FPGA Segmented Channel Routing Using Genetic Algorithms

Lipo Wang [1,2], Lei Zhou [2], and Wen Liu [2]

[1] College of Information Engineering, Xiangtan University, Xiangtan, Hunan, China

[2] School of Electrical and Electronic Engineering, Nanyang Technological University, Block S1, 50 Nanyang Avenue, Singapore 639798
elpwang@ntu.edu.sg

Abstract. A genetic algorithm approach for segmented channel routing in field programmable gate arrays (FPGA's) is presented in this paper. The FPGA segmented channel routing problem (FSCRP) is formulated as a special case of a matrix row matching problem which is known to be NP-complete. The goal of FSCRPS is to find a conflict-free net assignment in the tracks within the channel with the minimum routing cost. Simulations on 30 benchmark instances show that GA is able to obtain better solutions compared to the gradual neural network (GNN) approach.

1 Introduction

A Field Programmable Gate Array (FPGA) is a general-purpose, multi-level programmable logic device that is customized by the end users. FPGAs combine the benefits of custom VLSI with the advantages of standard logic IC's. The FPGAs have become a new approach to ASIC design, which can dramatically reduce manufacturing turn-around time and cost for low volume manufacturing.

FPGAs exist in many forms. Row-based architecture is one of the major FPGA architectures, which has been very promising and thus well-studied [1]-[12]. The row-based FPGA architecture is similar to the traditional mask-programmable gate arrays, where rows of logic cells are separated by routing channels. Each inputs and output of the cells are connected to a dedicated vertical line called vertical segment along a column of the channel. The horizontal line laid out along a row or track of the channel is usually divided or segmented into several intervals called horizontal segments by programmable switches. Programmable switches are located at the crossing points of vertical and horizontal segments, called cross-fuses, and also between two adjacent horizontal segments in the same track, called anti-fuses. Each switch can provide a low resistance bidirectional interconnection between the two crossings or adjoining segments as required. A net or interconnection between logic cell pins and/or external I/O pins can be realized by combining vertical and horizontal signal lines using programmable switches in the routing channel [1, 5].

In a segmented FPGA routing channel, the tracks are divided into segments of varying lengths. It allows each net to be routed using a single segment of the appropriate size in the same track. Greater routing flexibility is obtained by joining limited numbers of adjacent segments in the same track end-to-end using programmable switches. It has been demonstrated that a well-designed segmented channel needs only a few tracks more than a freely customized channel, and hence segmented channel routing is very important for FPGA design [5].

The FPGA Segmented Channel Routing Problems (FSCRP) is a special type of detailed net routing in the VLSI design process. The main task in channel routing is to connect pins of signals in a rectangular segmented routing channel without conflict. Channel routing on row-based segmented FPGA is different from traditional channel routing in which the routing resources can be freely customized by proper mask to obtain the desired connections. Segmented FPGA channel routing is more restricted because the routing is constrained to the wiring segments with predefined length and position in the channel [3, 4]. FPGA routing is carried out by programming the switch elements, such as the cross-fuses and anti-fuses, to provide the low resistance bidirectional interconnections. However any programmed switch will produce significant resistance and capacitance (RC) in the signal path. Propagation delay added by each RC stage of programmed switches in the signal path is one of the major factors in FPGA performance [12]. Hence the reduction of routing cost which affects the propagation delay by the programmed switches becomes very important. As a result, in order to achieve good performance, the number of switches used in routing a net must be restricted carefully to reduce the routing cost.

2 Previous Work

The FSCRP is known to be NP-complete [21]. Various routing approaches based on evolutionary strategies and neural networks have been proposed. In [2], Gamel *et al.* presented the row-based segmented channel FPGA architecture. Greene *et al.* [3]-[5] formulated the FSCRP and proved that the FSCRP can be adequately solved in practice, although this problem is NP-complete in general. The first known theoretical results on the combinatorial complexity

0-7803-9363-5/05/$20.00 ©2005 IEEE.

and algorithm design for segmented channel routing are presented by Roychowdhury, Greene, and Gamal in [5].

