Artificial Evolution of Active Filters: A Case Study

Ricardo Salem Zebulum [1,2] Marco Aurélio Pacheco [2] Marley Vellasco [2]

1 - Centre for Computational Neuroscience and Robotics
School of Biological Sciences
University of Sussex, BN1 9QG Brighton, UK;
email: ricardoz@cogs.susx.ac.uk

2- Centro de Inteligência Computacional Aplicada (ICA)
Departamento de Engenharia Elétrica
PUC-Rio – Brasil
e-mail:salem@ele.puc-rio.br

Abstract

This article focuses on the application of artificial evolution to the synthesis of analog active filters. The main objective of this research is the achievement of a new class of systems, with advantageous features compared to conventional ones, such as lower power consumption, higher speed and more robustness to noise. The particular problem of designing the amplifier of an AM receiver is examined in this work. Genetic algorithms are employed as our evolutionary tool and two sets of experiments are described. The first set has been carried out using a single objective, the desired frequency response of the circuit. In a second set of experiments, three other objectives have been included in the system. A new multi-objective evaluation methodology was conceived for this second set of experiments. A second approach for evolving active filters, using programmable chips, is also discussed in this paper.

1 Introduction

The evolutionary design of circuits based on bipolar transistors is the main focus of this work. Bipolar transistor technology is still important for high speed applications in electronics. Particularly, our work addresses some important issues of electronic circuits' evolution, such as the *multiple-objective* nature of the task; the *speed* of the evolutionary tool to produce a design from scratch; and the *implementability* of the produced circuits.

The synthesis of an amplifier for AM band is performed in this work. This circuit works as a band-pass filter, amplifying incoming signals located in the AM frequency band. Particularly, many practical radio amplifiers are still implemented through bipolar transistors technology [Sansen98].

Due to the fact that analog design is more complex for automation than its digital counterpart [Johns97], the use of search techniques represents an interesting alternative. Recently, the evolutionary approach applied to analog design has been proposed by many authors, and promising results have been achieved [Koza98][Layzell98][Lohn98][Stoica98.

A Genetic Algorithm (GA) [Goldberg89][Holland75] with integer representation is employed as the evolutionary tool in these experiments. We present results for both single and multiple objectives tasks. In the former, the desired frequency response of the amplifier is the only objective taken into account; in the later, power dissipation, symmetric excursion and noise are taken into account as well.

Additionally, a comparison between the performance of genetic algorithms and hillclimbing [Blickle96] for this problem is provided. We also compare the application of extrinsic and intrinsic methods [Zebulum98] for circuits' evolution.

This work is organised in five additional sections. Section 2 describes the behaviour of the circuit to be synthesised. Section 3 presents the main features of our genetic algorithm, including representation and fitness evaluation function. Section 4 shows the results obtained using this approach. Section 5 provides a discussion on results obtained using a programmable analog chip during the circuits' evaluation step. Finally, section 6 concludes this work.

2 Problem Description

We analyse the particular problem of designing an amplifier for a radio receiver tuned in the AM frequency band. This active filter must amplify incoming signals inside the frequency band ranging from 0.15 to 1.6MHz [Sansen98], whilst attenuating signals outside this frequency band. Figure 1 depicts a schematic of the desired system:

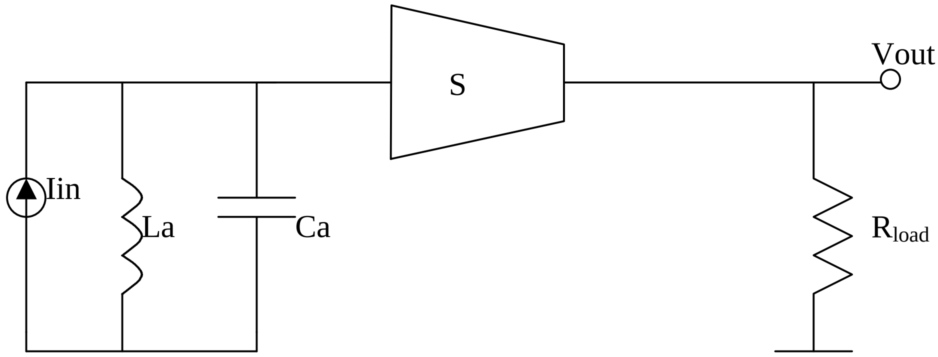


Figure 1 – Basic schematic of the AM receiver with inductive source.

In Figure 1, L_a represents the antenna inductance, and C_a is a parasitic capacitance, assuming values of 3.5mH and 7pF respectively [Sansen98]. The input signal, I_{in}, presents a magnitude of 1.5μA (around -117dB) [Sansen98]. The combination of L_a and C_a works as a trap to the incoming signal I_{in}. This trap has a resonant frequency at 1MHz. *S* represents the amplifier to be evolved, which must be tuned to the AM frequency band defined above. Finally, R_{load} is the load output to be driven.

Conventional design of active filters uses operational amplifiers as circuit building blocks. In contrast, this work does not enforce this conventional design principle, and low-level building blocks, such as transistors, resistors and capacitors, are utilised. Although this procedure increases the design complexity, novel circuits are more likely to be achieved. The discussion section of this article presents an additional case study, where operational amplifiers and switched capacitors are used as building blocks for the evolutionary system.

