The Scientific Temper (2024) Vol. 15 (3): 2634-2643
E-ISSN: 2231-6396, ISSN: 0976-8653
Doi: 10.58414/SCIENTIFICTEMPER.2024.15.3.31
https://scientifictemper.com/

RESEARCH ARTICLE

RPL-eSOA: Enhancing IoT network sustainability with RPL and enhanced sandpiper optimization algorithm

Kavitha V.*, **Panneer Arokiaraj S.**

Abstract

The internet of things (IoT) encompasses extensive networks of interconnected devices, playing a crucial role in various applications. However, managing these networks presents significant challenges, particularly in cluster head selection, which is critical for energy efficiency and sustainability. To eradicate these challenges, this paper combines the capability of routing protocol for low-power and lossy networks (RPL) with an enhanced sandpiper optimization algorithm (e-SOA) to dynamically optimize network configurations. This combination, termed RPL-eSOA, improves energy management and extends network longevity while maintaining robust communication pathways. Through simulation and comparative analysis, RPL-eSOA demonstrates superior performance in enhancing network lifetime and operational efficiency compared to traditional methods. It achieved a 100% packet delivery ratio (PDR) and significantly reduced latency to 475 ms.

Keywords: Cluster head selection, Dynamic optimization algorithm, Energy harvesting, Internet of things, Network lifetime extension.

Introduction

The IoT represents a vast network of interconnected devices, ranging from simple sensors to complex systems, that communicate and exchange data with each other (Sundaravadivazhagan *et al*., 2021). This network enables a wide array of applications across various sectors, including healthcare, agriculture, and smart cities, enhancing operational efficiency and enabling real-time decision-making (Chataut *et al*., 2023). However, the expansion of IoT also brings challenges, particularly in managing these devices' energy consumption, as many operate on limited battery power (Mohd Aman *et al*., 2021). Efficient use of energy is crucial in maximizing the lifespan and functionality of IoT devices (Kumari *et al*., 2024).

Energy harvesting in IoT introduces a sustainable approach by allowing devices to gather energy from their environment, such as solar power, vibrations, or thermal differences (Kataria *et al*., 2024). This technique can significantly extend the battery life of devices and reduce the need for frequent replacements, which is especially beneficial in hard-to-reach or maintenance-intensive areas (Pramodhini *et al*., 2024). Nevertheless, energy harvesting alone does not fully solve the energy management issue, as energy availability can be inconsistent and depends heavily on environmental conditions (Tang *et al*., 2018). Therefore, there is a pressing need to manage this harvested energy judiciously, ensuring that IoT devices operate efficiently under varying energy conditions.

As IoT networks grow in scale, they become increasingly complex, encompassing thousands or even millions of nodes (Paolone *et al*., 2018). This scalability brings further challenges, particularly in maintaining network performance and managing the vast amounts of data transmitted between devices (Dhar Dwivedi *et al*., 2024). The complexity increases the demand for the network's energy resources, making efficient energy management and routing protocols essential (Gomez *et al*., 2023). Here, the necessity of optimization algorithms becomes apparent, as they can significantly enhance network management by optimizing the use of available energy and extending the overall network lifetime (Ahmad *et al*., 2024). These

PG and Research Department of Computer Science, Thanthai Periyar Government Arts and Science College (Affiliated to Bharathidasan University), Tiruchirappalli, India.

***Corresponding Author:** Kavitha V., PG and Research Department of Computer Science, Thanthai Periyar Government Arts and Science College (Affiliated to Bharathidasan University), Tiruchirappalli, India., E-Mail: vkavithaccwcs@gmail.com

How to cite this article: Kavitha, V., Arokiaraj, P. S. (2024). RPL-eSOA: Enhancing IoT network sustainability with RPL and enhanced sandpiper optimization algorithm. The Scientific Temper, **15**(3):2634-2643.

Doi: 10.58414/SCIENTIFICTEMPER.2024.15.3.31

Source of support: Nil

Conflict of interest: None.

Received: 10/07/2024 **Accepted:** 10/08/2024 **Published** : 10/09/2024

optimization techniques are crucial for sustaining large-scale IoT deployments, making them more practical and effective across various applications (Abdul *et al*., 2023).

Motivation and Problem Definition

The rapid growth of IoT networks has underscored a critical issue: the efficient management of energy resources across extensive networks of devices. While energy harvesting provides a pathway to replenish energy autonomously, the unpredictable nature of energy availability poses significant challenges. Devices often face periods of low energy availability, which can compromise their functionality and reliability. Additionally, the uneven distribution of energy resources across the network can lead to imbalances, where some nodes may have surplus energy while others are energy-starved.

