International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 3 Issue 5, May - 2014

Efficient Placement Driven Routing Algorithm for FPGA Using PSO

Shobana V
PG Scholar
Dr. N.G.P. Institute of Technology
Coimbatore, India

Nagalakshmi Venuopal
Associate Professor
Dr. N.G.P. Institute of Technology
Coimbatore, India

***Abstract*—Placement plays vital role in FPGA design flow. Many heuristic optimization algorithms are used to solve the FPGA parameters which are channel width, wire length, power and delay. This paper proposed Particle Swarm Optimization (PSO) algorithm to reduce interconnection wire length between blocks by placement. This algorithm takes minimum wire length by taking minimum Personal best. Based on personal best the global best also minimum. The results are compared with other placement tools and algorithms.**

Keywords—placement, FPGA

I. INTRODUCTION

The Field Programmable Gate Array (FPGA) is a programmable chip [8] which is programmed by user desire functionality [14]. It can be used to implement any digital circuit. It can be easily reconfigured by the designer and that are flexible [7]. Figure 1 shows the architecture of FPGA. It consist Configurable Logic Block (CLB), Input Output Block (IOB) and horizontal and vertical routing channels [9]. The switch blocks are areas of intersection of horizontal and vertical routing channels [1]. There are many architectures are available based on switch blocks [6]. FPGA design flow consist synthesis, technology mapping, placement and routing. Placement is the one of the most important and time consuming problem in FPGA design [10] and placement is NP-Complete problem [11]. The placement is acceptable if the routability is 100% achieved. The objective of placement is minimizing the channel width and wire length. Many heuristic optimization techniques are used to minimize the wire length and channel width.

This paper organized by six sections. First one is Placement problem following this Related work after that proposed work and after this Experimental Result then finally Conclusion and finished with References.

II. PLACEMENT PROBLEM

In FPGA the placement and routing problems are NP-Complete problems [11]. The placement problem of FPGA is defined in [12]. Given a set of n modules $M = \{ m_1, m_2,\ldots,m_n \}$ and a set of r signals $S = \{s_1, s_2,\ldots,s_r \}$, we associate each module $m_i \in M$ with a set of signals S_{mi}, where $S_{mi} \subseteq S$ [12]. Here the modules are CLBs or IOBs. For all signal $s_i \in S$, represent as signal net by a set of modules $M_{si} = \{m_j \mid s_i \in S_{mj}\}$ [13]. The placement is defined as allocating or placing blocks in appropriate places in order to minimize the wire length and channel width.

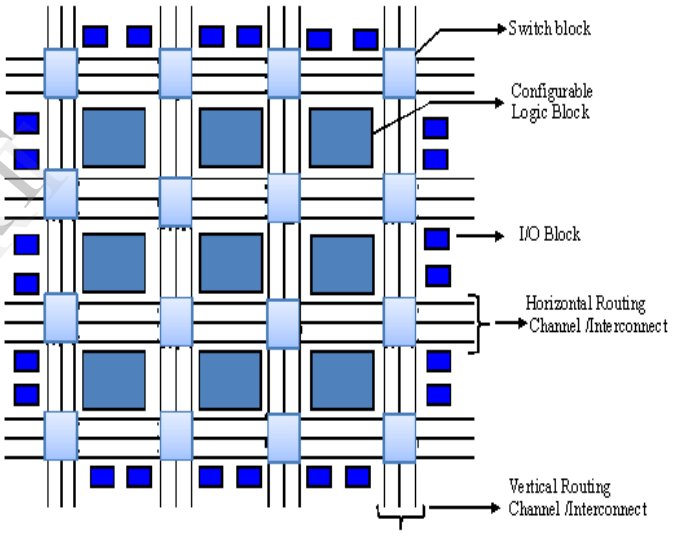

Fig. 1. FPGA Architecture

III. RELATED WORK

A. Genetic Algorithm with Artificial Neural Network

In [1] it presents the solution for both placement and routing problem together using Genetic Algorithm (GA) and Artificial Neural Network (ANN). The solution has four phases. The first phase of the solution solves the placement problem and other three phases solves the routability problem along with that placement which processed from first phase. Genetic algorithm used to set initial population of randomly generated solutions. With that ANN used to transform previous population solution to final set of population which is optimal. Here fitness function places a major role in selection process. Genetic operators are used to choose fitness function. These algorithms are implemented by SGI Origin-2000 12 node platform and MATLAB software. Here the results are obtained by giving large net files as input.

