Evolvable Hardware - User Experience

IT3708 - Bio-Inspired Artificial Intelligence

Sigve A. E. Skaugvoll
MIT, Artificial Intelligence

19.04.2018

Norwegian University of Science and Technology
Department of Computer Science

1 Literature Search Protocol — Review Protocol

1.1 Planning

1.1.1 Research Question

Before having read many research papers, I kept asking myself the same question over and over again,

- A) Are computers evolving and improving the user experience over time - when we have evolvable hardware?
- B) Are there still a future for evolvable hardware?

1.2 Review protocol

1.2.1 Search words, Sources and Search strings

I used the following search engines; IEEE Xplore and Springer. The search query structure included; evolvable hardware, adapting or improving, user experience, behaviour, etc. One search query I used was ”Evolvable OR adaptive AND hardware AND user experience”.

1.2.2 Selection Criteria

I wanted to exclude papers and research older than 2005.

Inclusion Criteria	**Quality Criteria**
Describes a relevant system Focuses on a relevant method Published at conference Published as a journal	Clear statement of the aim of the research Is objective and does not try to ”sell” their results

Table 1: Inclusion- and quality criteria

2 Evolvable Hardware

Evolvable hardware is a field within evolutionary algorithms. The purpose is to create hardware that can adapt and change according to its environment, to a better or optimal configuration.
The hardware environment can typically change in two ways. There is hardware failure and other components change its architecture or is updated, thus removing or breaking the direct or in-direct communication between the evolvable component and the updated.
Thus the goal of evolvable hardware is to modify the component- behavior or structure at run time in order to adapt changes of user requirements or working environments.

The research question 1.1.1.a states user experience, and the definition of user experience used here is [1]. Evolvable hardware is difficult to spot in day to day use when it comes to user experience, because hardware is not as visible to the user as an graphical user interface is. Thus the user can only experience the reconfiguration trough the usage and behaviour of the computer and is not explicitly made aware of the changes. It happens seamlessly with out the user ever knowing, but still as the time passes our computer become noticeably slower and less-responsive to the tasks we wish to execute. One should think the evolvable hardware would make the computer better suited for the usage pattern of the owner, and thus improve the computer over time. Why is it that this isn't the case?

2.1 Adapting to usage patterns

When searching for research done on evolving hardware to explicitly improve user experience, there was surprisingly few query matches. It seems like there still is a gap between evolvable hardware- and adapting software with focus on user experience. Evolvable hardware is more focused on the engineering-focus and hardware fault tolerance. Hardware failure is indirectly focusing on user experience since user want their product to work for as long as possible without having to manually repair it. "self-healing is the system's ability to carry out all the actions necessary to restore proper functioning without human intervention" [2]. I find this quite interesting, as it states 'proper functioning', how does one define proper? Is it that as long as the device starts and can execute and take as long time as it needs, or it there any criteria? It has to be able to maintain the same speed as when brand new out of the box? I sure hope that there is some criteria. And since the main

field within self-healing is evolutionary algorithms, there has to be some sort of fitness. Is it possible to write an algorithm that extracts and adapts to usage patterns. Such algorithms obviously has to be Multi-objective optimization, because there are many criteria, such as speed, memory cost, and durability of the system. There seems to be little research on finding usage patterns, such as, which programs are used together, what types of operation do they perform, are there often I/O operations which occupies the motherboard-bus and how can the hardware be optimized to most efficiently execute these instructions? E.g for a programmer, there is a text-editor or IDEA running, which uses both CPU and if desired and possible, the GPU. There is often I/O operations to secondary memory.

Can the hardware and circuit be re-configured to always strive to have the I/O bus ready, and can certain hardware communication and instruction stored in RAM which is not used while the identified programs are running, be switched out with the identified endpoint of the I/O operations? Thus increasing the read-write speed, and change the configuration of the CPU to give higher rank and if possible over-clock the more desired tasks and thus re-configure the CPU-schedule, find minimum distance and strongest connection between hardware components to give improved efficiency and effectiveness. If parts of the hardware can be optimized to execute such user-desired operations, evolvable hardware has a bright future! Hopefully [3] is on the right way with GPLC and MILP and they state: "The proposed concept is assumed to provide the productivity-increasing balance of core configurability between traditional "fixed" designs and full separate language/compiler infrastructures that requires relatively modest programming efforts to so significantly improve the designing process". One clearly sees that there is a huge focus on the engineering aspect, where the goal is to produce useful solutions, but one gathers inspirations from the science aspect - where one wants to understand and simulate the workings of the natural world.

2.2 Is there a future for evolvable hardware?

Already back in 1996 [4] and earlier, there were huge interest in evolvable hardware. One hoped that using science to figure out how one can simulate nature to reproduce how nature make biological changes, when the environment changes "research needs to address issues such as scalability, on-line adaption, generalisation, circuit correctness..." All of this has main focus on the engineering aspect, where the usefullness of the solution is key. Thus there were and still are two main focuses when it comes to evolvable hardware, science and engineering. One uses science aspect to figure out how new ways and try to create components that

are re-configurable and mimic nature, but the main goal is engineering focus, useful solutions. Now over 25 years later, how far have evolving hardware come? I think it has come a long way and is being used for very specific tasks such as ”self-repairing control circuit for brush-less dc motor”[5]. Even though it has not come as far as making it possible to create as general and abstract EA that can recognize usage patterns, and reconfigure big parts of the motherboard hardware. I think that evolvable hardware as simulating the environment to find best specific solution has come further than the adaptive evolving hardware (reacts to live-environment). I can say this with confidence based on the results I got during 1.2.

