

Diseño automático de filtros analógicos pasivos mediante un algoritmo genético

Aurora Torres Soto¹, María Dolores Torres Soto²,
Eunice Ponce de León¹, Braulio Jesús Montoya Padilla¹

¹ Universidad Autónoma de Aguascalientes,
Departamento de Ciencias de la Computación,
México

² Universidad Autónoma de Aguascalientes,
Departamento de Sistemas de Información,
México

{atorres, mdtorres, eponce}@correo.uaa.mx
a1188428@edu.uaa.mx

Resumen. En el área de diseño de circuitos analógicos, es común trabajar sobre plantillas de topología preestablecidas, en las que los elementos de circuito y sus conexiones no se alteran durante el proceso de optimización de sus valores [1, 2]; sin embargo, aún con topologías prometedoras, es común que éstas no conduzcan a resultados con buen desempeño, de manera que el proceso de optimización se debe comenzar desde el principio [3]. En este trabajo se propone la integración de un grupo de mecanismos de modelación evolutiva de circuitos, con un algoritmo genético para abordar su diseño de manera que se evolucionen tanto la topología como los valores de los elementos. Los resultados obtenidos son muy prometedores, pues el costo de procesamiento es razonable, no se generan circuitos no simulables y el desempeño de los filtros es comparable al obtenido mediante técnicas clásicas; además de que los circuitos generados ya exhiben valores comerciales de elementos de circuito, lo que reduce la brecha entre el dispositivo diseñado y el implementado.

Palabras clave: Algoritmo genético, circuitos analógicos, diseño automático, filtros, simulación, evolución de topología.

Automatic Design of Passive Analog Filters Using a Genetic Algorithm

Abstract. In the analog circuit design area, it is common to work on pre-established topology templates, in which the circuit elements and their

connections are not altered during the optimization process of their values [1, 2]; however, even with promising topologies, it is common that they do not lead to results with good performance, so the optimization process must be started from the beginning [3]. In this paper, the integration of a group of mechanisms for the evolutionary modeling of circuits is proposed, with a genetic algorithm to address their design in such a way that both the topology and the values of the elements evolve. The results obtained are very promising, since the processing cost is reasonable, non-simulable circuits are not generated and the performance of the filters is comparable to those obtained using classical techniques; in addition, the generated circuits already exhibit commercial values of circuit elements, which reduces the gap between the designed device and the implemented.

Keywords: Genetic algorithm, analog circuits, automated design, filters, simulation, topology evolution.

1. Introducción

Con el enorme desarrollo que los sistemas digitales han alcanzado en las últimas dos décadas, es común pensar que los sistemas analógicos han sido completamente desplazados por éstos; sin embargo, la parte analógica de los sistemas mixtos no solamente es insustituible, sino que además consume el 90% del tiempo de diseño de estos sistemas [4]. En la actualidad existe una gran variedad de herramientas de asistencia al diseño de circuitos digitales; desafortunadamente, la curva de crecimiento de este tipo de herramientas para el dominio analógico ha sido siempre más lenta y menos adecuada; probablemente porque el diseño analógico es menos sistemático, además de ser una tarea que requiere de conocimiento especializado y una amplia experiencia [5].

El diseño de circuitos analógicos, frecuentemente descrito como “el arte de crear un dispositivo eléctrico que cumpla un grupo de especificaciones previamente establecidas”; es un proceso que consta de dos pasos: el establecimiento de la topología del circuito y la determinación de los valores de los elementos que lo forman[6]. Mientras que en el primer paso se establece cómo se conectarán unos elementos con otros e incluso el tipo de elementos de circuitos que serán conectados; en el segundo paso se establece la magnitud de cada elemento de circuito.

Teniendo en cuenta, la enorme importancia del diseño de circuitos analógicos en el proceso de elaboración de circuitos integrados, así como su reconocido nivel de complejidad; especialmente en el establecimiento de la topología, en este trabajo proponemos la integración de un grupo de mecanismos de modelación en la construcción y prueba de filtros analógicos pasivos, aplicando un algoritmo genético. El algoritmo deberá establecer tanto la topología como los valores de los elementos partiendo de un diseño “de hoja en blanco”.

En la siguiente sección se expone el problema de diseño de circuitos en la que se incluye una breve descripción de los filtros analógicos, pues el sistema propuesto se ha puesto a prueba en el diseño de un filtro analógico pasa-bajas. En la sección 3 se

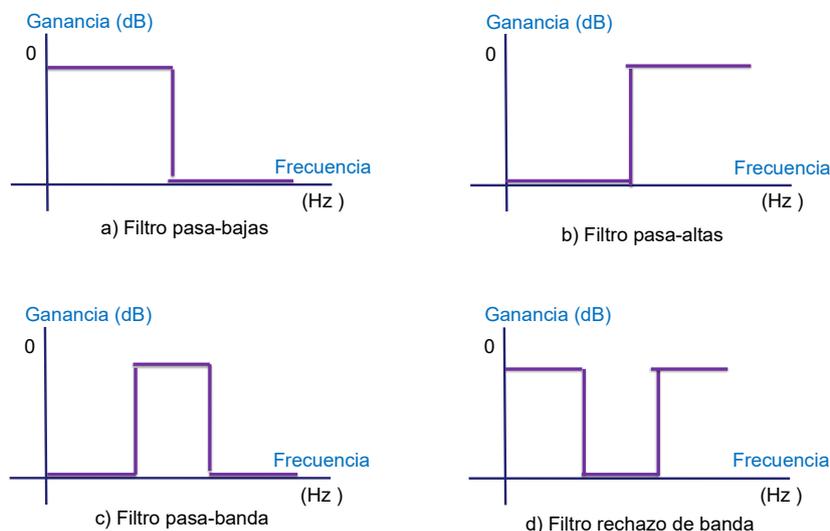


Fig. 1. Filtros ideales.

discuten brevemente los mecanismos de modelación y el algoritmo que los manipula. La sección 4 expone los experimentos realizados seguidos por los resultados y finalmente, en la sección 6 se discuten las conclusiones y el trabajo futuro.

