Intrinsic Evolution of Controllable Oscillators in FPTA-2

Lukáš Sekanina and Ricardo S. Zebulum

[1] Faculty of Information Technology, Brno University of Technology, Božetěchova 2, 612 66 Brno, Czech Republic
[2] Jet Propulsion Laboratory, California Institute of Technology, Pasadena, CA
sekanina@fit.vutbr.cz, rzebulum@mail2.jpl.nasa.gov

Abstract. Simple one- and two-bit controllable oscillators were intrinsically evolved using only four cells of Field Programmable Transistor Array (FPTA-2). These oscillators can produce different oscillations for different setting of control signals. Therefore, they could be used, in principle, to compose complex networks of oscillators that could exhibit rich dynamical behavior in order to perform a computation or to model a desired system.

1 Introduction

The conventional design of analog as well as digital oscillators is difficult since it requires a lot of experience. Designers must guarantee that their oscillators meet the specifications in terms of the frequency of oscillations, amplitude, phase, shape of signal, sufficient power and some other properties. Oscillators are also usually very sensitive to the environment (temperature, electromagnetic field, etc.) in which they operate. In the recent years various EA-based approaches have been proposed to design the oscillators automatically [1,3,8]. Oscillators were evolved at the opamp, transistor and gate levels. In general, the results show that evolution of oscillators with required properties is difficult.

Oscillators do play an important role not only in the area of electronic circuits. Oscillatory networks have been studied as information processors by many researchers because they can be constructed from realistic nonlinear dynamical systems and are biologically plausible (furthermore, for example, cellular neural networks or spiking neural networks have practical applications).

Networks of oscillators can be identified in neural systems or genetic regulatory networks. Recently, a synthetic network capable of producing sustained oscillations in protein concentrations was presented [2]. The "repressilator" consisted of three genes (for simplicity, called a, b, c), expressing three proteins (respectively, A, B, C). The network formed a ring: Protein A repressed transcription of gene b; B repressed c; and C repressed a. For certain biochemical parameters, this cyclic repression produced self-sustained roughly sinusoidal oscillations over the entire growth phase of the host *Escherichia coli* cells. In another work, a model for controlling a synthetic gene network of coupled oscillators was presented [11]. Unlike the repressilator, the oscillator consisted of

J.M. Moreno, J. Madrenas, and J. Cosp (Eds.): ICES 2005, LNCS 3637, pp. 98–107, 2005.

only two genes (x and y) and was of the relaxation type. Both proteins were under the control of a promoter that was activated by the protein X, and protein Y was a protease of X. Oscillations arose because Y degrades X and thus reduces its own expression level (because X activates transcription of y). Neural oscillators inspired by *olfactory cortex* models were investigated in [13]. They can be utilized as a dynamical context addressable memory [7] or to perform logic computation in which synchronized oscillations are considered as logic 1 and desynchronized oscillations as logic 0. Logic gates AND, NOR and NXOR were implemented by means of these networks [13].

Networks of oscillators can be composed of a controllable oscillator as a building block, i.e. of an oscillator whose output can be controlled using the input signals enabling or disabling oscillations. These signals are taken from the outputs of other oscillators in the network. The first step to build networks of oscillators is creating the controllable oscillators. Therefore, the objective of this paper is to explore whether controllable oscillators can be evolved intrinsically in a physical platform reconfigurable at the transistor level. We decided to utilize the transistor level because we assume that more various and richer dynamic behavior can be obtained than at the gate level. In next step of research the evolved controllable oscillators will be connected in oscillator networks. As we are not primarily interested in the frequency of oscillations, we propose a simple fitness function operating in the time domain. In this work the controllable oscillators are evolved directly in the Field Programmable Transistor Array (FPTA-2). The oscillators have one or two digital control inputs and produce various oscillations for different input stimuli.

In practice, the networks of oscillators could perform useful parallel asynchronous computation in the way similar to cellular automata, for example, in signal processing tasks. Having inspiration in the mentioned genetic regulatory networks, the evolved networks of oscillators could implement non-trivial genotype-phenotype mappings useful for embryonic electronics [5,9]. Furthermore, in addition to traditional models of genetic control networks developed by Kauffman and others [4], the system could be used to model and study natural gene regulations (see the evolution of limit cycle dynamics in electronic models in [10]).

The paper is organized as follows. Section 2 briefly introduces the area of evolutionary design of oscillators. In Section 3 FPTA chip and SABLES system are described. The proposed evolutionary design method is formulated in Section 4. While Section 5 summarizes the obtained results, Section 6 discusses them. Conclusions are given in Section 7.

