www.czasopisma.pan.pl PAN POLSKA AKADEMIA NAUK www.journals.pan.pl

INTL JOURNAL OF ELECTRONICS AND TELECOMMUNICATIONS, 2016, VOL. 62, NO. 2, PP. 179-186
Manuscript received July 4, 2015; revised June, 2016. DOI: 10.1515/eletel-2016-0025

Particle Swarm Optimization Algorithm for Leakage Power Reduction in VLSI Circuits

V.Leela Rani and M.Madhavi Latha

*Abstract***—*Leakage power is the dominant source of power dissipation in nanometer technology. As per the International Technology Roadmap for Semiconductors (ITRS) static power dominates dynamic power with the advancement in technology. One of the well-known techniques used for leakage reduction is Input Vector Control (IVC). Due to stacking effect in IVC, it gives less leakage for the Minimum Leakage Vector (MLV) applied at inputs of test circuit. This paper introduces Particle Swarm Optimization (PSO) algorithm to the field of VLSI to find minimum leakage vector. Another optimization algorithm called Genetic algorithm (GA) is also implemented to search MLV and compared with PSO in terms of number of iterations. The proposed approach is validated by simulating few test circuits. Both GA and PSO algorithms are implemented in Verilog HDL and the simulations are carried out using Xilinx 9.2i. From the simulation results it is found that PSO based approach is best in finding MLV compared to Genetic based implementation as PSO technique uses less runtime compared to GA. To the best of the author's knowledge PSO algorithm is used in IVC technique to optimize power for the first time and it is quite successful in searching MLV.*

*Keywords***—*Leakage power, PSO algorithm, Genetic algorithm, Minimum leakage vector, Verilog HDL implementation*

I. Introduction

ADVANCEMENTS in scaling with reduced threshold and supply voltages lead to increased leakages in MOS transistors. Many studies presented that leakage power consumption is up to 40% of total power consumption in nanometer technology [1]. Hence, reducing leakage power is of top concern in present day scenario. Number of techniques have been proposed previously to minimize leakage power by varying threshold voltage and adding sleep transistors [2][3][4][5]. Although above techniques are popular, they need extra process steps during fabrication. One of the popular approaches called Input Vector Control (IVC)is presented in this paper which is independent of process technology parameters [6]. Leakage current depends on input vector [7][8] due to stacking effect of transistors in the circuit. CMOS gate's sub threshold and gate oxide leakage currents vary with the input applied [9]. So, it is necessary to find input vectors which can optimize leakage. Forcing the circuit to low leakage state to reduce leakage power is the basic concept in IVC [10]. Minimum leakage vector is the low leakage state determined among different test inputs applied to circuit [11]. Furthermore, IVC does not require any circuit modifications and depends on transistor stacking effect. Fig1. shows design flow of IVC approach in sleep mode. Leakage power calculations for all input combinations of a test circuit are to be measured to form a Look Up Table. From Look Up Table, MLV is identified. This MLV is to be forced to a test circuit in a sleep mode to consume minimum power when it is in idle state.

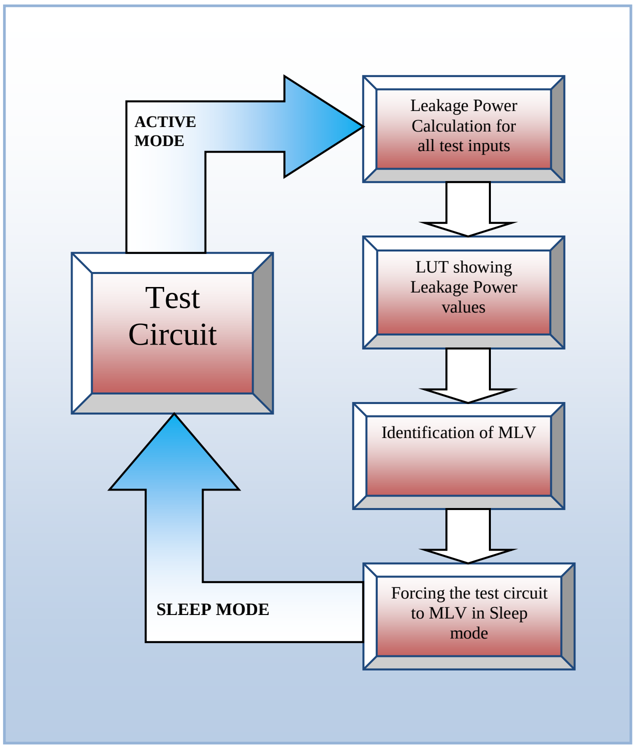

ACTIVE MODE
Leakage Power Calculation for all test inputs
LUT showing Leakage Power values
Test Circuit
Identification of MLV
SLEEP MODE
Forcing the test circuit to MLV in Sleep mode

Fig. 1. Design Procedure in IVC Approach

Many researchers have found that optimization algorithms are best in solving many complex problems in various fields of science and engineering. Researchers have used different algorithms like genetic algorithm, integer linear programming for power reduction in the area of VLSI. Genetic algorithm is used by many authors to determine the minimum leakage vector as best solution [12][13]. In the previous work of authors [11] genetic algorithm is used but implemented in Verilog HDL and comparison is performed with [14]. In this paper an attempt is made to implement Particle Swarm Optimization algorithm in the field of low power VLSI to search for MLV as an optimum solution. To the best of the author's knowledge, PSO algorithm is used for the first time in IVC approach to find MLV for leakage power reduction. The paper is organized as follows. Section II gives concepts of Genetic algorithm and section III presents brief description of Particle Swarm Optimization algorithm. Section IV deals with results and analysis of test circuits. Finally, section V concludes the paper.

V. Leela Rani is with G.V.P. College of Engineering (Autonomous), Department of ECE, Visakhapatnam, AP, India.
(email: leelarani.vanapalli@gmail.com).
M. Madhavi Latha is with JNTUH College of Engineering, Department of ECE, Hyderabad, Telangana, India.

