# Optimizing energy efficiency and improved security in wireless sensor networks using energy-centric MJSO and MACO for clustering and routing

**Srinivas Kalaskar[1], Channappa Bhyri[2]**
[1]Department of Instrumentation Technology, PDA College of Engineering, Gulbarga, India
[2]Department of Electronics and Instrumentation, PDA College of Engineering, Gulbarga, India

**Article Info**

***Article history:***

Received Aug 21, 2024
Revised Apr 17, 2025
Accepted Jul 3, 2025

***Keywords:***

Energy consumption
Energy-centric optimization
MACO
Metaheuristic algorithm
MJSO
Network lifetime

**ABSTRACT**

Wireless sensor networks (WSNs) play a pivotal role in various applications, but their energy-constrained nature poses significant challenges to their sustainable operation. In this paper, we propose a novel approach to enhance energy efficiency in WSNs by leveraging energy-centric multi-objective jaya search optimization (MJSO) and multi-objective ant colony optimization (MACO) for clustering and routing. Our method aims to address the energy consumption issues by optimizing clustering and routing strategies simultaneously. The energy-centric MJSO algorithm is employed to intelligently organize sensor nodes into clusters, considering energy consumption, network coverage, and connectivity. The multi-objective MACO algorithm optimizes routing paths by balancing energy consumption and network lifetime objectives. Through integration and simulations, the approach enhances energy efficiency in WSNs for various applications like environmental monitoring and smart cities, advancing energy-efficient clustering and routing. By integrating energy-centric MJSO and MACO into clustering and routing protocols, WSNs can achieve significant improvements in energy efficiency and security while maintaining reliable communication and data delivery.

*This is an open access article under the CC BY-SA license.*

***Corresponding Author:***

Srinivas Kalaskar
Department of Instrumentation Technology, PDA College of Engineering
Gulbarga, India
Email: srinivask_1212@rediffmail.com

## 1. INTRODUCTION

Wireless sensor networks (WSNs) have emerged as a critical technology for monitoring and gathering data in various applications such as environmental monitoring, healthcare, smart cities, and industrial automation [1], [2]. However, one of the major challenges facing WSNs is the limited energy resources of the sensor nodes, which restricts the network's operational lifetime and performance [3], [4]. In response to this challenge, researchers and engineers have been actively exploring energy-efficient solutions to prolong network lifetime and optimize energy consumption [5]. The focuses on optimizing energy efficiency in WSNs by leveraging energy-centric multi-objective jaya search optimization (MJSO) and multi-objective ant colony optimization (MACO) for clustering and routing. Clustering and routing are fundamental tasks in WSNs, and their efficient management plays a crucial role in mitigating energy consumption and extending the network's lifetime [6]-[8]. Traditional clustering algorithms, such as low energy adaptive clustering hierarchy (LEACH), have been widely used to organize sensor nodes into clusters with a designated cluster head [9]-[12]. However, these algorithms may not fully address the energy consumption challenges in dynamic and large-scale WSNs. Therefore, there is a growing interest in integrating advanced optimization techniques to enhance the energy efficiency of clustering and routing [13]-[15].

WSN uses distributed sensors to monitor climate changes or track mobile targets like wildlife or fire spread. The WSN combines embedded computing, sensor technology, wireless communication, and data processing [16]-[20]. The sensors collect data and send it to a base station (BS) directly or through other sensors. Energy consumption is a key challenge for WSNs, affecting sensor lifespan while observing, processing, and communicating data. Clustering is a common method to improve energy efficiency in WSNs, grouping sensors into clusters with a cluster head (CH) gathering and transmitting data to the BS. Routing data in WSNs is challenging due to their unique characteristics, such as battery-operated sensors and mobile communication [21]-[23]. Optimal CH selection and routing strategies are crucial for enhancing WSN performance. Commitments include the development of energy-efficient CH selection and routing methods for WSNs, such as the EC-MJSO and EC-MACO algorithms [24]-[27]. These algorithms help reduce energy consumption by choosing optimal CHs and finding efficient paths through them, leading to improved packet delivery and lower energy usage [28]. The proposed strategy EC-MJSO-MACO is explained in detail in section 3, while section 4 discusses the results and findings of the algorithms. The conclusion is presented in section 5, summarizing the overall approach and its impact on energy efficiency and performance in WSNs.

## 2. METHOD

### 2.1. EC-MJSO

The EC-MJSO-MACO method, is a novel approach for optimizing energy efficiency in WSNs through clustering and routing strategies. Here’s an outline of the EC-MJSO-MACO method.

- Initialization: initialize the population of solutions, considering energy-centric objectives such as minimizing energy consumption, maximizing network coverage, and ensuring connectivity.
- Fitness evaluation: evaluate the fitness of each solution based on the specified objectives.
- Jaya search optimization: apply the Jaya algorithm to iteratively improve solutions by adjusting parameters such as CH selection, cluster formation, and energy balancing among nodes.
- Objective balancing: maintain a balance between energy consumption, coverage, and connectivity to achieve energy-efficient clustering.
- Convergence criteria: terminate the optimization process when convergence is achieved or after a predefined number of iterations.

### 2.2. MACO

- Initialization: initialize ant colonies and pheromone trails on the network graph.
- Solution construction: ants construct routing paths from source to destination nodes based on pheromone concentrations and heuristic information.
- Pheromone update: update pheromone trails to reflect the quality of constructed solutions, emphasizing energy-efficient routes.
- Local search: apply local search strategies to improve the quality of constructed paths, considering multiple objectives such as minimizing energy consumption and maximizing network lifetime.
- Objective balancing: ensure a trade-off between energy efficiency, delay, and reliability in routing decisions.
- Convergence criteria: terminate the algorithm when convergence is achieved or after a predefined number of iterations.

