Journal of King Saud University – Computer and Information Sciences 32 (2020) 1095–1107

Contents lists available at ScienceDirect

Journal of King Saud University – Computer and Information Sciences

journal homepage: www.sciencedirect.com

Review on VLSI design using optimization and self-adaptive particle swarm optimization

S.B. Vinay Kumar [a,*], P.V. Rao [b], H.A. Sharath [c], B.M. Sachin [d], U.S. Ravi [e], B.V. Monica [a]

[a] School of Engineering and Technology, Jain University, Bangalore, Karnataka, Ramanagara District, 562 112, India
[b] Vignana Bharathi Institute of Technology, Ghatkesar, Hyderabad, Telangana St., India
[c] Maharaja Institute of Technology (MIT), Belawadi, Srirangapatna Tq, Mandya 571438, India
[d] Bangalore Institute of Technology, Bengaluru, Karnataka 560004, India
[e] J V Global Services LLP, Bangalore, India

ARTICLE INFO

Article history:
Received 5 September 2017
Revised 9 January 2018
Accepted 9 January 2018
Available online 31 January 2018

Keywords:
VLSI design
Optimization
Bio-inspired algorithm
Self-adaptive PSO
Floor planning

ABSTRACT

Today, VLSI technology has taken a fundamental role in developing most of the high-tech electronic circuits. Even though VLSI design is renowned for its smaller size, lower cost, lower power, high reliability and high functionality, the design process takes long time and produces high risk. So, to obtain the knowledge regarding the different contributions on VLSI design, about 52 papers on the design of VLSI using optimization are reviewed here. Accordingly, VLSI design optimization is analyzed through different bio-inspired algorithms and the performance measures of different VLSI experimentations are compared. Further, various improvements on Self-Adaptive Particle Swarm Optimization (SA-PSO) and VLSI design optimization, without the adoption of bio-inspired algorithms, are examined. Additionally, the floor planning problem of VLSI is considered and reviewed. Eventually, this paper provides the diverse research gaps and challenges in VLSI design, which may be helpful for the authors and the philosophers to contribute for further research.

Contents

* Corresponding author.
E-mail address: vinaykumar.sb27@gmail.com (S.B.V. Kumar).
Peer review under responsibility of King Saud University.

Production and hosting by Elsevier

https://doi.org/10.1016/j.jksuci.2018.01.001
1319-1578/

1. Introduction

VLSI is the *"process of forming an integrated circuit (IC) by incorporating thousands of transistors into a single chip*." The technologies dealing with communication and complex semiconductor were introduced in the year 1970, when VLSI was also established. In fact, there was little restriction on the functions of Integrated Circuit (IC), before the introduction of VLSI technology (Areibi and Yang, 2004; Lee and Lim, 2006; Mei et al., 2006; Huang et al., 2006). Usually, the required components for the design of electronic circuits are Random Access Memory (RAM), Read Only Memory (ROM) and Central Processing Unit (CPU), etc. These components can be designed in a single chip through the VLSI technology (Chakraborty et al., 2006; Sun and Zhang, 2005; Arora, 2005; Tsai and Yang, 2004; Chang and Lim, 2004). Over the last few decades, the electronics industry has attained a remarkable progression, due to the continuous development of VLSI and its applications on various system designs. For instance, the applications such as, video or image processing, telecommunication and consumer electronics are growing up at a quick pace (Balaji et al., 2015a,b; Sudha and Sridharan, 2004; Sheu et al., 2004; Hsia, 2003; Kim and Kim, 2003; Aubepart et al., 2003). However, the general problems of VLSI design are the lack of appropriate area, speed, power, application area specialization and knowledge. Moreover, the sudden change of technology, the requirement of a multi-disciplinary team, the need for huge design space and the persistence of short design cycle make VLSI design to be complex.

In fact, VLSI design for attaining the optimal solution is reasonably priced, yet time-consuming. In addition, the process of design often requires the appropriate selection of independent design parameters (Gang, 2002; Hodge and Newcomb, 2002; Hsiao et al., 2000). Therefore, the optimization process uses those parameters as variables to search the solution optimally. However, the process tends to be ineffectual, due to the massively long search process. To overcome the existing challenges in the VLSI design, various optimization techniques have been developed, particularly in optimizing the conventional design parameters (which includes total wire length and area). The later advancements in VLSI have focussed on furthermore design objectives and constraints. As a result, it is difficult to provide an optimal VLSI design with effective methods (Chen and Wei, 1999; Allen and Terman, 2000; He et al., 2000; Simon et al., 2000).

Recently, a renowned meta-heuristic algorithm, called PSO, has been developed as the effective algorithm for the optimization of non-linear functions (Chen et al., 2016; Chang and Lim., 2004, Vicente et al., 2004). Generally, PSO shares more functionality with evolutionary computation approaches like, the Genetic Algorithm (GA). The system is initialized with a population of random solutions as well as searches for optima by updating the generations. Nevertheless, not like GA, PSO has no evolution operators like, crossover as well as mutation. In PSO, the potential solutions (termed particles) fly through the problem space in subsequence to the current optimum particles. On comparing with the famous conventional optimization techniques, PSO is not dependent on any simplifying derivations, involving few parameters with easy implementation. Self-adaptive PSO is considered as the other simple technique for attaining an optimal VLSI design (Kessler and Ganesan, 1985). In Ardakani et al. (2016), Mansor et al. (2016), the VLSI circuit involving Neural Network (NN) has been exploited to detect any possible error, which occurred earlier in the manual circuit design. Here, NN has taken lower-level inputs and constructed higher-level abstractions through the composition of layers.

The contribution of this paper falls into three. First, the role of bio-inspired algorithms on VLSI design is reviewed. Second, a clear study regarding the improvements of SA-PSO is provided. Third, the floor planning problem of VLSI design is reviewed from different research contributions. The organization of this paper is as follows: Section 2 provides the review on VLSI design, with optimization based on the bio-inspired algorithms; Section 3 gives a review on SA-PSO; Section 4 presents a review on the general optimization of VLSI design; Section 5 presents a clear description of the floor-planning problem in VLSI and Section 6 suggests the research gaps and the future directions, followed by conclusion in Section 7.

1.1. Motivations and challenges

Since VLSI floorplanning is NP-hard, different heuristic approaches have been proposed. These approaches are classified into two, namely, the constructive approaches and the iterative approaches. Generally, the constructive approach exploits the heuristic scheme to construct a floorplan. The Bottom-Left (BL) and BL fill methods (Bernard, 1983) are the most common constructive approaches. An effective heuristic algorithm, in which two significant concepts such as, the corner-occupying action and the caving degree are introduced to guide the packing process, has also been presented. The iterative method uses the meta-heuristic approach such as, Simulated Annealing, Genetic Algorithm (GA) and Particle Swarm Optimization (PSO) algorithm, in order to create a final optimal solution. In contrast to the other population-based evolutionary algorithms, PSO is motivated by the simulation of social behaviour, rather than depending on the concept of the survival of the fitness. The major advantages of PSO are the simplicity in implementation and its ability to converge quickly as possible. The floorplanning problem has been solved by PSO and it is discussed in the subsequent section. The PSO approach has been exploited to generate an initial stage with overlap-free placement and to find out the potential optimal placement solution. Nevertheless, only area optimization has been considered and no implementation detail of the approach has been mentioned. Further, it has been unable to resolve the issues, which optimize the area and the wire length concurrently. Therefore, it has not been an effectual PSO approach for solving the general floorplanning problems. In order to overwhelm the above problems, various SA-PSO are analysed, with consideration on both the area and the wire length.

2. VLSI design with optimization based on bio-inspired algorithms

2.1. Review on bio-inspired algorithms and their objectives

The taxonomy of optimization algorithms in VLSI design is shown in Fig. 1. The optimization has been performed on the basis of Bio inspired optimization, deterministic optimization and other optimizations. The bio-inspired optimization algorithms are classified into two types, namely, evolutionary algorithms and swarm intelligence.

The contribution, regarding the application of bio-inspired algorithms in VLSI design, is shown in Table 1. In 2008, Giuseppe et al. (2008) have developed the Genetic Algorithm (GA) to balance the robustness, accuracy and computational effort of the design of analog circuits. Further, in 2009, Bo et al. (2009) have improvised the

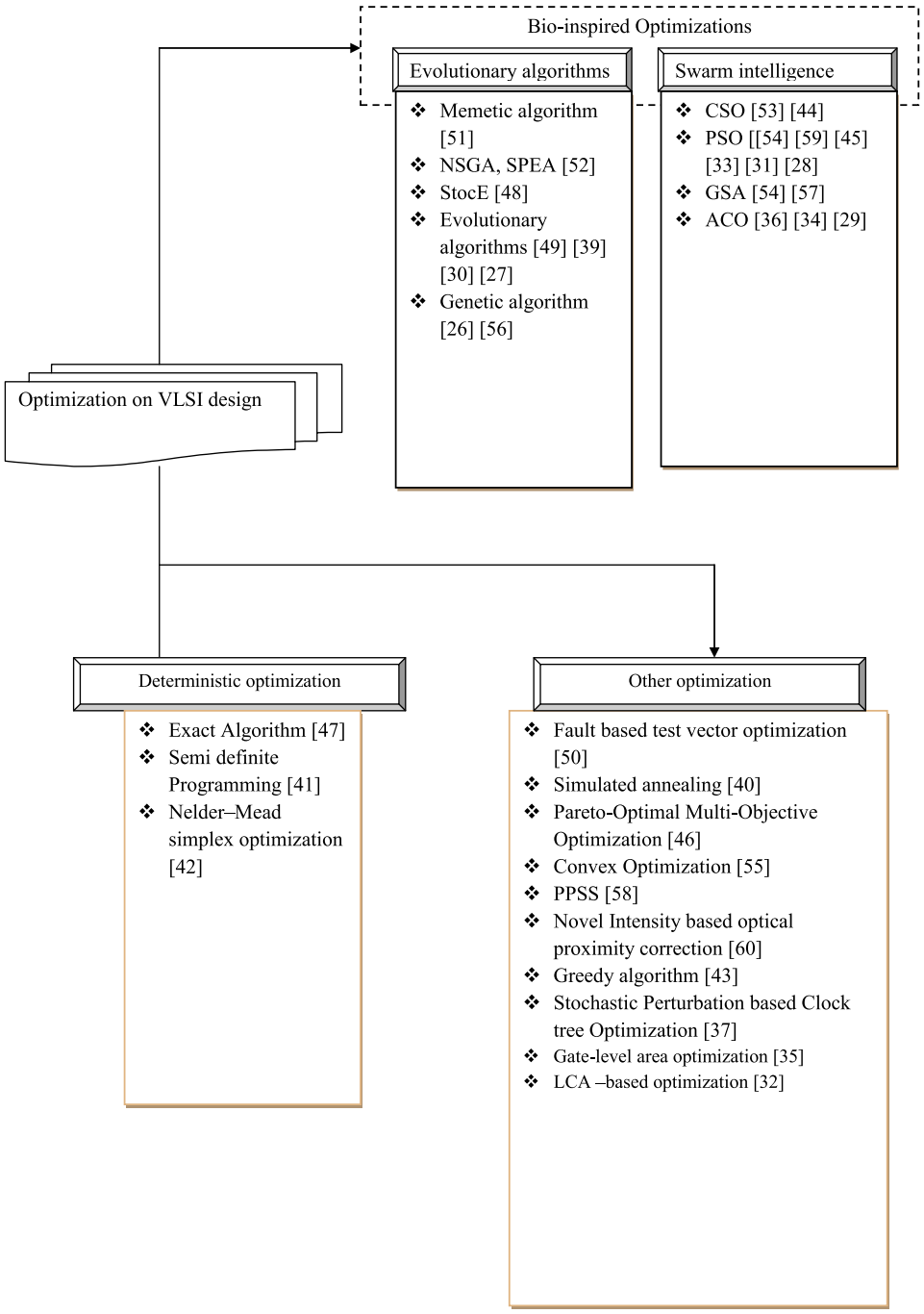

Fig. 1. Taxonomy of optimization algorithms in VLSI design.

