IOSR Journal of VLSI and Signal Processing (IOSR-JVSP)
Volume 9, Issue 1, Ser. I (Jan. - Feb. 2019), PP 34-41
e-ISSN: 2319 – 4200, p-ISSN No. : 2319 – 4197
www.iosrjournals.org

Timing Analysis, Area and Interconnect Length Optimization for VLSI by Using Genetic Algorithm and PSO Algorithm

G.Sujatha[1], Dr. Narayanam Balaji[2]
1. Associate Professor, Dept of ECE, Sree Rama Engineering College, Tirupathi
2.Professor in ECE Dept, JNTUK University of Engineering, Narsaraopet, A.P-India
Corresponding Author: G.Sujatha

***Abstract*:** *The Present research work, a new performance and area optimization algorithm for complex VLSI systems is presented. The floorplanning is employed to calculate to the relative location of blocks within the fixed outline. The planning obscurity is increasing and therefore the circuit size is obtaining additional. Thus ultimately area of the circuit gets rise and harder to optimize the Wirelength and area. By using the genetic algorithms in the performance and area optimization, we are able to find the optimal values for both delay and silicon area for the MCNC benchmark circuits. PSO is a recent intelligent heuristic search method in which the mechanism of algorithm is inspired by the swarming of biological populations. In fact, both of them use a combination of deterministic and probabilistic rules. New digital solutions are available to generate a hardware implementation of PSO Algorithms.*

Moreover, the inherent parallelism of these new hardware solutions with a large computational capacity makes the running time negligible regardless the complexity of the processing. During this paper we focus the particle swarm optimization (PSO) methodology to achieve global solution for fixed outline constraints for this we tend to taken MCNC and GSRC benchmark circuits.

Keywords: *VLSI, genetic Algorithm, PSO, MCNC benchmark*

Date of Submission: 25-02-2019 Date of acceptance:11-03-2019

I. Introduction

Over the last decade, several meta-heuristic algorithms are proposed to solve hard and complex optimization problems. The effectiveness of this algorithm give satisfaction to solve the most difficult problems for many algorithms related for various optimization problems. The proposed architecture is tested on some benchmarks functions. We have also analyzed the operators of GAs to describe how the performance of each one can be enhanced by incorporating some features of the other. We used standard benchmarks functions to make comparison between the two algorithms. In fact, PSO algorithm use the technique [1] that explores all the search space to fix parameters that minimizes or maximizes a problem. So, the ability and the simplicity to solve complex problems make the studies active in this area compared with many others optimization techniques [2] [3].

In the design of VLSI circuit, Power consumption, area and speed are the major design issues power dissipation has emerged as an important design parameter in the design of VLSI circuits, especially in portable computing and personal communication. Rapid advances in integration technology have tremendously increased the design complexity of very large scale integrated (VLSI) circuits, necessitating robust optimization techniques in many stages of VLSI design. A genetic algorithm (GA) is a stochastic optimization technique that uses principles derived from the evolutionary process in nature.

This research attempts to present that PSO has a good effectiveness to find the best global optimal solution as the GA but with a better computing efficiency (less using of resource hardware and execution time). The main objective of this paper is to compare the computational efficiency of our optimized PSO with GA and other PSO algorithms using a set of benchmark test problems. The results of this optimization algorithm could prove to be important for the future study of PSO. The organization of the paper is described as follow: The first chapter briefly introduces the general steps performing the mechanism of PSO. Especially, a brief introduction of pseudo random number generator [4]. The next section describes the background functional architecture which performs the GA and PSO algorithm. In chapter 3, a description of the architecture used in the hardware implementation of PSO and genetic algorithm; the second part illustrates the experimental results of some benchmarks functions applied into the PSO algorithm and compared with GA and others PSO algorithms. Finally, we conclude our work and we make some implications and directions for future studies.

II. Review of Recent Researches

Many researchers have found that optimization algorithms are best in solving many complex problems in various fields of science and engineering. Researchers have used different algorithms like genetic algorithm, integer linear programming for power reduction in the area of VLSI. Genetic algorithm is used by many authors to determine the minimum leakage vector as best solution [6] and [7]. In the previous work of authors [5] genetic algorithm is used but implemented in Verilog HDL and comparison is performed with [8]. In this paper an attempt is made to implement Particle Swarm Optimization algorithm in the field of low power VLSI to search for MLV as an optimum solution. To the best of the author's knowledge, PSO algorithm is used for the first time in IVC approach to find MLV for leakage power reduction.