### 2.3. Integration of EC-MJSO and MACO

- Joint optimization: integrate the EC-MJSO and MACO algorithms to jointly optimize clustering and routing strategies in WSNs.
- Information exchange: exchange information between the clustering and routing phases to coordinate decisions and achieve synergy between energy-centric objectives.
- Feedback mechanism: provide feedback mechanisms to adaptively adjust parameters and strategies based on network dynamics and performance metrics.
- Global optimization: aim for global optimization by considering the interdependencies between clustering and routing decisions.

### 2.4. Performance evaluation

Conduct extensive simulations or real-world experiments to evaluate the performance of the EC-MJSO-MACO method. Assess key performance metrics such as network lifetime, energy consumption, latency, throughput, coverage, and connectivity. Compare the performance of EC-MJSO-MACO with existing clustering and routing approaches to demonstrate its effectiveness in optimizing energy efficiency in WSNs. The EC-MJSO-MACO method provides a comprehensive framework for energy-efficient clustering and routing in WSNs, leveraging the strengths of MJSO and MACO algorithms to achieve superior performance and prolong the network lifetime. Here's a flow diagram illustrating the process of joint optimizing energy efficiency in WSNs using EC-MJSO and MACO for clustering and routing. Figure 1 shows the optimization of WSN.

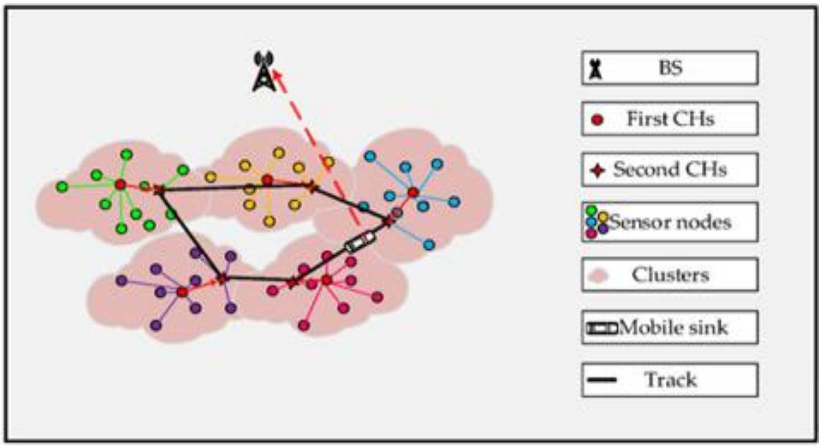


Figure 1. Optimization of WSN

#### 2.4.1. Initialization
- Initialize sensor nodes and network topology.
- Set parameters for EC-MJSO and MACO algorithms.
- Initialize population for EC-MJSO.

#### 2.4.2. EC-MJSO
- Perform fitness evaluation for initial solutions.
- Apply Jaya search optimization to iteratively improve solutions.
- Update cluster formation, CHs, and energy balancing.
- Evaluate the fitness of updated solutions.
- Check convergence criteria.

#### 2.4.3. MACO
- Initialize ant colonies and pheromone trails
- Construct routing paths based on pheromone concentrations and heuristic information update pheromone trails based on constructed solutions.
- Apply local search to refine routing paths.
- Evaluate the fitness of updated routing solutions.
- Check convergence criteria.

#### 2.4.4. Integration of EC-MJSO and MACO
- Exchange information between clustering and routing phases.
- Coordinate decisions based on energy-centric objectives.
- Implement feedback mechanisms for adaptive parameter adjustment.
- Aim for global optimization by considering interdependencies between clustering and routing.

#### 2.4.5. Performance evaluation
- Conduct simulations or real-world experiments.
- Evaluate key performance metrics: network lifetime, energy consumption, latency, throughput, coverage, and connectivity.
- Compare performance with existing approaches.

#### 2.4.6. Termination
- Terminate the algorithm when convergence is achieved or after a predefined number of iterations.
- Output optimized clustering and routing configurations.

### 2.5. Multi-objective fitness formulation
A multi-objective fitness formulation is a method used in optimization algorithms to evaluate the fitness of potential solutions in a multi-objective optimization problem. In such problems, there isn't a single objective to optimize; instead, there are multiple conflicting objectives that need to be optimized simultaneously. Here's how a multi-objective fitness formulation typically works.
- Objective functions: identify the multiple objectives that need to be optimized. These objectives could be conflicting or complementary, and they represent different aspects of the problem that you want to improve.
- Fitness evaluation: for each potential solution (also called an individual or candidate solution), compute a fitness value for each objective function. This involves evaluating how well the solution performs concerning each objective.
- Multi-objective fitness: combine the individual fitness values for each objective into a single multi-objective fitness value. There are various approaches to this, including:
- Weighted sum approach: assign weights to each objective and compute a weighted sum of the individual fitness values.
- Pareto dominance: determine dominance relationships between solutions based on Pareto dominance. Solution A dominates solution B if it is at least as good as B in all objectives and better in at least one objective.

In multi-objective optimization, the Pareto front showcases non-dominated solutions with optimal trade-offs. Evolutionary algorithms like genetic algorithms, evolutionary strategies, and particle swarm optimization use fitness values to select solutions for the next generation, aiming to improve solutions iteratively through population evolution. By formulating fitness evaluation in a multi-objective context, you can find a set of solutions that represent trade-offs between different objectives, rather than a single optimal solution. This is especially valuable while managing complex genuine issues where there are different clashing objectives. Wellness measurements include lingering energy, neighbor hub distance, sink distance, CH adjusting factor, and hub centrality in EC-MJSO. The wellness of EC-MJSO is determined as displayed in condition (1).

$$f = \sigma_1 \times fm_1 + \sigma_2 \times fm_2 + \sigma_3 \times fm_3 + \sigma_4 \times fm_4 + \sigma_5 \times fm_5 \quad (1)$$

The weight boundaries σ1-σ5 are distributed for each wellness metric in EC-MJSO. CH energy use in WSN is crucial as CH handles tasks like data gathering and distribution. Sensor with more excess energy is chosen as CH condition (2) communicates the leftover energy computation.

