Current Mode Circuits: Increasing the Chances of Evolution to Find a Way

Ricardo S. Zebulum
M. I. Ferguson

Adrian Stoica
Vu Duong

Didier Keymeulen
Taher Daud

Jet Propulsion Laboratory
California Institute of Technology
4800 Oak Grove Drive
Pasadena, CA 91109
ricardo@brain.jpl.nasa.gov

Abstract—Evolution of electronic circuits has been intensively investigated for the last five years. The main challenge of the area is the evolution of circuits for industrial applications and also to find methods to improve the performance of Evolutionary Computation in electronic circuit synthesis. The authors have been studying this problem using different approaches, such as new methods for circuit representation and fitness evaluation function. We describe in this paper experiments on the evolution of current mode circuits. Experimental results in the area of computational circuits suggest that the search space for this class of circuits is more amenable for evolution than their voltage mode counterparts. At the end of the paper we propose an application in the area of fuzzy control, which can provide benefits for the aerospace community, such as in applications for propulsion controlled aircrafts.

TABLE OF CONTENTS

1. INTRODUCTION
2. EVOLVABLE HARDWARE: AN OVERVIEW
3. PROPOSED METHOD
4. EXPERIMENTS
5. APPLICATIONS FOR PROPULSION CONTROLLED AIRCRAFT
6. CONCLUSIONS

1. INTRODUCTION

This article describes new results achieved by the authors in the area of evolutionary design of electronic circuits (Evolvable Hardware). Particularly, we focus on the problem of finding new methods that will allow the automatic design of circuits for industrial and aerospace applications.

Evolvable Hardware (EHW) encompasses applications in which Evolutionary Computation is applied to the design of electronic circuits [1]. In addition to design automation, EHW brings many other advantages to electronic design, such as yielding low-power circuits [2], fault tolerant design [3], polymorphism [4], circuits for extreme temperatures [5], and synthesis of compact fuzzy controllers. Other Evolvable Hardware applications found in the literature refer to the automatic design of certain electronic systems building blocks, such as amplifiers [6], filters [7], logic gates [8], digital multipliers [9] and digital-to-analog converters [10]. Sometimes evolution was able to find novel circuits and in other experiments human designed circuits were rediscovered by evolution. In this paper we focus on the synthesis of computational analog circuits and on fuzzy controllers, the latter being a promising case study for industrial applications.

In a broader domain, we study new alternatives to improve the performance of Evolutionary Computation when applied to circuit synthesis. This problem may be tackled by changing the representation, the fitness evaluation function, or by posing the same problem in

[1] 0-7803-7231-X/01/$10.00/© 2002 IEEE
IEEEAC paper #008, Updated November 17, 2001

a different way, such as changing the physical quantity to be measured at the circuit output.

This paper also proposes future applications of evolved fuzzy controllers in aircraft control systems, such as computer-controlled engine thrust to provide emergency flight control capability.

This paper is organized as follows: section 2 provides a brief overview of Evolvable Hardware; section 3 shows the method devised by the authors and applied to the experiments presented in this article. Section 4 describes results on the evolution of computational circuits, such as Gaussian circuit, a multiplier and a fuzzy controller. Section 5 discusses the benefits of this approach for the design of flight control systems and section 6 concludes the article.

2. Evolvable Hardware: An Overview

Evolutionary Algorithms are search tools whose main operators are inspired by the natural evolution. Among these algorithms, we selected Genetic Algorithms (GAs) to handle the proposed problem.

GAs carry out search through biological evolution simulation. Instead of focusing on just one potential solution to the problem, they sample a population of potential solutions. A population of individuals is initially randomly generated. Each individual is a string that encodes, by means of a particular mapping, a potential solution to the problem. Individuals are also denominated chromosomes. The GA performs then operations of selection, crossover and mutation over the individuals, corresponding, respectively, to the principles of survival of the fittest, recombination of genetic material and mutation observed in nature. The *selection* step is probabilistic, but it favors individuals that have been assigned higher *fitness* indexes in an evaluation step performed beforehand. The fitness is a scalar measure of the performance of an individual according to the problem specification. The *crossover* operator splices the contents of two randomly chosen strings, producing two new individuals or *offspring*. The *mutation* operator changes a particular string position at random and it is applied with a low probability. The search process is carried out through the generation of successive populations until a stop criteria is met. It is expected that the average population fitness will gradually increase along the generations.

