International Journal of Reconfigurable and Embedded Systems (IJRES)
Vol. 14, No. 2, July 2025, pp. 365~374
ISSN: 2089-4864, DOI: 10.11591/ijres.v14.i2.pp365-374

An optimized simulated annealing memetic algorithm for power and wirelength minimization in VLSI circuit partitioning

P. Rajeswari, Smitha Sasi
Department of Electronics and Telecommunication Engineering, Dayananda Sagar College of Engineering, Bangalore, India

Article Info

Article history:

Received Oct 30, 2024
Revised May 7, 2025
Accepted Jun 10, 2025

Keywords:

Area
Evolutionary computation
Interconnection
Length of wire
Partitioning
Very large scale integration

ABSTRACT

The development of physical architecture standards for very large scale integration (VLSI) single and multichip platforms is still in its early stages. To deal with the growing complexity of modern VLSI systems, it has become common practice to split large circuit architectures into smaller, easier-to-manage sub-circuits. Circuit partitioning improves parallel modeling, testing, and system performance by lowering chip size, number of components and interconnects, wire length (WL), and delays. VLSI partitioning's primary goal is to split a circuit into smaller blocks with as few connections as possible between them. This is frequently accomplished by recursive bi-partitioning until the required complexity level is reached. Thus, partitioning is a fundamental circuit design challenge. An efficient remedy that offers a heuristic method that explores the design space to iteratively enhance outcomes is evolutionary computation. In order to minimize WL, area, and interconnections, we provide an optimized simulated annealing memetic algorithm (OSAMA) that combines local search methods with evolutionary tactics. The efficiency of the method was evaluated using criteria like runtime, cost, delay, area, and WL. OSAMA's ability for effective partitioning is demonstrated by experimental results, which confirm that it dramatically lowers important design parameters in VLSI circuits.

Corresponding Author:

P. Rajeswari
Department of Electronics and Telecommunication Engineering, Dayananda Sagar College of Engineering
Bangalore-78, Karnataka, India
Email: prajeswarisugans@gmail.com

1. INTRODUCTION

The integrated circuits (IC) known as very large scale integration (VLSI) circuits contain multiple transistors and other electronic components on an individual chip. Such circuits, which are frequently found in electronic gadgets that range from computers and cell phones to industrial equipment and automobiles, are created to carry out particular occupations or activities. In enabling the development of more advanced and potent electronic components, VLSI technology has played a crucial role in changing the electronics sector [1]. Modern technology has benefited greatly from the breakthroughs in computer, networking, and consumer electronics that VLSI circuits have made possible. Moore's Law has directed the constant advancement of VLSI technology, which has resulted in ever-increasing processing power and efficiency, enabling the development of complex electronic gadgets that are now an essential component of daily life [2]. The division of circuits into simpler and smaller components is an essential and crucial stage in VLSI architecture. To achieve optimal efficiency, save area and electrical consumption, and mitigate probable timing problems throughout chip design, circuit partitioning is essential. As technologies develop, VLSI circuit complexity rises, necessitating increasingly advanced partition strategies to satisfy the demanding

requirements of contemporary electrical devices [3]. An innovative shift in the power sector regarding “ICs” with higher interconnections and complexities has been brought about by the growth of VLSI circuitry. Since chip density increases, a range of challenging issues emerge that require being handled from the outset of the development phase, such as architecture clarity, evaluation, increased prevention, and optimal utilization of interconnects [4]. To manage these difficulties, improved design of physical tools is necessary. One of the most significant steps in the actual construction of circuits based on VLSI is the division of the netlist. To make planning, laying out, and testing easier, an electronic device is divided into smaller units called sub circuits [5]. Architecture design, logic design, practical layout, physical design, circuit design, packaging, and manufactureare the 8 interconnected stages of the VLSI design process. A summary requirement serves as the starting point of the process, and a complex layout that could have been tested, evaluated, and put into practice serves as the final product. This stage involves evaluating and adjusting the design's requirements about factors including performance, operability, manufacturing technology, and physical size [6]. This design level produces a register transfer level (RTL) description that is specified in a hardware description language (HDL). The employment of semiconductors, logic gates, wire, and other components is addressed at the development stage of the circuit design procedure. During the circuit design, the physical design level provides the circuit structure and converts it into an architectural specification. The next part goes into detail about this level. The design is then sent for manufacture, packaging, and testing of the IC as the following stage of design, to ensure that the system's requirements and standards have been fulfilled [7]. Reducing the number of exterior wires connecting each partition is an essential goal of the partitioned procedure. The following phase after partitioning is the design of floors and placement. The exact position of each of the sub-circuit elements within every partition is evaluated at this vantage point. The objectives of floor planning and installation are to arrange the elements in a manner that reduces the area by properly organizing their placement and to satisfy the practical requirements for the interconnectivity region. The result is sent to routing, where connections between the parts located inside partitions are made. The process of routing can be characterized as one that chooses the most efficient paths within a given routing region to connect the elements across or between partitions [8].