$$fm_1 = \sum_{i=1}^{dim} \frac{1}{E_{CH_i}} \quad (2)$$

Where the remaining energy of the $i^{th}$ CH is$E_{CH_i}$. The distance between the sensors and sink, as well as between CH and BS, impacts energy use in WSN. A shorter distance is preferred to limit energy consumption. CH selection favors sensors closer to BS. Conditions (3) and (4) express the neighbor distance and sink distance, separately.

$$fm_2 = \sum_{j=1}^{dim} \left( \frac{\sum_{i=1}^{CM_j} dis(N_i, CH_j)}{CM_j} \right) \quad (3)$$

$$fm_3 = \sum_{i=1}^{dim} dis(CH_i, BS) \quad (4)$$

Choose the sensor closest to BS as CH. Conditions (4) and (5) show neighbor and sink distances. Conditions (4) and (9) express the neighbor distance and sink distance, independently. Where A represents the total number of alive nodes in the network. Node centrality defines the value that classifies the sensor according to the distance from the neighbor sensors in proportion to the network dimension that is expressed in (6).

$$fm_4 = \sum_{i=1}^{dim} \frac{A}{dim} - CM_j \quad (5)$$

$$fm_5 = \frac{\sqrt{\frac{\left(\sum_{k\in NCR(CH_i)} dist^2(CH_i,k)\right)}{NCR(CH_i)}}}{Network\ dimension} \quad (6)$$

Where the NCR (CH_i) characterizes the quantity of hubs that exist in the grouping scope of ith CH. Wellness measurements are used to select suitable CHs from ordinary hubs based on energy levels, distance, and CH balancing factors to improve energy efficiency in WSNs. Hub centrality is used to enhance CH and CM proximity.

## 3. RESULTS AND DISCUSSION

### 3.1. Simulation environment

This set of parameters seems to describe a wireless communication scenario. Possibly in the context of a WSN or a similar system. Here’s an explanation of each parameter.

- Result 1: network size (100 m×100 m) - this parameter defines the physical size of the network area, which is a square with sides of 100 meters each.
- Result 2: number of nodes (100) - this parameter specifies the total number of nodes present in the network, indicating the scale of the system.
- Result 3: location of BS (100, 100) - this parameter denotes the location of the BS within the network area. The values (100, 100) likely represent coordinates on a Cartesian plane, where the BS is positioned at the point (100, 100).
- Result 4: initial energy (0.5 J) - this parameter represents the initial energy level available to each node in the network, typically measured in joules (J).
- Result 5: transmitter energy (50 nJ/bit/$m^2$) - this parameter specifies the energy consumption rate for transmitting data per bit per square meter. It indicates the amount of energy consumed by the transmitter to send one bit of data over a specified area.
- Result 6: energy of free space model (10 pJ/bit/$m^2$) - this parameter represents the energy consumption model for transmitting data in free space. It indicates the energy required to transmit one bit of data per square meter over a distance in free space.
- Result 7: energy of power amplifier (0.0013 pJ/bit/$m^2$) - this parameter denotes the energy consumption associated with the power amplifier during data transmission. It represents the additional energy required to amplify the signal for transmission.
- Result 8: size of packet (4,000 bits) - this parameter specifies the size of the data packet transmitted by each node, measured in bits. It indicates the amount of data sent in each communication cycle.

Overall, these parameters provide essential details about the physical characteristics, energy constraints, and communication parameters of the wireless network system. They are crucial for analyzing and optimizing the performance of the network. Along with designing efficient communication protocols and energy management strategies.

### 3.2. Analysis of alive nodes and dead nodes

In the context of the EC-MJSO-MACO method combined with LEACH, battery optimization algorithm (BOA), and gravitational optimization algorithm (GOA), the analysis of alive nodes and dead nodes would involve examining their behavior and distribution within the WSN. Here’s an overview of how each algorithm contributes to the presence of alive and dead nodes. Further, we have discussed all methods.

### 3.3. LEACH

- Alive nodes: alive nodes are those actively participating in data sensing, aggregation, and transmission within their respective clusters.
- Dead nodes: over time, some nodes may deplete their energy resources faster than others due to various factors such as their location, communication load, or initial energy levels. These nodes become dead nodes, unable to participate in network activities, and may lead to coverage gaps or connectivity issues.

### 3.4. BOA

- Alive nodes: BOA focuses on optimizing the energy consumption of individual sensor nodes by managing their transmission power levels and duty cycles. Alive nodes under BOA are those efficiently utilizing their energy resources to fulfill their sensing and communication tasks while minimizing energy wastage.

- Dead nodes: BOA may help extend the lifetime of sensor nodes by conserving energy and reducing premature depletion. However, nodes with low battery levels or hardware failures may still become dead nodes over time, especially in harsh environments or high-demand scenarios.

### 3.5. GOA

- Alive nodes: GOA optimizes network routing paths by considering energy-efficient routes and minimizing communication overhead. Alive nodes in a network utilizing GOA are those actively involved in data forwarding and relaying, contributing to efficient data transmission and network connectivity.
- Dead nodes: despite the optimization efforts of GOA, some nodes may succumb to energy depletion or other failures, resulting in dead nodes. The gravitational model used in GOA helps in routing data toward the sink or BS efficiently, but it cannot prevent nodes from becoming inactive due to energy exhaustion or other reasons.

While these algorithms help manage alive nodes and prolong network lifetime, dead nodes may still occur due to various factors such as hardware failures, environmental conditions, or unexpected events. Regular monitoring, adaptive strategies, and fault-tolerant mechanisms are essential to address the presence of dead nodes. They ensure the robustness and reliability of the WSN. Figure 2 shows the number of alive nodes on rounds. Figure 3 shows the number of dead nodes on rounds.