B. Genetic Algorithm

[2] Proposed a new Genetic Algorithm (GA) which is slightly differ from standard genetic algorithm. First process is selection operator. Here 30% of populations of higher fitness function are selecting, Remaining are selected based on individual fitness by probabilistic method. Next process is greedy; it improves the fitness function by exchanging positions of two blocks. Next one is elitism; it decides the good solution for providing to next generation. This algorithm implemented by C program. It takes Microelectronics Center of North Carolina (MCNC) benchmark circuits as input. The performance of the genetic algorithm is improved here.

C. Ant Colony Optimization

Boolean satisfiability (SAT) problem [16] and Ant Colony Optimization (ACO) [15] are proposed in [3]. Here first step is global router, it assigns the net which having many blocks. After this assign to vector of Boolean function. After apply the Boolean variable no two nets use same routing resource. Finally Boolean routability function finds the all possible routes of the net. After that ACO combinatorial optimization is applied in order to optimize the length. This results are minimum than other Boolean based solvers like Grasp, Zchaff [3].

D. Min-cut bi-partitioning algorithm

Min-cut bi-partitioning algorithm and slicing lines are introduced in [4]. Taking only cut size is not good method to minimize length. In a congested area it is difficult to find feasible routing. The cut line is used to partitioning the routing area and find optimal routing. The slicing line may be 2, 4, 8 or 16. Based on the unbalance bi-partitioning it can be choose any one slicing lines. This algorithm implemented by C program. Microelectronics Center of North Carolina (MCNC) benchmark circuits are taken for the experiment.

E. Particle Swarm Optimization

Particle Swarm Optimization (PSO) [18] is population based algorithm which is proposed in [14]. This PSO algorithm consist three variations which are 1.Simple PSO 2.Constricted PSO (CPSO) and 3.Time Varying Inertia Weight (TVIW) PSO. Simple PSO consist normal Personal best (Pbest), Global best (Gbest) and velocity. Constricted PSO introduce constriction factor [18] to limit the growth of population. TVIW PSO introduces inertia weight [17] in velocity update rule. These algorithms are applied to Binary Coded Decimal (BCD) Counter. Among these three variations TVIW PSO produce better result than first two types.

F. Simulated Annealing

In [13] Simulated Annealing (SA) is used. SA is traditional temperature based algorithm for FPGA placement. The variation of SA ultra Low Temperature Simulated Annealing (LTSA) [19] is used. This placement consist four steps [13] which are 1.Min cut Partitioning of CLB netlist and FPGA array by recursive bi-partition 2.Allocation of partitions to regions on FPGA followed by redistribution of blocks to handle overflow within regions 3.Placement and 4.Improvement by LTSA.

IV. PROPOSED METHOD

Particle Swarm Optimization (PSO) [20] is Meta heuristic optimization technique based on populations. PSO having two factors Pbest and Gbest. Both the above two factors and the present velocity of the particle affects the velocity in the next iteration. The velocity is added to the present location of the particle to get the next location which will help it move towards the best location (gbest), achieved by the swarm, while still looking for an even better location(improving pbest).

The velocity and positions are calculated by the following equation

$$V_{i+1} = V_i + C_1 * rand_1 ( ) * (pbest_i - X_i) + C_2 * rand_2 ( ) * (gbest - X_i) \quad (1)$$

$$X_{i+1} = X_i + V_i \quad (2)$$

Where rand ( ) = any random in a range of (0, 1) generated each time when the function is evaluated. And C_1 and C_2 are two constants known as acceleration constants.

V. EXPERIMENTAL RESULT

This PSO algorithm is implemented by MATLAB software. Microelectronics Center of North Carolina (MCNC) benchmark circuits [21] are taken for the execution. This Benchmark circuits consist Net files for each circuits like alu4.net, apex2.net, ex5p.net, Etc. Those Net files contain specification about number of CLBs, IOBs and nets. Applying PSO algorithm to those net files and it places the particles in random positions. That is known as initial population. The initial populations of particles are shown in figure 2. Then the particle positions are updated by the velocity of the particles and distance between the particles. The optimized particles are shown in figure 3. And the cost functions of circuit is shown in figure 4. It produces the optimum channel width. Results are compared with previous works.

