Comparison of Ant Colony Optimization, Genetic Optimization, and Hamming Distance Algorithms for Vector Reduction

Kumar K K[1][a], Suhas Shirol[1], Kruthik Gupta B[1][b], Ashwini M Pattennavar[1], Spoorti Patil[1], Saroja V S[2], Vijay H M[2] and Rajeshwari M[2]

[1]*KLE Tech, KLE Technological University, Vidyanagar, Hubli, India*

[2]*Electronics and Communication, KLE Technological University, Hubli, India*

Keywords: Ant Colony Algorithm, Genetic Algorithm, Hamming Distance Algorithm, Test Vectors, K-Means, Verilog HDL Implementation, Minimum Power Leakage.

Abstract: In this paper, we present the Ant colony algorithm, Genetic algorithm, and Hamming Distance algorithm for vector reduction and reduced power leakage using algorithms. The main motivation is to compare Ant Colony Optimization (ACO), Genetic Optimization (GO), and Hamming Distance Algorithms for vector reduction to identify the most effective method for optimizing vector representation in various computational applications. This step aims to reduce the number of vectors and reduce power leakage. This paper proposes a new approach to the Ant colony algorithm based on switching activity, Hamming distance based on K-means clustering, and a Genetic algorithm based on fitness for vector reduction. In the comparative analysis of power leakage between algorithms, the Ant Colony algorithm exhibited the lowest power leakage of 0.17%. Therefore the Ant Colony algorithm is the most efficient in terms of power consumption

1 INTRODUCTION

The rapid advancement of technology into the nanometer scale has led to a significant increase in sub-threshold leakage currents, which rise exponentially as the supply voltage (Vdd) and threshold voltage (Vth) decrease. In contemporary CMOS technologies, sub-threshold leakage current has become the dominant component of total leakage current. Minimizing leakage power is especially crucial for portable devices, which often operate in standby mode, to extend battery life. Among various techniques for reducing leakage power, Input Vector Control (IVC) stands out due to its independence from process technology parameters and its reliance on the transistor stacking effect. IVC positions a circuit in its minimum leakage state without compromising performance.

Genetic Optimization, rooted in natural selection and genetics principles, is an evolutionary algorithm that iterates through generations of candidate solutions to find the best fit. This process involves selection, crossover, and mutation operations to evolve a population of potential solutions over successive generations. GO is particularly effective in global search optimization problems and can adaptively discover optimal subsets of vectors, making it a powerful tool for reducing the dimensionality of large datasets while preserving essential information.

Ant Colony Optimization, inspired by the foraging behavior of real ants, is a probabilistic technique primarily used for solving combinatorial optimization problems. Introduced by Marco Dorigo in the early 1990s, ACO leverages a population of artificial ants that iteratively construct solutions and exchange information via pheromone trails. These pheromones influence the likelihood of future ants selecting specific paths, guiding the algorithm toward optimal or near-optimal solutions. In vector

[a] https://orcid.org/0009-0008-3924-2775
[b] https://orcid.org/0009-0003-2257-3723

K K, K., Shirol, S., Gupta B, K., M Pattennavar, A., Patil, S., V S, S., H M, V. and M, R.
Comparison of Ant Colony Optimization, Genetic Optimization, and Hamming Distance Algorithms for Vector Reduction.
DOI: 10.5220/0013582500004664

In *Proceedings of the 3rd International Conference on Futuristic Technology (INCOFT 2025) - Volume 1*, pages 608-614
ISBN: 978-989-758-763-4

reduction, ACO can efficiently identify relevant components by exploring and exploiting the search space through cooperative behavior.

Hamming Distance, a dissimilarity measure between two strings of equal length, has been widely used in error detection and correction codes. When applied to vector reduction, algorithms based on Hamming Distance focus on identifying and removing redundant or less significant vectors by comparing binary data representations. This approach is particularly advantageous in scenarios where binary or categorical data needs to be simplified without losing critical information.

