Reconfigurable Analogue Hardware Evolution of Adaptive Spiking Neural Network Controllers

Brian Mc Ginley
BIRC Research Group
National University of Ireland, Galway
brian.mcginley @nuigalway.ie

Patrick Rocke
BIRC Research Group
National University of Ireland, Galway
patrick.rocke @nuigalway.ie

Fearghal Morgan
BIRC Research Group
National University of Ireland, Galway
fearghal.morgan @nuigalway.ie

John Maher
BIRC Research Group
National University of Ireland, Galway
john.maher @nuigalway.ie

ABSTRACT

This paper details the hardware evolution of adaptive Spiking Neural Network (SNN) controllers, implemented on a network of cascaded Field Programmable Analogue Arrays (FPAAs). The fixed architecture, feed forward SNNs are trained using a Genetic Algorithm (GA). An obstacle avoidance simulated robotics controller application is chosen to test the FPAA reconfigurable hardware evolution platform. Evolved behaviours, resulting from FPAA-based SNN controllers, are compared with those obtained using software-based SNN implementations. Results presented indicate the emergence of effective behaviours and adaptation to environmental change.

Categories and Subject Descriptors

B.8.1 [**Performance and Reliability**]: Reliability, Testing and Fault Tolerance

General Terms

Experimentation, Design.

Keywords

FPAA Hardware Evolution, Spiking Neural Networks, Analogue Neural Networks.

1. INTRODUCTION

This paper describes the evolution of fixed architecture hardware Spiking Neural Network (SNN) controllers. The hardware SNNs are implemented on a cascaded network of Anadigm Field Programmable Analogue Array (FPAA) devices [1] interfaced to a simulated obstacle avoidance robotics controller.

Adaptation to environmental change or faults is a key quality for systems where manual reconfiguration or repair is impractical. Plante et al [7] discuss the potential of FPAAs as an enabling technology for high reliability space flight systems while Hereford [4] employs evolution to generate robust fault-tolerant sensor systems using FPAA technology. Similarly, in this paper, adaptation is achieved through re-evolution. Results of FPAA SNN adaptation are presented and compared to those of a software SNN implementation.

GECCO'08, July 12–16, 2008, Atlanta, Georgia, USA.
ACM 978-1-60558-130-9/08/07.

FPAA-based evolved hardware ANN implementations have been reported in [2, 5, 8]. Berenson [2] et al evolved fault tolerant biped locomotion using 2 cascaded FPAAs. Berenson's system employed a 2-layer threshold based ANN to implement a robotics controller evolved solely in hardware. FPAAs have also been employed by Terry et al. [9] in evolving analogue controller circuits.

The SNN controller described in this paper is implemented in FPAA hardware and evaluated using a simulation model. This use of a hybrid hardware SNN and simulated environment for controller evolution offers a viable platform for evolutionary investigation of bio-inspired methodologies.

Results presented in this paper indicate that through the combination of GAs and SNNs, a network of FPAAs can be employed to successfully evolve adaptive obstacle avoidance behaviours.

2. FPAA HARDWARE PLATFORM AND SNNS

2.1 FPAA Hardware

The reconfigurable analogue hardware platform employed for SNN implementation is a network of AN221E04 FPAAs from Anadigm Inc. [1].

A leaky integrate-and-fire spiking neuron model [3] is used for FPAA SNN implementation. Genomes describing evolved SNN neuron weights and thresholds are translated for implementation on the FPAA array using Anadigm Designer 2.

2.2 Multi-FPAA Spiking Neural Network

Due to limited hardware resources on the AN221E04 FPAA, a maximum of two spiking neurons can be implemented using a single FPAA [8]. To realise a network capable of implementing functionality suitable for a robotics controller, four AN221E04 FPAA devices have been cascaded together to form the SNN.

3. GENETIC ALGORITHM (GA)

A Population size of 30 and elitism are employed. Fixed point crossover is applied at a rate of 0.8 while both random-re-initialisation and gaussian mutation are applied at rates of 0.01 and 0.05 respectively.

Fitness assessment of the robotics obstacle avoidance solutions is achieved using a fitness function [6] which rewards individuals based on: survival time (T), High speed (the distance (D) traversed by the robot) and effective exploration of the environment; i.e., the aggregate (AWS) of left (aws1) and right (aws2) average wheel speeds. Two constants (α, β) are introduced to the fitness assessment. By varying the relative values of each of the constants, different behaviours are evolved by the GA. Equation 1 describes the robotics fitness function formula.

$$F = \frac{T + D\alpha}{AWS^{\beta}} \quad (1)$$

4. RESULTS

4.1 Adaptation to Environmental Change

A robotics controller is first evolved in a simple world before being transferred to progressively more complex worlds at generation 30 and 60. Figures 1-2 compare FPAA-based SNN controller with software SNN controller implementations.