The genetic modeling developed to tackle our problem comprises the electronic circuit *representation* and its *evaluation*. They are both described next.

2.1 – Representation

The representation establishes a straightforward mapping between the electronic circuit topology and the integer strings processed by the GA. Each functional block of the string, also called gene, states the nature, value, and connecting points of a correspondent electronic component, which may include resistors, capacitors, bipolar transistors and MOS (Metal-Oxide-Semiconductor) transistors.

Figure 1 depicts an example of this kind of chromosome-circuit mapping for a common emitter amplifier.

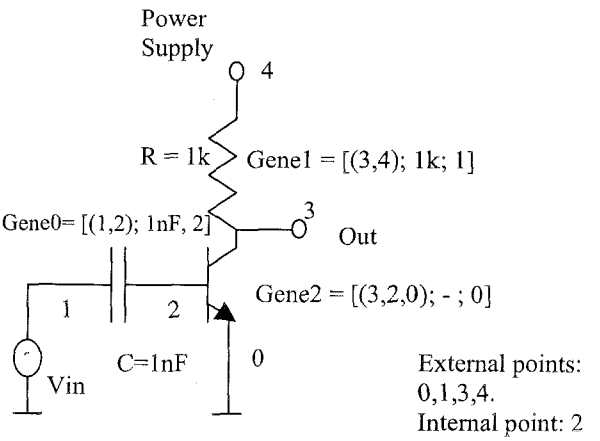

Figure 1– Analog Circuit's Representation. Gene = [Connecting points, Component value, Component nature]. The Component nature is given by:0 = transistor; 1 = resistor; 2 = capacitor

The chromosomes are made up of genes, each of which encodes a particular component. In the above figure, the chromosome will consist of three genes. The gene determines the nature, value and connecting points of the related component. 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.

One of the main advantages of this representation is the fact that it does not bias the search process to look for conventional topologies, by allowing unstructured connections among components. Additionally, we also borrow from nature the concept of each gene roughly encoding for a particular feature of the organism.

The case studies presented in this paper use NMOS and PMOS transistors as circuit building blocks.

2.2 – Evaluation

The evaluation refers to the way a measure of performance is assigned to each circuit sampled by the GA. So far, most works concerning the evolution of analog circuits have used only one goal to be attained by the evolved circuit. Genetic Algorithms requires scalar fitness information on which to work; this information is later used to drive the selection process within the GA. The next section will discuss more on this topic.

3. Proposed Method

Many times we can simplify the problem specification without losing the desired functionality, and pose the same problem to the Evolutionary Algorithm in a different way. As an example, if we need to evolve a circuit that maps a function *f*:

$$V_{out} = f(V_{in}) \qquad (1)$$

Where f is a hypothetical transfer function. Evolution can usually accomplish this task in the case of monotonic functions. We can, as an example, easily evolve a circuit that realizes a hyperbolic tangent function, described by:

$$V_{out} = Tanh(V_{in}) \qquad (2)$$

Figure 2 shows the fitness of the best circuits along the generations, where a quick convergence can be observed. Figure 3 compares the response of the evolved circuit against the target.

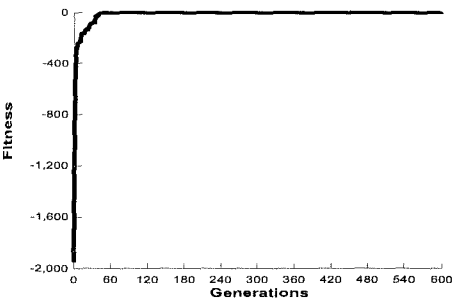

Figure 2 – Fitness along the generations for Hyperbolic Tangent Circuit Evolution (Maximum possible fitness is '0').

