [9] S. Müller, P. Massarini (2001).“Transfer-Function measurement with sweeps”, J. Audio Eng Soc., vol. 49, No. 6, pp 443.
- [10] H. Herlufsen, (1984).“Dual channel FFT Analysis (Part 1), Bruel & Kjaer Technical Review. N°1.
- [11] V. Dickason, (1997).“The Loudspeaker Design Cookbook”, Fifth Edition.
- [12] L. L. Beranek(1975).” Acoustics”,McGraw- Hill, New York.
- [13] H. F. Olson, (1943).”Dynamical Analogies”, Van Nostrand , New York.