In [6], Burman *et al.* presented the staggered non-uniform length segmentation model for high performance FPGA. In this model, a channel is partitioned into several regions. In each region, a certain number of tracks with equal length segments are allocated, and the segments are arranged in a staggered fashion. However, the segment length is not the same in different regions but varies regularly across the regions. They also developed a greedy heuristic algorithm called FSCR for this model. They showed that this model and the algorithm can drastically improve the longest net delay and average net delay as compared to the conventional uniformly segmented model. In [9], Pedram *et al.* studied the design problem of FPGA channel segmentation architecture and developed analytical models of routability of row-based FPGA based on the staggered non-uniform length segmentation model. They demonstrated that for the connections using probabilistic models for the origination point and length, the routability of an FPGA with properly designed segment length and distribution can be nearly as efficient as mask-programmable channel architecture.

In [1], Funabiki *et al.* proposed a novel gradual neural network (GNN) approach for the FSCRP, which is believed as one of the best methods to simultaneously resolve the constraints and optimize the objective function. The GNN approach consists of a binary neural network and a gradual expansion scheme. The neural network satisfies the constraints of the problem by solving a motion equation, while the gradual expansion scheme minimizes the cost by gradually increasing the number of activated neurons. Although the GNN approach can solve the FSCRP efficiently, the GNN may be trapped in local optima.

Genetic algorithms (GA) are robust stochastic search algorithms analogous to the biological evolution process [15]. During the last years, GA has become more and more popular in the domain of search, optimization and machine learning. In many areas they are superior to other classical optimization techniques. GA has been widely and successfully used in various routing problems, e.g., VLSI channel routing [14], [16]-[20]. In this paper, we apply GA to solve the FSCRP and show that solutions with significantly better qualities can be obtained.

3 Problem Definition of the FSCRP

In this paper, we follow Funabiki's problem definition of the FSCRP [1]. The channel routing problem is formulated as a row assignment problem where each net is assigned to one of the tracks within a channel without constraints violation. Each net within a channel may occupy at most one track due to the technology constraint which does not allow connecting anti-fuses in an L-shaped fashion [13].

In the FSCRP, a routing channel composed of M tracks with L columns, an $M \times L$ anti-fuse matrix F for the segment formulation and a set of N nets are given as input. Each net is described as a pair of the leftmost and rightmost columns to be connected, denoted by $left_i$ and $right_i$ for net i (i = 1, ... , N). The goal of the FSCRP is to find a conflict-free net assignment in the tracks within the channel with the minimum routing cost. The routing cost is given by the total number of anti-fuses programmed in the channel routing for simplicity. The routing cost w_{ij} associated with net i's assignment to track j is calculated by:

$$w_{ij} = \sum_{k=left_i}^{right_i-1} f_{jk}$$

where f_{jk} is the jkth element of anti-fuse matrix F, and f_{jk} (j = 1, ... , M, and k = 1, ... , L) is 1 if an anti-fuse is located between columns k and $(k+1)$ on track j, and 0 otherwise.

Let x_{ij} be a binary variable with x_{ij} = 1 for net i's assignment to track j and x_{ij} = 0 for no assignment. Then, the FSCRP is formulated as follows:

Definition 1: Given a channel of M tracks with an anti-fuse matrix F and a set of N nets $\{(left_1, right_1),...,(left_N, right_N)\}$, the FSCRP requires to

$$\text{Minimize} \sum_{i=1}^{N}\sum_{j=1}^{M} w_{ij}x_{ij}$$

$$\text{Subject to} \sum_{j=1}^{M} x_{ij} = 1 \quad \text{and}$$

$$\sum_{j=1}^{M}\sum_{k=1(k\neq i)}^{N} d_{ikj}x_{ij}x_{kj} = 0$$

for i =1, ... , N

d_{ikj} is 1 if nets i and k must share the same segment on track j, and 0 otherwise. d_{ikj} can be defined by

If f_{jp} = 1 for $\exists p \in \{1,..,L\}$ with

$right_i \le p < left_k$ or $right_k \le p < left_i$, then $d_{ikj} = 0$, else $d_{ikj} = 1$.