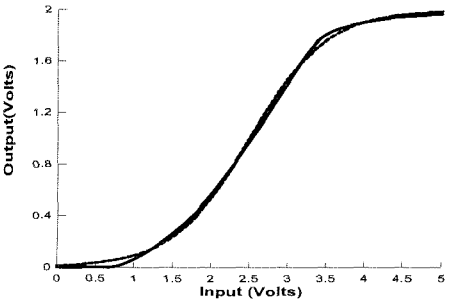

Figure 3 – Response of the evolved and of the target *Tanh* circuits.

Contrasting to the monotonic case, the task gets more difficult for more complicated DC transfer functions. For more complex functions than this, we can relax the operating point constraint, by specifying the fitness function in the following way:

$$I_{out} = k \cdot f(V_{in}) \qquad (3)$$

Where k is a proportionality constant that is "chosen" during evolution, i.e., it is another degree of freedom for evolution. The Evolutionary Algorithm will then, instead of finding circuits obeying the strict relationship depicted in equation (1), search for circuits obeying the family of relations depicted in equation (3), which are likely to be more frequent in the search space.

It is also advantageous to work with currents rather than voltages at the circuit output: the output current can span many decades, i.e., from nano-amps to mili-amps, providing more potential solutions for evolution, since we can find in the literature many circuits that perform current to voltage conversion for very low current levels [11]. It is possible, therefore, to explore very low-current circuits, and to use appropriate output

stages to convert back to the desired output voltage swing level, if necessary.

4. EXPERIMENTS

We apply this concept to evolve some computational circuits of practical interest, a Gaussian neuron, and a multiplier. We then extend the experiments to evolve a fuzzy controller using less than 10 transistors. Some of these case studies had been attempted previously without the technique described above, requiring, though, more computationally intensive experiments.

4.1. Gaussian

The evolution of this circuit has been successfully accomplished previously, however some mechanisms were employed to speed up evolution: hardware evolution and/or evolution on a multi-processor machine [12].

By using the proposed technique, the circuit of Figure 4 was achieved. Figure 5 and 6 depict, respectively, the current and the voltage response of this circuit. Figure 6 also includes the target Gaussian curve. The conversion constant k stayed around 10^5. After conversion, the voltage response exhibited an average error of 1.86% from the target. Physically, a simple resistance can realize the constant k. The evolutionary algorithm sampled only 40 individuals over 40 generations. A Sun SPARC workstation was used in the experiment.

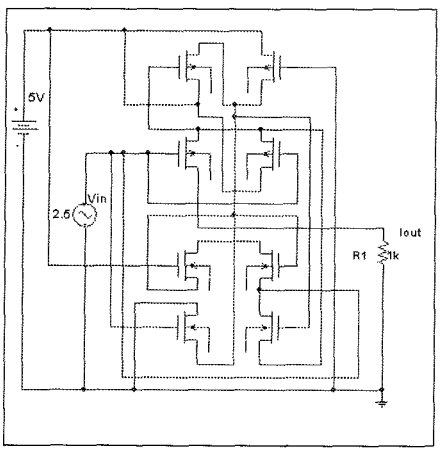

Figure 4 – Evolved Gaussian circuit. Transistor substrate connected to Vdd (PMOS) and to ground (NMOS).

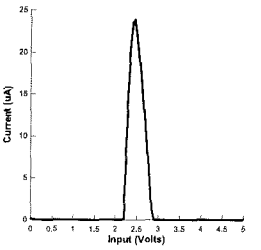

Figure 5 – Current (uA) response of the evolved Gaussian circuit.

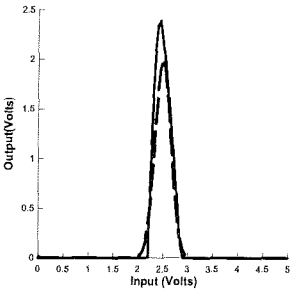

Figure 6 – Voltage response of the evolved Gaussian circuit (full line) and target (traces).

4.2 Transconductance Multiplier

Real-time multiplication of two signals is one of the most important operations in analog signal processing. The multiplier is used not only as a computational building block but also as a programming element in systems such as filters, neural networks, and as mixers and modulators in a communication system [13]. Particularly, transconductance multipliers yield the following output:

$$i_o(t) = k.\ v_1(t).\ v_2(t)$$

where $v_1(t)$ and $v_2(t)$ are input signals and k is a constant with suitable dimension. This equation fits well with our approach.