2 Evolutionary Design of Electronic Oscillators

Oscillators are difficult to design manually. Hence the evolutionary approach was utilized to perform this task. Oscillators are usually evolved in the way similar to other analog circuits evolution [15]. However, the evolutionary approach does not work as well as in case of other analog circuits (e.g. filters). That is also demonstrated in Koza's list of human-competitive results that does not contain any oscillator circuits; on the other hand it contains about 20 analog circuits [6]. The construction of fitness function is very important especially in case of evolution of oscillators. The analysis of circuit behavior performed in the fitness function can be based on various principles: time domain analysis, frequency domain analysis or transfer function analysis. Corresponding fitness landscapes are usually extremely rugged; oscillations appear only in a very specific parts of the search space.

Huelsbergen et al. evolved oscillators (astable multivibrators) from primitive logic components in Xilinx XC6216 FPGA [3]. They reported results of in Silico oscillator evolution for ten target frequencies in three cell-array sizes (6x8, 8x8, and 16x16). Considering all three cell-array sizes, the system discovered relatively accurate oscillators – over 97% of their pulses correct – for five of the ten frequencies and required only a small number of GA runs. In fitness function, the output signal was compared against a binary string containing the required combinations of 0s and 1s; thus the number of missed pulses could be calculated. It was not at all understood how the evolved circuits function. For example, relative to the speed of the FPGA's gates (nanosecond transition times), the evolved oscillators are of rather low frequency.

Aggarwal has used genetic algorithm to evolve opamp-based sinusoidal oscillators [1]. His algorithm looks for a suitable passive network (consisting of a given number of resistors and capacitors) connected to a single opamp. In fitness function a symbolic analysis was used to find out the transfer function which contains specific expressions indicating oscillations. It was found that the GA rediscovered all the twelve canonic single opamp-based topologies. Some new interesting opamp-based topologies of oscillators were also discovered.

Field programmable analog array MPAA020 of Motorola was utilized to evolve opamp-based oscillators [14]. The fitness function tried to maximize the voltage difference between samples of the outputs at specified time points. The evolved circuit generated a close-to-perfect square wave of 3 Volts amplitude and frequency of 200 kHz.

Layzell and Thompson evolved oscillators in Evolvable Motherboard at the transistor level [8]. The circuit population was rich on oscillator circuits and GA was used to optimize the frequency – measured directly in the fitness calculation process.

Except Aggarwal's results (who has worked at symbolic level), the aim of the mentioned approaches was to demonstrate that oscillators can be evolved in the given target platform. The evolved oscillators were not used in any application. No other types of evolved oscillators, such as controllable oscillators or voltage-controlled oscillators have been reported in literature.

3 Evolvable Platform: FPTA-2 and SABLES

A complete stand-alone board-level evolvable system (SABLES) is built by integrating the FPTA and a DSP implementing the Evolutionary design algorithm

[12]. The system is connected to the PC only for the purpose of receiving specifications and communicating back the result of evolution for analysis. The system fits in a box 8“ x 8“ x 3“. Communication between DSP and FPTA is very fast with a 32-bit bus operating at 7.5MHz. The evaluation time depends on the tests performed on the circuit. Many of the tests attempted here require less than two milliseconds per individual, and runs of populations of 100 individuals from 100 to 200 generations require only 20 seconds.

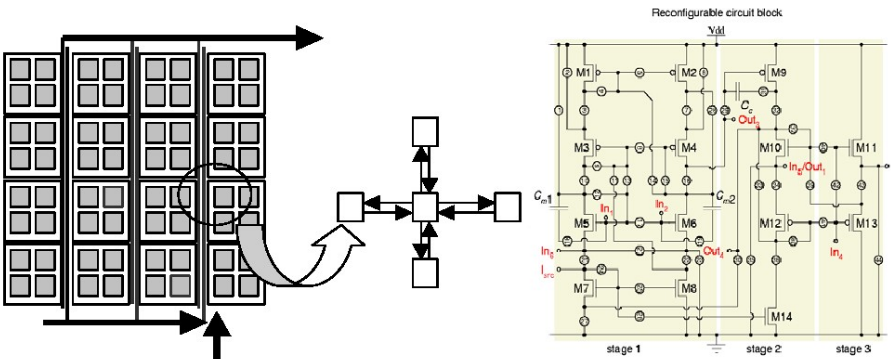

Fig. 1. FPTA-2 architecture (left) and schematic of cell transistor array (right). The cell contains additional capacitors and programmable resistors (not shown).

