Evolutionary Synthesis Algorithm - Genetic Operators Tuning

GREGOR PAPA, JURIJ ŠILC
Computer Systems Department
"Jožef Stefan" Institute
Jamova c. 39, SI-1000 Ljubljana
SLOVENIA

Abstract: - This paper presents the evaluation and fine-tuning of different values of genetic operators in the process of optimizing the designs of the integrated circuits. Due to the increasing usage of the evolutionary optimization in the area of the integrated circuit design, there is a need to find a proper combination of genetic operators parameters' value. We investigated the interdependence of various values of these parameters and their influence on the quality of the final solution. Generally, the quality of solution is influenced by parameters and the input design. Therefore, it is important to perform this kind of evaluation each time we are searching the optimal values of the genetic operators for some new problem to be solved.

Key-Words: - evolutionary, scheduling, allocation, genetic operators, tuning

1 Introduction

The area of evolutionary computation is very popular but there is always a problem of defining a proper value of parameters of genetic operators. A standard genetic algorithm uses four different parameters that have to be defined in advance, before the algorithm is actually used. These are: the number of generations, the size of the population, the probability of crossover, and the probability of mutation [1, 5].

There are some proposals for setting of these parameters according to the problem size and according to the area of the problem. But these proposals are not always applicable or are not suitable for all problems.

To find some dependencies between the parameters and the problem that has to be solved, we made the evaluation. We study an evolutionary approach that automatically generates circuit designs. We managed to point to some interesting dependencies between parameters themselves and to determine what values should be used in our optimizations when working with evolutionary-oriented algorithms.

The organization of the paper is as follows: Section 2 presents some details of test environment and evolutionary allocation-based scheduling algorithm; Section 3 describes the testbench; while Section 4 presents the results of evaluation.

2 ABS algorithm

To find the optimal values of the parameters of the genetic operators we used Allocation-based Scheduling (ABS) algorithm [5]. The ABS is evolutionary-oriented scheduling algorithm for use in high-level synthesis of the integrated circuit. The algorithm is capable of producing an optimal solution according to scheduled and allocated operations.

To be able to produce the optimal solution it uses the following cost function

$$Cost = \sqrt[2]{\begin{array}{l}\sum_{i=1}^{n}\left(w_{FU_i}\cdot FU_i\right)^2 + \\ +\left(w_{Reg}\cdot Reg\right)^2 + \\ +\left(w_{Bus}\cdot Bus\right)^2 + \\ +\left(w_T\cdot T\right)^2\end{array}} \qquad \text{(Eq. 1)}$$

which ensures the proper estimation and evaluation of the optimality. Here, FU_i is the largest number of a unit of type i, used in any of the control steps, Reg is the largest number of the variables, needed in any control step, Bus is the largest number of transitions (inputs/outputs into/from functional units) in any control step, and T is time needed to finish all scheduled operations.

According to different kinds of multi-objective cost functions and their effectiveness [2], the distance function in multi-dimensional space was used, where each coordinate presents one of the criterions. These criterions are weighted (w_{FU1}, ... w_{FUn}, w_{Reg}, w_{Bus}, w_T) to ensure compatibility of parameters.

The goal of the ABS algorithm is to optimize the design of the integrated circuit. Through the processes of scheduling and allocation, the circuit could be implemented with the optimal number of all resources (functional, storage and connection units).

3 Elliptic filter

In the evaluation process of the ABS algorithm the fifth-order elliptic filter was used [3], which is well known benchmark in the area of the automatic circuit design.

Data-flow graph (DFG) of the elliptic filter consists of 34 operations with only two operation types (adders and multipliers). This ensures the possibility of scheduling with different number of functional units and other resources. The DFG of the elliptic filter is presented in Fig. 1.

4 Evaluation

Considering 3125 different schedules of the elliptic filter with the ABS algorithm and 625 different combinations of parameters, we compared the results according to their cost function (Eq. 1). The weights used by the algorithm were:

- $w_{FU1} = 6.3$;
- $w_{FU2} = 7.1$;
- $w_{Reg} = 2.9$;
- $w_{Bus} = 1.1$ and
- $w_T = 50$.

To ensure most solutions being time-constrained (executed in shortest possible time) the weight w_T was set to extremely high value.