3 Existing solutions

There are few if any existing solutions to 1.1.1.a and not much explicit research on that topic can be found, I got 48 possible matches, see figure 1. I think such research and development is still limited by the fact that not all hardware is reconfigurable, and thus limited to circuit optimization. The majority of the query result are about network configuration and how to increase the hardware failure. Thus not directly the type of user experience this essay focuses on.

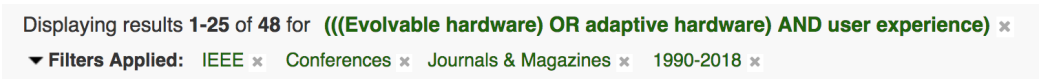

Figure 1: Query result

There certainly are a huge community researching evolving hardware and there have been found many interesting and important fields where evolving hardware is applied with good results. Existing solutions that applies evolving hardware can be found in the following list.

- Evolving both Quantum and regular circuits in advance towards known usage pattern and tasks [6].
- self healing/repair of circuit to handle hardware failure and give more stability in user experience [2].
- Resource management [7].

Even though there are few solutions regarding explicit user experience, it’s still possible to imagine a future wear such an solution exists.

4 Possible solutions

I personally think that in the future there is a method that allows evolving hardware to cooperate with software adaption and thus usage pattern-recognition and optimalization. Such an solution has some requirements such as more re-configurable hardware components. It would probably mean that the number of computers bought would go down and the accuracy and speed of computers would increase so that running specific tasks such as machine learning and Artificial intelligence would be much more efficient and thus further improve the computational power, which again would allow for more research and discoveries.

5 Conclusion

Evolvable hardware has come a long way, and are applied in very specific field such as circuit design, improving the hardware fault tolerance and network problems. There is little research done explicitly on 1.1.1.a. There is a missing bridge over the gap between evolvable hardware and adaptable software, which is necessary to build, before evolvable hardware and evolutionary algorithms can produce useful solutions which learns user behaviour and pattern, which the evolvable hardware can adapt to, so that the performance for the tasks performed is improved.

Regarding research question 1.1.1.b. Evolvable hardware has come along way and is applied to create and maintain circuits both in simulation before production and in production. I think there is a lot of possibilities with evolvable hardware and is eager to see what the future brings. If someday the evolvable and adaptive systems becomes seamlessly real, I think a lot is going to change because the computation power needed for further discoveries within information systems and research is now limited, and very expensive to build purpose dedicated machines. What is the probability of such a system and architecture in the future, my prediction is, high.

References

[1] “Iso-9241-210.” [Online]. Available: https://www.iso.org/standard/52075.html

[2] G. Martinović and I. Novak, "A combined architecture of biologically inspired approaches to self-healing in embedded systems," *Smart Systems and Technologies (SST), 2017 International Conference on*, pp. 17–22, 2017.

[3] A. Antonov, P. Kustarev, and S. Bikovsky, "Improving microarchitecture design and hardware generation using micro-language ip cores," *Nordic Circuits and Systems Conference (NORCAS): NORCHIP and International Symposium of System-on-Chip (SoC), 2017 IEEE*, pp. 1–6, 2017.

[4] X. Yao and T. Higuchi, "Yao x., higuchi t. (1997) promises and challenges of evolvable hardware. in: Higuchi t., iwata m., liu w. (eds) evolvable systems: From biology to hardware. ices 1996. lecture notes in computer science, vol 1259. springer, berlin, heidelberg," *Adaptive Hardware and Systems (AHS), 2017 NASA/ESA Conference on*, pp. 55–78, 2005.

[5] P. Zhu, R. Yao, and J. Du, "Design of self-repairing control circuit for brushless dc motor based on evolvable hardware," *Adaptive Hardware and Systems (AHS), 2017 NASA/ESA Conference on*, pp. 214–220, 2017.

[6] C. Ruican, M. Udrescu, L. Prodan, and M. Vladutiu, "Ruican c., udrescu m., prodan l., vladutiu m. (2010) adaptive vs. self-adaptive parameters for evolving quantum circuits. in: Tempesti g., tyrrell a.m., miller j.f. (eds) evolvable systems: From biology to hardware. ices 2010. lecture notes in computer science, vol 6274. springer, berlin, heidelberg," *Adaptive Hardware and Systems (AHS), 2017 NASA/ESA Conference on*, pp. 348–359, 2010.

[7] A. G. Arpit Christi and R. Gopinath, "Resource adaptation via test-based software minimization," *Self-Adaptive and Self-Organizing Systems (SASO), 2017 IEEE 11th International Conference on*, pp. 61–70, 2017.