2. Diseño de circuitos

El diseño de circuitos es el proceso que parte del establecimiento de un grupo de especificaciones de desempeño y termina con el dispositivo que exhibe el comportamiento deseado.

En función del tipo de señales que maneja un circuito, éste se puede clasificar como analógico, digital o mixto. Para este trabajo nos centraremos en los primeros. El área de los circuitos analógicos es extraordinariamente rica en variedad y versatilidad; sin embargo, todos comparten el hecho de que procesan señales analógicas o continuas; por esta razón, también son conocidos como sistemas de tiempo continuo. Este tipo de dispositivos tienen como objetivos clásicos: el filtrado, la amplificación y la fidelidad de las señales. Este trabajo se centra en el diseño de circuitos que realizan filtrado.

El problema que se afronta ha sido abordado empleando diferentes estrategias; inicialmente la tendencia fue abordar sus dos fases básicas (establecimiento de la topología y determinación de los valores de los elementos) de manera independiente; sin embargo, debido a que el establecimiento de la topología implica un problema mucho más retador que la optimización de los valores de elementos pre-colocados en un patrón; la optimización de los valores de un circuito fijo, fue la primera tarea abordada [1, 2] y en consecuencia, la primera en la que se emplearon técnicas de búsqueda no convencional [7–9].

Algunos de los primeros trabajos en los que se abordó el diseño desde la etapa de establecimiento de la topología se remontan a finales del siglo pasado [10-12]. Horrocks y Spittle [8] por ejemplo, abordaron el problema de la selección de los valores de los elementos de un filtro pasa-bajas activo usando solamente valores comunes de los elementos; idea que fue tomada posteriormente por Torres y sus colaboradores [13] para la elaboración de un mecanismo de representación que codifica tanto topología como dimensión de circuitos para ser evolucionados por herramientas evolutivas. Las herramientas de optimización y búsqueda heurística han sido protagonistas de numerosos esfuerzos por facilitar el diseño de circuitos durante los últimos veinte años.

Algunas de las metaheurísticas que han mostrado mayores beneficios en este campo son los algoritmos genéticos [6, 7, 14, 15] y la programación genética [12, 16]. Un filtro puede considerarse como un dispositivo selectivo a la frecuencia, que permite dar forma de una manera prescrita a la respuesta en magnitud o en fase [17]; es decir, es un circuito que bloquea las señales eléctricas de cierto rango de frecuencias, mientras permite el paso de aquellas con frecuencias en distinto rango [18].

Cuando un filtro se compone de elementos pasivos (resistencias, capacitores e inductancias), decimos que el filtro es pasivo y si posee elementos activos (transistores, amplificadores operacionales, entre otros); entonces es llamado filtro activo. Algunas de las distintas aplicaciones comunes de los filtros analógicos son: eliminación de ruido en señales de comunicaciones, detección de señales de radio y televisión, telefonía en DSL y audio, entre otras. De acuerdo con las bandas de frecuencia de las señales eléctricas que los filtros permiten pasar o que atenúan, éstos se clasifican en: pasa-altas, pasa-bajas, pasa-banda o rechazo de banda.

La figura 1 muestra el comportamiento ideal de estos circuitos. En esta figura se aprecia que la transición entre las bandas de paso (aquellas en las que la ganancia de voltaje es 0dB) y las de atenuación (aquellas en las que la ganancia es menor de 0dB), es instantánea; la señal de salida no presenta ruido y el grado de atenuación o paso de las señales es del 100% para los filtros ideales. Los filtros reales; sin embargo, se caracterizan por realizar las transiciones entre bandas más lentamente, además de que es muy común que las señales presenten ruido o rizo.

En la medida en la que un filtro se asemeja más al comportamiento ideal, es de mayor orden y por consiguiente de mejor calidad. Los filtros analógicos constituyen además un buen recurso de evaluación del desempeño de algoritmos, mecanismos y sistemas relacionados con el diseño automático de circuitos, pues éstos son los circuitos que los investigadores del área han utilizado con mayor frecuencia. De acuerdo con Goh y Li [7], el diseño de filtros es una disciplina bien comprendida; y teniendo presente que ha sido extensamente estudiada [19-23], su selección posibilita la comparación de los resultados de los mecanismos y algoritmos propuestos.