Oliveira *et al.* [9] conducted a thorough comparison of four partitioning tools, digging into how they fare on real-world benchmarks. Various hyper graph and edge-weighting diagram layout strategies are considered as they explore graph and hyper graph partitioning. Most VLSI computer-aided design (CAD) methods deal with NP-hard problems and have trouble scaling with larger and larger circuits. Rajeswari *et al.* [10] detailed how the genetic algorithm can be used to find the best solution with the fewest possible repetitions by the strategic use of controllable evolution at various phases and to do it in a way that has the least possible effect on the optimization outcome while maintaining a strong framework. Since the complexity of VLSI design has increased in tandem with its rapid development, they need new methods of miniaturized that require minimal human input.

Thakur *et al.* [11] introduced a simulation tool and methodology for performing quick direct current analysis of VLSI-based metal–oxide–semiconductor field-effect transistor circuits. A VLSI-based circuit's time-domain evaluation is calculated using a discrete constant pulse estimation. Luo *et al.* [12] provided a net clustering-based coarsening approach for k-way hypergraph division when network segmentation is employed to create a collection of beginning vertex with increased intrinsic similarities. VLSI circuit designing as well as other operations frequently employ hypergraph partitioning to lessen computational complexity in light of the ever-increasing size of semiconductor circuits. Lin *et al.* [13] presented an innovative approach in which the aim variable is no longer the biggest area of the boxes but rather their greatest diameter. The degenerate condition when certain rectangles' regions vanish could be problematic for this area-based paradigm. In the context of VLSI physical architecture, this scenario is irrelevant. Rodriguez *et al.* [14] focused on reducing the maximum path delay to its absolute minimum. Circuit partitioning for VLSI design is just one application of hypergraph partition. Such solutions perform by minimizing a min-cut function in which the number of hyperedges that must be cut is minimized.

Sebak *et al.* [15] suggested a heuristic method for locating a near-optimal VLSI circuit arrangement. The algorithm begins with a greedy starting position before beginning the iterative simulated annealing (SA) enhancement. Rajeswari *et al.* [16] provided a physical design that is a crucial part of VLSI circuit design since it determines how the circuit will be implemented on the chip. When designing an IC, the floorplan is used as a guide to determine where various components will go on the chip. Srinivasan *et al.* [17] provided an assessment of the architectural factors that can be used to lessen congestion throughout circuit manufacturing if utilized alongside the current architecture and optimization method used in the design of circuits engineering.

Lakshmanna *et al.* [18] provided a generic approach to translating the optimization of parameters into an application of reinforcement learning. Modern electronic design automation tools and VLSI physical layout could benefit from the automation of such variable settings for power-performance-area optimization.

Agnesina *et al.* [19] suggested a system for image-based generative learning for electrostatic analysis for estimating VLSI dielectric aging. VLSI dependability along with high-speed design of circuits depends on electromagnetic analysis that determines the potential of electricity and electrical field. Lamichhane *et al.* [20] intended to investigate different techniques that add to the problem of controlling aligning constraints, such as great position, ideal location, and quick run time. Utilizing optimization techniques, a study on the actual design of VLSI design is conducted to increase the performance of VLSI chips. Hussain and Kishore [21] demonstrated potential research possibilities and briefly explains relevant optimization methods and models that are frequently utilized in VLSI worldwide deployment. In the physical design of VLSI, location is one of the most important steps that affects how well later stages work.

Rajeswari and Chandra [22] suggested a successful lossless embedded compression (EC) technique to reduce memory bandwidth while maintaining visual quality. By using higher display resolution and pixel stages, the requirement for quality of vision has increased. However, in a video coding scheme, these two problems have greatly increased the bandwidth of memory. Lee *et al.* [23] addressed the issue of the growing complexity of VLSI chips. A large worldwide networking issue still exists with a trade-off between energy consumption and connectivity delay as a side effect of progress in the optimization of circuits using VLSI, which involves shrinking chip size from the micrometre to the nanoscale level and fabricating billions of devices on one semiconductor die. Nath *et al.* [24] presented an aggressive strategy for determining the starting design that may be applied by computerized annealing subsequently that uses a B* tree as the beginning design. The initial stage of VLSI construction is design [25].

The purpose of VLSI partitioning is to divide the circuit into numerous smaller circuits with few connections in between; this study proposes optimum partitions the VLSI circuit design using an optimised simulated annealing memetic algorithm (OSAMA) algorithm; the OSAMA algorithm's goal of reducing partitioning delay, reducing floor planning area, and reducing floor planning delay. The rest of the paper is organized as follows: section 2 illustrates circuit partitioning problem statement in VLSI circuits. Section 3 portrays the collection of data. The suggested circuit partitions and floor design in VLSI circuits utilizing the OSAMA algorithm is illustrated in section 3. Section 4 concludes this paper.

2. RESEARCH METHOD

2.1. Circuit partitioning issue solution

Graph partitioning issues are a typical notation for cases of circuit partitioning. Conventional mathematical representations of circuits are graphs, where nodes stand for individual circuit elements and edges reflect the relationships between components in Figure 1.

