Contents list available at JMCS
Journal of Mathematics and Computer Science
Journal Homepage: www.tjmcs.com

Optimization of JK Flip Flop Layout with Minimal Average Power of Consumption based on ACOR, Fuzzy-ACOR, GA, and Fuzzy-GA

Farshid Keivanian[*,1], **Ali Yekta** [*,2], **Nasser Mehrshad** [*,3]

[*] Department of Electrical and Computer Engineering, University of Birjand, Iran,
[1]*FarshidKeivanian@birjand.ac.ir*
[2]*Ali.Yekta@birjand.ac.ir*
[3]*NMehrshad@birjand.ac.ir*

Article history:
Received July 2014
Accepted October 2014
Available online November 2014

Abstract

The object of heuristic algorithms is to produce an optimum solution for solving a problem. When the number of variables in the problem is high the Heuristic Algorithms are used. In this article the goal is to find an optimum layout for JK Flip Flop for minimizing the average power. There are twenty MOSFETs with different channel widths. They make a twenty dimensional search space which are independent decision variables. Motivated by the convergence of Ant Colony Optimization in real domain (ACOR) and Genetic Algorithm (GA) and the link of MATLAB with HSPICE Software the optimized layout of JK Flip Flop is obtained. Based on ACOR, Fuzzy-ACOR, GA, Fuzzy-GA algorithms the best resulting JK Flip Flop layout in CMOS Technology with supply voltage of 5v has the average power consumption of 1.6 nW with Fuzzy-ACOR.

Keywords: Optimum layout, JK Flip Flop, ACOR, Fuzzy-ACOR, GA, Fuzzy-GA.

1. Introduction

Component layout plays an important role in the design and usability of many engineering products [1]. This article describes ACOR, Fuzzy-ACOR, GA, and Fuzzy-GA algorithms to solve the problem of the channel width of MOSFETs in JK Flip Flop such that a set of objectives such as delay and average power can be optimized. In this article the single object of optimization of the average power is proposed. The goal is to minimize the average power in JK Flip Flop layout in CMOS Technology. The issue of energy optimization has been proposed in other applications like wireless sensor networks based on heuristic algorithms [2]. To avoid a large number of calculations for power consumption of the circuit the heuristic algorithm is employed. Furthermore the performance of proposed heuristic algorithms including ACOR, and GA has been tested over problems [3]. Genetic Algorithm (GA) is an Evolutionary Computation algorithm and Ant Colony Optimization in continuous space (ACOR) is a Swarm Intelligence algorithm which both are used to achieve the optimum layout design of JK Flip Flop and all of the results are illustrated and compared. The circuit of JK Flip Flop in gate design level is shown in Figure 1, also in the level of transistor design is pictured in Figure 2, in addition in layout level it is illustrated in Figure 3.