As presented in Fig. 2 and Tab. 1, we can see that solutions with high quality are mostly obtained by the following values of parameters:

- number of generations is 100;
- population size is 10;
- probability of crossover is 0.5 and
- probability of mutation is 0.01.

The values of parameters in this combination are named as optimal values.

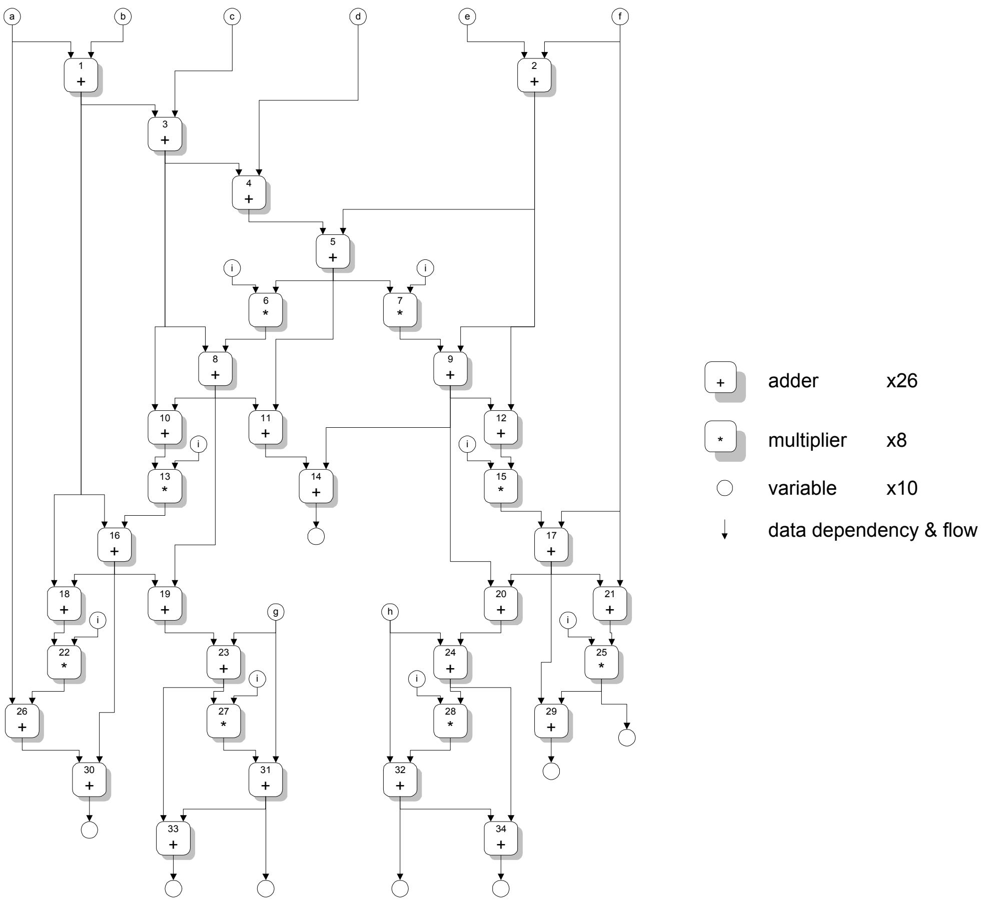

Fig. 1. Fifth-order elliptic filter DFG

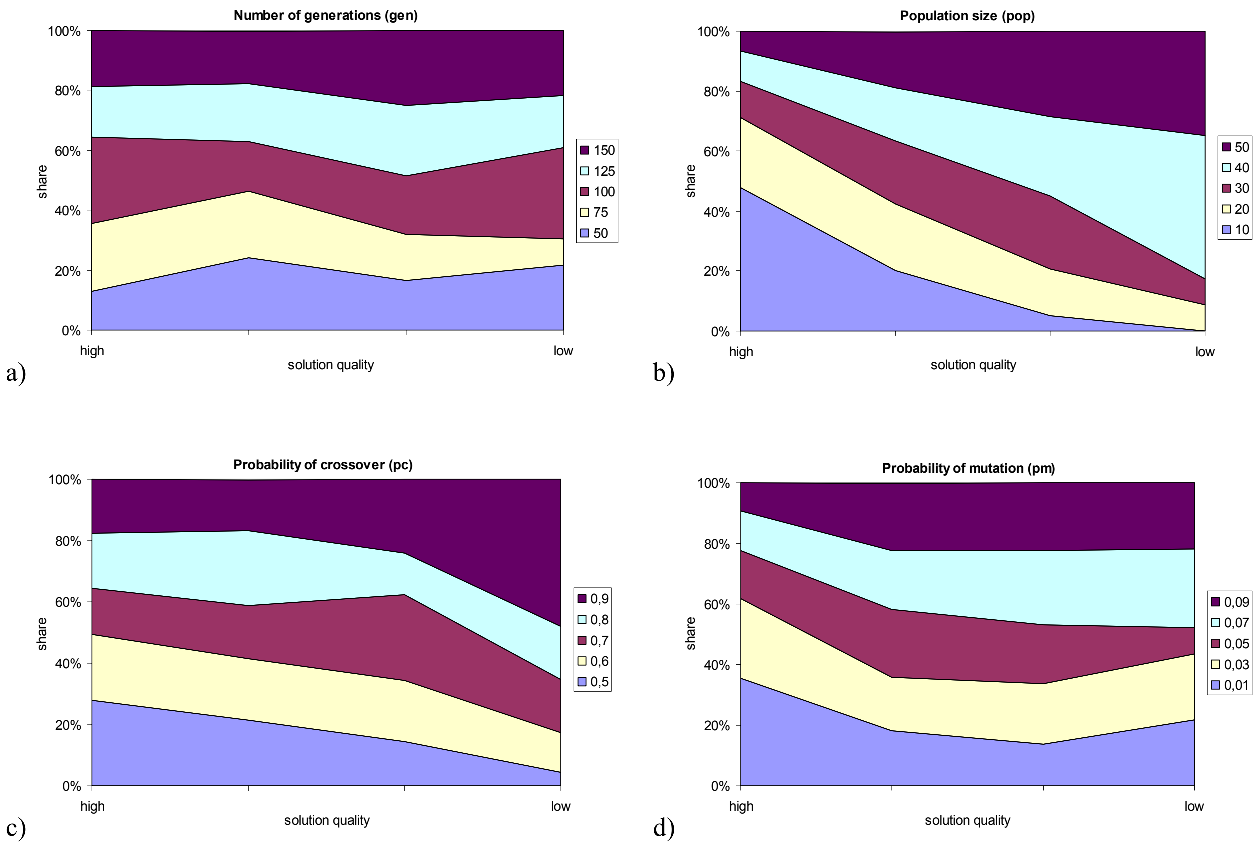

Fig. 2. The share of different parameters' values according to the influence on qualities of solutions: a) number of generations, b) population size, c) probability of crossover, d) probability of mutation

a)

quality	number of generations				
	50	75	100	125	150
high	13.1	22.4	**29.0**	16.8	18.7
good	**24.0**	22.4	16.5	19.3	17.4
bad	16.6	15.4	19.4	23.4	**25.1**
low	21.7	8.7	**30.4**	17.4	21.7

b)

quality	population size				
	10	20	30	40	50
high	**47.7**	23.4	12.1	10.3	6.5
good	20.2	**22.1**	20.9	17.8	18.7
bad	5.1	15.4	24.6	26.3	**28.6**
low	0.0	8.7	8.7	**47.8**	34.8

c)

quality	probability of crossover				
	0.5	0.6	0.7	0.8	0.9
high	**28.0**	21.5	15.0	17.8	17.8
good	21.5	19.9	17.4	**24.3**	16.5
bad	14.3	20.0	**28.0**	13.7	24.0
low	4.3	13.0	17.4	17.4	**47.8**

d)

quality	probability of mutation				
	0.01	0.03	0.05	0.07	0.09
high	**35.5**	26.2	15.9	13.1	9.3
good	18.1	17.8	**22.4**	19.3	22.1
bad	13.7	20.0	19.4	**24.6**	22.3
low	21.7	21.7	8.7	**26.1**	21.7

