ISSN: 0975-766X
CODEN: IJPTFI
Available through Online *Review Article*
www.ijptonline.com

A REVIEW ON VLSI FLOORPLANNING ALGORITHMS

B. Srinivasan* and S. Saravanan
Department of Information Technology, School of Computing, SASTRA University, Thanjavur, Tamilnadu, India.
Email: srinivasan@core.sastra.edu

Received on 22-04-2016 *Accepted on 18-05-2016*

Abstract

Floorplanning is an important phase in the automation of Very Large Scale Integration (VLSI) circuit design, which generates a plan for arranging *'N'* numbers of smaller rectangular electronics modules into the larger rectangular chip which optimize the design metric. The main two objectives in optimization of VLSI Physical Design are minimizing the die area and minimizing the interconnect length. The other objectives such as minimizing power, increasing the performance of chip in terms of speed, reducing manufacturing cost, minimizing the critical path length and maximizing the routability are highly depends on the area and wire length minimization. This paper presents an overview of up-to-date account on various floorplanning algorithms.

Keywords: VLSI physical design, Floorplanning, area optimization, Wirelength optimization

1. Introduction

Placement in VLSI physical design is the problem of handing over positions for the fixed rectangular modules on the layout surface [1]. A floorplan is defined as the placement of flexible or fixed rectangular modules with permanent area but unidentified dimensions which are usually an upper and a lower bound on the aspect ratio [2, 3]. The input for the floorplan are a set of electronic circuit components usually represented as rectangular modules, and a net list which is nothing but the interconnections between the modules.

The aim of designing circuit floorplanning algorithm is to generate a floorplan of the rectangular circuit modules such that the total area and the total wire length of the floorplan are minimized. No module should overlaps with another module. A Floorplan layout structure [4] can be categorized as (a) Slicing floorplan and (b) Non-slicing floorplan. A floorplan is said to be slicing floorplan, if a block from the floorplan can be sliced in two by horizontally or vertically.

A floorplan is said to be non-slicing floorplan, if a block cannot be cut in two. A binary tree is used in representing the slicing floor plan. The Vertical Constraint Graph (VCG) and Horizontal Constraint Graph (HCG) [5] are used in representing the non-slicing floorplan. The sample slicing and non-slicing floorplan are shown in the figure 1 and 2 respectively.

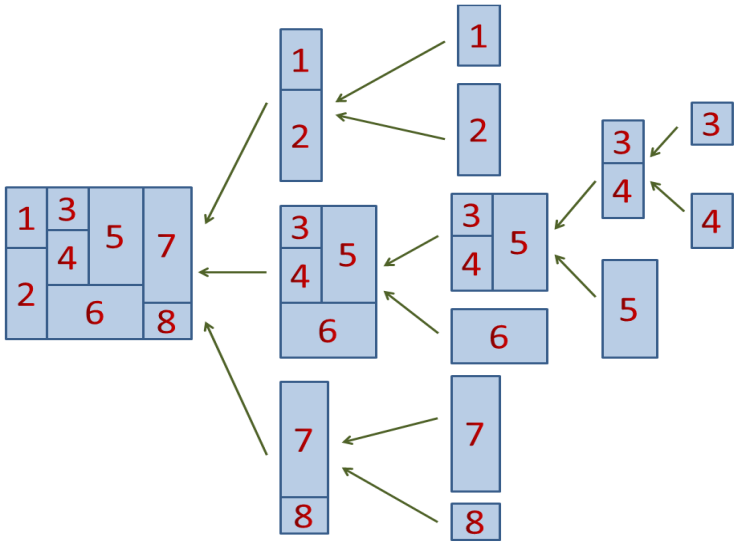

Fig. 1 Sample Slicing floorplan

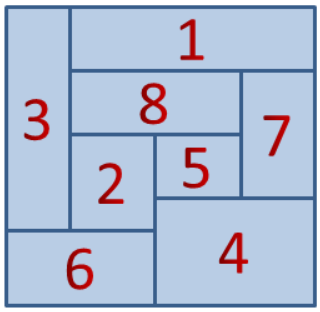

Fig. 2 Sample Non-Slicing floorplan

The representation for floorplanning is crucial to the effectiveness of a floorplanning problem specifically for the famous approach called Simulated Annealing. The floorplanning can be represented basically in two ways, topological representation and absolute representation. In the absolute representation, the exact location of each and every smaller module in the larger rectangle has to be specified. But in the topological representation, the absolute coordinates of each module are not specified. Because of the complexity and difficulty in absolute representation, the topological representations become more trendy and used in most of the research results. The example topological representations are Transitive Closure Graph (TCG) [6], O-Tree [7], Sequence Pair (SP) [8], B*-Tree [9] and Corner Block List (CBL) [10].