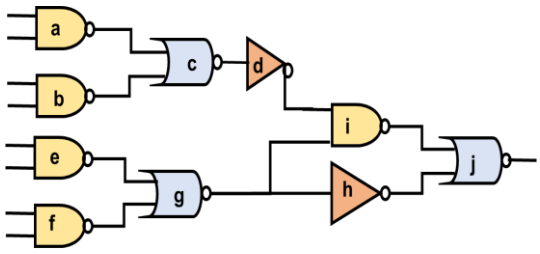

Figure 1. A logic circuit [16]

The following is a statement of the partitioning problem based on the graph concept: find $S = (C, A)$ for the vertices to $c \in C$through dimensions $g(c)$ related by border boundaries $a \in A$ with masses $u(a)$ are divided into r subsets $C_1, C_2, \ldots, C_r$ with $\bigcup_{j=1}^{r} C_j = C \; and \; C_j \cap C_i = 0 \; f \; or \; j \neq i$. It is possible to cut particular relations to divide the circuit depicted in Figure 2 into k subsets. The cost of the two partitions is the relevant metric, and it may be written as (1):

$$cost = \sum_{j=1}^{r} \sum_{i=1}^{r} V_{ji}, \text{ where } j \neq i \quad (1)$$

Dividing a collection into exactly two separate parts is the simplest possible partition problem. In contrast, the complexity of this example is already excessive despite identical nodes and unit border masses. Indeed, given a system with 2 m nodes, the number of neutral bipartitions is:

$$\sum_{j=0}^{r} n_j = \sum_{i=0}^{r} m_i \quad (2)$$

Where, j and i the vertices of edges need to be labelled and n_j and m_i are listings of nodes in each partition. Gain $\Delta s(C,U)$ in the process of exchanging two nodes C and U is the overall savings from this kind of trading rise. A positive outcome ($\Delta s > 0$) suggests a lessening of cost-cutting measures, contrasted with a positive loss ($\Delta s < 0$)suggests a rise in value. The benefit of exchanging nodes C and W is:

$$\Delta s(C,U) = J_c + J_u - 2a(C,U) \quad (3)$$

J_c and J_u increase at nodes C and U, and their equivalent nodes, and $a(C,U)$ represents the importance of the relationship among nodes C and U. In the event where C and U share an edge $a(C,U) = 1$; else, $a(C,U) = 0$.

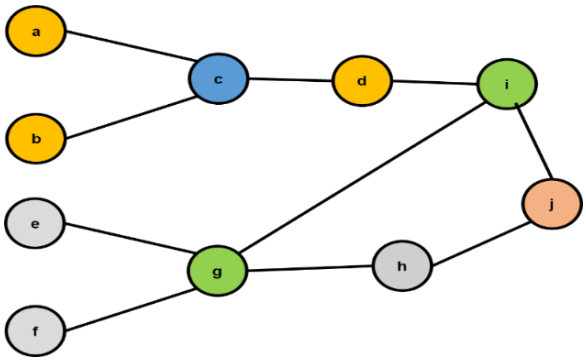

Figure 2. A graphical representation of the logic circuit [16]

The maximum gain a pair contributes in a particular phase with all possible node swapping is indicated by Δs_{max}. If each partition contains r nodes, then, the sum of all possible hops between nodes is r^2. Thus,

$$\Delta s_{max} = \max(\Delta s_j), \text{where} j = 1,2, \ldots r^2 \quad (4)$$

again, listing each partition and selecting the best one is impracticable for moderate quantities of n in (4), and it approaches difficult in greater circuits. Algorithms using heuristics are a useful tool for dealing with such issues. These algorithms typically produce an approach that is not optimal but is quick and easy to implement. All of these algorithms produce the same solutions to any specific problem. Stochastic algorithms, which depend on chance, produce different results for the same issue with each iteration. Various common partition issues can be solved using known stochastic techniques [16].

2.2. Data collection

To get rid of these two problems, it was proposed to employ a rank-based k-medoid technique in which the algorithm is prevented from locating local optimum sites at the beginning. Furthermore, the best clustering technique for large datasets was discovered to be the k-medoid method [26].

2.3. Optimised simulated annealing memetic algorithm

Both SA and memetic algorithms (MA) are optimization methods for locating approximations of solutions to challenging issues. To improve the search capabilities and effectiveness of the optimization process, an OSAMA can be created by combining these two methods. SA mimics cooling material to a low-energy state and is modelled after the metallurgical annealing procedure. It probabilistically searches the solution space and gradually progresses towards better solutions, allowing for sporadic uphill climbs to avoid local maxima. OSAMA is particularly well-suited for addressing issues with harsh and multimodal terrain thanks to this hybrid approach's ability to balance exploration and exploitation efficiently.

2.4. Simulated annealing

A probabilistic approach recurrent group of techniques includes SA. The metallic substance annealing process is modelled by this technique. The material is heated to extreme temperatures, providing the atoms with enough force to break their connections and allow permitted for movement. The metallic material

is then gradually melted to room temperature over its electrons to begin to crystallize in a highly organized condition. If we relate the division challenge with the technique of annealing, then achieving the global ideal, which includes the lowest cost, is analogous to achieving an efficient crystal framework. A basis, or condition, with a related expense in an equal-way divided problem, is the allocation of a circuit across two identical blocks. The local neighbor of a certain state G is represented by all other partitions created by switching elements between the partitions. The worst solution must occasionally be accepted to depart from the local optimal F to reach the worldwide optimal level S. A state with the lowest cost is certainly the worldwide optimal. The technique for simulating annealing examines the annealing procedure at a specific temperature, D. Initial division occurs to start the procedure. Go, at a starting temperature doconsequently; as the procedure progresses, the temperature is gradually lowered in a geometrical development:

$$D_j = \alpha . D_{j-1} \alpha < 1 \quad (5)$$

a recent partition $NewG$ is gained at every temperature D_j, and its cost, $Cost(NewG)$ is related to the prior cost, $Cost(G)$.