robustness and the quality of analog circuits using Co-evolutionary Differential Evolution (CODE), whereas Thakker et al. (2009) have solved the problem associated with the extraction of MOSFET parameters. Later, in 2010, Revna et al. (2010) have used the PSO algorithm for reducing the time and the error that occurred during the design of inverters. Manuel et al. (2010) have improved the efficiency of the design of analog IC using evolutionary optimization. Moreover, Mohammed et al. (2010) have adopted Discrete Cooperative PSO to minimize the total wire length of the Field Programmable Gate Arrays (FPGAs).

In 2011, Ali (2011) has developed the League Championship Algorithm (LCA)-based constrained global optimization to enhance the probability of an individual in generating a better solution. In contrast, Revna et al. (2011) have designed an inverter with diminished time consumption and error using PSO algorithm. Furthermore, Hyun and Sungho (2011) have introduced the Ant Colony Optimization (ACO) algorithm to reduce the energy consumption during task scheduling and voltage selection, minimizing total energy in a multiprocessor system.

Moreover, in 2012, Levent et al. (2012) have optimized the area of digit-serial constant multiplications, under the architecture of shift adder and subtractor using Gate-level area optimization. In 2013, Chyi-Shiang et al. (2013) have contributed the ACO algorithm for VLSI chip design, with the aim to optimize the area and wire length. Sina et al. (2013) have suggested the Stochastic Perturbation-based Clock Tree Optimization for reducing the design complexity of the chips. Further, in 2014, Ping et al. (2014) have improved the the efficiency and achieved high

Table 1
Contributions on bio-inspired algorithms for VLSI design optimization.

Algorithm	Authors	Optimization
Fault-based test vector optimization	Khera et al. (2017)	Reduction of test vector count
Adaptive hybrid memetic algorithm	Chen et al. (2017)	Optimization of wire length, the maximum temperature, and the average temperature
Simulated annealing	Martins et al. (2015)	Optimization of the placement of each proximity group, while automating the placement task (Martins et al., 2015)
NSGA, SPEA	Farnsworth et al. (2017)	Optimization of cost (Farnsworth et al., 2017)
CSO and PSO	Dash et al. (2017)	Reduction of error between the ideal frequency response and the prectican frequency as well as to enhance the design strategy
GSA	Dehbashian and Maymandi-Nejad (2017)	Creating a balance between the exploration and the exploitation capabilities
Pareto-Optimal Multi-Objective Optimization	Kourany et al. (2016)	To quantify the parameters that have higher effect on the output uncertainty
Convex Optimization	Khodabandeloo et al. (2017)	Reduction of peak temperature of the chip in CMOS technology
Hybridized Genetic algorithm	Zhen et al. (2017)	Reduction of interferences among the assembly system components of robot
Advanced GSA and PSO	Dehbashian and Maymandi-Nejad (2017)	Creating a balance between the exploration and the exploitation capabilities
Probabilistic Pairwise Swap Search (PPSS) algorithm	Aiman (2017)	Reduction of computational complexity
Modified PSO	Kaboli et al. (2017)	Reduction of power
Novel Intensity-based optical proximity correction	Awad et al. (2017)	Reduction of error and computational time
Exact Algorithm	Funke et al. (2016)	Minimization of wire length
StocE algorithm	Sait and Siddiqi (2016)	Solving the 2D global routing problems
Evolutionary algorithm	Povoa et al. (2016)	Reduction of time consumption
Semidefinite Programming-based Global Optimization	Hungerländer and Anjos (2015)	Optimization of Multi-Row Facility Layout Problem
Nelder–Mead simplex optimization	Malak et al. (2015)	Reduction of computational time
Greedy Algorithm-based optimization	Taassori et al. (2015)	Reduction of energy consumption, improving the area of Network on Chip (NoC) and optimizing the count of virtual channel
PSO	Revna (2010), Revna (2011), Karmakar and Chattopadhyay (2015)	Reduction of power consumption (Karmakar and Chattopadhyay, 2015), Reduction of time and error (Revna, 2010, 2011)
CSO	El-Maleh et al., (2015)	Minimization of area
Tabu-ant colonies hybrid modeling	Yang et al. (2014)	Improvement in efficiency and high convergence
Constrained hybrid evolutionary optimization	Subhojit and Susovon (2014)	Reduction of execution time and probability
ACO	FMichael et al. (2017)	Optimization of area and wire length
Stochastic Perturbation-based Clock Tree Optimization	Sina (2013)	Reduction in design complexity
Gate-level area optimization	Levent et al. (2012)	Optimizes the area of digit-serial constant multiplications
LCA-based constrained global optimization	Ali (2011)	Optimizes the probability
ACO	Hyun and Sungho (2011)	Reduction in energy consumption
Evolutionary Algorithm	Manuel et al. (2010)	Improvement in efficiency
Discrete Cooperative PSO	Mohammed et al. (2010)	Minimization of wire length
CODE	Bo et al. (2009)	Improvement in quality and robustness
Hierarchical PSO	Thakker et al. (2009)	Solves the problem of MOSFET parameter extraction
GA	Giuseppe et al. (2008)	Optimization of accuracy, robustness and computational effort

convergence of VLSI physical design using the Tabu-ant colonies hybrid modeling. On the other hand, Subhojit and Susovon (2014) have dealt with the Constrained hybrid evolutionary optimization to speed up the execution time and to attain high probability in designing fixed order compensator for the DC-DC convertors.

In 2015, Ricardo et al. (2015) have used multi-objective optimization to optimize the placement of each proximity group, while automating the placement task in the layout design of analog integrated circuits. Philipp and Miguel (2015) have adopted Semi definite Programming-based Global Optimization to address the Multi-Row Facility Layout Problem in VLSI design. Even more, Akram et al. (2015) have developed the space exploration of analog firm intellectual properties with reduced computational time using Nelder–Mead simplex optimization. Mehdi et al. (2015) have contributed the Greedy Algorithm-based optimization to minimize the energy consumption and to improve the area of Network on Chip (NoC), in addition to optimizing the count of virtual channel of every link. In addition, Aiman et al. (2015) have utilized Cuckoo Search Optimization (CSO) to reduce the area during the synthesis of sequential circuits and Rajit et al. (2015) have diminished the power values and power consumption of the System-on-Chip (SoC) test scheduling using PSO algorithm.

Subsequently, in 2016, Taher et al. (2016) have quantified the parameters having higher effect on the output uncertainty in the design of analog circuits using Pareto-Optimal Multi-Objective Optimization. Funke et al. (Funke et al., January 2016) have used Exact Algorithm to reduce the wire length and to avoid the given sets of blocked regions in VLSI design. Moreover, Sadiq et al. (2016) have affluently solved the 2D global routing problems in VLSI design using Stochastic Evolution (StocE) algorithm and Povoa et al. (2016) have dealt with the multi-objective evolutionary technique to diminish the time consumption, while designing the Radio-Frequency (RF) circuit.

To the next, in 2017, Vinod et al. (2017) have lowered the test vector count during the VLSI testing of standard ISCAS circuits using the fault-based test vector optimization. Further, Jianli et al. (2017) have used the adaptive hybrid memetic algorithm to optimize the wire length, the maximum temperature and the average temperature of a chip. Afterwards, Michael et al. (2017) have minimized the computational expense in the design of Micro Electro Mechanical System (MEMS) and Judhisthir et al. (2017) have utilized an improved CSO to reduce the error between the ideal frequency response and the practical frequency, so as to enhance the design strategy of linear phase multiband filters. Moreover, Maryam and Mohammad (2017a) have developed the Advanced

Table 2
Review on the performance measures and their optimized values.

