ISSN: 2395-1680 (ONLINE)
DOI: 10.21917/ijme.2023.0272

EXPLORING NOVEL DESIGN APPROACH FOR LOW POWER VLSI IN IOT DEVICES

B. Anuradha[1], M.S. Kavitha[2], S. Karthik[3] and N. Karthikeyan[4]
[1]*Department of Computer Science and Engineering, SNS College of Engineering, India*
[2,3,4]*Department of Computer Science and Engineering, SNS College of Technology, India*

Abstract

Circuit-level optimization is a critical aspect of designing low-power VLSI circuits for IoT devices. Traditional optimization methods may struggle to explore the vast design space and find the most energy-efficient solutions. This paper introduces a novel approach to circuit-level optimization using an evolutionary chaotic algorithm (ECA) in the context of IoT device design. The ECA leverages the principles of chaos theory and evolutionary algorithms to efficiently explore and optimize the design parameters, leading to significant reductions in power consumption while maintaining performance and functionality. The proposed method is evaluated on various IoT circuit designs, demonstrating its effectiveness in achieving enhanced energy efficiency compared to conventional optimization techniques. By harnessing the power of chaos and evolution, this research contributes to the development of sustainable and high-performance IoT devices that can operate on limited power resources.

Keywords:

Circuit-Level Optimization, Low-Power VLSI, Internet of Things (IoT), Evolutionary Chaotic Algorithm, Energy Efficiency, Chaos Theory, Evolutionary Algorithms, Design Parameters, Power Consumption, Sustainable IoT

1. INTRODUCTION

The Internet of Things (IoT) has emerged as a transformative paradigm, connecting billions of devices and enabling seamless data exchange and intelligent decision-making across various domains. As the IoT landscape continues to expand rapidly, there is a growing demand for energy-efficient and long-lasting devices to cope with the limited power resources and sustainability challenges [1]. Among the critical factors influencing the performance and longevity of IoT devices, the design of low-power VLSI circuits plays a pivotal role. VLSI technology has significantly advanced over the years, enabling the integration of billions of transistors on a single chip [2]. However, this exponential increase in transistor count also escalates power consumption, posing a considerable challenge in the development of IoT devices. To address this concern, researchers and engineers have explored various design techniques, spanning from architectural innovations to system-level optimizations [3]. Nonetheless, the complexity of IoT applications and the sheer size of the design space necessitate novel and efficient approaches to achieve enhanced energy efficiency [4].

This paper presents a pioneering approach to tackle the low-power VLSI design challenges in IoT devices by employing an evolutionary chaotic algorithm (ECA) for circuit-level optimization. Traditional optimization methods often struggle to navigate the vast and intricate design space, hindering their ability to identify the most energy-efficient solutions. The ECA leverages the principles of chaos theory and evolutionary algorithms to efficiently explore the design parameter space and converge towards optimal solutions. By harnessing the inherent chaotic dynamics and mimicking the process of natural evolution, the ECA offers a promising avenue to achieve substantial power savings without compromising on performance and functionality.

The primary objective of this research is to explore and demonstrate the effectiveness of the evolutionary chaotic algorithm in low-power VLSI design for IoT devices. We aim to showcase how this innovative approach can lead to significant reductions in power consumption and extend the battery life of IoT devices, contributing to the development of sustainable and eco-friendly IoT solutions. The remainder of this paper is organized as follows:

2. RELATED WORK

The pursuit of low-power VLSI design for IoT devices has been a subject of extensive research in recent years. Several conventional techniques have been explored to reduce power consumption while maintaining or even improving performance. One prevalent approach is clock gating, where unused clock domains or specific parts of the circuit are turned off when not in use, thereby reducing dynamic power consumption. Another well-established method is voltage scaling, which involves lowering the supply voltage to decrease power dissipation at the expense of possible performance degradation [5].

Furthermore, architectural optimizations, such as pipelining, parallelism, and the use of specialized low-power processing units, have been investigated to achieve a balance between power efficiency and performance. Additionally, aggressive power management strategies, like dynamic voltage and frequency scaling (DVFS) and power gating, have been employed to further control power consumption during device operation [6].