$$Cost(NewG) - Cost(G) \approx \Delta z \quad (6)$$

if the current value is beneficial, $\Delta z < I$ a novel method is accepted, if otherwise, then k is chosen randomly.

$$k = RANDOM(0,1) \quad (7)$$

$k < exp\left(-\frac{\Delta z}{D}\right)$ the mediocre answer will be accepted. The probability of the worst split being chosen is expressed as (8):

$$B = exp\left(-\frac{\Delta z}{D}\right) \quad (8)$$

annealing at different temperatures increases the probability of different outcomes,

$D \to \infty B \to 1$
$D \to 0B \to 0$

The probability it is that the cheaper (more expensive) split is going to be chosen is represented by the parameter B. The probability of occurring is expressed as an expression of two independent variables: the temperature D and the expense parameter variance Δz. Since the method takes time at each temperature, it will be performed multiple times at each setting. Time can be adjusted when temperatures drop according to:

$$Time = Time + N \quad (9)$$

We predicted the amount of time spent at each temperature using (9) with parameter N. What this means for us is the number of computations to be performed at various temperatures. There will be N iterations of the computation performed at the same temperature D during each moment.

2.5. Memetic algorithm

MAs are a type of evolutionary optimization that utilizes both GA and LS ideas. When applied to difficult optimization issues, like those found in VLSI design, these techniques excel. Successful applications of MA in the context of VLSI have included circuit segmentation, placement, routing, and floor planning. As the complexities of VLSI circuits continue to grow, they are ideally equipped to meet the difficulties they provide because of their capacity for combining worldwide exploration with local improvement. Algorithm 1 offers a general MA.

Algorithm 1. MA
Input:Variables Issue, and Limitations
Begin
Population ← $Init_{Pop}$(Parameters, Constraints);
  Repeat
    Fitness ← $l(Popultion)$;
Pop_{Cross} ← $Crossover(Population)$:

$Pop_{Mut} \leftarrow Mutation(Pop_{Cross})$
$Population \leftarrow LocalSearch(\ Pop_{Mut})$;
Until (Termination Criteria Satisfied):
End
Output:Ind* (Best Individual)
Take into account two spaces, G (for searching phenotypes) and V (for representing genotypes).

$$\rho: G \to V \tag{10}$$

This is a graphical operation that, responded to G, finds the corresponding chromosome in V. During this paper, we shall believe that b is injected despite being surjective. Let l represent the fitness function, which may be thought of as a mapping for our purposes:

$$l: V \to \mathbb{k}^{+} \tag{11}$$

assuming maximum fitness as the goal, we shall refer to the set of global optimums as V*⊂V.

Consider O to be a unary stochastic operator acting on V. For the time being, the stochastic component of such an operator can be accommodated by introducing a control O, from which a control parameter is drawn to determine which of the possible moves occurs. To change the jth bit of a binary string, for instance, one can employ a mask of bits as a parameter for control by setting that bit to 1 at position j. The resulting functional form of O will appear:

$$\mathcal{O}: G \times \mathcal{R}_{\mathcal{O}} \to G \tag{12}$$

$y \in V$ will be considered locally optimal in terms of O, or $\mathcal{O} - opt$, if an individual utilization of O cannot produce a chromosome using it that is fitter than y,

$$\forall \mathcal{R} \epsilon \mathcal{R}_{\mathcal{O}}: l\big(\mathcal{O}(y, r)\big) \leq l(y) \tag{13}$$

let V*⊂V be the set of o-opt chromosomes in V,

$$V_{\mathcal{O}} \underline{\Delta} \{y \in V | y is \mathcal{O} - opt\} \tag{14}$$

when applied to the problem of optimizing l over V, a genetic algorithm seeks to achieve a certain goal, such as discovering some or all optimum solutions in V* or rapidly improving approaching fitter chromosomes. The actual reality is, for every operator O on moves, therefore, recasting the search over V is acceptable. To define a hill-climber, we need an illustration space V, a move operation O, and a subspace V* of local optimums, and we will say that any stochastic, parameterized operation that, provided a chromosome V*⊂Vo, reaches the local optimum is a hill-climber,obtains a Vo-local optimal solution,

$$\mathcal{Z}: V \mathcal{R}_{\mathcal{Z}} \to V_{\mathcal{O}} \tag{15}$$

keep in mind that, while this is typically the scenario in behavior, there is no stipulation that the returning solution be near the initial solution. Typically, MA create offspring by recombining chromosomes from two parents and then introducing a little amount of random mutation.

