Kalganova T. (1999) A New Evolutionary Hardware Approach for Logic Design. Proc. of the GECCO'99 Student Workshop. Orlando, Florida, USA. Eds. Annie S. Wu. pp. 360-361.

A New Evolutionary Hardware Approach for Logic Design

Tatiana Kalganova

School of Computing
Napier University
219 Colinton Road,
Edinburgh, EH14 1DJ
t.kalganova@dcs.napier.ac.uk

ABSTRACT

This poster paper summarizes ongoing dissertation research defining an evolvable hardware methodology for evolving combinational binary and multiple-valued logic circuits. This dissertation provides an overview of current evolvable hardware approaches; defines the combinational logic design problem; describes the gate and function level evolvable hardware technique; and develops a new methodology for evolving binary and multiple-valued combinational logic circuits with and without automatically defined functions. The new methodology promises significant improvements over current conventional algebraic techniques.

1. Introduction

Traditionally the logic circuit design process contains two main phases: 1. Selection of an appropriate logic algebra; 2. Design of the circuit based on the chosen logic algebra. All of the well-known logic design methods suffer from problems when one tries to construct circuits from operators, which are not from different logic algebras. In addition there are some combinations of logic operators for which the algebraic properties are not completely understood. The idea of the evolvable hardware technique is to synthesise logic circuits using evolutionary algorithms that use assemble and test principles. This method merely connects the logic gates together and tests whether the resulting circuits are correct. The advantage of this method is that we can synthesise very novel structures and we have no restriction in terms of the logic algebra rules. This method is designed for multiple-valued and binary *n*-input *m*-output logic functions. The dissertation provides a gate- and function- level evolvable hardware technique for designing multiple-valued and binary combinational circuits. The evolution of the circuit layout, connectivity and functionality of a rectangular array of logic cells is the main feature of proposed method. At the gate level the logic cells implement simple logic unary and two-input operations, on the function level the logic cell describes either *n*-input one-output or *n*-input *m*-output logic functions. The *n*-input *m*-output logic function can be defined as standard logic function such as adder or multiplier. The developed technique allows us to define the *n*-input *m*-output logic function during evolution. Thus we can say that circuit can be evolved with automatically defined *n*-input *m*-output logic functions. Applying this technique to logic design allows us to synthesise circuits with logic cells associated with different sets of logic gates and from a mixture of different logic algebras. We intend to apply these techniques to a number of problems such as signal processing, arithmetic circuit design, etc.

2. Evolvable Hardware Approach

Phase I of this research establishes an evolvable hardware approach for evolving multiple-valued logic functions in different functionally complete bases and to investigate the conditions that allow us to firstly produce the most 100% functional solutions and secondly the most efficient circuit implementation. The extension of the method to multiple-valued logic allows us to solve problems in both binary and multiple-valued domains, because binary logic is a particular case. A number of experiments were performed to investigate which set of gates allows the desired functions to be most easily evolved. We have found that using the entire set of all possible logic gates produces relatively poor results. Experiments have shown that there is an optimal circuit layout, which allows us to produce the highest number of 100% functional solutions. Using a two-fitness strategy allows us to not only evolve 100% functional circuits but also to improve its implementation in terms of the number of logic gates used. This strategy is defined as follows: initially the circuit functionality is evolved, and then once the 100% functional circuit is obtained, the number of logic gates used is taken into account. The definition of two-fitness strategy gives us idea of using multi-objective fitness function instead. Thus, multi-

objective approaches such as the Pareto technique, with weight aggregation could be experimented with. Comparison of several multi-objective techniques allows us to define the best suitable fitness strategy for given problem.

In order to define the behaviour of a circuit layout during evolution, the method mentioned above for the flexible circuit layout is extended in phase II. The GA operators and the chromosome representation are changed according to the specific features of the circuit layout evolution. The experimental results show that the circuit layout of the best chromosome of the population is changed at the beginning and the circuit layout stabilised at a high number of generations.

The technique mentioned above allows us to synthesise the logic functions at gate-level, which may be inefficient compared with using subfunctions of large numbers of variables. For this reason, the multi-input one output and multi-input multi-output logic gates could be used. A multi-input one-output function is defined by the interactive connection of the same two-input one-output gates. In terms of multi-input multi-output representation of logic gates two strategies can be used. Multi-input multi-output logic gates can represent 1. A standard logic function; 2. An automatically defined function. The first strategy allows us to utilise the standard logic functions used in standard conventional logic design, such as adders and multipliers of any digits for gate representation. The second strategy represents logic gates based on the basic concept of gate connectivity. An automatically defined function is described by type of two-input one-output logic gates connection and types of logic gates used on the each level of structure. We use two main types of logic gates connections: 1. Interactive; 2. Cascaded. When interactive type of logic gate is used the two-input one-output gates are connected by chain principle, otherwise the cascaded principles to connect logic gates are applied. The structure of logic gate can be changed during circuit evolution. Using different representation of logic gates allows us to evolve a logic circuit on both gate and function level. This dissertation concludes during Phase III by extending the evolvable hardware method of Phases I-II to design the logic functions of large number of inputs and outputs.

We intend to apply these techniques to a number of interesting problems (i.e. signal processing, arithmetic circuit design).

3. Summary

The designed evolvable hardware approach allows us to synthesise logic circuits at the both gate- and function levels. Using multi-objective strategy allows us to evolve 100% functional circuits with the minimal number of elementary (two-input one output) logic gates. Using different types of logic functions to describe logic gates permits us to design combinational logic circuits of different complexity. Applying automatically defined strategies to introduced methods allows us to extract new basic blocks to design combinational logic circuits. The basic concept of this method can be applied to the different application problems connected with circuit design. A number of experimental results are discussed above have been reported in the publications listed below.

Acknowledgements

The author gratefully acknowledges the School of Computing, Napier University for its support of this research.

Bibliography

Kalganova, T., J. Miller and T. Fogarty (1998). Some Aspects of an Evolvable Hardware Approach for Multiple-Valued Combinational Circuit Design in *Proc. Of the 2nd Int. Conf. on Evolvable Systems (ICES'98).* Lausanne, Switzerland, Eds.: M. Sipper, et al. Publisher: Springer-Verlag, pp. 78-89.

Miller J., T. Kalganova, N. Lipnitskaya and D. Job (1999). The Genetic Algorithm as a Discovery Engine: Strange Circuits and New Principles in *Proc. Of AISB Symposium on Creative Evolutionary Systems (CES'99)*, Edinburgh, UK.

Kalganova T. and J. Miller (1999) Evolving more efficient digital circuits by allowing circuit layout evolution and multi-objective fitness. *Proc. Of the First NASA/DoD Workshop on Evolvable Hardware*, Pasadena, California, USA.

Kalganova, T., J. Miller and N. Lipnitskaya (1998). Multiple-Valued Combinational Circuits Synthesized using Evolvable Hardware Approach in *Proc. of the 7th Workshop on Post-Binary Ultra Large Scale Integration Systems (ULSI'98) in association with ISMVL'98*, Fukuoka, Japan. Publisher: IEEE Press. pp. 52-54.

Kalganova T., J. Miller and T. Fogarty (1999). Evolution of the digital circuits with variable layouts in *Proc. of the Genetic and Evolutionary Computation Conference (GECCO'99)*, Orlando, Florida, USA.