II. Genetic Algorithm

Genetic algorithm is a subclass of evolutionary algorithms inspired by the biological process of genetics. Initial population is generated and represented as a set of chromosomes. Fitness of each chromosome is to be evaluated and two best values are to be stored as parents. Off springs are generated from parent1 and parent2 through different process steps like cross over and mutation. Best fitness off springs will survive to form next generations. Process will iterate through number of generations until the algorithm converges. The pseudo code for genetic algorithm is given below.

```
Genetic algorithm_mlv()
{
Population_size = n;
Chromosome length = Number of primary inputs
Generation = 1;
Initialize a population of chromosomes
do
{
Evaluate fitness of each chromosome in the population
Select parent chromosomes
Generate off springs using cross over and mutation
}
While(generation ++ < No. of generations)
}
```

Though genetic algorithm is popularly used optimization algorithm in many areas, it has a problem of premature convergence to a local value before reaching to global solution. Along with this, genetic algorithm consumes more computational time to generate optimum solution compared to other algorithms. Hence, particle swarm optimization algorithm is implemented in this paper to overcome the limitations of genetic algorithm.

III. Particle Swarm Optimization Algorithm

Classical approaches like exhaustive search methods can’t provide a suitable solution for large search spaces. Optimization algorithms can give solutions even if search space is more. Over the past few decades algorithms have been developed based on natural phenomena. PSO is one of such algorithms inspired by social behavior of bird flocking or fish schooling [15]. PSO is a population based stochastic optimization algorithm proposed by J.Kennedy and R.Eberhart in 1995 [16]. Overview of optimization procedure in PSO Algorithm is presented in Fig.2.Inputs to the algorithm are specifications and initial values of various parameters. Algorithm uses number of agents in the form of particles which constitutes a swarm. This swarm of particles moves around the search space to give the best solution. Each particle keeps track of its own best solution achieved so far called personal best (p_{best}) and best solution by neighborhood particles which is called as global best (g_{best}) [17].

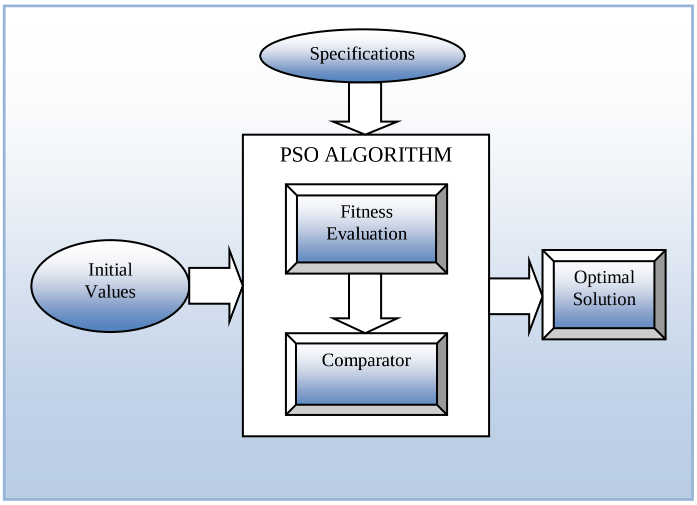


Fig. 2. Optimization Procedure using PSO

The basic concept in PSO algorithm is moving a particle from p_{best} to g_{best} position using position and velocity updation as shown in Fig.3. A particle moves towards a best solution called g_{best} in a search space by updating its velocity and position by following equations [18]

$$V_{k+1} = WV_k + C_1 r_1 (p_{best} - X_k) + C_2 r_2 (g_{best} - X_k) \quad (1)$$

$$X_{k+1} = X_k + V_{k+1} \quad (2)$$

where V_{k+1} is the velocity of the particle at (k+1)th iteration, X_k is the current position of the particle, X_{k+1} is the position of the particle at (k+1)th iteration after updating from current position. r_1 and r_2 are the random numbers between 0 and 1, C_1 and C_2 are accelerating factors. W is inertia weight. To have better performance and fast convergence, parameter values of the algorithm are taken as $C_1=C_2=2$, W=1. Objective of the fitness function is to find the minimum value of the required parameter. Equation (1)&(2) shows position and velocity updates at (k+1)th iteration with the initial position values generated in the predetermined search space. Initial position vector is the initial p_{best} value. P_{best} at iteration (k+1) is updated using the below equations.

$$p_{best\,(k+1)} = \begin{cases} p_{best\,(k)} & \text{if} \quad f(p_{best\,(k)}) \leq f(x_{(k+1)}) \\ x_{(k+1)} & \text{if} \quad f(p_{best\,(k)}) > f(x_{(k+1)}) \end{cases} \quad (3)$$

g_{best} value is updated from minimum of all p_{best} values as follows

$$g_{best} = \min\{ p_{best}^1, p_{best}^2, p_{best}^3 \ldots\} \quad (4)$$

Computation time is dramatically decreased with simple concepts in the proposed algorithm compared to other heuristic algorithms. In PSO, a balance between global and local exploration of the search space is there to eliminate premature convergence. This prevents from being trapped to local minima. For this a proper choice of inertia weight is required. Pseudo code for the algorithm is given below.

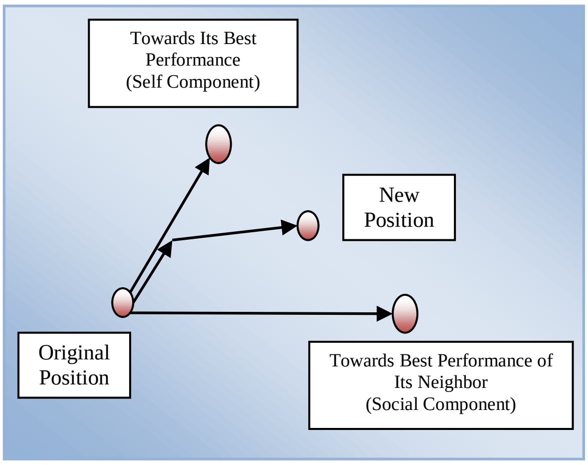

Towards Its Best Performance (Self Component)
New Position
Original Position
Towards Best Performance of Its Neighbor (Social Component)

