Genetic Programming and Evolvable Machines, 1, 171–174 (2000)
© 2000 Kluwer Academic Publishers. Manufactured in The Netherlands.

Review: First NASA / DOD Workshop on Evolvable Hardware 1999

Published by: IEEE Computer Society, ISBN 0-7695-0256-3

Editors: Adrian Stoica, Didier Keymeulen and Jason Lohn

The workshop took place at the Pasadena Convention Centre on July 19–21, 1999. There were 9 invited talks and 35 technical papers. The session topics were as follows: Evolution on FPGAs (4), Evolution of Digital Functions (3), Evolution of Analog and Mixed-Signal Circuits (5), Evolution of Cellular Automata and Brain Inspired Architectures (6), Reconfiguration Architectures and Dynamic Reconfiguration (4), Advanced Reconfigurable Devices (3), Applications to Design and Adaptation of Space Sub-Systems (5), GA Applications (5).

The nine invited talks were: NASA Rediscovers Technology (W. Huntress); Micro/Nano Systems for Future NASA Grand Challenges in Deep Space Exploration (L. Alkali); DARPA's Adaptive Computing Systems Program (J. Munoz); Identifying Requirements of Evolutionary Design Search, Exploration and Optimisation (I. Parmee); Evolvable Hardware for Industrial Applications (T. Higuchi); Evolving Circuits by Means of Natural Selection (J. Koza); Explorations in Design Space: Can Evolutionary Algorithms Practically Search Beyond the Scope of Conventional Electronics Design? (A. Thompson); Embriological Electronics (P. Marchal); The Effects of Extreme Environments on Measurement Equipment (M. Buehler) and Dusting off Some Evolvable Hardware (D. Fogel).

It was clear from some of the invited talks that reconfigurable and evolvable hardware systems were going to be increasingly important in remotely operated and autonomous spacecraft. Wes Huntress pointed out the high cost, large mass and long development time of early NASA spacecraft. This implied that relatively old technology was being used and infrequent space missions. Currently NASA expects up to 10 missions per year and is aiming at table-top sized spacecraft with flexible hardware and software that can adapt in situ. Leo Alkali pointed out that NASA had set up the Center for Integrated Space Microsystems (CISM) precisely to try and plug the gap between mission planning and the latest research. In future NASA missions there would have to be much less redundancy and a much greater sharing of common resources. Jose Munoz indicated that reconfigurability was a major theme in DARPA's Adaptive Computer Systems program. The big stumbling block at present is the large reconfiguration time ($\approx$ 10ms) of modern platforms.

There was a wide range of papers presented at the workshop and it was clear that the term evolvable hardware was seen to cover not only the design of electronic circuits using evolution but also helicopter and rocket booster design, satellite distribution, robot and general engineering structure design, antennae, resonant tunneling diodes, infrared filters, sorting and communication networks. Most of the papers were relatively unconnected with each other and addressed specific application issues. The field of evolvable hardware is still new and most workers are content to explore what evolvable hardware is capable of.

D. Levi of Xilinx Inc. made a noteworthy contribution with his GeneticFPGA system that allowed the evolution of digital circuits on mainstream FPGA devices (XC4000EX/XL). He used Jbits to transform a genetic bitstream into a valid configuration data and then a software interface (XHWIF) to configure live FPGAs on networked boards. GeneticFPGA eliminated the problems associated with the analog domain by forcing all the circuits to use synchronous signals. He was able to evolve 1-hot counters and frequency dividers. However to evolve an 8-bit 1-hot counter required about 15 hours of processing time. This was mainly caused by the slow reconfiguration time of the FPGA.

