STRATEGICAL COMPARISION OF PLACEMENT AND ROUTING IN VLSI DESIGN USING OPTIMIZATION ALGORITHMS

[1]Shaik Karimullah , [2]Dr.K.Murali Babu,[3]Dr.K.Riyazuddin

[1]Assistant Professor,[2]Associate Professor,[3]Associate Professor

[1]ECE Department

[1]Annamcharya Institute Of Technology and sciences, Rajampet, India.

Abstract: As technology advances at a rapid pace in the field of VLSI physical design, the maximum number of transistors and modules must be integrated within the comparatively small range. The aim and goal of physical design placement is generally to reduce the silicon chip area. Hence Optimization approach is considered for placement and routing strategies to achieve best solution. Most of the optimization techniques adopted nowadays are either nature-inspired or evolution-based. These methods of optimization take important attributes from our problem and use them as the variables in their algorithms to produce better solutions. Several algorithms such as Greedy Nearest Neighbor, Simulated Annealing, Ant Colony and Genetic Algorithms have been investigated in the present work to optimize processing blocks. This investigation has been carried out by using MATLAB software with optimization tool box using Travelling salesman problem.

Keywords: Optimization, Placement, Greedy Nearest Neighbor, Simulated Annealing, Ant Colony and Genetic Algorithms.

1. Introduction:

Physical Design is considered as a crucial step for evaluating layout of circuit models and various other specifications like temporal and logical circuit design.As placement and routing processes are of much significance in VLSI design to attain the best solution i.e. less time. Thus the problems in placement and routing may arise due to intermittent issues and non availability of computational blocks present in the VLSI Design issues. These problems may lead to design abruptions and the final outcome may lead to redundant in placement and routing. In order to avoid these related problems in these VLSI systems Optimization approaches are adopted.

Optimization refers to obtaining the best possible solution for a problem by giving group of limitations (or constraints). The Optimization related problems are very common in most of the real time disciplines and domains. In optimization related problems, we have to obtain optimal or near-optimal solutions with respect to some preferred goals. More often than not, we cannot solve problems in one go, but have to follow some method which directs us through for problem solving. Most often, the process of obtaining solution is broken into number of steps which are executed one by one. Commonly the steps are recognized and define the problem, construct and solve the prescribed model, and evaluate and implement the solutions.

Subjectively and objectively, the research work is carried out and analyzed; the results of the subjective approach support the discussion of the nodes I.E Computational blocks plank and a better optimal solution, while the objective approach clarifies the statistical and graphical comparison of existing methods.

2. Significance of Travelling Salesman Problem (TSP)

In this work we wishes to use travelling salesman problem which is the backbone for this research to get optimized solution. This is an optimization problem which is used to acquire shortest path to journey through the given number of cities [8]. Travelling salesman problem states that given a number of cities and with their distances, the traveler is to go through all the cities only once and back to the city from where he started and minimizing the cost of travel. In simple words it can be explained as a salesperson has to make a visit of all places with a condition that he has to go to a place only once and come back to his origin such as to calculate how much time has been spent to visit each place. This path is described as the tour and the path length is prescribed as the cost of the path [9].

3. Optimization Algorithms

3.1 Greedy Nearest Neighbour:

Algorithms intended to optimize issues usually follow sequence of steps, with some number of decisions at each step. The greedy algorithm always chooses the option that at that time seems to be the best. That is, it selects a local optimum value with a hope that the selected choice will lead to a global optimum solution. The Greedy algorithm, however, does not always provide an optimal solution, but it will provide the solution for many problems. The Greedy Algorithm requires step-by-step decisions without returning to the same state. The choice on each step is to enhance the present state in a narrow-minded (myopic) manner without thinking about the worldwide scenario.