$$Y: V \times V \times \mathcal{R}_{Y} \to V \tag{16}$$

Is the operator for the recombining operator:

$$\mathcal{N}: V \times \mathcal{R}_{\mathcal{N}} \to V \tag{17}$$

the role does mutation have in the reproductive process of genes assuming that Ks is determined by the sum of mutations and a process of, $\mathcal{K}_s = NoY$:

$$\mathcal{K}_s: V \times V \times \mathcal{R}_{\mathcal{N}} \times \mathcal{R}_{Y} \to V \tag{18}$$

defined by:

$$\mathcal{K}_s(y,x,\mathcal{R}_{\mathcal{N}},\mathcal{R}_Y)\underline{\Delta}\mathcal{N}(Y(y,x,\mathcal{R}_Y),\mathcal{R}_{\mathcal{N}}) \quad (19)$$

the hilA-climber Z is added to $\mathcal{K}_s$ for further composition, plus limited to V_O, a mechanism for memetic propagation $\mathcal{K}_n\underline{\Delta}$Z o$NoY$ results,

$$\mathcal{R}_n: V_O \times V_O \times \mathcal{R}_Z \times \mathcal{R}_{\mathcal{N}} \times \mathcal{R}_Y \to V_O \quad (20)$$

defined by:

$$\mathcal{K}_n(y,x,\mathcal{R}_Z,\mathcal{R}_{\mathcal{N}},\mathcal{R}_Y)\underline{\Delta}\mathcal{Z}(\mathcal{N}(y,x,\mathcal{R}_Y),\mathcal{R}_{\mathcal{N}}),\mathcal{R}_Z) \quad (21)$$

3. RESULTS AND DISCUSSION

The suggested method OSAMA methodological approach in comparison to existing models, like "parallel particle swarm optimization with sequence pair (P-PSO-OPFP) [26], nature-inspired hybrid optimization algorithm (BIOA-OPFP) [27], fixed-outline floor planning (LOA-OPFP) [28], and hybrid bio-inspired whale optimization and adaptive bird swarm optimization optimal partitioning and floorplanning (Hyb-BI-WO-ABSO-OPFP) [29]" techniques, all calculated with the benchmark. The S1196 and S1238 and the S3350 and the S8378 benchmark circuits (BC) are used in this study. The design parameters are then used to generate evaluation metrics such as "area, wire length (WL), delay, speed, and power value". Table 1 displays the simulation parameters. Minimizing routing WL and minimizing dead area on the floor layout is a couple of ways in which costs can be reduced when designing VLSI circuits.

Table 1. Simulation parameters [16]

Parameter	Value
Maximum value	+1
BC	4
Minimum value	-1
Number of iteration	500
Random number	(0,1)

3.1. Power usage

VLSI power use is the quantity of energy used by electronic components on a chip, such as ICs. Reducing power usage is simplified by designs that use less energy, smart power management strategies, and the use of more energy-efficient components. For economic and ecological reasons, it is crucial to limit electricity consumption. The comparison of power usage values for the BC is shown in Figure 3 and Table 2. For the S1196 BC, the proposed OSAMA technique achieved reductions in power usage of 25.38%, 29.65%, 30.24%, and 35.24%, power usage was reduced by 36.24%, 30.27%, 37.59%, and 36.52% for the S1238 BC, power usagein the S3350 BC is reduced by 25.34%, 32.56%, 41.28%, and 25.64%, the S8378 BCreduces power usage by 29.65%, 31.07%, 32.56%, and 26.35% as compared to the existing techniques, like (P-PSO-OPFP, BIOA-OPFP, LOA-OPFP, and Hyb-BI-WO-ABSO-OPFP). Comparing the present method to the previous one, this particular methods uses fewer watts of power.

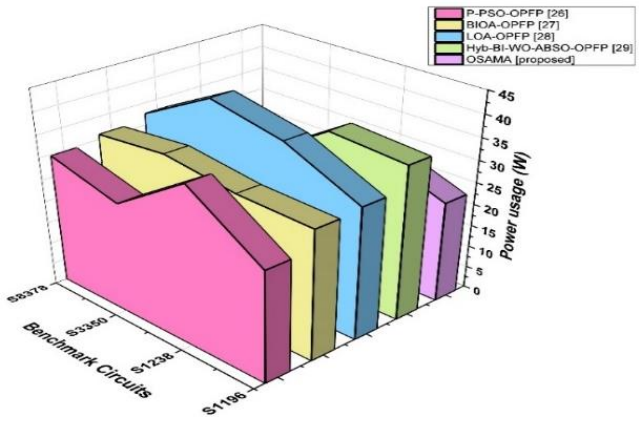

Figure 3. Power usage value comparison

Table 2. Comparison of power usage

Methods	S1196 (%)	S1238 (%)	S3350 (%)	S8378 (%)
P-PSO-OPFP	25.38	36.24	25.34	29.65
BIOA-OPFP	29.65	30.27	32.56	31.07
LOA-OPFP	30.24	37.59	41.28	32.56
Hyb-BI-WO-ABSO-OPFP	35.24	36.52	25.64	26.35
OSAMA [proposed]	23.16	29.45	22.33	23.19

3.2. Speed

The speed performance of a VLSI circuit or electronic system measures how rapidly information can be processed and delivered. It is a fundamental part of VLSI design, specifically for high-speed processing devices like microprocessors, DSPs, and RF communications systems. The comparison of speed values for the BC is displayed in Figure 4 and Table 3. The suggested OSAMA approach has improved speed for the S1196 BC by 332.08%, 41.27%, 28.57%, and 44.20%, S1238 BC, the speed increases are 26.54%, 30.24%, 22.15%, and 33.05%, S3350 BC speed improvements of 33.28%, 36.58%, 25.89%, and 41.78%, The S8378 BC achieved gains of 32.15%, 32.54%, 39.54%, and 41.76% in speed, where compared to existing techniques as (P-PSO-OPFP, BIOA-OPFP, LOA-OPFP, and Hyb-BI-WO-ABSO-OPFP). When compared to the current method, the proposed method's speed is much faster.