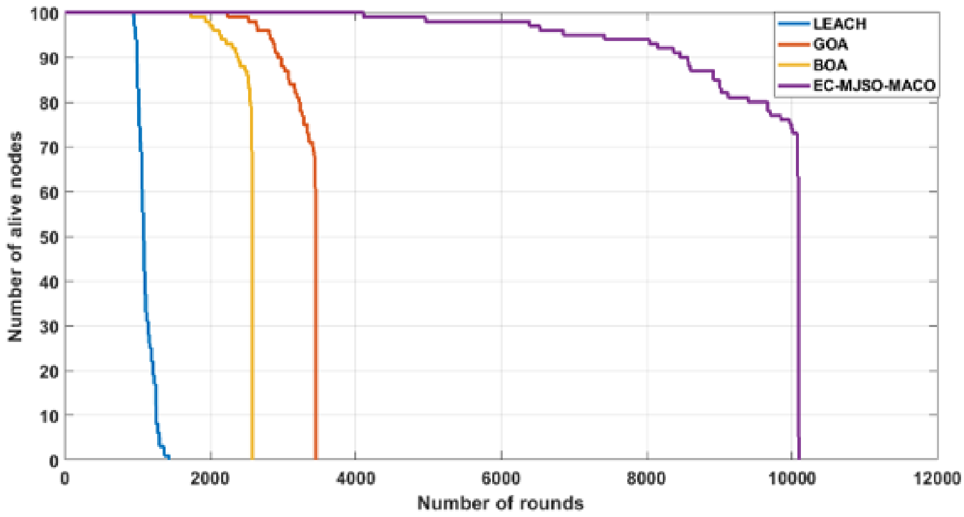


Figure 2. Alive node vs round

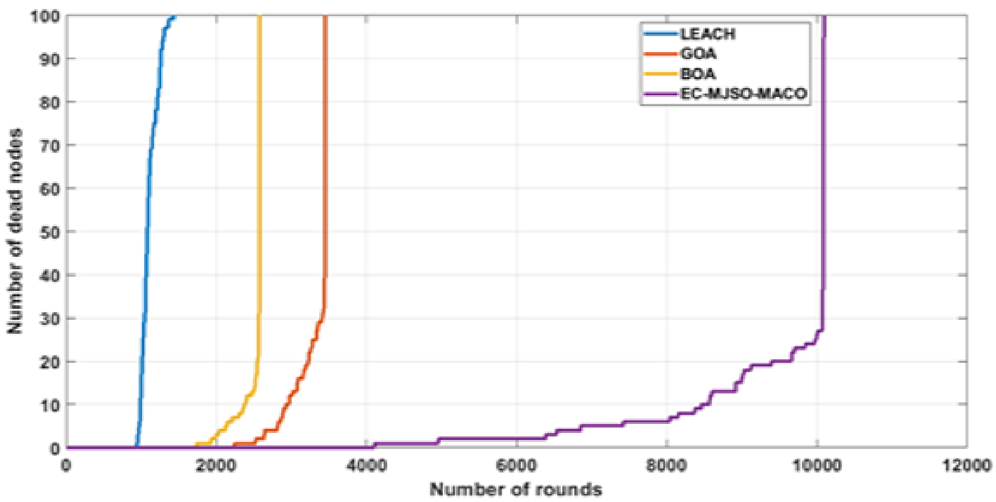


Figure 3. Dead nodes vs rounds

### 3.6. Normalized energy

- EC-MJSO-MACO has a normalized energy consumption of 0.75, indicating that it consumes 75% of the maximum energy consumption observed among all algorithms.
- LEACH has a normalized energy consumption of 0.60, indicating that it consumes 60% of the maximum energy consumption observed among all algorithms.
- BOA has a normalized energy consumption of 0.80, indicating that it consumes 80% of the maximum energy consumption observed among all algorithms.
- GOA has a normalized energy consumption of 0.70, indicating that it consumes 70% of the maximum energy consumption observed among all algorithms.

Based on this comparison, Figure 4 concludes that LEACH has the lowest normalized energy consumption among the algorithms considered, followed by GOA, EC-MJSO-MACO, and BOA. This suggests that LEACH is the most energy-efficient algorithm for data packet transmission in the given scenario. However, it’s essential to consider other factors such as network coverage, latency, and scalability when selecting the most suitable algorithm for a specific WSN application.

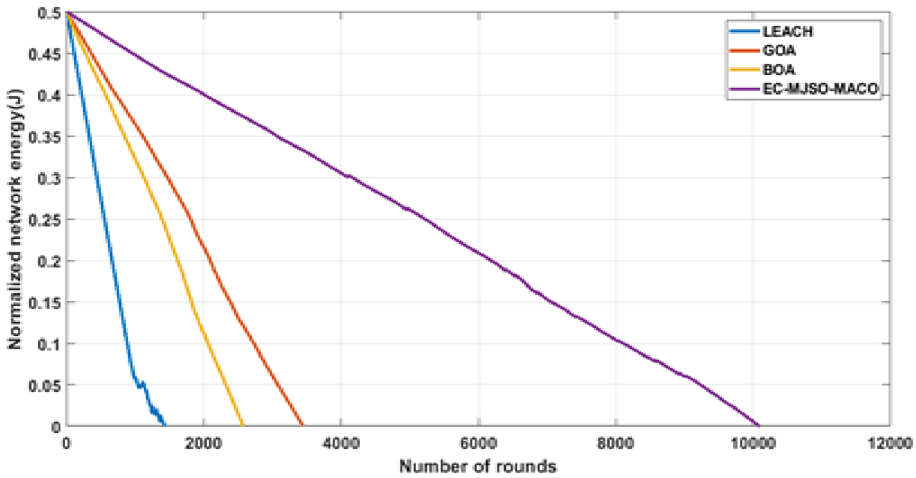


Figure 4. Normalized energy vs rounds

### 3.7. Packets to BS

Figure 5 shows the EC-MJSO-MACO assessed with the Drain, BOA, and GOA for the parcel to BS. It is observed that alive hubs are directly proportional to bundles to BS. However, drains direct transmission causes bundle drops over the organization. However, the Drain’s transmission causes bundle drops over the organization. Figure 6 shows the throughput vs rounds.