Figure 7 compares the schematic of a human-made design of a CMOS multiplier with the evolved design. Figure 8 shows the fitness value along the generations. The fitness is computed as the sum of the squared deviations between the actual output and the target, the ideal value being 0. Figure 9 compares the evolved circuit output (after being converted to voltage) to the target. This solution has been achieved after only 200 generations, sampling 50 individuals. The average percentage error to the target obtained was 2.43%, whereas the highest error was of 10.18%.

It can be observed from Figure 7 that the evolved multiplier uses 6 transistors, against 19 of the conventional circuit.

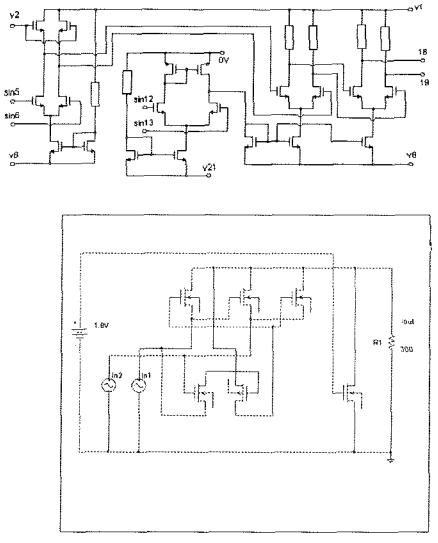

Figure 7 - Human designed multiplier (top) and evolved circuit (bottom). Substrate connections at ground for NMOS and Vdd for PMOS.

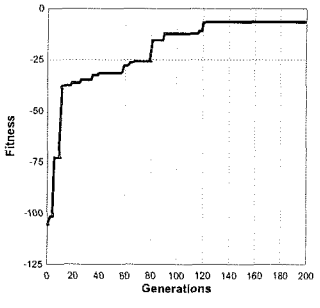

Figure 8 – Fitness along 200 generations for the multiplier.

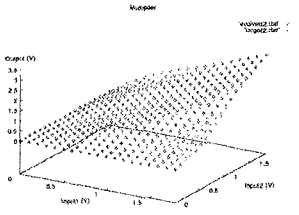

Figure 9 – Comparison between evolved and target surfaces for the multiplier.

4.3 Synthesis of Compact Fuzzy Controllers

The evolution of analog controllers is a promising path for showing the potential of evolutionary electronics applied to potential industrial applications. Particularly, this case study refers to the evolution of a current mode circuit implementing a fuzzy controller surface. Most of the traditional fuzzy systems in use however, are quite simple in nature and the computation can be expressed in terms of a simple surface. An example is the control surface of a two-input fuzzy controller. A fuzzy circuit could be synthesized to approximate this surface.

The example chosen is that of a fuzzy controller provided as a demo for the popular MATLAB software [14]. The "ball juggler" is one of the demos of the MATLAB Fuzzy Logic Toolbox. The fuzzy controller for the ball juggler has two inputs and one control output. A screen capture illustrating the membership functions is shown in Figure 10. The controller is a simple Sugeno-type with 9 rules. A screen capture of the control surface is shown in Figure 11.

A circuit approximating the control surface was evolved and is presented in Figure 12. The response, presented together with the target surface for comparison are shown in Figure 13. The average error achieved was of 1.93%, and the maximum error to the target surface was 6.7%.

The circuit is rather robust, and was tested at variations in transistor sizes, supply voltage and temperature, with the following results: decreasing the transistor sizes by a factor of 10 did not change the circuit response and the deviation from the target; average error of 1.98% and maximum error of 6.96% when decreasing the power supply voltage to 4.75V; average error of 1.94% and maximum error of 6.65% when increasing the power supply voltage to 5.25V; average error of 1.89% and maximum error of 6.3% when decreasing the temperature to 0°C; average error

of 1.98% and maximum error of 7.2% when increasing the temperature to 55°C.