3 Evolutionary Algorithm

A three operator evolutionary algorithm, including selection, crossover and mutation operations, is employed. The application of an evolutionary algorithm to this problem encompasses the choice of an efficient *representation* and *fitness evaluation function*. Both are now described.

3.1 Representation

An integer representation based on a linear string has been employed. This representation has been used previously by the authors in the synthesis of operational amplifiers [Zebulum98]. Figure 2 depicts an example of this kind of genotype-phenotype mapping for a common emitter amplifier.

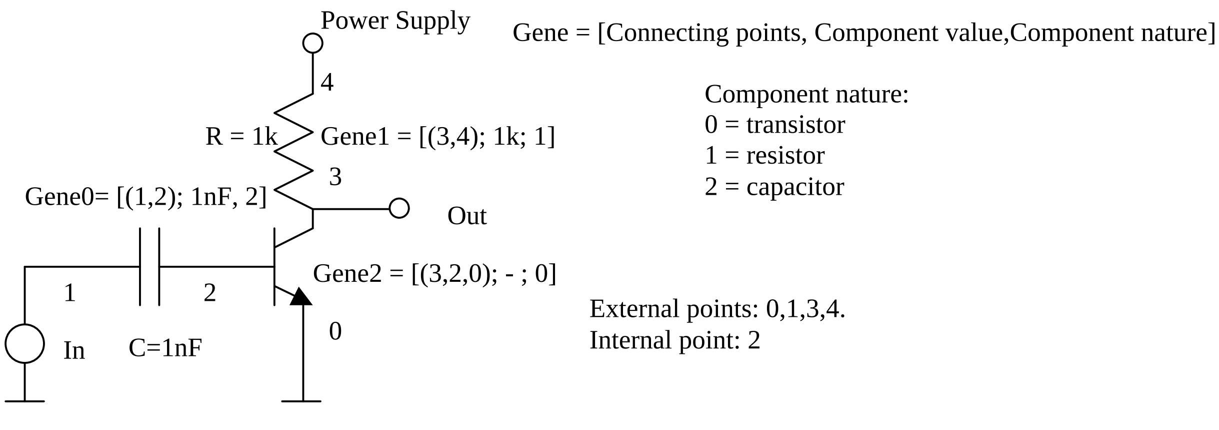


Figure 2 – Analog Circuit Representation

The genotypes are made up of genes, each of which encodes a particular component. The gene determines the nature, value and connecting points of the related component. First introduced in [Gribleby95] for the evolution of passive filters topologies, this representation is here extended to include the assignment of component values as well.

The total number of connecting points is a parameter to be set in this representation. This parameter is critical to the efficiency of the representation: if too few connecting points are considered, the number of possible topologies sampled by the evolutionary algorithm will be limited; conversely, if too many connecting points are considered, a higher number of unsimulatable topologies (with floating components) will arise. Additionally, each connecting point may be classified as internal or external. While the former does not serve for any special purpose, the latter is connected to one of the following signals: power supply, ground, input signal or probed output (Figure 2).

Due to the fact that a linear string is being used, we will refer to this evolutionary system as a genetic algorithm [Goldberg89] throughout this paper.

3.2 Evaluation

So far, most works concerning the evolution of analog circuits have used only one goal to be achieved by the evolved circuit. Nonetheless, the main challenge of applying genetic algorithms or any other search technique to analog design is the multi-objective nature of this task [Zebulum99]. Multi-objective optimisation concerns the need to integrate vectorial performance measures with the inherently scalar way in which most optimisation techniques reward performance. Because genetic algorithms require scalar fitness information on which to work, a scalarization of the objective vectors is always necessary [Fonseca95].

After testing some standard methods for multiple-objective optimisation, the authors devised a new one that is tailored for this class of problems. As it will be observed, our method is based on artificial neural networks learning algorithms [Churchland92].

Given a population of individuals, each one encoding an analog circuit in the way shown previously, a measure of performance or fitness is assigned to each individual in the following way:

$$Fitness = \sum_{i=1}^{n} w_i Fnorm_i \qquad (1)$$

According to the above equation, the fitness is computed by a weighted sum, where w is a weight vector; n is the number of objectives; and F_{norm} is the normalised fitness vector. This vector is defined by:

$$Fnorm_i = \frac{F_i}{\overline{F_i}} \qquad (2)$$

F_i is the individual's score with respect to a particular objective i, whereas the denominator of the above expression represents the average fitness, over all the individuals of the population, with respect to the same objective. This normalisation is accomplished to account for the fact that the objectives are measured in different units (decibels, Hertz, etc), and all of them must have the same influence in the fitness expression.

The main problem of this approach is the one of setting the weights' values. It is desirable to use a strategy in which the weights are dynamically updated according to the level of satisfaction of each objective; and also to take into account the user's specifications (design plan) for each particular objective.