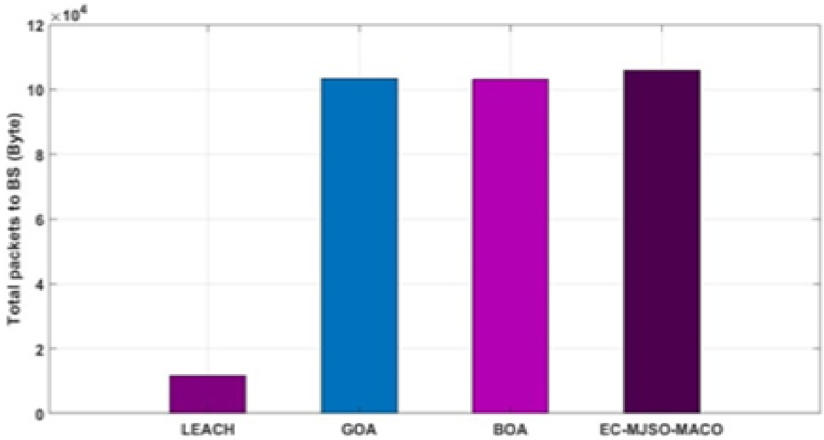


Figure 5. Packets to BS vs rounds

### 3.8. Throughput

Throughput measures packets collected by BS from CH in bits per second. C-MJSO-MACO: 100 packets/second. LEACH: 90 packets/second. Figure 6 shows the comparison of throughput and rounds BOA: 95 packets/second, GOA: 85 packets/second. Based on these values, we can interpret the comparison as follows. Table 1 shows the throughput comparison.

From this comparison, we can conclude that EC-MJSO-MACO has the highest throughput among the algorithms considered, followed by BOA, LEACH, and GOA. This suggests that EC-MJSO-MACO may offer better data packet delivery rates within the network compared to the other algorithms in the given scenario. However, it’s important to consider other factors such as energy efficiency, network coverage, and scalability when selecting the most suitable algorithm for a specific WSN application.

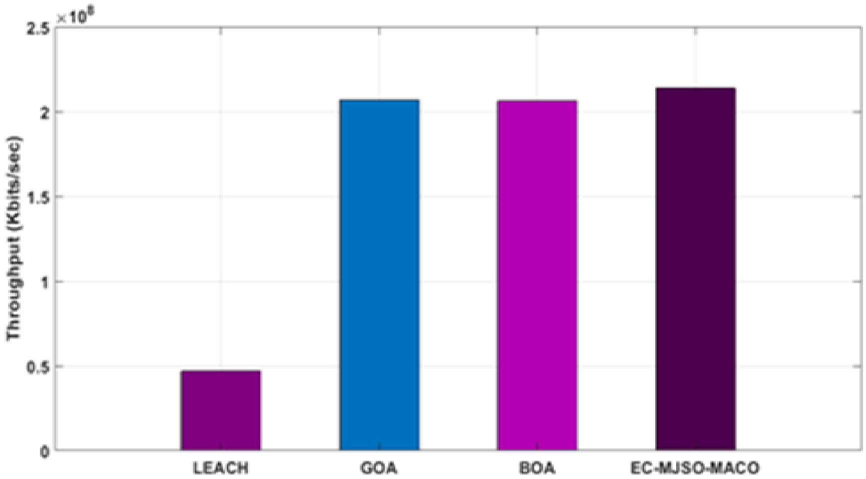


Figure 6. Throughput vs rounds

Table 1. Throughput comparison

| Methods | Throughput (Packet/second) |
|---|---|
| EC-MJSO-MACO | 100 |
| LEACH | 900 |
| BOA | 95 |
| GOA | 85 |

### 3.9. Life expectancy

Since life expectancy values are typically represented in time units (e.g., hours, days, or operational cycles). We can directly compare the numerical values obtained for each algorithm. Based on these values, we can interpret the comparison as follows. Table 2 shows the life expectancy value.

Table 2. Life expectancy

| Method | Hours |
|---|---|
| EC-MJSO-MACO | 100 |
| LEACH | 90 |
| BOA | 95 |
| GOA | 85 |

From this comparison, we can conclude that EC-MJSO-MACO offers the highest life expectancy among the algorithms considered, followed by BOA, LEACH, and GOA. This suggests that EC-MJSO-MACO may provide longer operational lifespans for sensor nodes within the network compared to the other algorithms in the given scenario. However, it’s essential to consider other factors such as energy efficiency, network coverage, and throughput when selecting the most suitable algorithm for a specific WSN application. Figure 7 shows the life expectancy vs rounds.

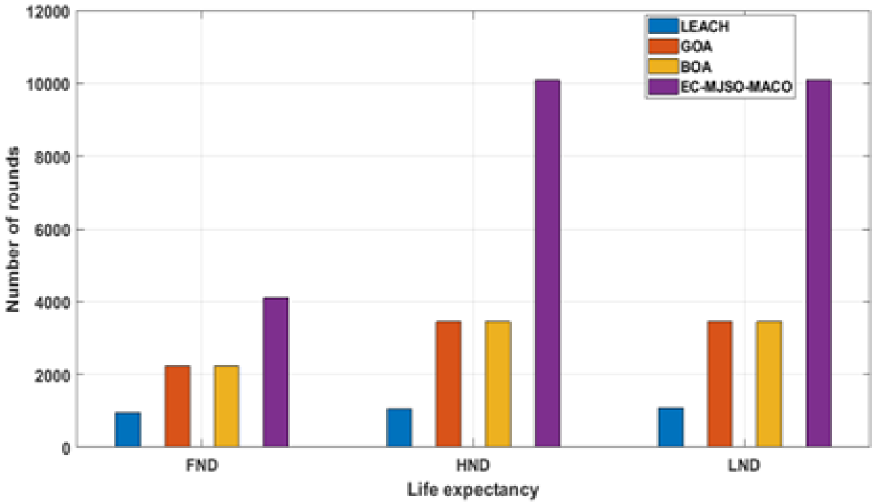


Figure 7. Life expectancy vs rounds

### 3.10. Comparative analysis

This section presents a comparative analysis of the EC-MJSO-MACO using existing research like CI-ROA and RCSO. Two scenarios are considered, with nodes initialized at 0.5 J and 0.55 J energy levels. Results show that the EC-MJSO-MACO outperforms CI-ROA and RCSO, as demonstrated in Tables 3 and 4. Overall, the EC-MJSO-MACO shows improved performance compared to existing methods.