3.2 Simulated Annealing:

Simulated annealing is a meta-heuristic technique based on searching locally. Simulated annealing strategy needs a planned representation as well as a neighborhood operator to move to a necessary candidate solution from the present solution. Annealing method allows the process to jump to worse solutions and thus often avoid local sub-optimal solution. Aarts, Laarhoven, and Lenstra outlined to schedule issues one of the first simulated annealing procedures. Simulated annealing is a metallurgical heating method and then cooling it down to a constantly structured state.

3.3 Ant Colony Optimization :

The Ant Colony Optimization (ACO) is a meta-heuristic process of optimization that is influenced by the actual foraging behavior of ants and has become a common method for near-exact optimization. The primary behavior here is the interaction between ants in order to discover the food resources by means of a chemical substance called' pheromone' track set in the route to the nest and the food sources . Path choice relies on how much pheromone is placed on the route that makes the fresh ants follow and add more pheromones. With a high probability of availability of pheromone in shorter food resources paths, the majority of ants impress to follow it. This indirect ant-to-ant communication is called "stigmergy," where the notion of positive feedback is subjected to finding the best possible route, depending on ants ' understanding of resources moved in that direction. Ant Colony's significant benefit over other meta-heuristic algorithms is that the issue instance may dynamically alter.

3.4 Genetic Algorithm:

The Genetic Algorithm (GA) is a search-based heuristic using natural biology evolution procedures established by John Holland in 1970 and Ingo Rechenberg in Germany. Although the genetic algorithms have been studied for many years, application for an appropriate issue is often seen as an art with an effective heuristic design. The implementation of genetic algorithms is often discovered in literature dedicated to rather easy issues. The fundamental implementation of a genetic algorithm to easy issues has frequently generated reasonable outcomes, whereas the implementation of genetic algorithms often results in bad performance for the bigger issues. This is due in large part to the natural features of genetic search operations and also to the interactions between a genetic representation and genetic operators.

4. Results and Discussion:

4.1 Greedy Nearest Neighbour Results:

The Placement and routing process using Greedy Nearest Neighbor algorithm for VLSI system Design has been investigated and simulation results of 45 blocks is shown in figure 4.1.

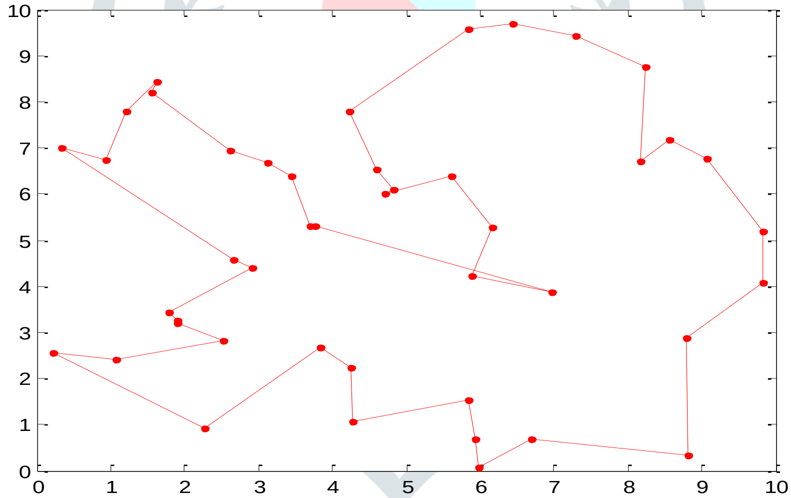

Figure 4.1. Greedy NN Placement and Routing of 45 blocks

4.2 Simulated Annealing Results

The Greedy Nearest Neighbor strategy described previous provides the best solution by gradually choosing the best possible block from its intrinsic property in terms of plan duration. Unlike the Greedy Nearest Neighbor strategy, the simulated annealing attempts to free itself from the local optimum by randomly selecting a fresh solution from the current solution's group of neighbors. The fresh solution acquired is acceptable if it is not the best solution. However, a poorer quality solution that is dependent on the down turn of the cost function can also be adopted with a certain likelihood.