The problem is further complicated by the mobility of nodes within certain IoT applications, such as wearable technologies or vehicle networks, which can lead to frequent changes in network topology. These dynamics require a robust mechanism to ensure that energy consumption and network performance are optimized under all conditions. Without effective management strategies, the potential of IoT systems to operate autonomously and sustainably is severely limited. Hence, there is a clear need for sophisticated optimization algorithms that can dynamically adjust to changing conditions and resource levels. Such algorithms must not only manage the energy efficiently but also ensure that the network remains robust and functional under varying operational demands. This necessitates the development of new models and strategies that can enhance the energy efficiency of IoT networks while maintaining high levels of performance and reliability.

Objectives and Scope of Research

This research aims to address the challenges outlined by proposing a novel optimization algorithm that effectively manages energy distribution and consumption in IoT networks. The following are the objectives of this research work:

- To develop a model that optimizes the selection of cluster heads in IoT networks.
- To ensure efficient energy use and prolong network longevity.
- To dynamically adapt changes in node energy levels and network topology.

Organization of the Paper

This paper is structured to provide a clear understanding of the proposed solution and its implications in IoT networks. It begins with section 2, which reviews related works to provide context and background, highlighting previous approaches and their limitations. Section 3 presents the proposed e-SOA, detailing its methodology, phases, and the specific mechanisms it employs for optimizing network efficiency. Section 4 discusses the implementation of the algorithm and evaluates its performance through various simulations. The results are analyzed to demonstrate the effectiveness of the e-SOA in extending network lifetime and enhancing energy efficiency. Finally, section 5 concludes the paper with a summary of the findings, contributions, and a discussion on future research directions.

Related Works

A clustering framework integrating intelligent algorithms with MCDM techniques was proposed, achieving notable improvements over existing models but limited by its assumption of two-dimensional node positions (Sahoo *et al*., 2024). A strategy for selecting the cluster head that improves energy metrics and node longevity by utilizing MDCNN and BCMO was devised by Jesi *et al*. (2024). A fuzzy logic-based cluster head selection algorithm was introduced, showing significant improvements in network lifetime and energy efficiency (Batra *et al*., 2024). SWARAM was developed using OOA for cluster head selection, improving packet delivery ratio and network lifetime by 10% (Somula *et al*., 2024). QOJ-LCH was proposed to enhance WSN lifespan and energy efficiency by balancing load distribution (Muthukkumar *et al*., 2023). An MDB-based clustering method was proposed, improving energy efficiency over PSO and LEACH methods (Chanpa *et al*., 2023). SOA was introduced for IoT networks, increasing network lifespan and throughput while reducing energy consumption (Sankar *et al*., 2023).

ESEERP, an enhanced smart energy efficient routing protocol, optimizes cluster head selection using a sail fish optimizer to enhance network longevity and energy efficiency in IoT environments, achieving significant performance improvements in terms of energy consumption, bandwidth utilization, and packet delivery ratios (Dogra *et al*., 2022).

Research Gap

From the above literature review, the following are the handicaps in existing research:

Static cluster head selection

Current methodologies often rely on static cluster head selections, failing to respond adequately to spontaneous changes in node status or environmental factors in real-world IoT applications.

Overlooked energy harvesting integration

There is a noticeable lack of comprehensive strategies that combine energy harvesting with dynamic cluster head selection and adaptive network reconfiguration to enhance energy efficiency and network longevity.

Scalability issues

Existing solutions often struggle to maintain efficiency and effectiveness when scaled up, highlighting the need for new optimization strategies designed to handle the complexities of large-scale, heterogeneous networks.

Methodology of RPL-eSOA

The integration of sensor networks into the IoT landscape necessitates advanced network management techniques to tackle the challenges of energy efficiency and network longevity. Traditional energy management and routing methods often fall short in addressing the dynamic and complex nature of IoT networks. This methodology introduces RPL-eSOA, a novel solution to optimize cluster head selection and adapt dynamically to changing network conditions. RPL-eSOA ensures efficient energy distribution and robust communication pathways. Ultimately, this approach enhances the sustainability and operational efficiency of IoT networks.