Performance measures	Authors [Citation]	Values [Citation]
Percentage of comparative reduction	Vinod et al. (2017)	15.04% (Revna et al., 2011)
Efficiency	Vinod et al. (2017), Taher et al. (2016), Ping et al. (2014)	100% (Taher et al., 2016)
Area	Jianli et al. (2017), Ricardo et al. (2015) Maryam and Mohammad (2017a,b), Zhen et al. (2017), Aiman (2017), Aiman et al. (2015), Chyi-Shiang et al. (2013), Levent et al. (2012)	48.48 mm^2 (Jianli et al., 2017)
Wire length	Jianli et al. (2017), Funke et al. (2016), Sadiq and Umair (2016), Sina et al. (2013)	1300000um (Sina et al., 2013)
Average temperature	Jianli et al. (2017)	93.85 °C (Jianli et al., 2017)
Peak temperature	Jianli et al. (2017), Behnam et al. (2017)	344.67 K (Behnam et al., 2017)
Run Time	Jianli et al. (2017), Ricardo et al. (2015), Judhisthir et al. (2017), Maryam and Mohammad (2017a), Behnam et al. (2017), Zhen et al. (2017), Maryam and Mohammad (2017b), Awad et al. (2017), Funke et al. (2016), Sadiq and Umair (2016), Philipp and Miguel (2015), Akram et al. (2015), Mehdi et al. (2015), Aiman et al. (2015), Ping et al. (2014), Chyi-Shiang et al. (2013), Sina et al. (2013), Revna et al. (2011, 2010), Bo et al. (2009)	3.77 ns (Revna et al. , 2010)
Cost	Jianli et al. (2017), Aiman (2017)	0.258 (Jianli et al., 2017)
Filter objective	Michael et al. (2017)	2364.157 (Michael et al., 2017)
Central frequency objective	Michael et al. (2017), Milad et al. (2017)	5000 kHz (Michael et al., 2017)
Supply voltage	Michael et al. (2017), Milad et al. (2017)	0.75 V (Milad et al., 2017)
Tank number	Michael et al. (2017)	2 (Michael et al., 2017)
Error	Judhisthir et al. (2017), Taher et al. (2016), Milad et al. (2017), Revna et al. (2011), Mohammed et al. (2010), Thakker et al. (2009), Giuseppe et al. (2008)	0.00026 (Taher et al., 2016)
Minimum iteration	Judhisthir et al. (2017), Ping et al. (2014)	207 (Judhisthir et al., 2017)
Gain	Maryam and Mohammad (2017a), Taher et al. (2016), Maryam and Mohammad (2017b), Milad et al. (2017), Akram et al. (2015), Subhojit and Susovon (2014), Manuel et al. (2010), Bo et al. (2009), Giuseppe et al. (2008)	85.11 dB (Maryam and Mohammad, 2017b)
Phase Margin	Maryam and Mohammad (2017a), Taher et al. (2016), Maryam and Mohammad (2017b), Akram et al. (2015), Subhojit and Susovon (2014), Manuel et al. (2010), Bo et al. (2009)	55.005 deg (Maryam and Mohammad, 2017b)
Slew Rate	Maryam and Mohammad (2017a,b), Bo et al. (2009)	12.34 V/μs (Maryam and Mohammad, 2017a)
Power Consumption	Maryam and Mohammad (2017a,b), Aiman (2017), Povoa et al. (2016), Akram et al. (2015), Mehdi et al. (2015), Hyun and Sungho (2011), Manuel et al. (2010), Bo et al. (2009)	0.332 mW (Maryam and Mohammad, 2017a)
Load Capacitance	Maryam and Mohammad (2017a)	10 pF (Maryam and Mohammad, 2017a)
Bandwidth	Taher et al. (2016), Subhojit and Susovon (2014)	1150 KHz (Taher et al., 2016)
Current consumption	Taher et al. (2016), Manuel et al. (2010), Giuseppe et al. (2008)	1.5 mA (Taher et al., 2016)
Success rate	Taher et al. (2016), Chyi-Shiang et al. (2013), Thakker et al. (2009)	100% (Taher et al., 2016; Chyi-Shiang et al., 2013)
Aspect ratio	Behnam et al. (2017)	1.28 (Behnam et al., 2017)
Input range	Milad et al. (2017)	−18 μA to 18 Ma (Milad et al., 2017)
Transistor count	Milad et al. (2017)	49 (Milad et al., 2017)
Amplitude imbalance	Povoa et al. (2016)	0.29 dB (Povoa et al., 2016)
Unity-gain frequency	Akram et al. (2015)	3.6524 MHz (Akram et al., 2015)
Latency	Mehdi et al. (2015)	20 cycles (Mehdi et al., 2015)
Skew	Sina et al. (2013)	37 ps (Sina et al., 2013)
Skew variance	Sina et al. (2013)	3 ps (Sina et al., 2013)
Percentage of feasibility	Ali (2011)	100% (Ali, 2011)
Capacitive load	Manuel et al. (2010)	1.1 pF (Manuel et al., 2010)

Gravitational Search Algorithm (GSA) to maintain a balance among the exploration and the exploitation abilities, which result in enhancing the efficiency and the accuracy of the analog design process of IC. Similarly, Behnam et al. (2017) have utilized Convex Optimization to lower the peak temperature of the chip in CMOS technology and Zhen et al. (2017) have suggested the Hybridized GA to avoid the interferences between the assembly system components, which were present in the layout of multi-robot cellular manufacturing systems. Furthermore, Maryam and Mohammad (2017b) have established the advanced GSA and PSO to create a balance between the exploration and the exploitation capabilities of IC, whereas Aiman (2017) have reduced the complexity and solved the state assignment problem of the Finite State Machines (FSMs). Likewise, Milad et al. (2017) have lowered the power intake, while designing CMOS using modified PSO. Awad et al. (2017) have used the Novel Intensity-based optical proximity correction for promoting mask optimization in the wafer image, with reduced error and computational time.

2.2. Performance measures

The information regarding the performance measures of various bio-inspired algorithms, used for VLSI design, is shown in Table 2. About 52 papers are reviewed and the optimized value of each measure is tabulated. Regarding each measure, 2.8% of contributions have measured the percentage of comparative reduction, 8.57% of contributions have measured the efficiency and 22.85% of contributions have measured the area, while designing the VLSI component. In addition, the measures such as, wire length, average temperature, and peak temperature have been orderly analyzed by 11.43%, 2.8%, and 5.71% of researchers. Moreover, run time has been considered as the basic measure of all the experiments and it has been contributed by 57.14% of researchers. Furthermore, the measures such as, cost, central frequency, supply voltage, minimum iteration and bandwidth have also been evaluated in about 5.71% of contributions. Subsequently, only 2.8% of contributions have measured the performance, in terms of filter objective, tank

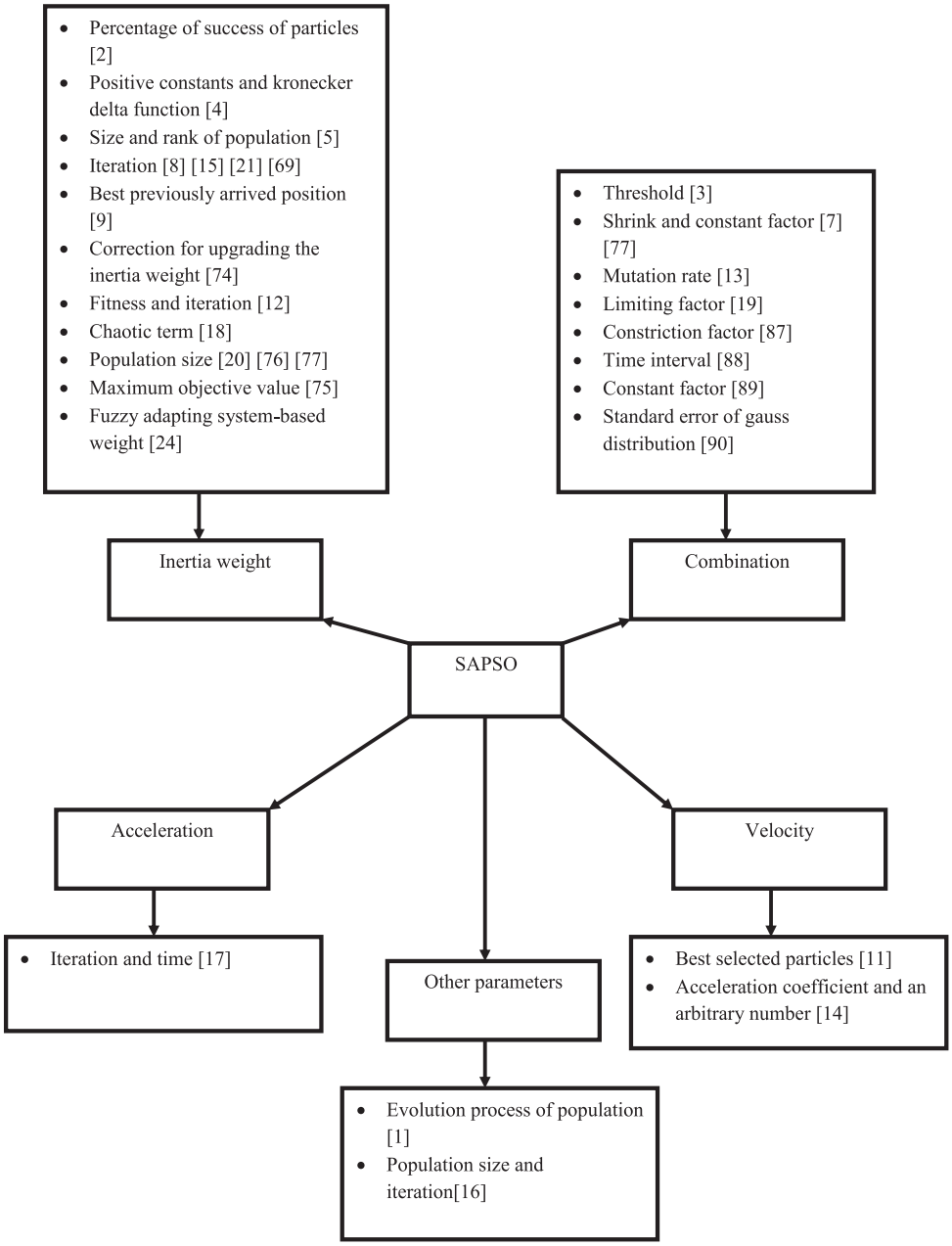

Fig. 2. Self-Adaptive variance of PSO in the literature.

number, aspect ratio, manipulation, input range, transistor count, amplitude imbalance, unity-gain frequency, latency, skew, skew variance, the percentage of feasibility and capacitive load. Meanwhile, 8.57% of researchers have analyzed the measures such as, success rate and current consumption. Even more, 25.71% of researchers have measured the performance, in terms of gain and power consumption. Similarly, phase margin has been analyzed by 20% of researchers, while attempting to design diverse devices associated with VLSI.