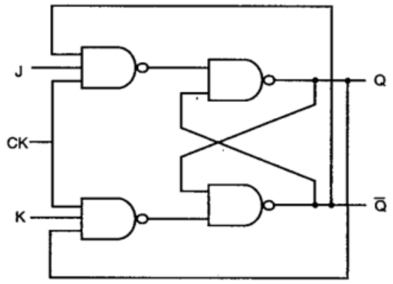


Figure 1. JK flip flop design in gate level

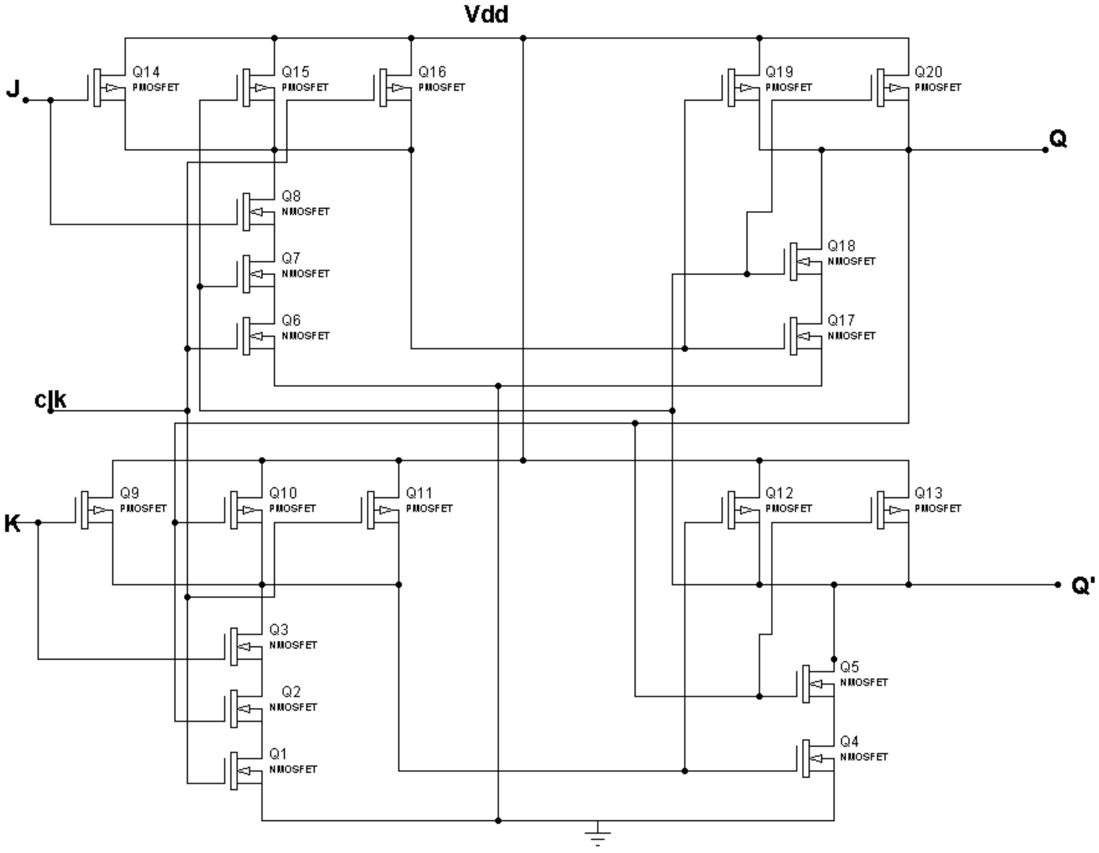


Figure 2. JK flip flop design in transistor level, CMOS technology

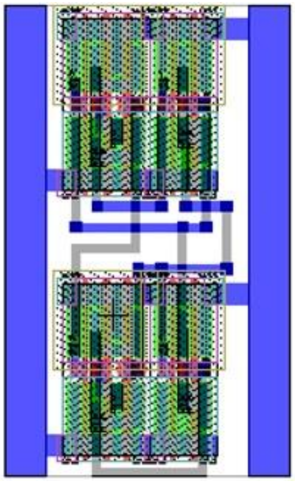


Figure 3. JK flip flop design in layout level, LEDIT layout

2. Heuristic algorithms: ACOR and GA

In the past few decades, there has been widespread interaction between researchers seeking various evolutionary computation methods to seek the best solutions to a given function. The Evolutionary Algorithms are developed by mimicking or simulating processes found in nature and mainly includes Genetic Algorithms, Mimetic Algorithms, Particle Swarm Optimization, Ant Colony Optimization, and Shuffled Frog Leaping Algorithm. Optimization algorithms have constituted most significant subjects in mathematics and industry to conceive more accurate and expeditious solutions.

2.1 Ant colony optimization (ACOR) algorithm in real domain to minimize the average power in JK flip flop

One of the first attempts to apply an ant-related algorithm to the continuous optimization problems was Continuous ACO (CACO) (Belchev and Parmee, 1995) [3]. The extension of ACO to the continuous domain, ACOR which is applied to the continuous optimization problem in this work is the one originally proposed by Schola and further extended by Schola and Dorigo [3]. The proposed ACOR in [3] compliances with ACO in discrete domain as the process of pheromone is available in both of them. In this part the ACOR in [3] is clarified and used for designing optimum JK Flip Flop. The figure 4, diagram 4-1 shows the ACO in discrete domain including several ways that the ants can pass through them and there is no change around each way so there will not be any distribution around each solution of a variable. The more ants are passed or the better ants are moved through a way the more pheromone value is created. Meanwhile for ACOR the variables are continuous, and in this work they are limited to the constraint from the minimum value (“VarMin=0.2µm”) to the maximum value (“VarMax=2µm”) for each channel width.

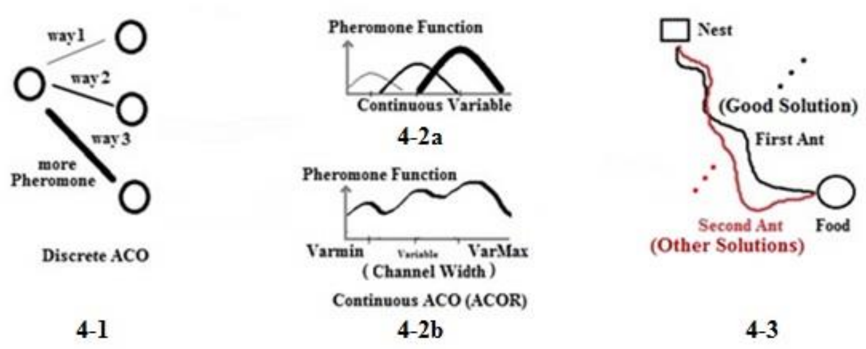


Figure 4. Diagram 4-1 is discrete ACO, diagram is ACO$_R$, and diagram 4-3 is the nature of continuity in ACO$_R$

Based on diagram 4-2a, each channel width or variable has different solutions and there are pheromone functions which are similar to normal distributions and the pheromone values are distributed over the different solutions. In diagram 4-2a, the continuous variable or channel width in horizontal axe has three possible values and normal distributions around them. The better solution or value has the more amount of pheromone around it, in comparison with other solutions. Also as it is seen in diagram 4-2b the pheromone functions have been merged or added together and the resulting pheromone function is more intensified around the better solution for the problem. As it is shown in diagram 4-3, naturally in a group of ants the second ant does not exactly pass through the way of the first ant, so there is a change around the exact way or specific solution which means there is nearly a normal distribution over each solution. And the centers of these normal distributions are the good solutions in comparison with the other solutions. So the archive of good solutions will be made and updated until they become the best solutions for the problem. Based on table 1 the best solution which is found for the channel width of MOSFET Q5 (“W5”) is 0.733024 µm. The nature of ants and their pass ways are shown in figure 5, graph 5-1. The variable W5 stands for the channel width of MOSFET Q5 in figure 5-2. The figure 5-3 shows the good solutions, for the variable of W5. Each of normal distributions as it is seen in graph 5-3 has its own probability of selection P. The probability of selection is based on Roulette Wheel Selection. Also these good solutions have their own cost function. The better solution has the lower cost function value and the higher probability of selection. In this article each fitness value or cost value is determined by the HSPICE Software. In fact HSPICE is the fitness evaluator of this problem since it calculates the average power of JK flip flop circuit in each iteration in ACOR in MATLAB software. Each of distribution is normal distribution with its mean and variance. The mean value is the good solution and the variance is the paths around the good way as it is seen in figure 5-1 and 5-3. The probabilities of selection Pi, P*, and Pj are independent of each other. Based on the principles of the probability, the independent probabilities are applied to the equations of 1 and 2, and based on them the formula of 3 is obtained since the probabilities of selection are multiplied by normal distributions and add together, so the resulting distribution is Gaussian Distribution of P(x_5) which is shown in equation 3. It is used for the total distribution over the random variable x.