Table 1.
The share (in %) of parameters' influence on the quality of solutions:
a) number of generations, b) population size, c) probability of crossover, d) probability of mutation

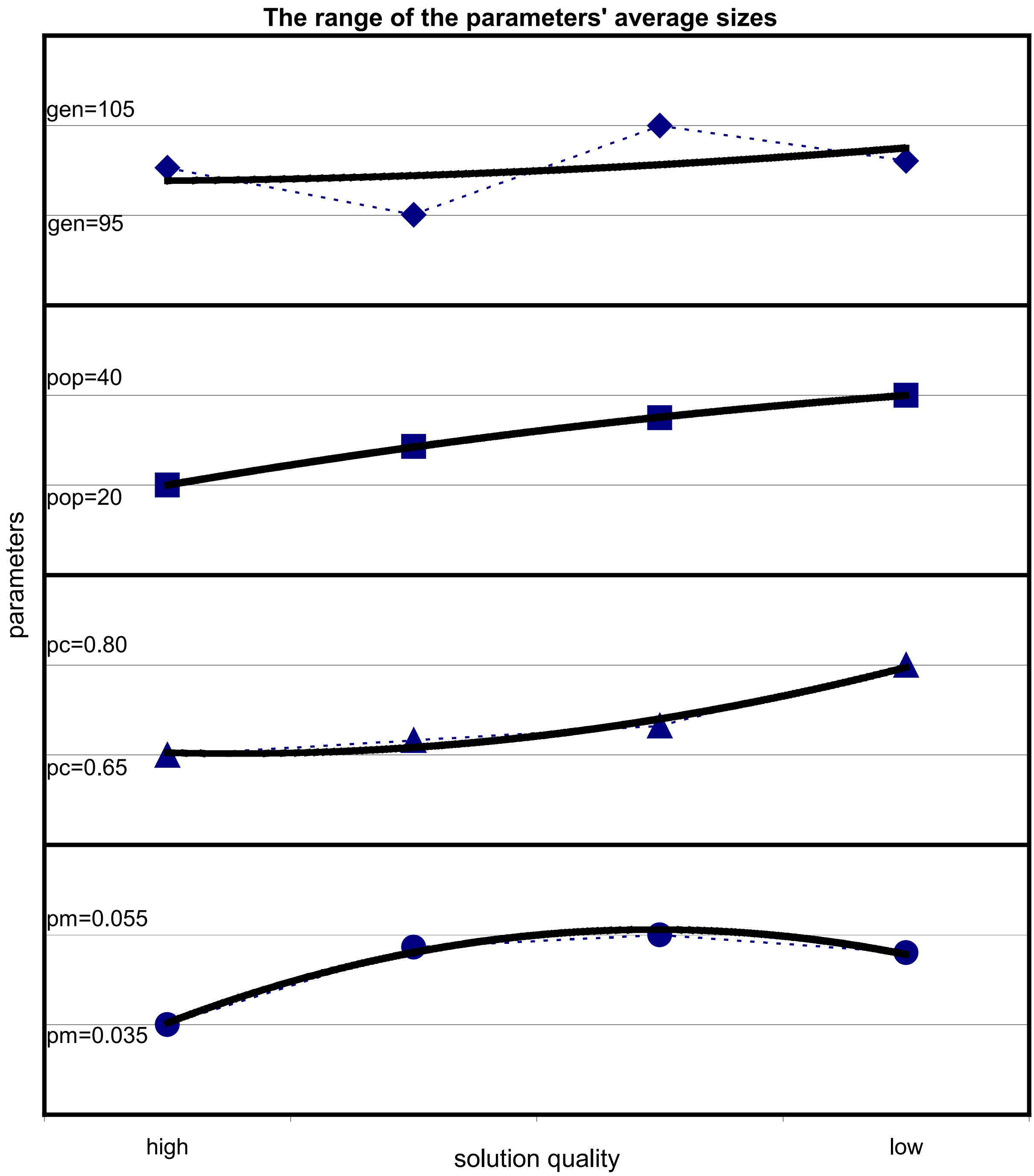

Fig. 3. The range and average value of parameters according to different qualities of solution

Fig. 3 presents the range of the average values of parameters according to different qualities of solutions. It can be seen that on average number of generations does not influence the quality significantly since there is always pretty similar quality, only the schedule time increases with the increasing number of generations. Next, smaller average population sizes (around 20) give much better solutions than bigger sizes. Again there is also the increase of schedule time according to increasing population size.