2 LITERATURE SURVEY

Several researchers have employed genetic algorithms (GAs) to find the MLV. Chen et al. introduced a genetic algorithm that uses an accurate leakage current model to search for the MLV. This approach demonstrated significant improvements in leakage power reduction compared to traditional methods. Xiaoying Zhao et al. enhanced this approach by using Circuit Status Difference (CSD) as the fitness function, which addressed some limitations of the previous technique. Their method showed that GAs could efficiently identify the MLV with fewer iterations and reduced runtime compared to random search methods. Implementing genetic algorithms in hardware description languages (HDLs) like Verilog has shown promising results regarding convergence speed and computational efficiency. The HDL-based approach allows for the simulation and synthesis of test circuits, making enforcing the MLV during standby mode easier to reduce leakage power. (Thong, and, Champrasert, 2023), (Sun, 2023),(Zhou and Gao, 2024), (Zhang, Li, et al. , 2023), (Pham, Nguyen, et al. , 2023).

The research paper explores Ant Colony Optimization (ACO) and its application to the Traveling Salesman Problem (TSP), inspired by the foraging behavior of real ants to find the shortest path. By simulating the collective intelligence of ant colonies, ACO efficiently solves complex optimization problems. The study focuses on constructing graphs for TSP, associating components with either edges or vertices and provides examples with four cities to demonstrate ACO's effectiveness. ACO algorithms have proven adaptable and efficient in finding near-optimal solutions across various optimization scenarios. (Awad, Hawash, et al., 2023), (Jayakumar and Khatri, 2007), (Lin, Qu, et al. , 2006), (Chen, , et al. , 1998), (Leelarani, Madhavilatha, et al., 2015)

The research paper presents a clustering algorithm tailored for categorical data, utilizing the Hamming distance as its metric. It transforms data into binary form before clustering, demonstrating promising performance in experiments on UCI machine learning repository datasets. The results suggest the algorithm's potential effectiveness across different data types. The authors developed a clustering algorithm for categorical data using the Hamming distance metric. They transformed the data into binary form to apply the Hamming distance effectively. Performance evaluations on UCI machine learning repository datasets showed the algorithm's effectiveness, surpassing existing methods in clustering categorical data. (Lan, 2023), (Starzec, Starzec, et al. , 2024), (Dorigo, Birattari, et al. , 2006), (Raman, Kumar, et al. , 2024)

3 DESIGN METHODOLOGY

3.1 Flowchart of Ant colony Algorithm

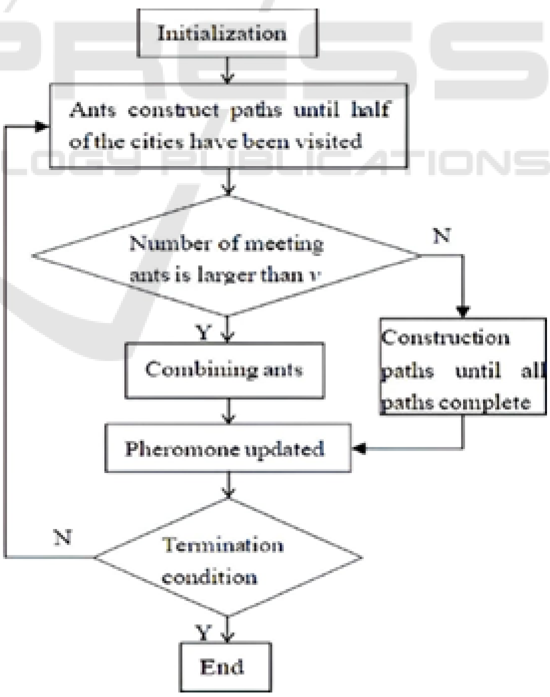

Figure 1: Flowchart of Ant colony Algorithm

The flowchart in Figure 1. illustrates the procedural steps involved in the Ant Colony Optimization (ACO) algorithm, tailored for a specific problem involving path construction. Initially, the process begins with an initialization phase where parameters such as pheromone levels are set, and a population of artificial ants is created. Each ant then constructs a path, visiting nodes or cities until they have covered half of them. The algorithm then checks if the number of ants encountering each other at nodes exceeds a predefined threshold (ν). If this condition is met, the paths of these ants are combined, which likely involves integrating their solutions to enhance the overall quality. If the threshold is not reached, the ants continue constructing their paths until all paths are complete. Once all paths are constructed, the pheromone levels are updated based on the quality of the constructed paths, reinforcing successful routes. This iterative process continues until a termination condition is met, such as a maximum number of iterations for convergence to a satisfactory solution, at which point the algorithm ends