3. Descripción de la propuesta

Las técnicas de búsqueda y optimización heurísticas han mostrado ser muy eficientes en la solución de problemas complejos; sin embargo, exigen una modelación cuidadosa del fenómeno a estudiar. De acuerdo con Talbi [24], entre los aspectos comunes de las

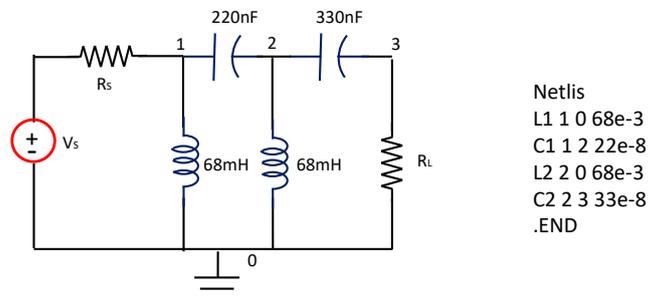


Fig. 2. Circuito eléctrico y su listado de conexiones.

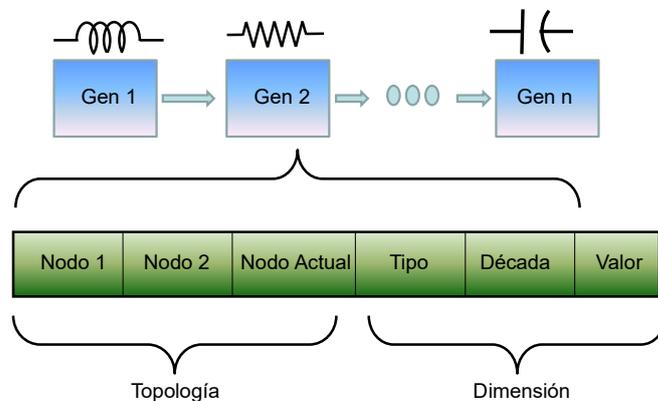


Fig. 3. Mecanismo de representación de circuitos.

metaheurísticas de mayor importancia están la representación de posibles soluciones y la función de adaptabilidad, que es quien conduce la búsqueda. A continuación, se describen tanto el mecanismo de representación, como el de evaluación que son empleados en esta propuesta.

3.1. Representación

En este trabajo, una solución al problema de diseño es una red eléctrica, constituida por elementos de circuito y nodos de conexión topológicamente correcta; es decir, que no debe contener conexiones a tierra desde ciertos nodos, ni tampoco dejar nodos desconectados. Adicionalmente, como se pretende que la herramienta sea capaz de establecer tanto su topología como los valores de los elementos, el mecanismo para su representación debe considerar la exploración de diferentes topologías y tamaños de los circuitos, así como una variedad suficiente de dispositivos comerciales. Finalmente, se pretende que los dispositivos generados por la herramienta sean simulados para comprobar su desempeño, así es que otra característica importante de la representación es su facilidad para la generación del “netlist” (palabra inglesa que significa “listado de conexiones”).

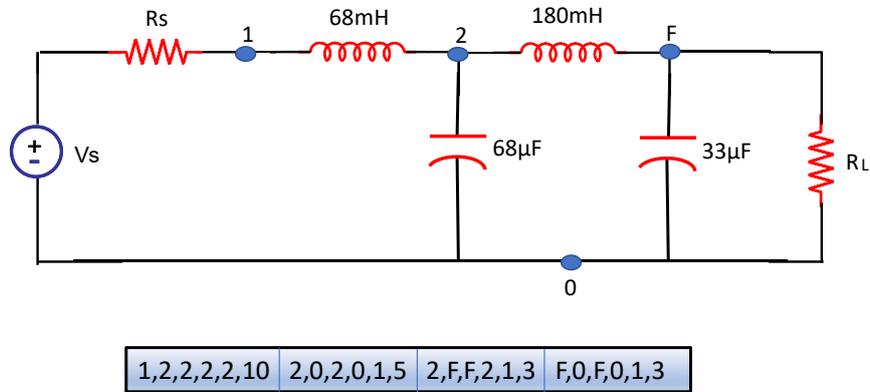


Fig. 4. Circuito y su codificación. Tomado de [13].

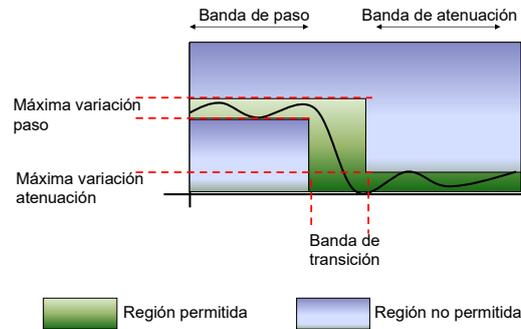


Fig. 5. Respuesta en frecuencia de un filtro pasa-bajas real.

Este listado de conexiones adicionalmente contiene instrucciones que se dictan al simulador para realizar un tipo de análisis. En la siguiente figura se muestran un circuito y su correspondiente netlis. Cabe hacer notar de la figura 2, que no se incluyen en la lista de conexiones la fuente de voltaje con su resistencia interna y la resistencia de carga.

Tomando como base la información contenida en las listas de conexiones y las otras consideraciones discutidas; en este trabajo se usa el mecanismo de representación descrito en [13].

Este mecanismo permite la construcción de circuitos de diferentes tamaños y topologías y adicionalmente usa dispositivos que son comercialmente comunes. La representación de un circuito (cromosoma) se almacena en una lista enlazada, cuya longitud corresponde con el número de elementos que tiene el circuito (genes).

Cada elemento de circuito se codifica en 6 campos que almacenan: los nodos de conexión y el tipo de elemento y su dimensión, como se muestra en la figura 3.