The prospective problems in the Simulated Annealing method depended on calculation time, acceptance probability, local optima, and cooling schedule. The cooling rate should be slow for better outcomes, here it has been held at 0.97. Thus, as the blocks increased, the calculation time improved gradually. The iterations must be increased if the initial temperature is high, So the calculation time improves as well. The likelihood of acceptance depends on the temperature and the cooling schedule, the original temperature of 2000 was regarded and the highest iterations were restricted to 2000, and the calculation time improved as the nodes increased from 20 to 100.

If we add more limitations to the issue, such as disintegrating defective nodes and electronic module defective information, more local optima will be developed. With this issue, the probability of finding a solution in one of the local optima is reduced and the chance of achieving the best results is reduced, i.e., global optima. As we ran the timetable pattern gradually changed owing to the above issue and it is shown in figure 4.2.

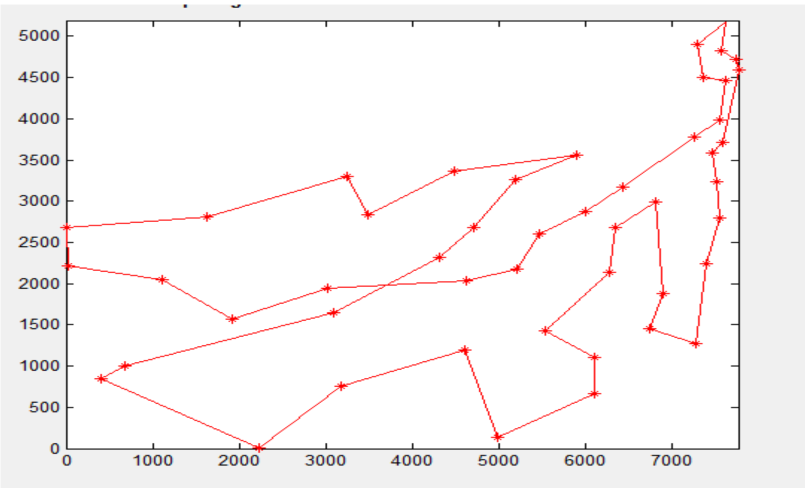

Figure 4.2. SA placement and routing for 45 blocks

4.3 Ant Colony Optimization Results :

For VLSI design apps with a big amount of blocks, positioning in simulated annealing method generates countless local optima. This gets worse as the best solution for the method of cooling gradually becomes focused on the present solution. Since this requires large number of iterations, the placement and routing process becomes slow to obtain best solution. Because of the above-mentioned limitations, the output of SA is obviously restricted. In this study job with TSP, experimental studies based on ACO have therefore been carried out.

Here the parameters are set to the following values: α=1(effect of ants' sight), β= 4 (trace's effect), e =.15 (evaporation coefficient). The highest iteration is set at 100 and m (ants number) is set at 10. The job allocation in ACO is based on the idea of how the simulated horns should select the next block depending on the quantity of pheromone in a path and the distance to the next block. The figure 4.3 presents the results obtained for this method and the best solution.