Based on these guidelines, the following weight updating equation has been adopted:

$$w_{i,t+1} = a \cdot w_{i,t} + (1 - a) \cdot e_{i,t} \qquad (3)$$

The above equation uses an additional temporal index t, which points to a particular generation of individuals. Hence, $w_{i,t+1}$ is the next value of the weight associated to objective i. It is computed using its present value, $w_{i,t}$, and an error measure, $e_{i,t}$. This equation is based on the *Backpropagation* learning algorithm for *Artificial Neural Networks* (ANN) [Churchland92]. The term α used in equation 3 can take real values from 0 to 1, and will balance the contribution of the error and of the current weight value in the updating equation. Through much experimentation, we found that a value of α=0.8 (Equation 3) produced best results. This term is analogous to the momentum term used in the backpropagation algorithm, which is related to the stability of the learning process. The error $e_{i,t}$ provides a measure of the overall system performance for the particular objective i, and it is computed by:

$$e_{i,t} = \frac{|\overline{F}_{i,t} - User_i|}{User_i} \qquad (4)$$

Where $User_i$ represents the user specification for objective i. Therefore, the error is calculated by the difference between the average value for objective *i* over all individuals, and the user specification. The weights' values will then reflect the state of the system at the particular instant *t*.

All the weights are initialised with an equal arbitrary absolute value: if the corresponding objective needs to be minimised, the weight must take a negative initial value, and a positive value if the corresponding objective needs to be maximised.

Concerning the actual implementation of measurements in the circuits sampled by the GAs, there are two standards modes in which this procedure can be carried out, intrinsic and extrinsic assessments [Zebulum98]. In the former, each individual of the evolutionary algorithm is downloaded into a programmable chip [Stoica98][Thompson98], whereas, in the latter, simulators accomplish the performance measurements. This work concentrates on extrinsic evolution, providing, though, a discussion on results achieved in intrinsic evolutionary experiments.

4 Results

We present the results of two classes of experiments, processing a single objective and multiple objectives respectively.

4.1 Single Objective Experiments

This first class of experiments took only the desired frequency response into account. The fitness is given by the following equation:

$$Fitness = \sum_{i=1}^{n} w_i \cdot (V_{out}(i) - V_{in}(i)) \qquad (5)$$

Where V_{out} is the circuit output and V_{in} is the incoming signal at the input of the circuit *S*, shown in Figure 1. As these voltages are measured in decibels, $(V_{out}(i) - V_{in}(i))$ is the amplifier gain at a particular frequency *i*. *n* is the total number of output samples. The weights w_i take positive values for frequency points inside the AM band, and negative values outside this band. These weights are not related to the ones presented in section 3.2, since we are tackling a single objective optimisation here. The weights' values have been set through experimentation, assuming a value of +33 for frequency points between 150 kHz and 2 MHz (passing band); -4 for frequency points above 2 MHz; and –1 for frequency points below 150 kHz.

A chromosome made up of 12 genes has been employed in this set of experiments. The components' nature are chosen from four options: npn transistor; pnp transistor; resistor; and capacitor. Eight connecting points are available for the topology arrangement, four of them being external ones (input, output, power supply and ground). A power supply of 3V, which is typical of radio batteries, has been used. The evolutionary algorithm can choose among eight different values for resistors and capacitors respectively. The resistor values range from 75Ω to 500kΩ and the capacitor values range from 0.1nF to 100μF. The authors' strategy was to let a small number of different component values available to the GA, and keep the design space in a manageable size. As a consequence, an interactive involvement with an expert may be necessary to further improve the evolved circuits.

The size of the search space can be calculated from the above values. There are a around 4×8^3 different genes in this representation[1]. As each chromosome is constituted by 12 genes, one can conclude that the size of the search space is given by $(4 \times 8^3)^{12}$ possible solutions, which is around 10^{33}.

In order to sample this search space, we ran 10 executions of the GA, each one including 40 individuals and 100 generations. Each execution lasts around 30 minutes in a Sun Ultra Enterprise 2 server with one 300 MHz ultra sparc processor. The small signal analysis of the SPICE simulator has been used in this set of the experiments. It has been verified that most executions produced circuits that conformed well to the specification. Figure 3 depicts the schematic of one circuit achieved in this set of experiments; Figure 4 shows its frequency response.

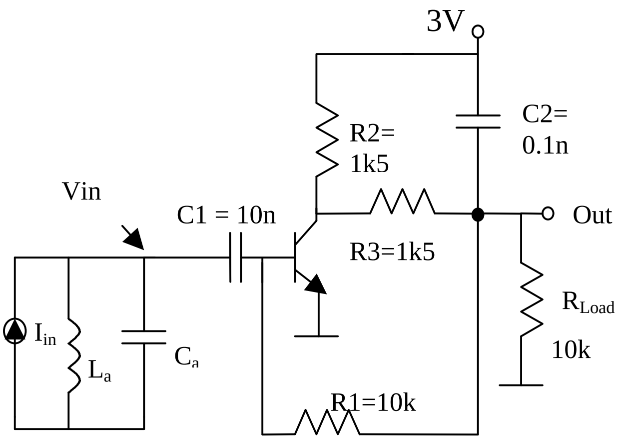


Figure 3 – Schematic of an amplifier obtained in the first set of experiments.