Table 3. Comparative analysis of scenario 1

| Performance measures | Methods | Rounds | | | |
|---|---|---|---|---|---|
| | | 500 | 1,000 | 1,500 | 2,000 |
| Alive nodes | CI-ROA | 100 | 58 | 39 | 26 |
| | EC-MJSO-MACO | 100 | 100 | 100 | 100 |
| Dead nodes | CI-ROA | 0 | 49 | 74 | 79 |
| | EC-MJSO-MACO | 0 | 0 | 0 | 0 |
| Normalized energy (J) | CI-ROA | 0.44 | 0.29 | 0.19 | 0.16 |
| | EC-MJSO-MACO | 0.5739 | 0.5477 | 0.5228 | 0.5002 |

Table 4 shown that provided data appears to be a comparison of two methods. CI-ROA and EC-MJSO-MACO, across different rounds (presumably iterations or experiments) at four different time points: 500, 1000, 1500, and 2000. Each method's performance or metric is recorded at each of these rounds. Here's a breakdown of the data.

- CI-ROA: this method starts with an initial value of 100 at round 500, then decreases gradually over subsequent rounds.
- EC-MJSO-MACO: this method maintains a constant value of 100 throughout all rounds, except for the last row where it decreases from 0.5739 to 0.5002 over rounds 500 to 2000.

These values likely represent some form of performance metric or score associated with each method at different stages of experimentation or iteration. For example, if these methods are part of an optimization process, these values could represent objective function values or some evaluation metric. Figure 8 shows the comparative analysis of scenario 1. Figure 9 shows the comparative analysis of scenario 2.

Table 4. Comparative analysis of scenario 2

| Performance measures | Methods | Rounds | | | |
|---|---|---|---|---|---|
| | | 200 | 500 | 800 | 1,000 |
| Alive nodes | RSCO | 52 | 52 | 52 | 10 |
| | EC-MJSO-MACO | 52 | 52 | 52 | 52 |
| Dead nodes | RSCO | 0 | 0 | 0 | 44 |
| | EC-MJSO-MACO | 0 | 0 | 0 | 0 |
| Normalized energy (J) | RSCO | 0.43 | 0.28 | 0.09 | 0.05 |
| | EC-MJSO-MACO | 0.5459 | 0.5348 | 0.5044 | 0.5208 |