In this paper author S. Lalwani in 2013 describes numerous problems encountered in real life that cannot be actually formulated as a single objective problem; hence the requirement of Multi-Objective Optimization (MOO) had arisen several years ago. Due to the complexities in such type of problems powerful heuristic techniques were needed, which has been strongly satisfied by Swarm Intelligence (SI) techniques. Particle Swarm Optimization (PSO) has been established in 1995 and became a very mature and most popular domain in SI. Multi-Objective PSO (MOPSO) established in 1999, become an emerging field for solving MOOs with a large number of extensive software, variants, literature, applications and codes. This paper reviews all the applications of MOPSO in miscellaneous areas followed by the study on MOPSO variants [9].

Floor-planning is the first stage of the very large scale integrated-circuit (VLSI) physical design technique,in this the resultant quality of this stage is very important for successive design stages. If we see this from the computational point of view, we conclude VLSI floor-planning is an NP-hard problem. In this paper, Jianli Chen in 2010, a hybrid genetic algorithm (HGA) for a non-slicing and hard-module VLSI floor-planning problem is presented. HGA uses an effective genetic search method to explore the search space and an efficient local search method to exploit information in the search domain. Experimental results on MCNC benchmarks show that the HGA is effective and promising in building block layout application [10].

III. Problem Definition

The main disadvantage of the slicing floor-plan approach is that for a particular set of circuit modules, the majority of the feasible layouts will be non-slicing [13]. In other words, the slicing floor-plan approach severely reduces the size of the search space and may eliminate the very best circuit layouts. On the other hand, reduction in the size of the search space is advantageous as long as the solutions are good enough. According to [15] this is indeed the case for problems where modules have exibility in their shape. Another feature of the slicing floor-plan that makes it an attractive proposition is the simple way that solutions can be represented as normalized post expressions [14]. It is interesting to note that techniques involving soft modules and slicing floor-plans are equally applicable to facility layout problems [11, 12] as they are to VLSI floor-planning.

This paper proposed by Russell C. Eberhart in 1998 compares two evolutionary computation paradigms: genetic algorithms and particle swarm optimization algorithm. Operators of each paradigm are reviewed, on focusing how each affects search behavior in the problem domain. The goals of the paper are to provide additional insights into how each paradigm performs, and to suggest ways in which performance might be improved by incorporating features from one paradigm into the other [16].

IV. Proposed Method

In the present research work, an efficient global optimization algorithm will be proposed for optimization of various parameters like power consumption, area and speed .In the initial phase one of the recently proposed global optimization algorithm, Modified Genetic Algorithm (MGA) will be used.. MGA combines the good methodologies of the two algorithms like global minimum converging property of GA algorithm and fast convergence rate of Hamming scan algorithm. GA has the drawback of premature convergence. The demerit of Hamming scan algorithm is that it gets stuck at the local minimum point, because it cannot distinguish between local minimum point and a global minimum point. Hence it is sub-optimal. MGA overcomes these drawbacks. MGA is used as a statistical technique for the synthesis of optimal VLSI circuit design. In this work, a multi-objective Modified genetic algorithm based used to optimize the area, power dissipation and speed.

Slicing floorplan: A slicing floorplan is obtained by cutting the floorplan either horizontally or vertically repetitively. Fig.1 (a) represents slicing floorplan. A slicing tree could be a binary tree. The pre-placed module could be a one during which modules coordinates' area unit given by the floorplanner. Let H denotes set of hard modules, S denotes set of soft modules and P denotes set of preplaced modules. Let M be the union of those three sets of modules. The illustration of floorplanning will be exhausted two layout forms, specifically the slicing structure and non-slicing that is used to represent a slicing floorplan. Generally, there are two cut sorts, + and -. The + (-) represents floorplan horizontal (vertical) cut. Fig.1 (b) shows a slicing tree of

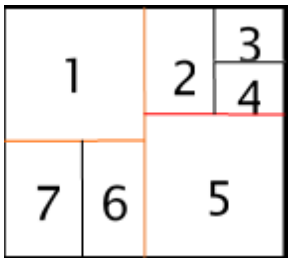
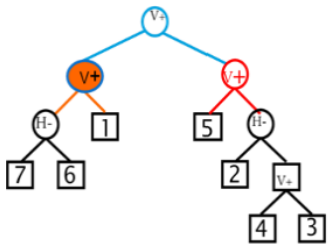


Figure.1. (a) slicing floorplan



(b) slicing tree

Non slicing floorplan: Non slicing floorplan is more common than slicing floorplan. All the children of the given cell cannot be obtained by bisecting the floorplan. This is called non-slicing floorplan. Horizontal constraint graph and vertical constraint graph can be used to model a non-slicing floorplan. In a constraint graph, a node represents a module.