Inspired by the methodologies outlined in the foundational works on energy-efficient routing protocols (Dogra *et al*., 2022) and optimization algorithms in sensor networks (Sankar *et al*., 2023), e-SOA integrates advanced optimization techniques with a deep understanding of network dynamics. RPL-eSOA leverages the strengths of both RPL and e-SOA to enhance network performance and energy efficiency. RPL, known for its effective routing in low-power and lossy networks, forms a reliable and energy-efficient routing topology. e-SOA, on the other hand, optimizes cluster head selection based on a composite consideration of energy levels, node density, communication overhead, and real-time network feedback. This combination facilitates a dual-optimization framework where RPL dynamically adjusts routing paths based on the evolving topology and energy conditions optimized by e-SOA. Figure 1 outlines the systematic process of the e-SOA in optimizing IoT network configurations. The workflow is divided into several key phases: fitness evaluation, migration, attacking, and convergence check, leading to the final output of cluster heads and network configuration. Each phase is designed to iteratively refine and optimize the network, adapting dynamically to changing conditions and ensuring robust and energy-efficient operations.

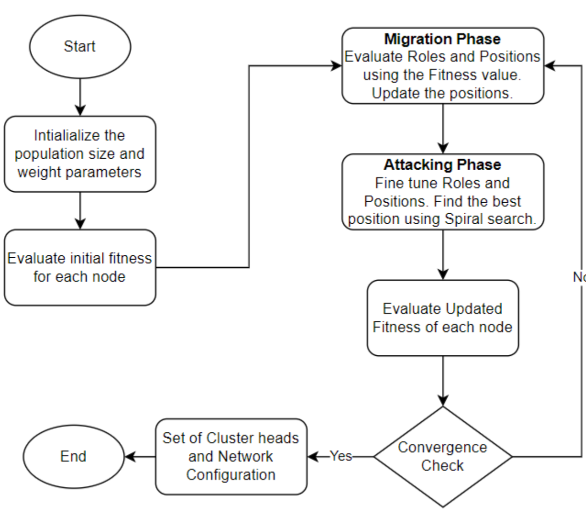

Figure 1: e-SOA workflow diagram

Fitness Function

In the proposed work, the fitness function is essential for evaluating and selecting the most suitable cluster heads in the IoT network. It ensures that nodes with higher energy efficiency, lower communication overhead, optimal node density, and better communication quality are chosen. This dynamic evaluation helps maintain network performance, extend network lifetime, and adapt to changing conditions in real time. The critical parameters for the fitness function are as follows:

Energy efficiency

Prioritizing nodes with higher residual energy for roles that demand greater energy expenditure.

Communication overhead

Minimizing the network resources expended in data transmission, thus conserving energy.

Node density

Selecting cluster heads in strategically less dense areas of the network to avoid bottlenecks.

Media access control (MAC) layer feedback

Incorporating real-time data on signal strength and error rates to adjust node roles dynamically.

Rank of node

Ranking the nodes in the IoT environment with help of RPL's expected transmission count (ETX).

Optimization Phases

The RPL-eSOA functions through two distinct operational phases, each designed to refine and optimize the network's efficiency and energy utilization. These phases are crucial for adapting to dynamic network conditions and achieving optimal configurations.

Migration phase

During this initial phase, nodes evaluate their current roles and positions within the network based on a predefined fitness function, optimizing both intra-cluster communication and inter-cluster routing paths facilitated by RPL. The nodes evaluate their current roles and positions within the network based on a predefined fitness function, F_i. The fitness function is computed for each node i as follows:

$$F_i = \omega_1 \cdot E_i + \omega_2 \cdot O_i + \omega_3 \cdot D_i + \omega_4 \cdot M_i + \omega_5 \cdot RPL_i \quad (1)$$

where:

$\omega_1, \omega_2, \omega_3, \omega_4, \omega_5$ are the weights assigned to energy efficiency, communication overhead, node density, and MAC layer feedback, respectively.

E_i, O_i, D_i, *and* M_i represent the energy level, communication overhead, node density, and MAC layer feedback of node i.

RPL_i could be a composite of ETX and node rank.

Nodes use this fitness score to determine their movement within the network, aiming to enhance their positioning relative to other nodes. This adjustment is aimed at achieving more efficient configurations that can handle network demands more effectively. The positions are updated according to the following rule:

$$P_{\text{new}} = P_{\text{current}} + \alpha \cdot \Delta P \tag{2}$$

Where P_{current} is the current position, ΔP represents the change in position derived from the fitness evaluations, and α is a scaling factor that moderates the speed and scale of position adjustments.

During the migration phase, the routing decisions made by the RPL protocol are influenced by the ongoing position and role adjustments. RPL adapts its destination-oriented directed acyclic graph (DODAG) construction based on the updated node positions and fitness scores, enhancing the routing efficiency and responsiveness to the changing network dynamics.