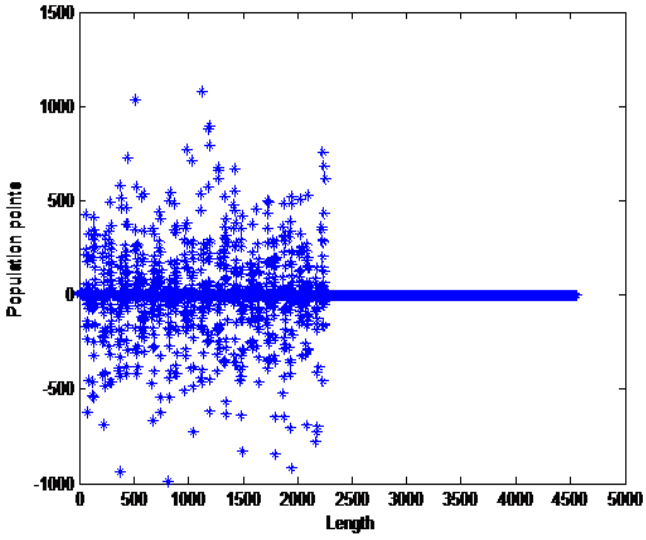

Fig. 2. Initial population

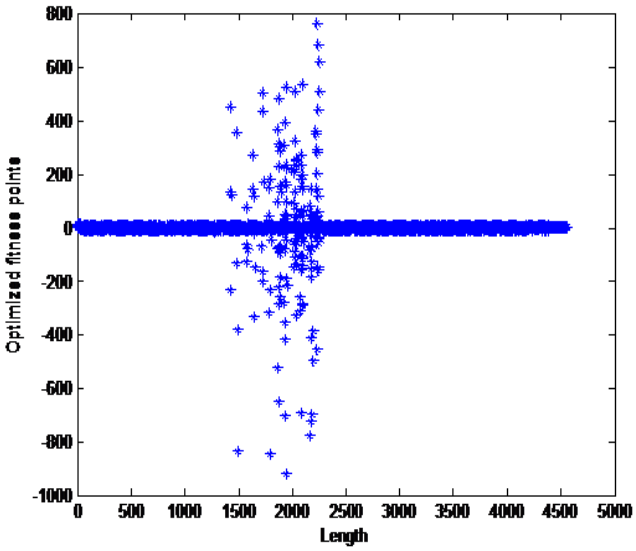

Fig. 3. Optimized population

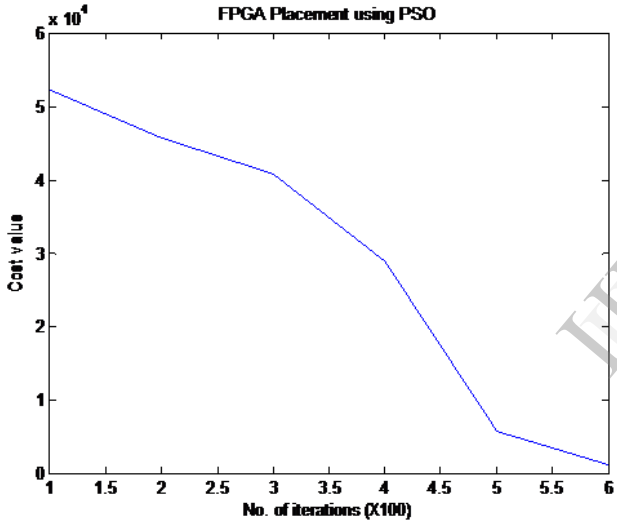

Fig 4. Cost Function

VI. CONCLUSION

In FPGA design flow the placement takes major role. The good implementation of FPGA digital circuits will be evaluated by many parameters. There are so many parameters are available to calculate efficiency. Some of the parameters are wire length, channel width, delay, running time, power. Many optimization algorithms are used to minimize the wire length and channel width. The proposed PSO algorithm produces better result and those results are compared in table 1with previous works like ACO [22], VPR [5].

REFERENCES

[1] Siva Nageswara Rao Borra, Annamalai Muthukaruppan, S. Suresh and V. Kamakoti, "A novel approach to the placement and routing problems for field programmable gate arrays", Applied Soft Computing, Elsevier, pp. 455–470, 2006

[2] M. yang and A.E Almaini, "FPGA Placement by using a Genetic Algorithm", EngineerIT, Electronics Technical, june, 2007

[3] Vinay Chopra and Amardeep Singh, "Ant Colony Optimization approach for Solving FPGA routing with minimum Channel Width", International Journal on Computer Science and Engineering (IJCSE), vol. 3, pp. 2855-2861, July, 2011

[4] Zoltan Baruch, Octavian Creţ and Kalman Pusztai, "Placement Algorithm for FPGA Circuits"