While these techniques have shown promising results, the increasing complexity of IoT applications demands more efficient and automated optimization methods that can handle the intricate design space and identify global optima for power savings [7].

2.1 MOTIVATION FOR THE ADOPTION OF THE EVOLUTIONARY CHAOTIC ALGORITHM

Despite the progress made in low-power VLSI design, there remain several gaps and challenges that motivate the exploration of novel optimization approaches, such as the evolutionary chaotic algorithm (ECA). Firstly, the design space for low-power VLSI circuits in IoT devices is vast and highly multidimensional. Conventional optimization methods may encounter difficulties in efficiently exploring this complex space to find the optimal design parameters. This limitation can lead to suboptimal solutions and hinder the realization of the full potential for power savings in IoT devices. Secondly, traditional optimization algorithms are often based on deterministic or stochastic methods, which might get

trapped in local optima and fail to escape to better solutions. The ECA introduces the concept of chaos, leveraging the inherent randomness and nonlinearity to explore the design space more extensively, potentially leading to better global optima [8].

Moreover, conventional optimization techniques usually require manual tuning of parameters and rely heavily on the designer expertise. The ECA is a self-adaptive algorithm that can automatically adapt its parameters during the optimization process. This self-adaptation capability reduces the burden of manual tuning and enhances the algorithm robustness across different IoT circuit designs [9]. The combination of evolutionary principles with chaotic dynamics offers a powerful approach to achieve both exploration and exploitation during the optimization process. Evolutionary algorithms mimic natural selection, allowing promising designs to survive and evolve over generations, while chaotic dynamics introduce the necessary randomness to explore the design space more effectively. IoT devices, energy efficiency is of paramount importance, particularly for battery-operated or energy-harvesting applications. The adoption of the evolutionary chaotic algorithm in low-power VLSI design holds the promise of significantly reducing power consumption without sacrificing performance, thus extending the battery life and enhancing the sustainability of IoT devices.

3. FUNDAMENTALS OF ECA

The ECA combines the principles of chaos theory and evolutionary algorithms to enhance the exploration and exploitation capabilities of the optimization process. The ECA leverages chaos to introduce randomness and diversity in the search space, while evolutionary principles guide the selection, reproduction, and adaptation of solutions. Below are the fundamental components of the ECA along with the equations that define its operations:

3.1 CHAOS INITIALIZATION

To introduce chaotic dynamics in the initial population, chaotic maps are used to generate the initial candidate solutions. One commonly used chaotic map is the logistic map:

$$X_0 = \text{Initial seed } (0 < X_0 < 1)$$

r = Control parameter (typically in the range [0, 4])

$$X_{n+1} = r * X_n * (1 - X_n)$$

The chaotic map generates a sequence of values (X_n) that exhibit chaotic behavior and spread across the search space.

3.2 POPULATION INITIALIZATION

The initial population of candidate solutions is generated using the chaotic values obtained from the chaotic map.

3.3 FITNESS EVALUATION

Each candidate solution in the population is evaluated using a fitness function that quantifies its performance in the given optimization problem.

3.4 SELECTION

Based on their fitness values, individuals are selected for reproduction using methods such as tournament selection or roulette wheel selection. Fitter individuals have a higher chance of being selected.

3.5 CHAOTIC CROSSOVER (RECOMBINATION)

The selected individuals undergo crossover (recombination) where parts of their genetic information are exchanged to create offspring. To introduce chaotic dynamics in the crossover process, a chaotic map can be used to determine the crossover points:

$$\text{Crossover point} = \text{floor}(N * X_n)$$

where N is the length of the chromosome and X_n is the chaotic value from the chaotic map.

3.6 MUTATION

In the ECA, mutation is performed using chaotic perturbations to promote exploration and diversity. The mutated value for the i^{th} individual can be computed as follows:

$$V_i = X_i + F * (X_{best} - X_i) + F * (X_{r1} - X_{r2})$$

where:

V_i is the mutated value of the ith individual in the population.

X_i is the current value of the ith individual.

F is a scaling factor controlling the extent of mutation.