Table 1. The best solutions in ACO$_R$ for W_1 to W_{20}

W1	W2	W3	W4	W5	W6	W7	W8	W9	W10
0.2	0.461414	1.257616	0.2	0.733024	0.28182	0.663069	0.2	0.665336	0.2
W11	W12	W13	W14	W15	W16	W17	W18	W19	W20
0.2	2	0.298566	0.549933	2	0.507826	0.2	0.819812	0.915831	0.305826

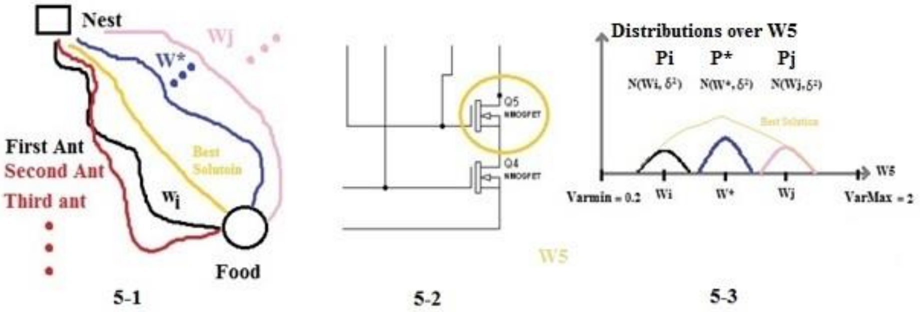


Figure 5. The nature of different solutions in the behavior of ants in 5-1, MOSFET Q5 in 5-2, normal distributions over solutions for the channel width of Q5, W5 in 5-3

$$P(i \cup j) = P(i) + P(j) \tag{1}$$

$$P(i \cap j) = P(i) \times P(j) \tag{2}$$

$$P(x_5) = P(i)N(Wi, \delta_i^{\,2}) + P(*)N(W*, \delta_*^{\,2}) + P(j)N(Wj, \delta_j^{\,2}) \tag{3}$$

$$G^5 \equiv P(x_5) \tag{4}$$

$$N(x;\mu,\delta^2) = \frac{1}{\sqrt{2\pi}}\frac{1}{\delta}.\exp(\ \frac{(x-\mu)^2}{2\delta^2}) \tag{5}$$

The graph 5-3 shows three spread places of pheromones in black, blue, and pink colors that are distributed around the centers Wi, Wj, and W* and the percentage of pheromones are determined by Pi, Pj and P*. By the equation 3 the probabilistic model of $P(x_5)$ for the channel width of W5 (“x5”) is established. The ants will pay more attention to the place or center of W* where P(x) will be highly assigned for it. And the best solution in yellow color in figure 5-3 for W5 is obtained from equation 3. It includes more pheromone than others around W*. P(x) is separately calculated for each of twenty variables (twenty channel widths) in this problem like $P(x_1)\ldots P(x_5)\ldots P(x_{20})$. The alternative names for $P(x_i)$ is G^i such that the equivalent name of G^5 is used for $P(x_5)$ in the next parts and figures. Also the equation 5 shows the normal distribution function for each variable with mean and variance. $P(x_5)$ in equation 3 can produce new different solutions for the random variable of x5 so the random values can be produced for the channel width of Q5 (“W5”) by means of equation 3. Each solution of the solution archive has its own sigma δ. If all sigma in equation 3 are zero then the exact solutions of solution archive will be produced but if all sigma are non-zero with different values then the local search will be happened around the solutions. Whatever the sigma is closer to zero further exploitation is done. When the sigma (δ) becomes great then the exploration becomes high and it increases the diversity of solutions. In this work the problem includes twenty channel widths (“W1…W20”) that have distinct probability distributions (“P(x1) … P(x20)”). So by controlling the parameter of sigma, δ the diversity of channel widths will be increased which will be clarified in part 2.2. Whatever the produced pheromone of a way or variable is more the probability value is more. In figure 5, graph 5-3, the simple solution archive of Wi, W*, and Wj create the probabilistic model of P(x5) which its work is to generate new random solutions or new population for the channel width of Q5 (“W5”). Then the probabilistic model will be sampled and the new population will be produced and they will be compared with the previous population in solution archive. By means of this competition the solution archive will be updated and the worse solutions will be eliminated.