2. Problem Statement

The VLSI floor planning problem is defined as follows: "The input for the problem are 'n' numbers of electronic rectangular modules, $S = \{ M_1, M_2, ..., M_n\}$ with the corresponding areas $\{ A_1, A_2 ... , A_n\}$, an enveloping rectangle *R* and the net list *N* which specifying the interconnections between the modules. The resultant floorplan *F* of the modules gives the location of each and every modules such as for each M_i, the coordinate is $C_i=(x_i, y_i)$. The following two basic constraints have to be considered during placement. 1. Containment Constraint: All the placed rectangles lie within the outer box and 2. Non-Overlap Constraint: The placed rectangles are pair wise interior-disjoint".

In addition to these two basic constraints, the floor plan has to determine the smallest rectangular area into which a given set of smaller rectangular modules can be packed and the minimum length of wire needed to connect the modules. The main objective of floorplanning is to optimize the circuit performance by minimizing the chip area and reducing the total wirelength. The most common objective function for optimizing floorplan is

$$Cost(F) = \alpha A + \beta L$$

where,

L → Total wirelength,

A → Total area of the packing,

α and β → Constants.

There are many algorithms designed for solving floorplanning problem. For a given *N* numbers of modules, there are *(N!*N!)* numbers of feasible solutions available. But the problem is to find the optimal solution from the set of *(N!*N!)* solutions. The optimal solution is nothing but a best solution from the feasible solution. So finding an optimal solution for the floorplanning problem is computationally expensive. The best solution is viewed as the placement that gives the smallest rectangular area which covers all the smaller modules.

Table 1: Time Complexity for generating flooorplan for N numbers of modules.

3 Modules Order	Number of Solutions (for 3 Modules)		In General (for N modules)
M3, M2, M1	06	3!	N!
M3, M1, M2	06	3!	N!

M1, M2, M3	06	3!	N!
M1, M3, M1	06	3!	N!
M2, M1, M3	06	3!	N!
M2, M3, M1	06	3!	N!
Total	**36**	**3! * 3!**	**N! * N!**

The huge search space for the floorplanning problem shows that the solution for the floorplanning can be verified in polynomial time but cannot be determined in polynomial time. So the problem of solving floorplanning is known as NP-hard problem. For example, consider the problem of generating floorplan for three numbers of modules *M1, M2, M3* with different dimensions. The time complexity analysis is shown in Table 1. The order in which the modules are going to be placed is *M3, M2* and *M1*. Similarly there are five more orders are possible. In general, for *N* numbers of modules, there are *'N!'* numbers of input orders. For any single input order, the number of feasible solution is again based on the number of modules. For example, there are 6 (=3*2*1) feasible solutions for the order *M3, M2* and *M1*. i.e, the number of feasible solutions is *'N!'* for any single order. So the total number of solutions for 3 modules is 6*6=36, i.e. in general, for N modules, (N!*N!).

3. Literature Review

Maolin Tang et al. (2007) [11] designed a new hybrid genetic algorithm called Memetic Algorithm (MA) for solving both slicing and non-slicing floorplanning problems. The floorplan on N rectangular modules is represented as a horizontal O-tree of *'N+1'* nodes. Along with the *N* nodes, an additional node is used for representing left bottom dummy rectangle. To optimize the local search in the genetic algorithm, the authors used a new bias search strategy to reduce the search time. The fitness of any population in the genetic algorithm is the inverse of the cost of the floorplan. A constant threshold '*v*' is used in bias search. The experimental result of MA algorithm shows that which can produce optimal or nearly optimal solution.

Pradeep Fernando and Srinivas Katkoori (2008) [12] designed a multi-objective genetic algorithm for optaining area and wirelength optimal floorplanning. The non-domination concepts is used in assign the rank for the solutions. The authors used two new crossover operators which generate the floorplans from good sub-floorplans. The comparison results of proposed algorithm with the previous floorplanners show that the efficiency of the proposed algorithm save 18% wirelength and 4.6% area for the standard GSRC floorplanning benchmarks and save 26% wirelength for MCNC

benchmarks for 1.3% increase in area. *Jarrod A. Roy et al. (2009)* [13] proposed an algorithm to obtain the wire length optimal floorplan. In this paper, the best features of floorplanning and placement are combined which form a new approach called as floorplacement. The placement in floorplacement is based on min-cut bisection method. The min-cut placement considers the packed larger blocks as obstacles.