On the other side, there are two parameters that influence the variability of solutions. As presented, better solutions can be obtained when probability of crossover is 0.65 and probability of mutation is 0.035.

In Fig. 4 generations and population size are fixed to their optimal value and the interdependence of probabilities is checked, while in Fig. 5 the probabilities are fixed and the interdependence of generations and population size is revealed.

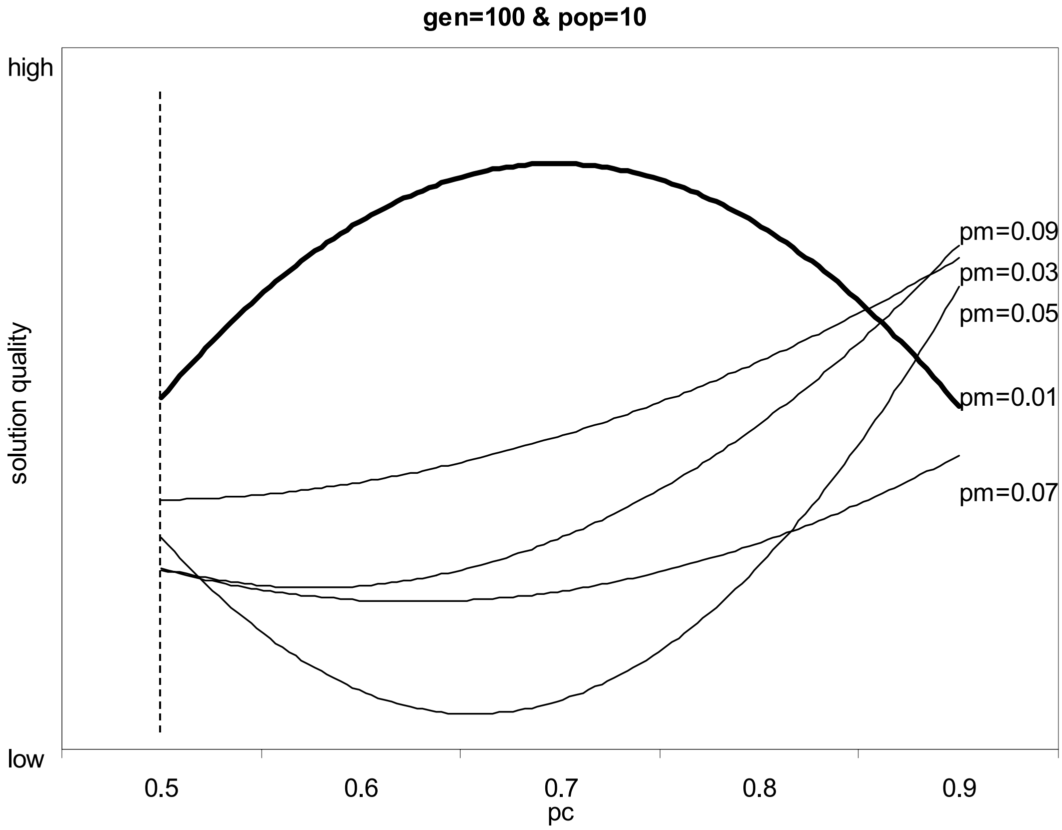

Fig. 4. Interdependence between crossover and mutation probabilities.