En relación con los valores de los elementos, se usan las series comerciales E6 para capacitores y E12 para inductancias y resistencias. El valor de los elementos se altera en función de la década (ver figura 3), que oscila entre $[10^{-5}, 10^{-9}]$ para capacitores, entre $[10^1, 10^6]$ para resistencias y entre $[10^{-1}, 10^{-6}]$ para inductancias; sin embargo, estos

Algoritmo Evolutivo General

1. Sea $G=0$ el contador de generaciones
 2. Crea e inicializa una población $P(0)$ n -dimensional
 3. Repite
 - Evalúa la aptitud $f(x)$, de cada individuo x , en la población $P(G)$
 - Produce la descendencia
 - Selecciona la población $P(G+1)$ de la nueva generación
 - Avanza a la nueva generación. $G=G+1$
 4. Hasta que la condición de paro sea cumplida
-

Fig. 6. Patrón general de un algoritmo evolutivo.

rangos pueden ser ampliados fácilmente. La figura 4 muestra un circuito y su codificación asociada.

De la figura anterior, cabe señalar que la codificación no incluye información de la fuente de alimentación V_S , de la resistencia de la fuente R_S y de la resistencia de carga R_L (pues esta parte es constante para todos los filtros); por lo que la codificación posee solamente 4 nodos (elementos de circuito) de 6 campos cada uno. Al lector interesado en profundizar sobre este mecanismo se le sugiere consultar [13].

4. Evaluación

Una vez que se ha resuelto el problema de creación de soluciones al problema que se aborda, es necesario establecer la forma como su calidad o desempeño será evaluada. Para este propósito las metaheurística emplean la función de adaptación o “fitness function”, que normalmente es un valor numérico que permite identificar las soluciones que mejor se adaptan al planteamiento del problema.

Si la herramienta de búsqueda realiza maximización, entonces en la medida que la solución es de mejor calidad se asociará con un valor mayor de esta función; y si la herramienta minimiza, entonces se espera que el valor de la función de adaptación sea menor a medida que mejora la calidad de la solución.

La función de adaptación es un elemento clave en el diseño de cualquier herramienta de búsqueda y/o optimización, pues es quien conducirá a la herramienta hacia zonas prometedoras del espacio de soluciones [24]; sin embargo, se debe tener presente que es también una de las partes más consumidoras de tiempo y recursos computacionales [25]. Una de las razones por las que el caso de diseño de circuitos analógicos no es una excepción a esta regla, se debe a que, para designar un valor de adaptabilidad, todos los individuos (circuitos) generados deben ser simulados.

En este trabajo se emplea SPICE (siglas de “*Simulation Program with Integrated Circuits Emphasis*”)[26], que actualmente es el software de simulación de circuitos de uso más extendido. Para definir la función de adaptabilidad, es necesario establecer los

parámetros que ésta maneja. Para ello nos basaremos en la respuesta en frecuencia del filtro pasa-bajas de la figura 5.

En esta figura se presentan los parámetros que deben ser considerados cuando se establecen las especificaciones de diseño de un filtro que tendrá un comportamiento real; en el eje de las abscisas se bosqueja a la frecuencia y en el eje de las ordenadas se tienen niveles de voltaje. A diferencia del filtro ideal (como los mostrados en la figura 1).

Un filtro real no realiza la transición entre bandas de forma instantánea; sino que lo hace de forma paulatina, consumiendo un rango de frecuencias conocido como “*banda de transición*”. El ancho de esta banda es un factor que normalmente está asociado con la complejidad y calidad del filtro. Si esta banda es estrecha, el filtro es mejor, aunque normalmente se constituye de un mayor número de elementos de circuito; y en la medida en que esta banda se vuelve más ancha, el filtro es menos complejo, aunque su desempeño sea menor.

También es común que exista cierta variación en la ganancia de voltaje tanto en la banda de paso, como en la de atenuación (rizado). Esta variación generalmente se acentúa en la medida que el circuito se vuelve más complejo. En la gráfica de la figura 5 se representa esta característica como “*Máxima variación paso y máxima variación atenuación*”. Cuando un filtro sobrepasa estos límites, la función de adaptabilidad debe verse afectada negativamente.

De esta forma, la región denotada como “*región permitida*” establece los límites aceptables para la presencia de señal eléctrica mientras que la “*región no permitida*” representa las zonas en las que la presencia de señal debe ser castigada. La función de adaptación diseñada y probada en este documento se encuentra en la ecuación 1:

$$f = \frac{1}{1 + \sqrt{\frac{\sum_{i=1}^N (\lambda \cdot \text{Error})^2}{N - 1}}}, \quad (1)$$

donde:

N: es el número de puntos de frecuencia para los que se realiza la simulación.

λ : es el valor del castigo que se deberá imponer en función de la frecuencia y la amplitud del voltaje de salida del filtro.

Error: Es la diferencia entre la respuesta en voltaje ideal y la que presenta el filtro diseñado.

La función de la ecuación 1 esta normalizada de manera que el valor para un filtro ideal sería de 1. Así, valores cercanos al 100% describen filtros de muy buena calidad.