Guolong Chen et al. (2010) [14] designed a new smart decision algorithm based on the particle swarm optimization (PSO) approach for obtaining a feasible floorplan. The authors used module number based integer coding and a different value of speeding up coefficients along with the PSO for obtaining optimal placement solution. To achieve a better diversity, the basic ideology of mutation and crossover operator in genetic algorithm is modified which results in a novel PSO algorithm.

Jianli Chen et al. (2011) [15] designed a hybrid simulated annealing (HSA) algorithm to solve nonslicing floorplanning problem. The proposed algorithm exploits a new greedy method for constructing the initial B*-tree to represent the floorplan. The basic search operation of B*-tree is modified as a new dynamic bias search strategy for balancing global searching and local utilization. The experimental results on MCNC benchmarks show that the proposed approach can construct the optimal or nearly optimal floorplan for all the experimented problems.

T. Singha et al. (2012) [16] presented a new approach based on genetic algorithm for solving VLSI non-slicing floorplanning problems. The floorplan is represented as a non-slicing B* tree structure. The authors named the new genetic algorithm as Iterative Prototypes Optimization with Evolved Improvement (POEMS) algorithm. In the new algorithm, the local search is performed by using a genetic algorithm. The proposed algorithm is mainly focused on optimizing the execution time of the algorithm. The comparison of the suggested algorithm with the Differential Evolution (DE) algorithms Simulated Annealing (SA) shows that the proposed algorithm can generate area optimized floorplanning with optimized execution time. A novel approach for solving the problem of Bus-Driven Floorplanning (BDF) within fixed outer rectangular die was proposed by *Wenxu Sheng et al. (2013)* [17]. The BDF is a floorplanning which is generated by packing rectangular blocks along with the unshaped buses within outer die. The authors designed a new algorithm named Least Flexibility First (LFF) which generates a final floorplan that satisfies the following constraints: (1) Packing of all smaller blocks within the outer fixed rectangle (2) All buses are routable. (3) Minimization of total bus area (4) Minimization of total floorplan area.

The LFF algorithm first starts filling the corners of the outer die with the rectangular blocks and gradually moving towards center based on the flexibility of the blocks to be packed. Once a rectangular block is packed, the LFF packs the bus components that connect this block to the already packed blocks. The experimental results shows that the new approach can give an excellent solution for the floorplanning with minimized area, minimized length of all buses, reasonable running time and satisfied fixed outline constraints.

Kanesan Jeevan et al. (2013) [18] presented a new approach for generating area and wire-length optimized floorplanning. In the proposed work, a variation of Ant Colony System (ACS) named the Variable Order Ant System (VOAS) was designed and combined with the well known floorplan representation Corner List (CL) for optimizing the search time. The existing ACS is having some drawbacks in optimizing the search space: (1) Since the route selection is based on the probabilistic rule, the best solutions might vanish. (2) ACS uses the constant pheromone values. This leads to the possibility of selecting the worst solution as an optimal solution.

To overcome this weakness of ACS, the authors modified the pheromone update mechanism and the probabilistic rules for route selection of basic ACS algorithm which is named as VOAS. In VOAS, the pheromone control parameters α and β are selected as variable orders instead of taken as constants. Along with the artificial ants group, a new group of ants named reconnaissance ant is introduced in VOAS which are used in investigating local information in each colony.

Based on the collected information from the reconnaissance, the basic ant groups will select the next colony. This reduces the possibility of missing best solutions.

A new floorplan model is introduced by the authors named Corner List (CL) in which the blocks are placed independent of its series onto the bends which are not enclosed by the outline. The authors tested the proposed algorithms with Microelectronics Centre of North Carolina (MCNC) and Gigascale Systems Research Center (GSRC) benchmarks. The test result shows that the proposed algorithm can generate purely area optimized floorplan with the optimized execution time.

P. Sivaranjani et al. (2013) [19] presented an optimization algorithm for VLSI floorplanning named Hybrid Particle Swarm Optimization (HPSO). In this paper, the modules are initially represented by the popular floorplan model B* Tree. The authors combined the Genetic Algorithm's crossover and mutation concepts with the traditional Particle Swarm Optimization (PSO) for getting optimal solution.