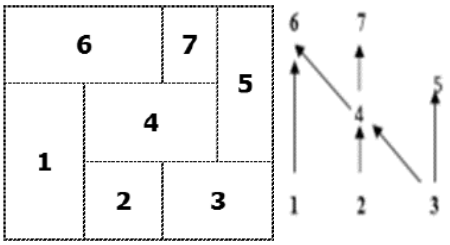
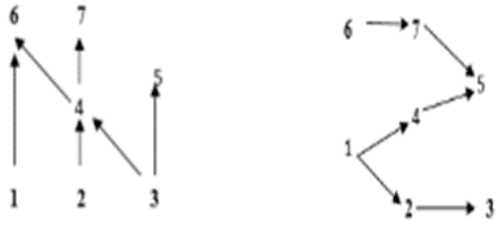


Figure.2. (a). Non slicing



(b) floorplan Vertical and Horizontal Constraint graph

The foremost aim of this paper to minimize the dead space (white space) & fix the module in fixed outline constraint. In this paper we dealt with slicing floorplanning

GA, presented in Algorithm 3 is an example of a steady state GA based on the classication given in [15]. It also uses the weaker parent replacement strategy _rst described in [1]. As outlined in Algorithm 3, our GA applies genetic operators to permutations of rectangle records. The _tness values are based on the amount of dead space produced in each oorplan, F, de_ned by the individual normalized post_x expressions encoded in the population. The percentage of dead space is de_ned as follows:

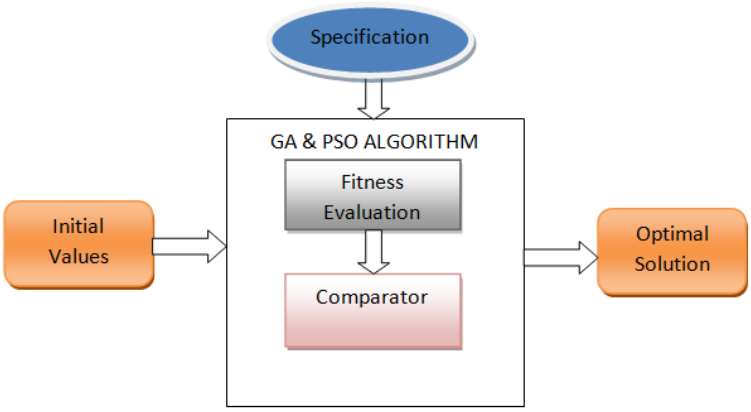


Figure 3. The Block diagram for proposed method

$$\frac{A(R_F)-\sum_{i=1}^{n}A(R_i)}{\sum_{i=1}^{n}A(R_i)}X100 \text{ ----------(1)}$$

where A(RF) is the area of the enclosing rectangle for the oorplan and A(Ri) is the area of the ith basic rectangle.

the _rst parent is selected deterministically in sequence, but the second parent is selected in a roulette wheel fashion, the selection probabilities for each genotype being calculated using the following formula:

$$Selection\,\Pr obability = \frac{(Rank)}{\sum Ranks} \text{----------(2)}$$

selection probability = P(Rank) Ranks where the genotypes are ranked according to the values of the waste that they have produced, with the worst ranked 1, the second worst 2, etc., and the best ranked highest.