3. SA-PSO

3.1. Adaptivess in inertia weight

Self-Adaptive PSO is performed by applying adaptiveness to the inertia weight, as in Elumalai et al. (2016), Karimi-Nasaba et al. (2015), Rastegar et al. (2017), Mahor and Rangnekar (2012), Montalvo et al. (2010), Chander et al. (2011), Yanjun Zhang et al. (2016), Zhai and Jiang (2015), Niknam and Farsani (2010), Taherkhani and Safabakhsh (2016), Li et al. (2013), Wang et al. (2011), Che (2013), Fan and Yan (2014), Bahmani-Firouzi et al. (2013), Zhang and Ding (2011), Jiang et al. (2013), Wang et al. (2012), and Naderi and Narimani (2017). The self-adaptive variance of PSO algorithm in the literature is shown in Fig. 2 and the description of each contribution is illustrated as follows:

In Elumalai et al. (2016), the inertia weight has been adaptively updated, as in Eq. (1). Here, (w^{max}, w^{min}) indicates the range of inertia weight that is selected to be [0, 1] and α defines the percentage of success of the particles.

$$w = (w^{max} - w^{min})\alpha + w^{min} \quad (1)$$

As per Karimi-Nasaba et al. (2015), the fundamental parameters of PSO are the inertia weight w and the associated accelerating coefficients c_1 and c_2, as expressed in Eqs. (2), (3) and (4). These parameters are dependent on the threshold ε and β is considered to be the value that is assigned between 0 and 0.5, $f_{\max}$ represents the maximum iteration and $f_{\min}$ represents the minimum iteration.

$$w = w - \varepsilon \tag{2}$$

$$c_1 = c_1 - \varepsilon \tag{3}$$

$$c_2 = c_2 - \varepsilon \tag{4}$$

$$\varepsilon \leqslant \beta(f_{\max} - f_{\min}) \tag{5}$$

Further, in Rastegar et al. (2017), the self-adaptive inertia weight has been stated, as in Eq. (6). In this equation, y_{pbest} and y_{gbest} represents the best result from the best previous position and the best output among all the particles, t indicates the present iteration, $I_{\max}$ is the count of iterations, a and b indicates the positive constants and $\delta(.)$ denotes the kronecker delta function.

$$w(t) = w^{\min} + (w^{\max} - w^{\min}) \times e\left(-\frac{a}{|y_{pbest} - y_{gbest}| + b} \times \frac{t-1}{I_{\max} - t + \delta(t - I_{\max})}\right) \times [1 - \delta(t - I_{\max})] \tag{6}$$

Based on Mahor and Rangnekar (2012), the self-adaptive inertia weight can be stated by Eq. (7). Here, P_s denotes the size of the population and R indicates the rank of the population.

$$w = \left(3 - \exp\left(-P_s/200\right) + (R/200)^2\right)^{-1} \tag{7}$$

In addition, the reducing factor of inertia weight over the whole generation in the PSO performance can be expressed, as in Montalvo et al. (2010), as:

$$w = 0.5 + \frac{1}{2(\ln(t) + 1)} \tag{8}$$

where t refers to the current iteration.

Further, the authors of Chander et al. (2011) have introduced the inertia weight w from Eq. (9) into the velocity function, which dynamically adapted over time during the searching process.

$$w = \sum_{m=1}^{k-1} P_m \tag{9}$$

where, P_m indicates the best previously arrived position of the mth particle.

As per Zhang et al. (2016), the searching ability of PSO can be enhanced by adaptively altering the inertia weight, as depicted in Eq. (10). In Eqs. (11) and (12), g_i denotes the fitness value of the i^{th} particle, g_{avg} represents the mean fitness of the entire group, $g_{\max}$ and $g_{\min}$ indicates the best and the worst adaptive values of the particle at respective iteration and w_0 refers to the constant value, while t and $t_{\max}$ represents the current and the maximum iterations.

$$w = e^{-h_2} + h_1 \tag{10}$$

$$h_1 = \begin{cases} w_0 + \frac{g_i - g_{avg}}{g_{\max} - g_{\min}}, & g > g_{avg} \\ w_0 + \frac{g_{avg} - g_i}{g_{\max} - g_{\min}}, & g \leqslant g_{avg} \end{cases} \tag{11}$$

$$h_2 = \frac{t}{t_{\max}} \tag{12}$$

According to Zhai and Jiang (2015) and Niknam and Farsani (2010), the linearly decreasing inertia weight with the increase of iterations can be represented by Eq. (13). Here, $w^{\max}$ and $w^{\min}$ denotes the maximal and the minimal weights, m indicates the iteration count and $I_{\max}$ refers to the number of iterations.

$$w = w^{\max} - (w^{\max} - w^{\min})\frac{m}{I_{\max}} \tag{13}$$

The proposed inertia weight adaptation by Taherkhani and Safabakhsh (2016) is given in Eq. (14), where, $I_{\max}$ indicates the maximum number of iterations and C represents the chaotic term.

$$w = (w^{\max} - w^{\min}) \times \frac{I_{\max} - t}{I_{\max}} + w^{\min} \times C \tag{14}$$

Furthermore, the self-adaptive inertia weight of i, based on Li et al. (2013), Wang et al. (2011), is provided in Eq. (15). Here, P_s indicates the population size.

$$w_i = \frac{\log(P_s - i + 1)}{\log(1) + \ldots \log(P_s)} \tag{15}$$

In Niknam and Farsani (2010), Che (2013), the self-adaptive inertia weight is applied to update the velocity of the particles, as per Eq. (16). Here, w^{end} and $w^{\max}$ indicates the final and the maximum iterations.

$$w = w^{\max} - \frac{w^{\max} - w^{end}}{I_{\max}} \times t \tag{16}$$

Moreover, the self-adaptive inertia weight of the x_j^{th} particle has been expressed in Fan and Yan (2014), as given in Eq. (17). M indicates the number of particles and $J_{\max}$ defines the maximum objective value associated with the present iteration, as in Eq. (18).

$$w = \frac{|J(x_j^{th}) - J_{\max}|}{\sum_{j=1}^{M} |J(x_j^{th}) - J_{\max}|} \tag{17}$$

$$J_{\max} = \max(J(x_j^{th})) \tag{18}$$

The fuzzy-adapting inertia weight has been introduced in Bahmani-Firouzi et al. (2013), where the mathematical expression is given as:

$$w_{t+1} = w_t + \Delta w_t \tag{19}$$

In Eq. (19), Δw_t indicates the fuzzy-adapting system-based weight.

In Zhang and Ding (2011), the authors have introduced the self-adaptive inertia weight and it has been linearly reduced as:

$$w = w^{\max} - n\left(\frac{w^{\max} - w^{\min}}{I_{\max}}\right) \tag{20}$$

where n indicates the particles in the population.

The adaptive inertia weight and the two constant parameters that have been developed in Jiang et al. (2013), Wang et al. (2012) are represented in Eqs. (21), (22) and (23). Here, ε denotes the shrink factor and b_1 as well as b_2 refers to the constant factors. At the starting stage, $b_1 > b_2$.

$$w = (w^{\max} - w^{\min}) \times e^{-\varepsilon t} + w^{\min} \tag{21}$$

$$c_1 = (b_2 - b_1)t/I_{\max} + b_1 \tag{22}$$

$$c_2 = (b_2 - b_1)t/I_{\max} + b_2 \tag{23}$$

Even more, the fuzzy system has maintained the inertia weight, as in Eq. (24) Naderi and Narimani (2017).

$$w^m = w^c + \Delta w \tag{24}$$

where w^m indicates the modified inertia weight, w^c denotes the present inertia weight and Δw refers to the correction for upgrading the inertia weight.

3.2. Adaptivess in other parameters

The Pitch Adjusting Rate (PAR) and the bandwidth have been adaptively changed in Zhao et al. (2015). The associated formulations are provided in Eqs. (25) and (26). In Eq. (27), R refers to the PAR, $R_{\max}$ and $R_{\min}$ indicates the maximum and the minimum PAR and N_s refers to the count associated with the generation of solution vectors. On the contrary, in Eq. (26), B denotes the bandwidth, M denotes the harmony memory and$d$ refers to the dimension of the fitness function.

$$R = R_{\min} + \frac{(R_{\max} - R_{\min})}{(1 + \exp(20^{*}t/N_s - 10))} R_{\min} \tag{25}$$

$$B = median(M(:,i)), \quad i = 1, 2 \ldots d \tag{26}$$

The self-learning PSO that has been developed in Zuo et al. (2014) is based on the four upgrading strategies in velocity, where one strategy is similar to the standard PSO and the other three strategies are provided by Eqs. (28), (29) and (30). In those equations, y_{kd} and y_{jd} denotes the arbitrary selected particles, p_{id} and p_{kd} indicates the *pbest* of the selected particles and s indicates a number, which is uniformly distributed among [0, 1].

$$v_{id}(t) = y_{kd}(t) - y_{jd}(t) \tag{27}$$

$$v_{id}(t+1) = s v_{id}(t) + s[p_{id}(t) - y_{id}(t)] \tag{28}$$

$$v_{id}(t+1) = w v_{id}(t) + sq[p_{kd}(t) - y_{id}(t)] \tag{29}$$

$$v_{id}(t+1) = w v_{id}(t) + 0.5[p_{kd}(t) - y_{id}(t) + p_{id}(t) - y_{id}(t)] \tag{30}$$

With the mutation rate M^{rate} in Eq. (33), the velocities and the initial values of the entire particles have been self-adapted in Leia et al. (2015) and they are expressed by Eqs. (31) and (32). Here, v_m and z_m indicates the velocity and the position of the mth particle, $p_m^{best}(t)$ indicates the pbest of the m^{th} particle and *gbest* indicates the global best of the entire particles.

$$v_m(t+1) = w_0 \times v_m(t) + M^{rate}(p_m^{best}(t) - gbest - 2z_m(t)) \tag{31}$$

$$z_m(t+1) = z_m(t) + v_m(t+1) \tag{32}$$

$$M^{rate} = 1 + \frac{gbest}{p_m^{best}} \tag{33}$$

$$w_0 = 1.1 + \frac{gbest}{p_m^{best}} \tag{34}$$

According to Kumar et al. (2016), the self-adaptive velocity and position of the particle can be represented by Eq. (34). Here, w represents the inertia weight between [0, 1]; η specifies the acceleration coefficient and f_i indicates the random number between [0, 1]. The average velocity of the entire particle is denoted as $v_{avg}(t)$, as in Eq. (37).

$$v_i(t+1) = w v_i(t) + \eta f_i(p_i^{best} - x_i(t)) \tag{35}$$

$$x_i(t+1) = x_i(t) + v_{avg}(t) \tag{36}$$

$$v_{avg}(t) = \sum_{i=1}^{N} \frac{v_i(t)}{P_s} \tag{37}$$

Subsequently, the self-adaptive check and repair operator has been introduced into the PSO algorithm in Chih (2015). The associated profit utility and profit density ratios are as shown in Eqs. (38) and (39).

$$\delta_{mn}^{utility} = \frac{c_{mn}}{a_{mn}} \tag{38}$$

$$\delta_{mn}^{density} = \frac{c_{mn}.b_m}{a_{mn}} \tag{39}$$

Consequently, the time-varying accelerating coefficients (C_a and C_b) have been adaptively changed in Mandal et al. (2015). This is stipulated, as in Eqs. (40) and (41), where C^{1f} and C^{1i} denotes the initial and the final values of the cognitive acceleration factors and C^{2f} and C^{2i} indicates the initial and the final values of the social acceleration factors.

$$C_a = (C^{1f} - C^{1i}) \frac{t}{I_{\max}} + C^{1i} \tag{40}$$

$$C_b = (C^{2f} - C^{2i}) \frac{t}{I_{\max}} + C^{2i} \tag{41}$$

Moreover, the authors of Chen and Hsiao (2016) have used the self-adaptive PSO, with alteration in the position and the velocity update (as in Eqs. (45) and (42)). The constants, c_1 and c_2, of these equations are used to determine the limiting factor κ.