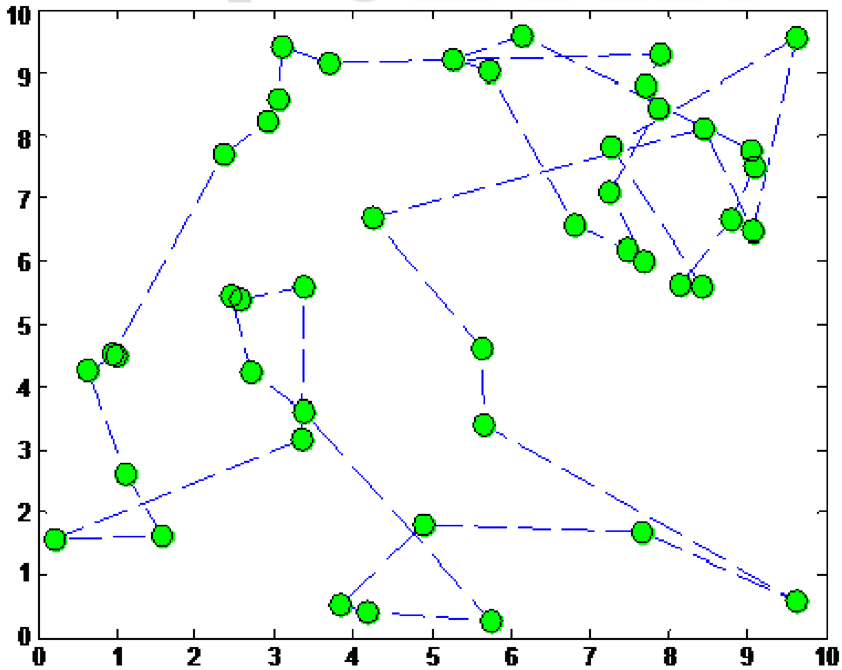

Figure 4.3. ACO based placement and routing of 45 blocks

4.4 Genetic Algorithm Results :

As the amount of ants in ACO subsequently increases the operating time, additional findings may not be good. The amount of ants must therefore be fixed at a lesser level. On the other hand, lower the number of ants will not yield a good solution even though better running times are achieved. Unlike ACO, Genetic Algorithm produces a new generation of solution better than the previous solution and this process continues until a stopping condition is met. For VLSI design systems, it is required to recognize for limited resources utility the algorithm has to run several times to yield optimal results, for which ACO may not be better choice.In this job, population size and matrix from point to point distances / costs are the primary concern inputs in order to fix the coding. The findings achieved show that the planning time is reduced by roughly 15-60% compared to prior algorithms. The Genetic Algorithm slows down when big population size is regarded, whereas a tiny size leads to a tiny search space. This is due to the crossover and mutation operators inherent with Genetic Algorithm

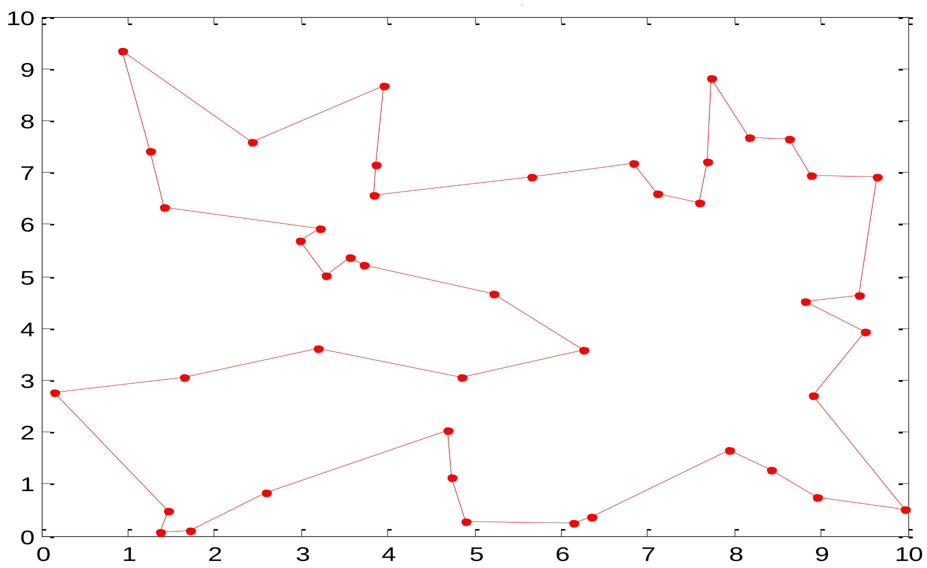

Figure 4.4. Genetic Algorithm placement and routing for 45 blocks

Objective Analysis :