Chyi-Shiang Hoo et al. (2013) [20] presented a new optimization method based on traditional Ant System (AS) for solving VLSI non-slicing floorplanning optimally. The proposed algorithm named Hierarchical-Congregated Ant System (H-CAS) performs the placement for VLSI based on the dimensions of the rectangular modules instead of area. In H-CAS, the artificial ants are designed as communicators those are able to share information between them. When the ants reach the target place, they share their travel experiences with their group by constructing a new pheromone trails. The creation of new group leads to reducing the further search space and the reducing the search time. At every level of bottom-up hierarchy, the ants are designed to initiate global information. A mathematical derivation for a novel relative whitespace formula for bottom-up hierarchy is made and be implanted in H-CAS's special update formula. The comparison of other floorplanning algorithms shows that the proposed H-CAS can generates the best near optimal solutions with respect to scaling, convergence, precision, stability, and reliability.

Yinshui Xia et al. (2014) [21] presented a Multi-supply voltage (MSV) technique for generating floorplan by optimizing area, wire length and power with minimized execution time. The authors used nonrandomized searching engine for generating efficient nonrectangular shaped voltage islands (NSVI) aware floorplanning. A generalized slicing tree is used in representing the floorplan. Based on the circuit voltage levels, a hypergraph which represent the circuit and subcircuits partitions for reducing the problem size is generated from the slicing tree.

The shape curve is used to keep the complete set of slicing floorplans. The floorplan can be generated by back-tracing the points in the shape curve. The NSVIs are produced according to the generated slicing tree and the physical location of the cores which optimize the power consumption.

Finally, the optimized floorplan is obtained by obtaining various partitioning results by using a heuristic method. The experimental results show the proposed technique produce the floorplan which is optimized in area, wire length, power and execution time.

Jianli Chen et al. (2014) [22] proposed a new model for estimating the wire length in finding wire-length optimal floorplan. The most common wire-length estimation method is Half Perimeter Wire Length (HPWL) which is not differentiable.

A powerful differentiable wire length approximation model named scaled Log-Sum-Exponential (sLSE) is presented in this paper for getting more exact approximation HPWL.

The authors presented an algorithm based on sLSE to find the wirelength optimal placement. The proposed sLSE based nonlinear solver used the multilevel framework of NTUplace3 with the scaled LSE wirelength model.

NTUplace3 is a high-class mixed size investigative placement algorithm proposed by *Jianli Chen and Wenxing Zhu.* In NTUplace3, the preplaced blocks and density constraints are primarily focused by the authors.

Xi Chen et al. (2014) [23] proposed a simple and effective algorithm for generating regularity constraint floorplan for multi-core processors. Apart from the optimization of area, wire length, power and temperature, the physical design of multi-core processors need to satisfy the regularity constraint.

In multi-core processors, the identical processors and memory cores are preferred to be placed in array layout. In this paper, the authors addressed the regularity constraints along with the area optimization of VLSI physical design. For handling symmetry constraints and the regularity constraint, the floorplan is represented on the most known sequence-pair floorplanning representation. The experimental result shows that the proposed algorithm outputs the regularity constraint floorplan with an average of 12% less wire length and reduced chip area.

Ahmet Unutulmaz et al. (2015) [24] designed an algorithm for slicing floorplan to optimize the area. The authors formulate the floorplan problem as a Convex Optimization Problem. The area of the compact floorplan is a convex function of its width or height. The optimization of area is done by proving convexity of the floorplan with the symmetry and proximity constraints of the constructed floorplan. They used a convex optimizer to prove this convexity which effectively optimizes the area. This proof is mainly impact floor- plan optimization of analog layouts which consists of capacitor blocks.

Kun He et al. (2015) [25] presented a new algorithm named Dynamic Reduction Algorithm (DRA), which is mainly designed to solve Rectangle Packing Area Minimization Problem (RPAMP) which is the sub problem of VLSI floorplanning. The RPP (Rectangle Packing Problem) is a problem of filling a larger rectangle with various smaller rectangles. The main objectives of RPAMP are to fill a larger rectangle with maximum number of smaller rectangles and to minimize the unused spaces in the larger rectangle.

In the dynamic reduction algorithm, the main floorplanning problem is transformed to a sequence of occurrence of the rectangle packing problem by finding the dimensions of the larger rectangle. The authors used an algorithm named Least Injury First (LIF) algorithm for solving the instance of RPP.