$$V_{mn}(t+1) = \kappa \begin{pmatrix} V_{mn}(t) + c_1 \times rand_1(pbest_{mn} - z_{mn}(t)) \\ + c_2 \times rand_2(gbest_{mn} - z_{mn}(t)) \end{pmatrix} \tag{42}$$

$$\kappa = \frac{2}{|2 - c - \sqrt{c^2 - 4c}|} \tag{43}$$

$$c = c_1 + c_2, \quad c_2 > 4 \tag{44}$$

$$z_{mn}(t+1) = z_{mn}(t) + V_{mn}(t+1) \tag{45}$$

The contribution of self-adaptive PSO in Elsayed et al. (2014) is the addition of the constriction factor τ^x and the alteration in the parameters such as, c_1 and c_2. Hence, Eqs. (46), (47) and (48) depicts the respective formulations. In those equations, $\dot{\tau}$ and $\ddot{\tau}$ represents the lower and the upper bounds of τ^x, $\dot{c}_1$ as well as $\ddot{c}_1$ indicates the lower and the upper bounds of c_1 and $\dot{c}_2$ as well as $\ddot{c}_2$ denotes the lower and the upper bounds of c_2, respectively. Subsequently, the newly updated velocity based on these parameters is shown in Eqs. (49), (50) and (51). Here, a_1, a_2 and a_3 refers to the random numbers that are uniformly distributed between [0, 1].

$$\tau^x = \ddot{\tau} + rand \times (\dot{\tau} - \ddot{\tau}) \tag{46}$$

$$c_1^x = \ddot{c}_1 + rand \times (\dot{c}_1 - \ddot{c}_1) \tag{47}$$

$$c_2^x = \ddot{c}_2 + rand \times (\dot{c}_2 - \ddot{c}_2) \tag{48}$$

$$v_\tau^x = 0.5a_1(\tau^x + a_2(pbest\tau^x - \tau^x) + a_3(gbest\tau - \tau^x)) \tag{49}$$

$$v_\tau^x = 0.5a_1(c_1^x + a_2(pbestc_1^x - c_1^x) + a_3(gbestc_1 - c_1^x)) \tag{50}$$

$$v_\tau^x = 0.5a_1(c_2^x + a_2(pbestc_2^x - c_2^x) + a_3(gbestc_2 - c_2^x)) \tag{51}$$

According to Semwal and Rastogi (2016), the velocity and the position update formulation can be represented by Eq. (52), where $v_i(t)$ and $x_i(t)$ are the velocity and the position of the current iteration and Δt is the time interval that takes a value of 1.

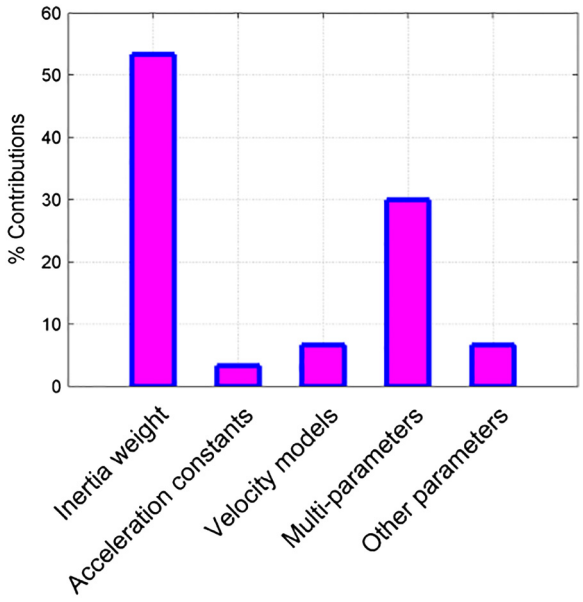

Fig. 3. Percentage contributions on PSO with different adaptiveness.

$$v_i(t+1) = w v_i(t) + c_1 rand\left(\frac{pbest_i - gbest_i}{\Delta t}\right) + c_2 rand\left(\frac{pbest_i - gbest_i}{\Delta t}\right) \quad (52)$$

$$x_i(t+1) = x_i(t) + v_i(t)\Delta t \quad (53)$$

In Wang et al. (2011), the self- adaptive PSO has been enhanced by altering the velocity model, as given in Eq. (53). Here, γ and d represents the constant factors as well as $v_i(t)$ and $x_i(t)$ represents the velocity and the position of the selected particle, at iteration t. In addition, k is the constant factor that is computed by the maximum value $k^{\max}$ and the minimum value $k^{\min}$ of k. The update of inertia weight is based on Eq. (20).

$$v_i(t+1) = k \times \begin{bmatrix} w \times v_i(t) + c_1 \times rand() \times (pbest_i - x_i(t)) \\ +c_2 \times rand() \times (gbest_i - x_i(t)) \end{bmatrix} \quad (54)$$

$$k = \begin{cases} k^{\max} - (k^{\max} - k^{\min}) \times 2^{(f(gbest^t) - f(gbest^{t-d}))/\gamma} & if\ t > d \\ k^{\max} & if\ t \leqslant d \end{cases} \quad (55)$$

Moreover, the authors of Qi (2011) have developed the self-adaptive PSO, having alteration in the velocity as well as the position of the particle and the inertia weight. In Eq. (55) to Eq. (58), β

Table 3
Review on the optimization of VLSI design and its design parameters.

Author [Citation]	Optimization	Parameters							
		Area	Time	Temperature	Bandwidth	Clock cycle	Frequency	Power consumption	Other parameters
Chung and Kuo (2008)	Power optimization					✓			
Xu et al. (2010)	Optimize the wire length and thermal area in 3D ICs	✓		✓					
Massimo et al. (2010)	Optimization of wire grid size	✓						✓	
Lihong and Zheng (2011)	Retargeting and optimizing the performance-constrained template-based layout	✓			✓				Gain
Song et al. (2011)	Optimization of statistical lifetime reliability	✓							
Saraju et al. (2012)	Optimization of power	✓						✓	Percentage of reduction
Aliakbar and Majid, 2013	Optimization of quantum cost								Quantum cost
Andrew et al. (2013)	Optimization of neural architecture	✓	✓						Frame length
Oghenekarho et al. (2014)	Optimization of power consumption with temperature measurement	✓		✓				✓	Sensitivity, voltage
Byunghyun and Taewhan (2014)	Optimization of data path	✓				✓			Speed, wire length,
Saraju et al. (2014)	Reduction of total leakage power								Gate leakage, propagation delay
Es-Sakhi and Chowdhury (2015)	Minimization of threshold swing								Film thickness, density
Vishnu et al. (2015)	Optimization of sine and cosine values		✓						Modulation angle
Varun et al. (2016)	Energy optimization		✓					✓	
Anthony et al. (2016)	Optimization of controllability and observability test								Percentage of fault
Jui-Hung et al. (2017)	Reduction of smallest possible rate–distortion cost among coding BW constraints				✓		✓		Search range, bit rate
Passos et al. (2017)	Reduction of error and simulating time	✓	✓				✓		Phase noise, quality factor, inductance

represents the self-adaptive coefficient, $\Delta\sigma$ indicates the standard error of normal Gaussian distribution, ρ refers to the coefficient of increment and α indicates the coefficient for controlling the velocity attenuation of the particle.

$$v_i(t+1) = (1-\rho)w_i(t) + \rho N(0,\sigma_i(t)) + c_1 b_1(pbest_i - x_i(t)) + c_2 b_2(gbest_i - x_i(t)) \quad (56)$$

$$x_i(t+1) = x_i(t) + v_i(t+1) \quad (57)$$

$$w_i(t) = \beta\left(1 - f(x_i(t))/f(x_j(t))\right) + (1-\beta)w_i(0)\exp(-\alpha i^2) \quad (58)$$

$$\sigma_i(t+1) = \sigma_i(t)\exp(N_i(0,\Delta\sigma)) \quad (59)$$

The various contributions that are based on PSO, with different adaptiveness such as, inertia weight, acceleration constants, velocity models, multi-parameters and other parameters such as, PAR, mutation rate, profile utilizty, profile density ratio and constriction factor, are shown in Fig. 3. Here, the adaptiveness on inertia weight is 52% better than the acceleration constants, 47% better than the velocity models as well as the other parameters and 25% better than the multi-parameters.

4. Optimization of VLSI design

The review on the general optimization of VLSI design and its design parameters is shown in Table 3. In 2008, Chung and Kuo (2008) have designed CMOS with optimized power. Later, in 2010, Xu et al. (2010) have optimized the wire length and thermal area of 3D Ics. In contrast, Massimo et al. (2010) have optimized the wire grid size of the differential MOS system. Further in 2011, Lihong and Zheng (2011) have retargeted and optimized the performance-constrained template-based layout for analog IC and Song et al. (2011) have improved the statistical lifetime reliability of the chip. Moreover, in 2012, Saraju et al. (2012) have optimized the power in CMOS Static Random Access Memories (SRAMs). Subsequently, in 2013, Aliakbar and Majid (2013) have optimized the quantum cost of the combinational Multiple-Valued Reversible Logic (MVRL) circuits. Further, Andrew et al. (2013) have made one of the best neural architectures for the nano-CMOS era.

Afterwards, in 2014, Oghenekarho et al. (2014) have reduced power consumption with temperature measurement in nano-CMOS, while Byunghyun and Taewhan (2014) have performed the optimization of data path in a 3D-stacked IC. In addition, Saraju et al. (2014) have reduced the total leakage power in a robust nanoscale chip design. In 2015, Es-Sakhi and Chowdhury (2015) have attained minimum threshold swing in the Partially Depleted Silicon-on-Ferroelectric Insulator Field Effect Transistor (PD SOFFET), whereas Vishnu et al. (2015) have optimized the sine and the cosine values in the digital domain design of VLSI. Subsequently, in 2016, Varun et al. (2016) have contributed energy optimization in the dynamic data view framework of VLSI design, whereas Anthony et al. (2016) have optimized the controllability and the observability tests of analog IC. Even more, in 2017, Jui et al. (2017) have lowered the smallest possible rate-distortion cost among the coding bandwidth constraints of the application processor systems and Passos et al. (2017) have reduced the error and the simulating time that was taken for the design of RF Circuit.

On considering the design parameters, area has been considered by 52.94% of the contributions, whereas 23.52% of the contributions have considered time and power consumption. Moreover, 11.76% of the contributions have considered parameters such as, temperature, bandwidth, clock cycle and frequency. The additional parameters that are considered are the gain, the percentage of reduction, the quantum cost, etc.