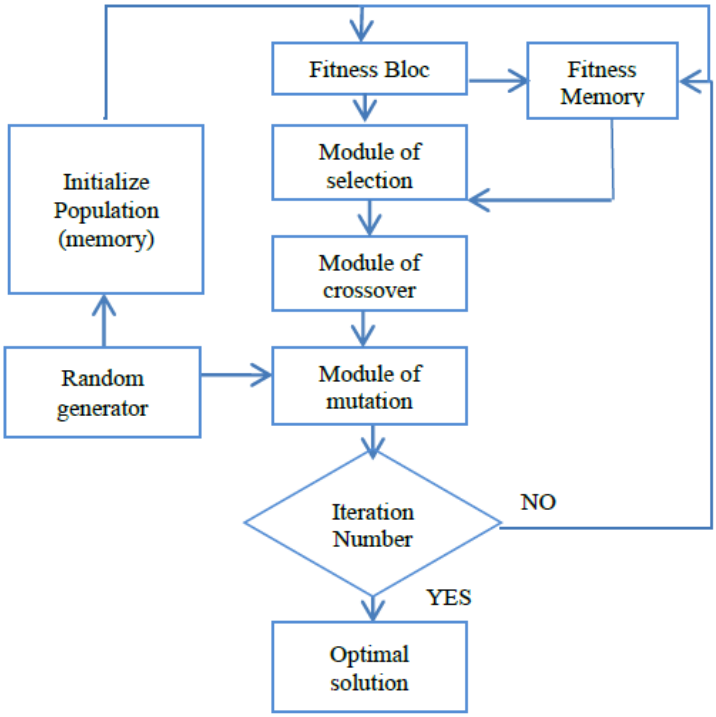


Figure 4: Flowchart for GA

The GA breeds permutations of records from which our decoder produces normalized post_x expressions. These expressions are, in turn, processed by adding the shape curves together (as described in Section 2) for each horizontal or vertical combination, recording the percent of dead space at the end.

The initial population consists of random permutations of records with each basic rectangle represented exactly once in each list. The op-type ag for each record is set to `+' or `*' with equal probability, and the value in the length _eld is generated in two stages:

Stage 1: length = 0, with a probability of 0.5

Stage 2: if the length is not set to zero, then it is generated from a Poisson distribution with mean 3

a).Particle Swarm Optimization (PSO)

PSO is an improvement technique impressed by swarm intelligence. PSO could be a population-based evolutionary formula within which the formula is initialized with a population of random solutions. During this algorithm essentially learned from animal's activity or behavior to resolve improvement issues. Every member of the population is named a particle and also the population is named a swarm. Beginning with an at random initialized population and occupation at random chosen directions, every particle goes through the looking area and remembers the simplest previous positions of itself and its neighbors. Every particles of swarm communicate its best positions to every alternative. Consequent step begins once all particles are moved. Finally, all particles tend to fly towards higher and higher positions over the looking method till the swarm move to close to an optimum of the fitness operate. The procedural flow of Particle Swarm Optimization

Step 1: Load the modules and initialize the parameters of the PSO algorithm.

Step 2: Generate an initial population with particle dimension corresponding to the number of modules to be optimized and initialize its positions.

Step 3: Calculate the fitness value of each particle using area and then assign the fitness values to its corresponding
particles. Let the initial global best be the lowest Pbest value.
Step 4: Update the velocity of the particle.
Step 5: In the consecutive iterations check every particle. If its fitness value is better than its corresponding previous
Pbest, then update its Pbest along with the fitness value and particle.
Step 6: Update Gbest for each and every iteration. If the earlier Gbest is higher than the Gbest obtained in current iteration, then update newer one as the final Gbest.

