Swarm and Evolutionary Computation 20 (2015) 37–47

Contents lists available at ScienceDirect

Swarm and Evolutionary Computation

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

A novel framework for retiming using evolutionary computation for high level synthesis of digital filters

Deepa Yagain *, A. Vijayakrishna

Department of ECE, PESIT, Bangalore, India

ARTICLE INFO

Article history:
Received 15 July 2014
Received in revised form
22 October 2014
Accepted 22 October 2014
Available online 12 November 2014

Keywords:
Solution space
Tournament selection
Digital filter optimization
Retiming
Data flow graphs
Non-dominated Sorting Genetic Algorithm (NSGA)

ABSTRACT

In this paper, design of a new algorithm and a framework for retiming the DSP blocks based on evolutionary computation process is explained. Optimal DSP blocks such as digital filter design is a high level synthesis problem which includes optimally mapping digital filter specifications on to FPGA (Field Programmable Gate Array) architecture. Retiming is the considered optimization method in this paper which gives optimality in terms of algorithm processing speed and digital filter operating frequency with register count as a constraint. The designed novel algorithm is for the synthesis of high speed digital filters for different signal processing applications based on nature inspired evolutionary computation method. The classical retiming algorithms such as clock period minimization and register minimization that are addressed in the literature provide a single heuristic solution based on the chosen optimization parameter such as clock period. However, for retiming which is multi-objective optimization, evolutionary approach can lead to better results. Using the designed evolutionary computation based retiming method, retimed solution database is generated with higher frequency and different output register counts by searching the digital block solution space. Depending on the clock period and register count constraint, designer can take a design decision. Here, various signal processing designs are used to facilitate the design analysis. Results also show that the CPU processing time needed to compute multiple solutions using the designed algorithm for filter circuits is reduced for designs whose maximum feasible solutions are less than 50. If the circuit is very big with the possible solution space greater than 50 solutions, then algorithm performs slower. A comparison is also provided in the Simulations section with respect to all the existing classical retiming methods in the literature such as clock period and register minimization retiming to prove the concept. Multi-objective genetic algorithms are the considered evolutionary computation method in this paper.

1. Introduction

High level synthesis (HLS) is an automated design process that interprets the algorithmic description of circuit behaviour and generates hardware implementation. In retiming, any complex function can be divided into independent sub-functions. This decreases the overall time by factor m if complex function is divided using $(m-1)$ flip-flops. This process is used to obtain synchronous circuits. Retiming technique is used to increase the speed of the synchronous circuit without changing the functionality and latency [1,2]. Here relative location of the registers is changed such that clock period is minimized. This improves the synchronous speed and preserves the circuit latency and functionality. In the literature, many polynomial time algorithms to compute retimed circuits have been discussed [1,2].

A retiming approach can also be generalized to find those non-critical gates which can be operated with lower supply voltages to reduce the overall system power consumption. In the literature multiple retiming methods are available for optimizing single objective such as clock period [3]. After retiming, the clock period gets minimized without altering the functionality. Retiming method might increase the number of registers slightly [4]. The solution which is retimed for clock period minimization can be an input to register minimization retiming [5] algorithm which minimizes the registers to some extent. However, the considered algorithms will give one optimum solution in terms of clock period. This paper focuses on designing an algorithm for retiming based on evolutionary computation which generates database of solutions with different clock periods and register counts instead of one. A model is developed to exploit the multi-objective optimization in retiming. The objectives considered in this paper are area and speed performance. For a considered filter circuit, solution is developed to map the filter architectures under considered objectives. Here, an environment is

* Corresponding author.

http://dx.doi.org/10.1016/j.swevo.2014.10.004
2210-6502/

designed for performing retiming of digital filters using evolutionary computation. This explores the solution space of entire digital filter design and gives out the best possible solution which has lower sample periods with the same functionality. However, the register count may vary slightly after retiming.

Evolutionary computation process [6] is an adaptive optimization technique. From the literature, it is evident that practical engineering problems can be solved efficiently with this nature inspired evolutionary algorithms [3,4]. Here, potential problem solutions are chromosomes which have set of genes which are nothing but characteristics of the solution. Evolution of next generation happens by mutation and selection. Evolutionary computation method requires only a suitable coding structure which can modify the solutions during reproduction and it also needs a system for assessing the quality of individual solutions. This method does not need any prior knowledge of solution search space [7]. Hence this approach is much advantageous.

The designed environment also provides multiple solutions with various register counts to an optimization problem of digital filter using retiming. Environment can also pick the best possible solution if needed by the user. This evolutionary computation based retiming algorithm converges much faster and obtains the results much quicker for smaller designs when compared to the existing methods. However, as the design size increases, algorithm becomes smaller. This optimization environment can be integrated as a part of high level synthesis flow. The genetic algorithms for multi-objective optimization are used for algorithm design. In particular, NSGA (Non-dominated Sorting Genetic Algorithm) is used to address the retiming for digital filters. NSGA is an extension of genetic algorithms where changes are made only for selection operator [8]. Various versions of this algorithm are also presented in [9–11]. The basic principle of non-dominated sorting remains the same. The computational complexity of this method is $O(mN^2)$ where m is the number of objectives and N is the size of the population.