The LIF algorithm defines new metric named injury degree which calculates the potential negative impact of the relative placement. The investigational outcome of DRA is the highly efficient and effective solution for area minimization for rectangle packing problem, especially on larger numbers of rectangular modules.

Behnam Khodabandeloo et al. (2015) [26] proposed an optimization framework to determine a Multiple Supply Voltage (MSV)-aware floorplan. The main goal of this work is to generate a circuit floorplan for optimizing the power and temperature in VLSI circuit design. In the proposed framework, the authors used Integer Linear Programming (ILP) formulation along with the Simulated Annealing (SA) to improve the design efficiency by reducing the execution time. For finding optimal solution by reducing the search space, a heuristic algorithm is designed by the authors.

The experimental results show that the proposed framework recommends floorplans that are more power and temperature-efficient floorplan.

Yinshui Xia (2016) [27] presented a method for optimizing the power in the design of integrated chips. The author used the method called multi-supply voltage (MSV) along with the voltage island partitioning and level shifters placement during the design of floorplanning. The virtual level shifters are assigned for the nets in the netlist to reserve the deadspaces for the actual level shifters. The feedback of physical locations of LS is used in assigning voltage during floorplanning. A heuristic based algorithm is proposed by the authors for voltage assignment to obtain efficient execution time and accuracy. The LS placement and Voltage assignment are operated iteratively to satisfy timing and physical constraints. Experimental results on GSRC benchmark suites shows that the placement of LS is achieved by reduced power cost.

Wei Liu (2016) [28] proposed a technique for improving the circuit performance by shortening the delay. The large thermal gradient and the high temperature in the metal layers may have an effect on the interconnect design, reliability, signal delay and routing congestion. The temperature dependent wire delay is taken into account as a performance metric during the design of floorplan. Since the locations of macroblocks used in determining the thermal profile on the die, it is possible to perform temperature dependent delay estimation at the floorplanning stage. The thermal aware floorplanning is based on the idea of placing a hot block in the middle of cool blocks which effectively reduce the peak temperature. The experimental result shows that the proposed method can generate floorplan with better wire reliability and better signal delay.

J. Schneider (2016) [29] presented a floorplanning algorithm to improve the chip performance by optimizing the wirelength and area. In-stead of using heuristic method to obtain optimal floorplan, the authors used different approach to find the exact floorplan which can take reduced area and wirelength. The SPARK algorithm is used in solving rectangle packing problem.

The wirelength is estimated during floorplanning by calculating Half Perimeter Wire Length (HPWL). The proposed algorithm can solve floorplanning instances with up to 20 rectangles including blockages and thousands of nets. The experimental result shows that the proposed algorithm can generate net length optimal floorplans for apte, hp and xerox MCNC benchmark circuits.

The optimization targets for the above discussed algorithms are summarized in Table-2. The advantages and disadvantages of different algorithms are given the Table-3.

Table 2: Optimization target for different floorplanning algorithms.

S.No	Article	Floorplan Type	Representation	Algorithm Used	Optimization Target
1	[11]	Non-Slicing	O-Tree	Hybrid Genetic Algorithm - Memetic Algorithm	Area
2	[12]	Non-Slicing	Sequence Pair	Based on Genetic Algorithm	Area and Wirelength
3	[13]	Non-Slicing	Balanced B* tree	Minimized size Min-Cut floorplace-ment algorithm	Wirelength and running time of the algorithm
4	[14]	Slicing & Non-Slicing	Modified Sequence pair	Based on Particle Swarm Optimization	Wirelength and Area
5	[15]	Non-Slicing	B* tree	Hybrid Simulated Annealing Algorithm	Area and Execution Time
6	[16]	Non-Slicing	B* tree based on Ordered Binary Tree	Based on iterative POEMS (Prototypes Optimization with Evolved Improvement) algorithm	Area
7	[17]	Both Slicing and Non-	Transitive Closure Graph (TCG)	Deterministic Lease Flexibility First Algorithm	Bus Area and Chip Area