4 Proposed Genetic Algorithm for FSCRP

4.1. Representation of Chromosome

The *Permutation Encoding* is used to represent the assignment of nets. For a channel routing problem of N nets with M tracks and L columns, the chromosome will consist of N scalar values. Each scalar value in the chromosome represents the position (track) where the corresponding net is located.

We represent each net in a chromosome by a scalar value from 0 to M-1 which represents the

number of the track located. With this representation, each of the net is assigned to one and only one of M tracks. Each chromosome which consists of N scalar values represents a solution of channel routing for N nets. In case l-th net is assigned to j-th track, the scalar value assigned for l-th net is $a_i = j$.

4.2. Objective Function

The goal of the FSCRP is to find a conflict-free net assignment in the tracks within the channel with the minimum routing cost. In this paper, the total fitness value is evaluated by the routing cost and constraints violations in the channel routing.

The routing cost is defined as the total number of programmed anti-fuses used in a channel routing. The routing cost is formulated as follows:

$$f(x) = \sum_{i=1}^{N}\sum_{j=1}^{M} w_{ij}x_{ij}$$

where the routing cost w_{ij} associated with net # i's assignment to track #j is calculated by: $w_{ij} = \sum_{k=left_i}^{right_i-1} f_{jk}$

The constraints are defined as the following:

- One net can be assigned to only one track;
- Two nets can not share the same segment in a track.

Then the total number of constraints violated for all nets assigned in a channel routing is formulated as follows:

$$P(x) = \sum_{i=1}^{N}\sum_{j=1}^{M}\sum_{k=1(k\neq i)}^{N} d_{ikj}x_{ij}x_{kj} \quad \text{, for } i = 1,...,N$$

We use one of the most popular techniques in genetic algorithms for tackling constraints: a fitness function with penalty terms. The penalty term introduces a penalty on the individual if the individual does not meet the constraints. In the paper, the final fitness function with a penalty term is expressed as following:

$$F(x) = f(x) + W \bullet P(x)$$
$$= \sum_{i=1}^{N}\sum_{j=1}^{M} w_{ij}x_{ij} + W \bullet \sum_{i=1}^{N}\sum_{j=1}^{M}\sum_{k=1(k\neq i)}^{N} d_{ikj}x_{ij}x_{kj}$$

where W represents the penalty coefficient. The choice of appropriate value for the penalty coefficient W is dependent on the problem. It may be different for a different problem. In this paper, the penalty coefficient W is selected based on the results of experiments. With various penalty coefficients W to solve a typical instance, the value which gives the best performance is selected as the final value of penalty coefficient W.

4.3. GA Operators

The selection strategy is responsible for choosing the parents among the individuals of the population to produce the offspring. Binary tournament selection is used for the selection process in this paper. Two individuals are taken at random from the population and compared with each other. The better individual is selected from the two, after which the two individuals are immediately replaced into the population for the next selection operation.

We use the single-point crossover operator that gives high-quality routing parts of the mates an increased probability of being transferred to their descendants. Mutation operators perform random modifications on an individual. The purpose is to overcome local optima and to exploit new regions of the search space. The mutation operator in this algorithm works as follows. Random value change – one of the scalar values in a chromosome is randomly selected, and the value of the bit selected is changed randomly one of the values between 1 to M track, which represents that the track of the net randomly selected is randomly assigned to another track. Elitism is implemented to prevent the loss of highly fit chromosomes.