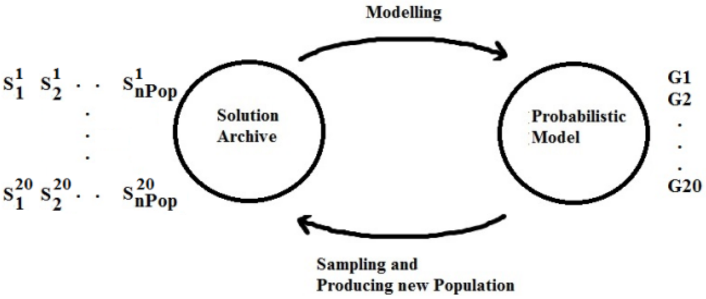
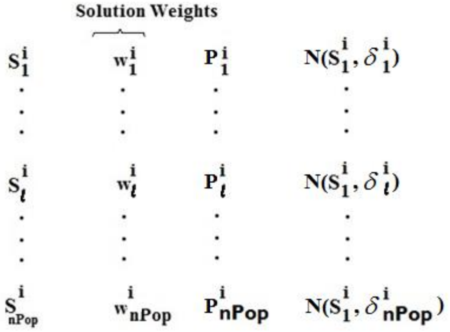

Modelling
S^1_1 S^1_2 . . S^1_{nPop}
S^{20}_1 S^{20}_2 . . S^{20}_{nPop}
Solution Archive
Probabilistic Model
G1
G2
G20
Sampling and Producing new Population

Figure 6. Trade off cycle between solution archive and probabilistic model



	Solution Weights		
S^i_1	w^i_1	P^i_1	$N(S^i_1, \delta^i_1)$
.	.	.	.
.	.	.	.
.	.	.	.
S^i_l	w^i_l	P^i_l	$N(S^i_1, \delta^i_l)$
.	.	.	.
.	.	.	.
.	.	.	.
S^i_{nPop}	w^i_{nPop}	P^i_{nPop}	$N(S^i_1, \delta^i_{nPop})$

Figure 7. The ith solution vector with different solutions from 1 to nPop, related solution weights, related probabilities of selection and normal distributions

The figure 6 shows a trade-off cycle between solution archive and probabilistic model. This cycle cause the population and probabilistic model to be gradually better. The solution archive includes solution vectors. Each solution vector of S^i includes solutions of S^i_1 … S^i_{nPop} and each solution has its own solution weight. The solution weights will be sorted based on the relation 6 and the better solution has more solution weight the figure 7 shows a solution vector with relative solution weight, probability of selection, and normal distribution. Every solution has its own specific distribution. The notation for solution archive size is nPop which is set to 30 in this work and this parameter is similar to the parameter of population size in Genetic Algorithm in part 3.1. As it is shown in the relation 7 the random vector of x is obtained from the Joint Gaussian Mixture of G.

$$w_1 > w_2 > ... > w_l > ... > w_{nPop} \tag{6}$$

$$\bar{x} \sim \overline{G} = (G^1, G^2, ..., G^{20}) \tag{7}$$

The lth weight, wl, is obtained from equation 8. As it is similar to Normal Distribution it is expressed by Normal Distribution, N, in equation 8. The solution archive size is nPop and l stands for lth solution weight and q is a positive parameter which affects on the variance.

$$w_l = \frac{1}{q.nPop.\sqrt{2\pi}}.\exp(\ \ \frac{1}{2}.(\frac{l \ \ 1}{q.nPop})^2) = N(l, \mu, \delta^2 = (q.nPop)^2) \tag{8}$$

The normal distribution of N with the mean μ and the standard deviation of δ is shown in equation 10. The δ is defined q.nPop in this article. The parameter q controls the solution selection, if q is zero then only the best member of solution archive will influence on producing new solutions while if q is high the solutions around the good solutions in solution archive will highly influence on producing new solutions. The role of parameter q in equation 8 is similar to the role of parameter α in Discrete Ant Colony Optimization (ACO) which is shown in equation 9 [4].

$$p_{ij}^k(t) = \begin{cases} \dfrac{[\tau_{ij}(t)]^\alpha [\eta_{ij}]^\beta}{\sum\limits_{k \in allowed_k} [\tau_{ik}(t)]^\alpha [\eta_{ik}]^\beta} & \text{if } j \in \text{allowed}_k \\ 0 & \text{otherwise} \end{cases} \tag{9}$$

The parameter α in $[\tau_{ij}(t)]^\alpha$ is the social intelligence factor in discrete ACO [4] also here in ACO_R the parameter q is the social intelligence , of course 1/q is associated with α. Based on equation 10, the probability, $p^i{}_l$ is the probability of selection of lth solution among the nPop solutions for the random variable of x^i or ith channel width. This probability is associated with the related solution weight of lth solution. The solution weights are normalized and $p^i{}_l$ is obtained. This probability is used in the equation 11 and the resulting Gaussian Distribution, G^i in fact is the probabilistic model of ith channel width which can produce the new other solutions for this variable. Whatever the variance δ^2 in the equation 11 is high, the diversity of solutions becomes high. The variance δ^2 in ACO_R determines the step of search in algorithm and it is called step size.

$$p^i{}_l = \frac{w^i{}_l}{\sum\limits_m w^i{}_m}, m:1...nPop \tag{10}$$

$$G^i = \sum_l P^i{}_l.N(S_l^i, \delta_l^{i^2}), l:1...nPop \tag{11}$$

Based on figure 8 the standard deviation $\delta_4{}^1$ is more than $\delta_2{}^1$, $\delta_4{}^1 > \delta_2{}^1$. Because the average distance for fourth solution of $S_4{}^1$ is more than other solutions, its high standard deviation causes more search space.