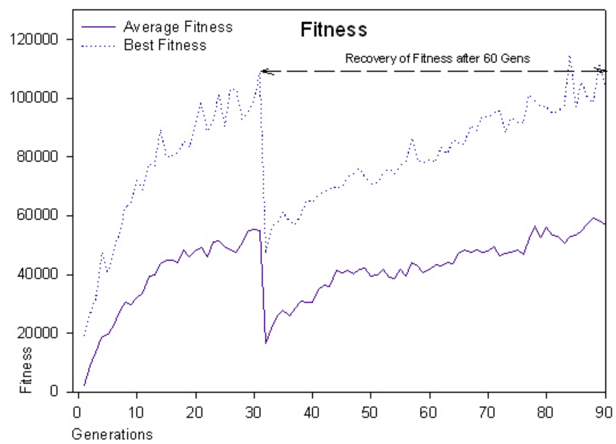

Figure 1: FPAA-based SNN Robotics Controller Fitness.

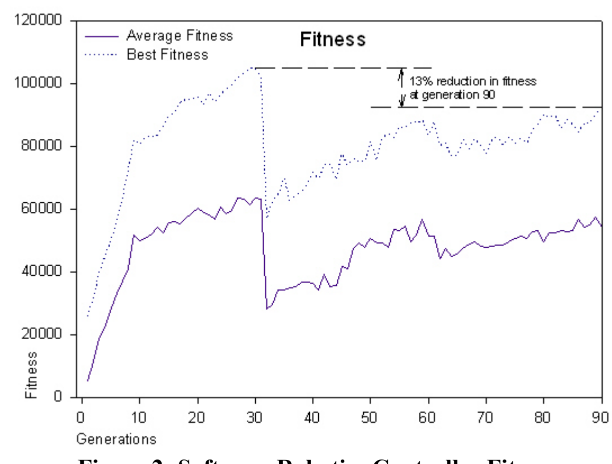

Figure 2: Software Robotics Controller Fitness.

Figure 1 illustrates that, following an environmental change event at generation 30, the FPAA-based SNN controller best-fitness is recovered at generation 90. This can be contrasted with the evolved software SNN implementation (Fig. 2) where fitness at generation 90 is still 13% below its initial peak fitness value.

Results indicate that the FPAA-based evolved SNN controller does not achieve the same level of fitness as the evolved software SNN. However, the FPAA system does display increased relative performance and robustness in adapting to changes in the environment. Noise inherent in the FPAA hardware SNN gives rise to these properties.

5. CONCLUSION

This paper describes the evolution of fixed architecture SNN-based controllers implemented on a network of cascaded FPAA devices, interfaced to a software robotics simulation model. This research focuses on evolution of SNN weights and thresholds.

Results demonstrate that FPAAs are a viable platform for implementing adaptive robust hardware-based reconfigurable SNN controllers.

Since FPAA device density is currently low, work on the creation of a large scale multi-FPAA module array is in progress to enable evolution of more complex SNNs. The array cascades FPAA devices to provide additional neurons and dynamic connection reconfiguration. The work reported in this paper can also be applied to future, higher density analogue platforms suitable for SNN implementation.

Research is also currently underway to employ FPAA-based SNNs as analogue input and output stages to/from large-scale FPGA-based SNN implementations.

6. ACKNOWLEDGMENTS

The authors thank the Irish Research Council for Science, Engineering and Technology (IRCSET) and Anadigm for providing FPAA devices.

7. REFERENCES

[1] Anadigm Field Programmable Analog Arrays. www.anadigm.com.

[2] D. Berenson, N. Estevez, and H. Lipson. Hardware Evolution of Analog Circuits for In-situ Robotic Fault-Recovery. Evolvable Hardware, 2:12–19.

[3] W. Gerstner and W.M. Kistler. Spiking neuron models. 2002.

[4] J. Hereford and C. Pruitt. Robust sensor systems using evolvable hardware. Evolvable Hardware, 2004. Proceedings. 2004 NASA/DoD Conference on, pages 161–168, 2004.

[5] J. Maher, B. Mc Ginley, P. Rocke, and F. Morgan. Intrinsic Hardware Evolution of Neural Networks in Reconfigurable Analogue and Digital Devices. Proceedings of the 14th Annual IEEE Symposium on Field-Programmable Custom Computing Machines (FCCM'06)-Volume 00, pages 321–322, 2006.

[6] S. Nolfi, D. Floreano, O. Miglino, and F. Mondada. How to evolve autonomous robots: Different approaches in evolutionary robotics. Artificial Life IV, pages 190–197, 1994.

[7] J. Plante, H. Shaw, L. Mickens, and C. Johnson-Bey. Overview of field programmable analog arrays as enabling technology for evolvable hardware for high reliability systems. Evolvable Hardware, 2003. Proceedings. NASA/DoD Conference on, pages 77–78, 2003.

[8] P. Rocke, J. Maher, and F. Morgan. Platform for Intrinsic Evolution of Analogue Neural Networks. Proceedings of the 2005 International Conference on Reconfigurable Computing and FPGAs (ReConFig'05) on Reconfigurable Computing and FPGAs, 2005.

[9] M.A. Terry, J. Marcus, M. Farrell, V. Aggarwal, and U.M. O'Reilly. GRACE: Generative Robust Analog Circuit Exploration. 9th European Conference on Genetic Programming, EVO-Workshops, EVOHOT track, 2006.