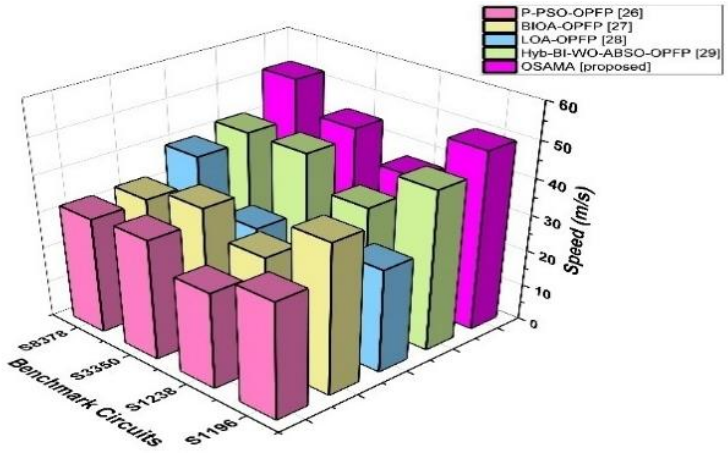

Figure 4. The benchmark speed comparison

Table 3. Comparison of speed

Methods	S1196 (%)	S1238 (%)	S3350 (%)	S8378 (%)
P-PSO-OPFP	32.08	26.54	33.28	32.15
BIOA-OPFP	41.27	30.24	36.58	32.54
LOA-OPFP	28.57	22.15	25.89	39.54
Hyb-BI-WO-ABSO-OPFP	44.20	33.05	41.78	41.76
OSAMA [proposed]	49.51	36.47	38.17	52.34

3.3. Wire length

At partition, boundaries are located in the components that are connected to other components that are located in different partitions. Strongly connected parts of a partition are additionally placed adjacent to one another to reduce the WL. The comparison of the WL values for the BC is shown in Figure 5 and Table 4. For the S1196 BC, the suggested OSAMA technique achieved WL reductions of 30.27%, 29.68%, 44.28%, and 32.57%, reduced WLs of 26.35%, 32.05%, 36.05%, and 21.05% for the S1238 BC, S3350 BC, the WL was reduced by 22.38%, 34.58%, 23.58%, and 20.18%, reduced WL for the S8378 BC of 33.65%, 29.56%, 35.67%, and 27.48% compared to the current approaches, for example, (P-PSO-OPFP, BIOA-OPFP, LOA-OPFP, and Hyb-BI-WO-ABSO-OPFP).

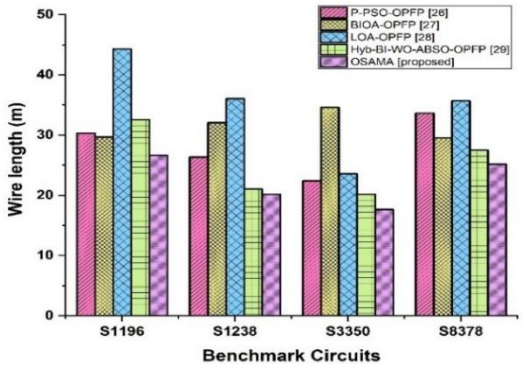

Figure 5. Comparison of the benchmark WL values

Table 4. Comparison of WL

Methods	S1196 (%)	S1238 (%)	S3350 (%)	S8378 (%)
P-PSO-OPFP	30.27	26.35	22.38	33.65
BIOA-OPFP	29.68	32.05	34.58	29.56
LOA-OPFP	44.28	36.05	23.58	35.67
Hyb-BI-WO-ABSO-OPFP	32.57	21.05	20.18	27.48
OSAMA [proposed]	26.65	20.18	17.64	25.16

4. CONCLUSION

VLSI chips contain numerous transistors and other electrical components. The challenge of dividing VLSI circuits is critical and difficult to the design and execution of IC. Partitioning is the technique of dividing a big circuit into more manageable sections to better optimize its performance metrics and design goals. In this research, we show that the OSAMA may be used to generate optimal results for layout and partitioning for VLSI. The S1196, S1238, S3350, and S8378 BCs were used in the benchmark evaluation. The proposed OSAMA algorithm can 23.16%, 29.45%, 22.33%, and 23.19% power usage is low, 49.51%, 36.47%, 38.17%, and 52.34% greater speed, 26.65%, 20.18%, 17.64%, and 25.16% shorter wiring distance for the BC compared to traditional techniques. This research aimed to improve upon previous efforts at outcome prediction for floor plans by utilizing an area, WL measure. In a further development, we are changing to dynamic community detection or overlapping community detection.

REFERENCES

[1] M. Bansal, R. Arora, and R. Bharti, "VLSI Layout: Concept to Realization," in *2022 International Conference on Applied Artificial Intelligence and Computing (ICAAIC)*, May 2022, pp. 1590–1596, doi: 10.1109/ICAAIC53929.2022.9792835.