5. Algoritmo GA-AF

La evolución natural puede ser vista como un proceso de optimización en el que el objetivo es mejorar la habilidad de organismos o sistemas para sobrevivir en un entorno dinámico y competitivo [27]. Debido al éxito que ha tenido la evolución en la mejora de las especies, la comunidad científica empezó a adoptar algunos de sus mecanismos en la solución de problemas complejos en la forma de lo que conocemos como

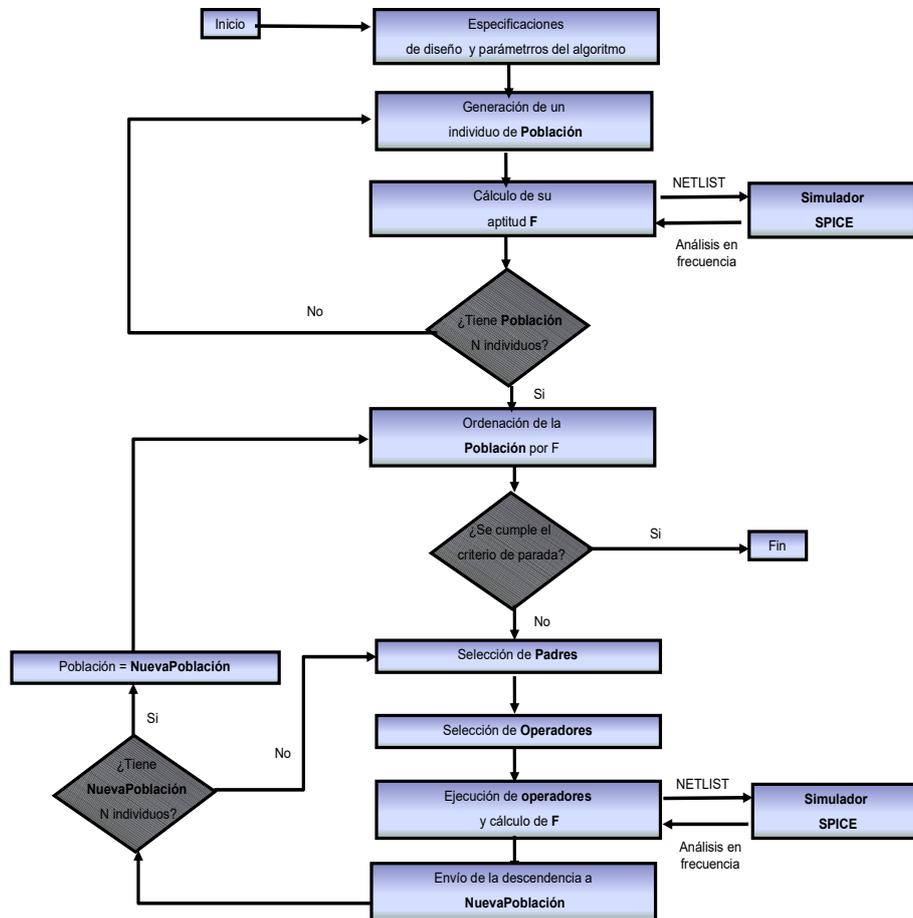


Fig. 7. Diagrama de flujo del AG-GA.

computación evolutiva. En la actualidad existe una amplia variedad de algoritmo evolutivos, sin embargo, todos tienen algunos pasos en común, al grado de que se puede establecer un algoritmo genérico con ellos.

La figura 6 muestra el esquema general de un algoritmo evolutivo. Entre algunos de los algoritmos más populares de la computación evolutiva podemos mencionar al algoritmo genético, cuyos principales precursores fueron John Holland [27] y David Goldberg [28]; la programación genética, inicialmente promovida por John Koza [16]; la programación evolutiva, difundida por Fogel [29]; las estrategias evolutivas, promovidas por Schwefel [30], los algoritmos de estimación de la distribución, descritos detalladamente por Pedro Larrañaga [31] y la evolución diferencial, discutida ampliamente por Vitaliy [32].

Sin embargo, los algoritmos evolutivos por excelencia son los algoritmos genéticos que inicialmente fueron llamado “planes reproductivos” por John Holland; pues fueron de los primeros emuladores de la evolución genética que produjeron resultados

Tabla 1. Especificaciones de diseño, simulación y evaluación.

Especificación	Descripción
Frecuencia de paso (F_p)	Establece el fin de la banda de paso cuando se trata de un filtro pasa-bajas (inicio de la banda de transición) y determina el inicio de la banda de paso (fin de la banda de transición) cuando se trata de un filtro pasa-altas.
Frecuencia de atenuación (F_a)	Establece el inicio de la banda de atenuación (fin de la banda de transición) cuando se trata de un filtro pasa-bajas y determina el fin de la banda de atenuación (inicio de la banda de transición) cuando se trata de un filtro pasa-altas.
Tipo de filtro	Establece el tipo que será diseñado. 0: Pasa-bajas. 1: Pasa-altas.
Objetivo de paso	Establece el valor de voltaje que debería exhibir el filtro ideal en la banda de paso.
Objetivo de atenuación	Establece el valor de voltaje que debería exhibir el filtro ideal en la banda de atenuación.
Error máximo	Establece el valor de la máxima variación de voltaje permitido tanto en la banda de paso como en la de atenuación.
Voltaje de la fuente (V_s)	Determina el valor de voltaje de corriente alterna que será aplicado al circuito para su simulación.
Resistencia de la fuente (R_s)	Es el valor de resistencia asociado con la fuente de alimentación.
Frecuencia inicial (f_i)	Frecuencia en la que se debe iniciar la simulación
Frecuencia final (f_f)	Frecuencia en la que debe terminar la simulación
Castigo (λ)	Es el valor del castigo que será aplicado para las frecuencias en las que el filtro tiene señales en la región no permitida.

prácticos. En 1989, cuando Goldberg publicó su libro [28], ya citaba más de 70 aplicaciones exitosas de este paradigma, que con el paso del tiempo sigue ganando popularidad. En este trabajo, se emplea un algoritmo genético que está basado en el algoritmo genético simple [28]. El diagrama de la figura 7 refleja el flujo de pasos que contempla el algoritmo GA-AF, cuyo propósito es diseñar automáticamente un filtro analógico pasivo que cumpla un grupo de requerimientos establecidos.