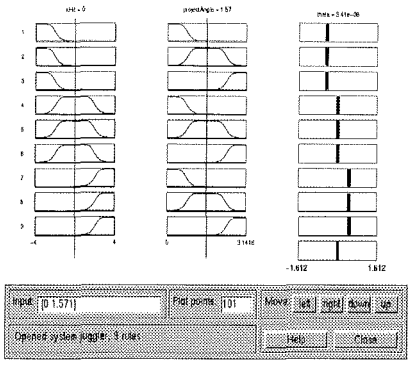

Figure 10 – Membership functions for the ball-juggler fuzzy controller.

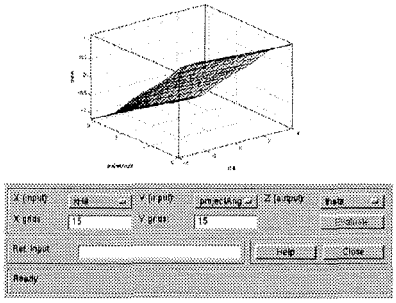

Figure 11 – Surface of the ball juggler fuzzy controller.

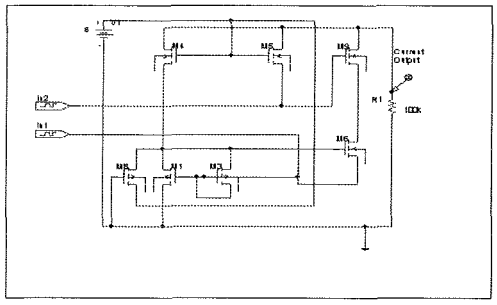

Figure 12 – Evolved circuit realizing ball juggler fuzzy controller. Transistor substrate connections at 5V for PMOS and 0V for NMOS.

Finally, a different model, (specific for a HP 0.5 MOS fabrication process) led to qualitatively the same result, with slight increase in the error. That error became small again when evolution targeted a circuit in that specific process.

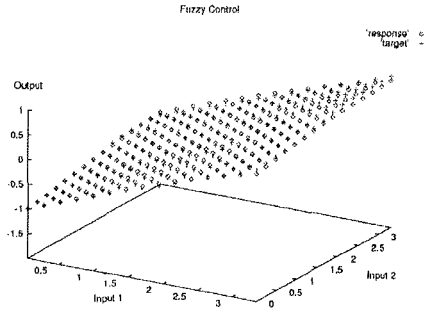

Figure 13 – Comparison between response and target for the evolved fuzzy controller.

5. Applications For Propulsion Controlled Aircrafts

One of the goals of this research is to evolve fuzzy control surfaces for aircraft propulsion control [15]. This objective of this system is to employ pilot flight path inputs and aircraft sensor feedback parameters to provide appropriate engine thrust commands for emergency flight control, reducing thus the number of accidents. The control system acts both on the longitudinal control using collective thrust and lateral-directional control, using differential thrust.

Contrasting to other systems reported in the literature [15], the objective is to have a low power consumption analog controller, for manned and unmanned aircrafts. This will require a strong interaction with experienced pilots/engineers to design the fuzzy control surface: for instance how to determine the flight path and bank angle commands given airspeed, roll rate, yaw rate, etc. In addition, the low-power analog controller could also generate other commands for the aircraft, such as flap controls when landing.

6. Conclusions

We observed in these experiments that evolving circuits that work in current mode facilitates the task of the Evolutionary Algorithm, when comparing to circuit evolution sampling a voltage at the output. We showed results for a Gaussian circuit, a multiplier circuit and a fuzzy controller.

The evolution of a compact fuzzy controller is a promising result for industrial applications of Evolvable Hardware, and the authors will tackle

aircraft control using this technique. Even though important from a practical point of view, the evolution of fuzzy control surfaces turned out to be a simple problem for evolution *when using this technique.* This result encourages the authors to address more complex case studies in the near future.

ACKNOWLEDGEMENTS

This research was performed at the Center for Integrated Space Microsystems, Jet Propulsion Laboratory, California Inst. of Technology and was sponsored by the Defense Advanced Research Projects Agency (DARPA) under the Adaptive Computing Systems Program.

References

[1] - Zebulum, R.S., Vellasco, M., Pacheco, M.A., "Evolvable Systems in Hardware Design: Taxonomy, Survey and Applications", Proceedings of First International Conference in Evolvable Systems, Tsukuba, Japan,, Lecture Notes in Computer Science, Vol. 1259, Springer-Verlag, October, pp. 344-358, 1996.

