Book Reviews

Proceedings of the First NASA/DoD Workshop on Evolvable Hardware—A. Stoica, D. Keymeulen, and J. Lohn, Eds. (Los Alamitos, CA: IEEE Computer Society, 1999, 267 pp.) *Reviewed by Hugo de Garis.*

I. Introduction

America's first workshop on evolvable hardware, EH99, took place 19–21 July 1999, in Pasadena, CA, under the auspices of NASA, the U.S. Department of Defense (DoD), and Caltech's Jet Propulsion Laboratory (JPL). The long-term aim of NASA and JPL is to build spacecraft that can detect their own errors and repair themselves, even on missions lasting 100 years or more. Evolvable hardware may contribute towards this goal.

The fact that this workshop was held in the United States was a source of some satisfaction. For several years, I have been nagging that America was falling behind Europe and Japan in this critical technology. This workshop proved that America has closed the gap and that the field itself is well and truly launched.

One of the most significant auguries of the workshop was the presence of Jose Muñoz, the founder of DARPA's (Defense Advanced Research Projects Agency) Adaptive Computing Systems Program. His presence for the full three days of the workshop suggests that American E-Hard (evolvable hardware) will be supported for the next few years. This is very encouraging for the growth of the field internationally.

The first E-Hard workshop was held in Lausanne, Switzerland, in 1995. The first E-Hard conference was held in Tsukuba, Japan, under the title of ICES96 (International Conference on Evolutionary Systems), and the second such ICES was held in Lausanne in 1998. The third will be held in Edinburgh, U.K., in 2000 and the fourth in Tsukuba in 2001. The fact that the ICES conference will be held yearly instead of biannually reflects the field's growth. The sites of these conferences show clearly that until very recently, E-Hard has been largely a Euro–Japanese phenomenon. The American organizers of EH99, however, say they will organize an annual workshop. There is talk that ICES02 may be held in Washington, DC. After that, I expect the United States will dominate the field as it does with so many other sciences.

Of the 36 papers published in the *Proceedings*, 18 were from Americans, 11 from Europeans (of which seven were from the United Kingdom), but only three from Japanese (of which two were from Westerners working at Japan's ATR lab in Kyoto). Actually, the number of Japanese papers was surprisingly low given the historical contribution that the Japanese have made to this field. This may be due, however, to a "regional effect," where the proportion of "X"-ians is high if the conference is held in region X. Alternatively, it could mean that the E-Hard field is slow getting off the ground in Japan (where I have lived in for the past eight years). One way to check this was to look at Adrian Thompson's web site (http://www.cogs.susx.ac.uk/users/adrianth/EHW_groups.html) which contains a list of E-Hard centers around the world and observe what proportion of them were Japanese. This showed that of the 34 centers listed (as of October 1999), 20 were in Europe (11 from the United Kingdom, four from Germany), 11 were in the United States and Canada, and only two were in Japan. One of those is my own group, but as I am leaving Japan in January 2000, that leaves only Higuchi's group. Maybe I should be nagging Japan from now on not to miss the boat in this critical technology. (For additional background information on evolvable hardware, see the September 1999 issue of IEEE Transactions on Evolutionary Computation.)

II. The Workshop

The workshop itself was attended by about 130 people who listened to nine invited talks and 35 refereed talks for three days in a single session framework. The field is not yet large enough for parallel sessions. The 35 refereed talks were divided into eight topic headings, whose meanings are self-explanatory:

1) evolution on field programmable gate arrays (FPGA's) (i.e., intrinsic E-Hard);
2) evolution of digital functions;
3) evolution of analog and mixed-signal circuits;
4) evolution of cellular automata and brain-inspired architectures;
5) reconfiguration architectures and dynamic reconfiguration;
6) advanced reconfigurable devices;
7) applications to design and adaptation of space subsystems;
8) evolutionary algorithm applications.