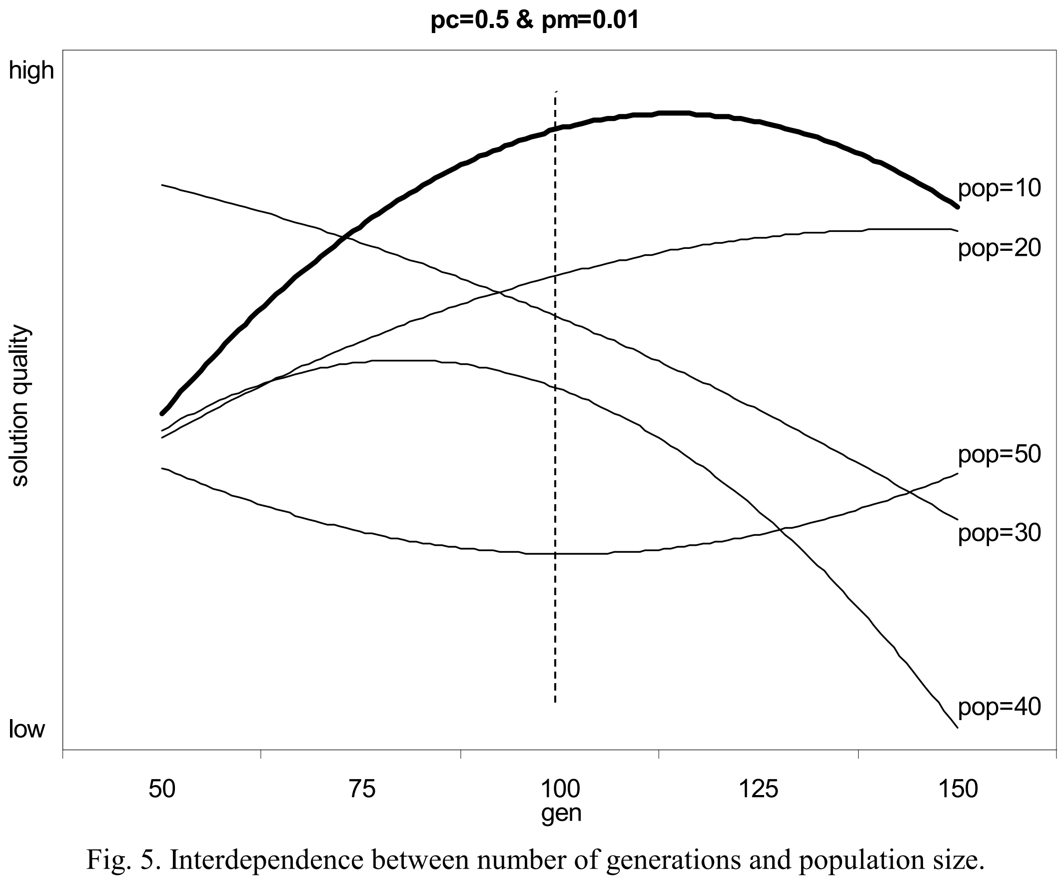

Fig. 5. Interdependence between number of generations and population size.

The cost of solution obtained by optimal values is not the best in the range of over 3000 solutions but is in top 10% of combinations ranked according to the cost function value. Despite the difference between the optimal-values result and the best results (as seen in Fig. 4 and Fig. 5) there is a difference in costs less than 2% of the cost function value.

5 Conclusion

As presented there is a lot of work to fine-tune the proper values of the genetic operators. To achieve compatible results in optimization of the fifth-order elliptic filter design it is appropriate to use the values obtained by our investigation.

Generally, the quality of solution is always influenced by parameters and the input DFG. Therefore, it is important to perform this kind of evaluation each time we are in search of the optimal values of the genetic operators for some new problem to be solved.

References:

[1] T. Bäck, *Evolutionary Algorithms in Theory and Practice: evolution strategies, evolutionary programming, genetic algorithms*, Oxford University Press, New York, 1996.

[2] C.A. Coello Coello, A Comprehensive Survey of Evolutionary-Based Multiobjective Optimization Techniques, *Knowledge and Information Systems*, Vol. 1, No. 3, 1999, pp. 269-308.

[3] S.Y. Kung, H.J. Whitehouse, T. Kailath, *VLSI and Modern Signal Processing*, Prentice-Hall, Englewood Cliffs, NJ, 1985.

[4] G. Papa, J. Šilc, Multi-objective genetic scheduling algorithm with respect to allocatin in high-level synthesis, *Proc. 26th Euromicro Conf. Digital Systems Design Symposium*, Maastricht, The Netherlands, September 2000, pp. 339-346.

[5] G. Papa, J. Šilc, Scheduling algorithms based on genetic approach, *Proc. 4th Conference on Neural networks and their applications*, Zakopane, Poland, May 1999, pp. 469-474.