[2] - Zebulum, R.S., Pacheco, M.A., Vellasco, M.,"A Multi-Objective Optimisation Methodology Applied to the Synthesis of Low-Power Opertational Amplifiers", proceedings do XIII International Conference in Microelectronics and Packaging, Vol. 1, Ivan Jorge Chueiri and Carlos Alberto dos Reis Filho (editors), pp. 264-271, Curitiba, Brazil, August, 1998.

[3] - D. Keymeulen, A. Stoica, R. Zebulum, Raoul Tawel, Taher Daud, Anil Thakoor. Fault-Tolerant Evolvable Hardware using Field Programmable Transistor Arrays. In *IEEE Transactions on Reliability*, Special Issue on Fault-Tolerant VLSI Systems, vol. 49, No. 3, 2000 September, pp. 305-316,. IEEE Press.

[4] - A. Stoica, R. Zebulum, and D. Keymeulen, "Polymorphic Electronics", to be publishing in the Proc. of the International Conference on Evolvable Systems (ICES2001), to be held in Tsukuba, Japan, October, 2001.

[5] - A. Stoica, D. Keymeulen, and R. Zebulum, "Evolvable Hardware Solutions for Extreme Temperature Electronics", Third NASA/DoD Workshop on Evolvable Hardware, Long Beach, July, 12-14, 2001, pp.93-97, IEEE Computer Society.

[6] - Zebulum, R.S., Pacheco, M.A., Vellasco, M., "Analog Circuits Evolution in Extrinsic and Intrinsic Modes", International Conference in Evolvable systems, ICES98, Lecture Notes in Computer Science 1478, Springer-Verlag, pp. 154-165, Lausanne, September, 1998.

[7]- Zebulum, R. S., Pacheco, M. , Vellasco, M., "Artificial Evolution of Active Filters", Proceedings of First NASA/DoD Workshop on EvolvableHardware (EH'99)",pp. 66-75 IEEE Computer Society press, ISBN 0-7695-0256-3.

[8] - Zebulum, R.S., Pacheco, M.A., Vellasco, M.,"Evolutionary Design of Logic Gates", Workshop in Evolutionary Design, pp. 12-17, AID98, Lisbon, July, 1998.

[9] - Miller, J. F., Thomson, P., and Fogarty, T., "Designing Electronic Circuits Using Evolutionary Algorithms. Arithmetic Circuits: A Case Study", in Genetic Algorithms Recent Advancements and Industrial Applications, Chapter 6. Editores: D. Quagliarella, J. Periaux, C. Poloni and G. Winter, Wiley, Nov., 1997.

[10] - Zebulum, R.S, Astoica, A., Keymeulen, D., "Experiments on the Evolution of Digital to Analog Converters", published in the Proceedings of the 2001 IEEE Aerospace conference, March, 2001, Montana. ISBN: 0-7803-6600-X (published in CD).

[11] - A. Khan, A. Deval, and L. A. Akers, " A Locally Adaptive Multimode Photodetector Circuit", Analog Integrated Circuits and Signal Processing, 18 255-275 (1999).

[12] - Stoica, Adrian Stoica, Ricardo Zebulum, Didier Keymeulen, Raoul Tawel, Taher Daud, and Anil Thakoor. Reconfigurable VLSI Architectures for Evolvable Hardware: from Experimental Field Programmable Transistor Arrays to Evolution-Oriented Chips. *IEEE Transactions on VLSI*, IEEE Press , Volume 9, Number 1, ISSN 1063-8210, pp. 227-232, February 2001.

[13] – Han, G., Sanchez-Sinencio, E., "CMOS Transconductance Amplifiers: A Tutorial", IEEE Transactions on Circuits and Systems – II Analog and Digital Signal Processing, Vol. 45, No. 12, December, 1998.

[14] – http://www.mathworks.com

[15] – Kaneshige, J., Bull, J., Kudzia, E., Burcham, F., "Propulsion Control With Flight Director Guidance as an Emergency Flight Control System", American Institute of Aeronautics and Astronautics, AIAA-99-3962.