International Journal of Swarm Intelligence Research
Volume 14 • Issue 1

Very Large-Scale Integration Floor Planning on FIR and Lattice Filters Design With Multi-Objective Hybrid Optimization

Pushpalatha Pondreti, Jawaharlal Nehru Technological University, India*
Babulu Kaparapu, Jawaharlal Nehru Technological University, India

ABSTRACT

Floor planning is indeed an obvious design process in VLSI physical layout since it specifies the dimensions, structure, as well as positions of components upon the chip; in addition, information regarding the overarching silicon area, interlinks, and latency is also provided. VLSI floor planning is an NP-hard issue as the floor plan representations are a crucial component in this process. The intricacy, as well as solution space of the floor plan layout, is influenced by the floorplan visualizations. To tackle the VLSI floor plan challenge, numerous researchers have developed numerous meta-heuristic optimization techniques. The main objective of this work presents a novel multi-objective hybrid optimization method for solving the floor plan optimization issue. Standard DOX and ALO are conceptually combined in the proposed hybrid optimization referred to as Dingo Updated Ant Lion Optimization (DUALO) model. The multi-objectives like wire length, area, and penalty function are taken into consideration.

KEYWORDS

DUALO, Floor Plan, Multi-Objectives, VLSI

Nomenclature

Abbreviation	Description
SA	Simulated Annealing
ALO	Ant Lion Optimizer
SRAM	Static Random-Access Memory
NoC	Network On Chip
LOA	Lion Optimization Algorithm’
OB	Order Based
P-PSO	Parallel Particle Swarm Optimization
BCSA	B*Tree Crossover Simulated Annealing Technique
EHAFO	Energy And Heat-Aware Firefly Optimization
SP	Sequence Pair
DOX	Dingo Optimizer
MOFO-FP	Multi-Objective Firefly Optimization-Based Floorplanning

DOI: 10.4018/IJSIR.321237
*Corresponding Author

1. INTRODUCTION

The art of the physical design is referred to as floor planning. Several systems have used the Very Large Scale Integration (VLSI) design approach to open up new possibilities for high-performance computing, telecommunications, and ordinary electronic devices (e.g., entertainment). Moore's Law states that as VLSI technology has advanced, the number of manufactured devices in a single integrated circuit has steadily increased. To ensure quality, stability, and extensibility, it is more important to regulate the design process due to the rise in complexity, which includes significant bookkeeping and administrative responsibilities. With recent technological advancements, system design has expanded significantly, as well as the count of transistors on a chip has surpassed the range of possible combinations. As a consequence, Physical Design has become increasingly essential throughout the VLSI designing phase (Zhang et al. 2020; Vipin 2019; Singh and Baghel 2021). Floor planning seems to be the fundamental step in the Physical Design Flow process. This specifies the positions and size of modules upon the chip to maximize the chip's surface area. One key limitation to considering the locations of the modules is that there's no overlapping between two modules (Sivaranjani and Senthil Kumar 2015; Shunmugathammal et al. 2020a; Srinivasan and Venkatesan 2021; Prakash and Lal 2021). To deal with the rising complexity of architecture, hierarchical and IP-based components are utilized, making floor planning throughout the VLSI design cycle is increasingly realistic. Researchers have recently discovered the benefits of VLSI floor planning (Vipin 2019; Lin et al. 2021a; Mohapatra et al. 2020), particularly comprising a large number of chips. The architecture that must be recognized has been put together in floor planning, and specific space should indeed be allocated with different satisfactory limits to bring things close together, thereby reducing the dead space and cost incurred (Singh and Baghel 2021; Sivaranjani and Senthil Kumar 2015). The dynamics of the model is reduced. In the integrated circuit design floor planning is the major step (Andrukhiv et al. 2020; Singhet al. 2021).

A floor plan should be developed based on the design space and takes into account a variety of modules, such as macros used to represent memory, design, and IP cores, as well as their space requirements. The aspect ratio, as well as the IO architecture, is considered while designing the floors (Shunmugathammal et al. 2020b) (Giorgini et al. 2018; Wang et al. 2020; Sadeghi et al. 2020). If the floor plan would be of poor quality, there's also a risk of routing bottleneck and dying area wastage (Prakash and Lal 2021). As the available resources of routing are restricted, velocity and area are the critical computational factors to be taken into account. When the area becomes decreased, it guarantees that the portions are closer together and enables the architecture to use fewer resources. As a result, the amount of routing elements can be minimized, and the internal components could be connected with a small piece of wire. Also possible are signal pathways with much more space, shorter routing timeframes, and high-speed end-to-end measurements (Ren et al. 2020; Huang et al. 2020; Khan et al. 2020; Lin et al. 2021a). Therefore, the floor planning technique's main objective is to reduce cost constraints such as channel length, layout design area, and so on, to enhance the performance by cutting costs, space, and wire length. The risk of NP-hard combinatorial optimization in VLSI design (Shunmugathammal et al. 2020a) occurs as it contains structural abnormalities of circuit size, and therefore it is difficult to find the optimal solution. This heuristic approach (Srinivasan and Venkatesan 2021; Guler and Jha 2020) could be used for a global solution. The floor layouts are being shown using a rectangular dissection, and the borders have a rectangle form as well (Teng et al. 2018; Chen et al. 2018; Lin et al. 2021b; Mohapatra et al. 2020; Vehring et al. 2020). These lines are given vertically or horizontally, and the modules are dumped in a rectangle form to improve the automation process. The elimination of the module in floor planning is indeed a significant issue (Guler and Jha 2020; Sari and Psarakis 2017; Kouchi et al. 2020; Guo et al. 2018; Choudhury and Pradhan 2019). Typically, there are 2 types of floor plans: (1) Slicing the floor plan, i.e. dividing the floor plan into two parts along vertical and horizontal lines, and (2) Non-slicing of the floor design, in which the modules are not partitioned (Shunmugathammal et al. 2020b). The VLSI design develops

from an architectural blueprint created using the product's requirements. Floor plans for the chip are dependent on the connectivity, accessibility, and space of the components with restrictions on block arrangement. The next step is to lay out the electrical plans in a power-grid topology, which includes the power requirements provided over the topology for even distribution across all of the chip's components and standard cells. In order to minimize timing and shorten wire lengths, the proposed model is developed.

Research Questions:

1. What is the importance of floor planning in the VLSI design?
2. What is the main objective of floor planning?
3. What are the factors included in effective floor planning?
4. How to solve the optimization issues?
5. How to reduce the amount of space in the floor plan?

The major contributions of this research are

- The suggested hybrid optimization known as the Dingo Updated Ant Lion Optimization (DUALO) model conceptually combines standard DOX and ALO.
- It takes into account multiple goals like wire length, area, and penalty function.
- The optimal wire size and area estimation reduce the amount of space in the floor plan.

This paper's remainder is structured as follows: The published works on VLSI floor planning are discussed in Section 2. The proposed floor planning in VLSI is covered in Sections 3, 4, and 5, which provide an overview, a challenge model for the floor planning in VLSI, and the results obtained, accordingly.

Section 6 provides a conclusion to this essay.

2. LITERATURE REVIEW

2.1. Related Works

Zhang et al. (2020) have projected a machine learning-based framework for efficient floorplanning. Moreover, to minimize the time consumed for designing an integrated circuit, the authors have considered the feature selection and model stacking approaches. In the early floorplan stage, the projected model was capable of predicting the post-route slack of SRAM in the framework. The selected relevant features were initially ranked and the stacking model was constructed using integrating the different kinds of models for boosting the accuracy. The evaluation of the projected model was accomplished in a 28 nm technology, wherein the authors achieved an efficient floor plan within 60 seconds. Vipin (2019) has introduced Fat trees for NoC implementations by alleviating the low bisection bandwidth problem. The authors have used highly thick links for interconnecting the switches and the root node. Moreover, the authors have utilized the fatter links to achieve higher by altering the clock frequency between the switches' bandwidth Singh and Baghel (2021) have proposed an OB representation with the SA algorithm for a fixed outline floorplan. For hard IP modules, the authors have considered two physical quantities like the wire length and area. Moreover, to acquire a clear and good floor plan with no overlap among the modules, the authors have proposed an optimization model. Better optimal performances were achieved via the Simulated Annealing algorithm on the IC floor plan. Sivaranjani and Senthil Kumar (2015) have projected a "smart decision-making hybrid particle swarm optimization-genetic algorithm" to lessen the "area, wire length, and hotspot" by distributing the temperature evenly across the chip has been presented. The initial floorplan was generated using the B*-tree, and then the optimal placement solution was acquired with the aid of the

PSO-GA-based hybrid algorithm. Shunmugathammal et al. (2020a) have utilized the LOA to solve the floorplan optimization problem. The for multi-objective floor planning problem, the authors have used the B*tree crossover operator, which introduced additional perturbations in the initial B*tree structure. Srinivasan and Venkatesan (2021) have created a MOFO-FP approach to increase floor planning performance while consuming the least amount of energy and generating the least amount of heat. With defined outline limitations for VLSI floor planning, the major goal of the MOFO-FP approach has been to decrease heat production; area occupied, and wire length. The EHAFO method has been used to optimize the floor plan design once it has been constructed. For effective floor planning in VLSI design, the EHAFO method calculates multi-objective functions such as heat generation; space occupied, and wire length. The light intensity of each firefly has been computed using the objective functions. To find the optimal place, lesser-brightness fireflies are transported toward higher-brightness fireflies. Finally, the intensity of the ranking technique used to rank the fireflies has been used to determine the best tree structure for VLSI floor planning. As a result, the MOFO-FP technology was suggested, which minimizes energy usage, wire length, and heat output. In Prakash and Lal (2021) to solve the floor planning issue, researchers created a metaheuristic strategy known as P-PSO. The P-PSO explores the search space for an optimum solution using a novel "greedy operation" on the SP. The applied P-PSO may be utilized to provide an optimum solution, according to analysis results on the "Microelectronic Centre of North Carolina and Gigascale Systems Research Center benchmarks". In Shunmugathammal et al. (2020b) for a fixed-outline floor planning issue, the BCSA was suggested. With the efficient simulated annealing approach, a unique crossover in B*tree has been introduced. Simple SA algorithm's exploration capabilities are improved by the proposed ways. The "Microelectronics Centre of North Carolina benchmark circuits" are used to test BCSA. The results are superior to those of most state-of-the-art algorithms. BCSA has a lower percentage of dead space. For bigger tasks, the BCSA method is more efficient.