Attacking phase

This phase is a more targeted optimization process where nodes fine-tune their roles and positions based on local performance metrics. This phase involves a deeper analysis of the local neighborhood around each node to identify the most efficient positioning and role adjustments. Nodes perform a detailed search around their current best position P_{best}, described by a spiral search pattern that minimizes the potential for conflict and maximizes coverage efficiency:

$$P_{\text{new}} = P_{\text{best}} + \beta \cdot \text{spiral}(P_{\text{current}}) \tag{3}$$

where β is a scaling factor that controls the extent of movement towards the P_{best}, and spiral is a function that computes a spiral trajectory from the current position to explore nearby positions more thoroughly.

The transition from the migration phase to the attacking phase ensures that initial broad optimizations are refined through localized, detailed adjustments. This two-step process allows the e-SOA to dynamically adapt to changes within the IoT network, promoting optimal energy usage and enhanced network performance. The optimization loop continues until the network achieves stability in its configuration, as indicated by minimal changes in the positions and roles of the nodes.

Key Components

Each component of the e-SOA fitness function plays a crucial role in the holistic optimization of the network:

Energy component

The energy component plays a pivotal role in evaluating the suitability of nodes for the role of cluster heads based on their energy efficiency, ensuring that nodes with higher residual energy are prioritized, thereby enhancing the overall network sustainability. İt is denoted as E_i, for a node i, quantifies the residual energy of the node relative to its initial or maximum energy capacity. This measure is crucial as it directly impacts the node's ability to continue functioning effectively over time, especially in roles that require higher energy consumption, such as that of a cluster head. The energy component is calculated using the following formula:

$$E_i = \frac{Current_{\ Energy} \text{ of node } i}{Max_{\ Energy}} \tag{4}$$

Where:

- $Current_{\ Energy}$ of node i: This is the remaining energy level of node i at any given point during the network's operation.
- $Max_{\ Energy}$: Represents the maximum energy with which the node started, typically the battery capacity or the full energy level post a full charge.

The energy component is crucial for ensuring that the nodes selected as cluster heads can sustain prolonged operational periods, which is vital for maintaining continuous network functionality without frequent recharges or replacements. This consideration is especially important in remote or inaccessible deployment areas where maintenance and energy replenishment opportunities are limited.

Overhead component

The overhead component is vital for minimizing the communication overhead of each node in a sensor network. This optimization enhances network efficiency by prioritizing nodes with lower overheads as cluster heads, reducing overall resource consumption and boosting performance. Overhead component, O_i, for a node i, measures the communication overhead, which includes resources used for network connectivity, routing, and data transmission. Reducing these overheads it conserves energy and improves network longevity. Furthermore, it reduces network collision, and retransmission, optimizing routing and data processing across the network. The overhead component is represented as given in the following equation:

$$O_i = \frac{1}{1+\text{Comm Overhead of node } i} \tag{5}$$

where the communication overhead of node i involves its participation in data forwarding, routing, and other maintenance tasks.

Density component

To enhance the efficiency of calculating node density in sensor networks, especially for large-scale deployments where traditional methods may be computationally intensive. To overcome this, a novel approach is proposed. This method aims to quickly approximate node density using a combination of statistical methods and grid-based spatial partitioning. It helps to reduce the computational overhead typically associated with direct distance calculations for every node pair. The proposed method utilizes a grid-based spatial partitioning to approximate the local node density around a node i.

Grid formation

Divide the total area of the network (e.g. $100m\ X\ 100m$) into smaller grid cells of size ($s\ \times s$). The size s can be chosen based on the desired granularity and the average communication range of the nodes.

Node assignment to cells

Assign each node to a corresponding cell based on its geographical coordinates modified by the angle-based distance estimation (ABDE) (Hadded *et al.*, 2017). This step categorizes all nodes into their respective cells, allowing for a localized density calculation that more accurately reflects communication potentials. The following equation depicts the ABDE adjustment in the grid assignment:

$$D_{\text{effective}}(i,j) = \frac{r_i + r_j}{2} \times \cos(\theta_{ij}) \tag{6}$$

Utilize $D_{\text{effective}}$ to adjust node assignments, ensuring nodes that have better communication compatibility are grouped together, even if they are geographically distant.

Density calculation

For any node i, located in cell $C(x,y)$, the density D_i is calculated by counting the number of nodes in the surrounding cells, including $C(x,y)$ itself and its immediate neighbors. This provides an effective density measure that reflects the local clustering of nodes. The following equation represents the statistical calculation for density estimation:

$$D_i = \frac{\sum_{j=-1}^{1}\sum_{k=-1}^{1} N_{C(x+j,y+k)}}{9 \times s^2} \tag{7}$$

Where:

- $N_{C(x+j,y+k)}$ represents the number of nodes in the cell at position $(x+j, y+k)$.
- The denominator $9 \times s^2$ accounts for the total area of the $3\ \times 3$ grid cells considered around $C(x,y)$.