3.2 Flowchart of Hamming Distance Algorithm

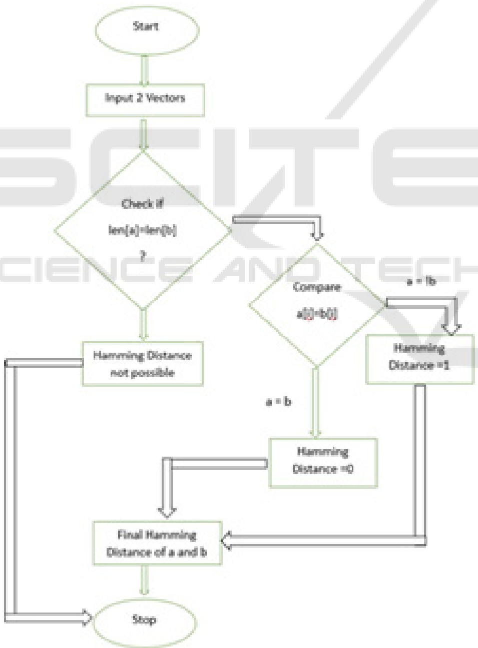

Figure 2: Flowchart of Hamming Distance Algorithm

Figure 2. shows that the Hamming Distance starts by taking two input vectors and checking if they have the same length; if not, calculating the Hamming distance is not possible. If the lengths are equal, each corresponding element of the vectors is compared. If the elements are the same, the distance for that position is 0; if it is different, the distance is 1. This process is repeated for all elements, and the final Hamming distance is the sum of these individual distances.

3.3 Flowchart of Genetic Algorithm

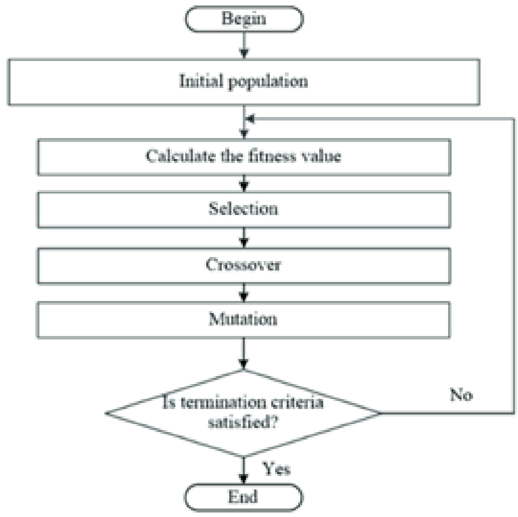

Figure 3: Flowchart of Genetic Algorithm

Figure 3. shows the genetic algorithm starts by generating a random initial set of solutions called chromosomes. Each chromosome's fitness is assessed to determine how well it solves the problem. Two chromosomes are then selected based on their fitness, using methods like a roulette wheel or tournament selection. These selected chromosomes undergo crossover to produce offspring, combining parts of the parents' chromosomes. A mutation is applied to the offspring to introduce random changes and maintain genetic diversity. The algorithm checks if the stopping criterion is met; if not, it repeats the evaluation and selection process with the new population. Once the stopping criterion is met, the algorithm outputs the best solution found and terminates.

3.4 Final Design

The functional block diagram outlines a systematic approach to optimize circuit design and reduce power consumption is shown in Figure 4. It commences with Test Vectors, initial input data intended for circuit testing, which are processed initially in MATLAB to prepare them for optimization. Following this preparation, three distinct algorithms are applied. The outputs from these algorithms are the Reduced Vectors, subsequently integrated into the design flow using the Cadence tool Within Cadence, Verilog (HDL) code is generated from the Reduced Vectors to instantiate and simulate the OR Gate Circuit, where the actual hardware design and implementation occur.

Power measurement assesses the power consumption of the implemented circuit, followed by power analysis, which compares power metrics before and after algorithmic optimizations. The resulting Power Reduction quantifies the efficiency gains achieved through algorithms compared to the initial Power without Algorithm. Ultimately, Power with Algorithms signifies the reduced power consumption realized through systematic optimization techniques in Verilog-based circuit design using MATLAB and the Cadence tool.