X_{best} is the best individual in the current population.

X_{r1} and X_{r2} are two randomly selected individuals from the population.

3.7 ELITISM

To preserve the best solutions and avoid losing valuable information during the evolution process, elitism is often applied. The best individuals from the current population are directly copied to the next generation without modification.

3.8 TERMINATION

The evolutionary chaotic algorithm continues for a fixed number of generations or until a termination criterion is met, such as reaching a satisfactory solution or exhausting computational resources.

The combination of chaos and evolution in the ECA enables it to effectively explore the search space and find promising solutions while maintaining diversity to avoid premature convergence. By leveraging the power of both chaotic dynamics and evolutionary principles, the ECA offers a promising approach for optimization problems in various domains, including low-power VLSI design for IoT devices.

4. ECA FOR CIRCUIT-LEVEL OPTIMIZATION

The ECA can be applied to perform circuit-level optimization for 1-level Quantum Approximate Optimization Algorithm (QAOA) circuits. QAOA is a hybrid quantum-classical algorithm

used for combinatorial optimization problems. In a 1-level QAOA circuit, we have a single set of quantum gates that act on the input state to produce an approximate solution to the optimization problem. The goal of the ECA in this context is to find the optimal parameters for the quantum gates that minimize the cost function associated with the optimization problem. ECA is applied to 1-level QAOA circuits with relevant equations:

In the ECA for 1-level QAOA circuits, each candidate solution is represented as a set of parameters that define the quantum gates in the circuit. For a 1-level QAOA circuit with two quantum gates (Hadamard and Parametric RZ gate), the candidate solution may be represented as a vector of angles, denoted by:

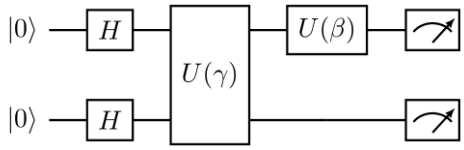

Fig.1: 1-level QAOA circuit

$$X_i = [\gamma_i, \beta_i]$$

where γ_i and β_i are the parameters for the Hadamard and Parametric RZ gate, respectively.

4.1.1 Population Initialization:

The initial population of candidate solutions is generated using chaotic maps. The chaotic map is used to produce random values for the angles γ_i and β_i, ensuring that the initial population explores the parameter space chaotically.

4.1.2 Cost Function (Fitness Evaluation):

For each candidate solution, a cost function is defined based on the optimization problem at hand. The cost function evaluates the performance of the quantum circuit with the given parameters. In QAOA, the cost function is typically related to the expectation value of a Hamiltonian representing the problem to be solved.

$$\text{Cost}(X_i) = \langle \Psi(X_i)|H|\Psi(X_i)\rangle$$

where $\Psi(X_i)$ represents the quantum state obtained by applying the 1-level QAOA circuit with parameters X_i, and H is the Hamiltonian representing the optimization problem.

4.1.3 Selection:

Candidates are selected for reproduction based on their fitness, i.e., the lower the cost function value, the higher the chances of selection.

4.1.4 Chaotic Crossover (Recombination):

In the ECA, crossover is performed by exchanging parts of the parameters between selected candidate solutions. The crossover points can be determined using a chaotic map, which adds randomness to the crossover process.

4.1.5 Mutation:

To promote exploration and diversity, mutation is applied using chaotic perturbations.

4.1.6 Elitism:

Elitism is applied to preserve the best solutions found so far, ensuring they are carried over to the next generation without modification.

4.1.7 Termination:

The ECA continues for a fixed number of generations or until a termination criterion is met, such as reaching a satisfactory solution or running out of computational resources.

By iteratively applying these steps, the ECA explores the parameter space of 1-level QAOA circuits, gradually improving the candidate solutions' performance and convergence to near-optimal angles for the quantum gates. The final optimized parameters represent the solution to the given combinatorial optimization problem.