En la figura 7 se aprecia que el algoritmo presenta los pasos generales del conocido como SGA y adicionalmente, procesa la información de sus individuos y construye automáticamente el “*netlist*” con la descripción del circuito diseñado y la definición del tipo de análisis que debe ser ejecutado (especificaciones de diseño). Una vez que el simulador “*SPICE*” tiene el resultado, éste es procesado por el algoritmo para el cálculo de su función de adaptación “*F*”.

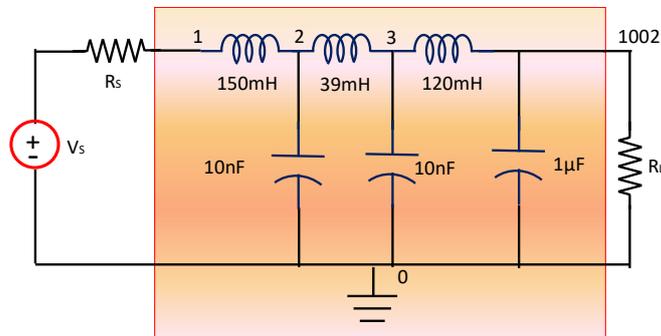


Fig. 8. Filtro pasa-bajas diseñado

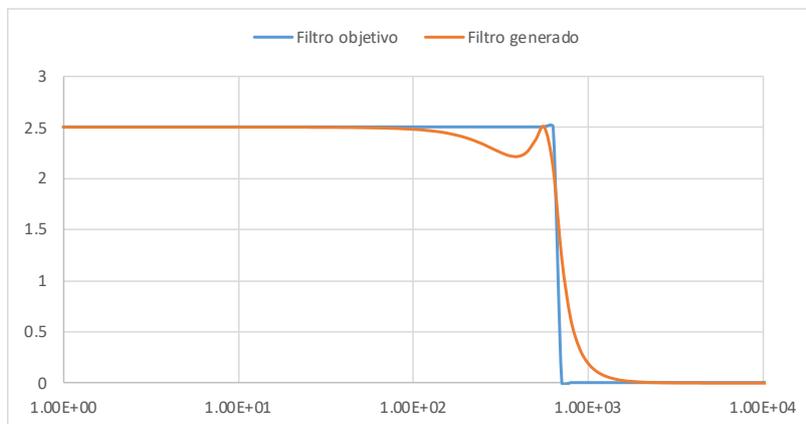


Fig. 9. Respuesta en frecuencia del filtro pasa-bajas.

Los parámetros del algoritmo son aspectos que definen exclusivamente la forma como el algoritmo genético debe ser ejecutado; tales como tamaño de la población, número de generaciones, probabilidad de cruzamiento y probabilidad de mutación. El criterio de paro que se está empleando inicialmente es un número máximo de generaciones.

En la tabla 1 se presentan los requerimientos de diseño de un filtro pasa-bajas, de acuerdo con las características mencionadas en la figura 5; las especificaciones necesarias para establecer el tipo de simulación que se debe realizar, y las especificaciones para la evaluación.

6. Experimentación

Una vez realizada la integración de todos los elementos que constituyen al algoritmo genético para el diseño de filtros analógicos “GA-AF”, se estableció un diseño factorial para el ajuste de los parámetros del algoritmo durante el proceso de diseño de un filtro pasa-bajas. Los parámetros manipulados y sus diferentes niveles se muestran en la tabla

Tabla 2. Diseño factorial.

Parámetro	Valores probados
Tamaño de la población	30, 50, 80, 100
Número de Generaciones	10, 15, 20, 25

Tabla 3. Resultados promedio.

Experimento Poblac_Gener	Evaluación	Tiempo	Número de elementos	Orden
30_5	0.72637	30.252499	4	1
30_15	0.773651	72.071495	7	1
30_20	0.754078	98.592506	5	1
30_25	0.825051	120.401405	4	2
50_5	0.719181	50.686996	4	1
50_15	0.785788	131.832794	5	2
50_20	0.789104	169.884979	5	1
50_25	0.819447	199.445297	4	2
80_5	0.751592	80.316002	4	1
80_15	0.844826	195.389313	4	2
80_20	0.822482	283.444275	5	1
80_25	0.828484	334.687836	5	2
100_5	0.803453	103.451096	4	2
100_15	0.839116	270.96344	4	2
100_20	0.859822	341.658234	6	2
100_25	0.855491	416.877502	5	2

2. Se realizaron 10 réplicas con cada combinación de parámetros, y se realizó un análisis de las variables de respuesta “Evaluación”, “Tiempo”, “Número de elementos”, “Orden”.

Los valores de probabilidad de cruzamiento “Pc” y probabilidad de mutación “Pm” se fijaron empíricamente en 0.9 y 0.3 respectivamente. Las especificaciones del objetivo de diseño son: filtro pasa-bajas con una banda de transición comprendida entre 700 y 750 Hz, con una variación máxima de voltaje en las bandas de paso y atenuación de 0.8V; se esta empleando una fuente de voltaje de corriente alterna con una resistencia interna de 500Ω y se supone que la carga tiene un valor de 500Ω también.