		Slicing			
8	[18]	Non-Slicing	Corner List	Variable Order Ant System Optimization	Execution Time and Area
9	[19]	Non-Slicing	B* Tree	Hybrid Particle Swarm Optimization	Chip Area and Interconnection Wire length
10	[20]	Both Slicing and Non-Slicing	*p*-nary tree, where $1 \leq p \leq 5$	Hierarchical Congregated Ant Systems	Area and Execution Time
11	[21]	Slicing	Hypergraph and generalized slicing tree	Based on Randomized Algorithm	Power, Wirelength and Space
12	[22]	Non-Slicing	Hypergraph	Scaled Log-Sum Exponential (sLSE) wirelength model.	Wirelength
13	[23]	Non-Slicing	Sequence Pairs	Based on Simulated Annealing	Address the Regularity Constraints for Multi-core processors.
14	[24]	Slicing	Binary Tree	Based on a convex function of its width or height of rectangular blocks.	Area
15	[25]	Non-Slicing	B* Tree	Based on Dynamic Reduction Algorithm – Uses the Least Injury First approach.	Area
16	[26]	Non-Slicing	Directed Acyclic Graph	Integer Linear Programming and Simulated Annealing	Power and Temperature.
17	[27]	Non-Slicing	Generalized slicing tree.	A new algorithm *DeFer* – is a deferred decision making (DDM) technique.	Power

				Heuristic algorithm for voltage assignment.	
18	[28]	Non-Slicing	Sequence Pairs	Fast Simulated Annealing	Temperature and wire delay.
19	[29]	Non-Slicing	Sequence Pairs	SPARK – An wire length optimal rectangle packing algorithm	Wire length

Table 3: Advantages and Disadvantages of different floorplanning algorithms.

S.No	Article	Pros.	Cons.
1	[11]	➢ Uses an efficient genetic search method. ➢ Uses an effective local search algorithm ➢ Give the better performance than MDA and GA	➢ Use of static threshold bias search reduces the performance of the algorithm
2	[12]	➢ Simultaneously reduce the wirelength and area.	➢ Longer response time of Genetic Algorithm
3	[13]	➢ Use of enhanced B* tree reduces the white spaces. ➢ Improved scalability and robustness of floorplacement ➢ Lower HPWL and lower running time.	➢ Not suitable for slicing floorplan ➢ Area minimization is not considered.
4	[14]	➢ Solve the discrete problems those cannot be solved by traditional PSO. ➢ Suitable for Slicing and Non-Slicing.	➢ Decreases in performance for multi-objective floorplanning.
5	[15]	➢ Use of dynamic threshold reduces the search space. ➢ Uses an improved bias search method. ➢ Better improvement on area	➢ Not suitable for slicing FP ➢ Use of greedy method may not give the exact solution. ➢ Wirelength, Power and other optimization factors are not

		minimization over LFF, MA and PSO	considered.
6	[16]	➢ Uses GA for minimizing the local search space ➢ Better improvement over SA and DE algorithms.	➢ Not suitable for slicing floorplan ➢ Wirelength, Power and other optimization factors are not considered.
7	[17]	➢ No limitations on bus shape and number of bendings. ➢ Variable aspect ratio (1/3 to 3) for slicing floorplan	➢ For hard blocks, the wire length is not optimized. ➢ Complex flexibility modification algorithm.
8	[18]	➢ Use of variable order property in basic Ant Systems gives a better choice of modules for floorplanning. ➢ The reconnaissance ants performs efficient local search. ➢ Improved results interms of Area optimization.	➢ Cannot handle slicing floorplan ➢ Power minimization and thermal optimization are not considered.
9	[19]	➢ Use of crossover and mutation from GA give the optimal placement solution. ➢ Better improvement over Hyprid Simulation Annealing (HAS)	➢ Execution time is not optimized. ➢ Power and Thermal optimizations are not considered.
10	[20]	➢ Simple bottom up hierarchical floorplan construction. ➢ Improved Scalability, Convergence and Reliability.	➢ The whitespace and the outline of a circuit are not controllable and predictable due to the use of bottom up approach. ➢ Communication latency is more.
11	[21]	➢ MSV-aware floorplan. ➢ Variable aspect ratios. ➢ Reduce the power consumption	➢ Cannot handle non-slicing floorplan ➢ Expensive execution time
12	[22]	➢ Simple method for estimating the interconnection wire length	➢ Area, Power and Thermal optimizations are not considered.