Table 1 Comparison of algorithms based on best solution

Optimization Methods	Greedy Nearest Neighbour	Simulated Annealing	Ant Colony Optimization	Genetic Algorithm
Best Solution for 45 blocks (in Secs)	61.8	20006.181	159.50	56.48

The best solution values for 45 blocks are provided in table 1 for comparison. From the table we can conclude that it is because of clustering that the genetic algorithm provides better outcomes and the outcome is better than other algorithms. We can see from the table Simulated annealing method is quite slow because if the best solution is further lowered, the cooling coefficient is maintained at 0.97 can be obtained earlier but the possibility of landing in optimal solution is less. The Greedy Nearest Neighbour gives better value than Simulated Annealing and Ant Colony technique; this is due to sorting in short distances but no guarantee of getting optimum solution.

5. Conclusion:

While greedy strategy is easy and quick, but it may not yield to optimum value, multi-node scheduling is also not feasible. In order to get the best solution, the greedy nearest neighbor visits all possible blocks. In Simulated Annealing the Cooling rate plays a vital role, if it is on higher note achieving of best solution takes more computation time else it moves towards Greedy nature. When it comes to the coefficient of evaporation of the Ant Colony method and the variation of the sight of ants indulges in obtaining the best solution, whenever the coefficient of evaporation is increased or the sight of ants is decreased, it results in more local optimum making it difficult to obtain the optimal global solution. Genetic algorithm has obtained better outcomes for optimization in comparison.

References:

1. K.Sivasubramanian, and K.B.Jayanthi, "Music-Inspired Harmony Search Algorithm for Fixed-Outline Non-Slicing VLSI Floorplanning," World Academy of Science, Engineering and Technology International Journal of Electrical, Computer, Energetic, Electronic and Communication Engineering, vol. 9, no.6, 2015.
2. Tung-Chieh Chen, and Yao-Wen Chang, "Modern floorplanning based on fast simulated annealing," Proceedings of the 2005 international symposium on Physical design, April 03-06, 2005, San Francisco, California, USA.
3. Gracia Nirmala Rani., Rajaram .S "Analysis & Design of VLSI Floorplanning Algorithms for Nano-Circuits" IntJAdv Engg Tech/Vol. VII/Issue I/Jan.March.,2016/527-532.
4. Fernando, Pradeep Ruben, "Genetic algorithm based design and optimization of VLSI ASICs and reconfigurable hardware" (2009). Graduate Theses and Dissertations.
5. Venkatraman.S, Dr.M.Sundhararajan, " Optimization of VLSI floorplanning using genetic algorithm" Journal of Chemical and Pharmaceutical Sciences, JCPS Volume 10 Issue 1, January - March 2017,pp 311-316.

6. Venkatraman.S, Dr.M.Sundhararajan, "Particle swarm optimization algorithm for VLSI floorplanning problem" Journal of Chemical and Pharmaceutical Sciences, JCPS Volume 10 Issue 1, January - March 2017,pp 311-316.
7. T. Padmapriya and V. Saminadan, "Improving Throughput for Downlink Multi user MIMO-LTE Advanced Networks using SINR approximation and Hierarchical CSI feedback", International Journal of Mobile Design Network and Innovation- Inderscience Publisher, ISSN : 1744-2850 vol. 6, no.1, pp. 14-23, May 2015.
8. V. Vegh, G.K. Pierens, Q.M. Tieng, A variant of differential evolution for discrete optimization problems requiring mutually distinct variables. Int. J. Innov. Comput. Inf. Control 7(2), 897–914 (2011)
9. Shaik Karimullah, Dr. D.Vishnuvardhan "**A Review Paper On Optimization Of Placement And Routing Techniques**"at *NC'e-TIMES # 1.0 IN2018, IJET* ISSN: 2395-1303.
10. Shaik Karimullah , Abdul Rahim B C.Ushasree**"Design And Verification Of Online Bist For Different Word Sizes Of Memories"Ijastr,** Issue 4 volume 4, July-August 2014, ISSN 2249-9954.