Fig. 3. Velocity and Position Updates in PSO Algorithm

```
PSO algorithm_mlv ( );
Swarm_size = n
Particle length = No. of primary inputs
Generation = 1
Initialize a random swarm of particles
Initialize position and velocity
Initialize control parameters
Evaluate fitness of each particle in the swarm
Find initial $p_{best}$ and $g_{best}$
do
{
Velocity and position updates to generate new swarm of particles
If (fitness of new particles < fitness of old particles)
Update $p_{best}$ and $g_{best}$
While (generation++ < No. of Generations)
}
End
```

Fig.4. presents flowchart of the PSO algorithm. Process of the algorithm is initiated with random population of particles, which forms the swarm. Position, velocity and other control parameters are also initialized and fitness of initial population is to be evaluated to find p_{best} and g_{best}. Velocity and position updates are made to generate new population. Fitness values of new population are to be evaluated and compared with old fitness values. If fitness of new population is less than old population, p_{best} and g_{best} are to update, if not, old best values are carried out for the next iteration. Complete process is carried out till the stop criteria is satisfied and algorithm converges to best solution.

IV. Experimental Analysis

PSO and Genetic algorithms are implemented in Verilog HDL to find minimum leakage vector. Simulation and synthesis is carried out using Xilinx 9.2i.Table I shows leakage power values for all possible input combinations of a two input NAND gate. These leakage power values in Table I are measured using H-spice tool and are used in calculating overall leakage power of test circuits.

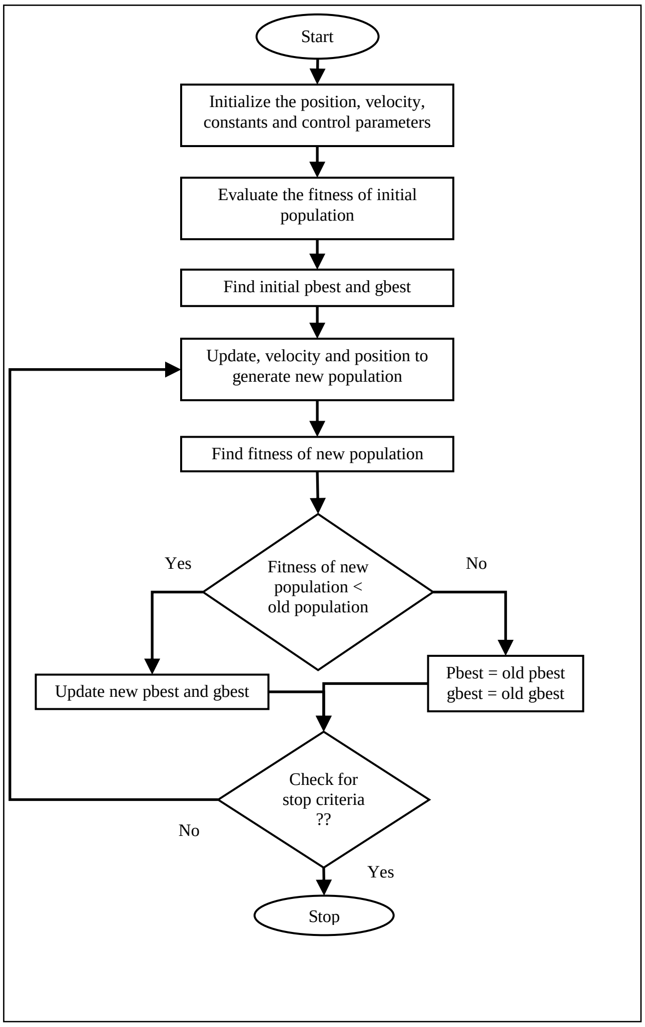

Start
Initialize the position, velocity, constants and control parameters
Evaluate the fitness of initial population
Find initial pbest and gbest
Update, velocity and position to generate new population
Find fitness of new population
Fitness of new population < old population
Yes
No
Update new pbest and gbest
Pbest = old pbest
gbest = old gbest
Check for stop criteria ??
No
Yes
Stop

Fig. 4. PSO Algorithm Flowchart

TABLE I
Leakage Power Values for Two Input NAND Gate

S. No.	Input Vector	Leakage power(w)
1	00	154.77f (best)
2	01	5.73p
3	10	5.44p
4	11	10.48p(worst)

Among the four possible test inputs “00” offers least leakage power and “11” offers highest leakage power. So,”00” is known to be MLV which can offer minimum leakage. Both the algorithms are applied to 4input, 6input, 8input, 10input NAND gate based test circuits and one benchmark circuit C17. Fig. 5 to Fig. 9 are the combinational test circuits examined for the effectiveness of algorithm.