The FPTA is an evolution-oriented reconfigurable architecture (EORA). It has a configurable granularity at the transistor level. It can map analog, digital and mixed signal circuits. The architecture of the FPTA consists of an 8x8 array of re-configurable cells. Each cell has a transistor array as well as a set of programmable resources, including programmable resistors and static capacitors. Figure 1 provides a broad view of the chip architecture together with a detailed view of the reconfigurable transistor array cell. The reconfigurable circuitry consists of 14 transistors connected through 44 switches. A total of 5000 bits is used to program the whole chip. The pattern of interconnection between cells is similar to the one used in commercial FPGAs: each cell interconnects with its north, south, east and west neighbors. The reader can refer to [12] for more information on the FPTA-2.

4 Design Method

The controllable oscillators will be designed using a standard genetic algorithm operating directly with configurations of FPTA-2 as chromosomes. Only a few cells of the FPTA will be utilized for the experiments. Figure 2 shows the cells and the connection of input and output signals. No external components (such as RC circuits) were considered for these experiments. The frequency of oscillations depends only on the configuration and internal characteristics (such as delay of transistors) of FPTA-2.

The genetic algorithm running in a DSP uses the roulette-wheel selection, crossover and mutation. Candidate solutions are evaluated directly in FPTA-2. In this process, all possible combinations of logic values over the input control signals (a and b) are applied at the circuit inputs and oscillations are detected at the output y. The genetic algorithm must promote the chromosomes that cause oscillations if they are required and keep the output invariable otherwise. In particular 240 values are sampled, digitized and utilized during the evaluation of a candidate circuit. Because of simplicity we decided to evaluate candidate circuits in the time domain. Oscillators controlled using a single input signal $a[i]$ are designed using the fitness function whose basic structure is given in the following pseudo-code:

Algorithm 1:

```
i = 0; fitness = 0;
while (i < samples)
{
  // oscillations
  ones = 0; zeroes = 0; penalty = 0;
  while (i < samples and a[i] is High)
  {
      if (y[i] < LL) zeroes = zeroes + 1;
      else if (y[i] > HL) ones = ones + 1;
      else penalty = penalty + 1;
  }
  fitness = fitness + k1 * abs(ones – zeroes) + kp * penalty;

  // no oscillations
  ones = 0; zeroes = 0; penalty = 0;
  while (i < samples and a[i] is Low)
  {
      if (y[i] < LL) zeroes = zeroes + 1;
      else if (y[i] > HL) ones = ones + 1;
      else penalty = penalty + 1;
  }
  fitness = fitness + k2 * (zeroes + ones – abs(zeroes + ones)) + kp * penalty;
}
```

If $a[i]$ is at log. 1 (High), the circuit should oscillate; otherwise, the circuit should not. Here, $i = 1 \ldots 240$ samples are evaluated at the circuit output $y[i]$. The *zeroes* counter indicates the number of output values that are considered as lower than a given threshold value LL ($LL = 0.45MV$ where MV determines the maximum output voltage 1.8V). The *ones* counter indicates the number of output values that are considered as higher than a threshold value HL ($HL = 0.55MV$). Note that, here, the fitness should be minimized. The situation in which the circuit should oscillate (i.e. the number of *zeroes* and *ones* is similar but non-zero) is evaluated in the first nested while loop. The second nested loop deals with the situation in which the output should not oscillate. *Penalty* counter is used to avoid staying in the middle of MV range. The values of constants k_1, k_2 and k_p are determined experimentally, and $k_p \gg k_1 = k_2$. A very similar fitness function has been utilized to design oscillators controlled using two bits.

5 Experimental Results

If a single cell of FPTA is configured as an inverter and its output is connected to its input then oscillations are always observable. We utilized this property in our approach. Figure 2 shows the experimental setup used to evolve controllable oscillators using four and five FPTA-2 cells. The solid lines in Fig. 2 denote external physical connections (wires) used to connect the cells. These connections were utilized to promote a specific design pattern which is typical for the conventional oscillators composed of three inverters. In addition to these connections, the evolution could interconnect the cells using the internal switches of the FPTA-2. Behavior of a cell is defined using 77 configuration bits. However, three words (48 bits) are not evolved for the cells that belong to the cells that are connected in a ring; indeed, they are taken from the configuration bitstream of a conventional inverter and used during all experiments. This strategy is applied in order to obtain some oscillations in a shorter time. We know that conventional oscillators can be designed in this way. In fact we were not able to evolve any oscillators without this setup. Parameters of GA are as follows: the population size = 100, the crossover probability = 70%, and the mutation probability = 10%. Depending on experiment 300-1000 generations were produced.