2.2. Review

Floor planning determines how to arrange the elements on a chip so that none of them overlap while controlling the area, wire length, and other performance characteristics to achieve the best results. As a consequence, a good floor-planning method that delivers much more optimal chip architectural solutions. A slicing architecture can be represented by a binary tree, with leaves symbolizing components and internal nodes representing horizontal or vertical cut lines. For non-slicing floor layouts, there have been several options, notably sequence pair (Zhang et al. 2020), BSG, O-tree, and B* tree (Shunmugathammal et al. 2020a). By simulating annealing, sequence pairings may be utilized to floor plan hard rectangular blocks (Srinivasan and Venkatesan 2021; Giorgini et al. 2018). This research proposes an iterative optimized framework for handling non-slicing floor plans that leverages an optimization algorithm for local search on each iteration and applies an ordered non-overlapping oriented constraint for the positioning of non-overlapping rectangular components.

VLSI floor planning is an NP-Complete issue that can be tackled using heuristic and meta-heuristic algorithms. The floor planning step of the VLSI circuit confronts advanced limitations such as fixed outline, and the focus has been placed on minimizing the dead space in the floor plan, which in turn minimizes the overall wire length, resulting in reduced signal transmission latency. The goal of floor planning optimization with the fixed-die-outline constraint is to reduce the amount of dead space in the floor plan layout while reducing chip area and interconnecting wire length. A mixed technique based on the fusion of adaptive behavior models of biological systems and composite architectures of solution algorithms is used to portray the VLSI floor planning problem as evolutionary change. This enables large-scale problems to be addressed and high-quality outcomes to be obtained in a reasonable amount of time. For VLSI non-slicing floor plan representation, a meta-heuristic approach is presented. These algorithms are capable of either diversification or intensification. As a result, a successful optimization algorithm must strike a balance between exploration and exploitation. Table 1 deploys the review on existing methods.

Table 1.
Review on existing methods

Author	Method	Advantages	Disadvantages
Zhang et al. (2020)	static random-access memory (SRAM)	• The prediction accuracy is high. • The speed is high.	• Need consideration to select the proper features.
Vipin (2019)	binary tree-based Network on Chip (NoC)	• The throughput is high. • The resource utilization is also maximized.	• The asynchronous switches are not considered.
Singh and Baghel (2021)	Simulated Annealing algorithm	• The power is minimized. • The thermal issue is also reduced.	• Need to reduce computational time
Sivaranjani and Senthil Kumar (2015)	PSO-GA algorithm	• The use of temperature is reduced. • The accuracy is high.	• Need to concentrate on th e error analysis.
Shunmugathammal et al. (2020a)	B*tree crossover simulated annealing algorithm	• The exploration capabilities are improved.	• The speed is reduced.
Srinivasan and Venkatesan (2021)	multi-objective firefly optimization-based floorplanning (MOFO-FP)	• The space consumption is reduced. • The heat generation is also reduced.	• Need to concentrate on the usage of more parameters.
Prakash and Lal (2021)	Parallel particle swarm optimization.	• The accuracy is high.	• The loss is occurred.
Shunmugathammal et al. (2020b)	Lion optimization algorithm.	• Better area minimization.	• The time required is high.

2.3 Objectives

- To improve the accuracy of the model.
- To reduce the amount of space in the floor plan.
- To minimize the cost function.

3. PROPOSED FLOOR PLANNING IN VLSI: AN OVERVIEW

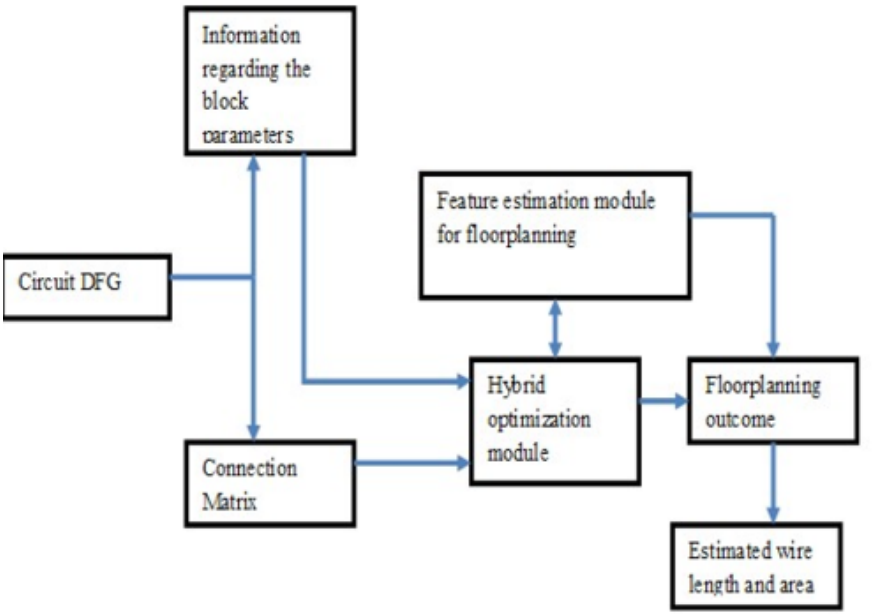
To provide a promising solution to the floorplan optimization problem, this research offers a novel multi-objective hybrid optimization. DUALO is a suggested hybrid optimization model that combines the standard DOX (Chen et al. 2019) with ALO (Bairwa et al. 2021). The ideal wire length estimate, area estimation, penalty function, and over-lapping parameter are all carried out using the newly suggested multi-objective hybrid optimization. The floor plan's area will be lowered while the ideal wire length estimation is being done, which helps to cut down on delays. The layout space is subsequently reduced. The suggested work's flowchart is depicted in fig 1.

3.1. Objective Function

The mathematical interaction between the entities that need to reduce the cost consumption of floor planning can be expressed as the objective function. The final step will not yield a better result until the variables of the functions are established. The objective function guides the development of a solution. Because the objective function includes two parameters: internal wire length and area. The quantity generates a non-overlapping restriction by raising the cost factor and adding a substantial penalty amount to the cost violation.

The weighted aggregate of all the components determines the objective functions that encompass the length "l"and area of wire a for constructing a floor plan F, and it is provided as per Eq. (1).

Figure 1.
Flowchart of proposed work



$$\cos t\left(F\right)=\alpha\times\left(\frac{Area\left(F\right)}{Area*}\right)+\left(1-\alpha\right)\times\left(\frac{wirelength\left(F\right)}{wirelength*}\right)+ \\ +PF\times T_{v},0\le\alpha\le1 \tag{1}$$

The area's weight is indicated by α, whereas the wire length's weight is symbolized by $1-\alpha$. To prevent overlapping PF specifies the penalty factor and T_v represents the overall number of violations. In addition, $Area*$ and $wirelength*$ points to the overall area and wire length, respectively.

In this research, floor planning is referred to as a diminished problem wherein the cost must therefore be kept as low as reasonably achievable, as stated in Eq. (2).

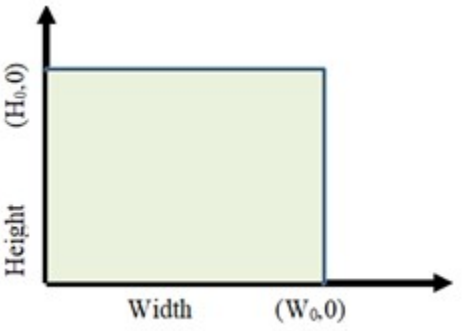
$$obj=\min\left[\cos t\left(F\right)\right] \tag{2}$$

4. PROBLEM MODEL FOR FLOOR PLANNING IN VLSI

The major intention behind IC floor planning is to arrange the modules optimally into the provided layout shown in Fig.2.

Dead Space: Within the fixed floorplan region, the area not occupied by any module is referred in Figure 3. As a result, if the optimum area is lowered, dead space on the chip can be minimized, lowering the silicon wafer cost. A new hybrid optimization model known as DUALO is presented in this research effort to reduce this dead space, thereby leading to a reduction in the design cost. Eq. (3) is used to compute the proportion of dead space.

Figure 2.
Provided layout: an illustration



$$DeadSpace = \frac{O - S}{O} * 100 \tag{3}$$

Here, O, S points to the optimal floor plan area and the sum of all modules area.