Incorporating density-aware adaptive clustering (DAAC)

Leverage the calculated densities to form clusters using the principles of DAAC. Nodes within high-density areas that also have high residual energy become candidates for cluster heads. This selection is dynamically adjusted based on changes in node density and energy levels. Nodes with a density score above a set threshold and sufficient energy are elected as cluster heads. Other nodes join the nearest cluster based on both geographic proximity and effective communication paths calculated using ABDE.

By incorporating ABDE into the grid assignment, the method ensures that distances reflect actual communication efficacy, enhancing accuracy in density calculations. Reduces computational complexity by avoiding direct pairwise distance calculations, using a simple summation over a limited number of grid cells. Adapting to large-scale and dynamic environments where nodes may move, updating the grid with node movements is straightforward and cost-effective. The representation of the grid cells around a central node C(x,y) is given below:

C(x–1,y+1)	C(x,y+1)	C(x+1,y+1)
C(x–1,y)	C(x,y)	C(x+1,y)
C(x–1,y–1)	C(x,y–1)	C(x+1,y–1)

Each cell represents the location of nodes relative to the central node C(x,y). The central cell contains the node whose density is being calculated. The surrounding cells include the immediate neighbors in all directions (north, northeast, east, southeast, south, southwest, west, northwest). This layout helps in calculating a comprehensive local density by considering the nodes in and around the central cell.

MAC layer feedback

The MAC layer feedback, denoted as M_i, for a node i, encompasses several key parameters that directly influence network performance. These parameters typically include:

- Signal strength (S_i): Represents the quality of the signal received by node i, which impacts its ability to communicate effectively within the network.
- Error rates (E_i): Quantifies the rate of errors in the packets received by the node, indicating the reliability of the communication channel.
- Channel occupancy (O_i): Measures the level of traffic or congestion in the communication channel used by node i, affecting its communication latency and throughput.

The MAC layer feedback component M_i in the fitness function of e-SOA is calculated using a weighted sum of the normalized values of these parameters:

$$M_i = w_s \cdot \frac{S_i}{S_{max}} + w_e \cdot \left(1 - \frac{E_i}{E_{max}}\right) + w_o \cdot \left(1 - \frac{O_i}{O_{max}}\right) \tag{8}$$

Where:

- w_s, w_e, and w_o are the weights assigned to the signal strength, error rates, and channel occupancy respectively, reflecting their relative importance in the network's operational context.
- S_{max}, E_{max}, and O_{max} represent the maximum observed values for signal strength, error rates, and channel occupancy, ensuring normalization of these metrics.

RPL metrics component

This component enhances the synergy between RPL's routing strategies and e-SOA's optimization techniques, leading to an adaptive and efficient network architecture. The primary purpose of including the RPL metrics component is to combine key routing metrics from RPL into the decision-making processes of e-SOA, particularly in cluster head selection and node positioning. This component utilizes RPL-specific metrics ETX, node rank, and link quality to adjust the fitness evaluations in e-SOA.

These metrics provide real-time insights into the routing efficiency and reliability, which are critical for maintaining effective communication pathways in low-power and lossy network environments.

$$RPL_i = w_{etx} \cdot \frac{1}{ETX_i} + w_{rank} \cdot \text{Rank}_i \quad (9)$$

where,

- ETX_i is the expected transmission count for node i, indicating the link quality.
- Rank_i is the node's rank within the RPL's DODAG, reflecting its relative position in the routing hierarchy.
- w_{etx} *and* w_{rank} are the weights assigned to ETX and rank metrics, respectively.

Convergence and Output

The convergence criteria and the output of the RPL-eSOA algorithm are critical elements that determine the effectiveness and efficiency of the network optimization process. Here, convergence is defined as the point at which further iterations do not yield significant improvements in the network's operational metrics, particularly the fitness scores of potential cluster heads. The convergence is evaluated based on a combination of factors that indicate stability and optimization in the network's configuration. It occurs when both the node optimizations by e-SOA and the routing decisions by RPL stabilize, indicating that further adjustments yield negligible improvements in network performance. Another criterion is the minimal improvement in the overall network fitness, defined by a small constant ϵ. If the improvement in network fitness between iterations falls below ϵ, the network is deemed to have reached its optimal state under current conditions. Mathematically, convergence can be expressed as:

$$\text{Converged if } \max(\Delta F_i,\ \Delta RPL_i) < \epsilon \quad (10)$$

where $\Delta F_i, \Delta RPL_i$ is the change in fitness score of any node *i* considered for a cluster head role between two consecutive iterations.