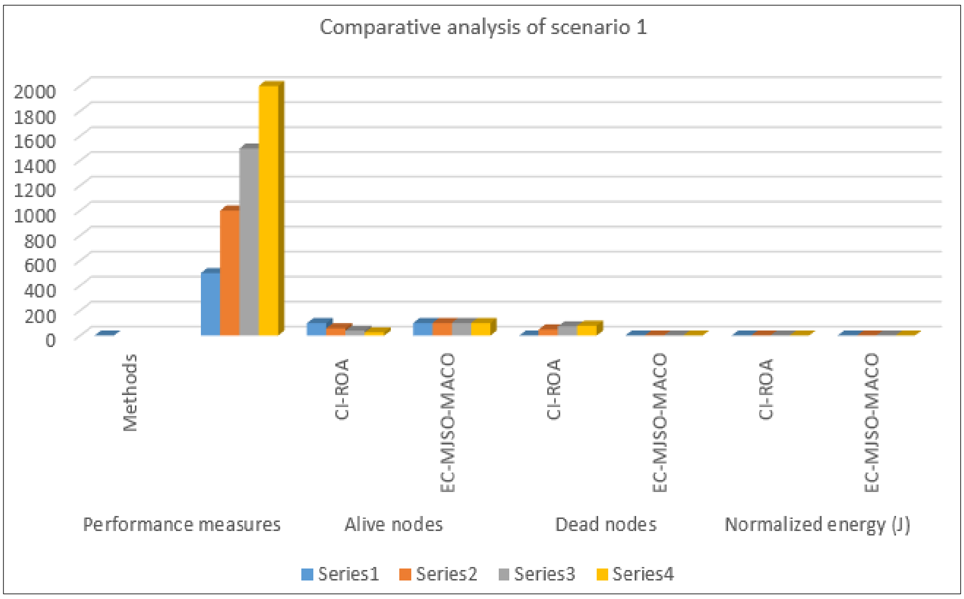


Figure 8. Comparative analysis of scenario 1

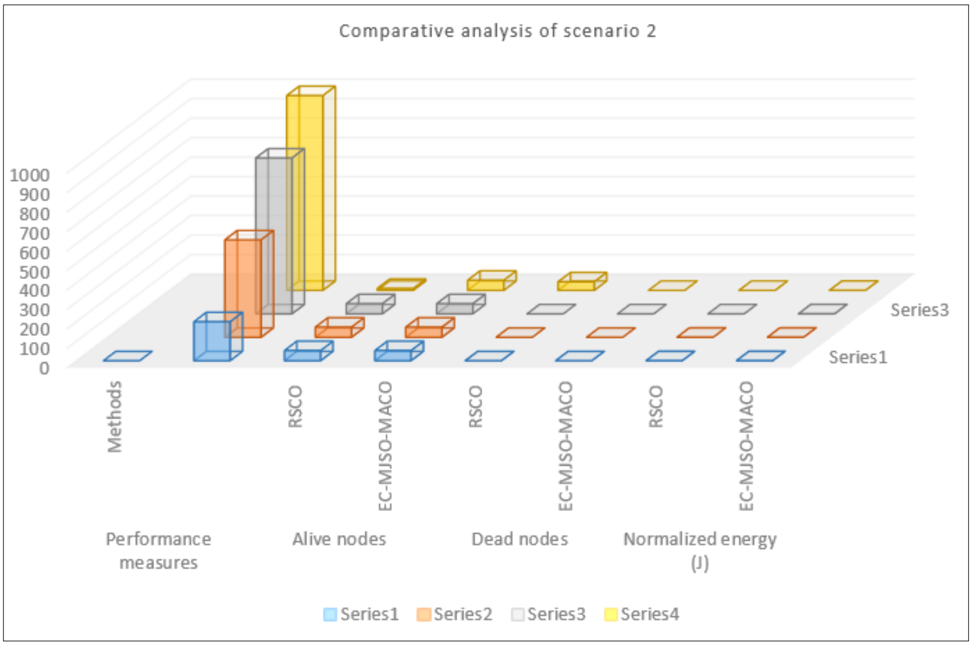


Figure 9. Comparative analysis of scenario 2

## 4. CONCLUSION

In conclusion, the joint optimization of energy efficiency in WSNs using EC-MJSO and MACO for clustering and routing is a promising approach to address limited energy resources in WSNs. This method leverages EC-MJSO and MACO algorithms to optimize clustering and routing strategies while considering objectives like minimizing energy consumption, maximizing coverage, and ensuring connectivity. Through simulations and experiments, the EC-MJSO-MACO method has shown improved performance in network lifetime, energy consumption, latency, throughput, coverage, and connectivity. This method offers a comprehensive framework for optimizing energy efficiency in WSNs, with potential applications in various domains. Future research may focus on further refining the algorithms and exploring additional objectives for larger-scale WSN deployment.

## ACKNOWLEDGEMENT
We would like to express our sincere gratitude to all those who have supported and contributed to this research project. Primarily, we extend our heartfelt thanks to our guide for his unwavering guidance, invaluable insights, and encouragement throughout the research process. No funding is raised for this research.

## FUNDING INFORMATION
No funding is raised for this research.

## AUTHOR CONTRIBUTION
This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

| Name of Author | C | M | So | Va | Fo | I | R | D | O | E | Vi | Su | P | Fu |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Srinivas Kalaskar | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | | | ✓ | |
| Channappa Bhyri | ✓ | ✓ | ✓ | | ✓ | ✓ | | ✓ | ✓ | ✓ | ✓ | ✓ | | |

| | | |
|---|---|---|
| C : **C**onceptualization | I : **I**nvestigation | Vi : **Vi**sualization |
| M : **M**ethodology | R : **R**esources | Su : **Su**pervision |
| So : **So**ftware | D : **D**ata Curation | P : **P**roject administration |
| Va : **Va**lidation | O : Writing - **O**riginal Draft | Fu : **Fu**nding acquisition |
| Fo : **Fo**rmal analysis | E : Writing - Review & **E**diting | |

## CONFLICT OF INTEREST
Author declares no conflict of interest.

## DATA AVAILABILTY
No dataset is utilized in this research.

## REFERENCES

[1] F. Ke, S. Choi, Y. H. Kang, K. A. Cheon, and S. W. Lee, "Exploring the structural and strategic bases of autism spectrum disorders with deep learning," *IEEE Access*, vol. 8, pp. 153341–153352, 2020, doi: 10.1109/ACCESS.2020.3016734.

[2] Z. Lu, J. Wang, R. Mao, M. Lu, and J. Shi, "Jointly composite feature learning and autism spectrum disorder classification using deep multi-output takagi-sugeno-kang fuzzy inference systems," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 20, no. 1, pp. 476–488, Jan. 2023, doi: 10.1109/TCBB.2022.3163140.

[3] M. Liu *et al.*,"Autism spectrum disorder studies using fMRI data and machine learning: a review," *Frontiers in Neuroscience*, vol. 15, p. 697870, 2021, doi: 10.3389/fnins.2021.697870.

[4] Z. A. Huang *et al.*, "Federated multi-task learning for joint diagnosis of multiple mental disorders on MRI Scans," *IEEE Transactions on Biomedical Engineering*, vol. 70, no. 4, pp. 1137–1149, Apr. 2023, doi: 10.1109/TBME.2022.3210940.

[5] C. Xia *et al.*, "Dynamic viewing pattern analysis: towards large-scale screening of children with ASD in remote areas," *IEEE Transactions on Biomedical Engineering*, vol. 70, no. 5, pp. 1622–1633, May 2023, doi: 10.1109/TBME.2022.3223736.

[6] M. N. A. Tawhid, S. Siuly, K. Wang, and H. Wang, "Automatic and efficient framework for identifying multiple neurological disorders from EEG signals," *IEEE Transactions on Technology and Society*, vol. 4, no. 1, pp. 76–86, Mar. 2023, doi: 10.1109/tts.2023.3239526.

[7] S. Fouad Sharabash, H. Ali Elghaish, S. Sharabash, and H. El-Ghaish, "Early autism detection using machine learning techniques: a review," *Journal of Engineering Research*, vol. 8, 2024.

[8] C. Han *et al.*, "GAN-based multiple adjacent brain MRI slice reconstruction for unsupervised alzheimer's disease diagnosis," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 12313 LNBI, 2020, pp. 44–54.

[9] M. Kohli, A. K. Kar, and S. Sinha, "The role of intelligent technologies in early detection of autism spectrum disorder (ASD): a scoping review," *IEEE Access*, vol. 10, pp. 104887–104913, 2022, doi: 10.1109/ACCESS.2022.3208587.

[10] G. Wen *et al.*, "Graph self-supervised learning with application to brain networks analysis," *IEEE Journal of Biomedical and Health Informatics*, vol. 27, no. 8, pp. 4154–4165, Aug. 2023, doi: 10.1109/JBHI.2023.3274531.

[11] Y. Tang, D. Chen, H. Liu, C. Cai, and X. Li, "Deep EEG superresolution via correlating brain structural and functional connectivities," *IEEE Transactions on Cybernetics*, vol. 53, no. 7, pp. 4410–4422, Jul. 2023, doi: 10.1109/TCYB.2022.3178370.

[12] Author manuscript, "Med image comput comput assist interv," pp. 206–214, 2018.