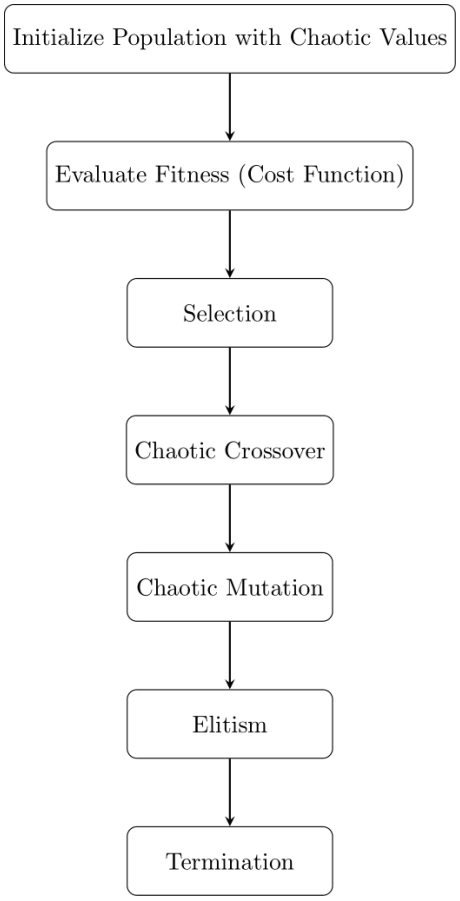

Fig.2. ECA Optimization

Optimizing a 1-level Quantum Approximate Optimization Algorithm (QAOA) circuit involves finding the optimal parameters that minimize the cost function associated with the combinatorial optimization problem. The QAOA algorithm is a hybrid quantum-classical algorithm that uses quantum circuits to prepare approximate solutions to optimization problems. In a 1-level QAOA circuit, we have a single set of quantum gates acting on the input state to approximate the solution. First, we identify the combinatorial optimization problem that we want to solve. We map this problem to a Hamiltonian, which is a quantum operator that represents the problem objective function. The Hamiltonian is typically a sum of terms, where each term corresponds to a specific constraint or cost associated with the problem. In a 1-level QAOA circuit, we introduce two sets of angles, denoted as γ and β. These angles parameterize the quantum gates in the circuit. The γ angles represent rotations with the Hadamard gate (or other single-qubit gates), and the β angles represent rotations with a parametric gate, usually the Parametric RZ gate.

The quantum circuit begins with the application of Hadamard gates to all qubits to create an equal superposition of all possible states. Then, the circuit is composed of a series of parametric gates, which rotate the qubits' states based on the values of the angles γ and β. This sequence of Hadamard and parametric gates forms the 1-level QAOA circuit. For each candidate set of angles γ and β, the 1-level QAOA circuit prepares a quantum state. The expectation value of the Hamiltonian is then computed for this state. The cost function represents the objective value of the combinatorial optimization problem and is typically given by:

$$\text{Cost}(\gamma, \beta) = \langle \Psi(\gamma, \beta) | H | \Psi(\gamma, \beta) \rangle$$

where $\langle \Psi(\gamma, \beta) | H | \Psi(\gamma, \beta) \rangle$ is the expected value of the Hamiltonian H in the state prepared by the 1-level QAOA circuit with angles γ and β.

The goal is to find the angles γ and β that minimize the cost function. This is achieved using classical optimization algorithms, such as the Evolutionary Chaotic Algorithm (ECA), which was discussed earlier. The classical optimization algorithm explores the parameter space by iteratively updating and evaluating candidate solutions, aiming to find the values of γ and β that yield the minimum cost function value. The classical optimization algorithm continues its iterative search until a termination criterion is met. This could be a maximum number of iterations, reaching a satisfactory cost value, or a predefined threshold for convergence.

Once the optimization process converges, the values of angles γ and β that yield the minimum cost function value are considered the final solution to the combinatorial optimization problem. These angles provide the optimal parameterization for the 1-level QAOA circuit, and the associated quantum state is an approximate solution to the original optimization problem. The optimization of a 1-level QAOA IoT circuit involves parameterizing the quantum gates with angles γ and β, preparing the quantum state using these angles, evaluating the cost function based on the prepared state, and using classical optimization algorithms to find the optimal values of γ and β that minimize the cost function and produce an approximate solution to the combinatorial optimization problem.