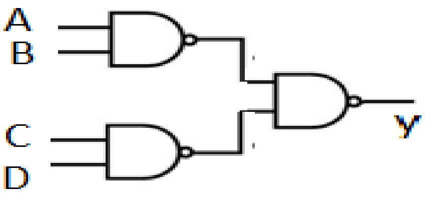


Fig. 5. Four Input NAND based Test Circuit

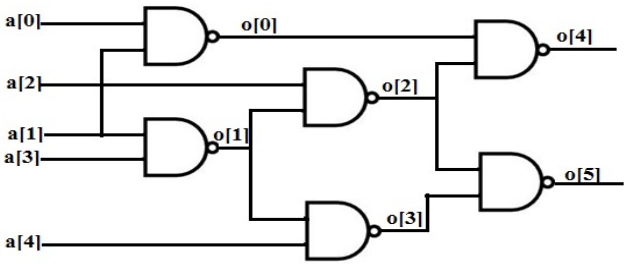


Fig. 6. C17 Benchmark Circuit

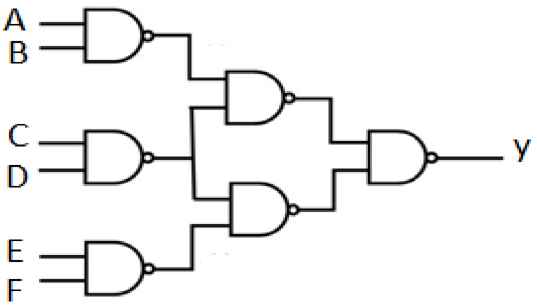


Fig. 7. Six Input NAND based Test Circuit

Fig.10 and Fig.11 presents simulation results of PSO algorithm using Xilinx for a test case of 10 input circuit. From the simulation results in Fig. 10, it can be observed that “pop” is a swarm size and is taken as 8. “g_{best}” is the final optimum solution and is found as “0000000000”. This is the minimum leakage vector for 10 input circuit which can give minimum leakage power. As given in Table I minimum leakage combination for NAND gate is “00”. Hence, 10 input test circuit consisting of NAND gates should give minimum leakage for all inputs at ‘0’state.This is proved with the algorithm, as it has given same value as optimum solution. Number of iterations algorithm takes to converge is “62” for the 10 input circuit as shown in Fig. 11.Simulation results of a C17 benchmark circuit are presented in Fig. 12 and Fig.13 for a swarm size of 10. From simulation results it can be observed that algorithm takes one iteration only to converge to optimal value called g_{best} i.e. “10100”.

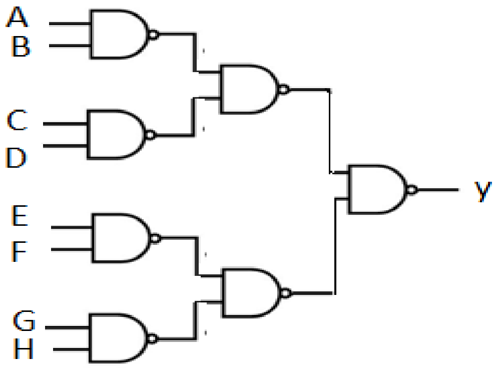


Fig. 8. Eight Input NAND based Test Circuit

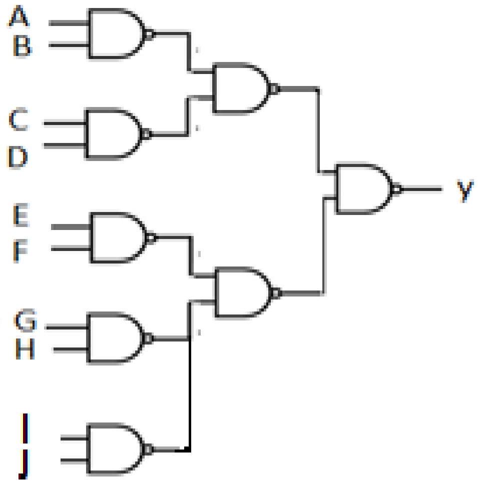
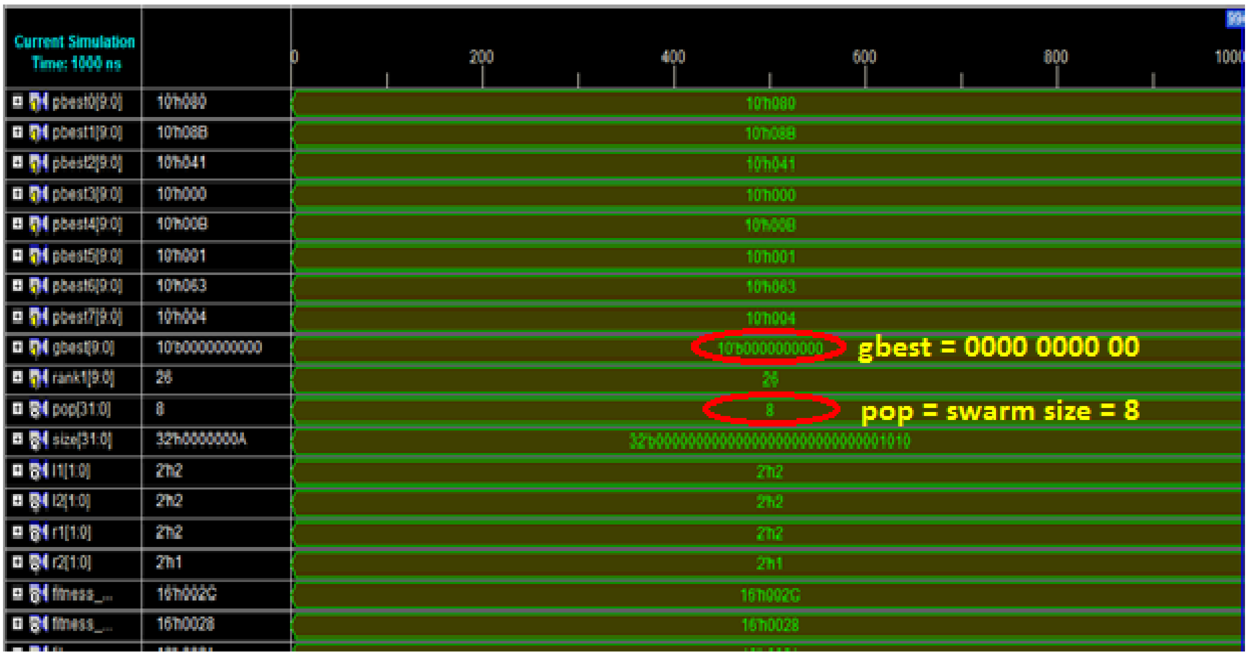


Fig.9. Ten input NAND based test circuit

All test circuits are checked using CADENCE Spectre in exhaustive approach also. Table II shows the results of PSO algorithm for two different swarm sizes of 8 and 50. From the results it can be observed that algorithm converges quickly to the optimal solution with increasing global search space.

