

IC Floorplanning Optimization using Simulated Annealing with Order-based Representation

Rajendra Bahadur Singh

School of Information & Communication Technology, Gautam Buddha University, Greater Noida, India
E-mail: rbs2006vlsi@gmail.com

Anurag Singh Baghel

School of Information & Communication Technology, Gautam Buddha University, Greater Noida, India
E-mail: anuragsbaghel@gmail.com

Received: 16 February 2020; Revised: 13 May 2020; Accepted: 16 October 2020; Published: 08 April 2021

Abstract: Integrated Circuits (IC) floorplanning is an important step in the integrated circuit physical design; it influences the area, wire-length, delay etc of an IC. In this paper, Order Based (OB) representation has been proposed for fixed outline floorplan with Simulated Annealing (SA) algorithm. To optimize the IC floorplan, two physical quantities have been considered such as area, and wire-length for hard IP modules. Optimization of the IC floorplan works in two phases. In the first phase, floorplans are constructed by proposed representation without any overlapping among the modules. In the second phase, Simulated Annealing algorithm explores the packing of all modules in floorplan to find better optimal performances i.e. area and wire-length. The Experimental results on Microelectronic Center of North Carolina benchmark circuits show that our proposed representation with SA algorithm performs better for area and wire-length optimization than the other methods. The results are compared with the solutions derived from other algorithms. The significance of this research work is improvement in optimized area and wire-length for modern IC.

Index Terms: Floorplanning, simulated annealing, order based representation, MCNC benchmark, NP-hard.

1. Introduction

Integrated Circuit (IC) industry has advanced drastically after 1990. The feature sizes of transistors are downscaled from micrometer to nanometer so the circuit complexity is increasing rapidly in terms of number of components on ICs. Due to high complexity of modern ICs design, EDA tools play a very important factor in designing a high-performance IC. IC physical design process includes circuit partitioning, floorplanning, placement, routing and compaction. In the future, the rapid growth in IC circuit will depend on the development of physical design automation tools [1, 2].

In the IC physical design [3] process of chip, floorplanning [4-6] is an important step. It establishes the ground work for a good layout. It also gives an early prediction about architectural design, determines chip area, wire-length, and delay. It is relevant during architectural design because component locations affect the wire-length delay which finally impacts the clock frequency and delay of circuit. Basically, it is used to determine the position and orientation of modules on a chip without overlapping of two modules so that minimum chip area will occupy which finally reduces the silicon cost. Computationally, floorplanning problem is an NP-hard problem so there is no any exact algorithm to optimize it. Once numbers of modules are increases, complexity and computational time exponentially increases hence to optimize for larger number of modules on chip, there is meta-heuristic algorithms. These algorithms give better optimal result in reasonable time.

Basically, floorplanning determines the size, shapes, and locations of each module before fabrication of ICs. Consequently, it estimates total chip area, wire-length and delays. Many researchers have suggested various algorithms to optimize it. The fundamental problem of IC floorplanning is its representation. A clever representation scheme reduces the size of search space.

2. Related Works

Many researchers have been applied Genetic Algorithms[7-9], Simulated Annealing[10,11], Particle Swarm Optimization (PSO) Algorithm[6,12,13,25] and Binary PSO [24] for the optimization of the IC floorplanning problems in last four decades using different representation scheme.

Tang et al. [7] proposed GA algorithm with O-tree representation. The proposed algorithm applied on MCNC

benchmark circuits and simulated. This results can consistently produce better results than the deterministic algorithms. Valenzuela and Pearl Y.[8] used a novel encoding to breed normalized postfix expressions for macro cell placement and area optimization for soft modules using GA. Jianli Chen and Wenxing Zhu discussed a HGA [14] with the B-tree representation to optimize the VLSI floorplanning problem for a non-slicing and hard IP module. A Memetic Algorithm(MA) [15] has been proposed in this article. A non-slicing floorplan for hard IP module using MA is optimized to find better area and wire-length. Jens Lienig [16] presented a parallel GA to optimize the VLSI channel and switchbox routing problems. This algorithm optimizes both physical constraints (length of nets, number of vias) and crosstalk (delay due to coupled capacitance). S.Nakaya presented an Adaptive GA[17] to solve the floorplanning problem in VLSI layout design using the sequence-pair[23] representation. Anurag et al. [24] proposed BPSO algorithm to optimize the IC floorplanning. In this work, optimization of the hard IP MCNC benchmark module has been simulated. The optimized area and the wire-length have been compared with another algorithm.