		➢ More exact approximation of HPWL	➢ Not suitable for slicing floorplan.
13	[23]	➢ Optimize the area and wirelength in reasonable amount. ➢ First algorithm which include the regularity constraints to the floorplanning for the multi core processors.	➢ Complexity of algorithm is more. ➢ Since this concentrate on regularity constraints, only suitable for multi-core processors. ➢ Power and thermal optimizations are not considered.
14	[24]	➢ Efficiently optimize the area of slicing floorplan using a convex optimizer.	➢ Not suitable for non-slicing floorplan. ➢ Other optimization factors like wirelength, power and temperature optimization are not addressed.
15	[25]	➢ Suitable for the large scale benchmarks. ➢ Compactness, Feasibility, non-inferiority and halting of the compacting algorithm	➢ Wire length, power and thermal optimizations are not addressed.
16	[26]	➢ Can generate Power and Temperature optimal floorplan. ➢ Lower execution time.	➢ The optimization of Area and Wirelength are not addressed.
17	[27]	➢ Post-power optimization is achieved during floorplanning. ➢ Reservation of dead space for LSs in advance.	➢ Deadspace management is not addressed to handle LSs with different sizes ➢ Wire length and other optimization factors are not considered during floorplanning.
18	[28]	➢ Temperature dependent factors such as congestion and interconnect reliability are considered as one of the optimization factors.	➢ No methods are included for optimizing area and power. ➢ Moderate increase in area and wirelength.

19	[29]	➢ Able to handle blockages ➢ Can generate wirelength optimal floorplan	➢ Cannot find the solution for large instances (ami33, ami49, N100, N200, N300, pcb500 etc.,) in reasonable runtime. ➢ Optimization of area, power and thermal are not considered.

4. Conclusion

The design challenges in solving the floorplanning problem are discussed in this paper. In the design of an optimal floorplan, the various metrics have to be addressed. A good floorplan should take as minimized die area, reduced wire length, optimized power, optimized thermal, regularity constraints and etc. From the above literature survey, the followings are observed. Many floorplanning algorithms have been designed separately for (1) Optimizing AREA (2) Optimizing WIRELENGTH (3) Optimizing POWER (4) Considering REGULARITY CONSTRAINTS (5) Considering THERMAL (6) Optimizing RUNNING TIME. Also some multi-objective algorithms have been designed by combining (1) Area with Wirelength (2) Area with Thermal (3) Area with Power (4) Area with Regularity Constraints. The advantages and drawbacks of all the algorithms are discussed in this paper. The main problem in these multi-objective algorithms is very expansive computational cost. Most of the algorithms are suitable for either soft module or hard modules. So far there is no single algorithm is designed for generating floorplan which consider all the optimization factors and special constraints.

5. References

1. Markov I.L, Hu J.b, Kim M.C. 2015. Progress and Challenges in VLSI Placement Research. Proceedings of the IEEE; 103(11):1985-2003.
2. Guolong Chen, Wenzhong Guo, Yuzhong Chen. 2010. A PSO-based intelligent decision algorithm for VLSI floorplanning, Springer, Soft Computing. 1329-1337.
3. D. F. Wong, C. L. Liu. 1986. A new algorithm for floorplan design. Proceeding of the ACWIEEE Design Antomation Conference. 101-107.
4. Kimura Y. Ida K. 2006. Improved genetic algorithm for VLSI floorplan design with non-slicing structure. Computers and Industrial Engineering; 50(4): 528-540.

5. Dong S, Zhou S, Hong X, Cheng C, Gu J, Cai Y. 2002. An optimum placement search algorithm based on extended corner block list. Journal of Computer Science and Technology; 17(6):699-707.
6. Jai-Ming Lin and Yao-Wen Chang. 2005. TCG: A Transitive Closure Graph-Based Representation for General Floorplans, IEEE transactions on very large scale integration (VLSI) systems.13(2):288-292.
7. Guo, Pei-Ning, Cheng, Chung-Kuan, Yoshimura, Takeshi. 1999. O-tree representation of non-slicing floorplan and its applications. Proceedings - Design Automation Conference; 268-273.
8. 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, V01. 15, N0. 12, 1996, pp. 1518-1524.
9. Tung-Chieh Chen, and Yao-Wen Chang. 2006. Modern Floorplanning Based on B*-Tree and Fast Simulated Annealing. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems. 25(4):637-650.
10. Yuchun Ma', Sheqin Dong', Xianlong Hong', yici Cai', Chung-Kuan Cheng2, Jun Gu3. 2001. VLSI Floorplanning with Boundary Constraints Based on Corner Block List, IEEE.509-514.
11. Maolin Tang, Member, IEEE, and Xin Yao, Fellow, IEEE. A Memetic Algorithm for VLSI Floorplanning. IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART B: CYBERNETICS, VOL. 37, NO. 1, FEBRUARY 2007. 62-69.
12. Pradeep Fernando and Srinivas Katkoori. 2008. An Elitist Non-Dominated Sorting based Genetic Algorithm for Simultaneous Area and Wirelength Minimization in VLSI Floorplanning. 21st International Conference on VLSI Design, IEEE. 337-342.
13. Jarrod A.Roy, Aaron N.N, Rajat Aggarwal, Venky Ramachandran, Igor L.Markov. 2009. Solving modern mixed-size placement instances. INTEGRATION, the VLSI journal 42 (2009) 262–275.
14. Guolong Chen, Wenzhong Guo, Yuzhong Chen. 2010. A PSO-based intelligent decision algorithm for VLSI floorplanning. Soft Computing Volume 14, Issue 12 , pp 1329-1337.
15. Jianli Chen, Wenxing Zhu, and M. M. Ali. 2011. A Hybrid Simulated Annealing Algorithm for Nonslicing VLSI Floorplanning, IEEE transactions on systems, man and cybernetics-part c: applications and reviews. 41(4):544-553.