5. Floor-planning in VLSI

The design of IC tends to be complex, when the count of transistors used in VLSI become infinite. Under such circumstances, the floor planning mechanism is used, since it is a well-arranged methodology. In general, floor planning can predict the feedback that is associated with the design of architecture, chip area estimation and the congestion as well as the delays caused by wiring. In 2015, Maji et al. (2015) have proposed an evolutionary algorithm, called Craziness-Based PSO (CRPSO), for promoting optimization in the floor planning of VLSI chip. They have focussed on the objective of minimizing the wire length and the area of the chip. The meta-heuristic methods offer the most effectual optimal solution for the VLSI-based fixed outline constraints floorplanning. A floorplan has an area cost, which is measured by the area of the smallest rectangle that encloses all the modules, and an interconnection cost (specifically, the Wire length cost, which is the total length of the wires that fulfill the interconnections between the modules). Metaheuristics offer an adequately optimal solution for an optimization problem, particularly, with imperfect information or limited computational capacity. In fact, Metaheuristics might create some assumptions about the optimization problem being solved and as a result, they may be exploitable for a variety of problems. Moreover, in 2008, Fernando and Katkoori (2008) have developed a multi-objective genetic algorithm to reduce the area and the wire length. Furthermore, in 2011, Han et al. (2011) have suggested the new GPU-based floorplanning algorithm, which evaluated numerous concurrent moves that evaluate the multiple concurrent moves. Even more, Singha et al. (2012) have dealt with the Prototypes Optimization with Evolved Improvement (POEMS) algorithm to solve the problems, regarding floor planning in VLSI, through the usage of GA in 2012 itself. In 1990, Wing et al. (1990) have implemented the hierarchical method for achieving floor-planning in VLSI, which locates and routes proper data-paths in the chips.

6. Research gaps and future directions

The improvement in the silicon chip technology has been witnessed over the past decades, and now, it has extended from the development of 2D to 3D chips (Loh and Xie, 2010). Even though the complex problems associated with chip manufacturing has been solved, additional improvements through the invention of new techniques are essential. The performance of the 3D chips exhibit improvement over the 2D chips, as they can diminish the wire length and the interconnection delay with the stacked location of components. One of the critical problems arising in the chip manufacturing process is the attainment of undesirable temperature, which could be solved only by optimized floor-planning. In fact, a number of components such as, memory, processors and linking elements are used to manufacture the chip. While manufacturing a chip, the main objective of exhibiting a perfect trade-off between the performance and the temperature should be considered (Cuesta, 2012). The frequency and the delay in transformation might influence the wire length of the chip. Moreover, the communication of data becomes slow due to the corruption of carrier mobility, which is an effect from high temperature. The feasibility and the life expectancy of the chip are also affected by high temperature. Thus, the best performance in VLSI design can be attained by lowering the temperature and the wire length. Therefore, different meta-heuristic algorithms like, Simulated Annealing (Kessler and Ganesan, 1985) and Genetic Algorithm (Michalewicz, 1996) are used to solve the existing problems and to attain those

objectives. Though different algorithms have been developed to solve the 2D floorplanning problems, the computational complexity still exists as an unsolved problem (Mansor, 2016; Hsu et al., 2006). Further, the 3D floor planning problems are usually solved by the techniques like, Mixed Integer Linear Programming, (MILP). But, this model remains complex, due to the larger problem size. The adoption of several recent meta-heuristic algorithms can evade this issue. In addition, multi objective is considered as another direction, where more constraints and variables that affect the quality of the physical design are present. Moreover, the case partitioning is too regarded with multiple objectives such as, delay, power and area, with respect to the minimum cut. In case of the floor planning issue, VLSI designing can be extended to contemplate the other objectives, namely, chip area, size, delay of critical path and total wire length. Additionally, the few approaches can also be extended to resolve the other VLSI physical issues and it would be necessary to analyze the different objective functions, which affect the quality of final solutions in the VLSI design methodology.

7. Conclusion

This paper has surveyed the recent trends in the advancements and further developments of VLSI design. Most of the electronic circuits are depending on the VLSI design due to its smaller size, less expense, less power consumption and proper functionality with high quality. Accordingly, this paper has reviewed about 52 papers, which are associated with VLSI design using optimization. Moreover, the optimization of the VLSI design with various bio-inspired algorithms has also been analyzed in this paper. In addition, the respective performance measures have also been evaluated to fix the optimal values accurately. Further, various issues and major challenges of VLSI have also been reviewed. A review on the different contributions of SA-PSO and the floor planning problems of VLSI design has also been made. Finally, the research gaps and the future directions have also been revealed through this review.

References

Aiman, H. El-Maleh, 2017. A probabilistic pairwise swap search state assignment algorithm for sequential circuit optimization. Integr. VLSI J. 56, 32–43.

Aiman, H. El-Maleh, Sadiq, M. Sait, Abubakar, Bala, 2015. State assignment for area minimization of sequential circuits based on cuckoo search optimization. Comput. Electr. Eng. 44, 13–23.

Akram, Malak, Yao, Li, Ramy, Iskander, Francois, Durbin, Farakh, Javid, Jean-Marc, Guebhard, Marie-Minerve, Louërat, André, Tissot, 2015. Fast multidimensional optimization of analog circuits initiated by monodimensional global Peano explorations. Integr. VLSI J. 48, 198–212.

Ali, Husseinzadeh Kashan, 2011. An efficient algorithm for constrained global optimization and application to mechanical engineering design: League championship algorithm (LCA). Comput. Aided Des. 43 (12), 1769–1792.

Aliakbar, Niknafs, Majid, Mohammadi, 2013. Synthesis and optimization of multiple-valued combinational and sequential reversible circuits with don't cares. Integr. VLSI J. 46, 189–196.

Allen, J., Terman, C.J., 2000. An interactive learning environment for VLSI design. Proc. IEEE 88 (1), 96–106.

Andrew, S. Cassidy, Julius, Georgiou, Andreas, G. Andreou, 2013. Design of silicon brains in the nano-CMOS era: spiking neurons, learning synapses and neural architecture optimization. Neural Netw. 45, 4–26.

Anthony, Coyette, Baris, Esen, Wim, Dobbelaere, Ronny, Vanhooren, Georges, Gielen, 2016. Automatic generation of test infrastructures for analog integrated circuits by controllability and observability co-optimization. Integr. VLSI J. 55, 393–400.

Ardakani, A., Leduc-Primeau, F., Onizawa, N., Hanyu, T., Gross, W.J., 2016. VLSI implementation of deep neural networks using integral stochastic computing. In: 9th International Symposium on Turbo Codes and Iterative Information Processing (ISTC), Brest, 2016, pp. 216–220.

Areibi, S., Yang, Z., 2004. Effective memetic algorithms for VLSI Design = Genetic Algorithms + Local Search + Multi-Level Clustering. Evol. Comput. 12 (3), 327–353.

Arora, N.D. et al., 2005. Interconnect characterization of X architecture diagonal lines for VLSI design. IEEE Trans. Semicond. Manuf. 18 (2), 262–271.

Aubepart, F., Poure, P., Chapuis, Y.A., Girerd, C., Braun, F., 2003. HDL-based methodology for VLSI design of AC motor controller. IEE Proc. – Circuits Devices Syst 150 (1), 38–44.

Awad, A., Takahashi, A., Tanaka, S., Kodama, C., 2017. A fast process-variation-aware mask optimization algorithm with a novel intensity modeling. IEEE Trans. Very Large Scale Integr. (VLSI) Syst. 25 (3), 998–1011.

Balaji, G.N., Subashini, T.S., Chidambaram, N., 2015a. Detection of heart muscle damage from automated analysis of echocardiogram video. IETE J. Res. 61 (3), 236–243.

Balaji, G.N., Subashini, T.S., Chidambaram, N., 2015b. Automatic classification of cardiac views in echocardiogram using histogram and statistical features. Proce. Com. Sci. 46, 1569–1576.

Bahmani-Firouzi, Bahman, Farjah, Ebrahim, Azizipanah-Abarghooee, Rasoul, 2013. An efficient scenario-based and fuzzy self-adaptive learning particle swarm optimization approach for dynamic economic emission dispatch considering load and wind power uncertainties. Energy 50, 232–244.

Behnam, Khodabandeloo, Ahmad, Khonsari, Masoomeh, Jasemi, Golnaz, Taheri, 2017. A fast temperature-aware fixed-outline floorplanning framework using convex optimization. Integr. VLSI J. 58, 101–110.

Bernard, C., 1983. The bottom-left bin packing heuristic: an efficient implementation. IEEE Trans. Comput. 32 (8), 697–707.

Bo, Yan, Liu, Yu, Wang, Leibo, Zhiping, Miao, Liu, Zheng, Li, Lu, Wang, Fernandez, JingFrancisco V., 2009. Analog circuit optimization system based on hybrid evolutionary algorithms. Integr. VLSI J 42 (2), 137–148.

Byunghyun, Lee, Taewhan, Kim, 2014. Algorithms for TSV resource sharing and optimization in designing 3D stacked ICs. Integr. VLSI J. 47 (2), 184–194.

Chakraborty, K., Lvov, A., Mukherjee, M., 2006. Novel algorithms for placement of rectangular covers for mask inspection in advanced lithography and other VLSI design applications. IEEE Trans. Comput. Aided Des. Integr. Circuits Syst. 25 (1), 79–91.

Chander, Akhilesh, Chatterjee, Amitava, Siarry, Patrick, 2011. A new social and momentum component adaptive PSO algorithm for image segmentation. Expert Syst. Appl. 38 (5), 4998–5004.

Chang, R.C., Lim, B.H., 2004. Efficient IP routing table VLSI design for multigigabit routers. IEEE Trans. Circuits Syst. I Regul. Pap. 51 (4), 700–708.

Che, JinXing, 2013. Support vector regression based on optimal training subset and adaptive particle swarm optimization algorithm. Appl. Soft Comput. 13 (8), 3473–3481.

Chen, Mu-Chen, Hsiao, Yu-Hsiang, Reddy, ReddivariHimadeep, Tiwari, Manoj Kumar, 2016. The self-learning particle swarm optimization approach for routing pickup and delivery of multiple products with material handling in multiple cross-docks. Transp. Res. Part E: Logistics Transp. Rev. 91, 208–226.

Chen, J.M., Wei, C.H., 1999. VLSI design for high-speed LZ-based data compression. IEE Proc. – Circuits Devices Syst. 146 (5), 268–278.

Chih, Mingchang, 2015. Self-adaptive check and repair operator-based particle swarm optimization for the multidimensional knapsack problem. Appl. Soft Comput. 26, 378–389.

Chung, B., Kuo, J.B., 2008. Gate-level dual-threshold static power optimization methodology (GDSPOM) using path-based static timing analysis (STA) technique for SOC application. Integr. VLSI J. 41 (1), 9–16.