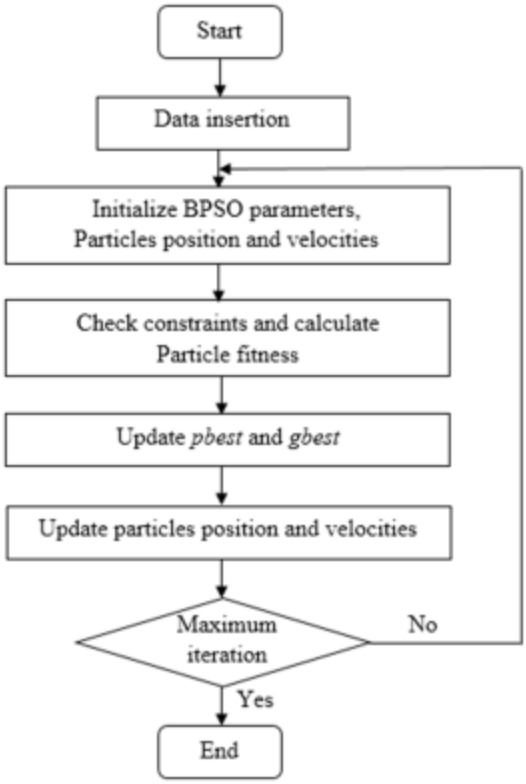


Figure.5. Flow chart of Particle Swarm Optimization algorithm

Step 7: Repeat step 3 to step 6 till the termination condition is reached. The termination condition or stopping criteria may be the end of number of iteration or the repetitive occurrence of the same output for certain number of iterations specified by the user.
Step 8: halt the process, if termination condition is satisfied.

V. Experimental Results

The experimental results are compared with the existing method and proposed method we shown that Power is saved and leakage current, area reduced. Further power can be saved by using high and low threshold voltage and by using various parameters through Layout optimization.

In the following Table, contains all optimization algorithms that are explained and also contains the method involved, their comparisons with other optimization algorithms, along with the drawbacks. The table containing all these is shown below:
Area Estimation: It‘s the area of rectangle of minimum size, enclosing all the blocks as shown in Fig 4.

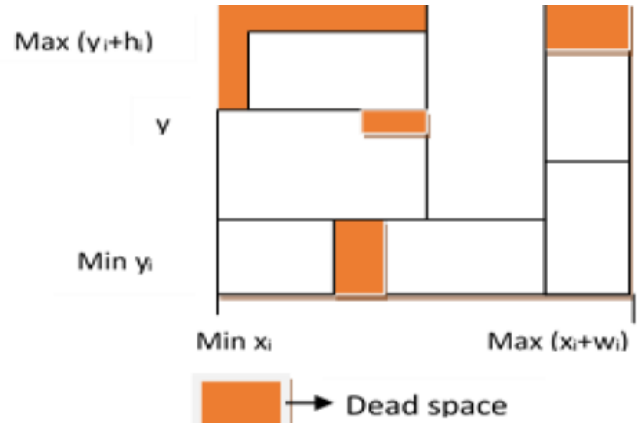


Figure.6. Block position & dead space

Therefore the total area will be *Area (F)* = (max (xi+ wi) – min (xi)) (max (yi+ hi)–min (yi))

Fitness Function: The VLSI floor planning is a minimization problem, and the objective is to minimize the cost of floorplan F, i.e., cost (F).Thus, the fitness of an individual in the population is defined as follows:

F(x, wh) = 1/ cost (f) (3)

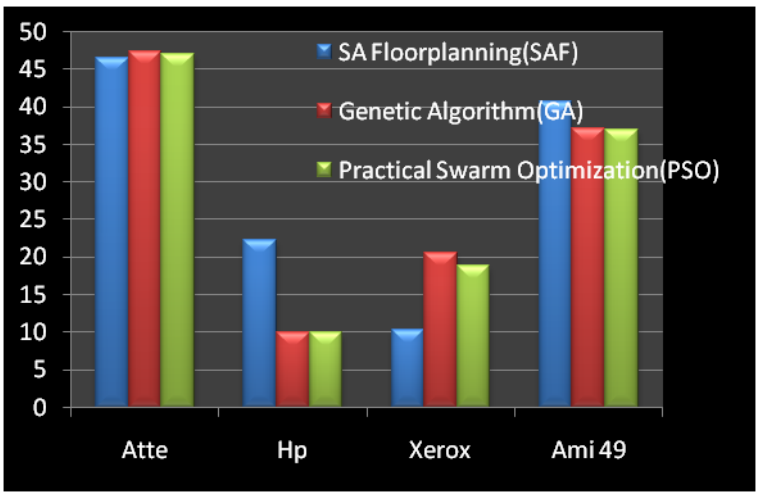
Where, f(x, w) is the corresponding floorplan of (x, wh),

Cost (F) is the cost of floorplan, x is a matrix which has the (x, y)

Location of each module and w is a matrix which has corresponding width and height of each module.