System Model: Let the set of modules be $M = \left\{m_1, m_2 .... m_N\right\}$, in which the notation N point to the count of modules. There are three different types of rectangular modules: soft modules S, hard modules H and pre-placed modules P. The soft modules feature a changeable magnitude interconnection with a set separation at fastened intervals. Each residence and the numerical connection is fundamentally established in hard modules. The frilly description is as follows: Floor plan with slicing: Cutting a floor plan horizontally or vertically repeatedly creates a slicing floor plan. Optimally placed modules with the DUALO model shows in Fig.4

Figure 3.
Dead space

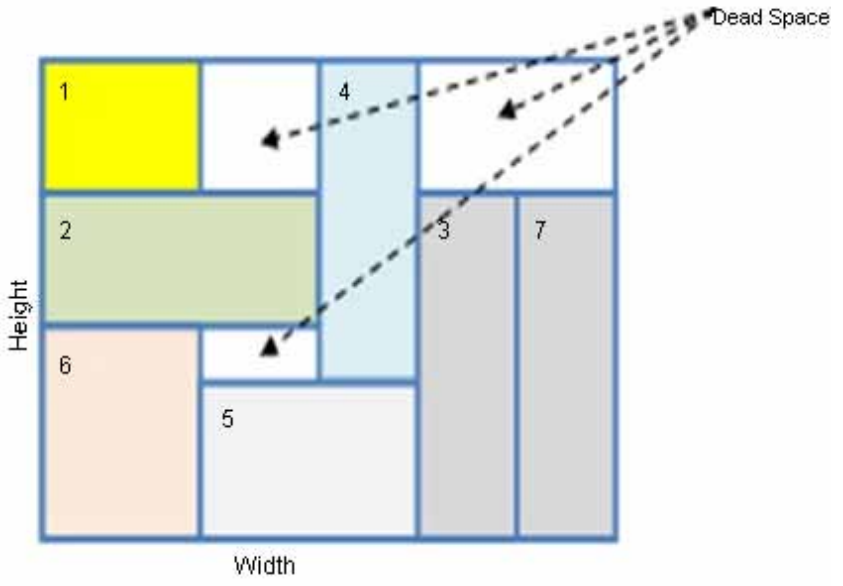
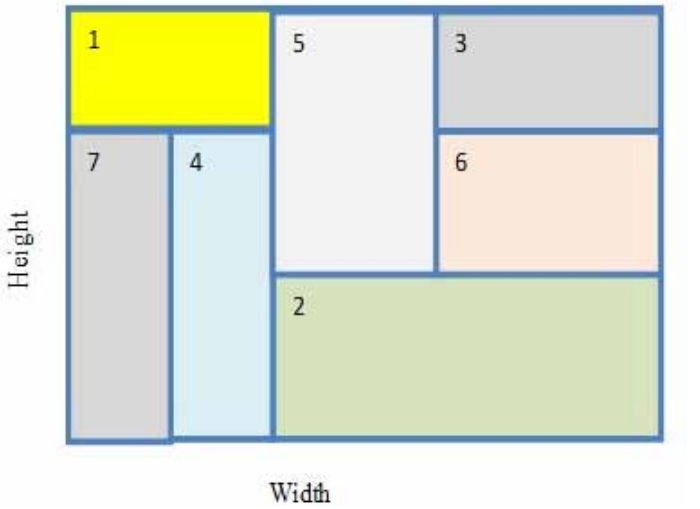


Figure 4.
Optimally placed modules with a DUALO model



Each of the rectangular modules M_i has a height H_i, width W_i and area $A_i = W_i * H_i$. The height, as well as width, ranges from $1 \le i \le N$. For M_i, the magnitude relation is denoted as H_i / W_i. Several modules combine to form a super-module, and this is also referred to as a subfloor plan. For M modules, a floor plan encapsulates an enveloping rectangle R.

Each rectangle must have the capability of housing the component to which it was assigned. The task specifies a set of hard modules as well as outline limitations. The modules inside the provided outline have the opportunity to independently move about. Since all of the modules within the outline should not replicate and overlap each other, the first quadrant is feasible packaging. The objective is to produce a viable floor plan R that minimizes the floor plan R's total area while meeting the floor planning-overlapping constraint.

The following are the inputs for the floor planning problem:

1) A collection of M rectangular modules $M = \left\{m_1, m_2 .... m_N\right\}$, each with an area of A_i, wherein $1 \le i \le N$.
2) The interconnection matrix $C_n = \left[c_{ij}\right]$, $1 \le i; j \le n$ 1, wherein $\left[c_{ij}\right]$ signifies interconnectedness between modules i, j.
3) A collection of widths as well as heights $\left(W_1, H_1\right)\left(W_2, H_2\right)$, ..., $\left(W_m, H_m\right)$ wherein W_i and H_i are indeed the module i $(1 \le i \le m)$ widths and height, accordingly.

The bottom left $\left(x, y\right)$ coordinates of every module, as well as the width and heights of each module on the finalized floor plan, are the floor plan's outcome, which must match the basic guidelines:

- There are no modules that overlap.
- For every module, bottom left coordinates $\left(x_i, y_i\right)$, width W_i, and height H_i are used, such as area, $A_i = H_i * W_i$, $1 \le i \le m$.

All of the modules are encased in the arrangement, which optimizes overall area, wire length, and other performance parameters. The goal of optimization is to reduce the overall area of a form floorplan while staying within certain limits. In the IC floor planning problem, there are two constraints:

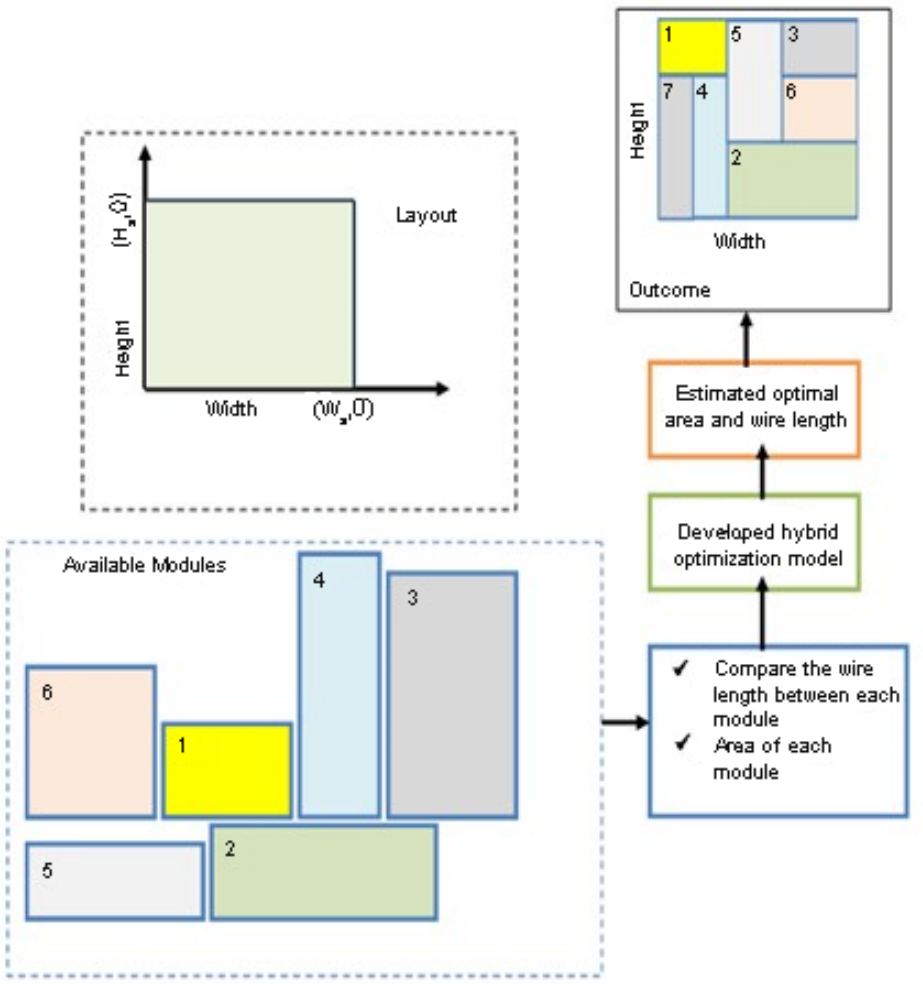
1) All components need to fit tightly inside a specific boundary region.

Assume that the fixed outline floor plan's width and height are W_0 and H_0, respectively.

2) There should be no modules piled on top of each other.

In the IC floor planning issue, there seem to be two types of modules that may be used: hard and soft modules. The size and dimensions of hard modules are fixed. The area of soft modules is set, but the dimensions are not. The Diagrammatic representation of floor planning model is given in Fig.5.

Figure 5.
Diagrammatic representation of the projected floor planning model



4.1 Non-Overlapping Constraint

An important characteristic of floor planning is the achievement of making floor planning feasible with absolute zero violation. For there to be no overlap between m_i and m_j, any one of the modules between m_i and m_j must be satisfied. The modules should be completed in the phases below, with their rectilinear coordinates as stated. The coordinates are given in the x and y directions, with w_i signifying the width in the x direction and h_i denoting the height in the y direction of the i^{th} module.

Step 1: The restriction specification would be $x_i + w_i <= x_j$, whenever the comparative placement of m_i in non-overlapping has been on the left side of m_j.

Step 2: The constraint description will indeed be $x_i - w_i >= x_j$ once the comparative placement of m_i in non-overlapping seems to be on the right hand side of m_j.

Step 3: The constraint formulation would be $y_i - h_i >= y_j$ when the comparative positioning in non-overlapping of m_i would be at the base of m_j.

Step 4: The restriction formulation will be $y_i + h_i >= y_j$ once the relative placement in non-overlapping of m_i has been at the apex of m_j.

4.1.1 Estimation of Wire Length & Area

Analyzing the length of wire between two connection modules is assessed by assessing the centers of Euclidean distance from an automated standpoint. Consider the necessary wire in each of the connections when calculating the total length of wire for the floor. Eq.(4) and Eq. (5) is used to calculate the length of wire between module centers, where c_{ij} the point of connection between is i and j blocks.

$$wirelength\left(f\right) = \sum_{k=1}^{n}\sum_{m=1}^{n} c_{ij} d_{ij} \tag{4}$$

$$A_i = W_i * H_i \tag{5}$$

The formula is used to compute the Manhattan distances d_{ij} between rectangular blocks i and j. The floor plan's total area is determined as the little area of a rectangle that is covered in each of the blocks.

4.2 Proposed Optimization Model

For optimal floor planning, a new optimization model has been developed by blending the concepts of DOX (Chen et al. 2019) with ALO (Bairwa et al. 2021), respectively. These both optimization algorithms are good at solving complex optimization problems with higher convergence. Moreover, to acquire the global solutions within the search space, and to enhance the convergence rate of the solutions to its utmost potential, the DOX model has been introduced within the ALO model. Therefore, the projected model is referred to as the DUALO model. DUALO methods are manifested below:

Step 1: Randomly initialize the search agents (ants and antlions). Initialize the current iteration itr and the maximal iteration count as $\max^{itr}$

Step 2: Compute the fitness of the search agent using Eq. (2).