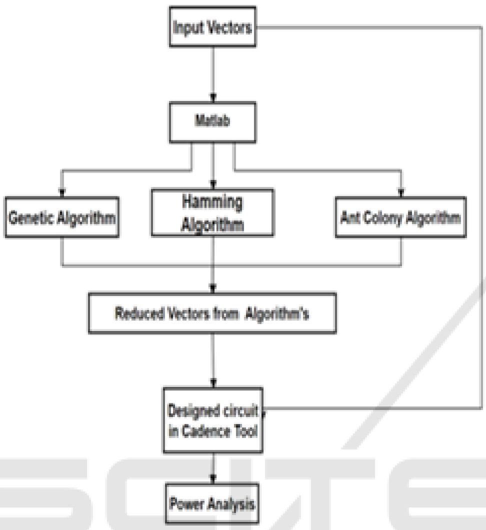

Figure 4: Circuit Design for Optimizing Power Efficiency

4 RESULTS AND ANALYSIS

4.1 Results obtained in MATLAB

4.1.1 Ant Colony Algorithm

This Algorithm uses an ant colony algorithm to reduce vectors based on the switching activity between the test vectors. This algorithm ensures that the sequence provided by the algorithm is more efficient for testing the vectors, as shown in Figure 5

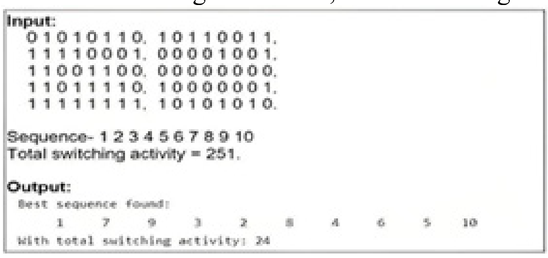

Figure 5: Results of Ant colony based on switching

4.1.2 Hamming Distance Algorithm

The k-means algorithm begins by randomly selecting four initial centroids from the dataset. Each vector is assigned to the nearest centroid based on the Hamming distance, forming clusters. The centroids are updated by calculating the mean for each bit position of the vectors within each cluster. These steps are repeated until the centroids stabilize, indicating convergence, and after reduction, the reduced vectors are given as input to any circuit as a testbench, as shown in Figures 6 and 7

