Applied Soft Computing 55 (2017) 436–451

Contents lists available at ScienceDirect

Applied Soft Computing

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

Simplified symbolic transfer function factorization using combined artificial bee colony and simulated annealing

Mohammad Shokouhifar*, Ali Jalali

Department of Electrical Engineering, Shahid Beheshti University G.C., Tehran, Iran

ARTICLE INFO

Article history:
Received 6 November 2016
Received in revised form 21 February 2017
Accepted 26 February 2017
Available online 1 March 2017

Keywords:
Symbolic simplification
Transfer function
Factorization
Artificial bee colony
Simulated annealing

ABSTRACT

Symbolic circuit analysis inherits the exponential growth of transfer function complexity with the circuit size. Therefore, symbolic simplification is an NP-hard problem. Although many simplification techniques have been presented, the simplified transfer functions are not written in a factorized form, and consequently, it is difficult to assess the contribution of poles and zeros on the circuit behavior. In this paper, a swarm intelligence based methodology is presented for the simplified factorized symbolic analysis of analog circuits. In this method, an extension of the root splitting technique is utilized to rewrite the expanded transfer function of the circuit into a factorized form comprising DC-gain, poles, and zeros. Then, the derived factorized transfer function is simplified using a hybrid Global and Local search algorithm based on Artificial Bee Colony and Simulated Annealing (named GLABCSA). The objective function is defined to minimize the complexity of the symbolic factorized transfer function while minimizing the DC-gain error and pole/zero displacements. The presented approach has been successfully developed in MATLAB. The program can derive the simplified factorized symbolic transfer function automatically from the input text netlist of the circuit. Symbolic and numerical results over two analog amplifiers are given to illustrate the efficiency of the presented methodology.

1. Introduction

Recently, multi-stage cascade amplifiers have become increasingly exploited, as they provide high gain and large output swing with low overdrive voltages [1–6]. Unfortunately, as gain stages are cascaded, bandwidth is progressively reduced, because each additional stage inevitably have own poles and zeroes, and more importantly, requires additional compensation capacitors to avoid compromising stability. Therefore, a frequency compensation procedure should be designed enabling high gain and wide bandwidth with adequate stability margin. To achieve this purpose, symbolic pole/zero analysis techniques can be employed to assist the designer to make straight decisions during the frequency compensation procedure [7,8].

Circuit modeling is of major importance in the symbolic analysis. Different ways are used for the modeling of different circuits. For instance, active devices can be modeled with controlled sources or pathological elements (e.g., nullor equivalents) [9,10]. Moreover, semiconductor devices (e.g., BJT and MOSFETs) are substituted by their small-signal models [11]. Non-ideal effects of transistors (e.g., parasitic port resistances and parasitic capacitors) can be considered in the small-signal model according to the application specifications [12].

The main problem in the practical use of symbolic tools is the difficulties to handle large symbolic formula. Although many approximation techniques have been proposed in the literature, the simplified transfer function is not written in a factorized form, and consequently, it is difficult to assess the contribution of poles and zeros. The existing pole/zero extraction techniques exhibit some drawbacks which limit their usage for practical analog circuits: First, pole/zero displacements are not effectively taken into account during the pole/zero extraction process. Second, in the root splitting method, the condition for which the expanded transfer function can be factorized, is based on the relative magnitudes of the polynomials, and the pole/zero errors are not under control. Third, the resultant pole/zero expressions are expected to be not so compact as it could be, because no simplification-after-generation is performed. Fourth, pole/zero displacements are not under consideration, and consequently, significant pole/zero errors may be generated in the simplified expressions.

In this paper, a combined swarm intelligence algorithm based on artificial bee colony (ABC) and simulated annealing (SA) is presented for simplified factorization of symbolic transfer functions. The main contributions in this paper can be summarized as follows:

* Corresponding author.
E-mail addresses: m_shokouhifar@sbu.ac.ir, shokoohi24@gmail.com (M. Shokouhifar), a_jalali@sbu.ac.ir (A. Jalali).

http://dx.doi.org/10.1016/j.asoc.2017.02.029
1568-4946/

A simplified symbolic factorization methodology is presented for analog amplifiers.

- To the best of our knowledge, this is the first algorithm specifically designed for simplifying the symbolic pole/zero factorized transfer functions.
- The traditional root splitting technique is extended to efficiently factorize the expanded transfer function.
- The factorization condition of the root splitting method is modified, in which, the pole/zero errors due to the factorization are taken into account.
- A hybrid Global and Local search algorithm based on ABC and SA (named GLABCSA) is introduced for the simplification of factorized transfer function. The motivation is to gain the advantages of ABC and SA, i.e., good global exploration mechanism of ABC and good local search ability of SA.
- The proposed method guarantees that the maximum pole/zero displacements in both factorization and simplification procedures don't exceed from user-specified thresholds.

The rest of the paper is organized as follows: In Section 2, the existing pole/zero extraction and simplification techniques are discussed. Section 3 provides an overview of the proposed methodology. In Section 4, the factorization method via the extended root splitting technique is presented. Section 5 introduces the GLABCSA simplification algorithm. In Section 6, the developed approach is applied to extract the simplified factorized transfer function of two analog amplifiers. Finally, conclusion remarks can be seen in Section 7.