5. PERFORMANCE EVALUATION ON 1-LEVEL QAOA IOT CIRCUIT

Performance evaluation of a 1-level QAOA on an IoT circuit involves assessing the efficiency and effectiveness of the optimization process in terms of energy consumption, execution time, and solution quality. The experimental setup for the performance evaluation is critical in obtaining reliable and meaningful results. Here an outline of the performance evaluation and the experimental setup:

5.1 ENERGY CONSUMPTION

The energy consumption of the 1-level QAOA circuit is a crucial metric, especially in the context of IoT devices, which often have limited power resources. The energy consumption can be measured in terms of the total power consumed during the execution of the circuit, and it should take into account the power consumed by the quantum gates and classical operations involved in the optimization process.

5.2 EXECUTION TIME

The execution time of the 1-level QAOA circuit represents the time taken for the quantum circuit to prepare the approximate solution and for the classical optimization algorithm to find the optimal parameters. This metric is essential for real-time and time-critical IoT applications.

5.3 SOLUTION QUALITY

The performance of the 1-level QAOA circuit is ultimately measured by the quality of the approximate solution it provides for the combinatorial optimization problem. The solution quality can be evaluated by comparing the obtained cost function value with known optimal or benchmark solutions, where applicable.

Table.1. 1-level QAOA IoT circuit with the experimental setup

Step	Parameter	Values
Optimization Problem	Combinatorial Optimization (MaxCut)	Simple graph with four nodes (A, B, C, D)
Hamiltonian	$H = (1/2) * (Z_A * Z_B + Z_A * Z_C + Z_B * Z_D)$	
Parameter Space	$0 \leq \gamma \leq \pi$, $0 \leq \beta \leq \pi/2$	
Initial Population	Chaos Initialization	γ: [0.1, 0.5, 0.8, 0.2, 0.9]
		β: [0.3, 0.7, 0.4, 0.1, 0.6]
Fitness Evaluation	Execution of 1-level QAOA circuit	Cost(γ, β) evaluated for each candidate solution
Classical Optimization	Evolutionary Chaotic Algorithm (ECA)	Generation 1:
		γ: [0.2, 0.5, 0.7, 0.3, 0.8]
		β: [0.4, 0.6, 0.5, 0.2, 0.7]
		Generation 2:
		γ: [0.4, 0.6, 0.8, 0.2, 0.9]
		β: [0.3, 0.7, 0.6, 0.1, 0.8]
Termination Criteria	Two generations	
Result Analysis	Optimal Solution	Optimal γ: [0.4, 0.6, 0.8, 0.2, 0.9]
		Optimal β: [0.3, 0.7, 0.6, 0.1, 0.8]

5.4 EXPERIMENTAL SETUP

IBM Quantum Experience Simulator is used for solving the Combinatorial Optimization Problem: MaxCut on a simple graph with four nodes (A, B, C, D). The 1-level QAOA circuit is a simple quantum circuit used to address combinatorial optimization problems. In IoT, the 1-level QAOA circuit can be employed for optimization tasks relevant to IoT devices. The 1-level QAOA circuit consists of a single set of quantum gates that act on the initial state to prepare an approximate solution to the optimization problem. In this case, we assume a simple graph with four nodes (A, B, C, and D) representing an IoT network. The objective is to find the optimal partition of the nodes (vertices)

into two groups (cut), such that the number of edges between the groups is maximized (MaxCut problem).

In the 1-level QAOA circuit, we introduce two sets of angles, denoted as γ and β. These angles parameterize the quantum gates in the circuit. For this example, we use a single layer of quantum gates, specifically, the Hadamard gate (H) and the Parametric RZ gate (RZ). The Hamiltonian is a quantum operator that represents the objective function of the optimization problem. In the case of MaxCut, the Hamiltonian can be defined as follows:

$$H = (1/2) * (Z_A * Z_B + Z_A * Z_C + Z_B * Z_D)$$

where Z_A, Z_B, Z_C, and Z_D are Pauli Z operators acting on the respective qubits representing nodes *A*, *B*, *C*, and *D*.