Once e-SOA converges, the output is a robust network configuration that includes:

Cluster head selection

A list of nodes designated as cluster heads, selected based on their final fitness scores. These nodes are expected to manage communication and data aggregation within their respective clusters efficiently.

Cluster formation

Based on the final positions and roles of the cluster heads, clusters are formed dynamically. Each node in the network is assigned to the nearest cluster head, forming a sub-network that optimizes local communication and reduces energy consumption.

Clusters are typically defined by a proximity metric, ensuring that each node is associated with the closest cluster head, minimizing the communication distance and, thus the energy required for transmission:

$$C_i = \{n_j \mid d(n_i, n_j) \le d(n_i, n_k)\ \forall\, k \ne j\} \quad (11)$$

where C_i is the cluster headed by node n_i, and $d(n_i, n_j)$ is the distance between node n_i and node n_j.

RPL-eSOA Algorithm

The e-SOA improves upon the traditional sandpiper optimization algorithm (SOA) by integrating RPL-specific metrics into its fitness function. This integration enhances routing efficiency and network performance. The e-SOA utilizes a two-phase optimization process: the migration phase adjusts node positions based on comprehensive fitness evaluations, while the Attacking Phase conducts detailed spiral searches for optimal cluster head selection. Additionally, e-SOA introduces a grid-based spatial partitioning method for efficient node density calculation, which dynamically makes cluster head selection. These enhancements enable e-SOA to adapt to changing network conditions, improve energy efficiency, and optimize overall IoT network performance. The proposed e-SOA algroithm is given below.

Enhanced Sandpiper Optimization Algorithm (e-SOA)

Inputs

- C: Set of all sensor nodes
- S: Sink location, acting as the root for RPL's DODAG
- E_i, O_i, D_i, M_i: Initial energy, communication overhead, node density and MAC layer feedback for node i

Outputs

- CH: Selected cluster heads

Parameters

- $\omega_1, \omega_2, \omega_3, \omega_4, \omega_5$: Weights for energy, overhead, density, and MAC layer feedback

Algorithm

Initialization

$\forall i \in C: Initialize(E_i, O_i, D_i, P_i)$

Set routing topology using RPL

Optimization loop

Migration Phase:

$\forall i \ \in C:$

P_i^{new} = Update based on F_i from previous iteration

Attacking phase:

$\forall i \ \in C:$

P_i = Spiral search around P_{best}

Fitness evaluation

$\forall i \ \in C:$

Calculate: $E_i, O_i, D_i, M_i, RPL_i$

F_i using equation 1.

Dynamic cluster head selection

Select $i \in C$ with max F_i as CH

#re-evaluating the selection based on changes in node positions and network dynamics.

Convergence check

Repeat steps 2-4 until stability or max iterations

#considering changes in the network topology due to node mobility.

End

Results and Discussion

The experiments were conducted based on the simulated IoT network setup that mimics real-world IoT operations, enabling a detailed analysis of the algorithm's effectiveness under varying conditions. The simulation was conducted using the Contiki-NG operating system and the Cooja simulator. Table 1 represents the parameter settings considered for the simulation. Each packet transmitted in the simulation had a payload size of 125 bytes. This size was chosen to balance the communication overhead and energy consumption. The performance of RPL-eSOA was compared against traditional algorithms such as ESEERP [35], RPL, PSO, ACO (Ant Colony Optimization), and SOA over multiple iterations to assess its efficiency and network lifetime improvement. Figure 2 depicts the screenshot taken from cooja simulator.

To quantify the improvements offered by e-SOA, the following metrics were tracked:

Total Energy Consumption

The total energy consumption E_{total} of the network during the simulation can be calculated by summing up the energy used by all nodes over the duration of the simulation. This is given by:

$$E_{\text{total}} = \sum_{i=1}^{N} \sum_{t=1}^{T} E_{i,t} \tag{12}$$

where N is the total number of nodes, T is the total number of time steps or rounds in the simulation, and $E_{i,t}$ is the energy consumed by node i at time t.

Number of Live Nodes Over Time (Network Lifetime)

To track the number of operational (live) nodes over time, the function $L(t)$ can be defined as:

$$L(t) = \text{Count of nodes } i \text{ such that } E_i(t) > 0 \tag{13}$$

where $E_i(t)$ is the remaining energy of node i at time t. This function counts the number of nodes that still have positive energy reserves at each time step. This helps to identify the network lifetime.