The following is a summary of GA parameters used in this paper, which are selected based on experiments:

- Selection probability: 0.90
- Crossover probability: 0.85
- Mutation probability: 0.005
- Penalty coefficient: 2

5 Simulations for Performance Evaluation

The GA approach is evaluated on the benchmark of Funabiki's GNN method [1]. We generated 30 FSCRP instances with same statistical distribution as used by Funabiki [1]. The Staggered Non-uniform Length Segmentation model is used in the simulation. Burman *et al.* proposed this new channel segmentation model for high performance FPGAs in [6]. In this paper, the unit segment length S indicates the length of the shortest segment, which is equal to the number of columns of the shortest segment in the channel, and the unit segment length is increased in a staggered fashion by S columns in every four tracks. For example , if tracks 1 to 4 have a segment length S, then tracks 5 to 8 have a length 2S, and so on. The offset parameter is set to the one fourth of the corresponding segment length in the region. As for the net list, the leftmost column of a net is randomly generated with a uniform distribution and the net length is randomly generated with a gamma distribution. This condition comes from the results analyzed in [9].

For each instance, a total of 50 or 100 simulation runs is repeated by using different initial populations, and the minimum, maximum routing costs, average routing costs and the standard deviation of solutions are used for performance evaluation. 10 instances and corresponding performance using the GNN approach [1] are used for comparison with the GA approach. The comparison results are shown in Table 1. The routability (routing ratio) is defined as the ratio of number of total feasible solutions in total simulation runs.

The results show that the GA is superior to the GNN in terms of best cost, average cost, standard deviation of cost distribution and routability. The comparison between the best, worst, and average solution quality in Table 2 shows that GA is able to deliver better solution qualities compared to the GNN. The smaller standard deviations obtained by GA compared to the GNN also show that the GA method is less sensitive to the initial conditions.

6 Conclusions

In this paper a GA approach is presented for the FPGA segmented channel routing problems. After the study of the results we have reached to the following conclusions: The algorithm gives good distributions and feasible routing solutions in the FSCRP and is verified as a practical algorithm to solve the FSCRP. The comparison between GA and GNN methods indicates that GA is superior to GNN method which is considered as one of the best published methods for the FSCRP.

It was also observed that the runtimes of GA are largely depends on setting of parameters. Further improvements can be done through the parallelization of fitness function evaluation as our future work.

References

[1] N. Funabiki, M, Yoda, J. Kitamichi, and S. Nishikawa "A gradual neural network approach for FPGA segmented channel routing problems," *IEEE Transactions on Systems, Man, and Cybernetics*, vol, 29, pp481-489, No.4, August 1999

[2] A. E. Gamal, J. Greene, J. Reyneri, E. Rogoyski, K. A. El-Ayat, and A. Mohsen, "An architecture for electrically configurable gate arrays," *IEEE J. Solid-State Circuits*, vol. 24, pp. 394-398, Apr. 1989.

[3] J. Greene, V. Roychowdhury, s. Kaptanoglu, and A. E. Gamal, "Segmented channel routing," in *Proc. 27th Design Automat. Conf.*, 1990, pp. 567-572.

[4] A. E. Gamal, J. Greene, and V. Roychowdhury, "Segmented channel routing is nearly as efficient as channel routing (and just as hard)," in *Proc. 13th Conf. Advanced Research VLSI*, 1991, pp. 192-211.

[5] V. P. Roychowdhury, J. W. Greene, and A. E. Gamal, "Segmented channel routing," *IEEE Trans. Computer-Aided Design*, vol. 12, pp. 79-95, Jan. 1993.

[6] S. Burman, C. Kamalanathan, and N. Sherwani, "New channel segmentation model and associated routing algorithm for high performance FPGA's," in *Proc. IEE Int. Conf. Computer Aided Design*, 1992, pp. 22-25.

[7] K. Roy, "A bounded search algorithm for segmented channel routing for FPGA's and associated channel architecture issues," *IEEE Trans. Computer-Aided Design*, vol. 12, pp. 1695-1750, Dec. 1993.

[8] K. Zhu and D. F. Wong, "On channel segmentation design for row-based FPGA's," in *Proc. IEEE Int. Conf. Computer-Aided Design*, 1992, pp. 26-29.

