Expert Systems with Applications 42 (2015) 7620–7626

Contents lists available at ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

Analysis of evolutionary techniques for the automated implementation of digital circuits

Mohammad Hossein Zarifi [a,*], Hajar Satvati [b], Mehdi Baradaran-nia [c]

[a] Department of Electrical and Computer Engineering, University of Alberta, Canada
[b] Department of Computer Engineering East Azerbaijan Science and Research Branch, Islamic Azad University, Tabriz, Iran
[c] Control Engineering Department, Faculty of Electrical and Computer Engineering, University of Tabriz, Tabriz, Iran

ARTICLE INFO

Article history:
Available online 11 June 2015

Keywords:
Evolvable Hardware
Logic circuit
Evolutionary algorithm
Genetic algorithm

ABSTRACT

Evolvable Hardware (EHW) is hardware that can dynamically change its behavior and architecture by interacting with its environment. EHW can make use of evolutionary algorithms (EAs) to optimally synthesize electrical circuits. In this paper, five different mutation methods are implemented in a conventional genetic algorithm technique to automatically realize a digital 8 bit full adder (FA). The main achievement of this work is the comparison between those techniques in terms of number of iterations required to converge on the required logical structure. Results demonstrate that the sequential mutation method (SMM) and circular gene method (CGM) reduce the convergence time, and a substantial reduction is observed when these methods are combined with the Adaptive Group Mutation (AGM) method for the full adder implementation.

1. Introduction

Evolvable Hardware (EHW) is a crossing point at which Artificial Intelligence, reconfigurable hardware and reliable circuits meet (Ren, Zhao, Tang, Tong, & Luo, 2011). Techniques such as evolutionary algorithms (EAs) can provide powerful tools to a piece of EHW, allowing it to optimally solve complicated problems within an acceptable time limit (Smith and Eiben, 2003). EAs are population-based optimization algorithms which utilize principles inspired from biological evolution and have been applied to a variety of applications ranging from biomedical to flight navigation (Swarnalatha & Shanthi, 2014; Uçan & Altılar, 2012; Zarifi, Frounchi, Asgarifar, & Nia; Zarifia, Ghalehjogh, & Baradaran-nia, 2015).

Evolutionary computing and algorithms have recently demonstrated a huge potential in different applications. A variable topology for evolutionary hardware design was introduced and applied designing a multiplier circuit and demonstrated less code length for evolutionary hardware description (Chen & Hwang, 2009). As another approach, genetic strategy was used to design cellular automata based block ciphers. It demonstrated a fast and secure block cipher using nonuniform second-order cellular automata. That work showed theoretically that use of non-uniform transition rules can lead to optimal performances compared to uniform ones. Also the combination of several simple rules having regular and poor characteristics can raise complex and chaotic unexpected behaviors (Faraoun, 2014). Having hybrid intelligent systems increases the efficiency of the final algorithm. The effect of hybrid intelligent systems was examined in medical data classification (Seera & Lim, 2014). The experimental outcomes from that study demonstrated that the hybrid intelligent system was successful not only in classification of medical data but also to explain its knowledge base with a decision tree.

Recently, reconfigurable hardware such as field programmable gate arrays (FPGA) has made these algorithms much more attractive by permitting a real time physical implementation of the solution circuit. Artificial evolutionary algorithms such as EA or artificial neural networks (ANN) were originally designed and envisioned to operate sequentially but they seem to have an inherent potential for parallelism. Emerging technologies in hardware implementation can allow them to run in parallel, which tremendously reduces the process time (Alizadeh, Frounchi, Baradaran Nia, Zarifi, & Asgarifar, 2008; Nia, Shogian, & Zarifi, 2008; Nedjah et al.).

Genetic algorithms (GA) produce a new system configuration in circuit design since they require a hardware structure capable of improving the functionality of the system. A GA is a common and powerful search algorithm, which also takes its inspiration from natural evolution. By using natural patterns, GAs may serve as a suitable substitute for previous types of search algorithms.