Henrik Esbensen[9] proposed a genetic algorithm for the macro cell placement problem. In ref [5] discussed a Hybrid Simulated Annealing (HSA) algorithm has been proposed for a non-slicing VLSI floorplanning problem using the B-tree representation. In ref [18], the area and the wire-length have been optimized for fixed outline constraint. Simulated Annealing (SA) algorithm based heuristic, namely Simulated Spheroidizing Annealing algorithm [19] has been proposed. It has been developed and improvements in the proposed heuristic algorithm are also suggested to improve its performance. Chen, Tung-Chieh et al [20] proposed a fast simulated annealing for optimization of VLSI floorplanning with B-tree representation.

The IC floorplanning is an NP-hard problem [24]. There is no polynomial time exact algorithm for this problem to give a quick solution in a reasonable time. A fundamental problem in the IC floorplanning is representation because it determines the size of the search space and the complexity of the transformation between a representation and its corresponding floorplan. IC faces enormous challenges in both technology and physical design. Typical problems in ICs are reliability, alignment accuracy, reuse of existing IP blocks, testability, thermal issues and optimization of the area and the wire length. In mobile devices, with the increasing application power consumption reduction has become highly critical for chip designers.

The total interconnect in any IC is large. During high speed interconnect analysis; interconnect effects like signal delay, crosstalk and distortion become more significant. Almost all the solutions proposed so far, of the problem, through heuristics and meta-heuristics, either explored constrained search space or used representations with redundancies.

In this paper, an Order Based (OB) representation scheme has been proposed to represent the IC floorplanning problem for the MCNC benchmark circuits. Finally, it has been optimized using simulated annealing (SA) algorithm. The floorplanning problem is implemented for MCNC benchmark circuits to especially optimize the area and interconnect of chip. Proposed OB representation with SA algorithm has been compared to other algorithm with different representation results for same benchmark circuits.

3. Problem Statement

IC floorplanning is to arrange the modules into given fixed outline region as shown in figure 1. The goal of floorplanning is to determine the positions and dimensions of each module on IC to optimize the circuit performance such that all the modules are enveloped in the rectangular floorplan.

The inputs for a floorplanning problem are given as follows:

- 1) A set of m rectangular modules $S = \{b_1, b_2, b_3, \dots, b_m\}$ with area a_i , where $1 \leq i \leq m$.
- 2) An interconnection matrix $C_{n \times n} = [c_{ij}]$, $1 \leq i, j \leq n$, where c_{ij} indicates the connectivity between modules i and j .
- 3) A list of widths, heights $(w_1, h_1), (w_2, h_2), \dots, (w_i, h_i)$, where w_i and h_i are the width and height of the module i respectively ($1 \leq i \leq m$).

The outputs of floorplan are lower left (x, y) coordinates of each modules, width and heights of each module on the final floorplan as shown in figure 2, which has to satisfy the following requirements:

- No two modules overlap each other.
- Lower left coordinates (x_i, y_i) , width w_i and height h_i for each modules such that area, $a_i = w_i \times h_i$, $1 \leq i \leq m$.

All the modules are enveloped in the floorplan and the total area, wirelength, and other performance metrics are optimized. The objective of optimization is to minimize the total area of fixed outline floorplan, under given constraints.

There are two constraints during the IC floorplanning problem:

- 1) All modules must be packed inside given fixed outline region.

Assume that W_0 and H_0 are width and height of fixed outline floorplan, respectively.

2) No modules should overlap each other.

There are two kinds of modules i.e., hard and soft modules can be used in the IC floorplanning problem. In hard modules, its area and dimensions are fixed. In soft modules, its area is fixed but dimensions are not fixed.