From Figure 3, it can be seen that the evolved solution uses only 6 components (not considering L_a, C_a, and R_{Load}); the other 6 components encoded in the chromosome were not effectively contributing to the

[1] Number of genes = (# different components) . (# connecting points)^2 (# different component values) = 4×8^3

circuit's behaviour. The most interesting aspect of this circuit is its parsimony, which stems from the fact that only one objective had to be fulfilled by the GA. It is also interesting to note that this amplifier is configured in a conventional common-emitter topology. The circuit connected in its collector, R2, R3 and C2, works as a low-pass filter, with cut-off frequency around 5 MHz; the capacitor C1 attenuates low frequency signals. The resistor R1 sets the DC operating point of the amplifier. Even though temperature variations were not taken into account in the fitness evaluation function, this biasing configuration is advantageous to compensate effects of temperature changes [Laker94].

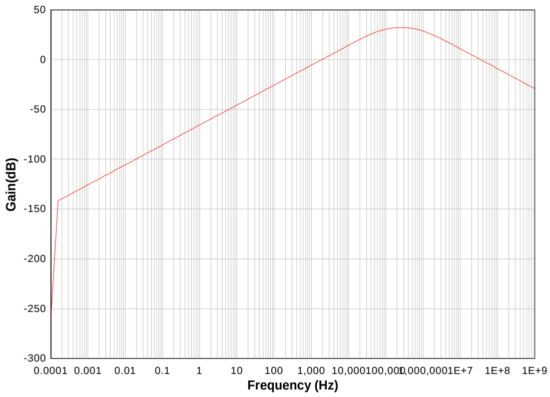


Figure 4 – Frequency response of the amplifier shown in Figure 3.

The graph of Figure 4 shows the amplifier's gain in the frequency domain. Focusing on the AM frequency band, the minimum gain achieved by the circuit is 28 dB, at 1.5MHz; and the maximum gain is 34dB, at 400kHz. Conventional circuits present an average minimum gain of 16dB; and an average maximum gain of 37dB [Sansen98].

Further design improvements can be accomplished by optimising the resistors and capacitors values. For instance, a simple inspection of the evolved circuit shows that reducing the value of C1 is a way to enhance the transfer function, by shifting the lower half of the circuit passing band from 20kHz to 50 kHz.

Finally, we performed a comparison between the performance of our GA and the hillclimbing search technique [Blickle96]. A total of 20 GA executions were performed, each one processing 40 individuals along 100 generations. The same number of individuals has been processed in the hillclimbing method, i.e., 20 executions processing one individual along 4000 generations. We note that the GA was optimised in terms of mutation rate (around 1 mutation per genotype) and of selection pressure (exponential selection with parameter c equal to 0.9 [Blickle96]). The graph of Figure 5 shows the average fitness obtained in both experiments. Although the GA outperformed hillclimbing, the latter performed surprisingly well for this problem.

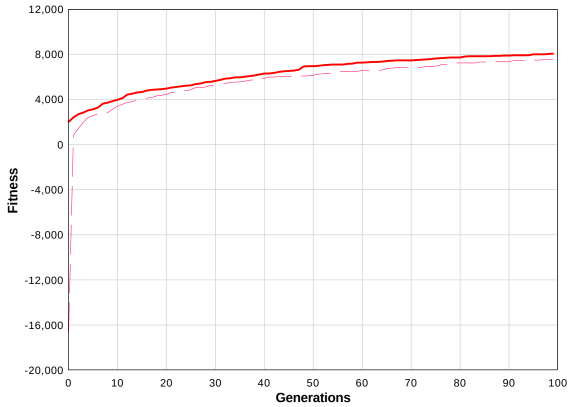


Figure 5 – Average fitness along the generations for two experiments: GA (full line) and Hillclimbing (traces). In the case of hillclimbing, the fitness values were taken within an interval of 40 generations, in order to match the number of 100 fitness points.

4.2 Multi-Objectives Experiments

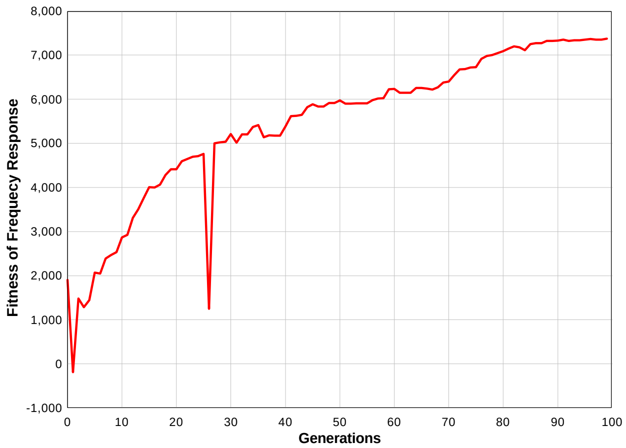
Although hillclimbing and GAs produced comparable performances in the single-objective experiments, the genetic algorithm is better tailored for the multiple-objective application. This is due to the fact that, as it is described in section 3.2, our multi-objective fitness evaluation method requires the computation of an average over a set of individuals (equation (4)). Since hillclimbing focuses on only one individual at a time, it can not be applied in the context of this technique.

Four objectives have been considered: the frequency response fitness, computed in the way shown in equation (5); the minimisation of the power dissipation; the maximisation of the Maximal Symmetric Excursion (MSE); and the minimisation of the integrated output noise. The power dissipation and the integrated output noise are directly measured by the simulator; the MSE is maximised by keeping the DC value of the output voltage, $V_{dc}(out)$, at half of the power supply value (1.5V); this is accomplished by *minimising* the quantity $|V_{dc}(out) – 1.5|$.