* Corresponding author.
E-mail addresses: zarifidi@ualberta.ca (M.H. Zarifi), h_satvati81@yahoo.com (H. Satvati), mehdi.baradaran@gmail.com (M. Baradaran-nia).

http://dx.doi.org/10.1016/j.eswa.2015.06.005
0957-4174/

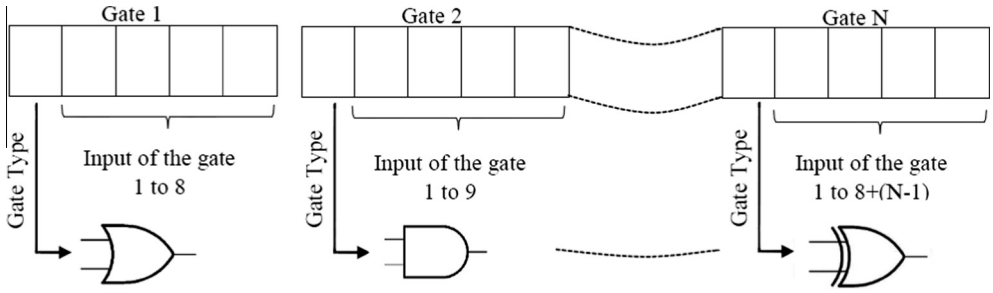

Fig. 1. The structure of a chromosome used in this work.

Table 1
Logic cell types and the assigned numbers.

Cell type gene	Cell function
−1	AND Gate
−2	OR Gate
−3	XOR Gate
−4	Not Active

Table 2
Input *s* and the related assigned numbers.

Assign Integer number	Input
1	0
2	1
3	X
4	$\overline{X}$
5	Y
6	$\overline{Y}$
7	Z
8	$\overline{Z}$

The full adder (FA) circuit is a basic circuit building block in synthetic logic circuits and serves as an excellent demonstration of the EHW's ability to synthesize simple circuit components. Developing an FA is a first step towards designing bigger and more complicated circuits.

The authors in (Soliman & Abbas, 2003) suggested an evolutionary approach to the design of multi objected circuits. Following their work, Yan et al. have put forward a new method for the design of logic circuits using a group of logic gates (Yan, Wu, & Liu, 2012). Their results show that they have been able to optimize the number of gates required in the circuit. EHW has been implemented on Field Programmable Gate Arrays (FPGA) in previous research papers (Ren et al., 2011), and the authors in (Kalganova, 2000) have presented a new scheme for cellule structure EHW. Yan et al. have demonstrated a new method to design electronic circuits utilizing some set of logic gates.

In this work the design and implementation of a digital full adder is considered as a case study for the application and comparison of multiple genetic algorithm methods. The various methods are compared in accuracy and conversion rate form factors to determine the optimal method for EHW circuit implementation. For mutation methods (see Section 2.5), this study has made use of the sequential mutation method (SMM) and circular gene method (CGM) due to their high efficiency.

2. Method and algorithm

This section describes the mechanism of the genetic algorithm in detail. In the proposed method, a single population consists of 80 chromosomes, where each chromosome has 25, 50 or 45 bits based on its type. The first population of chromosomes is randomly generated from the various types of gates and their inputs. Each chromosome in the created population is representative of a logic circuit. After generating the first population, the objective function evaluates the fitness of each chromosome. If none of the created circuits in the first generation has satisfied the aim of circuit, genetic operands are applied to the population. In the next step, the selection operation selects some pairs of chromosomes by considering their fitness value. These pairs of chromosomes serve as

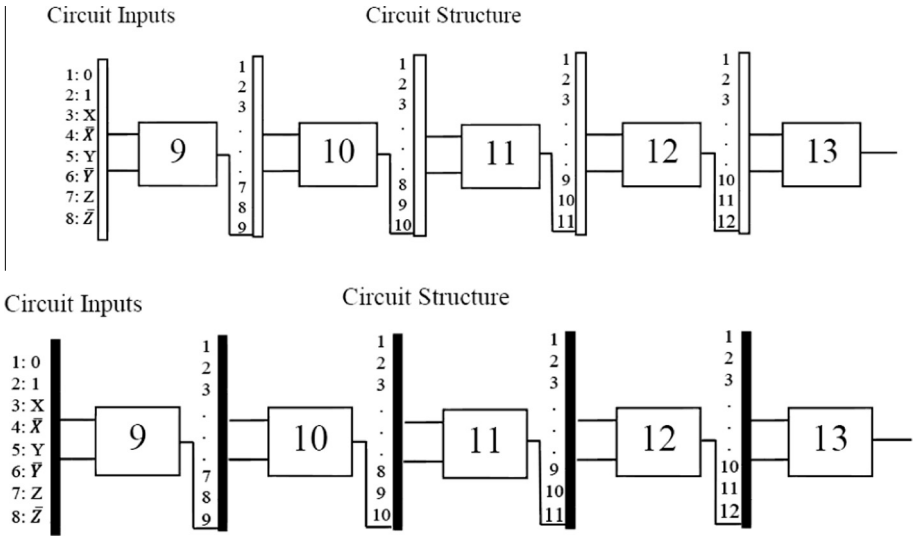

Fig. 2. First type of chromosome: a 1×5 matrix.