16. T. Singha, H. S. Dutta, M. De. 2012. Optimization of Floor-planning using Genetic Algorithm. Procedia Technology 4 (2012) 825 – 829.
17. Wenxu Sheng, SheqinDong. 2013. Multi-bend bus-driven floorplanning considering fixed-outline constraints. INTEGRATION, the VLSI journal; 46:142–152.
18. Chyi-Shiang Hoo, Kanesan Jeevan, Velappa Ganapathy, Harikrishnan Ramiah. 2013. Variable-Order Ant System for VLSI multiobjective floorplanning. Applied Soft Computing; 13:3285–3297.
19. P Sivaranjani and K K Kawya. 2013. Performance Analysis of VLSI Floor planning using Evolutionary Algorithm. IJCA Proceedings on International Conference on Innovations In Intelligent Instrumentation, Optimization and Electrical Sciences ICIIIOES; 9:42-46.
20. Chyi-Shiang Hoo, Hock-Chai Yeo, Kanesan Jeevan, Velappa Ganapathy, Harikrishnan Ramiah, Irfan Anjum Badruddin. 2013. Hierarchical congregated ant system for bottom-up VLSI placements. Engineering Applications of Artificial Intelligence 26 (2013) 584–602.
21. Zhufei Chu,Yinshui Xia, LunyaoWang, JianWang. 2014. Efficient nonrectangular shaped voltage island aware floorplanning with nonrandomized searching engine. MicroelectronicsJournal45(2014)382–393.
22. Jianli Chen and Wenxing Zhu. 2014. A Scaled LSE Wirelength Model for VLSI Global Placement. 11th International Conference on Fuzzy Systems and Knowledge Discovery, IEEE; 881 – 885.
23. Xi Chen, Jiang Hu, Ning Xu. Regularity-constrained floorplanning for multi-core processors. INTEGRATION, the VLSI journal. 47(2014)86–95.
24. Ahmet Unutulmaz, Gunhan Dundar, FranciscoV.Fernandez. 2015. On the convex formulation of area for slicing floorplans. INTEGRATION, the VLSI journal; 50:74–80.
25. Kun He, Pengli Ji,Chumin Li. 2015. Dynamic reduction heuristics for the rectangle packing area minimization problem. European Journal of Operational Research 241 (2015) 674–685.
26. Aminollah Mahabadi, Ahmad Khonsari, Behnam Khodabandeloo, Hamid Noori, Alireza Majidi, 2015. Critical path-aware voltage island partitioning and floorplanning for hard real-time embedded systems. INTEGRATION, the VLSI journal; 48:21-35.

27. Zhufei Chu,Yinshui Xia, Lunyao Wang. 2016. Multi-supply voltage (MSV) driven SoC floorplanning for fast design convergence. INTEGRATION,theVLSIjournal52(2016)335–346.
28. Andreas Thor Winther, Wei Liu, Alberto Nannarelli, Sarma Vrudhula. 2015. Thermal aware floorplanning incorporating temperature dependent wire delay estimation. Microprocessors and Microsystems 39 (2015) 807–815.
29. J. Funke, S.Hougardy, J.Schneider. 2016. An exact algorithm for wirelength optimal placements in VLSI design. INTEGRATION, the VLSIjournal52(2016)355–366.

Corresponding Author:
B. Srinivasan*,
Email: *srinivasan@core.sastra.edu*