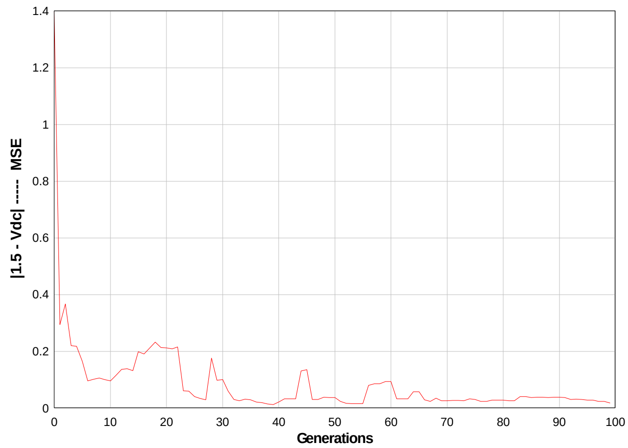
These four objectives are aggregated in the way shown in equation (1), and the weights are updated through the expression presented in equation (3). The design plan (user's specifications) used in this experiment was set to: fitness of the frequency response (equation (5)) equal to 10000; power consumption equal to 0.1 mW; value of $|V_{dc}(out) –$

1.5| equal to 0.5V, corresponding to a 1.0V excursion in the output; and integrated output noise equal to –120dB. We ran 10 GA's executions, each one including 40 individuals and 100 generations. The overall experiment lasted around 4 hours in one 300 MHz ultra sparc processor. The SMASH [SMASH93] simulator was used in this set of the experiments. The graphs of Figure 6 display the average values taken by the four objectives during this experiment. It can be seen that the GA tries to optimise the frequency response, MSE, and dissipation, keeping control of the output noise simultaneously.

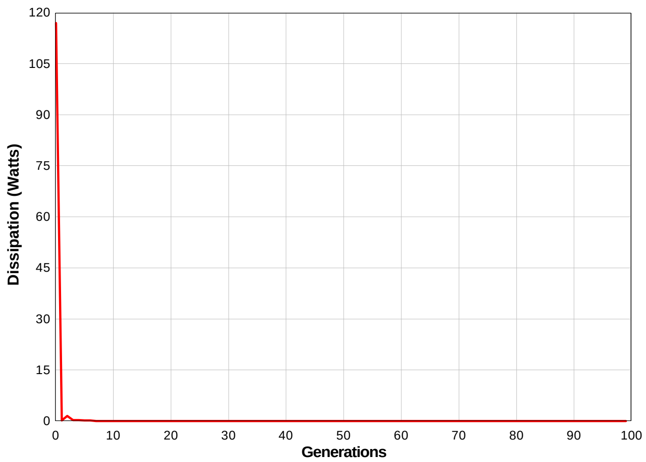
Figure 7 shows the schematic of the best circuit achieved in this second set of experiments.



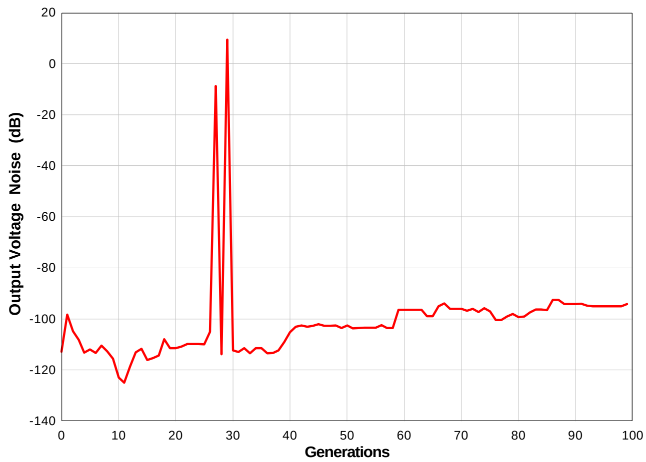
(A)



(B)



(C)



(D)

Figure 6 – Average values of the objectives along the evolutionary process: frequency response(A); MSE (B); Dissipation (C); Noise (D).