3.1. Dead Space

The area which not covered by any module in the fixed floorplan region, called dead space [21] or white space as shown in figure 2. Hence dead space can be reduced on chip if optimized area is reduced which affect the cost of silicon wafer. The percentage of dead space is calculated using equation 1.

$$\text{Dead space} = \frac{\text{optimal FP Area} - \text{sum of all modules area}}{\text{optimal FP Area}} * 100 \tag{1}$$

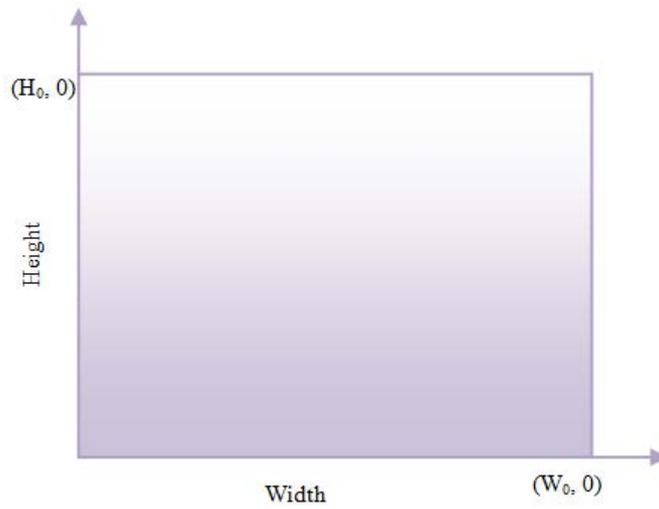


Fig.1. Fixed outline floorplan region with width W_0 and height H_0

The dead space of a floorplan is the area which is not covered by the modules; figure 2 shows the floorplan which consists of six rectangular modules. In this paper, optimization of the IC floorplanning for hard modules has been done.

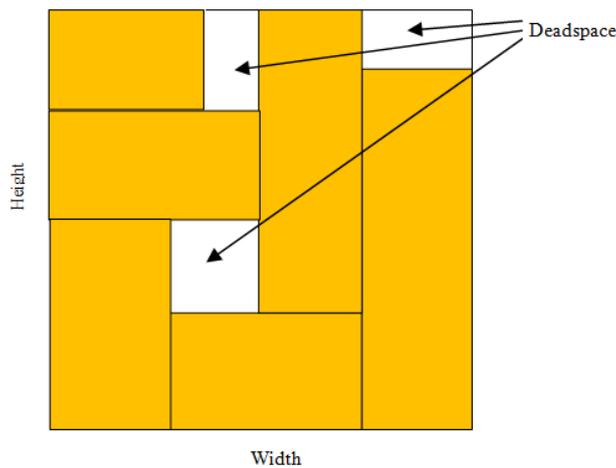


Fig.2. Fixed outline floorplan

3.2. Cost Function

When area and wire-length both are minimized simultaneously then cost function can be written as

$$\text{cost} = \alpha \cdot \frac{A}{A_{norm}} + \beta \cdot \frac{Q}{Q_{norm}} \tag{2}$$

Where, A = represent floorplan area after packing of all modules in given fixed outline floorplan, Q =represent total interconnect.

A_{norm} and Q_{norm} are calculated by averaging area and wire-length of floorplan, respectively. The constant weights are α and β . $0 < \alpha < 1$ and $\beta = 1 - \alpha$.

4. Research Method

In this section, proposed order based (OB) representation for the IC floorplanning has been explained and SA algorithm has been also explained to find the better optimal result.

4.1. Order Based Representation

In OB representation method, first randomly generate the sequence of the modules, after this all modules are placed in a given fixed outline region as follows-

1. The first module of the sequence is situated at the bottom left corner.
2. Second, third, and other modules are placed in a row from left to right manner.
3. If any module of the sequence to be placed exceeds the width of a fixed outline region, a new row is started for putting the rest modules as per order in a given sequence.
4. Step-iii is repeated until all modules are not placed in a given fixed outline region.

4.2. Order Based To floorplan representation

Let there are eight modules, and their random sequence in OB representation method is [4, 5, 6, 7, 2, 3, 1, 8]. All modules' shapes are shown in figure 3.

This representation is proposed for modern IC floorplan, as modern IC floorplan has a fixed outline region. Width and height of fixed outline floorplan are fixed, as shown in figure 1. All modules are placed as per sequence is given in OB representation from left {start from (0, 0) coordinate} to the right ($W_0, 0$) in fixed outline region}. If the next module is overlap with the width of a fixed outline, then that module and rest unplaced module will start to place from (0, H_1) coordinates to (W_0, H_1). Where H_1 = maximum (height of modules placed initially between (0, 0) coordinate to ($W_0, 0$)). If all modules are placed in a given fixed outline region, then the final floorplan layout is obtained. If still few modules are not placed then start to place them from coordinate (0, H_2) to (W_0, H_2), where $H_2 = H_1 +$ maximum (height of modules placed between coordinates (0, H_1) to (W_0, H_1)).