[2] F. H. Khan, M. A. Pasha, and S. Masud, "Advancements in Microprocessor Architecture for Ubiquitous AI—An Overview on History, Evolution, and Upcoming Challenges in AI Implementation," *Micromachines*, vol. 12, no. 6, p. 665, Jun. 2021, doi: 10.3390/mi12060665.

[3] S. M. Abbas, T. Tonnellier, F. Ercan, and W. J. Gross, "High-Throughput VLSI Architecture for GRAND," in *2020 IEEE Workshop on Signal Processing Systems (SiPS)*, Oct. 2020, pp. 1–6, doi: 10.1109/sips50750.2020.9195254.

[4] S. Shreyanth, D. S. Harshitha, and S. Niveditha, "Implementation of Machine Learning in VLSI Integrated Circuit Design," *SN Computer Science*, vol. 4, no. 2, Jan. 2023, doi: 10.1007/s42979-022-01580-5.

[5] R. Manikandan, R. Parameshwaran, J. Prassanna, and K. R. Sekar, "A Study on Specific Computational Algorithms for VLSI Cell Partitioning Problems," *International Journal on Emerging Technologies*, vol. 10, no. 2, pp. 67–70, 2019.

[6] R. P, T. S Chandra, and M. K. Singh, "Effective K-way Partitioning of VLSI Circuits with Hetero-Homo Status Based Models using Evolutionary Computation," *International Journal of Engineering Trends and Technology*, vol. 71, no. 5, pp. 240–250, May 2023, doi: 10.14445/22315381/ijett-v71i5p225.

[7] J. K. Kurian, "Study on recent approaches of power optimization techniques in VLSI design," *IET Conference Proceedings*, vol. 2023, no. 2, pp. 54–58, May 2023, doi: 10.1049/icp.2023.1417.

[8] R. Karthick, A. Senthilselvi, P. Meenalochini, and S. S. Pandi, "An Optimal Partitioning and Floor Planning for VLSI Circuit Design Based on a Hybrid Bio-Inspired Whale Optimization and Adaptive Bird Swarm Optimization (WO-ABSO) Algorithm," *Journal of Circuits, Systems and Computers*, vol. 32, no. 08, May 2023, doi: 10.1142/s0218126623502730.

[9] I. Oliveira, M. Danigno, P. F. Butzen, and R. Reis, "Benchmarking Open Access VLSI Partitioning Tools," in *2021 IEEE 12th Latin America Symposium on Circuits and System (LASCAS)*, Feb. 2021, pp. 1–4, doi: 10.1109/lascas51355.2021.9459131.

[10] P. Rajeswari, T. S. Chandra, and A. K. Kumar, "Synthesis of VLSI Structural Cell Partitioning Using Genetic Algorithm," in *ICT Systems and Sustainability*, Springer Singapore, 2020, pp. 279–287, doi: 10.1007/978-981-15-8289-9_26.

[11] N. Thakur, S. Saxena, and D. Kumar, "Circuit Simulation Techniques of VLSI Circuits," *International Conference on Electronic Design Innovation and Technology*, 2015.

[12] G. Luo, X. Chen, and S. Nong, "Net Clusting Based Low Complexity Coarsening Algorithm In k-way Hypergraph Partitioning," *Journal of Physics: Conference Series*, vol. 2245, no. 1, p. 12019, Apr. 2022, doi: 10.1088/1742-6596/2245/1/012019.

[13] L. Lin, T. Wu, and Z. Zhang, "A diameter-based model of the rectilinear partitioning problem in VLSI physical design," in *2020*