5.1 One-Bit Controllable Oscillators

Various one-bit controllable oscillators were evolved using the setup from Fig 2A. Figure 3 shows typical oscillations we obtained (the frequency of oscillations is 90.9kHz). Similar other oscillators we evolved that operate at the following frequencies: 41.6kHz, 22.7kHz, 83.3kHz, and 38.5 kHz. The shape of the output signal is usually very close to the sine wave; however, with some distortions. We also attempted to change the frequency of oscillations by means of increasing

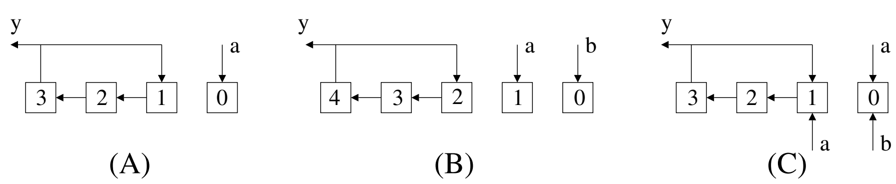

Fig. 2. Cells used and their connection. a and b are control signals; y is the output signal.

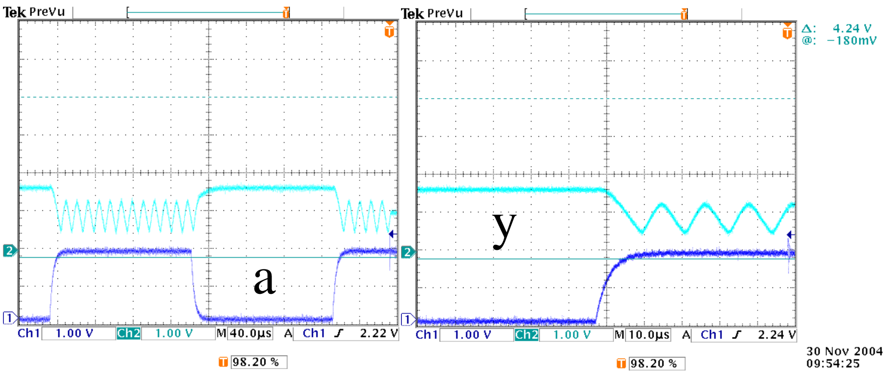

Fig. 3. Evolved 1-bit controllable oscillator (f = 90.9kHz)

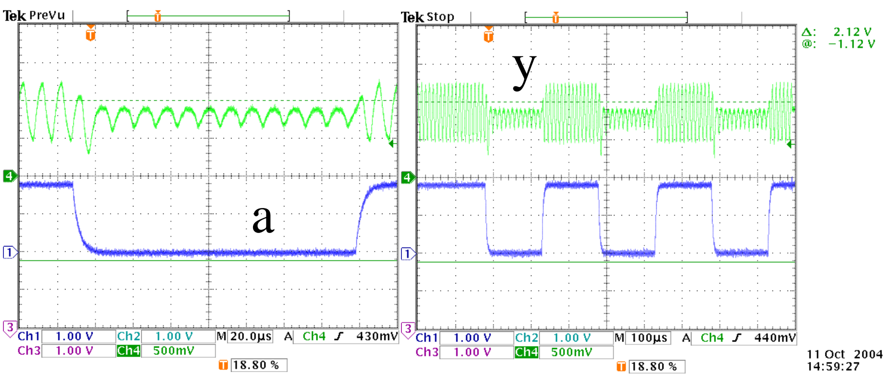

Fig. 4. Evolved 1-bit controllable oscillator (f = 83.3 kHz for both waves)

voltage at the control input. However, we were not able to evolve such a kind of controllable oscillators. In another setup, a circuit producing two types of oscillations was evolved (Figure 4). In order to obtain this result, we required in the fitness functions that $ones = 2 * zeroes$ when the control signal is at logic 0.