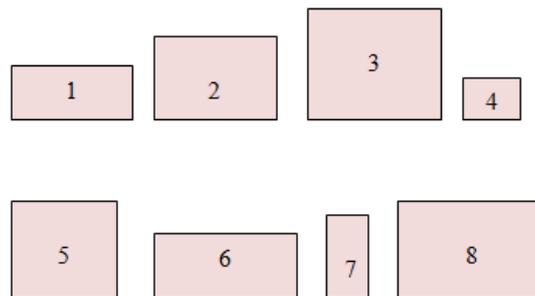


Fig.3. Eight modules with different dimension

Table 1. Dimension of modules

Module No.	Width	Height
1	3	2
2	3	3
3	3	4
4	1	1
5	3	3
6	4	2
7	1	3
8	4	3

Table 2. Lower left coordinates of each module using OB representation

Module No.	x-coordinate	y-coordinate
1	$x_1 = x_3 + w_3 = 3 + 3 = 6$	$y_1 = 3$
2	$x_2 = 0$	$y_2 = \max(h_4, h_5, h_6, h_7) = 3$
3	$x_3 = x_2 + w_2 = 0 + 3 = 3$	$y_3 = 3$
4	$x_4 = 0$	$y_4 = 0$
5	$x_5 = x_4 + w_4 = 0 + 1 = 1$	$y_5 = 0$
6	$x_6 = x_5 + w_5 = 1 + 3 = 4$	$y_6 = 0$
7	$x_7 = x_6 + w_6 = 4 + 1 = 5$	$y_7 = 0$
8	$x_8 = 0$	$y_8 = y_2 + \max(h_2, h_3, h_1) = 3 + \max(3, 4, 2) = 3 + 4 = 7$

All modules are placed inside a fixed outline region using OB method. Figure 4 show all modules placement in a fixed outline region using OB approach. If any module overlaps fixed outline region, then that module and rest unplaced modules are placed at the next level, and this process will be repeated until all modules are not placed in the fixed outline region. Table 2 shows the lower left coordinate of each module using the OB method. According to these lower left coordinates all modules are placed in a fixed outline region as shown in figure 4. The current optimized solution can be further improved after applying the perturbation.

- Swapping a random pair of modules in the sequence.
- Rotating a randomly selected module by 90 degrees.

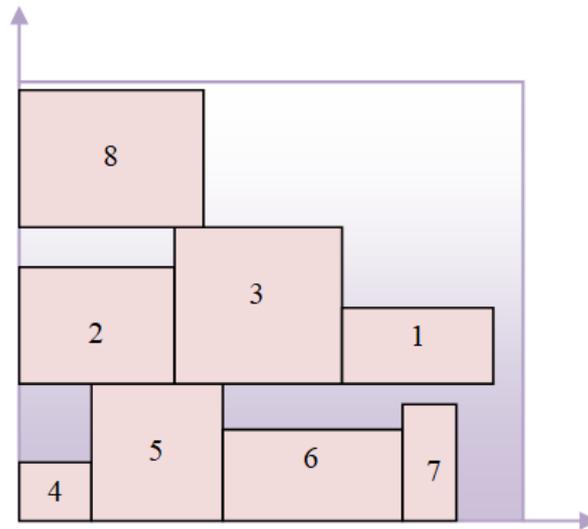


Fig.4. Modules packing in fixed outline region.

4.3. Simulated Annealing Algorithm

Simulated Annealing (SA) is inspired by an analogy between the physical Annealing of solids (crystals) and combinatorial enhancement. In the physical annealing process, a solid is first melted and then cooled very slowly, spending a long time at low temperatures, to obtain a perfect lattice structure corresponding to a minimum energy state. SA transfers this process to local search algorithms for combinatorial enhancement problems. It does so by associating the set of solutions of the problem attacked with the states of the physical system, the objective function with the physical energy of the solid, and the optimal solutions with the minimum energy states.

Simulated Annealing [5], [10, 11], [19-21] is a widely known algorithm, and it can be effectively applied in the IC physical design [1-3] and other fields.