Chyi-Shiang, Hoo, Kanesan, Jeevan, Velappa, Ganapathy, Harikrishnan, Ramiah, 2013. Variable-order ant system for VLSI multiobjective floorplanning. Appl. Soft Comput. 13 (7), 3285–3297.

Cuesta, David, Risco-Martín, José L., Ayala, José L., 2012. 3D thermal-aware floorplanner using a MILP approximation. Microprocess. Microsyst. 36 (5), 344–354.

Elsayed, Saber M., Sarker, Ruhul A., Mezura-Montes, Efrén, 2014. Self-adaptive mix of particle swarm methodologies for constrained optimization. Inf. Sci. 277, 216–233.

Elumalai, Vinodh Kumar, Ganapathy, Subramanian Raaja, Jovitha, Jerome, 2016. Adaptive PSO for optimal LQR tracking control of 2 DoF laboratory helicopter. Appl. Soft Comput. 41, 77–90.

Es-Sakhi, A.D., Chowdhury, M.H., 2015. Partially depleted silicon-on-ferroelectric insulator field effect transistor- parametrization & design optimization for minimum subthreshold swing. Microelectron. J. 46 (10), 981–987.

Fan, Qinqin, Yan, Xuefeng, 2014. Self-adaptive particle swarm optimization with multiple velocity strategies and its application for p-Xylene oxidation reaction process optimization. Chemometrics Intell. Lab. Syst. 139, 15–25.

Fernando, P., Katkoori, S., 2008. An Elitist non-dominated sorting based genetic algorithm for simultaneous area and wirelength minimization in VLSI floorplanning. In: 21st International Conference on VLSI Design (VLSID 2008), Hyderabad, pp. 337–342.

Funke, J., Hougardy, S., Schneider, J., 2016. An exact algorithm for wirelength optimal placements in VLSI design. Integr. VLSI J. 52, 355–366.

Gabriel H. Loh, Yuan Xie, 2010. Loh3D stacked micro processor: are we there yet? IEEE Micro. 3, 60–64.

Giuseppe, Nicosia, Salvatore, Rinaudo, Eva, Sciacca, 2008. An evolutionary algorithm-based approach to robust analog circuit design using constrained multi-objective optimization. Knowl.-Based Syst. 21 (3), 175–183.

Han, Y., Chakraborty, K., Roy, S., Kuntamukkala, V., 2011. A GPU algorithm for IC floorplanning: specification, analysis and optimization. In: 2011 24th International Conference on VLSI Design, Chennai, pp. 159–164.

He, Zhong-Li, Tsui, Chi-Ying, Chan, Kai-Keung, Liou, M.L., 2000. Low-power VLSI design for motion estimation using adaptive pixel truncation. IEEE Trans. Circuits Syst. Video Technol. 10 (5), 669–678.

Hodge, A.M., Newcomb, R.W., 2002. Semistate theory and analog VLSI design. IEEE Circuits Syst. Mag. 2 (2), 30–51. Second Quarter.
Hsia, Shih-Chang, 2003. Parallel VLSI design for a real-time video-impulse noise-reduction processor. IEEE Trans. Very Large Scale Integr. VLSI Syst. 11 (4), 651–658.
Hsiao, Shen-Fu, Tai, Yor-Chin, Chang, Kai-Hsiang, 2000. VLSI design of an efficient embedded zerotree wavelet coder with function of digital watermarking. IEEE Trans. Consum. Electron. 46 (3), 628–636.
Hsu, H.Y., Wu, A.Y., Yeo, J.C., 2006. Area-Efficient VLSI Design of Reed-Solomon Decoder for 10GBase-LX4 Optical Communication Systems. IEEE Trans Circuits Syst. II: Express Briefs 53 (11), 1245–1249.
Huang, Wei, Ghosh, S., Velusamy, K., Sankaranarayanan, K., Skadron, K., Stan, M.R., 2006. HotSpot: a compact thermal modeling methodology for early-stage VLSI design. IEEE Trans. Very Large Scale Integr. VLSI Syst. 14 (5), 501–513.
HyunJin, Kim, Sungho, Kang, 2011. Communication-aware task scheduling and voltage selection for total energy minimization in a multiprocessor system using Ant Colony Optimization. Inf. Sci. 181 (18), 3995–4008.
Jiang, Yan, Li, Xuyong, Huang, Chongchao, 2013. Automatic calibration a hydrological model using a master–slave swarms shuffling evolution algorithm based on self-adaptive particle swarm optimization. Expert Syst. Appl. 40 (2), 752–757.
Jianli, Chen, Yan, Liu, Ziran, Zhu, Wenxing, Zhu, 2017. An adaptive hybrid memetic algorithm for thermal-aware non-slicing VLSI floorplanning. Integr. VLSI J.
Judhisthir, Dash, Bivas, Dam, Rajkishore, Swain, 2017. Optimal design of linear phase multi-band stop filters using improved cuckoo search particle swarm optimization. Appl. Soft Comput. 52, 435–445.
Jui-Hung, Hsieh, Jian-Hao, Huang, Hung-Ren, Wang, 2017. DVFS-aware motion estimation design scheme based on bandwidth–rate–distortion optimization in application processor systems. Integr. VLSI J. 57, 74–80.
Karimi-Nasaba, M., Modarresb, M., Seyedhoseinic, S.M., 2015. A self-adaptive PSO for joint lot sizing and job shop scheduling with compressible process times. Appl. Soft Comput. 27, 137–147.
Kessler, A.J., Ganesan, A., 1985. Standard cell VLSI design: a tutorial. IEEE Circuits Devices Mag. 1 (1), 17–34.
Kim, Jae-Gon, Kim, Yeong-Dae, 2003. A linear programming-based algorithm for floorplanning in VLSI design. IEEE Trans. Comput. Aided Des. Integr. Circuits Syst. 22 (5), 584–592.
Kumar, Ravi Shankar, Kondapaneni, Karthik, Dixit, Vijaya, Goswami, A., Thakur, L.S., Tiwar, M.K., 2016. Multi-objective modeling of production and pollution routing problem with time window: a self-learning particle swarm optimization approach. Comput. Ind. Eng. 99, 29–40.
Lee, S.W., Lim, S.C., 2006. VLSI design of a wavelet processing core. IEEE Trans. Circuits Syst. Video Technol. 16 (11), 1350–1361.
Leia, Baiying, Zhoub, Feng, Tanc, Ee-Leng, Nia, Dong, Leid, Haijun, Chena, Siping, Wanga, Tianfu, 2015. Optimal and secure audio watermarking scheme based on self-adaptive particle swarm optimization and quaternion wavelet transform. Signal Process. 113, 80–94.
Levent, Aksoy, Cristiano, Lazzari, Eduardo, Costa, Paulo, Flores, José, Monteiro, 2012. High-level algorithms for the optimization of gate-level area in digit-serial multiple constant multiplications. Integr. VLSI J. 45 (3), 294–306.
Li, Xiao-li, Li, Li-hong, Zhang, Bao-lin, Guo, Qian-jin, 2013. Hybrid self-adaptive learning based particle swarm optimization and support vector regression model for grade estimation. Neurocomputing 118, 179–190.
Lihong, Zhang, Zheng, Liu, 2011. Directly performance-constrained template-based layout retargeting and optimization for analog integrated circuits. Integr. VLSI J. 44 (1), 1–11.
Mahor, Amita, Rangnekar, Saroj, 2012. Short term generation scheduling of cascaded hydro electric system using novel self adaptive inertia weight PSO. Int. J. Electr. Power Energy Syst. 34 (1), 1–9.
Maji, K.B., Ghosh, A., Kar, R., Mandal, D., Ghoshal, S.P., 2015. An evolutionary algorithm based approach for VLSI floor-planning. In: 2015 International Conference on Science and Technology (TICST). Pathum Thani, pp. 248–253.
Mandal, K.K., Mandal, S., Bhattacharya, B., Chakraborty, N., 2015. Non-convex emission constrained economic dispatch using a new self-adaptive particle swarm optimization technique. Appl. Soft Comput. 28, 188–195.
Mansor, Mohd. Asyraf, Shareduwan, Mohd., Kasihmuddin, Mohd., Sathasivam, Saratha, 2016. VLSI circuit configuration using satisfiability logic in Hopfield network. I.J. Intell. Syst. Appl. 9, 22–29.
Manuel, Barros, Jorge, Guilherme, Nuno, Horta, 2010. Analog circuits optimization based on evolutionary computation techniques. Integr. VLSI J. 43 (1), 136–155.
Maryam, Dehbashian, Mohammad, Maymandi-Nejad, 2017b. A new hybrid algorithm for analog ICs optimization based on the shrinking circles technique. Integr. VLSI J. 56, 148–166.
Maryam, Dehbashian, Mohammad, Maymandi-Nejad, 2017a. An enhanced optimization kernel for analog IC design automation using the shrinking circles technique. Eng. Appl. Artif. Intell. 58, 62–78.
Massimo, Alioto, Stéphane, Badel, Yusuf, Leblebici, 2010. Optimization of the wire grid size for differential routing: analysis and impact on the power-delay-area tradeoff. Microelectron. J. 41 (10), 669–679.
Mehdi, Taassori, Meysam, Taassori, Sadegh, Niroomand, Béla, Vizvári, Sener, Uysal, Abdollah, Hadi-Vencheh, 2015. OPAIC: an optimization technique to improve energy consumption and performance in application specific network on chips. Measurement 74, 208–220.
Mei, K., Zheng, N., van de Wetering, H., 2006. High-speed and memory-efficient VLSI design of 2D DWT for JPEG2000. Electron. Lett. 42 (16), 907–908.
Michael, Farnsworth, Ashutosh, Tiwari, Meiling, Zhu, 2017. Multi-level and multi-objective design optimisation of a MEMS bandpass filter. Appl. Soft Comput. 52, 642–656.
Michalewicz, Zbigniew, 1996. Genetic Algorithms + Data Structures = Evolution Programs. Springer-Verlag, Berlin.
Milad, Kaboli, Behzad, Ghanavati, Majid, Akhlaghi, 2017. A new CMOS pseudo approximation exponential function generator by modified particle swarm optimization algorithm. Integr. VLSI J. 56, 70–76.
Mohammed, El-Abd, Hassan, Mohab, Hassan, Kamel, AnisMohamed S., Elmasry, Mohamed, 2010. Discrete cooperative particle swarm optimization for FPGA placement. Appl. Soft Comput. 10 (1), 284–295.
Montalvo, Idel, Izquierdo, Joaquín, Perez-García, Rafael, Herrera, Manuel, 2010. Improved performance of PSO with self-adaptive parameters for computing the optimal design of Water Supply Systems. Eng. Appl. Artif. Intell. 23 (5), 727–735.
Naderi, Ehsan, Narimani, Hossein, Fathi, Mehdi, Narimani, Mohammad Rasoul, 2017. A novel fuzzy adaptive configuration of particle swarm optimization to solve large-scale optimal reactive power dispatch. Appl. Soft Comput. 53, 441–456.
Niknam, Taher, Farsani, EhsanAzad, 2010. A hybrid self-adaptive particle swarm optimization and modified shuffled frog leaping algorithm for distribution feeder reconfiguration. Eng. Appl. Artif. Intell. 23 (8), 1340–1349.
Oghenekarho, Okobiah, Saraju, P. Mohanty, Elias, Kougianos, 2014. Nano-CMOS thermal sensor design optimization for efficient temperature measurement. Integr. VLSI J. 47 (2), 195–203.
Passos, F., Roca, E., Castro-Lopez, R., Fernández, F.V., 2017. An inductor modeling and optimization toolbox for RF circuit design. Integr. VLSI J.
Philipp, Hungerländer, Miguel, F. Anjos, 2015. A semidefinite optimization-based approach for global optimization of multi-row facility layout. Eur. J. Oper. Res. 245 (1), 46–61.
Ping, Yang, Haiying, Yang, Wei, Qiu, Shuting, Wang, Chuanquan, Li, 2014. Optimal approach on net routing for VLSI physical design based on Tabu-ant colonies modeling. Appl. Soft Comput. 21, 376–381.
Povoa, R., Bastos, I., Lourenco, N., Horta, N., 2016. Automatic synthesis of RF front-end blocks using multi-objective evolutionary techniques. Integr. VLSI J. 52, 243–252.
Qu, Gang, 2002. Publicly detectable watermarking for intellectual property authentication in VLSI design. IEEE Trans. Comput. Aided Des. Integr. Circuits Syst. 21 (11), 1363–1368.
Rajit, Karmakar, Santanu, Chattopadhyay, 2015. Window-based peak power model and Particle Swarm Optimization guided 3-dimensional bin packing for SoC test scheduling. Integr. VLSI J. 50, 61–73.
Rastegar, Saeid, Araujo, Rui, Mendes, Jerome, 2017. Online identification of Takagi-Sugeno fuzzy models based on self-adaptive hierarchical particle swarm optimization algorithm. Appl. Math. Model. 45, 606–620.
Revna, Acar Vural, Ozan, Der, Tulay, Yildirim, 2010. Particle swarm optimization based inverter design considering transient performance. Digital Signal Process. 20 (4), 1215–1220.
Revna, Acar Vural, Ozan, Der, Tulay, Yildirim, 2011. Investigation of particle swarm optimization for switching characterization of inverter design. Expert Syst. Appl. 38 (5), 5696–5703.
Ricardo, Martins, Nuno, Lourenco, Nuno, Horta, 2015. Multi-objective optimization of analog integrated circuit placement hierarchy in absolute coordinates. Expert Syst. Appl. 42 (23), 9137–9151.
Sadiq, M. Sait, Umair, F. Siddiqi, 2016. A stochastic evolution algorithm based 2D VLSI global router. Integr. VLSI J. 53, 115–125.
Saraju, P. Mohanty, Jawar, Singh, Elias, Kougianos, Dhiraj, K. Pradhan, 2012. Statistical DOE–ILP based power–performance–process (P3) optimization of nano-CMOS SRAM. Integr. VLSI J. 45 (1), 33–45.
Saraju, P. Mohanty, Mahadevan, Gomathisankaran, Elias, Kougianos, 2014. Variability-aware architecture level optimization techniques for robust nanoscale chip design. Comput. Electr. Eng. 40 (1), 168–193.
Semwal, Girish, Rastogi, Vipul, 2016. Design optimization of long period waveguide grating devices for refractive index sensing using adaptive particle swarm optimization. Opt. Commun. 359, 336–343.
Sheu, Ming-Hwa, Lin, Su-Hon, Chen, Chichyang, Yang, Shyue-Wen, 2004. An efficient VLSI design for a residue to binary converter for general balance moduli $(2n-3, 2n+1, 2n-1, 2n+3)$. IEEE Trans. Circuits Syst. II Express Briefs 51 (3), 152–155.
Simon, P., Luchies, J.M., Maly, W., 2000. Identification of plasma-induced damage conditions in VLSI designs. IEEE Trans. Semicond. Manuf. 13 (2), 136–144.
Sina, Basir-Kazeruni, Hao, Yu, Fang, Gong, Yu, Hu, Chunchen, Liu, Lei, He, 2013. SPECO: stochastic perturbation based clock tree optimization considering temperature uncertainty. Integr. VLSI J. 46 (1), 22–32.
Singha, T., Dutta, H.S., De, M., 2012. Optimization of floor-planning using genetic algorithm. Procedia Technol. 4, 825–829.
Song, Jin, Yinhe, Han, Huawei, Li, Xiaowei, Li, 2011. Statistical lifetime reliability optimization considering joint effect of process variation and aging. Integr. VLSI J. 44 (3), 185–191.
Subhojit, Ghosh, Susovon, Samanta, 2014. Fixed structure compensator design using a constrained hybrid evolutionary optimization approach. ISA Trans. 53 (4), 1119–1130.
Sudha, N., Sridharan, K., 2004. A high-speed VLSI design and ASIC implementation for constructing Euclidean distance-based discrete Voronoi diagram. IEEE Trans. Robot. Autom. 20 (2), 352–358.

