Evolutionary Electronics: Automatic Synthesis of Analog Circuits by GAs

Esteban Tlelo-Cuautle and Miguel A. Duarte-Villaseñor

INAOE. Luis Enrique Erro No. 1. Tonantzintla, Puebla. 72840 Mexico,
e.tlelo@ieee.org

Summary. This paper introduces guidelines for the automatic synthesis of analog circuits by performing evolutionary operations. It is shown that the synthesis of unity-gain cells (UGCs) can be done from nullor-based descriptions. In this manner, UGCs such as: the voltage follower (VF), current follower (CF), voltage mirror (VM), and current mirror (CM), are described using a new genetic representation consisting of ordered genes. Furthermore, a genetic algorithm (GA) is introduced in order to search for the best UGC by performing crossover and mutation operations, and by selecting UGCs by elitism. On the other hand, since GAs operate on the principle of survival of the fittest, the proposed GA has the capability to generate new design solutions, i.e. new UGCs. Additionally, the synthesized UGCs are evolved to design more complex circuits, namely: current conveyors (CCs), inverting CCs (ICCs), and current-feedback amplifiers (CFOAs). Finally, some analog circuit evolution approaches are described to synthesize practical applications such as: active filters, single-resistance controlled oscillators (SRCOs), and chaotic oscillators, which are implemented using UGCs, CCs, and CFOAs.

1 Introduction

The proper aspects of the life, as the self-organization, adaptation and evolution, are giving place to a revolution in various engineering areas. Artificial life (Alife) is the name given to this new discipline which studies natural life to recreate biological phenomena by using computers and other artificial systems. The main goal is to apply biological principles to hardware and software technology, medicine, nanotechnology, and multiple engineering projects such as evolutionary electronics [1–3], where Alife is giving place to an authentic heuristic revolution on the automatic design of electronic circuits [4–12].

Since the world is fundamentally analog in nature, all the electronic systems that interface with the external world require analog circuits, such as: cellular telephones, magnetic disk drives and speech recognition systems [4]. Even though analog design automation (ADA) began four decades ago it is
still in its infancy compared to the digital domain [5]. That way, new techniques are required to improve the ADA process in order to shorten the time to market [6], to enhance the quality and optimality of integrated circuits (ICs) designs, and to reduce production costs through increased manufacturability.

Analog design is much amenable for evolutionary techniques, where contrasting with digital design, there is no solid set of design rules or procedures to automate circuit synthesis [3,7]. For instance, in [1,3,7–12], have been introduced synthesis methodologies based on genetic algorithms (GAs). Since GAs operate on the principle of “survival of the fittest” [1], they have the capability to generate new design solutions from a population of existing solutions, and discarding the solutions which have an inferior performance or fitness. GAs begin with an initial collection of random solutions called initial population. Each individual in the population is called chromosome and represents a possible solution to the problem. A chromosome is a chain of symbols called genes, which generally are represented by binary strings. The chromosome evolves through iterations called generations. In each generation the chromosomes are evaluated using an aptitude measure. The next population is formed by descendents created by combining two chromosomes of the current generation using the crossover and the mutation operators. Henceforth, this chapter introduces guidelines to automate the synthesis of practical analog ICs by applying GAs, and by using the nullor element [13] to codify the abstract behavior of analog circuits.

2 Analog Design Automation

In electronics, complete systems that occupied one or more boards are now integrated on a few chips or even on one chip [4,5]. This has put significant pressure on the development of synthesis methodologies to enhance ADA. Furthermore, this section describes the usefulness of intelligent systems for the development of new synthesis tools to design practical analog ICs.

2.1 Circuit Synthesis

The goal of analog circuit synthesis is devoted to make the transition from research to practice. This process seeks to transform abstract descriptions into a working circuit. On this direction, a complex design can be carried-out by partitioning or modularizing the design into blocks or cells that can be shared and reassembled in a manageable and repeatable form [6–12]. To do that, an ADA developer should to perform the ability to view the design into three levels: system-level, circuit-level and layout synthesis [6]. Elsewhere, analog synthesis can be performed in a top-down approach by beginning with module descriptions, and further by performing a refinement process until obtaining a practical analog IC.