VI. Results

The experiments during this study used GSRC and MCNC benchmarks for the proposed floorplanner and compare with simulated annealing (SA) and fast simulated annealing (FSA). All the cells were set as hard modules. The simulation programs were compiled in MATLAB, and also the results were obtained on a Pentium 4 1.7 GHz with 512MB RAM. The PSO experiments with w, c1 and c2 initializations were 0.4, 1 and 1.5, severally. The particle range is about as five. We have a tendency to run the each floorplanner ten times and calculated their average outcomes of chip area and run time. The experiment results of each floorplanner are shown in Table one. Compare with SA and FSA, our methodology will notice an improved floorplan solution in even less computation time. Beneath an equivalent tree structure, that's to mention, our technique has a lot of efficiency and solution searching ability for floorplan. Though the SA in adopted three an equivalent operations that mentioned higher than, however it'd randomly pick up the operation (somewhat sort of a reasonably trial and error strategy) however not following the previous expertise whereas making an attempt to seek out another higher resolution. This may lead to the floorplanner waste an excessive amount of time on trapping into native minimal and tougher to get a stronger solution. Our technique will overcome these drawbacks. Thus, the appropriate resolution will find out in shorter computational time. Relative to each strategies, our technique possesses a lot of strength to prevent the solution from falling into local minimal. It'd be useful to seek out a stronger solution in shorter time.



Graph 1. Area Comparison of MCNC Benchmark Circuits

The proposed VLSI floor planning based on Particle Swarm Based Optimization with polish expression on fixing modules with in fixed outline constraint. The PSO ability to identify the feasible solution for soft and hard modules instead of simulated annealing, fast simulated annealing (FSA).

Table 1. Comparisons for existing and Proposed methods

Benchmark circuit	SA based Floor planning (SAF)	GA	PSO
Apte	46.56	47.3	46.95
hp	22.22	10.05	10.02
Xerox	10.28	20.56	18.83
Ami 49	40.66	37.16	36.89

The graph clearly explains the experimental result of floorplanning delivers global solution with GSRC benchmark circuits compared with SA, FSA algorithm.

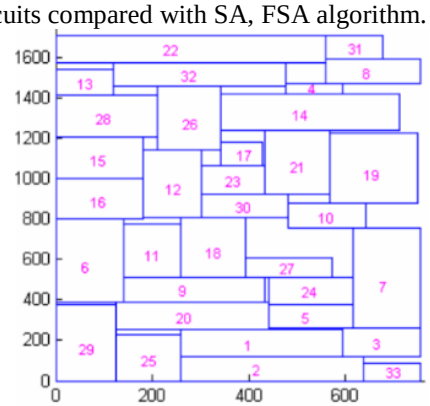


Figure.7. Floorplanning result of Hp circuit

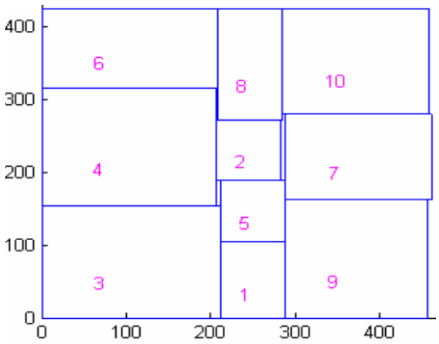


Figure.8. Floorplanning result of Xerox circuit

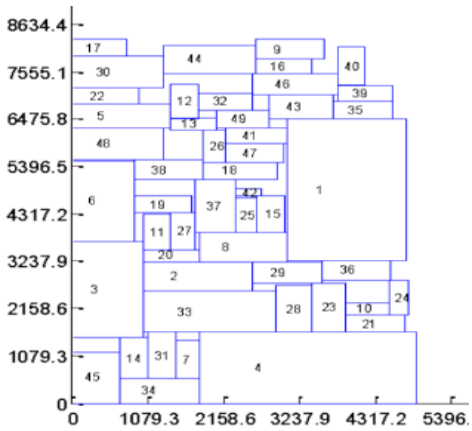


Figure.9. Floor planning of ami49 benchmark circuit

VII. Conclusion

VLSI design processes are such that own special circuits of considerable complexity can be designed by system engineers. This provides an improved degree of freedom for designers. Heuristic algorithms are suitable tools to determine the optimum solutions of the engineering problems. But, their applications are limited by the high computational cost of the slow convergence rate. The various optimization algorithms used for floor-plan based design methodology have been reviewed.

References

[1]. P.K. Tripathi, S. Bandyopadhyay, K.S. Pal, Multi-objective particle swarm optimization with time variant inertia and acceleration coefficients, Information Sciences 177 (22) (2007) 5033–5049.