Before using it in the IC floorplanning for optimal results, the needs for concern are four ingredients-

1. Solution Space: Solution space is searching space. If the solution space is large then it will take high computation time. If solution space is too small, then it will search within short computation time, and it may not find the right solution. Hence always choose a smart value of solution space so that chances of the good solution get.
2. Perturbation: This ingredient used to find the neighborhood solution by perturbing the existing solution.

3. Cost function: Typical, this is used to decide the cost of the problem. Here area and wire-length parameters are used in a cost function; also, some other parameters such as noise, power, and temperature can also be part of cost function.
4. Annealing Structure: This ingredient relates in initial temperature, termination temperature, cooling rate, and the number of iterations at each temperature

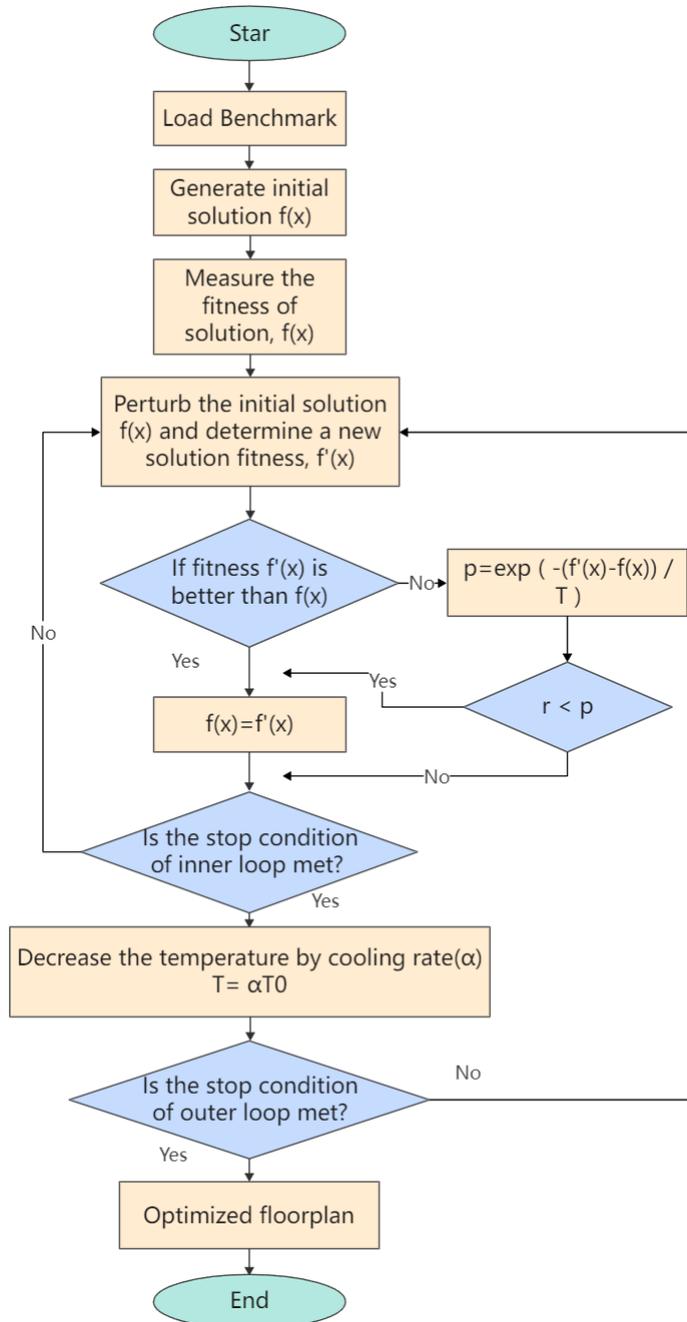


Fig.5. Flow chart of SA for IC floorplanning

4.4. Parameter Settings

There are few parameters in the SA algorithm, which are essential and deciding parameters in the optimization of area and wire-length. Important parameters in the SA algorithm are initial temperature, termination temperature, cooling rate, and the maximum number of trials at each temperature.

The parameters of SA algorithm were set as follows: initial temperature=1000, cooling rate(α) = 0.95, termination temperature= 0.01.