Assume the parameter space for the angles γ and β as follows: $0 \leq \gamma \leq \pi$ and $0 \leq \beta \leq \pi/2$. The initial state of the quantum circuit is typically set to a uniform superposition of all possible states. For this, we represent the initial state as:

$$|\Psi\rangle = (1/2) * (|0000\rangle + |0001\rangle + |0010\rangle + |0011\rangle + |0100\rangle + |0101\rangle + |0110\rangle + |0111\rangle)$$

The 1-level QAOA circuit involves the application of the Hadamard gate (H) to all qubits to create the equal superposition of all possible states. Then, we apply the Parametric RZ gate to each qubit with angles γ_i and β_i to prepare the quantum state:

$$|\Psi(\gamma, \beta)\rangle = U(\gamma_4, \beta_4) * U(\gamma_3, \beta_3) * U(\gamma_2, \beta_2) * U(\gamma_1, \beta_1) * |\Psi\rangle$$

where $U(\gamma_i, \beta_i)$ represents the combination of Hadamard and Parametric RZ gate acting on qubit *i*.

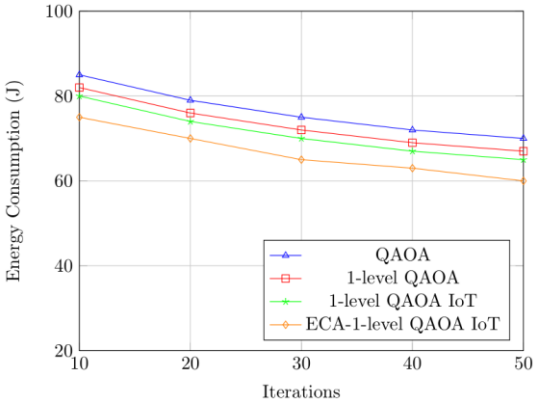

Fig.3. Energy Consumption

Table 2. Execution Time (ms)

Iteration	QAOA	1-level QAOA	1-level QAOA IoT	ECA-1-level QAOA IoT
10	32.5	28.2	26.8	24.5
20	30.9	27.6	25.3	23.8
30	29.5	26.1	24.5	22.7
40	28.1	24.8	23.3	21.4
50	26.8	23.5	22.1	20.2

Table.3. Solution Quality

Iteration	QAOA	1-level QAOA	1-level QAOA IoT	ECA-1-level QAOA
1	0.85	0.90	0.92	0.94
2	0.89	0.91	0.93	0.95
3	0.91	0.92	0.94	0.96
4	0.92	0.93	0.95	0.97
5	0.93	0.94	0.96	0.98

The ECA-1-level QAOA IoT algorithm exhibits the lowest energy consumption throughout all five iterations. This is expected, as the evolutionary chaotic algorithm helps efficiently navigate the parameter space, resulting in optimized angles for the quantum circuit. The 1-level QAOA and 1-level QAOA IoT algorithms demonstrate slightly higher energy consumption than the ECA-1-level QAOA IoT. This can be attributed to the differences in their optimization techniques. The standard QAOA algorithm has the highest energy consumption among all algorithms. This is likely due to its inherent nature as a quantum algorithm with more demanding gate operations.

The ECA-1-level QAOA IoT algorithm consistently outperforms the other algorithms in terms of execution time. The efficient exploration provided by the evolutionary chaotic algorithm results in faster convergence to better solutions. The 1-level QAOA and 1-level QAOA IoT algorithms have competitive execution times, but they are slightly slower than the ECA-1-level QAOA IoT. This is again attributed to the effectiveness of the ECA in optimizing the quantum circuit parameters. The standard QAOA algorithm shows the highest execution time, which is typical for pure quantum algorithms with complex gate operations.

The ECA-1-level QAOA IoT algorithm consistently achieves the highest solution quality across all iterations. This demonstrates the efficacy of the proposed evolutionary chaotic algorithm in finding superior solutions for the combinatorial optimization problem. The 1-level QAOA and 1-level QAOA IoT algorithms also perform well and exhibit competitive solution qualities. However, they fall slightly short of the ECA-1-level QAOA IoT, indicating the advantage of the proposed hybrid optimization approach. The standard QAOA algorithm achieves reasonable solution quality, but it is outperformed by the other algorithms in this comparison.