[2]. A.S. Mohais, R. Mohais, C. Ward, Earthquake classifying neural networks trained with dynamic neighborhood PSOs, in: Proceedings of the 9th Annual Conference on Genetic and, Evolutionary Computation (GECCO'07), pp. 110–117.

[3]. G.S. Tewolde, D.M. Hanna, R.E. Haskell, and Multi-swarm parallel PSO: hardware implementation, in: Proceedings of the 2009 IEEE Symposium on Swarm (2009).

[4]. D.H. Lehmer, Mathematical methods in large-scale computing units, Ann. Computing Lab. Harvard Univ. 141-146, 1951.

[5]. V.LeelaRani, M.Madhavi Latha, "Implementation of genetic algorithm for minimum leakage vector in input vector control approach", IEEE conference,SPACES,Jan2015.
[6]. Zhanping chen,Liqiong Wei,Mark johnson,Kaushik Roy, "Estimation of standby leakage power in CMOS circuits considering Accurate modeling of Transistor stacks", ACM transactions, international symposium on Low power electronics and design, ISLPED, pp.239-244,1998.
[7]. Mehdi Kamal, AliAfzali-Kusha, SaeedSafari, MassoudPedram, "Design of NBTI-resilient extensible processors", INTEGRATION, the VLSI Journal,vol.49,pp.22-34,2015.
[8]. Abdoul Rjoub,Almotasem Belleah Alajlouni,Hassan Almanasrah, " A Fast input Vector Control Approach for subthreshold Leakage power reduction" , IEEE electro technical international conference,MELECON,2012.
[9]. S. Lalwani, S. Singhal, R. Kumar and N. Gupta,"A Comprehensive Survey:Applic-ations Of Multi-Objective Particle Swarm Optimization(MOPSO) Algorithm
[10]. Chiu Wing Sham,Evangeline F.Y. Young," Routability Driven Floorplanner With Buffer Block Planning", ISPD '02, April 7-10, 2002, San Diege, California, USA))))))000000
[11]. Jaime Garces-Perez, Dale A. Schoenefeld and Roger L. Wainwright, Solving Facility Layout Problems Using Genetic Programming. Proc. 1st Annual Conference in Genetic Programming, pages 182-190, 1996.
[12]. Kazuhiro Kado, Peter Ross and David Corne, A Study of Genetic Algorithm Hybrids for Facility Layout Problems. Proc. 6th International Conference on Genetic Algorithms, pages 498-505, 1995.
[13]. S. Nakatake, K. Fujiyoshi, H. Murata and Y. Kajitani. Module Placement on BSG-Structure and IC Layout Applications, Proceedings of ICCAD, pages 484-491, 1996
[14]. D. F. Wong, H. W. Leong and C. L. Liu, Sim- ulated Annealing for VLSI Design, Kluwer Academic Press, Boston MA, 1988.
[15]. F.Y. Young and D.F.Wong, How Good are Slicing oorplans, Integration, the VLSI Journal, Vol 23, pages 61-73, 1997.
[16]. Russell C. Eberhart, Yuhui Shi," Comparison Between Genetic Algorithms And Particle Swarm Optimization"
[17]. Dayadi.Lakshmaiah, Dr.M.V.Subramanyam and Dr.K.Satya Prasad "Low power 1 bit-CMOS Full adder based on Reduction of parasitic capacitance through Layout optimization" International Journal of Scientific & Engineering Research Volume 3, Issue 12, pp:1-7, December-2012
[18]. S. Venkatraman*, M. Sundhararajan "|Particle Swarm Optimization Algorithm for VLSI Floorplanning Problem" Journal of Chemical and Pharmaceutical Sciences. 311 JCPS Volume 10 Issue 1, pp:311-317, January - March 2017
[19]. Shivetansh Tickoo and Saurabh Kohli, "A Review on: Floorplanning – Based Design Methodology" International Journal of Advanced Research in Computer Science and Software Engineering, Volume 6, ssue 6, pp:184-188,June 2016.

G.Sujatha " Timing Analysis, Area and Interconnect Length Optimization for VLSI by Using Genetic Algorithm and PSO Algorithm" IOSR Journal of VLSI and Signal Processing (IOSR-JVSP) , vol. 9, no. 1, 2019, pp. 34-41.