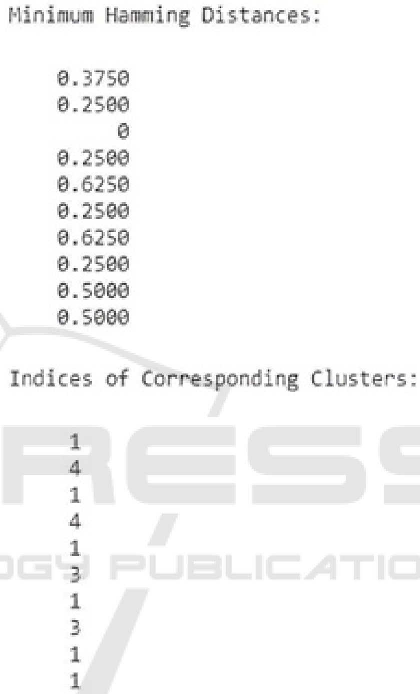

Figure 6: Results of Indices Corresponding Clusters and Minimum Hamming Distance

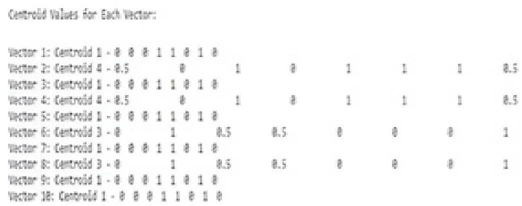

Figure 7: The Centroid Values of each vector

4.1.3 Genetic Algorithm

A genetic algorithm is a computational optimization technique that mimics the process of natural selection to evolve a population of potential solutions to a problem, iteratively improving them through selection, crossover, and mutation operations. In the Genetic algorithm, we are using tournament selection to reduce the size of vectors in the genetic algorithm.

```
Best Solution:
      0   0.0419        0   0.1247   0.3937   0.1180   0.1139   0.2330

Initial Population:
    0    1    0    1    0    1    1    0
    1    0    1    1    0    0    1    1
    1    1    1    1    0    0    0    1
    0    0    0    0    1    0    0    1
    1    1    0    0    1    1    0    0
    0    0    0    0    0    0    0    0
    1    1    0    1    1    1    1    0
    1    0    0    0    0    0    0    1
    1    1    1    1    1    1    1    1
    1    0    1    0    1    0    1    0

Reduced Population:
      0   0.0419        0   0.7626   0.4245   0.1969   0.1139   0.1241
      0   0.0419   0.3601   0.1247   0.4245   0.1969   0.1139   0.1241
      0   0.0419        0   0.1247   0.4245   0.1969   0.1139   0.2330
 0.2131   0.0419        0   0.1247   0.4245   0.1969   0.3889   0.2330
      0   0.0419        0   0.3030   0.4245   0.7742   0.1139   0.1185
      0   0.0419        0   0.1247   0.4245   0.5368   0.1139   0.1185
 0.8567   0.0419        0   0.1247   0.8786   0.1969   0.1139   0.1241
      0   0.0419        0   0.1247   0.4245   0.1969   0.1139   0.1241
      0   0.0419        0   0.1247   0.3937   0.1180   0.1139   0.2330
      0   0.0419   0.8521   0.1247   0.4245   0.1969   0.1139   0.1185
```

Figure 8: Results of reduced vectors of Genetic algorithm

4.2 Results Analysis pf power

An analysis of power leakage reduction using three different algorithms those are Ant Colony, Genetic, and Hamming Distance is shown in Table 1. When no algorithm is applied, the power leakage is 0.20 % across the board. However, applying the Ant Colony algorithm reduces power leakage to 0.17 %, showing the most significant improvement. The Genetic and Hamming Distance algorithms both reduce power leakage to 0.19 %, which, while better than the baseline, are not as effective as the Ant Colony algorithm. Overall, all three algorithms help decrease power leakage compared to not using any algorithm

Table 1: Comparison of power leakage between algorithms in percentage

SL.NO	ALGORITHMS	Power leakage (with Algorithm) (%)	Power leakage (without Algorithm) (%)
	ANT COLONY	0.17	0.20
	GENETIC	0.19	0.20
	HAMMING DISTANCE	0.19	0.20

4.2.1 Power with Normal Test Vectors

```
Instance: /ORGate8
Power Unit: W
PDB Frames: /stim#0/frame#0
  -----------------------------------------------------------------
    Category      Leakage     Internal    Switching       Total   Row%
  -----------------------------------------------------------------
      memory  0.00000e+00  0.00000e+00  0.00000e+00  0.00000e+00   0.00%
    register  0.00000e+00  0.00000e+00  0.00000e+00  0.00000e+00   0.00%
       latch  0.00000e+00  0.00000e+00  0.00000e+00  0.00000e+00   0.00%
       logic  5.76257e-09  1.83968e-06  1.04976e-06  2.89521e-06 100.00%
        bbox  0.00000e+00  0.00000e+00  0.00000e+00  0.00000e+00   0.00%
       clock  0.00000e+00  0.00000e+00  0.00000e+00  0.00000e+00   0.00%
         pad  0.00000e+00  0.00000e+00  0.00000e+00  0.00000e+00   0.00%
          pm  0.00000e+00  0.00000e+00  0.00000e+00  0.00000e+00   0.00%
  -----------------------------------------------------------------
    Subtotal  5.76257e-09  1.83968e-06  1.04976e-06  2.89521e-06 100.00%
  Percentage        0.20%       63.54%       36.26%      100.00% 100.00%
  -----------------------------------------------------------------
```

Figure 9: Power with Normal Test Vectors

4.2.2 Power with Reduced Test Vectors of ant colony algorithm

```
nstance: /ORGate8redant
ower Unit: W
DB Frames: /stim#0/frame#0
  -----------------------------------------------------------------
    Category      Leakage     Internal    Switching       Total   Row%
  -----------------------------------------------------------------
      memory  0.00000e+00  0.00000e+00  0.00000e+00  0.00000e+00   0.00%
    register  0.00000e+00  0.00000e+00  0.00000e+00  0.00000e+00   0.00%
       latch  0.00000e+00  0.00000e+00  0.00000e+00  0.00000e+00   0.00%
       logic  5.76257e-09  1.83968e-06  1.04976e-06  2.89521e-06 100.00%
        bbox  0.00000e+00  0.00000e+00  0.00000e+00  0.00000e+00   0.00%
       clock  0.00000e+00  0.00000e+00  0.00000e+00  0.00000e+00   0.00%
         pad  0.00000e+00  0.00000e+00  0.00000e+00  0.00000e+00   0.00%
          pm  0.00000e+00  0.00000e+00  0.00000e+00  0.00000e+00   0.00%
  -----------------------------------------------------------------
    Subtotal  5.76237e-09  1.83668e-06  1.06976e-06  2.89521e-06 100.00%
  Percentage        0.17%       61.44%       38.39%      100.0% 100.00%
  -----------------------------------------------------------------
```