Step 3: Based on the fitness function, the best ant is identified; and it is denoted as the elite
Step 4: While $itr < \max^{itr}$ do
Step 5: For all search agents do
Step 6: With the Roulette wheel select an antlion
Step 7: Update c, d using the Sliding ants towards antlion. The corresponding expression is given in Eq. (6) and Eq. (7), respectively.

$$c^{itr} = \frac{c^{itr}}{I} \tag{6}$$

$$d^{itr} = \frac{d^{itr}}{I} \tag{7}$$

Here, c^{itr} and d^{itr} points to the smallest and largest values for each variable at the iteration itr.

Step 8: A random walk is created and it is normalized using Eq. (8) and Eq. (9). Eq. (8) corresponds to the random walk exhibited by the ants during their movement toward the food.

$$\begin{aligned} X(itr) = [0, cumsum(2.r(itr_1) - 1), cumsum(2.r(itr_2) - 1), ... \\ ..., cumsum(2.r(itr_n) - 1)] \end{aligned} \tag{8}$$

Moreover, to keep the random walks within the defined search space, the min-max normalization function has been utilized. This min-max normalization is manifested in Eq. (9).

$$X_i^{itr} = \frac{\left(X_i^{itr} - a_i\right) * \left(d_i - c_i^{itr}\right)}{\left(d_i - a_i\right)} + c_i \tag{9}$$

Here, a_i denotes the random walk's minimal value corresponding to the i^{th} variable, c_i^{itr} corresponds to the i^{th} variable's minimal value at itr^{th} iteration and d_i^{itr} corresponds to the i^{th} variable's minimal value at itr^{th} iteration.

Step 9: Ant's position is updated. Our contribution resides in this phase. Here, a new mathematical expression has been formulated as shown in Eq. (10). This expression corresponds to the guarantying of the movement of the random walk within the search space. To make this more efficient a sigmoidal chaotic map function is implied.

$$Ant_i^{itr} = \frac{\lambda R_A^{itr} + R_E^{itr}}{2} * W \tag{10}$$

$$W = \exp\left[-\frac{itr}{\max^{itr}}\right] \tag{11}$$

Here, λ points to the function corresponding to the sigmoidal chaotic map. The notation R_A^{itr} and R_E^{itr} corresponds to the roulette wheel-based random walk exhibited by the search agent around the antlion itr^{th} iteration and the random walk exhibited around the elite at itr^{th} iteration. In addition,

at itr^{th} iteration, the position of i^{th} ant is denoted as Ant_i^{itr}. The value of R_A^{itr} is computed using Eq. (8) and R_E^{itr} is computed using the Scavenger model of DOX (proposed). Dingoes engage in scavenging behavior when they come upon the carrion to consume while out and about in their habitat. This model is expressed mathematically in Eq. (12).

$$X_i^{itr+1} = \frac{1}{2}\left[e^{C_2} * X_{r_1}^{itr} - (-1)^{\beta} * X_i^{itr}\right] \tag{12}$$

Here, X_i^{itr+1} point to the movement of the dingo, χ_2 is a random value generated between [0,1] and r_1 is the random value generated from 1 to maximal solution size. In addition, β is the binary random value.
Step 10: End for
Step 11: Compute the fitness of the search agent using Eq. (2).
When a new and is found to be fitter than the existing one, then it is updated using the "Catching prey and re-building the pit" mechanism. This is shown mathematically in Eq. (13).

$$Antlion_j^{itr} = Ant_i^{itr}; iff\left(Ant_i^{itr}\right) > f\left(Antlion_j^{itr}\right) \tag{13}$$

Here, $Antlion_j^{itr}$ and Ant_i^{itr} points to the i^{th} antlion's position at itr^{th} iteration and points to the j^{th} antlion's position at itr^{th} iteration
Step 13: The elite is updated until an antlion becomes fitter than the elite
Step 14: End while
Step 15: Return elite

5. RESULTS AND DISCUSSION

5.1 Simulation Procedure

In MATLAB, the suggested hybrid optimization model referred to as DUALO for resolving issues with floor planning in VLSI was implemented, and the outcomes are achieved. The analysis was carried out for lattice filter, 3rd order FIR, 4th order FIR and 5th order FIR filters, respectively. The testing was carried out by altering the number of iterations (DUALO) from 0, 100, and 200, respectively. The projected model is compared with the existing models like ISI-APSO (Vinay Kumar et al. 2019), ACO, SA, ALO, and DOX in terms of convergence analysis, overlap check, rectangle area, and wire length as well.

5.2 Filter Description

For 3rd, 4th and 5th order FIR filters and lattice filters, the floor planning is accomplished using the proposed hybrid optimization model.

3rd order FIR filter: The dimensions of "Adder, Multiplier, and Differentiator" for the 3rd order FIR filter is manifested in Table 2. Fig 6 shows the Circuit diagram of the 3rd Order FIR filter. The 3rd Order FIR filter exhibiting the wire connection between the modules is shown in Fig.7. In addition, the 3rd Order FIR filters: Dimensions and forward path wire connection are manifested in Table 3.

4th order FIR filter: the dimensions of "Adder, Multiplier, and Differentiator" for the 4th order FIR filter are manifested in Table 4. Fig 8 shows the Circuit diagram of the 4th Order FIR filter. The 4thOrder FIR filter exhibiting the wire connection between the modules is shown in Fig.9. In addition, the 4thOrder FIR filter: Dimensions and forward path wire connection is manifested in Table 5.

5th order FIR filter: the dimensions of "Adder, Multiplier, and Differentiator" for the 5th order FIR filter are manifested in Table 6. Fig. 10 shows the digrammatic representation of the 5th Order

Figure 6.
Circuit diagram of 3rd Order FIR filter

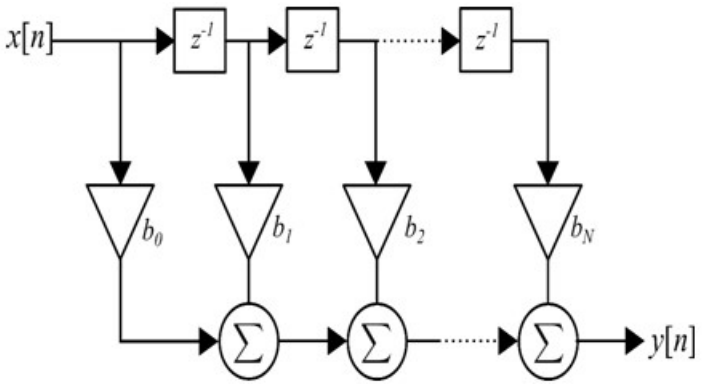
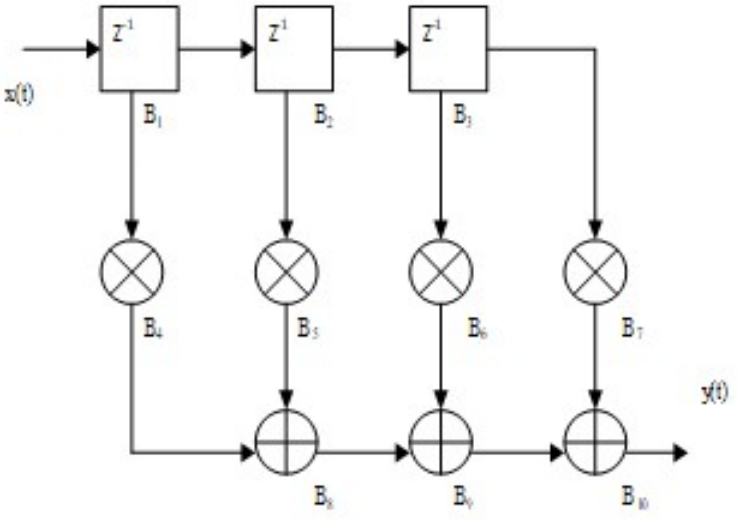


Table 2.
3rd Order FIR filter: module description

Module	Length	Width
Adder	150	100
Multiplier	200	150
Differentiator	50	50

Figure 7.
3rd Order FIR filter exhibiting the wire connection between the modules



FIR filter. The 5thOrder FIR filter exhibiting the wire connection between the modules is shown in Fig.11. In addition, the 5thOrder FIR filters: Dimensions and forward path wire connection are manifested in Table 7.

Table 3.
3rd Order FIR filter: dimensions and forward path wire connection

module	Length(unit length)	Width(unit length)	Wire connection
B1	0.5	0.5	B2,B5
B2	0.5	0.5	B3,B6
B3	0.5	0.5	B7
B4	2	1.5	B8
B5	2	1.5	B8
B6	2	1.5	B9
B7	2	1.5	B10
B8	1.5	1	B9
B9	1.5	1	B10
B10	1.5	1	B2,B5

Figure 8.
Circuit diagram of 4th Order FIR filter

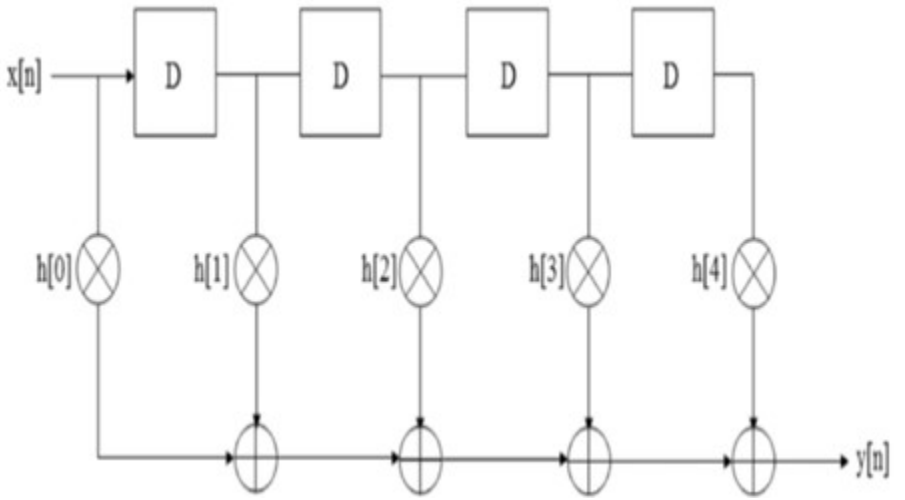


Table 4.
4th Order FIR filter: module description

Module	Length	Width
Adder	150	100
Multiplier	200	150
Differentiator	50	50

Fig 12 shows the Circuit diagram of 8th Order FIR filter.

