Development of Combinational Circuits Using Non-Uniform Cellular Automata: Initial Results

Michal Bidlo
Faculty of Information Technology
Brno University of Technology
Božetěchova 2
61266 Brno, Czech Republic
bidlom@fit.vutbr.cz

Zdenek Vasicek
Faculty of Information Technology
Brno University of Technology
Božetěchova 2
61266 Brno, Czech Republic
vasicek@fit.vutbr.cz

ABSTRACT

A non-uniform cellular automata-based model is presented for the evolutionary development of digital circuits at the gate level. The main feature of this model is the modified local transition function of the cellular automaton in which a gate is associated with each rule of the transition function. A logic gate is generated by each cell when the cell determines its next state according to the appropriate rule. An evolutionary algorithm is utilized to design a non-uniform cellular automaton (its local transition function) for the development of a target circuit. In this paper, initial results will be presented that were obtained using the non-uniform cellular automata.

Categories and Subject Descriptors

B.6.1 [**LOGIC DESIGN**]: Design Styles—*cellular arrays and automata, combinational logic*

General Terms

Algorithms, Design, Experimentation

Keywords

Evolutionary algorithm, development, cellular automaton, combinational circuit.

1. INTRODUCTION

In nature, the development is a biological process of ontogeny representing the formation of a multicellular organism from a zygote. It is influenced by the genetic information of the organism and the environment in which the development is carried out.

In the area of computer science and evolutionary algorithms in particular, the computational development has been inspired by that biological phenomena. Computational development is usually considered as a non-trivial and indirect mapping from genotypes to phenotypes in an evolutionary algorithm. In such case the genotype has to contain a prescription for the construction of target object. While the genetic operators work with the genotypes, the fitness calculation (evaluation of the candidate solutions) is applied on phenotypes created by means of the development. The principles of the computational development together with a brief biological background and selected application of this bio-inspired approach are summarized in [4].

Cellular automata (CA) represent one of the methods to perform the computational development (i.e. to simulate the basic principles of that biological phenomena by means of digital computational devices). Since the invention of the basic concept of the cellular automata in 1966 [7], CA have been successfully applied to investigate many complex problems in different areas. The detailed survey of the principles and analysis of various types of cellular automata and their applications (including emulation of circuits and computer systems) is summarized in [8]. Sipper investigated the computational properties of cellular automata and proposed an original evolutionary design method for cellular automata called the cellular programming [5]. Other applications of CA include, for example, synthesis of easily testable combinational logic [3], pattern development and self-organization [2], FPGA-based cellular automata development of sequential circuits [6] or gate-level development of combinational circuits using 1D uniform CA [1].

In this paper, the evolutionary design of non-uniform CA will be presented for the development of digital circuits at the gate level. The impact of the cellular automata size (i.e. the number of cells) on the success rate and computational effort of the evolutionary processes will be investigated.

2. CELLULAR AUTOMATA-BASED DEVELOPMENTAL SYSTEM

A logic gate is assigned to each rule of the local transition function of each cell (i.e. for each combination of states in the cellular neighbourhood for which the next state is calculated). The gate will be generated by the cells during the development of the cellular automaton. The gate to be generated is specified by the rule that is applied to determine the next state of the given cell. The developmental step is considered as the calculation of the next state for each cell of the CA together with generating a gate by the cells. Considering this developmental scheme, one level of the circuit is generated in one developmental step of the CA. In case of the first developmental step, the inputs of the gates being generated are connected to the primary inputs of the target circuit. The inputs of gates generated in the next developmental steps are connected to the specified outputs of the gates generated in the previous developmental step. The outputs of the appropriate gates generated in the last developmental step represent the primary outputs

GECCO'09, July 8–12, 2009, Montréal, Québec, Canada
ACM 978-1-60558-325-9/09/07.

Table 1: Success rate and computational effort of the evolutionary development of combinational dividers for the evolving and fixed initial state of the CA. W denotes the number of cells, S denotes the number of states and T denotes the number of developmental steps of the CA in which the circuits were developed.

			Evolved init. st.		Fixed init. st.	
W	S	T	S. rate	C. effort	S. rate	C. effort
10	3	4	35	16k	28	16k
		5	72	16k	70	11k
	4	4	27	17k	20	21k
		5	74	17k	65	19k
15	3	4	51	15k	51	15k
		5	83	10k	89	7k
	4	4	36	20k	32	22k
		5	81	13k	81	13k
20	3	4	53	18k	55	14k
		5	96	8k	96	8k
	4	4	36	22k	37	19k
		5	81	14k	86	15k

of the target circuit. No other interconnection of the gates are allowed. The building blocks considered for the circuit development include identity function (a wire), two input logic gates AND, OR, XOR and their inverted variants.