5.2 Two-Bit Controllable Oscillators

The two-bit controllable oscillators utilize two input signals, a and b, to control the oscillations. As shown in Fig. 2B, they consist of five cells. The oscillations, controlled through cells 0 and 1, should emerge in cells 2, 3 and 4. The proposed fitness function has been modified in order to consider all four combinations over the inputs a and b. For instance, we required to have oscillations only when $a = b = 1$. Figure 5 shows a typical behavior we obtained. Let us define the following logic interpretation of that behavior. Let oscillations mean logic 1 and let no oscillations mean logic 0. Then the evolved circuit whose behavior

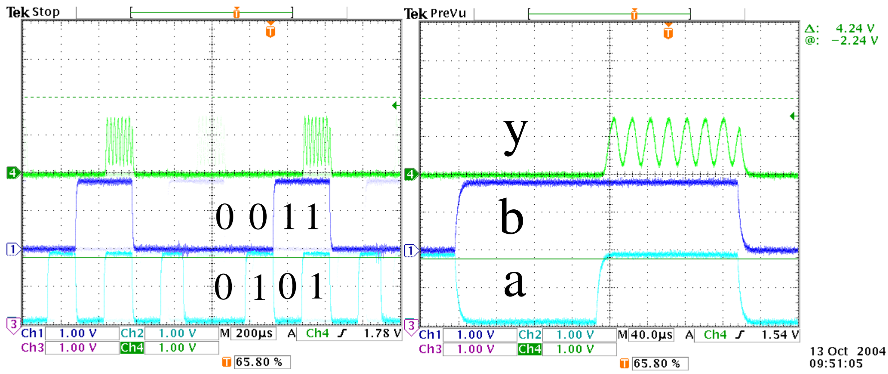

Fig. 5. Evolved 2-bit controllable oscillator operating as AND (f = 50 kHz)

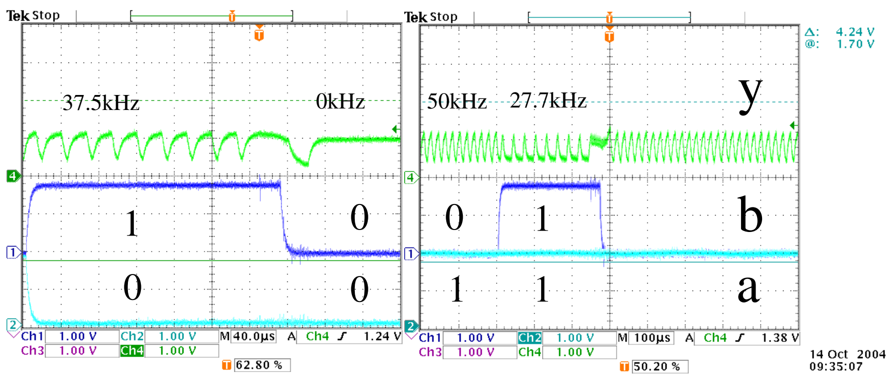

Fig. 6. Evolved 2-bit controllable oscillator generating four different behaviors

is depicted in Fig. 5 can be understood as logic function AND. Considering this interpretation we were able to evolved various other logic functions, and surprisingly, we also evolved exclusive-or (XOR) function.

In another experiment we evolved a circuit that exhibits four different behaviors for four different combinations of the control inputs. It generates a signal of frequency 27.7kHz for $a = 1$ and $b = 1$, 50kHz for $a = 1$ and $b = 0$, 35.7kHz for $a = 0$ and $b = 1$ and no oscillations for $a = 0$ and $b = 0$ (see Fig. 6).

6 Discussion

The presented work has addressed the question whether the evolutionary approach is able to discover controllable oscillators at the transistor level. The answer is positive, i.e. the transistors available for the evolutionary design can be composed together by means of an automated evolutionary process in order to establish one- and two-bit controllable oscillators. The search was not performed completely from scratch. We promoted some "ring"-based structures and partially preconfigured the cells in the ring. No oscillations have appeared in case of a complete evolution from scratch. On the other hand no information showing a way how to stop/enable oscillations was provided for the evolution. Therefore, the evolutionary approach really discovered how to create controllable oscillators. It is interesting that we were able to repeat almost all experiments reported in Section 5.2 also using only four cells of FPTA-2. The setup is shown in Fig. 2C.

The success of evolution also depends on values of coefficient k_1, k_2 and k_p. If the penalty for oscillations is too high, no oscillating candidate circuits are visible. If the penalty for no oscillations is too high, the population contains many oscillators; however, it is impossible to control the oscillations via the input control signals. Looking for suitable values of these coefficients is a very time consuming experimental work requiring tens of runs of the GA. Once the values of coefficients are fixed, a 1-bit controllable oscillator is usually found in approximately 30% of runs and 2-bit controllable oscillator in 10% of runs.