Sun, Fei, Zhang, Tong, 2005. Parallel high-throughput limited search trellis decoder VLSI design. IEEE Trans. Very Large Scale Integr. VLSI Syst. 13 (9), 1013–1022.
Taher Kourany, Maged Ghoneima, Emad Hegazi, Yehea Ismail, 2016. PASSIOT: a Pareto-optimal multi-objective optimization approach for synthesis of analog circuits using Sobol' Indices-based Engine. In: 2016 IEEE 59th International Midwest Symposium on Circuits and Systems (MWSCAS), October, pp.16–19.
Taherkhani, Mojtaba, Safabakhsh, Reza, 2016. A novel stability-based adaptive inertia weight for particle swarm optimization. Appl. Soft Comput. 38, 281–295.
Thakker, R.A., Patil, M.B., Anil, K.G., 2009. Parameter extraction for PSP MOSFET model using hierarchical particle swarm optimization. Eng. Appl. Artif. Intell. 22 (2), 317–328.
Tsai, T.H., Yang, Y.C., 2004. Low power and cost effective VLSI design for an MP3 audio decoder using an optimised synthesis-subband approach. IEE Proc. – Comput. Digital Techniques 151 (3), 245–251.
Varun, Venkatesan, Swamy, D. Ponpandi, Akhilesh, Tyagi, 2016. Shaping data for application performance and energy optimization in dynamic data view framework. Integr. VLSI J.
Vicente, Juan D., Lanchares, Juan, Hermida, Román, 2004. Annealing placement by thermodynamic combinatorial optimization. ACM Trans. Des. Autom. Electron. Syst. 9, 310–332.
Vinod, Kumar Khera, Sharma, R.K., Gupta, A.K., 2017. A heuristic Fault based optimization Approach to reduce Test Vectors Count in VLSI Testing. J. King Saud Univ. – Comput. Inf. Sci.
Vishnu, G., Karthik, P., Jabeen, Fathima, 2015. VLSI design and implementation of efficient software defined radio using optimized quadrature direct digital frequency synthesizer on FPGA. Procedia Comput. Sci. 58, 414–421.
Wang, Yu, Li, Bin, Weise, Thomas, Wang, Jianyu, Yuan, Bo, Tian, Qiongjie, 2011. Self-adaptive learning based particle swarm optimization. Inf. Sci. 181 (20), 4515–4538.
Wang, Ying, Zhou, Jianzhong, Youlin, Lu., Qin, Hui, Wang, Yongqiang, 2011. Chaotic self-adaptive particle swarm optimization algorithm for dynamic economic dispatch problem with valve-point effects. Expert Syst. Appl. 38 (11), 14231–14237.
Wang, Ying, Zhou, Jianzhong, Zhou, Chao, Wang, Yongqiang, Qin, Hui, Youlin, Lu., 2012. An improved self-adaptive PSO technique for short-term hydrothermal scheduling. Expert Syst. Appl. 39 (3), 2288–2295.
Wing, K. Luk, Alvar, A. Dean, John, W. Mathews, 1990. Partitioning and floor-planning for data-path chip (microprocessor) layout. Integr. VLSI J. 10 (1), 89–108.
Wu, Qi, 2011. A self-adaptive embedded chaotic particle swarm optimization for parameters selection of Wv-SVM. Expert Syst. Appl. 38 (1), 184–192.
Xu, He, Sheqin, Dong, Yuchun, Ma, 2010. Signal through-the-silicon via planning and pin assignment for thermal and wire length optimization in 3D ICs. Integr. VLSI J. 43 (4), 342–352.
Zhai, Shijun, Jiang, Ting, 2015. A new sense-through-foliage target recognition method based on hybrid differential evolution and self-adaptive particle swarm optimization-based support vector machine. Neurocomputing 149 (Part B), 573–584.
Zhang, Jiuzhong, Ding, Xueming, 2011. A multi-swarm self-adaptive and cooperative particle swarm optimization. Eng. Appl. Artif. Intell. 24 (6), 958–967.
Zhang, Yanjun, Zhao, Yu, Fu, Xinghu, Xu, Jinrui, 2016. A feature extraction method of the particle swarm optimization algorithm based on adaptive inertia weight and chaos optimization for Brillouin scattering spectra. Opt. Commun. 376, 56–66.
Zhao, Fuqing, Liu, Yang, Zhang, Chuck, Wang, Junbiao, 2015. A self-adaptive harmony PSO search algorithm and its performance analysis. Expert Syst. Appl. 42 (21), 7436–7455.
Zhen, Yang Lim, Ponnambalam, S.G., Izui, Kazuhiro, 2017. Multi-objective hybrid algorithms for layout optimization in multi-robot cellular manufacturing systems. Knowl.-Based Syst. 120, 87–98.
Zuo, X., Zhang, G., Tan, W., 2014. Self-adaptive learning PSO-based deadline constrained task scheduling for hybrid IaaS cloud. IEEE Trans. Autom. Sci. Eng. 11 (2), 564–573.

Further reading

Chen, Tung-Chieh, Chang, Yao-Wen, 2006. Modern floor-planning based on Bn-tree and fast simulated annealing. IEEE Trans. Comput. Aided Des. Integr. Circuits Syst. 25, 637–650.