Figure 10: Power with reduced test vectors of ant colony algorithm

4.2.3 Power with Reduced Test Vectors of Hamming Distance Algorithm

```
Instance: /ORGate8redham
Power Unit: W
PDB Frames: /stim#0/frame#0
  -----------------------------------------------------------------
    Category      Leakage     Internal    Switching       Total   Row%
  -----------------------------------------------------------------
      memory  0.00000e+00  0.00000e+00  0.00000e+00  0.00000e+00   0.00%
    register  0.00000e+00  0.00000e+00  0.00000e+00  0.00000e+00   0.00%
       latch  0.00000e+00  0.00000e+00  0.00000e+00  0.00000e+00   0.00%
       logic  5.76267e-09  1.83988e-06  1.04974e-06  2.89521e-06 100.00%
        bbox  0.00000e+00  0.00000e+00  0.00000e+00  0.00000e+00   0.00%
       clock  0.00000e+00  0.00000e+00  0.00000e+00  0.00000e+00   0.00%
         pad  0.00000e+00  0.00000e+00  0.00000e+00  0.00000e+00   0.00%
          pm  0.00000e+00  0.00000e+00  0.00000e+00  0.00000e+00   0.00%
  -----------------------------------------------------------------
    Subtotal  5.86237e-09  1.84868e-06  1.06976e-06  2.89521e-06 100.00%
  Percentage        0.19%       61.64%       38.17%      100.0% 100.00%
  -----------------------------------------------------------------
```

Figure 11: Power Analysis for the reduced vectors of Hamming Distance Algorithm

4.2.4 Power with Reduced Test Vectors of Hamming Distance Algorithm

Instance: /ORGate8_genetic
Power Unit: W
PDB Frames: /stim#0/frame#0

Category	Leakage	Internal	Switching	Total	Row%
memory	0.00000e+00	0.00000e+00	0.00000e+00	0.00000e+00	0.00%
register	0.00000e+00	0.00000e+00	0.00000e+00	0.00000e+00	0.00%
latch	0.00000e+00	0.00000e+00	0.00000e+00	0.00000e+00	0.00%
logic	5.76266e-09	1.83980e-06	1.04981e-06	2.89521e-06	100.00%
bbox	0.00000e+00	0.00000e+00	0.00000e+00	0.00000e+00	0.00%
clock	0.00000e+00	0.00000e+00	0.00000e+00	0.00000e+00	0.00%
pad	0.00000e+00	0.00000e+00	0.00000e+00	0.00000e+00	0.00%
pm	0.00000e+00	0.00000e+00	0.00000e+00	0.00000e+00	0.00%
Subtotal	5.86237e-09	1.84868e-06	1.06976e-06	2.89521e-06	100.00%
Percentage	0.19%	61.60%	38.21%	100.0%	100.00%

Figure 12: Power Analysis for the reduced vectors of Genetic Algorithm

Figure 9,10,11 and 12 shows the detailed power analysis for all the algorithms that were implemented

5 CONCLUSION

A comparative analysis of power leakage between algorithms is conducted, and the Ant Colony Optimization (ACO) algorithm exhibits the lowest power leakage among other algorithms at 0.17 %. This significant finding suggests that the Ant Colony algorithm is the most efficient in power consumption, making it an attractive option for applications where energy efficiency is paramount. Furthermore, the results demonstrate that using these algorithms can lead to a notable reduction in power leakage, with the power leakage without using the algorithm consistently higher at 0.20 % across all three algorithms. Therefore, the Ant Colony algorithm is the top choice for minimizing power leakage, closely followed by the Genetic and Hamming Distance algorithms.