[9] M. Pedram, B. S. Nobandegani, and B. T. Preas, "Design and analysis of segmented routing channels for row-based FPGA's," *IEEE Trans Compute-Aided Design*, vol. 13, pp. 1470-1479, Dec. 1994.

[10] W. N. Li, "The complexity of segmented channel routing," *IEEE Trans. Compter-Aided Design*, vol. 14, pp. 518-523, Apr. 1995.

[11] K. Roy and S. Nag, "On routability for FPGA's under faulty conditions," *IEEE Trans. Comput.*, vol. 44, pp. 1296-1305, Nov. 1995.

[12] S. M. Trimberger, *Field-programmable Gate Array Technology*. Norwell, MA: Kluwer, 1994.

[13] J. Kouloheris and A. El Gamal, "FPGA performance versus cell granularity," *Proc. IEEE Custom Integrated Circuit Conf.*, 1991

[14] M. Geraci, P. oriando, F. Sorbello and G. Vasallo, "A genetic algorithm for the routing of VLSI circuits," Euro Asic '91, Parigi 27-31 Maggio, Los Alamitos, CA: *IEEE Computer Society Press*, pp. 218-223, 1991.

[15] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Matching Learning, Reading*. MA: Addison-Wesley, 1989.

[16] J. Lienig and K. Thulasiraman, "A genetic algorithm for channel routing in VLSI circuits," *Evolutionary Computation*, vol. 1, no. 4, pp. 293-311, 1994.

[17] Y.-L. Lin, Y.-C. Hsu and F.-S. Tsai, "SILK: A simulated evolution router," *IEEE Trans. On Computer-Aided Design*, vol. 8, no. 10, pp. 1108-1114, Oct. 1989.

[18] B. B. P. Rao, L. M. Patnaik and R. C. Hansdah, "Parallel genetic algorithm for channel routing problem," *Proc. of the IEEE, Third Great Lakes Symposium on VLSI Design*, pp. 69-70, March 1993.

[19] L. Davis. *Handbook of Genetic Algorithms*. Van Nostrand Reeinhold, New York, 1991.

[20] Hee Il Ahn, Seung K. H, Tae W. C, "Genrouter: a genetic algorithm for channel routing problems," *IEEE*, 0-7803-2624-5/95, 1995.

[21] T.G. Szyanski. "Dogleg channel routing is NP-complete," *IEEE Tran. On CAD*, 4(1): 31-41, 1985.

[22] Srinivas, E. and Patnaik, L. M., "Genetic algorithms: a survey," *Computer*, Vol. 24/6, 17-26, 1994

[23] K.C. Tan, M. H. Lim, X. Yao, and L.P. Wang (Eds.), *Recent Advances in Simulated Evolution And Learning*, World Scientific, Singapore, 2004.

[24] L.P. Wang, Kay Chen Tan, Takeshi Furuhashi, Jong-Hwan Kim, and Xin Yao (Editors), *Proceedings of the 4th Asia-Pacific Conference on Simulated Evolution And Learning (SEAL'02)*, vol.1 – vol.2, 2002.

TABLE 1

Routing Cost of Soulutions by GA and GNN

	GA				GNN			
Instances	Routing Ratio	Avg cost	Min cost	Std dev	Routing Ratio	Avg cost	Min cost	Std dev
#1	1.00	14.19	11	2.07	0.96	24.07	17	3.31
#2	1.00	18.00	16	1.14	0.92	26.55	22	2.40
#3	1.00	12.08	8	1.98	0.69	23.35	17	2.46
#4	1.00	10.71	7	1.93	0.95	19.81	12	3.68
#5	1.00	14.28	11	1.53	0.90	21.03	15	2.79
#6	0.96	15.85	11	2.77	0.47	26.51	20	3.04
#7	1.00	18.56	17	1.03	0.56	28.46	25	2.19
#8	0.96	14.96	13	1.61	0.08	28.50	24	3.35
#9	0.80	17.05	15	1.30	0.27	24.30	19	2.99
#10	1.00	12.44	9	2.19	0.75	22.53	16	3.36