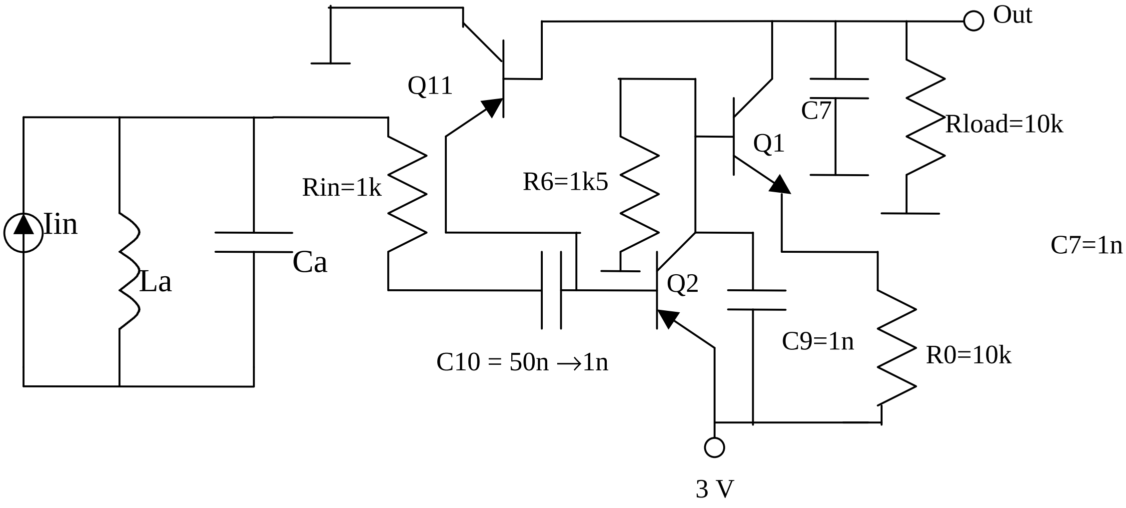
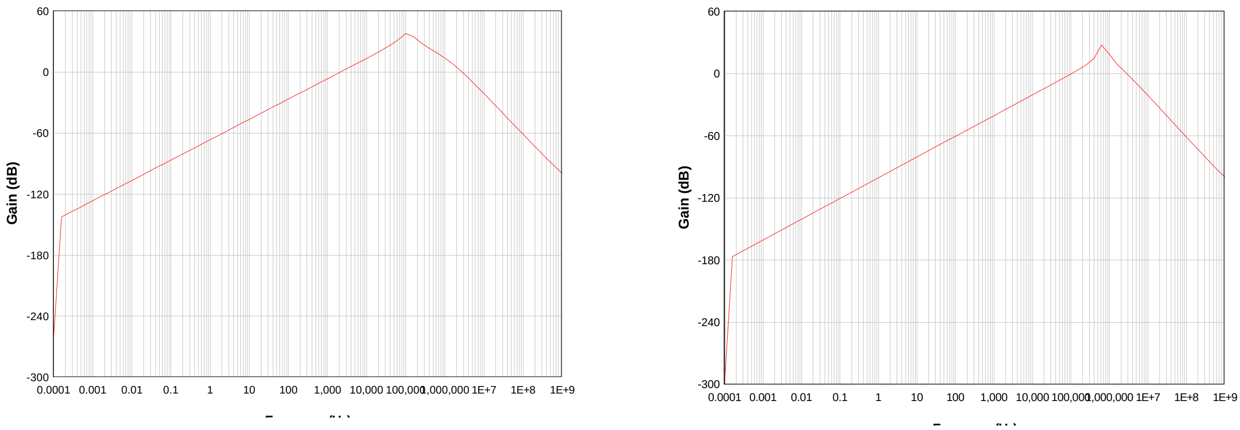


Figure 7 – Best amplifier achieved in the second set of experiments.

In the above circuit, transistors Q11 and Q2 work in the linear region and transistor Q1 works in the reverse region. Q2 is performing the signal amplification and delivering it to the pair Q1 and Q11, which is setting the DC output to the desired value. As shown in the schematic, this design can be improved by a simple inspection: the value of C_{10} may be decreased from 50nF to 1nF; and the input impedance of the amplifier may be increased by inserting R_{in}. These changes improve the passing band boundaries within the AM frequency region, by shifting it to the right. The graphs of Figure 8 show the frequency response of the circuit, with and without these changes.



(A) (B)

Figure 8 – Gain of the circuit shown in Figure 7: without changes (A); improved design (B).

We can draw a comparison between the performance of the circuit obtained in the first set of experiments and the improved version of the circuit shown in Figure 7. Both of them display a gain close to 30dB at the frequency of 500kHz. The design of Figure 7 presents an integrated noise in the output of -94 dB, against -83dB observed in the circuit of Figure 3. A reference value for noise in bipolar amplifiers can be taken from [Johns97], where the output noise for a common emitter amplifier, with DC operating point optimised for noise attenuation, assumed a value around –80dB at 300K (the same temperature used in our experiment). The power consumption of the above circuit is 5.0 mW, against 3.9mW of the circuit depicted in Figure 3. This difference is due to the fact that the amplifying transistor of the above circuit, Q2, is draining more current from the power supply than the single transistor of the first circuit. Finally, the MSE of the circuit in Figure 7 is around 1.5V (DC output value is 1.5V), against 0.9V for the circuit shown in Figure 3. Therefore, we can conclude that the second circuit is better in terms of output excursion and noise.