REFERENCES

S. Thong-ia and P. Champrasert, "Gene-Ants: Ant Colony Optimization with Genetic Algorithm for Traveling Salesman Problem Solving," *2023 International Technical Conference on Circuits/Systems, Computers, and Communications (ITC-CSCC)*, Jeju, Korea, Republic of, 2023, pp. 1-5, doi: 10.1109/ITC-CSCC58803.2023.10212945.

L. Sun, "Path Planning of Mobile Robot Based on Improved Ant Colony Algorithm," *2023 IEEE 11th Joint International Information Technology and Artificial Intelligence Conference (ITAIC)*, Chongqing, China, 2023, pp. 985-989, doi: 10.1109/ITAIC58329.2023.10409046.

Y. Zhou and Z. Gao, "Simulation of Industrial Internet Reliability Assessment Model Based on Ant Colony Optimization Algorithm and Machine Learning," *2024 Asia-Pacific Conference on Software Engineering, Social Network Analysis and Intelligent Computing (SSAIC)*, New Delhi, India, 2024, pp. 248-252, doi: 10.1109/SSAIC61213.2024.00053.

S. Zhang, J. Li and Y. Li, "Reachable Distance Function for KNN Classification," in *IEEE Transactions on Knowledge and Data Engineering*, vol. 35, no. 7, pp. 7382-7396, 1 July 2023, doi: 10.1109/TKDE.2022.3185149

T. Xuan Pham, T. Tan Nguyen and H. Lee, "Hamming-Distance Trellis Min-Max-Based Architecture for Non-Binary LDPC Decoder," in *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 70, no. 7, pp. 2390-2394, July 2023, doi: 10.1109/TCSII.2023.3241112.

A. Awad, A. Hawash and B. Abdalhaq, "A Genetic Algorithm (GA) and Swarm-Based Binary Decision Diagram (BDD) Reordering Optimizer Reinforced With Recent Operators," in *IEEE Transactions on Evolutionary Computation*, vol. 27, no. 3, pp. 535-549, June 2023, doi: 10.1109/TEVC.2022.3170212.

Jayakumar, Nikhil, and Sunil P. Khatri. "An algorithm to minimize leakage through simultaneous input vector control and circuit modification." 2007 *Design, Automation & Test in Europe Conference & Exhibition. IEEE*, 2007.

Yuan, Lin, and Gang Qu. "A combined gate replacement and input vector control approach for leakage current reduction." *IEEE transactions on very large scale integration (vlsi) systems 14.2* (2006): 173-182.

Chen, Zhanping, et al. "Estimation of standby leakage power in CMOS circuits considering accurate modeling of transistor stacks." *Proceedings of the 1998 international symposium on Low power electronics and design.* 1998.

Leelarani, V., and M. Madhavilatha. "Verilog implementation of genetic algorithm for minimum leakage vector in input vector control approach." 2015 *International Conference on Signal Processing and Communication Engineering Systems. IEEE,* 2015.

Y. Lan, "Binary-like Real Coding Genetic Algorithm," *2023 International Conference on Pattern Recognition, Machine Vision and Intelligent Algorithms (PRMVIA)*, Beihai, China, 2023, pp. 98-102, doi: 10.1109/PRMVIA58252.2023.00023.

Grażyna Starzec, Mateusz Starzec, Leszek Rutkowski, Marek Kisiel-Dorohinicki, Aleksander Byrski, Ant colony optimization using two-dimensional pheromone for single-objective transport problems, *Journal of Computational Science*, Volume 79, 2024,102308, ISSN 1877-7503

M. Dorigo, M. Birattari and T. Stutzle, "Ant colony optimization," in *IEEE Computational Intelligence Magazine*, vol. 1, no. 4, pp. 28-39, Nov. 2006, doi: 10.1109/MCI.2006.329691.

R. Raman, V. Kumar, B. G. Pillai, D. Rabadiya, S. Patre and R. Meenakshi, "The Impact of Enhancing the k-

Means Algorithm through Genetic Algorithm Optimization on High Dimensional Data Clustering Outcomes," *2024 International Conference on Knowledge Engineering and Communication Systems (ICKECS)*, Chikkaballapur, India, 2024, pp. 1-5, doi: 10.1109/ICKECS61492.2024.10617268.