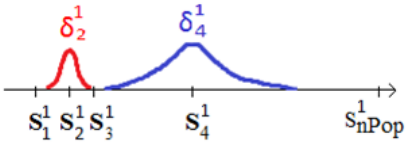


Figure 8. The solutions in solution archive for the first solution vector S_i^1 , i: 1 … nPop

If the search space is limited only to the ith solution vector, then the distance of the solution S_l^i from the other solutions of S_r^i (r : 1 … nPop) is calculated by equation 12. Also the number of other solutions except the solution of S_l^i is "nPop – 1", so the equation 12 shows the average distance of lth solution to the other solutions. When the average distance of the solution from the other solutions is so high, then the variance of the normal distribution around that solution should be greater than others to increase the search capability around it.

$$AverageDis = \frac{\sum_r \left|S_l^i - S_r^i\right|}{nPop - 1}, r : 1 ... nPop, r \neq l \tag{12}$$

As it is shown in equation 13 the standard deviation is proportional to the average distance. The parameter zeta ζ is a positive control parameter which will be controlled in fuzzy-ACO$_R$ in part 2.2. If lth solution is far away than the other solutions then the amount of Sigma Σ and subsequently the amount of standard deviation δ becomes high and it causes the coarse search step while if zeta is low then the search step becomes fine.

$$\delta_l^i = \zeta \frac{\sum_r \left|S_l^i - S_r^i\right|}{nPop - 1}, r : 1 ... nPop, r \neq l \tag{13}$$

The equation 13 determines the standard deviation around each solution which is shown in figure 8 and it is used in the power of two for the variance in the Normal Distribution in equation 11. The least average power of JK Flip Flop is converged to 8.2 nW which is shown in figure 9. The Pseudo Code of ACO$_R$ is shown in box 1.

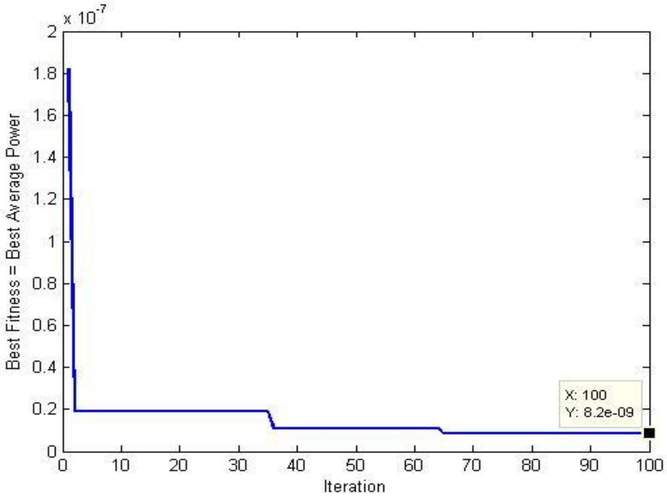


Figure 9. Convergence of average power to 8.2nw based on ACOR

Box 1. ACO$_R$ Pseudo-Code in this article

```
Define the problem of JK Flip Flop
Set the Solution Archive Size to nPop, the Sample Size to nSample, Selection Pressure to q
Set the deviation distance ratio to zeta
Generate Initial Solutions, Evaluate, Sort, and place them in the Solution Archive
Calculate the Solution Weights $W_l$ and the Selection Probabilities $P_l$
  Do While termination conditions do not meet
    Create probabilistic model $G_i$ for each variable separately,
    Compute random samples based on probabilistic model and create new population (newpop)
    Merge New Population (Samples) and Main Population (Archive), Sort, and Delete Extra Members
    Update the best solution ever found and Store Best Cost
  End Do
Plot Results
```

2.2 Fuzzy ant colony optimization in real domain (Fuzzy-ACOR) for the optimization of JK flip flop

As noted previously by controlling the parameter of zeta ζ the speed of convergence and the performance of ACO$_R$ will be improved. When ζ is high the exploration in search process becomes high while when ζ is low the exploitation becomes high also the search step will be fine or small and the search space is more effectively searched. At the beginning of search process when the cost function is high, zeta must be high. Also the size of new population (Sample Size), nSample in this article is another parameter that is controlled and by increasing it the diversity of search will be increased. The figure 10 shows the block diagram of fuzzy-ACO$_R$ which is used in this work. Based on the figure 10 both HSPICE and MATLAB software run together simultaneously. In each iteration the software of MATLAB will produce layout of JK flip flop, and HSPICE will calculate its average power until the best average power is obtained. The Pseudo Code for Fuzzy-ACO$_R$ is illustrated in Box 2.

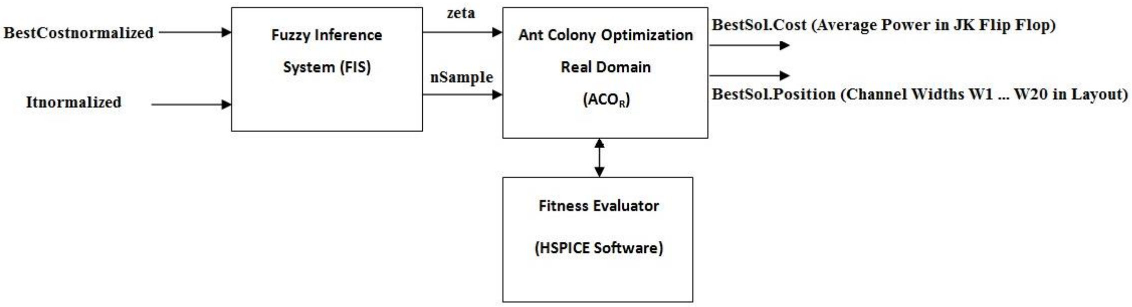

BestCostnormalized
Itnormalized
Fuzzy Inference System (FIS)
zeta
nSample
Ant Colony Optimization Real Domain (ACO$_R$)
BestSol.Cost (Average Power in JK Flip Flop)
BestSol.Position (Channel Widths W1 ... W20 in Layout)
Fitness Evaluator (HSPICE Software)