Zebulum et al. presented their latest work on the extrinsic evolution of active filters using the SPICE simulator and showed that it was possible to produce filters that satisfied a number of user defined criteria, such as, frequency response, minimal power dissipation, etc. They also evolved a low pass filter intrinsically on a Motorola FPAA (MPAA020). Intrinsic evolution was again slow due to large reconfiguration times. They suggested that at present it is more practical and effective to evolve the circuits in simulation and with appropriate care the discrepancy between simulated circuit and the same design realized in hardware can be minimized. This contrasted with the findings of Stoica et al. in their work on evolving circuits on a programmable transistor array. They did this in hardware and in a SPICE simulation. They found that the former did not work in simulation and the latter didn't work when implemented in hardware. However the hardware evolved solutions did work on other transistor arrays. One should be cautious about concluding that one should only evolve analog circuits intrinsically. It is clear that some types of circuits are much easier to simulate than others and it can be quicker to evolve these in simulation than in real hardware. Tufte and Haddow argued that for larger designs with more complex fitness criteria a Complete Hardware Evolution (CHE) system is needed. In CHE both the fitness evaluation and evolutionary algorithm are completely implemented in hardware. Such a CHE system would be quite fault tolerant as the designs could be re-evolved.

De Garis et al. and members of the Polish "Evolutionary Engineering Club" presented some work on the capabilities of CAM-Brain CoDi-1Bit neural net modules. They found that it was possible to evolve some of the desired functions quite easily even though the chromosome length was of the order of 90K bits. The paper was improved by the reduction in hyperbole that is usually associated with "brain building." Pollack et al. asked "Where are the robots?" They noted that many researchers in robotics had grossly underestimated the complexity of brains and bodies of living things. They suggested that we should be co-evolving robot brains and bodies both simultaneously and continuously. They are attempting to do this in three ways: evolving structures inside sophisticated CAD tools, evolving buildable machines and evolving directly in real hardware.

In one of the most fascinating papers in the proceedings Linden and Altschuler described their work on the use of genetic algorithms in designing wire antennas. They pointed out that current methods of designing was a very laborious process that required engineers to have a very good knowledge of the universe of conventional antenna designs. Instead GAs did not require an initial reasonable design and the engineer was required to supply only a small amount of information. The designs produced by their GA had excellent gain characteristics with bizzare shapes that would not have looked out of place in a gallery of modern art. Interestingly in different runs the GA would produce designs with similarly good characteristics but very different shapes.

I was also impressed by the paper of Nicholas Macias on the design of a massively parallel fine grained self-reconfigurable infinitely scalable architecture (PIG). The PIG is a non von Neumann computing machine with no designed memory, no instructions and registers. It is a pure dataflow machine. Every evolvable hardware research group should have one (they are very affectionate!).

I welcomed the friendly argument of Marek Perkowski et al. who thought that "black box evolution" was not going to be as effective as symbolic learning. They claimed not to be able to find a single example where a GA based algorithm was superior to a hand-crafted algorithm for the design of a binary logic network. Although I personally dispute this it is important that evolutioneers engage in debate with those from the more established design and artificial intelligence communities.

I found George Milne's paper on his attempts to develop a formal language for modeling the concurrent and self-reconfigurable systems thought provoking. I was also intrigued by the paper by Mjolsness et al. who described a computational model of a key aspect of morphogenesis in plants, the shoot apical meristem. The essential aim was to learn from and reproduce (robotically) methods of engineering in biological systems.

Klimek et al. described some very interesting work on using a genetic algorithm in association with quantum device design and analysis tool (called NEMO) to design quantum resonant tunneling devices. Simulation is the only way a large number of different designs can be evaluated. There is probably a lot of scope for interesting research by using GAs together with sophisticated simulators in many fields of study (engineering, chemistry, physics, etc.).

Masner et al. presented some work on evolving sorting networks both in simulation and using reconfigurable hardware. They examined three different chromosome representations. They defined an interesting metric for gate-level resilience to faults: bitwise stability. It appeared that evolution inherently improves this and that tree structured chromosomes had more bitwise stability than linear chromosomes.

The workshop closed with a lively panel session panel session. Adrian Thompson stressed the need for fully open architecture reconfigurable chips with fast and partial reconfiguration. Steve Trimburger of Xilinx suggested that researchers would be very happy with the new Virtex chip. Leo Alkali urged people not to be too obsessed with current technologies and instead to look at bio-inspired devices at the molecular level. There was a general feeling that much better software tools were still required to help make the process of reconfiguring chips much easier.

Unfortunately it is not possible to comment on all of the many interesting papers at the workshop. It certainly opened my eyes to many of the future directions in the area of evolvable machines.

J. F. Miller