3. EVOLUTIONARY SYSTEM SETUP

The simple genetic algorithm was utilized to design the local transition functions of the CA that can develop the target circuit. Each cell has associated a complete local transition function encoded in the genome. The local transition function of a cell is represented by its rules that are encoded by 4-tuples. These rules contain the next state of the cell, the gate (function) and the indices of inputs of that gate that is generated when the rule is activated, i.e. when the cell determines its next state according to the given rule. The population consists of 200 randomly initialized genomes. A tournament selection operator with the base 4 is applied. The mutation randomly selects 6 genes, each of which are mutated with the probability 0.96. No crossover is applied. The fitness function is calculated as the number of correct output bits of the target circuit using all the binary test vectors. In order to evaluate the average success rate and computational effort, 100 independent evolutionary experiments were conducted.

4. EXPERIMENTAL RESULTS

The initial experiments of the non-uniform CA-based approach considers the development of combinational dividers because this class of circuits we have not been able to develop successfully using the uniform CA. In the initial experiments, 5-input dividers were considered. In this case, there are 3 bits of a dividend and 2 bits of a divisor. The quotient is then expressed on 3 bits. If the divisor equals zero, the quotient is not defined (i.e. its bits are considered as don't care values during the circuit evaluation). The results are summarized in Table 1. The maximal number of generations for measuring the success rate was 50k. The evolution has succeeded in the design of of both CA considering the evolving initial state and CA in which the initial state of all the cells was set to zeros and only the local transition function was evolved. As Table 1 shows, the success rate increases with increasing the size of the CA. However, it does not differ very markedly if compared the evolving and fixed initial states. In contrast to [1], where the fixed initial state provided better results in most cases, in the experiments presented herein the fixed initial state exhibits worst results in many cases. It is surprising because the fixed initial state reduces the search space.

5. CONCLUSIONS

A non-uniform cellular automata-based developmental system was introduced for the evolutionary design of combinational circuits. The initial experiments showed that evolution is able to design non-uniform CA for the development of combinational dividers. The results considering the design of 5-input dividers were presented. In the next research, other classes of circuits will be investigated with respect to the proposed approach (e.g. combinational multipliers). Moreover, the ability of the non-uniform CA to develop larger circuits will be investigated.

6. ACKNOWLEDGMENTS

This work was partially supported by the Grant Agency of the Czech Republic under contract No. GA102/07/0850 Design and hardware implementation of a patent-invention machine, No. GD102/09/H042 Mathematical and Engineering Approaches to Developing Reliable and Secure Concurrent and Distributed Computer Systems and the Research Plan No. MSM 0021630528 Security-Oriented Research in Information Technology.

7. REFERENCES

[1] M. Bidlo and Z. Vasicek. Gate-level evolutionary development using cellular automata. In *Proc. of The 3nd NASA/ESA Conference on Adaptive Hardware and Systems, AHS 2008*, pages 11–18. IEEE Computer Society, 2008.

[2] J. F. Miller. Evolving developmental programs for adaptation, morphogenesis and self-repair. In *Advances in Artificial Life. 7th European Conference on Artificial Life, Lecture Notes in Artificial Intelligence, volume 2801*, pages 256–265, Dortmund DE, 2003. Springer.

[3] S. Nandi and P. P. Chaudhuri. Theory and applications of cellular automata for synthesis of easily testable combinational logic. In *Proceedings of the 10th Anniversary Compendium of Papers from Asian Test Symposium 1992-2001*, page 146. IEEE Computer Society, 1995.

[4] S. Kumar and P. J. Bentley (eds.). *On Growth, Form and Computers*. Elsevier Academic Press, 2003.

[5] M. Sipper. *Evolution of Parallel Cellular Machines – The Cellular Programming Approach, Lecture Notes in Computer Science, volume 1194*. Springer-Verlag, Berlin, 1997.

[6] G. Tufte and P. C. Haddow. Towards development on a silicon-based cellular computing machine. *Natural Computing*, 4(4):387–416, 2005.

[7] J. von Neumann. *The Theory of Self-Reproducing Automata*. A. W. Burks (ed.), University of Illinois Press, 1966.

[8] S. Wolfram. *A New Kind of Science*. Wolfram Media, Champaign IL, 2002.