De acuerdo con los valores mostrados en la tabla 2, se realizó un total de 160 ejecuciones. La tabla 3 muestra los resultados promedio de las 10 réplicas para cada

combinación de los parámetros de entrada de este filtro. De la tabla anterior se aprecia que aparentemente el algoritmo exhibe su mejor desempeño con poblaciones de 100 individuos y 20 generaciones.

Luego de realizar un análisis a los resultados de los experimentos, se comprobó estadísticamente que no existe diferencia significativa entre las ejecuciones del algoritmo empleando 80 individuos y evolucionando durante 15 generaciones. Este análisis se realizó mediante la prueba de comparación de grupos independientes de Kruskal-Wallis.

7. Resultados

Para el diseño del filtro descrito en la sección 4, el GA-AF encontró el circuito que se muestra en la figura 8. Como se aprecia, este circuito es de aspecto comparable a aquellos diseñados por ingenieros expertos mediante el procedimiento de Chebyshev, y, sin embargo, fue elaborado sin fijar ni la topología ni los elementos de circuito, pues partió de una perspectiva de hoja en blanco. Con el objeto de validar que el diseño arrojado por el algoritmo efectivamente cumple con las especificaciones establecidas, en la figura 9 se presenta su respuesta bajo diferentes condiciones de frecuencia (línea roja) comparada con la del filtro ideal (línea azul).

Cabe mencionar que este filtro tiene un grado de atenuación de casi 24dB por octava, lo que describe a un filtro de orden 3. En esta figura se aprecia que el filtro ideal presenta la transición en 720 Hz aproximadamente, mientras que el diseñado sigue ese comportamiento general. A frecuencias bajas, el filtro permite el paso de la señal de voltaje (debajo de 700Hz) y por encima de 750Hz, la señal de voltaje se atenúa.

8. Conclusiones y trabajo futuro

Para este trabajo el objetivo principal fue la creación de circuitos que exhibieran el comportamiento dado por las especificaciones de diseño; razón por la cual, la función objetivo que conduce al algoritmo se basa exclusivamente en la respuesta de los circuitos a un análisis de frecuencia realizada mediante simulación.

El resultado obtenido exhibe un comportamiento de alrededor de tercer orden, lo que implica que tiene una banda de transición muy estrecha y, por consiguiente, posee un muy buen desempeño. El número total de circuitos que no fueron simulables se redujo a cero aún cuando el tamaño de los circuitos sugiere que no faltó diversidad en la búsqueda, lo que nos alienta a declarar que la clase de circuitos válidos se mantuvo cerrada durante el proceso de evolución.

Este hecho representa una enorme contribución en este campo, pues uno de los principales problemas reportados en el área, es el enorme costo computacional y tiempo que se deben invertir en la composición de redes eléctricas que no producen resultados como lo reporta Koza [12]. Los mecanismos de modelación empleados son simples y efectivos para mantener la clase de circuitos válidos cerrada.

El tiempo promedio que le lleva al GA-AF encontrar un circuito que se adapte a las especificaciones es de 5 minutos aproximadamente. Finalmente, el algoritmo propuesto

no requiere de grandes poblaciones ni un número considerable de generaciones para encontrar circuitos con una función de adaptación por encima de 0.85; los experimentos realizados sugieren que con poblaciones de 80 individuos evolucionando durante 15 generaciones son suficientes.

Como trabajo futuro, se pretende integrar un enfoque multiobjetivo en el proceso de diseño de circuitos, pues claramente, este es un problema en el que se deben contemplar al menos, complejidad del circuito, tiempo de diseño y desempeño. Adicionalmente se están haciendo las modificaciones necesarias a la función de adaptabilidad para trabajar con otros tipos de filtros.

Referencias

1. El Dor, A., Fakhfakh, M., Siarry, P.: Multiobjective differential evolution algorithm using crowding distance for the optimal design of analog circuits. *Journal of Electrical Systems*, vol. 12, pp. 612–622 (2016)
2. Sanabria-Borbón, A. C., Tlelo-Cuautle, E.: Sizing analog integrated circuits by combining gm/ID technique and evolutionary algorithms. In: *57th International Midwest Symposium on Circuits and Systems*, pp. 234–237 (2014)
3. Shi, G.: Toward automated reasoning for analog IC design by symbolic computation – A survey. *Integration, the VLSI Journal*, vol. 60, pp. 117–131 (2018) doi: 10.1016/j.vlsi.2017.08.005.
4. Ochotta, E. S., Rutenbar, R. A., Carley, L. R.: Synthesis of high-performance analog circuits in ASTRX/OBLX. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 15, pp. 273–294 (1996) doi: 10.1109/43.489099.
5. Das, A., Vemuri, R.: A graph grammar based approach to automated multi-objective analog circuit design. In: *Proceedings of the Conference: Design, Automation & Test in Europe*, pp 700–705 (2009) doi: 10.1109/DATE.2009.5090755.
6. Torres-Soto, A.: *Metaheurísticas evolutivas en el diseño de circuitos analógicos*. Benemérita Universidad Autónoma de Aguascalientes (2010)
7. Goh, C., Li, Y.: GA automated design and synthesis of analog circuits with practical constraints. *Proceedings of the 2001 Congress on Evolutionary Computation* (2001)
8. Horrocks, D., Spittle, M.: Component value selection for active filters using genetic algorithms. *Proceedings of IEE Workshop on Natural Algorithms in Signal Processing*, vol. 1, pp. 131–136 (1993)
9. Horrocks, D. H., Khalifa, Y. M. A.: Genetically derived filter circuits using preferred value components. *IEE Colloquium on Analogue Signal Processing*, (1994)
10. Zebulum, R. S., Pacheco, M. A., Vellasco, M.: Comparison of different evolutionary methodologies applied to electronic filter design. In: *IEEE International Conference on Evolutionary Computation Proceedings. IEEE World Congress on Computational Intelligence*, pp. 434–439, (1998) doi: 10.1109/ICEC.1998.699812.
11. Lohn, J. D., Colombano, S. P.: Automated analog circuit synthesis using a linear representation. In: Sipper, M., Mange, D., Pérez-Urbe, A. (eds) *Evolvable Systems: From Biology to Hardware*. ICES 1998. *Lecture Notes in Computer Science*, Springer, Berlin, Heidelberg, vol. 1478, pp. p 125–133, (1998) doi: 10.1007/BFb0057614.