Proper choice of parameters is made to avoid the algorithm to be trapped into local minima. Fig.14 illustrates PSO algorithm applied to various test circuits for different swarm sizes. The proposed heuristic algorithm has given same solution as that of exhaustive approach. Table III and Fig.15 presents comparison of genetic and PSO algorithms in terms of number of iterations. Simulation has been carried out for swarm size of 8. Results explore that PSO algorithm takes less time to converge to optimal solution compared to genetic algorithm. Table IV presents minimum leakage vector for each test circuit. Column 3 shows total leakage current of the circuit. Leakage power of test circuit is calculated by adding sum of leakage power of individual gates [19].



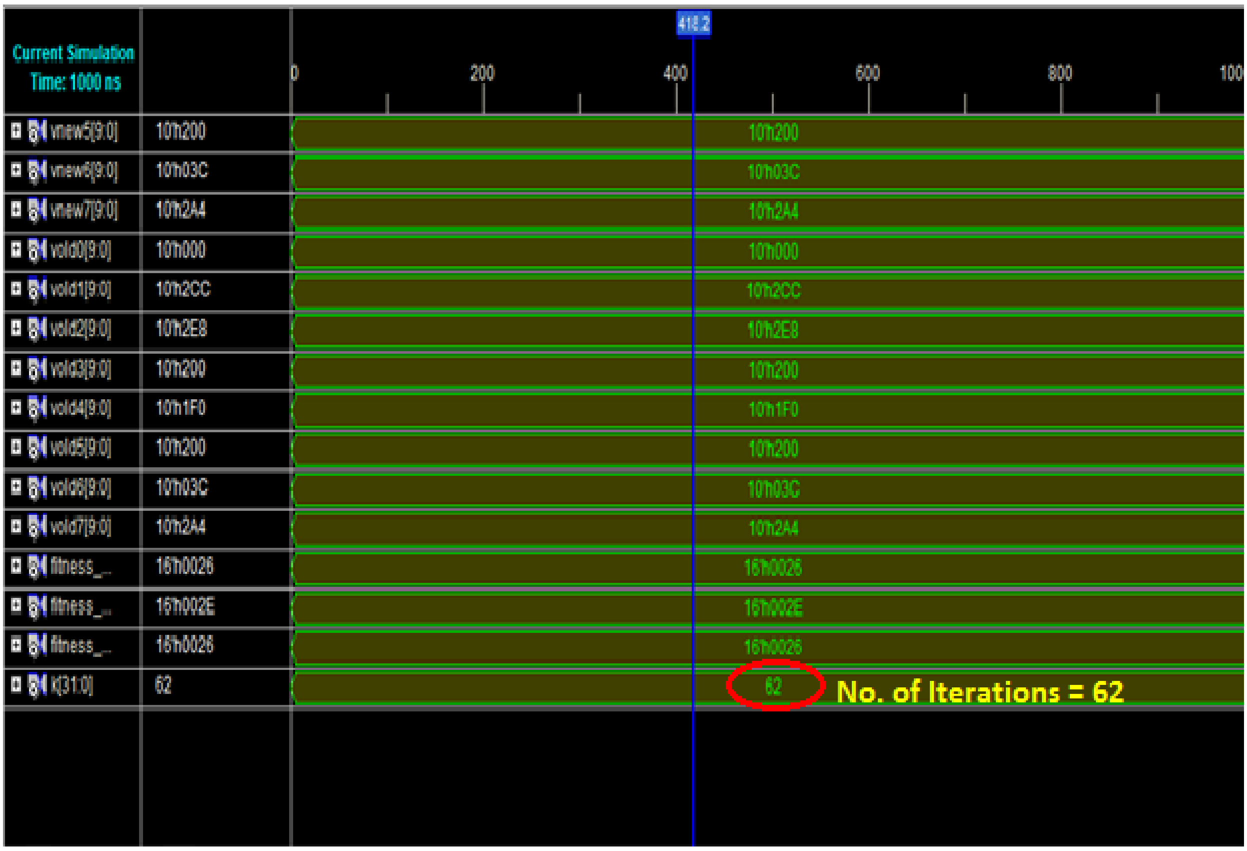
Current Simulation Time: 1000 ns	
pbest0[9:0]	10'h080
pbest1[9:0]	10'h08B
pbest2[9:0]	10'h041
pbest3[9:0]	10'h000
pbest4[9:0]	10'h00B
pbest5[9:0]	10'h001
pbest6[9:0]	10'h063
pbest7[9:0]	10'h004
gbest[9:0]	10'b0000000000
rank1[9:0]	28
pop[31:0]	8
size[31:0]	32'h0000000A
l1[1:0]	2'h2
l2[1:0]	2'h2
r1[1:0]	2'h2
r2[1:0]	2'h1
fitness_...	16'h002C
fitness_...	16'h0028