Chinese Automation Congress (CAC), Nov. 2020, pp. 2610–2615, doi: 10.1109/cac51589.2020.9327644.
[14] J. Rodriguez, F. Pellegrini, G. François, and L. Zaourar, "Circuit partitioning with path delay-based minimization," *31st European Conference on Operational Research*, 2021.
[15] M. A. Sebak, M. A. Fawzy, M. A. Ibrahim, and R. G. Elsaid, "A Greedy-Simulated Annealing approach for placement of VLSI circuits," *ResearchGate*, 2020.
[16] P. Rajeswari, T. Chandra S., and S. Sasi, "Efficient k-way partitioning of very-large-scale integration circuits with evolutionary computation algorithms," *Bulletin of Electrical Engineering and Informatics*, vol. 13, no. 6, pp. 4002–4007, Dec. 2024, doi: 10.11591/eei.v13i6.5781.
[17] B. Srinivasan, R. Venkatesan, B. Aljafari, K. Kotecha, V. Indragandhi, and S. Vairavasundaram, "A Novel Multicriteria Optimization Technique for VLSI Floorplanning Based on Hybridized Firefly and Ant Colony Systems," *IEEE Access*, vol. 11, pp. 14677–14692, 2023, doi: 10.1109/access.2023.3244346.
[18] K. Lakshmanna, F. Shaik, V. K. Gunjan, N. Singh, G. Kumar, and R. M. Shafi, "Perimeter Degree Technique for the Reduction of Routing Congestion during Placement in Physical Design of VLSI Circuits," *Complexity*, vol. 2022, no. 1, Jan. 2022, doi: 10.1155/2022/8658770.
[19] A. Agnesina, S. Pentapati, and S. K. Lim, "A general framework for VLSI tool parameter optimization with deep reinforcement learning," in *34th Conference on Neural Information Processing Systems (NeurIPS 2020)*, 2020, pp. 1–6.
[20] S. Lamichhane, S. Peng, W. Jin, and S. X.-D. Tan, "Fast Electrostatic Analysis For VLSI Aging based on Generative Learning," in *2021 ACM/IEEE 3rd Workshop on Machine Learning for CAD (MLCAD)*, Aug. 2021, pp. 1–6, doi: 10.1109/mlcad52597.2021.9531320.
[21] S. N. Hussain and K. H. Kishore, "Heuristic Approach to Evaluate the Performance of Optimization Algorithms in VLSI Floor Planning for ASIC Design," in *Modern Approaches in Machine Learning and Cognitive Science: A Walkthrough*, Springer International Publishing, 2020, pp. 213–225, doi: 10.1007/978-3-030-38445-6_16.
[22] P. Rajeswari and S. T. Chandra, "A survey on an optimal solution for VLSI circuit partitioning in physical design using DPSO & DFFA algorithms," in *2017 International Conference on Intelligent Sustainable Systems (ICISS)*, Dec. 2017, pp. 868–872, doi: 10.1109/iss1.2017.8389301.
[23] Y.-H. Lee, C.-H. Kuei, Y.-Z. Kao, and S.-S. F. Jiang, "Algorithm and VLSI Architecture Designs of a Lossless Embedded Compression Encoder for HD Video Coding Systems," *Journal of Circuits, Systems and Computers*, vol. 30, no. 04, p. 2130004, Aug. 2020, doi: 10.1142/s021812662130004x.
[24] S. Nath, J. K. Sing, and S. K. Sarkar, "Wire length optimization of VLSI circuits using IWO algorithm and its hybrid," *Circuit World*, vol. 50, no. 2/3, pp. 205–216, Jul. 2021, doi: 10.1108/cw-08-2020-0215.
[25] B. N. B. Ray, S. S. Sahoo, and S. K. Mohanty, "G-NSVF: A Greedy Algorithm for Non-Slicing VLSI Floorplanning," in *Digital Democracy – IT for Change*, Springer Singapore, 2021, pp. 48–58, doi: 10.1007/978-981-16-2723-1_6.
[26] R. P. Guru and V. Vaithianathan, "An efficient VLSI circuit partitioning algorithm based on satin bowerbird optimization (SBO)," *Journal of Computational Electronics*, vol. 19, no. 3, pp. 1232–1248, Apr. 2020, doi: 10.1007/s10825-020-01491-9.
[27] S. J. Basha and B. A. Raheem, "VLSI Floorplanning Using Nature-Inspired Hybrid Optimization Algorithm," in *ICDSMLA 2019: Proceedings of the 1st International Conference on Data Science, Machine Learning and Applications*, 2020, pp. 1959–1967, doi: 10.1007/978-981-15-1420-3_200.
[28] R. P and T. S. Chandra, "Partitioning of VLSI Circuits on the basis of Standard Genetic Algorithm and Comparative Analysis of Partitioning Algorithms," *International Journal of Electrical and Electronics Engineering*, vol. 9, no. 12, pp. 126–133, Dec. 2022, doi: 10.14445/23488379/ijeee-v9i12p111.
[29] B. S. Yıldız, N. Pholdee, S. Bureerat, M. U. Erdaş, A. R. Yıldız, and S. M. Sait, "Comparision of the political optimization algorithm, the Archimedes optimization algorithm and the Levy flight algorithm for design optimization in industry," *Materials Testing*, vol. 63, no. 4, pp. 356–359, Apr. 2021, doi: 10.1515/mt-2020-0053.

BIOGRAPHIES OF AUTHORS

Dr. P. Rajeswari received the Bachelor of Engineering degree from Government College of Technology, Coimbatore and Master's Degree in VLSI Design from Anna University, Chennai. She completed her Ph.D. degree with Department of Electronics and Communication Engineering, Dayananda Sagar University, Bangalore-78. Currently working as an Asst. Professor in Department of Electronics and Telecommunication Engineering, Dayananda Sagar College of Engineering, and Bangalore-78. Her research interests include embedded system design, VLSI signal processing, circuit optimization, and CAD algorithms for VLSI architectures. She has published research papers in refereed international journals and research papers in the proceedings of various international conferences. She received funds from KSCST. She can be contacted at email: prajeswarisugans@mail.com.

Dr. Smitha Sasi working as an Associate Professor in the Department of Electronics and Telecommunication Engineering, Dayananda Sagar College of Engineering has about 17 years of teaching experience. She received her B.Tech degree in Information Technology from Calicut University and M.Tech degree in Digital Communication and Networking with rank from Visvesvaraya Technological University, Belagavi, and Karnataka. She is received Doctoral degree in Cryptography from Visvesvaraya Technological University. She has published research papers in refereed international journals and research papers in the proceedings of various international conferences. She received funds from various funding agencies in the cryptography and security domain. She is authored various book chapters in the area of cryptography. She can be contacted at email: aaazekry@hotmail.com.