12. Koza, J. R.: Genetic programming as a means for programming computers by natural selection. *Statistics and Computing*, vol. 4, pp. 87–112 (1994). doi: 10.1007/BF00175355.
13. Torres, A., Torres, M., Ponce de León, E.: Mecanismo de representación para la evolución automática de circuitos analógicos. *Research in Computing Science*, vol. 149, pp. 253–265 (2020)
14. Lohn, J. D., Colombano, S. P.: A circuit representation technique for automated circuit design. *IEEE Transactions on Evolutionary Computation*, vol. 3, no. 3, pp. 205–219, (1999). doi:10.1109/4235.788491.
15. Torres, A., Ponce de León, E. E., Torres, M. D., Padilla, F.: Comparison of two evolvable systems in the automated analog circuit synthesis. In: *Mexican International Conference on Artificial Intelligence*, pp. 3–8, (2009) doi: 10.1109/MICAI.2009.25.
16. Koza, J. R., Poli, R.: Genetic Programming. *Search Methodologies*. In: Burke, E.K., Kendall, G. (eds) *Search Methodologies*. Springer, Boston, MA. pp. 127–164. doi: 10.1007/0-387-28356-0_5.
17. Ambardar, A., Urbina, E., Nagore, G., Hernández, L.: *Procesamiento de señales analógicas y digitales*. 2^{da} Ed. Thomson Learning, México (2009)
18. Johnson, C. D.: *Process control instrumentation technology*. Publication History, (1999)
19. He, J., Zou, K., Liu, M.: Section-representation scheme for evolutionary analog filter synthesis and fault tolerance design. In: *Third International Workshop on Advanced Computational Intelligence*, pp 265–270 (2010) doi: 10.1109/IWACI.2010.5585181.
20. Kim, K. J., Wong, A., Lipson, H.: Automated synthesis of resilient and tamper-evident analog circuits without a single point of failure. *Genetic Programming and Evolvable Machines*, vol. 11, pp. 35–59 (2010) doi: 10.1007/s10710-009-9085-2.
21. Ando, S., Iba, H.: Analog circuit design with a variable length chromosome. In: *Proceedings of the 2000 Congress on Evolutionary Computation*, vol. 2, pp 994–1001 (2000) doi: 10.1109/CEC.2000.870754.
22. Sapargaliyev, Y., Kalganova, T.: Constrained and unconstrained evolution of “LCR” low-pass filters with oscillating length representation. In: *IEEE International Conference on Evolutionary Computation*, pp. 1529–1536, (2006) doi: 10.1109/CEC.2006.1688490.
23. Hu, J., Zhong, X., Goodman, E. D.: Open-ended robust design of analog filters using genetic programming. In: *GECCO '05: Proceedings of the 7th annual conference on Genetic and evolutionary computation*, pp. 619–1626 (2005) doi: 10.1145/1068009.1068283.
24. Talbi, E. G.: *Metaheuristics: From Design to Implementation*. Interfaces (Providence), vol. 42, pp. 414–415.
25. McPhee, N. F., Poli, R., Langdon, W. B.: *A Field Guide to Genetic Programming* (2008)
26. Fernández, H., Franco, Z. E.: Usando el PSPICE orientado al análisis de los circuitos del controlador booster instalado en CVG Venalum. *Universidad, Ciencia y Tecnología*, vol. 13, no. 51, pp. 085–095 (2009)
27. Holland, J. H.: *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control and Artificial Intelligence*. MIT Press, Cambridge, MA, USA (1992)
28. Goldberg, D. E., Holland, J. H.: *Genetic Algorithms and Machine Learning*. *Machine Learning*, vol. 3, pp. 95–99 (1988) doi: 10.1023/A:1022602019183.
29. Lawrence, J. F.: *On the Organization of Intellect*. University of California, Los Angeles, Engineering (1964)

Aurora Torres Soto, María Dolores Torres Soto, Eunice Ponce de León, et al.

30. Schwefel, H. P.: *Evolutionsstrategie und numerische Optimierung* (1975)
31. Larrañaga, P., Lozano, J. A.: *Estimation of Distribution Algorithms*. Springer US, Boston, MA (2002)
32. Feoktistov, V.: *Differential Evolution*. Springer US, Boston, MA (2006)