Data Transmission Efficiency

Data transmission efficiency is assessed by measuring the ratio of successfully delivered data packets to the total data packets sent. This can be quantified using the efficiency ratio η, calculated as follows:

$$\eta = \frac{\text{Number of successfully delivered packets}}{\text{Total number of packets sent}} \times 100\% \tag{14}$$

where *"successfully delivered packets"* are those that reach the designated sink node without being lost due to energy

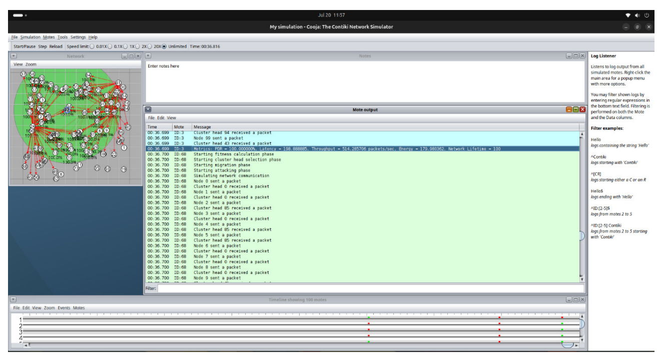

Figure 2: Simulation window for 100 nodes

Table 1: Parameter settings for RPL-eSOA simulation

Parameter	*Value*	*Description*
Total nodes	100	The total number of nodes within the network
Simulation area	100m x 100m	The physical dimensions of the area for the simulation
Cluster heads	10	The number of cluster heads selected based on fitness
Communication threshold	10 meters	Maximum distance for direct communication
Initial energy range	50 to 100 units	Initial energy is assigned randomly to each node
Transmission range	1 to 10	Number of transmissions for each node
Route updates range	1 to 10	Number of route updates performed by each node
Signal strength range	0.5 to 1.0	Strength of the signal for communications
Error rate range	0 to 0.1	Probability of error in transmissions
Channel occupancy range	0 to 1	Level of channel occupancy
Fitness weights	Energy: 0.4, Overhead: 0.1, Density: 0.3, MAC Feedback: 0.2	Weights used to calculate node fitness

Table 2: Efficiency comparison of RPL-eSOA with traditional algorithms

Algorithm	*PDR (%)*	*Latency (ms)*	*Throughput (Mbps)*	*Energy (joules)*	*Network lifetime (nodes alive)*
PSO	92	966.67	1.03	1.5	96
SOA	90	975.00	1.02	2.0	90
ACO	90	975.00	1.02	2.0	90
ESEERP [35]	96	-	0.52	0.5	98
RPL-eSOA	100	475.00	2.22	0.2	100

depletion or other network failures, and *"total number of packets sent"* is the total packets attempted to be sent by all nodes.

Analysis

The performance analysis of different algorithms is presented in Table 2. This table demonstrates the effectiveness of the proposed IRPL-sSOA across multiple key metrics compared to other methods. The proposed RPL-eSOA algorithm demonstrates substantial improvements over traditional methods by dynamically optimizing network configurations, which effectively balances energy consumption and enhances network performance. Specifically, RPL-eSOA achieves a 100% packet delivery ratio (PDR), significantly reduces latency to 475 ms, and increases throughput to 2.22 Mbps. These results indicate that RPL-eSOA ensures reliable and efficient data transmission, which is crucial for real-time applications in IoT networks.

Furthermore, the energy efficiency of RPL-eSOA is evident, with an energy consumption of only 0.2 joules, which is substantially lower than that of other algorithms like PSO, SOA, and ACO. This efficient energy management translates into a prolonged network lifetime, with all 100 nodes remaining operational throughout the simulation.

One of the primary advantages of RPL-eSOA is its dual-phase approach to optimization, comprising the migration and attacking phases. This dual-phase strategy ensures that both broad and localized adjustments are made to improve network configurations. The migration phase allows nodes to reposition themselves based on a comprehensive fitness function that evaluates energy levels, communication overhead, and node density. This phase ensures that nodes are positioned optimally to minimize energy consumption and enhance communication efficiency. The attacking phase then fine-tunes these positions through localized adjustments, further optimizing the network's performance by refining node roles and their spatial distribution.

This comprehensive approach not only extends the operational lifespan of IoT networks but also enhances their reliability and scalability, making RPL-eSOA a superior choice for diverse IoT applications, particularly those deployed in remote or maintenance-intensive environments. The comparative analysis given in Table 1 is represented in the chart as depicted in Figures 3 and 4. İt highlights significant differences in performance metrics.