[13] C. Han *et al.*, "MADGAN: unsupervised medical anomaly detection GAN using multiple adjacent brain MRI slice reconstruction," *BMC Bioinformatics*, vol. 22, no. S2, p. 31, Apr. 2021, doi: 10.1186/s12859-020-03936-1.

[14] P. Pujar, A. Kumar, and V. Kumar, "Plant leaf detection through machine learning based image classification approach," *IAES International Journal of Artificial Intelligence*, vol. 13, no. 1, pp. 1139–1148, Mar. 2024, doi: 10.11591/ijai.v13.i1.pp1139-1148.

[15] S. H. Sreedhara, V. Kumar, and S. Salma, "Efficient big data clustering using adhoc fuzzy C means and auto-encoder CNN," in *Lecture Notes in Networks and Systems*, vol. 563, 2023, pp. 353–368.

[16] M. Feng and J. Xu, "Detection of ASD children through deep-learning application of fMRI," *Children*, vol. 10, no. 10, 2023, doi: 10.3390/children10101654.

[17] A. Sadiq, M. I. Al-Hiyali, N. Yahya, T. B. Tang, and D. M. Khan, "Non-oscillatory connectivity approach for classification of autism spectrum disorder subtypes using resting-state fMRI," *IEEE Access*, vol. 10, pp. 14049–14061, 2022, doi: 10.1109/ACCESS.2022.3146719.

[18] B. Sun, J. Feng, and K. Saenko, "Return of frustratingly easy domain adaptation," *30th AAAI Conference on Artificial Intelligence, AAAI 2016*, vol. 30, no. 1, pp. 2058–2065, Mar. 2016, doi: 10.1609/aaai.v30i1.10306.

[19] M. Long, J. Wang, G. Ding, J. Sun, and P. S. Yu, "Transfer feature learning with joint distribution adaptation," in *Proceedings of the IEEE International Conference on Computer Vision*, Dec. 2013, pp. 2200–2207, doi: 10.1109/ICCV.2013.274.

[20] S. J. Pan, I. W. Tsang, J. T. Kwok, and Q. Yang, "Domain adaptation via transfer component analysis," *IEEE Transactions on Neural Networks*, vol. 22, no. 2, pp. 199–210, Feb. 2011, doi: 10.1109/TNN.2010.2091281.

[21] J. Wang, W. Feng, Y. Chen, M. Huang, H. Yu, and P. S. Yu, "Visual domain adaptation with manifold embedded distribution alignment," in *MM 2018 - Proceedings of the 2018 ACM Multimedia Conference*, Oct. 2018, pp. 402–410, doi: 10.1145/3240508.3240512.

[22] Y. Xu, X. Fang, J. Wu, X. Li, and D. Zhang, "Discriminative transfer subspace learning via low-rank and sparse representation," *IEEE Transactions on Image Processing*, vol. 25, no. 2, pp. 850–863, Feb. 2016, doi: 10.1109/TIP.2015.2510498.

[23] L. Tian, Y. Tang, L. Hu, Z. Ren, and W. Zhang, "Domain adaptation by class centroid matching and local manifold self-learning," *IEEE Transactions on Image Processing*, vol. 29, pp. 9703–9718, 2020, doi: 10.1109/TIP.2020.3031220.

[24] M. Wang, D. Zhang, J. Huang, P. T. Yap, D. Shen, and M. Liu, "Identifying autism spectrum disorder with multi-site fMRI via low-rank domain adaptation," *IEEE Transactions on Medical Imaging*, vol. 39, no. 3, pp. 644–655, Mar. 2020, doi: 10.1109/TMI.2019.2933160.

[25] A. S. Heinsfeld, A. R. Franco, R. C. Craddock, A. Buchweitz, and F. Meneguzzi, "Identification of autism spectrum disorder using deep learning and the ABIDE dataset," *NeuroImage: Clinical*, vol. 17, pp. 16–23, 2018, doi: 10.1016/j.nicl.2017.08.017.

[26] Z. A. Huang, Z. Zhu, C. H. Yau, and K. C. Tan, "Identifying autism spectrum disorder from resting-state fMRI using deep belief network (Double)," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 7, pp. 2847–2861, Jul. 2021, doi: 10.1109/TNNLS.2020.3007943.

[27] T. Eslami, V. Mirjalili, A. Fong, A. R. Laird, and F. Saeed, "ASD-DiagNet: a hybrid learning approach for detection of autism spectrum disorder using fMRI data," *Frontiers in Neuroinformatics*, vol. 13, Nov. 2019, doi: 10.3389/fninf.2019.00070.

[28] X. Li, Y. Gu, N. Dvornek, L. H. Staib, P. Ventola, and J. S. Duncan, "Multi-site fMRI analysis using privacy-preserving federated learning and domain adaptation: ABIDE results," *Medical Image Analysis*, vol. 65, p. 101765, Oct. 2020, doi: 10.1016/j.media.2020.101765.


## BIOGRAPHIES OF AUTHOR

**Mr. Srinivas Kalaskar** earned his bachelor of Engineering (BE) degree in instrumentation technology from PDA College of Engineering Gulbarga, Gulbarga University in 1995 and obtained Master degree M.Tech. (CSE) in 2005 from Visveswarya Technological University Belagavi. Presently he is a research scholar in instrumentation technlogy at PDA College of Engineering Gulbarga from Visveswarya Technological University Belagavi. He is also working as Assistant Professor in KBN University Gulbarga Karnataka since from 1998. He is attended many workshops induction programs and FDPs. He can be contacted at email: srinivask_1212@rediffmail.com.

**Dr. Channappa Bhyri** earned his B.E. degree in instrumentation technology from PDA College of Engineering, Gulbarga in the year 1990 and M.Tech. from IIT Roorkee in the year 1998. He has obtained his Ph.D. from SGGS Institute of Engineering and Technology Nanded, affliated to Sri Ramanand Teerth Marathawada University Nanded (MS). His areas of interest are biomedical signal and image processing. His total teaching experience is 33 years. He can be contacted at email: channubhyri@yahoo.com.