Evolvable Hardware

Marc Ebner

Systems whose architecture evolves over time are called Evolvable Systems [1]. The research area of Evolvable Hardware consists mainly of the evolution of analog and digital circuits [5]. Each individual of the evolving population represents a particular circuit. The individuals are either evaluated in simulation (*extrinsic evolution*) or directly in hardware (*intrinsic evolution*). The time needed to manually build the circuits is prohibitive. Therefore, dynamically reconfigurable chips such as field programmable gate arrays (FPGAs) are used to evaluate the individuals. Usually, the individuals are evaluated one after the other on a single reconfigurable chip. The topology of the circuit is determined by the genetic material of the individual. We will first address evolution of analog circuits and then turn to evolution of digital circuits.

Koza et al. [3] performed a large number of experiments evolving analog circuits in simulation. To evaluate the circuits, a modified version of the SPICE simulator was used. The fitness function defined the desired function of the circuit. The SPICE simulator is able to accurately simulate circuits containing resistors, capacitors, inductors, diodes, transistors, voltage and current sources. A cellular encoding was used to represent the circuits. Decoding of the individual starts with a small circuit which is called an embryo. This circuit initially consists of at least one modifiable wire or component. The entire circuit is grown from the small embryo similar to the development of a natural organism from a single cell. The organism develops by cell division. Each cell contains the same DNA as all the other cells of the organism. Similarly, in a cellular encoding a program sequence is stored in each individual. This sequence is used as the individual's DNA. It defines what happens to the modifiable wires. The modifiable wires can divide like the cells of an organism. Through this division process, components can be arranged in parallel or in series. Finally, for each modifiable wire, a resistor, a capacitor or some other component is chosen depending on the program sequence stored inside the individual. The sizing of the components is also defined by the program sequence. Koza et al. [3] evolved a large number of filters such as lowpass, highpass, bandpass, Butterworth, Chebychev or elliptic filters, amplifier circuits, computational circuits, or a temperature-sensing circuit. Some of the results are comparable to existing patented circuits.

Stoica et al. [8] developed a reconfigurable hardware architecture which consists of a transistor array, called programmable transistor array. A genetic algorithm was used to specify the connections between the transistors. Initial experiments include circuit synthesis with a Gaussian input-output characteristic.

A somewhat different approach is used for the evolution of digital circuits. In this case, the basic logic elements are often arranged in a matrix. The basic elements include logic gates such as `AND`, `XOR`, or `MUX`. The genotype of the individual specifies how these elements are distributed over the matrix and how they are connected. Again, the circuit can be evaluated either in software or hardware. Miller et al. [5] evolved 1-bit, 2-bit, 3-bit adders as well as 2-bit and 3-bit multipliers in software. The evolved circuits were 20% more efficient in number of used gates compared to existing circuits. Thompson evolved circuits to distinguish between a 1kHz and a 10kHz square signal [9]. He used a reconfigurable VLSI chip (a field programmable gate array) to evaluate the individuals. Evolution was not allowed to use an external clock to solve this problem. The circuit had to be assembled as a recurrent network solely from the available logic elements. Even though only digital components were used, the circuit was basically an analog circuit. It only worked on the field programmable gate array which was used to evolve the circuit because it used physical properties of the reconfigurable device.

When FPGAs are used to evaluate the individuals, a host computer is used to run the evolutionary algorithm. In this case, all of the genetic material is stored on the host computer and downloaded to the FPGA when the individual is evaluated. Some researchers have developed customized chips for evolvable hardware [2, 10]. These chips contain memory for the genetic material of the individuals, some memory which is used to store training data, a reconfigurable section which is used to evaluate the individuals, and a section which implements the evolutionary algorithm. Therefore, these chips integrate all aspects of the evolutionary algorithm.

An approach called Embryonics (short for embryonic electronics) tries to introduce self repairing capabilities into electronic circuits [4]. This approach is different from the one described above by Koza et al. The embryonic architecture is based on reconfigurable hardware and also draws its inspiration from nature. It is hierarchically organized. Like nature, it consists of molecules, cells and organisms. A complex circuit is created by a mechanism which is similar to what happens during cell division. The circuit starts with a single cell which contains the blueprint for the complete circuit. The cell then divides, i.e. it occupies a reconfigurable area of the chip which is free to use. Each cell has access to the same blueprint of the circuit. The cells specialize and then perform a particular function depending on the genetic blueprint and its position on the circuit. If a particular cell happens to be damaged, other cells can take over the function of the damaged cell. How such an embryonic architecture may be used was demonstrated on simple circuits such as an electronic watch, a random number generator or an implementation of a Turing machine.