2. Related works

Generally, symbolic analysis methods can be classified according to the basic mechanism used in the analysis engine, into graph-based and matrix-based approaches [13]. Up-to-date symbolic programs work, with only minor exceptions, on the basis of matrix-based nodal analysis (NA) and its extensions, e.g., modified NA (MNA) [14] and reduced MNA (RMNA) [15]. These techniques can also be extended to VLSI networks [16,17]. Comprehensive reviews on symbolic analysis techniques can be seen in [13,18].

2.1. Symbolic simplification techniques

The main problem in the practical use of symbolic tools is the difficulties to handle large formula in terms of computational complexity, memory, and CPU time [19]. Experiments show that a few terms contain the majority of relevant information of the circuit behavior [20,21]. According to the step in which simplification is performed, three types of techniques can be distinguished: simplification-after-generation (SAG), simplification-during-generation (SDG), and simplification-before-generation (SBG) [22]. The SAGs are applied once the symbolic analysis has been performed and the exact transfer function has been generated [23].

2.1.1. Classical approaches

Let H_E represent the small-signal transfer function of the circuit in the expanded form, which has been generated either by matrix-based or graph-based analysis methods. It can be expressed as

$$H_E(s) = \frac{N(s)}{D(s)} = \frac{f'_0 + f'_1 s + f'_2 s^2 + \ldots + f'_{n'} s^{n'}}{f_0 + f_1 s + f_2 s^2 + \ldots + f_n s^n} \quad , \tag{1}$$

where numerator N and denominator D are functions of complex frequency s and circuit parameters x. Each polynomial f'_j or f_i is a sum-of-products of the set of symbolic parameters x, and can be expressed as $h_k = h_{k1} + h_{k2} + \ldots + h_{kT}$, where h_{kl} is the l-th symbolic term of the k-th polynomial of the transfer function, h_k, which has a total of T terms.

There are four classical SAG criteria, which were implemented in almost all symbolic tools. In these criteria, the different polynomials are simplified separately. In the first criterion [24], the term h_{kl} can be discarded from the polynomial h_k, if $|h_{kl}(x)| \leq \varepsilon \times max(|h_{k1}(x)|, |h_{k2}(x)|, \ldots, |h_{kT}(x)|)$, where ε $(0 < \varepsilon < 1)$ is the maximum allowable error for each polynomial. The drawback is that accumulated error is not under control. Three criteria were introduced to overcome this drawback. In the second criterion [25], the condition on P candidate terms from polynomial h_k, for which they can be eliminated, is $|\sum_{l=1}^{P} h_{kl}(x)| < \varepsilon \times |\sum_{l=1}^{T} h_{kl}(x)|$. The criteria 3 and 4 [12] can be formulated as $\sum_{l=1}^{P} |h_{kl}(x)| < \varepsilon \times |\sum_{l=1}^{T} h_{kl}(x)|$ and $\sum_{l=1}^{P} |h_{kl}(x)| < \varepsilon \times \sum_{l=1}^{T} |h_{kl}(x)|$, respectively. Although the four classical criteria are simple and easy to implement, they simplify different polynomials separately. If the same error ε_M occurs for all polynomials, there is no magnitude/phase error. However, because of the discrete nature of simplification, the actual errors of polynomials are different. Different solutions were reported to overcome this problem. In [25], an adaptive ε, in which, the term-pruning is done by increasing the value of ε step by step, and monitoring the pole/zero displacements at the each step. The simplification can be stopped when such displacements are beyond a user-specified margin.

2.1.2. Metaheuristic based approaches

Symbolic simplification is an NP-hard (non-deterministic polynomial-time hard) problem [23]. The size of search space can be calculated as $\sum_{l=1}^{L} \binom{L}{l} = \binom{L}{1} + \binom{L}{2} + \ldots + \binom{L}{L} = 2^L - 1 \approx 2^L$, where L and l are the total number of terms and the number of selected terms, respectively. Generally, in metaheuristic-based simplification techniques, a feasible solution is represented as a binary string. The k-th variable is "1", if the corresponding element or term is presented in the solution, and it is "0", for the otherwise.

A technique based on genetic algorithm (GA) was presented in [26] to simplify the equivalent small-signal circuits. The objective is to minimize the magnitude error within a frequency interval. The main drawback is that the phase of the transfer function is not under consideration. Moreover, the results are expected to be complex, as no SAG is performed to clean-up the final symbolic expression.

An ant colony optimization (ACO) has been proposed [23], in which, the simplification problem space is represented as a routing graph with L nodes, wherein L is the number of all symbolic terms. To construct a feasible solution, each ant selects some terms according to the probability rule of ACO. The drawback is that only gain and phase margin errors are considered to evaluate the simplified expressions represented by ants, where the circuit may does not have a certain gain or phase margin.

Deviation in pole/zero locations is not efficiently taken into account in the above methods. Two GA-based technique was presented in [27,28] to overcome this drawback. The objective is to minimize the generated error in terms of some characteristics, e.g., magnitude/phase error and pole/zero displacement. However, the worst case error in pole/zero displacements is not considered in [27,28]. This problem has been addressed and solved in [21,29,30]. Although the metaheuristic techniques outperform the conventional classical criteria in term of accuracy, these methods are only