III. Highlights of the Workshop

The selection below of the most interesting papers reflects only partially my personal bias and interests. I discussed the selection with colleagues during the workshop and after so that the work and ideas mentioned below reflect the views of several people.

Nick Macias' Self-Configuring Circuits

It is not often that I go to a workshop or conference and get excited by something I hear. Most papers are just minor extensions to principles already known. With Nick Macias's paper on self-configuring electronic cells, however, I felt I was being introduced to the next paradigm after FPGA's (FGPA's = programmable logic). Many colleagues shared my enthusiasm with Macias' paper. His ideas are probably still a decade ahead of what electronic technology can deliver, since he talks in terms of millions/billions of his self-configuring cells, a number that is just not implementable right now. His prototype chip contains only 3*4 Macias cells.

What is so significant in Macias's ideas? With traditional reconfigurable electronics, the reconfiguring instructions are external to the circuitry that is being reconfigured. With Macias's cells, each cell can be a configuring cell at one moment and be configured by a neighboring cell at the next moment. In other words, cells can be both configuree and configurer. This implies that the configuring circuitry can be distributed throughout the whole circuit.

Imagine a two-dimensional (2-D) grid of identical square cells, each with four immediate neighbors. Each side of the square cell has two input lines and two output lines from and to its neighboring square cell. For a given side, one of the two input lines contains a data bit Din, the other line contains a configuration instruction bit Cin, and similarly for the two output bits Dout and Cout. At any

Manuscript received October 4, 1999.

The reviewer is with Starlab, Brussels, Belgium (e-mail: degaris@starlab.net).

Publisher Item Identifier S 1089-778X(99)09860-4.

1089–778X/99$10.00 © 1999 IEEE

given moment, a cell is either in one of two modes: data mode or configuration mode. If all four of the Cin's are zero, then the cell is in data mode. The four Din bits are used as an address in a 16*8 bit truth table. The 8 bits are output as data to the four Cout's and the four Dout's. If one of the Cin's is a one, then the cell switches to configuration mode and (quoting Macias) "the D inputs are serially shifted into the cell's internal truth table. This allows one cell to write another cell's truth table, which subsequently affects that cell's behavior when it returns to D mode. As the new truth table is shifted into the cell, the cell's prior truth table is shifted out on its D outputs, and is available for reading. When a cell is in C mode, only the D inputs and outputs on sides where Cin = 1 are relevant." Hence "any cell can control the mode of any neighboring cell. By placing a neigboring cell in C mode and reading and writing that neighbor's D lines, a cell can read and write the truth table of any neighboring cell, and thereby configure it to subsequently perform any combinatorial function desired (after returning the neighbor to D mode). Since the neighbor's new combinatorial function can produce any desired C and D outputs, that neighbor can be configured to itself configure any of its neighboring cells."

Using these capabilities, Macias has been able to create a circuit that

a) replicates a cell,
b) copies a remote cell,
c) acts as a cell library,
d) builds a wire,
e) builds an expanding 12-bit counter,
f) fills a space,
g) autonomously self-replicates,
h) acts as a guard wall (internal cells cannot be reconfigured).

These rather elementary circuits can then be used as components to construct more complex circuits, such as those capable of performing E-Hard.

Macias's cell matrix allows distributed reconfiguration control, which can be used to study not only parallel execution of algorithms in hardware, but parallel configuration of hardware. The matrix can implement circuits that create new circuits, which themselves create and modify other circuits.

Such cells in their millions/billions would be suited for implementation using nanotechnology. Unfortunately, state-of-the-art electronics is not yet up to meeting Macias' dream, so he may have to wait a decade or more before he can build "self-repairing systems," that is, systems for evolving complex circuits, multimodal systems that switch between multiple circuit configurations, or learning systems that analyze an algorithm's execution and synthesize hardware to capture its functionality.

I was excited by Macias' ideas because for years I have been dreaming of the possibility of doing electronic evolutionary embryology, i.e., having a large number of electronic cells behaving like embryonic cells, migrating, reproducing, differentiating, influencing their neighbors, etc. Macias' cells may be the electronic equivalent of living embryonic cells. I hope to collaborate with him in the future on electronic evolutionary embryology.