Lattice filter: the dimensions of "Adder, Multiplier and Differentiator" for lattice filter is manifested in Table 8. The Lattice filter exhibiting the wire connection between the modules is manifested in Fig.13. Fig 14. Shows the Lattice filter exhibiting the wire connection between the modules

Figure 9.
4th Order FIR filter exhibiting the wire connection between the modules

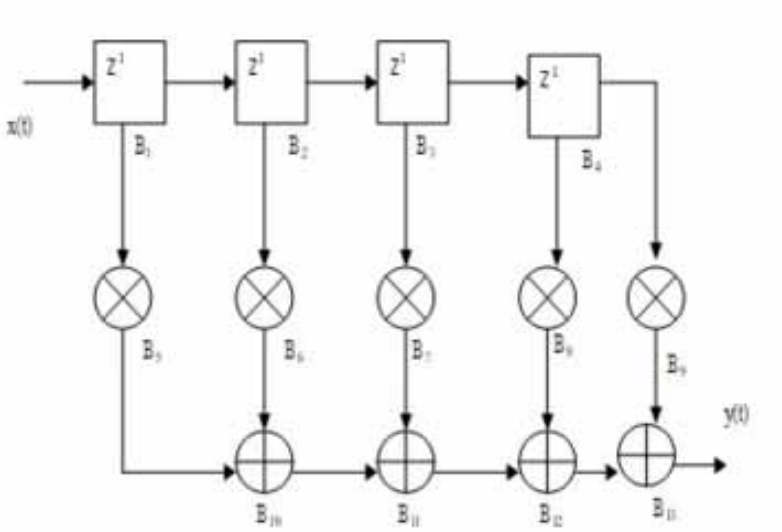


Table 5.
4th Order FIR filter: dimensions and forward path wire connection

module	Length(unit length)	Width(unit length)	Wire connection
B1	0.5	0.5	B2,B6
B2	0.5	0.5	B3,B7
B3	0.5	0.5	B4,B8
B4	0.5	0.5	B9
B5	2	1.5	B10
B6	2	1.5	B10
B7	2	1.5	B11
B8	2	1.5	B12
B9	2	1.5	B13
B10	1.5	1	B11
B11	1.5	1	B12
B12	1.5	1	B13
B13	1.5	1	

5.3 Convergence Analysis

A novel optimisation known as the DUALO method has been presented in this study endeavour. for successful floor planning with minimized cost. In literature (Beno et al. 2014) [31](Malik 2020), it has been suggested that the hybrid optimization models are good in enhancing the convergence of the solutions and thereby they aid in achieving the defined objective function. However, it's crucial to verify the predicted model's effectiveness in attaining the optimal solution specified in Eq (2). In this research work, cost minimization is the overall objective of this research work. The approach that achieves the minimized cost function is said to be the most optimal one. Here, we've accomplished the convergence analysis for the "lattice filter, 3rd order FIR filter, 4th order FIR filter, and 5th order

Figure 10.
Circuit diagram of 5th Order FIR filter

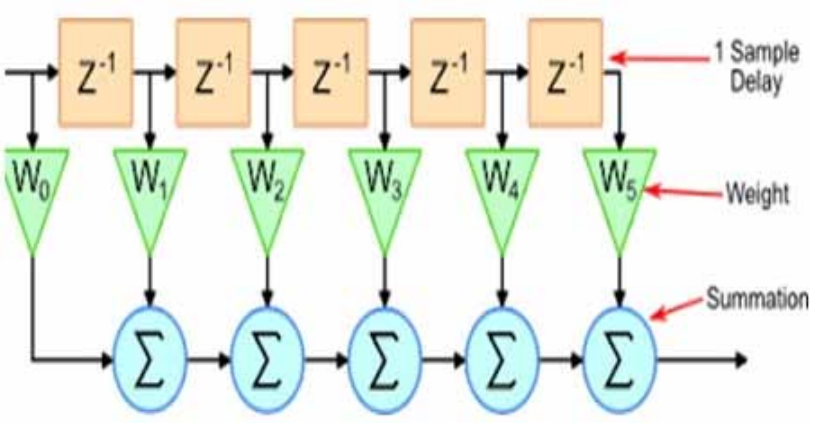
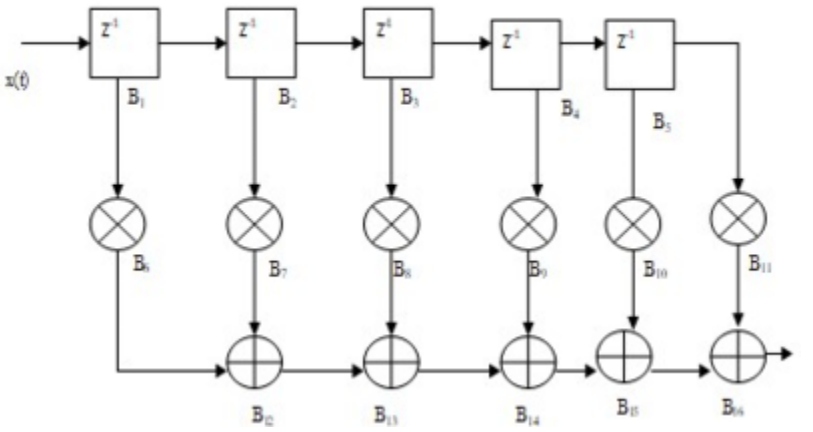


Table 6.
5Th Order FIR filter: module description

Module	Length	Width
Adder	150	100
Multiplier	200	150
Differentiator	50	50

Figure 11.
5th Order FIR filter exhibiting the wire connection between the modules

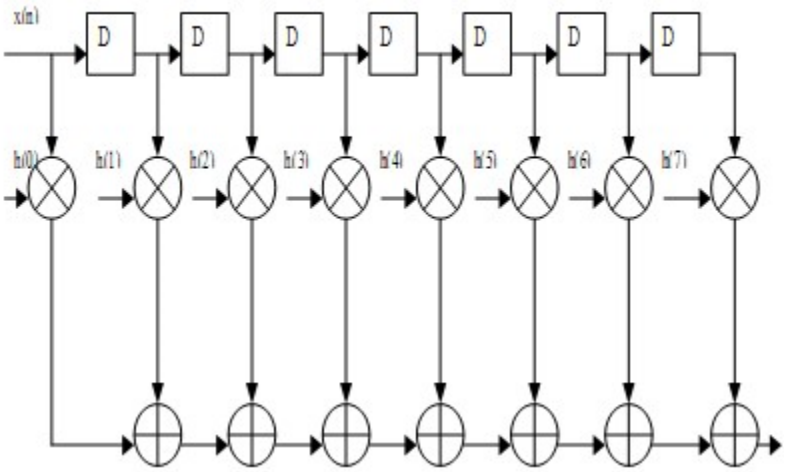


FIR filter", respectively. Fig. 15 shows the analysis of results. The projected model has achieved the least value in all aspects. This is only due to the introduction of the ensured ransom walks of the search agents. In the case of the 3rd-order FIR filter, the cost function of the projected model is lower than ISI-APSO, ACO, SA, ALO, and DOX models, respectively. This optimal outcome has been

Table 7.
5th Order FIR filter: dimensions and forward path wire connection

module	Length(unit length)	Width(unit length)	Wire connection
B1	0.5	0.5	B2,B7
B2	0.5	0.5	B3,B8
B3	0.5	0.5	B4,B9
B4	0.5	0.5	B5,B10
B5	0.5	0.5	B11
B6	2	1.5	B12
B7	2	1.5	B12
B8	2	1.5	B13
B9	2	1.5	B14
B10	2	1.5	B15
B11	2	1.5	
B12	1.5	1	
B13	1.5	1	
B14	1.5	1	
B15	1.5	1	

Figure 12.
Circuit diagram of 8th Order FIR filter



recorded for every variation in the number of iterations. In addition, for the 4th order FIR filter, the cost function of the projected DUALO as well as the existing model is identified to be higher for minimal iteration count. After the 50th iteration count, the DUALO has achieved the most minimal cost function, when compared to the existing works. Until the 50th iteration count to the cost function of the projected model is lower than certain existing models like DOX. At the 200th iteration count, the DUALO has achieved the minimal count function as 58, which is the best score when compared to ISI-APSO=88, ACO=85, SA=72, ALO=85, and DOX=68. In addition, the projected model has been found to achieve the defined objective function effectively over every variation in the iteration count in the case of 4th order FIR filter and the 5th-order FIR filter. This is only due to the introduction of the ensured ransom walks of the search agents. It proposes a brand-new hybrid multi-

Figure 13.
Circuit diagram of lattice filter

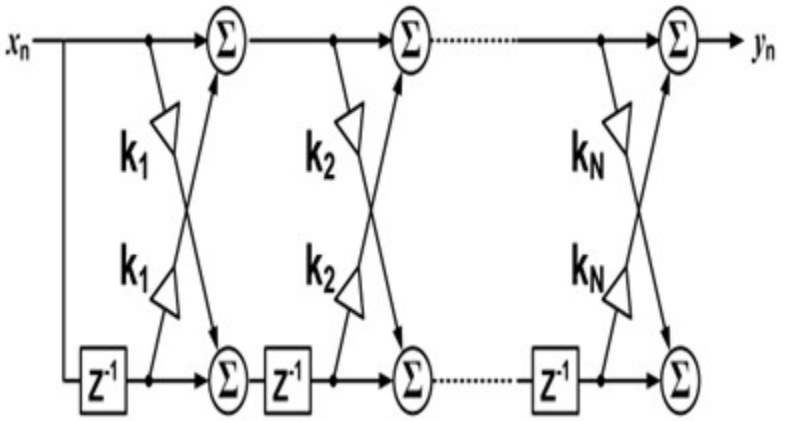
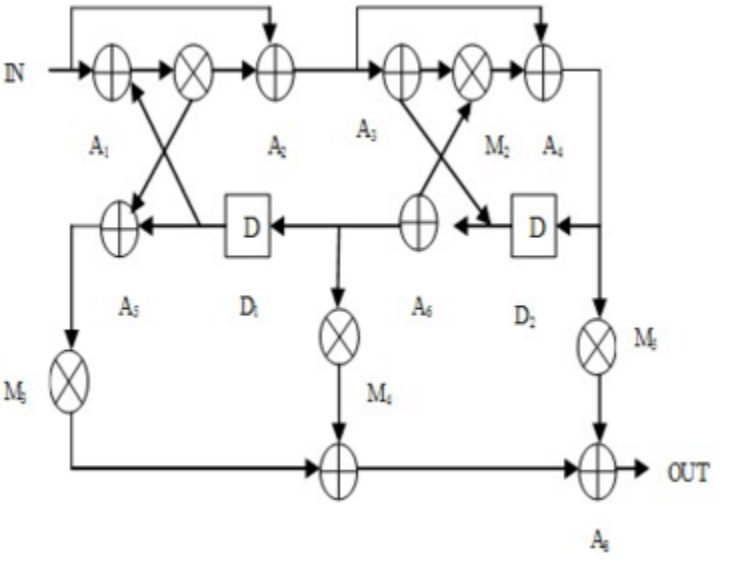