The main disadvantage of the proposed fitness function is that it is difficult to specify the frequency of oscillations and shape of the wave. The time domain analysis allowed us to specify only the required number of values higher or lower than a given threshold value. More sophisticated search for a given frequency of oscillations (e.g. a multiobjective method) would probably require the analysis in the frequency domain which, however, requires more computational effort. On the other hand the oscillators in network have not to work at a predefined frequency. They can operate at different frequencies that are suitable for a given platform.

7 Conclusions

Simple one- and two-bit controllable oscillators were intrinsically evolved using only four cells at the transistor level directly in FPTA-2. We can control the oscillations using logic signals which in principle allows us to build networks of oscillators. The question for future research is whether the output oscillations are able to control other oscillators in order to connect them into a complex network.

Acknowledgments

The research described in this paper was performed at the Jet Propulsion Laboratory, California Institute of Technology and was sponsored by the National Aeronautics and Space Administration (NASA). Lukas Sekanina was supported by the Fulbright scholarship and from the research project of the Grant Agency of the Czech Republic under No. 102/04/0737 *Modern methods of digital system synthesis.*

References

1. Aggarwal, V.: Evolving Sinusoidal Oscillators Using Genetic Algorithms. In Proc. of the 2003 NASA/DoD Conference on Evolvable Hardware, Chicago, USA, IEEE Computer Society, 2003, p. 67–76
2. Elowitz, M. B., Leibler, S.: A Synthetic Oscillatory Network of Transcriptional Regulators. Nature (London) Vol. 403, 2000, p. 335–338
3. Huelsbergen, L., Rietman, E., Slous, R.: Evolving Oscillators in Silico. IEEE Trans. on Evolutionary Computation. Vol. 1, No. 3, 1999, p. 197–204
4. Kauffman, S. A.: The Origins of Order: Self Organization and Selection in Evolution. Oxford University Press, 1993
5. Koopman, A.: Hardware-Friendly Genetic Regulatory Networks in POEtic Tissues. MSc Thesis. Utrecht University, 2004, 75 p.
6. Koza, J.: 36 Human-Competitive Results Produced by Genetic Programming. http://www.genetic-programming.com/humancompetitive.html
7. Kozma, R., Freeman, W.: Chaotic Resonance-Methods and Applications for Robust Classification of Noisy and Variable Patterns. Neural Networks. Vol. 4, No. 1., 2001, p. 103–121
8. Layzell, P., Thompson, A.: Understanding Inherent Qualities of Evolved Circuits: Evolutionary History as a Predictor of Fault Tolerance. In Proc. of the 3rd Evolvable Systems: From Biology to Hardware Conference, LNCS 1801, Springer-Verlag, 2000, p. 133–142
9. Mange, D. et al.: Towards Robust Integrated Circuits: The Embryonics Approach. Proc. of IEEE. Vol. 88, No. 4, 2000, p. 516–541
10. Mason, J., Linsay, P. S., Glass, L.: Evolving Complex Dynamics in Electronic Models of Genetic Networks. Chaos, Vol. 14, No. 3, 2004, p. 707–715
11. McMillen, D. et al.: Synchronizing Genetic Relaxation Oscillators by Intercell Signaling. Proc. of the National Academy of Sciences of the USA, Vol. 99, No. 2, 2002, p. 679–684
12. Stoica, A. et al.: Evolving Circuits in Seconds: Experiments with a Stand-Alone Board Level Evolvable System. In Proc. of the 2002 NASA/DoD Conference on Evolvable Hardware, Alexandria Virginia, USA, IEEE Computer Society, 2002, p. 67–74
13. Xu, D., Principe, J. C., Harris, J. G.: Logic Computation Using Coupled Neural Oscillators. In Proc. of IEEE Intl. Symposium on Circuits and Systems, 2004, p. 788–791
14. Zebulum, R. S., Pacheco, M., Vellasco, M.: Analog Circuits Evolution in Extrinsic and Intrinsic Mode. In Proc. of the 2nd Evolvable Systems: From Biology to Hardware Conference, LNCS 1478, Springer-Verlag, 1998, p. 154–165
15. Zebulum, R., Pacheco, M., Vellasco, M.: Evolutionary Electronics – Automatic Design of Electronic Circuits and Systems by Genetic Algorithms. CRC Press, Boca Raton 2002