Figure 10. The block diagram of Fuzzy-ACOR system which is defined in this article

Box 2. Fuzzy-ACO$_R$ Pseudo-Code in this article

```
Define the problem of JK Flip Flop
Set the Archive Size to nPop, the Sample Size to nSample, Selection Pressure to q
Set the deviation distance ratio to zeta
Generate Initial Solutions, Evaluate, Sort, and place them in the Solution Archive
Calculate the Solution Weights Wl and the Selection Probabilities Pl
  Do While termination conditions do not meet
    Create probabilistic model Gi for each variable separately,
    Compute random samples based on probabilistic model and create new population (newpop)
    Merge New Population (Samples) and Main Population (Archive), Sort, and Delete Extra Members
    Update the best solution and the worst solution that ever found and Store Best Cost and Worst Cost
    Normalize itnormalized = it / Maxit, and Bestnormalized = [ WorstCost - BestCost(it) ] / WorstCost
    Read Fuzzy Inference System File and Fuzzy Rules (Fuzzy_ACOR_FIS.fis)
    Define Input Variables for FIS and Fire the Rules
  End Do
Plot Results
```

Table 2 shows the Fuzzy-ACO$_R$ rules. The optimum width channels are shown in table 3. The figure 11 shows the least average power of 1.6 nW.

Table 2. The fuzzy rules in Fuzzy-ACOR

If	Then
Itnormalized is High and BestCostnormalized is High	zeta is High and nSample is High
Itnormalized is High and BestCostnormalized is Low	zeta is Low and nSample is Low
Itnormalized is Low and BestCostnormalized is High	zeta is High and nSample is High
Itnormalized is Medium and BestCostnormalized is Medium	zeta is Medium and nSample is Medium

Table 3. The best solutions in Fuzzy-ACOR for $W_1 \ldots W_{20}$

W1	W2	W3	W4	W5	W6	W7	W8	W9	W10
0.2	2	0.263429	0.2	0.2	0.50878	1.295758	0.739012	1.509186	0.2
W11	W12	W13	W14	W15	W16	W17	W18	W19	W20
0.995304	1.243184	0.291615	0.73219	2	1.379721	2	0.2	1.392469	0.2

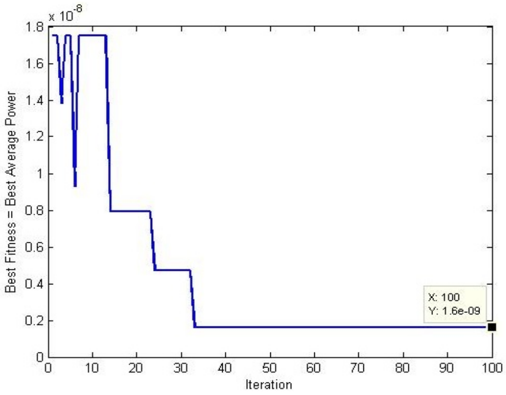


Figure 11. The convergence of the best average power to 1.6nw for JK flip flop based on Fuzzy-ACOR

The membership functions are in trapmf type as it is shown in the Figure 12.

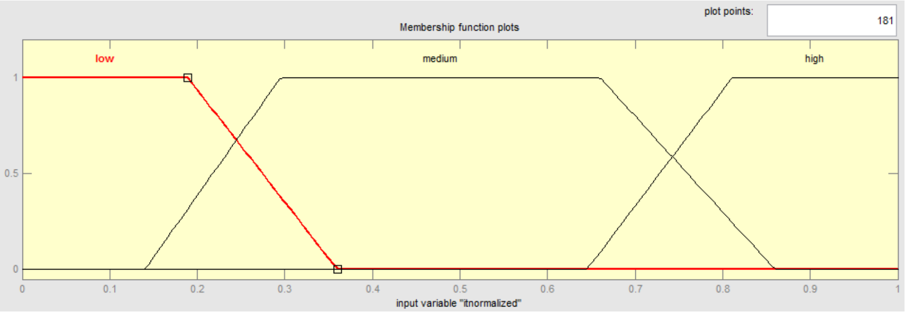


Figure 12. Membership functions for normalized iteration, best cost normalized, zeta, and nsample

2.3 Genetic algorithm to minimize the average power in proposed JK flip flop

Genetic Algorithm as another heuristic algorithm is used, the Pseudo Code for GA for this article is in Box 3, the optimum layout obtained by GA is shown in table 4, and the resulting convergence is shown in Figure 13.