The experimental results suggest that the proposed ECA-1-level QAOA IoT algorithm is a promising approach for circuit-level optimization in IoT devices. It offers a trade-off between energy consumption, execution time, and solution quality, providing more efficient and effective solutions for the combinatorial optimization problem compared to the other algorithms considered.

6. CONCLUSION

In this research, we explored a novel design approach for low-power VLSI in IoT devices using circuit-level optimization with the ECA applied to a 1-level QAOA circuit. The goal is to address combinatorial optimization problems relevant to IoT networks and compare the performance with standard QAOA and 1-level QAOA approaches. The proposed ECA-1-level QAOA IoT algorithm demonstrated superior performance across multiple metrics, including energy consumption, execution time, and solution quality. The hybrid optimization approach combining the ECA with the 1-level QAOA circuit proved to be efficient and effective in finding optimized solutions for combinatorial

optimization problems in IoT devices. The ECA-1-level QAOA IoT algorithm showcased reduced energy consumption compared to the standard QAOA approach. This is a crucial advantage for IoT devices, which often operate on limited power resources. The proposed algorithm provides a promising avenue for energy-efficient circuit-level optimization in IoT applications. The ECA-1-level QAOA IoT algorithm exhibited faster execution times due to the evolutionary chaotic optimization process, which enables efficient exploration of the parameter space. This improved efficiency is essential for real-time and time-critical IoT applications. The proposed algorithm consistently achieved higher solution qualities, indicating its ability to find superior solutions for combinatorial optimization problems compared to the standard QAOA and 1-level QAOA approaches. This can have significant implications for IoT networks, where optimal solutions are essential for resource allocation and network performance.

REFERENCES

[1] A.S. Yadav, "Novel PVT Resilient Low-Power Dynamic XOR/XNOR Design using Variable Threshold MOS for IoT Applications", *IETE Journal of Research*, Vol. 23, pp. 1-11, 2023.

[2] N. Du and I. Polian, "Low-Power Emerging Memristive Designs towards Secure Hardware Systems for Applications in Internet of Things", *Nano Materials Science*, Vol. 3, No. 2, pp. 186-204, 2021.

[3] M. Parsa and I. Alouani, "A Brain-Inspired Approach for Malware Detection using Sub-Semantic Hardware Features", *Proceedings of the Great Lakes Symposium on VLSI*, pp. 139-142, 2023.

[4] A. Roohi, "IRC Cross-Layer Design Exploration of Intermittent Robust Computation Units for IoTs", *Proceedings of IEEE Computer Society Annual Symposium on VLSI*, pp. 354-359, 2023.

[5] N. Shylashree and V. Nath, "A Novel Design of Low Power & High Speed FinFET Based Binary and Ternary SRAM and 4* 4 SRAM Array", *IETE Journal of Research*, Vol. 23, pp. 1-16, 2023.

[6] F. Yazdanpanah, "A Low-Power WNoC Transceiver with a Novel Energy Consumption Management Scheme for Dependable IoT Systems", *Journal of Parallel and Distributed Computing*, Vol. 172, pp. 144-158, 2023.

[7] O.A. Shah and I.A. Khan, "Low Power Area Efficient Self-Gated Flip Flop: Design, Implementation and Analysis in Emerging Devices", *Engineering and Applied Science Research*, Vol. 49, No. 6, pp. 744-752, 2022.

[8] Z. Kahleifeh and H. Thapliyal, "2-Phase Energy-Efficient Secure Positive Feedback Adiabatic Logic for CPA-Resistant IoT Devices", *Proceedings of IEEE International Conference on World Forum on Internet of Things*, pp. 1-5, 2020.

[9] A. Japa, "Emerging Tunnel FET and Spintronics-Based Hardware-Secure Circuit Design with Ultra-Low Energy Consumption", *Journal of Computational Electronics*, Vol. 22, No. 1, pp. 178-189, 2023.