5. Performance Analysis and Simulation Results

The Simulated Annealing algorithm is implemented in MATLAB programming language, and the experiment was executed on an Intel Core 2 Duo (2.4 GHz) 4GB RAM machine running with windows 2007. The SA algorithm is simulated for MCNC (Microelectronic Centre of North Carolina) benchmark. Table 3 shows the details of MCNC benchmark circuits. Column two presents the name of the benchmark circuit. The third column of table 3 shows the number of modules within the benchmark circuit. The fourth column displays the number of nets connecting the modules within the circuits. The fifth column presents the total number of pins within the circuit, and the sixth column shows the total area of all modules. The optimization of IC floorplanning problem tested for hard IP and final optimized results with OB representation are compared with other approaches.

The parameters of the SA algorithm are set as follows. (1) The cooling factor is 0.95. (2) Initial temperature=1000. (3) The algorithm terminates if the temperature reaches 0.01. Given SA algorithm with OB representation, it is considered as a failure if it violates any fixed-outline constraint. Each benchmark is tested 20 times and minimum and average value of optimized area and optimized wire-length is given in table-4 for different constant weights. If the number of modules is increased in MCNC benchmarks, the search space increases exponentially with the increase of the number of modules in IC floor plan problems. Consequently, simulation time to get an optimized result also increases. In Table 5, the simulation time of MCNC benchmarks has been shown.

The experimental results for the OB representation using SA algorithm are presented in table 4. Three sets of weights for α and β were used in the experiment: $\{\alpha =0.5, \beta=0.5\}$, $\{\alpha =1, \beta=0\}$, $\{\alpha=0, \beta=1\}$. Hard IP modules are simulated for three sets of α and β . For the first set, we recorded the minimum and average value of the optimized area and wire-length. When using the second set of weights, we noted the minimum and average optimized area and wire-length. Table 4 shows that for the second set of weight, the area is better optimized than wire-length, and for the third set of weights, wire-length is better optimized than area.

Consequently, in the optimization of multi-objective problem [22], the weight value is more important. These weights values decide which parameter will optimize better than other parameters. From table 4, it is clear that if $\alpha > \beta$ then the area is optimized better than wire-length and vice-versa.

Table 3. The details of MCNC benchmark circuits

Problems	Modules	Nets	Terminals	Pins	Total area of all modules
apte	9	97	73	287	46.56
xerox	10	203	2	698	19.35
Hp	11	83	45	309	8.83
ami33	33	123	42	522	1.15

Table 4. Minimum / Average optimized area and wire-length using SA with OB representation for hard IP modules

Circuits Hard IP	$\alpha=0.5, \beta=0.5$				$\alpha=1, \beta=0$				$\alpha=0, \beta=1$			
	Area (mm ²)		Wire (mm)		Area (mm ²)		Wire (mm)		Area (mm ²)		Wire (mm)	
	Min.	Avg.	Min.	Avg.	Min.	Avg.	Min.	Avg.	Min.	Avg.	Min.	Avg.
Apte	47.56	48.3	337	378	47.07	47.89	409	460	48.43	56.45	302	351
Xerox	20.36	21.48	513	570	20.26	20.56	624	787	21.6	24.47	498	547
Hp	9.2	9.83	213	285	9.2	9.4	314	344	11.41	13.48	190	195
Ami33	1.30	1.47	89	101	1.28	1.38	105	123	1.68	2.0	85	90

Table 5. Simulation time of MCNC benchmark circuits

Benchmark circuits	Apte	Xerox	HP	Ami33
Simulation time(S)	89	124	62	294

Table 6. Comparison of results ($\alpha=0.5$ and $\beta=0.5$)

Algorithm	Apte		Xerox		HP		Ami33	
	Area	Wire	Area	Wire	Area	Wire	Area	Wire
HPSOHS[13]	47.44	263	20.1	462	9.4	152	1.23	49.28
GA-Otree[7]	46.9	191	20.2	500	9.85	68.3	1.29	46.2
HPSOFF[6]	47.44	263	20.1	481	9.46	153	1.23	57.2
PSO[24]	47.31	263	20.2	477	9.5	136	1.28	69
Proposed	47.56	337	20.3	513	9.2	213	1.30	89

There is a tradeoff between two optimization objectives, i.e. area, and total interconnect for IC floorplanning. SA algorithm with two different representations is tested on the MCNC benchmark circuit for both hard IP modules and soft IP modules with different weight parameter values of α and β . Every benchmark is tested 20 times then the best and average values are recorded. For the value of $\alpha=0$ and $\beta=1$, the objective function total interconnect was optimized and for the value of $\alpha=1$ and $\beta=0$, the objective function area of IC was optimized. If both $\alpha=0.5$ and $\beta=0.5$, then equal importance was given to area and total wire length. Table-6 shows the comparison of proposed representation scheme with other algorithm. Proposed results give competitive optimized area and optimized wire-length result with other existing algorithm results.