32'b00000000000000000000000000001010

gbest = 0000 0000 00

pop = swarm size = 8

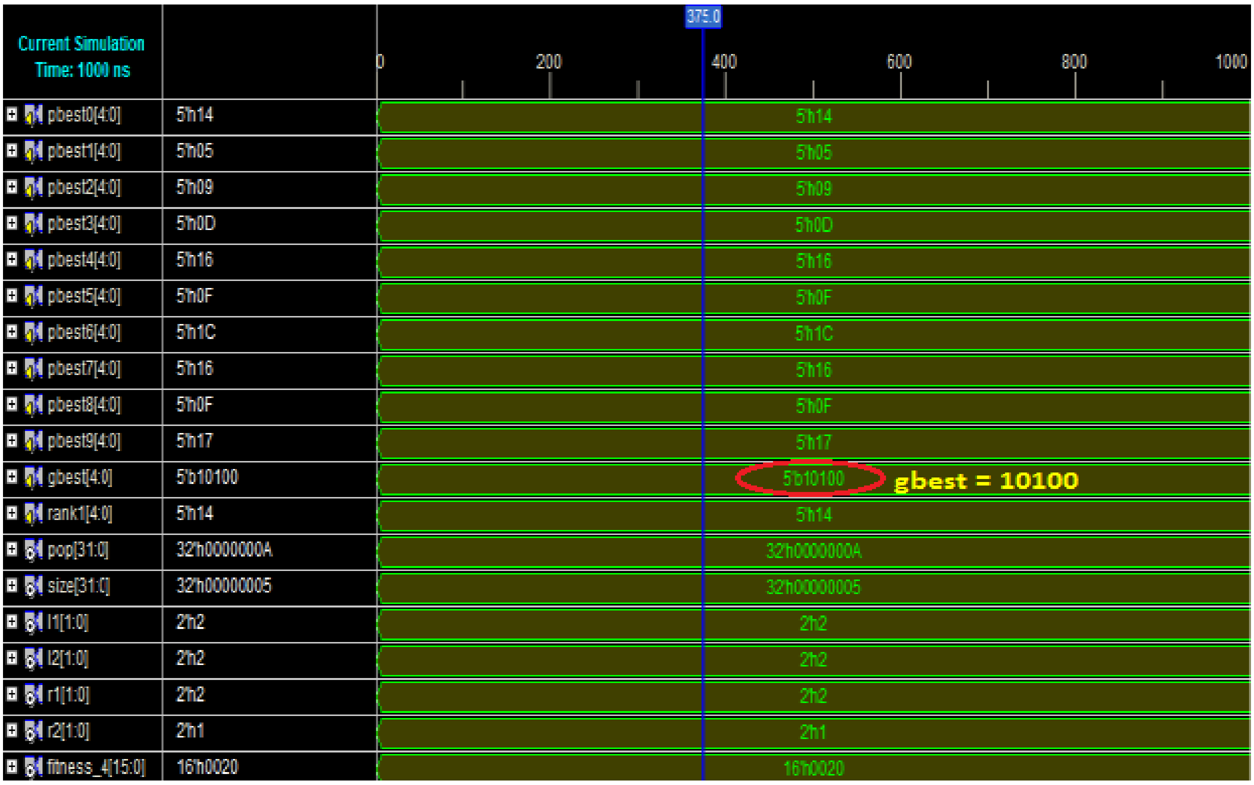
Fig. 10. Simulation results of PSO Algorithm for 10 input circuit



Current Simulation Time: 1000 ns	
vnew5[9:0]	10'h200
vnew6[9:0]	10'h03C
vnew7[9:0]	10'h2A4
vold0[9:0]	10'h000
vold1[9:0]	10'h2CC
vold2[9:0]	10'h2E8
vold3[9:0]	10'h200
vold4[9:0]	10'h1F0
vold5[9:0]	10'h200
vold6[9:0]	10'h03C
vold7[9:0]	10'h2A4
fitness_...	16'h0026
fitness_...	16'h002E
fitness_...	16'h0026
k[31:0]	62

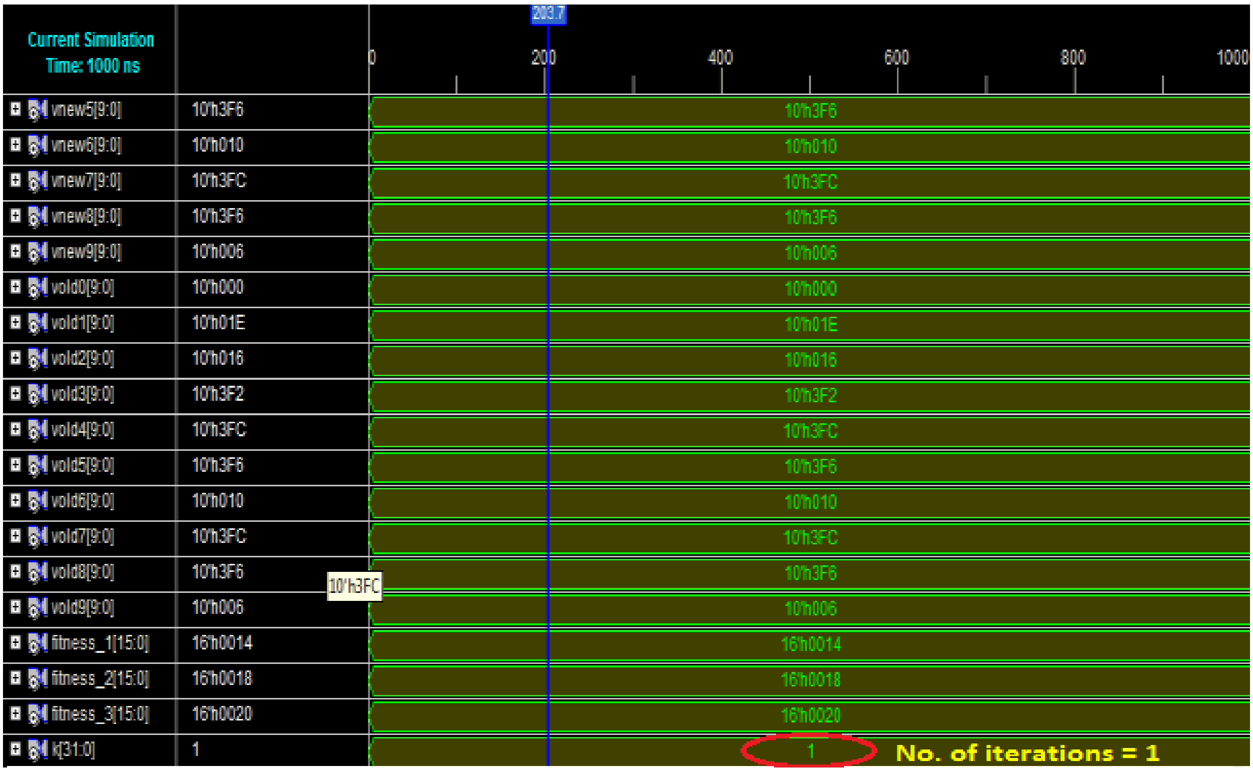
No. of Iterations = 62

Fig. 11. Simulation results of PSO Algorithm for 10 input circuit



Current Simulation Time: 1000 ns		
pbest0[4:0]	5'h14	5'h14
pbest1[4:0]	5'h05	5'h05
pbest2[4:0]	5'h09	5'h09
pbest3[4:0]	5'h0D	5'h0D
pbest4[4:0]	5'h16	5'h16
pbest5[4:0]	5'h0F	5'h0F
pbest6[4:0]	5'h1C	5'h1C
pbest7[4:0]	5'h16	5'h16
pbest8[4:0]	5'h0F	5'h0F
pbest9[4:0]	5'h17	5'h17
gbest[4:0]	5'b10100	5'b10100
rank1[4:0]	5'h14	5'h14
pop[31:0]	32'h0000000A	32'h0000000A
size[31:0]	32'h00000005	32'h00000005
l1[1:0]	2'h2	2'h2
l2[1:0]	2'h2	2'h2
r1[1:0]	2'h2	2'h2
r2[1:0]	2'h1	2'h1
fitness_4[15:0]	16'h0020	16'h0020