Table 8.
Lattice filter: module description

Module	Length	Width
Adder	150	100
Multiplier	200	150
Differentiator	50	50

Figure 14.
Lattice filter exhibiting the wire connection between the modules

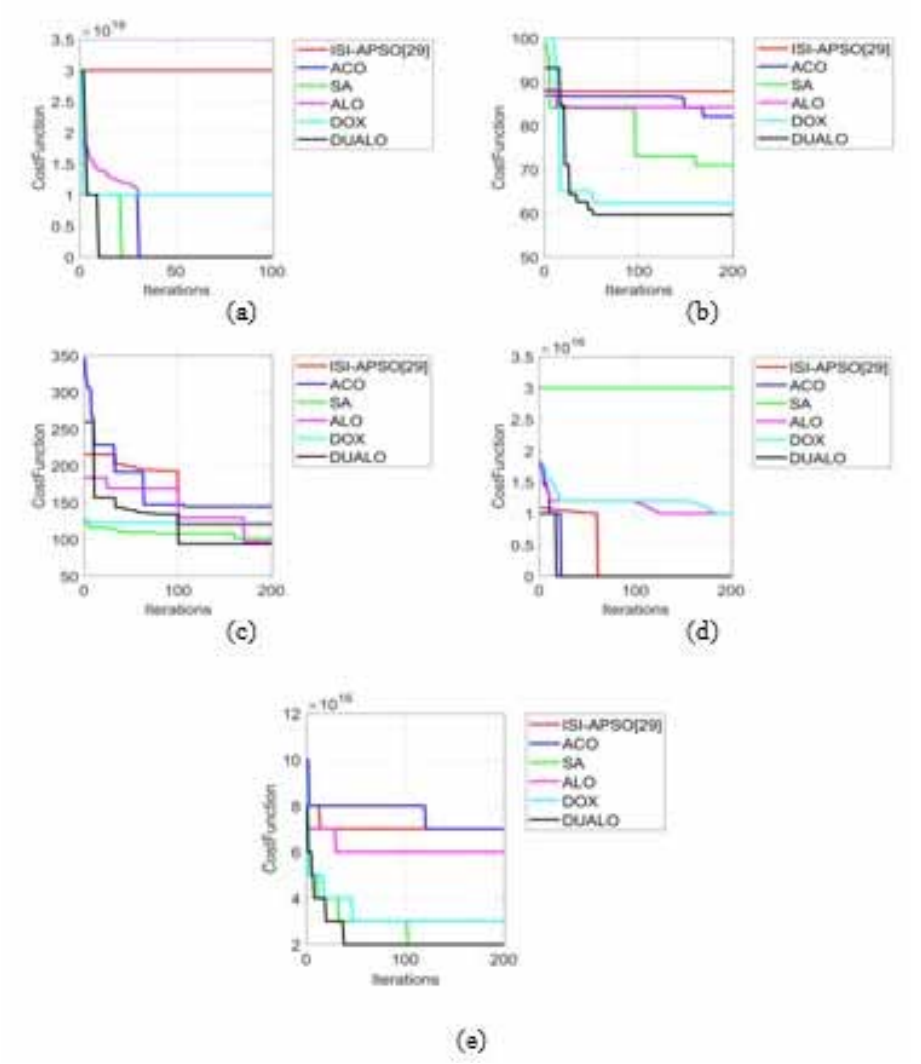


objective optimization technique to address the floor plan optimization problem. The proposed hybrid optimization known as the Dingo Updated Ant Lion Optimization (DUALO) model conceptually combines standard DOX and ALO. It takes into account multiple objectives including wire length, area, and penalty function. The optimal wire size and area calculation reduce the amount of space in the floor plan. As a result, the layout space is smaller. Therefore, with the projected model, efficient floor planning can be achieved with minimal cost.

5.4 Analysis on Overlap

The overlapping among the rectangular box in the VLSI clip needs to be minimal, to lessen the interconnection cost. To prove that our projected model satisfies the non-overlap constraint, we've validated our model on "lattice filter, 3rd order FIR filter, 4th order FIR filter, and 5th order FIR filter", respectively. Fig.16 demonstrates the resultant outputs. Following observation of the obtained results,

Figure 15.
Convergenence analysis of "(a) lattice filter, (b)3rd order FIR filter, (c) 4th order FIR, (d) 5th order FIR filter and (e) 8th order FIR filter"

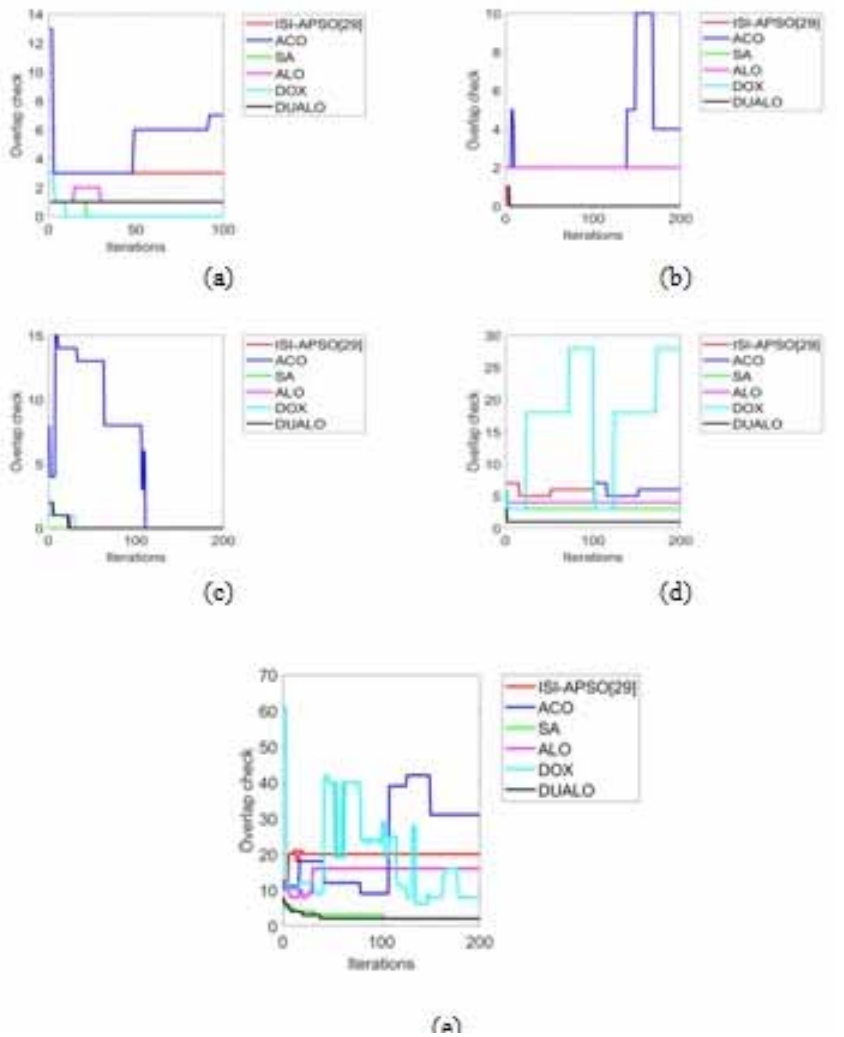


the predicted models has exhibited minimal overlap under lattice filter, 3rd order FIR filter, 4th order FIR filter, and 5th order FIR filter, respectively. This is owing to the optimal positioning of the modules in the floor plan with the fresh optimization approach. In the case of the 3rd-order FIR filter and the 4th-order FIR filter, the projected model has achieved the most minimal overlap value, which is close to zero (0). In addition, under the 5th order FIR filter evaluation, the suggested method has achieved the minimal overlap value of 2, which is better than ISI-APSO=10, ACO=8, SA=4, ALO=5, and DOX=30 at the 200th iteration. The newly proposed higher convergent optimization model is mostly responsible for this development. This model has ensured the random walk and has utilized the chaotic map that is good in avoiding the local optima as well as speeding the convergence of the solutions.