Box 3. GA Pseudo-Code in this article

```
Define the problem of JK Flip Flop and Upper band and Lower band of Variables
Determine GA Parameters (Population Size, Crossover Percentage, and Mutation Percentage)
Set the crossover percentage to pc and mutation percentage to pm
Generate Initial Population, Evaluate, Sort, and place them in the Best Solutions
  Do While termination conditions do not meet
    Calculate Selection Probabilities
    Select Parents indices and apply crossover and evaluate off springs, in Crossover Operator
    Select Parents indices and apply mutation and evaluate Mutant, in Mutation Operator
    Create Merged Population, sort population
    Update and store the best solution and Cost that ever found
  End Do
Plot Results
```

Table 4. The best solutions in GA for W_1 to W_{20}

W1	W2	W3	W4	W5	W6	W7	W8	W9	W10
1.167544	0.333518	1.493428	1.411993	0.922975	0.273223	1.759053	1.604558	1.335302	0.986943
W11	W12	W13	W14	W15	W16	W17	W18	W19	W20
1.668262	1.23976	1.364574	1.140913	0.663053	0.738244	0.595428	1.470576	1.383409	0.857879

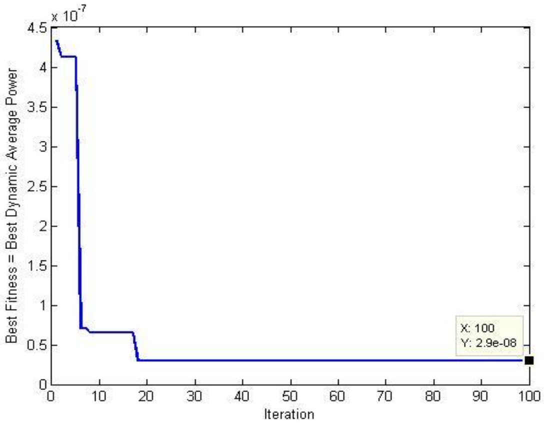


Figure 13. The convergence of the best average power to 1.6nw for JK flip flop based on Fuzzy-ACOR

2.4 Fuzzy genetic algorithm (Fuzzy-GA) for optimization of JK flip flop layout

The block diagram of Fuzzy-GA is shown in Figure 14 [5]. The parameters of mutation percentage (“pm”), crossover percentage (“pc”), and the number of population (“nPop”) are optimally controlled to get the better results. The fuzzy-GA rules are illustrated in table 5. The pseudo code of Fuzzy-GA is shown in Box 4 [6]. The optimum channel widths are shown in table 6. Also the least average power after 100 iterations based on Fuzzy-GA is shown in Figure 15. The membership functions are also in trapmf type, the same as Fuzzy-ACOR [7].

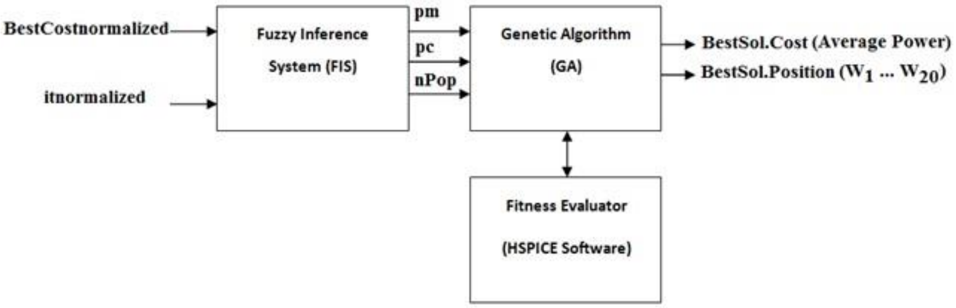

BestCostnormalized
itnormalized
Fuzzy Inference System (FIS)
pm
pc
nPop
Genetic Algorithm (GA)
BestSol.Cost (Average Power)
BestSol.Position (W_1 ... W_{20})
Fitness Evaluator (HSPICE Software)

Figure 14. Block diagram of Fuzzy-GA system which is defined in this article

Table 5. Fuzzy rules in Fuzzy-GA

If	Then
Itnormalized is High and BestCostnormalized is High	pm is High and pc is Low and nPop is High
Itnormalized is High and BestCostnormalized is Low	pm is Low and pc is High and nPop is Low
Itnormalized is Low and BestCostnormalized is High	pm is High and pc is Medium and nPop is High
Itnormalized is Medium and BestCostnormalized is Medium	pm is Medium and pc is Medium and nPop is Medium

Box 4. Fuzzy-GA Pseudo-Code

```
Define the problem of JK Flip Flop and Upper band and Lower band of Variables
Determine GA Parameters (Population Size, Crossover Percentage, and Mutation Percentage
Set the crossover percentage to pc and mutation percentage to pm
Generate Initial Population, Evaluate, Sort, and place them in the Best Solutions
  Do While termination conditions do not meet
    Calculate Selection Probabilities
    Select Parents indices and apply crossover and evaluate off springs, in Crossover Operator
    Select Parents indices and apply mutation and evaluate Mutant, in Mutation Operator
    Create Merged Population, sort population
    Update the best solution and the worst solution that ever found and Store Best Cost and Worst Cost
    Normalize itnormalized = it / Maxit, and Bestnormalized = [ WorstCost - BestCost(it) ] / WorstCost
    Read Fuzzy Inference System File and Fuzzy Rules (Fuzzy_ACOR_FIS.fis)
    Define Input Variables for FIS and Fire the Rules
  End Do
Plot Results
```

Table 6. The best solutions in Fuzzy-GA for W_1 to W_{20}

W1	W2	W3	W4	W5	W6	W7	W8	W9	W10
0.746587	0.781259	0.486344	0.793128	0.680617	0.623516	0.440017	0.896774	1.258428	1.235993
W11	W12	W13	W14	W15	W16	W17	W18	W19	W20
1.719993	1.247373	0.97647	1.682751	1.44475	1.736418	0.633273	0.800001	2	0.733739