6. Conclusion and Future Scope

This paper has proposed an Order Based (OB) representation to tackle IC floorplanning problem using simulated annealing algorithm. We conclude that optimization of the IC floorplanning using simulated annealing with OB representation gives better result for MCNC benchmarks. From table 6, it is obvious that it gives competitive optimized area, and wire-length compares to other algorithms for the MCNC benchmark circuits. The proposed method is implemented in MATLAB programming. The major contribution of this work is the design and implementation of a tool that could produce IC floorplanning solution using OB representation with SA algorithm. The novelty of this work is to give an optimal solution in reasonable time and can be used to develop a vital tool that product better result compares to other stochastic algorithms.

In this work, optimizations of power and thermal issues have not been covered. So further research in the IC floorplanning design problem could be minimization of power and thermal issues. Many issues are still open for further research in this area. Currently, the proposed framework considers modules/blocks with two dimensions only. Blocks with three-dimension (3D floorplanning) can be used for further work. The next research direction may focus on temperature aware floorplanning. This work can be enhanced by using other heuristic optimization techniques and representations to solve the floorplanning problems to find better optimal results.

References

- [1] C. J. Alpert and D. P. Mehta, Handbook of algorithm for physical design automation", Auerbach Publications, pp. 139-142, 2008.
- [2] S. K. Lim, "Practical problems in VLSI physical design automation," Springer Science & Business Media, 2008.
- [3] Sarrafzadeh, Majid, and C. K. Wong. An introduction to VLSI physical design. McGraw-Hill Higher Education, 1996.
- [4] Singh, Rajendra Bahadur, Anurag Singh Baghel, and Ayush Agarwal. "A review on VLSI floorplanning optimization using metaheuristic algorithms." Electrical, Electronics, and Optimization Techniques (ICEEOT), International Conference on. IEEE, 2016, pp. 4198 – 4202.
- [5] Jianli Chen, Wenxing Zhu, and M.M.Ali, "A Hybrid Simulated Annealing Algorithm for Nonslicing VLSI Floor-planning," IEEE Trans.on Systems,Man and Cybernetics, Part C, vol.41, No.4, pp. 544–553, July 2011.
- [6] Sivaranjani, P., and A. Senthil Kumar. "Hybrid Particle Swarm Optimization-Firefly algorithm (HPSOFF) for combinatorial optimization of non-slicing VLSI floorplanning." Journal of Intelligent & Fuzzy Systems 32.1 (2017): 661-669.
- [7] Tang, Maolin, and Alvin Sebastian. "A genetic algorithm for VLSI floorplanning using O-tree representation." In Workshops on Applications of Evolutionary Computation, pp. 215-224. Springer, Berlin, Heidelberg, 2005.
- [8] Valenzuela, Christine L., and Pearl Y. Wang. "VLSI placement and area optimization using a genetic algorithm to breed normalized postfix expressions." IEEE Transactions on Evolutionary Computation 6.4 (2002): 390-401.
- [9] Esbensen, Henrik. "A genetic algorithm for macro cell placement." In Proceedings of the conference on European design automation, IEEE Computer Society Press, 1992 pp. 52-57.
- [10] Zou, Dexuan, et al. "A memory-based simulated annealing algorithm and a new auxiliary function for the fixed-outline floorplanning with soft blocks." Journal of Ambient Intelligence and Humanized computing (2017) pp.1-12.
- [11] Rajendra Bahadur Singh, Anurag Singh Baghel, "Simulated Annealing algorithm for VLSI floorplanning for soft blocks", International Journal of Computer Science & Information Security (IJCSIS), April 2018.
- [12] Tsung-Ying Sun, Sheng-Ta Hsieh, Hsiang-Min Wang, and Cheng-Wei Lin, "Floorplanning Based on Particle Swarm Optimization," Proceedings of the 2006 Emerging VLSI Technologies and Architectures (ISVLSI'06), 2006.
- [13] Paramasivam, Sivaranjani, et al. "Optimization of Thermal Aware VLSI Non-Slicing Floorplanning Using Hybrid Particle Swarm Optimization Algorithm-Harmony Search Algorithm." Circuits and Systems 7.05 (2016): 562.