5.5 Analysis of Rectangular Area

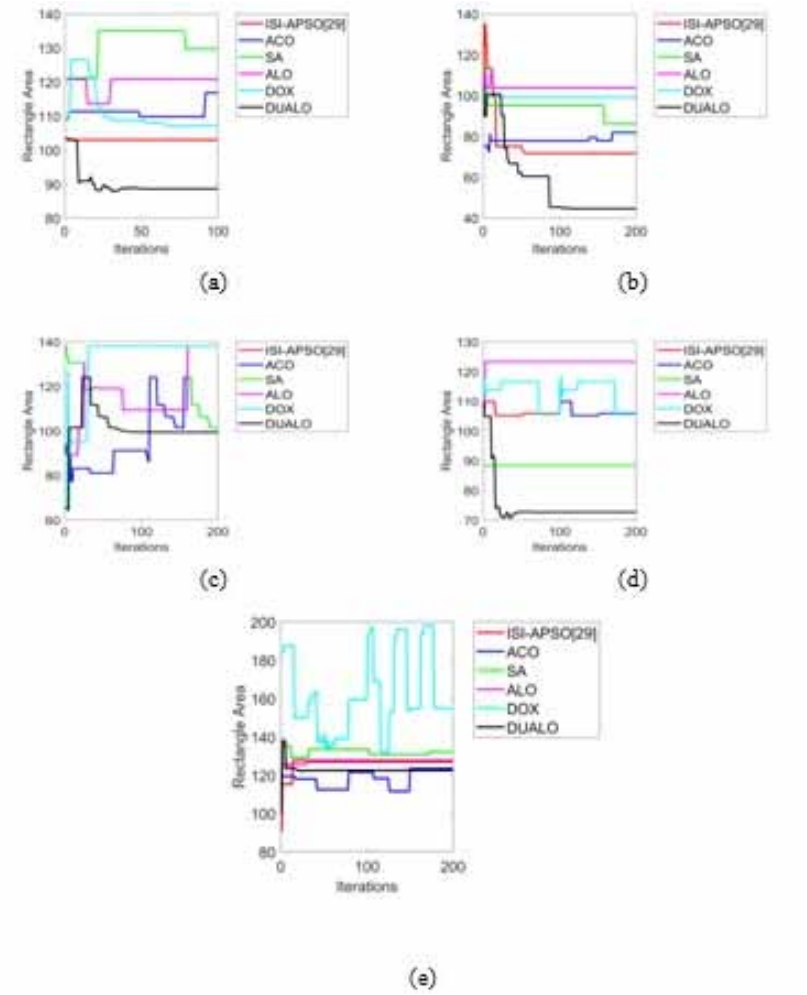
An important constraint of this research work is to make use of the clip area optimally, thereby reducing the cost incurred for floor planning. The rectangular area consumed by each of the modules needs to be

Figure 16.
Overlap analysis of "(a) lattice filter, (b)3rd order FIR filter, (c) 4th order FIR, (d) 5th order FIR filter and (e) 8th order FIR filter"



lower so that the floor planning can be accomplished efficiently in a compact manner. In this research work, the rectangle area incurred by the projected as well as existing models during optimal floor planning under "lattice filter, 3rd order FIR filter, 4th order FIR filter and 5th order FIR filter" is manifested. By changing the number of repetitions from 0, 50, and 100, correspondingly, this assessment is carried out. Fig.17 illustrates the results of rectangular area .The required results are obtained under lattice filter, 3rd order FIR filter, 4th order FIR filter, and 5th order FIR filter, the projected model has consumed the lower rectangular area, and this is owing to the utilization of the chaotic map for enhancing the convergence speed of the solutions. For every variation in the iteration count for the lattice filter, it is discovered that the projected model consumes substantially less rectangle area than the current models. In the case of the 3rd order FIR filter, the rectangular area of the projected as well as the existing model has been identified to be higher at the lowest iteration count (i.e. 0th iteration count). As the iteration count increases, the rectangular area consumed tends to lessen. At the 200th iteration, the projected model recorded the least value as 42, which is the optimal one. Moreover, in the case of the 4th order FIR filter and 5th order FIR filter, the suggested method has captured the least rectangular area. Consequently, it is claimed that the predicted model is ideal for efficient floor planning.

Figure 17.
Analysis on rectangular area of "(a) lattice filter, (b)3rd order FIR filter, (c) 4th order FIR, (d) 5th order FIR filter and (e) 8th order FIR filter"



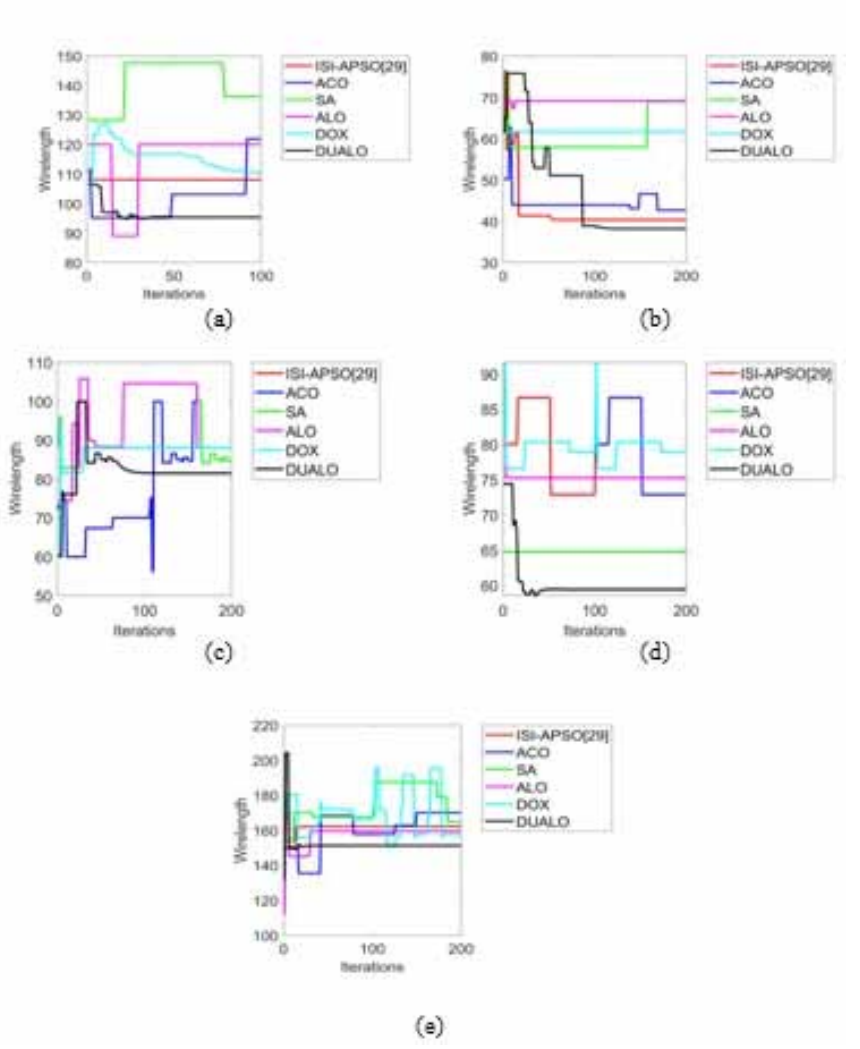
5.6 Analysis on Wirelength

The wire length consumed by the projected model for efficient floor planning is investigated in this section. This evaluation has been undergone for "lattice filter, 3rd order FIR filter, 4th order FIR filter and 5th order FIR filter". The corresponding results are furnished in Fig. 18. The recorded results are capturing, the projected model has exhibited the consumption of the less wire length. This reduction owes to the utilization of the guaranteed random walks by the solutions. In the case of the 5th order FIR filter, the projected model has consumed the least wire length as 60 at the 200th iteration, which is the best value than the existing models ISI-APSO=80, ACO=75, SA=65, ALO=74, and DOX=80.

5.7 Effects on Varying Population Size

The floor planning evaluation in VLSI by changing the population numbers for "lattice filter, 3rd order FIR filter, 4th order FIR filter and 5th order FIR filter" are given in Table 9 and Table 10, respectively.

Figure 18.
Analysis on wire length of "(a) lattice filter, (b)3rd order FIR filter, (c) 4th order FIR, (d) 5th order FIR filter and (e) 8th order FIR filter"



5.8 Analysis of Area and Wire Length

Table 11 shows that the analysis results of area and length. The total count of iterations is 200, the population size is 200, and the FIR order has been varied.

6. CONCLUSION

This paper has projected the new DUALO-based multi-objective model for solving the floor plan optimization issue. Conceptually, the proposed hybrid optimization known as the DUALO model combines DOX and ALO. The optimal wire length estimate and optimal partitioning have been carried out using the newly suggested multi-objective hybrid optimization. The space in the floorplan

Table 9.
Analysis of floor planning techniques for various population sizes for 3rd order FIR filter, 4th order FIR filter, 5th order FIR filter, and 8th order filter

Pop_size=200 and iteration count =200	DUALO	DOX	ALO	SA	ACO	ISI-APSO(Vinay Kumar et al. 2019)
3rd order FIR	59.682	62.293	84.264	71.024	82.114	87.884
4th order FIR	94.041	122.99	96.539	101.25	144.61	120.14
5th order FIR	99.463	1.00E+16	1.00E+16	3.00E+16	209.81	112.52
8th order FIR	2.00E+16	3.00E+16	6.00E+16	2.00E+16	7.00E+16	7.00E+16

Table 10.
Analysis of floor planning techniques for various population sizes for Lattice Filter

Pop_size=200 and iteration count =200	DUALO	DOX	ALO	SA	ACO	ISI-APSO(Vinay Kumar et al. 2019)
LatticeFilter	108.16	1.00E+16	1.00E+16	131.67	202.95	3.00E+16

Table 11.
Analysis of results in area and wire length is demonstrated

Area (sq.U.L)						
Pop_size=200 and iteration count =200	**DUALO**	**DOX**	**ALO**	**SA**	**ACO**	**ISI-APSO(Vinay Kumar et al. 2019)**
3th order FIR	38.214	38.214	38.214	38.214	38.214	38.214
4th order FIR	81.462	81.462	81.462	81.462	81.462	81.462
5th order FIR	59.41	154.63	59.41	59.41	59.41	59.41
8th order FIR	122.47	151.21	126.85	132	123.48	127.61
Wire length (sq.U.L)						
3th order FIR	44.597	44.597	44.597	44.597	44.597	44.597
4th order FIR	99.432	99.432	99.432	99.432	99.432	99.432
5th order FIR	72.706	72.706	72.706	72.706	72.706	72.706
8th order FIR	151.21	157.81	159.47	164.89	170.02	162.19

is decreased as a result of the optimal wire length estimation. Therefore, the layout space is reduced. The evaluation has been carried out in terms of convergence analysis, and wire length as well. As the iteration count increases, the rectangular area consumed tends to lessen. Interestingly, a gradual fall captured in the predicted schemes at the 80^{th} iteration count. In 200^{th} iteration, the model of projections captured the fewest value as 42, which is the optimal one. Furthermore, in the case of the 4th order FIR filter and 5^{th} order FIR filter, the least rectangular area was captured by the predicted model. Moreover, the presented method is said to be optimal for successful floor planning.