The complexity increases as the population size increases. In this method, population is initialized and population is classified based on the identification of non-dominated individuals. A dummy fitness is initially assigned. This population is later shared in multiple generations and reproduction of population occurs as per the dummy fitness function. Once the population has been sorted, crossover and mutation is performed and the process repeated till the exit criteria are satisfied. In this algorithm, retaining the best individuals in a generation unchanged in the next generation, is called elitism or elitist selection, is used along with sharing of non-dominated set. It is a successful variant of the general process of constructing a new population. This algorithm is also referred as modified version of NSGA or NSGA-II. Hence NSGA can be used for multi-objective optimization to obtain multiple Pareto-optimal solutions. With all the considered advantages, it motivates the authors to build on algorithm for retiming using evolutionary computation approach.

2. Background

This section presents the main concepts related to the algorithm proposed in this paper, gives the problem definitions, and an overview on algorithms that are previously proposed in the literature.

2.1. Data flow graphs (DFGs)

DFGs are powerful representation of digital filters which are a part of signal processing algorithms [12]. They represent the entire operations using nodes which are finite in number. DFGs are made of nodes and arcs. Nodes in the DFGs represent functions multiplication or addition in filters. They fire after input is available. Filters consume and process streams of input data and produce output data. Arcs represent the directed signals from one node to another. The DFGs represent the data dependencies. Synchronous DFGs are the subset of DFGs which is the synchronous nodes network. Most of the Digital Signal Processing (DSP) applications can be modelled as synchronous DFGs. A synchronous DFG is connected, non-terminating structure [13,14]. They can capture the data driven feature of DSP blocks such as digital filters. They can be represented as

$$G = (V, E, d, w), \tag{1}$$

where V represent nodes such as adders or multiplier elements in digital filters, E is the set of directed edges that represent communication between the nodes, d is the computation time of the node and w is the weight in terms of delays on the edge. The DFG is constructed from the filter block diagram based on the design under consideration. The block diagram of 6th order low pass FIR filter is as shown in Fig. 1(a). The specifications of the considered filter are the following:

- Minimum stop band attenuation = 30 dB.
- Transition region −20 to 55 kHz.

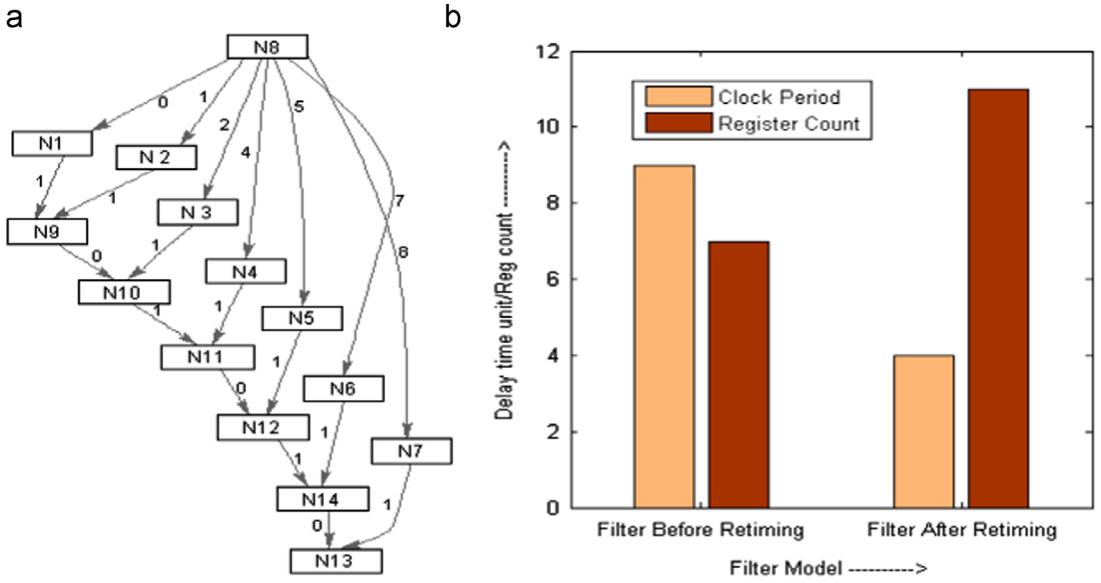

Fig. 1. 6th order Low pass FIR filter after clock period minimization retiming: (a) DFG after retiming, and (b) clock period and register count before and after retiming.