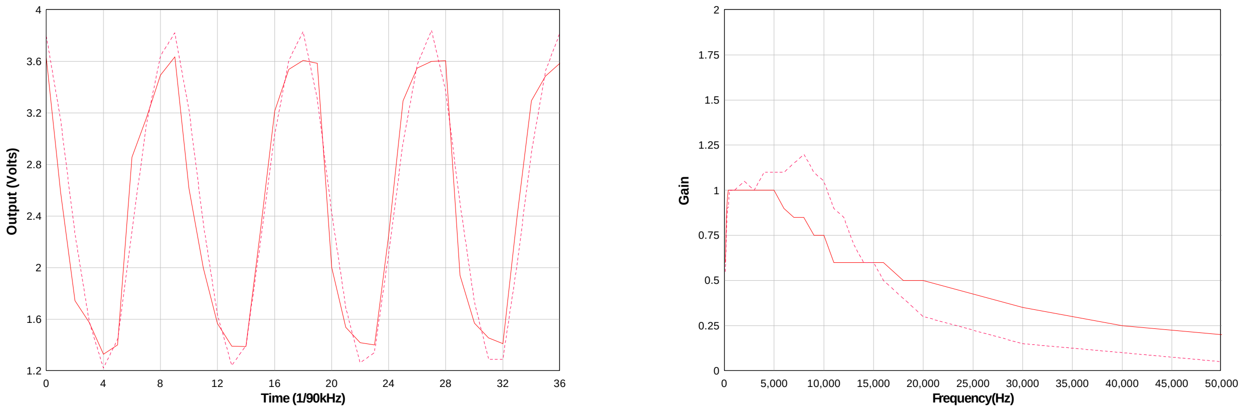
An important issue concerning circuits' evolution through simulation is their implementability. It has been verified, in a previous work [Zebulum98], that simulators might bias transistors in overvoltage and overcurrent conditions, which could not be reproduced in practice. The minimisation of the circuit dissipation along the evolutionary process, accomplished in this experiment, was the means whereby these conditions could be avoided.

5 Discussion

We present, in this final section, related results achieved through intrinsic evolution. We have used the field programmable analog array MPAA020, from Motorola [Motorola97]. This chip's architecture consists of an array of operational amplifiers, connected through switched capacitors. The GA controls the circuit connectivity, capacitors values and other programmable features encoded in a linear bitstring. Further details can be found in [Zebulum98].

In a particular experiment, we focused on the evolution of a biquad low-pass filter [Johns96], with cutoff frequency close to 10kHz. The GA was allowed to manipulate a chip's region including two operational amplifiers, which is the standard size of conventional biquad filters [Motorola97]. The fitness evaluation function was computed by comparing the transient response of the sampled circuits with the one of a target filter. This transient response was taken by applying a 10 kHz square wave to the circuit's input. The graphs of Figure 9 compare the time and frequency domain responses, respectively, of the evolved circuit and the target filter. This experiment processed an order of 10^3 individuals over many GA executions.

The transient analysis is more attractive than the frequency analysis for intrinsic evolution, because the evaluation step is less time consuming, not requiring the computation of the Fast Fourier Transform (FFT). Another approach to simplify the fitness evaluation function is to consider only frequency points inside the passing band. We ran another experiment focusing only on the circuit's gain at a particular frequency. A sine wave with amplitude equal to 1Vp-p and 2kHz frequency was applied to the circuit's input. The fitness evaluation function used as target output signal a 2Vp-p and 2kHz sine wave. The GA manipulated the connections of only one operational amplifier. The frequency response of the obtained circuit is plotted in Figure 10. It can be seen that the circuit behaves as a band pass filter, with maximum gain of 2.2, at the frequency of 2kHz. In this case, the amplifier gain is limited by feedback connections. Even though the fitness evaluation function does not enforce particular values of cutoff frequencies, we observed that most of the evolved solutions were narrow band amplifiers.



(A) (B)

Figure 9 – Intrinsic evolution - Comparison between the evolved circuit (line) and target filter (points) response: (A) – time domain; (B) – frequency domain.

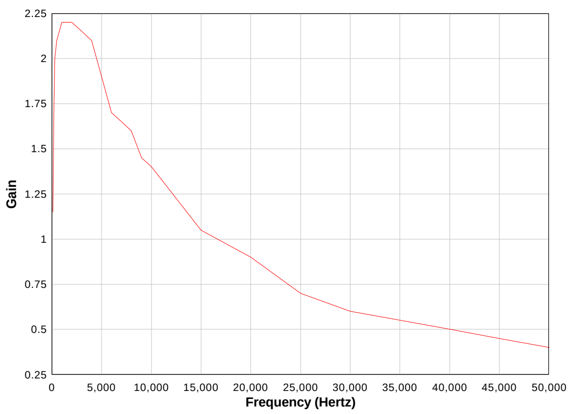


Figure 10 - Gain of the amplifier obtained in the experiment using the programmable analog chip.

Each GA execution of the experiments reported above lasted around 4 hours. Most of the execution time is dominated by the chip downloading time. The genotypes are around 500 bits long, resulting in a very large genome space ($\sim 10^{150}$), compared to the extrinsic experiments. One of the most important aspects in intrinsic evolution is the fact that the evolved circuits will always work in reality. However, the use of simulators, in the context of extrinsic evolution, provides a more straightforward way to accomplish multiple performance measures, such as frequency response, dissipation and output noise.

6 Conclusions

This work investigated the application of genetic algorithms to the synthesis of an active filter for AM band. A new method to aggregate multiple specifications into the fitness function was introduced. Four objectives have been addressed in our experiments: frequency response; power consumption; symmetric excursion; and output noise. The search space sampled in these experiments has been limited, by restricting the number of possible component values. Conversely, we have followed a particular strategy for design improvement, consisting of inspection and small changes in the evolved circuit. The authors are currently studying problems of impedance coupling between the trap (L_a and C_a) and the evolved amplifier, by including this as another objective of the evolutionary process. The results are encouraging when compared to conventional circuits [Sansen98].