Tetsuya Higuchi's Industrial E-Hard Chips and Applications

Higuchi is easily the most prominent E-Hard researcher in Japan, with one of, if not the, largest E-Hard research groups in the world, working at the Electro Technical Lab (ETL) in Tsukuba (60 km north of Tokyo). Oddly, his group is the only Japanese group appearing in Thompson's E-Hard list. Perhaps some Japanese companies are working in the field, but if so they are not advertising themselves. Higuchi has obtained considerable funding from Japan's MITI (Ministry of International Trade and Industry) to develop ASIC (i.e., custom) evolvable chips with very concrete applications in mind. His approach is much more engineering than science based, concentrating on such applications as controlling the motions of fingers on an artificial hand using electrical signals from arm muscles. The E-Hard chips in the controller adapt to the person rather than the other way round, hence the patient-machine learning time is much quicker (e.g., 5 min instead of a month). The artificial E-Hard based hand should be commercialized in 1999. Higuchi's team was the first to produce a complete E-Hard system on a chip (chromosomes, evolving circuit, fitness measurement, genetic algorithm logic) in 1997–1998, a concept I was calling "1-chip E-Hard" in 1996. Amongst his other applications are analog E-Hard for cellular phones, data compression for digital printing (to be commercialized in 2000), reconfigurable DSP (digital signal processor) chips using neural net techniques for adaptive signal equalization of mobile phones, a device to aid ALS (Stephen Hawking's disease) patients to move a cursor on a computer screen via facial muscles using a methodology similar to their artificial hand work, an evolvable clock timing architecture for high speed LSI, etc. If the field of E-Hard is to find its "killer app" in the next few years, it is likely that it will be Higuchi who does it.

Don Levi's Democratization of Robust Evolvable Chips

Don Levi's work will probably be revolutionary for E-Hard and have a profound practical impact on the field. Levi is an E-Hard researcher inside the programmable chip manufacturing company Xilinx, the maker of the famous XC6200 family that enabled the E-Hard field to get off the ground. Levi has made two major advances in the field. He has overcome the brittleness of gate-level intrinsic E-Hard and has made other Xilinx chips, besides the XC6200 family, evolvable, namely the 4000 family, and the new Virtex (million programmable gate) family. Intrinsic gate-level E-Hard is brittle. When Thompson moved the 10*10 cell array he was using on his XC6216 Xilinx chip, he found that the evolved behavior did not transfer to the new site. Also, when he changed chip manufacturers or raised the temperature or voltage, the original evolved behavior did not reoccur. Levi was able to use his Xilinx insider knowledge to make the evolution of his Xilinx chips robust, employing what he calls "synchronous evolutionary techniques." The lack of robustness comes from contention of electronic signals, due to the possibility that each line may have more than one driver. The trick he employed to ensure robustness was to get his software to enable only one driver per wire, i.e., he avoided contention by construction. By discarding asynchronous logic, he was able to ensure reproducibility. Thompson, on the other hand, has been taking another approach to evolving more robust solutions, by using multicriteria fitness definitions, by having several evolutionary tests (e.g., different voltage, different site on the chip, different temperature, etc.) whose performance values contribute to the total fitness value (e.g., the sum of the subfitnesses). Recently Thompson has succeeded in this approach, but I suspect that Levi's approach will be the one to endure.