Fig. 12. Simulation results of PSO Algorithm for C17 Circuit



Current Simulation Time: 1000 ns		
vnew5[9:0]	10'h3F6	10'h3F6
vnew6[9:0]	10'h010	10'h010
vnew7[9:0]	10'h3FC	10'h3FC
vnew8[9:0]	10'h3F6	10'h3F6
vnew9[9:0]	10'h006	10'h006
vold0[9:0]	10'h000	10'h000
vold1[9:0]	10'h01E	10'h01E
vold2[9:0]	10'h016	10'h016
vold3[9:0]	10'h3F2	10'h3F2
vold4[9:0]	10'h3FC	10'h3FC
vold5[9:0]	10'h3F6	10'h3F6
vold6[9:0]	10'h010	10'h010
vold7[9:0]	10'h3FC	10'h3FC
vold8[9:0]	10'h3F6	10'h3F6
vold9[9:0]	10'h006	10'h006
fitness_1[15:0]	16'h0014	16'h0014
fitness_2[15:0]	16'h0018	16'h0018
fitness_3[15:0]	16'h0020	16'h0020
k[31:0]	1	1

Fig. 13. Simulation results of PSO Algorithm for C17 Circuit

TABLE II.
PSO ALGORITHM APPLIED TO DIFFERENT TEST CIRCUITS

Test circuit	Swarm size	Number of iterations
		PSO Algorithm
4 input circuit	8	1
	50	1
C17 Bench mark circuit	8	1
	50	1
6 input circuit	8	5
	50	1
8 input circuit	8	26
	50	2
10 input circuit	8	62
	50	2

TABLE III
COMPARISON OF GENETIC AND PSO ALGORITHMS

Test circuit	Swarm size	Number of iterations	
		Genetic Algorithm	PSO Algorithm
4 input circuit	8	1	1
C17	8	1	1
6 input circuit	8	15	5
8 input circuit	8	34	26
10 input circuit	8	96	62

TABLE IV
MINIMUM LEAKAGE VECTOR FOR TEST CIRCUITS

Test circuit	MLV	Total Leakage Power (watts)
4 input circuit	0000	10.792p
C17 Bench mark circuit	10100	32.717p
6 input circuit	000000	21.585p
8 input circuit	00000000	21.739p
10 input circuit	0000000000	21.894p

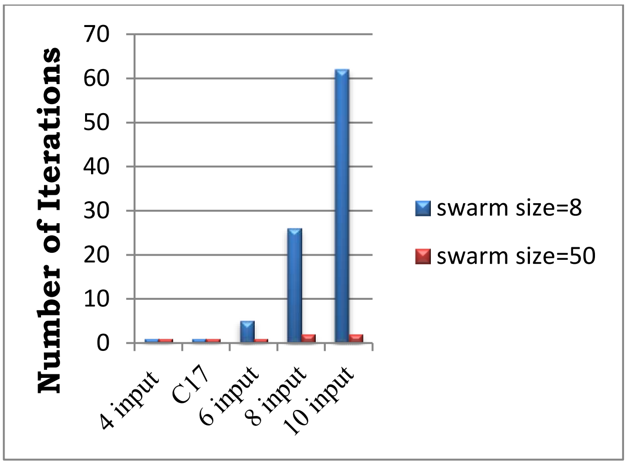


Fig. 14. PSO Algorithm applied to Various Test Circuits for different Swarm Sizes

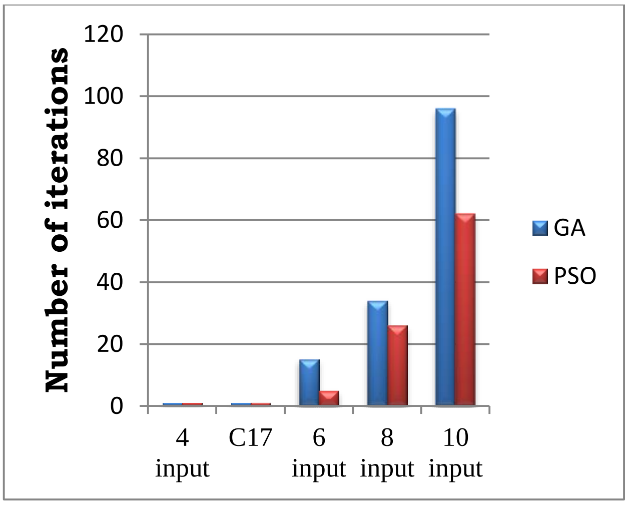


Fig. 15. Comparison of GA and PSO Algorithms

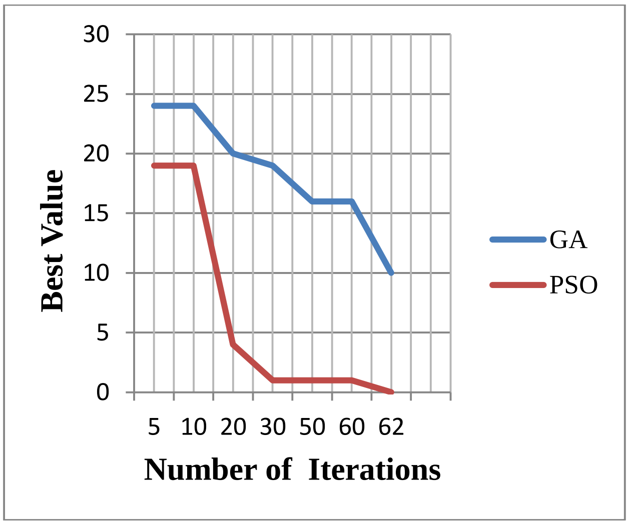


Fig. 16. Comparison of GA and PSO Algorithms for swarm size of 8

Best solution from the algorithm is the input vector which can give minimum leakage. As mentioned above, test circuit can be forced to this input vector in sleep mode to reduce static power [7] [10] [20]. Fig.16 presents a graph showing comparison of GA and PSO algorithms for a test case called 10 input NAND based circuit for a swarm size of 8. From the graph it can be observed that PSO algorithm converges at 62nd iteration and can give optimal solution, whereas genetic algorithm is still iterating through search space to find best solution.