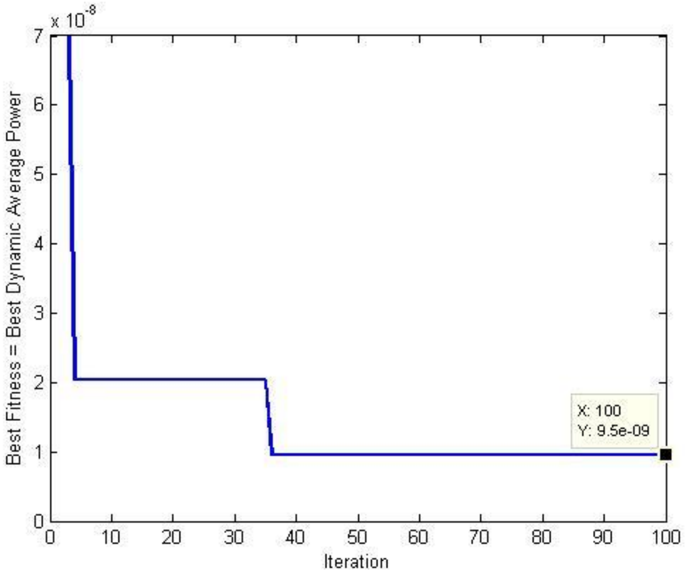


Figure 15. Convergence of the best average power to 9.5nw for JK flip flop based on Fuzzy-GA

3. Conclusion

By implementation of heuristic algorithms such as ACOR, Fuzzy-ACOR, GA, Fuzzy-GA the optimum JK flip flop is obtained. The best layout consumes only 1.6nW based on Fuzzy-ACOR. The comparisons are illustrated in table 7 and figure 16. Also the fuzzy-GA and fuzzy-ACOR showed the better results since the fuzzy rules has increased the performance of algorithm and finally the most optimum JK Flip Flop with the least average power consumption is obtained by Fuzzy-ACOR.

Table 7. The least average power consumption are compared

Heuristic Algorithms	The best fitness value or lowest average power which is obtained
ACO_R	8.2 nW
Fuzzy-ACO_R	1.6 nW
GA	29 nW
Fuzzy-GA	9.5 nW

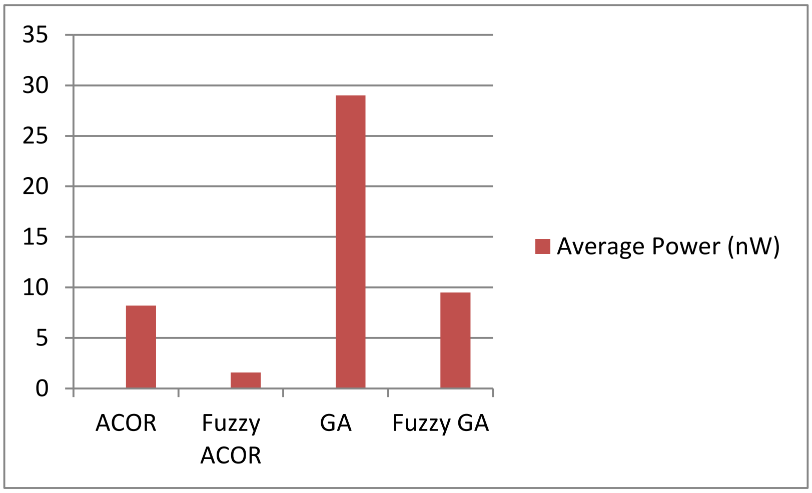


Figure 16. The comparison of convergences of different heuristic algorithms: ACOR, Fuzzy-ACOR, GA, and Fuzzy-GA

References

[1] I. Mihajlovic , Z. Zivkovic, N. Strbac, D. Zivkovic, A. Jovanovic, "*Using Genetic Algorithms To Resolve Facility Layout Problem*", Serbian Journal of Management 2 (1) (2007) 35-46.

[2] M. Eslami, J. Vahidi, M. Askarzadeh, "*Designing and Implementing a Distributed Genetic Algorithm for Optimizing Work Modes in Wireless Sensor Network*", Journal of Mathematics and Computer Science 11 (2014) 291-299.

[3] K. Socha, M. Dorigo, "*Ant colony optimization for continuous domains*", European Journal of Operational Seacrh 185 (2008), pp. 1155-1173.

[4] Dorigo M. and G. Di Caro, "*The Ant Colony Optimization Meta-Heuristic*" In D. Corne, M. Dorigo and F. Glover, editors, New Ideas in Optimization, McGraw-Hill, 11-32 , (1999).

[5] Keivanian, F., Mehrshad, N., Zahiri, S. H. "*Optimum Layout of Multiplexer with Minimal Average Power based on IWO, Fuzzy-IWO, GA, and Fuzzy GA*", ACSIJ Advances in Computer Science : an International Journal, Vol. 3 (2014), Issue 5, No. 11.

[6] S. M. Kalami Heris, PhD of Control Engineering, K. N. Toosi University of Technology, Tutorial website of *www.matlabsite.com.*

[7] Mitchell, Melanie, "*An Introduction to Genetic Algorithms*", Cambridge, MA: MIT Press. ISBN 9780585030944, (1996).