One of the several advantages of the XC6200 family of programmable chips for E-Hard was its public architecture, so that non-Xilinx people understood what each configuring bit does. Also, the XC6200 family can accept random configuring bit strings without "blowing up." This is not the case with the 4000 and new Virtex families. Hence outsiders cannot send random bit strings to these chips, making these chips useless for intrinsic E-Hard. However, Levi is a Xilinx insider. He knows the function of each configuring bit and hence was able to write a piece of Java code to enable anyone to perform E-Hard on the 4000 and Virtex chips. His software is called GeneticFPGA and can be obtained at Delon.Levi@xilinx.com Levi hopes to build electronic boards containing his GeneticFPGA, an evolvable chip (4000 or Virtex), and a PC interface for about $1000 to $5000. Such a kit would open up the field of E-Hard to researchers, although if Xilinx were to open up its architectures, the field would be opened up more. But, if there are no more chips due to Xilinx going broke, that would be even worse. I asked an electronic engineer acquaintance if there is any alternative to Xilinx's Virtex for the next generation of E-Hard. He said effectively no, but had doubts about the partial reconfigurability of the Virtex. Hopefully future versions of the Virtex will have better partial reconfigurability. Independently of whether this happens in the next year or two or not, Levi's contributions will ensure that intrinsic E-Hard is robust from now on and that E-Hard can be executed on a broader range of chips. This is significant. It will save the field, making Levi an E-Hard hero.

Hugo de Garis' E-Hard Based Artificial Brain

I spoke about the CAM-Brain Machine (CBM), a 72-Xilinx XC6264 chip piece of hardware (costing $400 000 each, manufactured by Genobyte, Inc. under the direction of M. Korkin) that can evolve a cellular automata-based neural net module of some 1000 neurons in about 1 s. The module is then downloaded into a gigabyte of RAM containing up to 64000 of these modules interconnected according to the design of human "brain architects" to make a 75-million neuron artificial brain. The goal of the work is to have a CBM update the RAM brain at 130 billion three-dimensional CA cells a second to control a life-sized robot kitten possessing hundreds of different behaviors in real time.

I also spoke about some recent simulation experiments in evolving 2-D static and dynamic pattern detector modules and their ability to generalize.

IV. Remarks

The workhorse of the evolvable hardware field has traditionally been Xilinx's XC6216 chip which enabled the field. A. Thompson used this chip to evolve intrinsically his famous frequency detector circuit (which output 0 V if the input square wave signal oscillated at 1KHz and output 5 V if the input square wave oscillated at 10 KHz). De Garis and Korkin use the XC6264 chips to evolve their cellular automata-based neural net circuit modules for their artificial brain. The XC6264 chips are no longer purchasable since Xilinx took them off the market due to too little demand from the electronic engineering community. Korkin bought the remaining chips (untested) which were enough for only seven CBM's.

Similarly, the number of field programmable analog array (FPAA) chip manufacturers has dropped from three to one (Zetex). Motorola, who used to manufacture programmable analog arrays, has stopped doing so.

Oddly, the lack of evolvable hardware chips was not an issue at the workshop, and it should have been. Higuchi is designing his own ASIC chips. Levi has allowed such Xilinx chip families as the XC4000 series and the recently unveiled "Virtex" family to be evolvable, so long as you obtain the appropriate software from him.

The problem with the Virtex chips (1–2 million programmable gates) is that they are less partially reconfigurable as compared to the XC6200 family. Partial reconfigurability is fundamental to evolvable hardware because one does not want to have to reconfigure the whole bit string when only a tiny portion of it is mutated by an evolutionary (genetic) algorithm. One wants to be able to reconfigure only the correspondingly tiny part of the circuit, leaving the rest intact. Evolvable hardware would not be much faster than evolvable software if one had to reconfigure a 100K bit string every time one mutated one bit.

Hopefully Xilinx will hear the "cri du coeur" from the evolvable hardware research community and will make their future Virtex chip versions more partially reconfigurable. Evolvable hardware may prove to be the future of their industry. Right now the Xilinx company is cashing in heavily with the growing demand for reconfigurable chips. If killer applications can be found using E-Hard, the evolvable hardware market should take off. That has not happened yet, and unless Xilinx can make its Virtex chips more fine-grained reconfigurable, it may never happen. Xilinx may be effectively shooting itself in the foot through lack of vision. Evolvable hardware is the obvious logical extension to reconfigurable hardware. I believe it is the next mass market.