[5] V. Betz, and J. Rose, "VPR: A New Packing, Placement and Routing Tool for FPGA Research", in Proceedings of the 7th International Workshop on Field-Programmable Logic and Applications, pp 213- 222, 1997

[6] R. M.I. Masud and S.J.E.Wilton, "A new switch block for segmented FPGAs", Proceedings of InternationalWorkshop on Field Programmable Logic and Applications, Glasgo, Scotland, August, 1999

[7] Pritha Banerjee, Subhasis Bhattacharjee, Susmita Sur-Kolay, Sandip Das and Subhas C. Nandy, "Fast FPGA Placement Using Space-Filling Curve", International Conference on Field Programmable Logic and Applications, pp. 415 – 420, IEEE, 2005

[8] Sang-Joon Lee And Dr. Kaamran Raahemifar, "FPGA Placement Optimization Methodology Survey", Electrical And Computer Engineering, pp. 1981 – 1986, IEEE, 2008

[9] Y.W. Chang, D.F.Wong and C.K.Wong, "FPGA global routing based on a new congestion metric", in: Proceedings of IEEE International Conference on Computer Design, Austin, TX, October, pp. 372–378, IEEE, 1995

[10] H. Bian, A.C. Ling, A. Choong and J. Zhu, "Towards scalable placement for FPGAs", FPGA '10: Proceedings of the 18th Annual ACM/SIGDA International Symposium on Field Programmable Gate Arrays, ACM, New York, USA, pp. 147–156, 2010.

[11] V. Betz and J. Rose, "FPGA routing architecture: segmentation and buffering to optimize speed and density", in: Proceedings of ACM/SIGDA International Symposium on Field Programmable Gate Arrays, Monterey, CA, February, pp. 59–68,1999

[12] J. M. Emmert, S. Blanacha, and D. K. Bhatia, "Physical Layout Techniques for Field Programmable Gate Arrays", *manuscript*, Personal communication

[13] Banerjee, P.; Sur-Kolay, S. "Faster Placer for Island Style FPGAs", International Conference on Computing: Theory and Applications, ICCTA '07, pp. 117 – 121, 2007

[14] Prakash Kumar Rout, D.P.Acharya and G.Panda, "Novel PSO based FPGA Placement Techniques", International conference on computer and communication technology, pp. 630-634, IEEE, 2010

[15] Alaya, I. Solnon and C. Ghedira, "Ant Colony Optimization for Multi-Objective Optimization Problems", Tools with Artificial Intelligence, ICTAI Patras volume: 1, pp. 450-457, October, 2007

[16] R. Sethuram and M. Parashar, "Ant Colony Optimization and its Application to Boolean Satisfiability for Digital VLSI Circuits", Advanced Computing and Communications, International conference ADCOM, January, 2007

[17] Y.-L. Zheng, L.-H. Ma, L.-Y. Zhang, and J.-X. Qian, "Empirical study of particle swarm optimizer with an increasing inertia weight," in Proc. IEEE Congr. Evol. Comput., pp. 221-226, 2003

[18] M. Clerc and J. Kennedy, "The particle swarm-explosion, stability, and convergence in a multidimensional complex space," *IEEE* Transactions on Evolutionary Computation, vol. 6, no. 1, pp. 58-73, Febraury, 2002

[19] M. Huang, F. Romeo and A. Sangiovanni Vincentelli, "An Efficient General Cooling Schedule for Simulated Annealing", in Digest of IEEE/ACM ICCAD, pp. 381-384, IEEE, 1986

[20] V. G. Gudise and G. K Venayagamoorthy, "FPGA Placement and Routing Using Particle Swarm Optimization", Proc. of the IEEE Computer Society Annual Symposium on VLSI Emerging Trends in VLSI Systems Design (ISVLSI'04), 2004

[21] Wenyao Xu, Kejun Xu and Xinmin Xu, "A Novel Placement Algorithm for Symmetrical FPGA", IEEE, 2007

[22] Kai Wang and Ning Xu, "Ant Colony Optimization for Symmetrical FPGA Placement", IEEE, 2009

TABLE 1. CHANNEL WIDTH (MCNC Benchmark circuits)

Circuit	PSO	LTSA [13]	ACO [22]	VPR [5]
ex5p	3	14	14	14
apex4	6	13	15	13
alu4	7	11	10	10
seq	8	12	12	12
apex2	8	12	11	13
pdc	6	17	18	17
ex1010	6	13	10	11
spla	7	17	15	17