In another set of experiments, a programmable analog chip was applied to the intrinsic evolution of filters. The authors have been investigating different approaches for fitness evaluation; currently, the use of the FFT is being studied.

Acknowledges

The authors wish to thank CAPES, brazilian federal agency, and Motorola for the support; and Dr. Adrian Thompson and Mr. Paul Layzell, from the University of Sussex, for the important suggestions.

References:

[Blickle96] - Blickle, T., "Theory of Evolutionary Algorithms and Application to System Synthesis", Doctoral Thesis, Swiss Federal Institute of Technology, Zurique, 1996.

[Churchland92] - Churchland, P. S., Sejnowski, T. J., "The Computational Brain", MIT Press, 1992.

[Fonseca95] - Fonseca, C. M, Fleming P. J. , "An Overview of Evolutionary Algorithms in Multiobjective Optimization", Evolutionary Computation, MIT Press, Vol. 3, No.1, pp.1-16, 1995.

[Goldberg89] - Goldberg, D., "Genetic Algorithms in Search, Optimization and Machine Learning", Addison-Wesley Publishing Company, Inc., Reading, Massachusetts, 1989.

[Grimbleby95] - Grimbleby, J. B. , "Automatic Analogue Network Synthesis Using Genetic Algorithms", Proceedings of the First IEE/IEEE International Conference on Genetic Algorithms in Engineering Systems (GALESIAS - 95), pp.53-58, UK, 1995.

[Holland75] - Holland, J. H., "Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology", Control, and Artificial Intelligence. Ann Arbor, MI: University of Michigan Press 1975. Second edition. Cambridge, MA: The MIT Press 1992.

[Johns97] - Johns, D. A., Martin, K., "Analog Integrated Circuit Design", John Wiley & Sons, Inc., 1997.

[Koza98] - Koza, John R., Bennett III, Forrest H, Andre, David, Keane, Martin A, and Dunlap, Frank, "Automated synthesis of analog electrical circuits by means of genetic programming", IEEE Transactions on Evolutionary Computation. 1(2). pp. 109-128, 1998.

[Laker94] - Laker, K. R., Sansen, W., "Design of Analog Integrated Circuits and Systems", Mc. Graw-Hill Inc., 1994

[Layzell98] – Layzell, P., "A New Research Tool for Intrinsic Hardware Evolution", in Proceedings of the Second International Conference on Evolvable Systems: From Biology to Hardware (ICES98), Lausanne, Switzerland, September, 23-26, 1998. M.Sipper, D.Mange and A. Pérez-Uribe (editors), vol. 1478, pp. 47-56, LNCS, Springer-Verlag, 1998.

[Lohn98] – Lohn, J. D., Colombano, S. P., "Automated Analog Circuit Synthesis Using a Linear Representation", in Proceedings of the Second International Conference on Evolvable Systems: From Biology to Hardware (ICES98), Lausanne, Switzerland, September, 23-26, 1998. M.Sipper, D.Mange and A. Pérez-Uribe (editors), vol. 1478, pp. 125-133, LNCS, Springer-Verlag, 1998.

[Motorola97] - Motorola Semiconductor Technical Data, "Advance Information Field Programmable Analog Array 20-Cell Version MPAA020", Motorola, Inc., 1997.

[Sansen98] - Sansen, W., "Low-Noise Wide-Band Amplifiers in Bipolar and CMOS Technologies", in notes on the course: "Low-noise Analog CMOS &BiCMOS Design", promoted by the Imperial College, February, 25-27, London, 1998

[SMASH93] - "SMASH User and Reference Manual", Dolphin Integration, France, 1993.

[Stoica98] - Stoica, A., Fukunaga, A., Hayworth, K., Salazar-Lazaro, C., "Evolvable Hardware for Space Applications", in Proceedings of the Second International Conference on Evolvable Systems: From Biology to Hardware (ICES98), Lausanne, Switzerland, September, 23-26, 1998. M.Sipper, D.Mange and A. Pérez-Uribe (editors), vol. 1478, pp. 166-175, LNCS, Springer-Verlag, 1998.

[Thompson98] – Thompson, A., "On the Automatic Design of Robust Electronics Through Artificial Evolution", in Proceedings of the Second International Conference on Evolvable Systems: From Biology to Hardware (ICES98), Lausanne, Switzerland, September, 23-26, 1998. M.Sipper, D.Mange and A. Pérez-Uribe (editors), vol. 1478, pp. 13-24, LNCS, Springer-Verlag, 1998.

[Zebulum98] - Zebulum, R.S., Pacheco, M.A., Vellasco, M., "Analog Circuits Evolution in Extrinsic and Intrinsic Modes", in Proceedings of the Second International Conference on Evolvable Systems: From Biology to Hardware (ICES98), Lausanne, Switzerland, September, 23-26, 1998. M.Sipper, D.Mange and A. Pérez-Uribe (editors), vol. 1478, pp. 154-165, LNCS, Springer-Verlag, 1998.

[Zebulum99] - Zebulum, R.S, Pacheco, M.A., Vellasco, M., "A Novel Multi-Objective Optimisation Methodology Applied to the Synthesis of CMOS Operational Amplifiers", in review by the Journal of Solid-State Devices and Circuits, ISSN 0104 – 9631, edited by Wilhelmus Van Noije.