REFERENCES

Andrukhiv, A., Sokil, M., Fedushko, S., Syerov, Y., Kalambet, Y., & Peracek, T. (2020). Methodology for increasing the efficiency of dynamic process calculations in elastic elements of complex engineering constructions. *Electronics (Basel)*, *10*(1), 40. doi:10.3390/electronics10010040

Bairwa, A. K., Joshi, S., & Singh, D. (2021). Dingo optimizer: A nature-inspired metaheuristic approach for engineering problems. *Mathematical Problems in Engineering*, *2021*, 2021. doi:10.1155/2021/2571863

Beno, M. M., I. R, V., S. M, S. & Rajakumar, B. R. (2014). Threshold prediction for segmenting tumour from brain MRI scans. *International Journal of Imaging Systems and Technology*, *24*(2), 129–137. doi:10.1002/ima.22087

Chen, S., Ge, M., Li, Z., Huang, J., Xu, Q., & Wu, F. (2019). Generalized fault-tolerance topology generation for application-specific network-on-chips. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, *39*(6), 1191–1204. doi:10.1109/TCAD.2019.2952134

Chen, S., Huang, J., Xu, X., Ding, B., & Xu, Q. (2018). Integrated optimization of partitioning, scheduling, and floor planning for partially dynamically reconfigurable systems. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, *39*(1), 199–212. doi:10.1109/TCAD.2018.2883982

Choudhury, S. R., & Pradhan, S. N. (2019). DOTFloor–A diffusion oriented time-improved floorplanner for macrocells. *IEEE Access : Practical Innovations, Open Solutions*, *7*, 172074–172087. doi:10.1109/ACCESS.2019.2956595

Giorgini, M., Aleotti, J., & Monica, R. (2018). Floorplan generation of indoor environments from large-scale terrestrial laser scanner data. *IEEE Geoscience and Remote Sensing Letters*, *16*(5), 796–800. doi:10.1109/LGRS.2018.2880042

Guler, A., & Jha, N. K. (2020). McPAT-Monolithic: An area/power/timing architecture modeling framework for 3-D hybrid monolithic multicore systems. *IEEE Transactions on Very Large Scale Integration (VLSI). Systems*, *28*(10), 2146–2156.

Guo, P., Hou, W., Guo, L., Yang, Q., Ge, Y., & Liang, H. (2018). Low insertion loss and non-blocking microring-based optical router for 3D optical network-on-chip. *IEEE Photonics Journal*, *10*(2), 1–10. doi:10.1109/JPHOT.2018.2796094

Huang, Z., Lin, Z., Zhu, Z., & Chen, J. (2020). An improved simulated annealing algorithm with excessive length penalty for fixed-outline floorplanning. *IEEE Access : Practical Innovations, Open Solutions*, *8*, 50911–50920. doi:10.1109/ACCESS.2020.2980135

Khan, N., Castro-Godínez, J., Xue, S., Henkel, J., & Becker, J. (2020). Automatic Floor planning and Standalone Generation of Bitstream-Level IP Cores. *IEEE Transactions on Very Large Scale Integration (VLSI). Systems*, *29*(1), 38–50.

Kouchi, T., Kakoi, M., Kumazaki, N., Sugahara, A., Imamoto, A., Kajiyama, Y., Terada, Y., Sanad, B., Kanagawa, N., Kodama, T., Fukuda, R., Komai, H., Asaoka, N., Ohnishi, H., Isomura, R., Handa, T., Yamamoto, K., Ishizaki, Y., Deguchi, Y., & Yoshihara, M. (2020). A 128Gb 1-bit/Cell 96-Word-Line-Layer 3D Flash Memory to Improve the Random Read Latency With tProg= 75 μs and tR= 4 μs. *IEEE Journal of Solid-State Circuits*, *56*(1), 225–234. doi:10.1109/JSSC.2020.3028393

Lin, J. M., Chang, W. Y., Hsieh, H. Y., Shyu, Y. T., Chang, Y. J., & Lu, J. M. (2021b). Thermal-Aware Floor planning and TSV-Planning for Mixed-Type Modules in a Fixed-Outline 3-D IC. *IEEE Transactions on Very Large Scale Integration (VLSI). Systems*, *29*(9), 1652–1664.

Lin, J. M., Chen, T. T., Hsieh, H. Y., Shyu, Y. T., Chang, Y. J., & Lu, J. M. (2021a). Thermal-aware fixed-outline floor planning using analytical models with thermal-force modulation. *IEEE Transactions on Very Large Scale Integration (VLSI). Systems*, *29*(5), 985–997.

Malik, N. (2020). Enery-aware routing in MANET: Hybrid genetic and group search algorithm. *Journal of Networking and Communication Systems*, *3*(4), 35–41.

Mandala, J. & Sekhara Rao, M. C. (2019). HDAPSO: Enhanced privacy preservation for health care data. *Journal of Networking and Communication Systems*, *2*(2), 10–19.

Mirjalili, S. (2015). The ant lion optimizer. *Advances in Engineering Software*, *83*, 80–98. doi:10.1016/j.advengsoft.2015.01.010

Mohapatra, S., Vendra, S. K., & Chrzanowska-Jeske, M. (2020). Fast Buffer Count Estimation in 3D IC Floorplanning. *IEEE Transactions on Circuits and Wystems. II, Express Briefs*, *68*(1), 271–275. doi:10.1109/TCSII.2020.3007858

Prakash, A., & Lal, R. K. (2021). Floor planning for Area Optimization Using Parallel Particle Swarm Optimization and Sequence Pair. *Wireless Personal Communications*, *118*(1), 323–342. doi:10.1007/s11277-020-08015-5

Ren, Z., Alqahtani, A., Bagherzadeh, N., & Lee, J. (2020). Thermal TSV optimization and hierarchical floor planning for 3-D integrated circuits. *IEEE Transactions on Components, Packaging, and Manufacturing Technology*, *10*(4), 599–610. doi:10.1109/TCPMT.2020.2970382

Sadeghi, A., Lighvan, M. Z., & Prinetto, P. (2020). Automatic and Simultaneous Floor planning and Placement in Field-Programmable Gate Arrays with Dynamic Partial Reconfiguration Based on Genetic Algorithm. *Canadian Journal of Electrical and Computer Engineering*, *43*(4), 224–234. doi:10.1109/CJECE.2019.2962147

Sari, A., & Psarakis, M. (2017). Scrubbing-aware placement for reliable FPGA systems. *IEEE Transactions on Emerging Topics in Computing*, *8*(3), 564–576. doi:10.1109/TETC.2017.2757978

Shunmugathammal, M., Christopher Columbus, C., & Anand, S. (2020, February). (20208). A novel B* tree crossover-based simulated annealing algorithm for combinatorial optimization in VLSI fixed-outline floorplans. *Circuits, Systems, and Signal Processing*, *39*(2), 900–918. doi:10.1007/s00034-019-01054-9

Shunmugathammal, M., Columbus, C. C., & Anand, S. (2020a). A nature inspired optimization algorithm for VLSI fixed-outline floorplanning. *Analog Integrated Circuits and Signal Processing*, *103*(1), 173–186. doi:10.1007/s10470-020-01598-w

Singh, R. B., & Baghel, A. S. (2021). IC Floor planning Optimization using Simulated Annealing with Order-based Representation. *International Journal of Intelligent Systems & Applications*, *13*(2), 62–70. doi:10.5815/ijisa.2021.02.05

Singh, R. B., & Baghel, A. S. (2021). IC Floorplanning Optimization using Simulated Annealing with Order-based Representation. *International Journal of Intelligent Systems & Applications*, *13*(2), 62–70. doi:10.5815/ijisa.2021.02.05

Sivaranjani, P., & Senthil Kumar, A. (2015). Thermal-aware non-slicing VLSI floor planning using a smart decision-making PSO-GA based hybrid algorithm. *Circuits, Systems, and Signal Processing*, *34*(11), 3521–3542. doi:10.1007/s00034-015-0020-x

Srinivasan, B., & Venkatesan, R. (2021). Multi-objective optimization for energy and heat-aware VLSI floor planning using enhanced firefly optimization. *Soft Computing*, *25*(5), 4159–4174. doi:10.1007/s00500-021-05591-x

Teng, X., Guo, D., Guo, Y., Zhao, X., & Liu, Z. (2018). SISE: Self-updating of indoor semantic floorplans for general entities. *IEEE Transactions on Mobile Computing*, *17*(11), 2646–2659. doi:10.1109/TMC.2018.2812752

Vehring, S., Ding, Y., Scholz, P., & Gerfers, F. (2020). A 3.1-dBm E-band truly balanced frequency quadrupler in 22-nm FDSOI CMOS. *IEEE Microwave and Wireless Components Letters*, *30*(12), 1165–1168. doi:10.1109/LMWC.2020.3028053

Vinay Kumar, S. B., Rao, P. V., & Singh, M. K. (2019). Optimal floor planning in VLSI using improved adaptive particle swarm optimization. *Evolutionary Intelligence*, 1–14.

Vipin, K. (2019). Asyncbtree: Revisiting binary tree topology for efficient fpga-based noc implementation. *International Journal of Reconfigurable Computing*, *2019*, 2019. doi:10.1155/2019/7239858

Wang, J., Kang, Y., Wu, W., Xing, G., & Tu, L. (2020). DUPRFloor: Dynamic modeling and floor planning for partially reconfigurable FPGAs. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, *40*(8), 1613–1625. doi:10.1109/TCAD.2020.3022345

Zhang, S. Z., Zhao, Z. Y., Feng, C. C., & Wang, L. (2020). A Machine Learning Framework with Feature Selection for Floorplan Acceleration in IC Physical Design. *Journal of Computer Science and Technology*, *35*(2), 468–474. doi:10.1007/s11390-020-9688-x