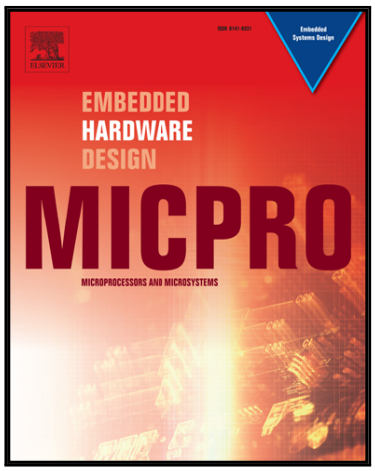

Evolutionary Design for Energy-Efficient Approximate Digital Circuits

Hamed Norouzi , Mostafa E. Salehi

PII: S0141-9331(16)30179-X
DOI: 10.1016/j.micpro.2018.01.002
Reference: MICPRO 2650

To appear in: *Microprocessors and Microsystems*

Received date: 7 October 2016
Revised date: 5 November 2017
Accepted date: 18 January 2018

Please cite this article as: Hamed Norouzi , Mostafa E. Salehi , Evolutionary Design for Energy-Efficient Approximate Digital Circuits, *Microprocessors and Microsystems* (2018), doi: 10.1016/j.micpro.2018.01.002

Evolutionary Design for Energy-Efficient Approximate Digital Circuits

Hamed Norouzi and Mostafa E. Salehi
School of Electrical and Computer Engineering
University of Tehran, Tehran, Iran, 14395-515
Email: {norouzi_hamed, mersali}@ut.ac.ir

***Abstract*—Energy and computation efficiency are of the major concerns in ever-growing embedded systems. Approximate computing as a new design methodology trades precision for energy efficiency. Evolutionary algorithms as an optimization approach would explore the possible space of the solution to find the best and efficient solutions and hence, are compatible with approximate computing objectives. This paper exploits Cartesian Genetic Programming (CGP) as a powerful design approach to bring novel and newfound approximate solutions. Our contributions are twofold: First, proposing a new simple yet effective seeding approach for CGP which decreases the evolution time and computational effort and also increases the precision of the resulted evolved circuits. Second, proposing an offline pre-evolution approach in order to reduce the complexity of design and hence, make it possible to use CGP for designing more complex problems. The results of evolving arithmetic benchmarks show improvement of the proposed seeding technique both in precision of evolved circuits and also the required computational effort. Also, exploiting the pre-evolution approach for multiplier benchmark reduce the size of truth tables over 94% and not only make it possible to use CGP to design larger multipliers, but also breaks down the power delay product (PDP) parameter more than 65% in compression with some state of the art approximate and exact multipliers.**

Keywords: Approximate Computing, Evolutionary Algorithm, Cartesian Genetic Programming, Discrete Cosine Transform

I. INTRODUCTION

Energy efficiency alongside with computational efficiency is the main concern in digital design especially in embedded systems. These concerns are becoming more and more important by ever-growing pervasive embedded systems which are battery powered and need to work for a long time. Many conventional techniques consider these parameters as a two dimensional design space and exploit energy-efficient methods to make trade-offs appropriate between performance and energy [1], [2]. By scaling the technology and shrinking the feature size, power consumption limits the available improvements of the performance [3]. However, moving towards decreasing delay, area and power consumption are essential requirements for portable embedded systems. Approximate computing is a new design approach which trades precision for energy efficiency and performance. In this approach, precision can be considered as a new dimension of the design space [1]. In fact, some applications such as sound and image processing, which are vastly used in embedded systems, interact with human beings which have limited perceptual ability to interpret the imperfections [2]. Therefore, some levels of imprecision are acceptable in these applications.

Traditional approximation methods, such as [4]–[6], try to manually eliminate some less-important parts of the precise circuit that have little effect on the results. These approaches will enable the designer to reduce the complexity and improve other parameters of the design such as power and delay. In addition to functional approximation techniques mentioned above, some levels of approximation are introduced in algorithmic and architectural levels [7], [8]. In such methods, the designer requires a deep knowledge of the design to introduce a new architecture or algorithm. These manual approaches have limited abilities to produce different circuits. Newer proposed techniques such as [9], try to develop some kinds of automated tools that are firstly flexible to generate approximate solutions for a wider range of required circuits and applications. Second, these approaches generate more efficient results to use and third, they are usable for common designers which are interested in using approximate solutions.

Considering above mentioned advantages, working on automatically designing methodologies such as evolutionary algorithms would be an attractive topic. The Evolutionary design approach which is based on probabilistic and search-based algorithms is vastly employed in automatic design of digital circuits [10]. The Evolutionary design can bring a novelty of optimized solutions which may not be reachable by conventional design methodologies. Furthermore, evolutionary algorithms have considerable compatibility with approximate computing which is looking for solutions with negligible deviations from desired answers and benefit from better area, delay and power consumptions [11]. One of the main features of the search-based algorithms is their ability to find an approximate or partially working solution in limited resources [12]. These algorithms as an optimization approach select the best answers as the next generation's parents in the evolution process hoping to find the globally optimal solutions.

However, as mentioned in [13], exploiting the evolutionary approach in design of large circuits will face the problem of scale which is known as the major problem in the field of evolutionary circuit design. Different solutions to this problem have been proposed in literature, such as [10] and [11]. These techniques which are usually based on divide and conquer or modularity aim to improve the scalability problem with some kinds of redundancy. This could consequently lead to