V. Conclusion

This paper implemented a heuristic approach called Particle Swarm Optimization algorithm to find minimum leakage vector as optimal solution. Algorithm is validated on a set of combinational test circuits. An attempt is made to implement PSO and Genetic algorithms using Verilog HDL and successful to have MLV as optimal solution. From the simulation results it is concluded that PSO algorithm converges to best solution with runtime savings compared to Genetic algorithm. The PSO technique proved its effectiveness in finding minimum leakage vector. Finally, the results obtained by PSO algorithm in most cases gives superior results and in all cases are comparable with GA. The proposed algorithm can be extended to examine sequential circuits in future.

References

[1] Ing-Chao-Lin,Chin-Hong-lin,Kuan-Hui-li,"Leakage and aging optimization using Transmission gate based technique", IEEE Transactions on Computer aided design of integrated circuits and systems, vol.32, No.1, Jan2013.

[2] Kaushik Roy, Saibal Mukhopadhyay and Hamid Mahmoodi-Meimand, "Leakage Current Mechanisms and Leakage Reduction Techniques in Deep-Sub micrometer CMOS Circuits," PROCEEDINGS OF THE IEEE, VOL. 91, NO. 2, FEBRUARY 2003.

[3] Se Hun Kim, Vincent J. Moone, "Sleepy keeper:a New Approach to Low-leakage Power VLSI Design," VLSI SOC conference, PP. 367-372,2006.

[4] Jun Cheol Park and Vincent J.Moone III , "Sleepy Stack Leakage Reduction", IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION(VLSI)SYSTEMS, vol.14, No.11, pp. 1250-1263, November 2006.

[5] Farzan Fallah, Massoud Pedram "Standby and Active Leakage Current Control and Minimization in CMOS VLSI Circuits" in IEICE Transactions, pp.509-519,2005.

[6] Zhanping chen,Liqiong Wei,Mark johnson,Kaushik Roy, "Estimation of standby leakage power in CMOS circuits considering Accurate modeling of Transistor stacks", ACM transactions, international symposium on Low power electronics and design, ISLPED, pp.239-244,1998.

[7] Nikhil Jayakumar, Sunil P Khatri,"An Algorithm to Minimize Leakage through Simultaneous Input Vector Control and Circuit Modification", IEEE Design Automation conference,DATE07,2007.

[8] Wei wang,Yu Hu,Yin-He Han,Xiao-Wei Li,You sheng Zhang, "Leakage current optimizations during test based on don't care bits assignments", Journal of computer science and technology,pp.673-680,Sep 2007.

[9] Lin Yuan, Gang Qu, " A combined gate replacement and input vector control approach for leakage current reduction" , IEEE transactions on very large scale integration (vlsi) systems, vol. 14, no. 2, february 2006

[10] Afsin abdollahi,Farzan fallah,Massoud pedram, "Leakage current reduction in CMOS vlsi circuits by input vector control", IEEE transactions on very large scale integration (vlsi) systems, vol. 12, no. 2, february 2004.

[11] V.LeelaRani, M.Madhavi Latha, "Implementation of genetic algorithm for minimum leakage vector in input vector control approach", IEEE conference,SPACES,Jan2015.

[12] Zhanping chen,Liqiong Wei,Mark johnson,Kaushik Roy, "Estimation of standby leakage power in CMOS circuits considering Accurate modeling of Transistor stacks", ACM transactions, international symposium on Low power electronics and design, ISLPED, pp.239-244,1998.

[13] Mehdi Kamal, AliAfzali-Kusha, SaeedSafari, MassoudPedram, "Design of NBTI-resilient extensible processors", INTEGRATION, the VLSI Journal,vol.49,pp.22-34,2015.

[14] Abdoul Rjoub,Almotasem Belleah Alajlouni,Hassan Almanasrah, " A Fast input Vector Control Approach for subthreshold Leakage power reduction" , IEEE electro technical international conference,MELECON,2012.

[15] Mourad Fakhfakh, Yann Cooren, Amin Sallem, Mourad Loulou, Patrick Siarry, "Analog circuit design optimization through the particle swarm optimization technique", Analog Integrated Circuits and Signal processinh,vol. 63, pp.71-82, Springer, 2010.

[16] James Kennedy,Russell Eberhart, " Partcle Swarm Optimization", IEEE international conference, 1995.

[17] Christopher W. Cleghorn,Andries P. Engelbrecht, "A generalized theoretical deterministic particle swarm model" , Swarm Intelligence,vol.8,pp.35-59, Springer,2014.

[18] Mohammad Reza Bonyadi, Zbigniew Michalewicz," A locally convergent rotationally invariant particle swarm optimization algorithm", Swarm Intelligence,vol.8,pp.159-198, Springer,2014.

[19] Xiaoying Zhao, Jiangfang Yi, Dong Tong, Xu Cheng, "Leakage power reduction for cmos combinational circuits", IEEE international conference on solid state and integrated circuit technology,2006.

[20] Xiaotao Chang, Dongrui Fan, Yinhe Han, Zhimin Zhang, "SoC Leakage Power Reduction Algorithm by Input Vector Control" , IEEE international conference, 2005.