What is interesting about the field of evolvable hardware is that often an encoding is used, which is similar to the development of an organism from a single cell. The desired topology of the circuit is not known initially. Both the topology of the circuit and the sizing of the components must be found by evolution. Simple parameter optimization cannot be used. Instead, one must look for an encoding which is evolvable and spans the entire space of circuits. In using these innovative encodings it is now possible to automatically search the space of electronic circuits.

Apart from the evolution of analog and digital circuits, evolutionary robotics also belongs to the area of evolvable hardware. In evolutionary robotics, researchers evolve control algorithms for mobile robots [6]. The underlying hardware and the structure of the control algorithm, e.g. a neural network, is often fixed. However, some also try to evolve the morphology and the robot's sensors [7]. Evolutionary robotics makes it possible to automate programming of mobile robots. Instead of manually programming and debugging the control algorithms, the robot is able to adapt to any given environment or task. Experiments are usually not carried out in simulation because control algorithms which are evolved in simulation may not work when transfered to a real robot. The miniature mobile robot Khepera is used in many evolutionary robotics experiments. Even though a 100% accurate simulation is not possible, carrying out experiments in simulation does have the advantage that experiments may be completed in a shorter amount of time.

Kontakt

Dr. Marc Ebner
Universität Würzburg
Lehrstuhl für Informatik II
Am Hubland, 97074 Würzburg
e-mail: ebner@informatik.uni-wuerzburg.de
http://www2.informatik.uni-wuerzburg.de/staff/ebner/welcome.html

Literatur

[1] T. Higuchi, T. Niwa, T. Tanaka, H. Iba, H. de Garis, and T. Furuya. Evolving hardware with genetic learning: A first step towards building a Darwin machine. In J.-A. Meyer, H. L. Roitblat, and S. W. Wilson, eds., *From Animals to Animats II: Proc. of the 2nd Int. Conf. on Simulation of Adaptive Behavior*, pp. 417–424, Cambridge, MA, 1993. The MIT Press.

[2] I. Kajitani, T. Hoshino, D. Nishikawa, H. Yokoi, S. Nakaya, T. Yamauchi, T. Inuo, N. Kajihara, M. Iwata, D. Keymeulen, and T. Higuchi. A gate-level EHW chip: Impelementing GA operations and reconfigurable hardware on a single LSI. In M. Sipper, D. Mange, and A. Pérez-Uribe, editors, *Proc. of the 2nd Int. Conf. on Evolvable Systems: From Biology to Hardware, Lausanne, Switzerland*, pp. 1–12, Berlin, 1998. Springer-Verlag.

[3] J. R. Koza, F. H. Bennett III, D. Andre, and M. A. Keane. *Genetic Programming III. Darwinian Invention and Problem Solving*. Morgan Kaufmann Publishers, 1999.

[4] D. Mange, M. Sipper, A. Stauffer, and G. Tempesti. Towards robust integrated circuits: The embryonics approach. *Proc. of the IEEE*, 88(4):516–541, 2000.

[5] J. F. Miller, D. Job, and V. K. Vassilev. Principles in the evolutionary design of digital circuits – part I. *Genetic Programming and Evolvable Machines*, 1(1-2):7–35, 2000.

[6] S. Nolfi and D. Floreano. *Evolutionary Robotics: The Biology, Intelligence and Technology of Self-Organizing Machines*. The MIT Press, Cambridge, MA, 2000.

[7] J.B. Pollack, H.Lipson, G.Hornby, and P.Funes. Three generations of automatically designed robots. *Artificial Life*, 7(3):215–223, 2001.

[8] A. Stoica, C. Salazar-Lazaro, D. Keymeulen, and K. Hayworth. Evolution of CMOS circuits in simulations and directly in hardware on a programmable chip. In W. Banzhaf, J. Daida, A. E. Eiben, M. H. Garzon, V. Honavar, M. Jakiela, and R. E. Smith, eds., *Proc. of the Genetic and Evolutionary Computation Conference, Orlando, FL*, pp. 1198–1203, San Francisco, CA, 1999. Morgan Kaufmann Publishers.

[9] A. Thompson. An evolved circuit, intrinsic in silicon, entwined with physics. In T. Higuchi, M. Iwata, and W. Liu, eds., *Proc. of the 1st Int. Conf. on Evolvable Systems: From Biology to Hardware, Tsukuba, Japan*, pp. 390–405, Berlin, 1997. Springer-Verlag.

[10] A. M. Tyrrell, E. Sanchez, D. Floreano, G. Tempesti, D. Mange, J.-M. Moreno, J. Rosenberg, and A. E. P. Villa. POEtic tissue: An integrated architecture for bio-inspired hardware. In A. M. Tyrrell, P. C. Haddow, and J. Torresen, eds.,

Proc. of the 5th Int. Conf. on Evolvable Systems: From Biology to Hardware, Trondheim, Norway, pp. 129–140, Berlin, 2003. Springer-Verlag.