- [14] Jianli Chen and Wenxing Zhu, "A Hybrid Genetic Algorithm For VLSI Floor-planning," IEEE Transaction on Intelligent Computing and Intelligent Systems , vol.2, pp.128-132, 29-31 Oct. 2010.
- [15] Tang, Maolin, and Xin Yao. "A memetic algorithm for VLSI floorplanning." IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics) 37.1 (2007): 62-69.
- [16] Lienig, Jens. "A parallel genetic algorithm for performance-driven VLSI routing." IEEE Transactions on Evolutionary Computation 1.1 (1997): 29-39.
- [17] Nakaya, Shingo, Tetsushi Koide, and Si Wakabayashi. "An adaptive genetic algorithm for VLSI floorplanning based on sequence-pair." Circuits and Systems, 2000. Proceedings. ISCAS 2000 Geneva. The IEEE International Symposium on. Vol. 3. IEEE, 2000.
- [18] Anand, S., S. Saravanasankar, and P. Subbaraj. "A multiobjective optimization tool for Very Large Scale Integrated nonslicing floorplanning." International Journal of Circuit Theory and Applications 41.9 (2013): 904-923.
- [19] Anand, S., S. Saravanasankar, and P. Subbaraj. "Customized simulated annealing based decision algorithms for combinatorial optimization in VLSI floorplanning problem." Computational Optimization and Applications 52.3 (2012): 667-689.
- [20] Chen, Tung-Chieh, and Yao-Wen Chang. "Modern floorplanning based on B/sup*/-tree and fast simulated annealing." IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems 25.4 (2006): 637-650.
- [21] Singh, Rajendra Bahadur, and Anurag Singh Baghel. "Dead space reduction of floorplan using simulated annealing algorithm." In 2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS), pp. 2108-2112. IEEE, 2017.
- [22] Chen G L Guo W Z, Tu X Z, Chen H W. "An Improved Genetic Algorithm for Multi-objective Optimization". ISICA'2005: Progress in Intelligent Computation and its Applications: pp. 204-210, 2005.
- [23] H. Murata, K. Fujiyoshi, and Y. Kajitani, "VLSI module placement based on rectangle-packing by the sequence-pair," IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems, vol. 15, No. 12, 1996, pp. 1518-1524.
- [24] Singh, Rajendra B., Anurag Singh Baghel, and Arun Solanki. "A Binary Particle Swarm Optimization for IC Floorplanning." Recent Advances in Computer Science and Communications (Formerly: Recent Patents on Computer Science) 13, no. 1 (2020): 13-21.
- [25] Chen, Guolong, Wenzhong Guo, and Yuzhong Chen. "A PSO-based intelligent decision algorithm for VLSI floorplanning." Soft Computing 14.12 (2010): 1329-1337.

Authors' Profiles



Rajendra Bahadur Singh has completed B.Tech, M.Tech and Ph.D. He is working as Research/ Faculty Associate in the department of Electronics & Communication Engineering, Gautam Buddha University, India. His areas of interest are VLSI design, IC physical design using soft computing, Micro-electronics. He has supervised more than 40 M.Tech dissertations.



Anurag Singh Baghel has completed his M.Tech (Electronics) in 2000 and D.Phil in 2010 both from University of Allahabad, Allahabad, India. He served as Lecturer (Electronics) from 2004 to 2011 in Banasthali University, India and since then he is working as Assistant Professor (Computer Science) in Gautam Buddha University, Greater Noida, India. His areas of interest are Meta-heuristics and applications, Software Engineering, and Big Data. He has published more than 60 research publications in various journals and international conferences. He has supervised more than 65 M.Tech dissertations and Ph.D. scholars.

How to cite this paper: Rajendra Bahadur Singh, Anurag Singh Baghel, "IC Floorplanning Optimization using Simulated Annealing with Order-Based Representation", International Journal of Intelligent Systems and Applications(IJISA), Vol.13, No.2, pp.62-70, 2021. DOI: 10.5815/ijisa.2021.02.05