Algorithm Complexity

The computational complexity of the proposed RPL-eSOA algorithm is an essential factor in assessing its practical applicability and efficiency in sensor network environments. The complexity of e-SOA derives primarily from its iterative process involving fitness evaluation, cluster head selection, and the dynamic adjustment of network configuration. Each of these components contributes to the overall computational demand of the algorithm.

Fitness evaluation

The fitness of each node is calculated based on multiple parameters, including energy, overhead, density, and MAC layer feedback. The complexity for calculating the fitness of one node is O(1), as it involves constant time arithmetic

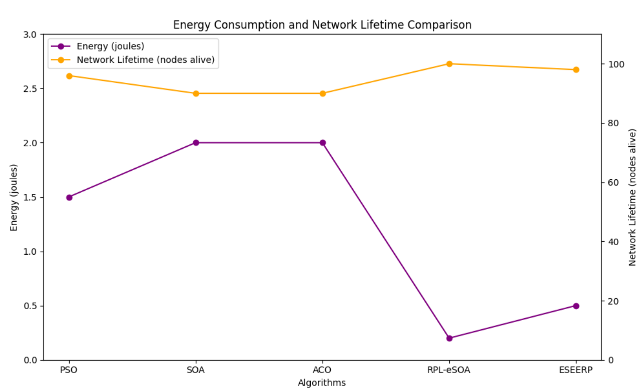

Figure 3: Comparative results of energy consumption and network lifetime

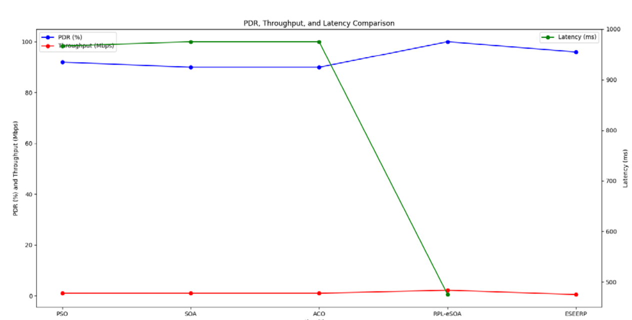

Figure 4: Comparative results of PDR, latency and throughput

operations. Therefore, the complexity for evaluating all n nodes in the network is O(n).

Cluster head selection

After calculating fitness scores, the algorithm selects cluster heads based on these scores. The selection process involves sorting the fitness scores to identify the top-performing nodes as cluster heads. Sorting typically has a complexity of $O(n \log n)$.

Cluster formation and optimization phases

Post cluster head selection, e-SOA adjusts node positions and refines cluster boundaries based on the migration and attacking phases, which involve re-evaluating node positions and fitness. This can be seen as a form of iterative optimization where each iteration's complexity depends on the number of nodes and their respective cluster members. Assuming each node recalculates its position relative to k nearest neighbors, the complexity for this step in each iteration can approximate $O(nk)$.

Convergence check

At each iteration, the algorithm checks if the change in the configuration has stabilized (based on predefined thresholds). This check is $O(n)$, as it requires a pass through each node's current and previous fitness scores.

Overall complexity

Combining these factors, the per-iteration complexity of e-SOA can be estimated as $O(n \log n + nk + n)$. If the algorithm converges after t iterations, the total complexity is $O(t(n \log n + nk + n))$. The dominant term in this complexity expression is typically $O(n \log n)$, due to the sorting operation for selecting cluster heads. However, the factor k (the average number of neighbors considered in local optimization phases) and the number of iterations t also play significant roles in scaling the complexity, especially in dense networks or under dynamic conditions where multiple iterations are needed to achieve convergence.

The derived complexity indicates that while e-SOA is efficient for moderate-sized networks, its scalability to very large sensor networks requires careful consideration of the parameters k and t. Optimizations in the algorithm's design, such as limiting the number of iterations or reducing the neighborhood size k for local optimizations, can help manage and potentially reduce the computational load, making e-SOA more practical for larger deployments.

Conclusion

The RPL-eSOA methodology successfully addresses the challenges of energy efficiency and reliable routing in IoT networks. By combining RPL with the e-SOA, this study has demonstrated significant improvements in network performance. The proposed algorithm achieved a 100% PDR and reduced latency to 475 ms, indicating efficient and reliable data transmission. Additionally, the energy consumption was minimized to 0.2 joules, substantially enhancing the network's operational lifespan by keeping all 100 nodes active throughout the simulation. These results highlight the effectiveness of RPL-eSOA in optimizing energy usage and maintaining robust communication pathways, making it a valuable solution for sustainable IoT deployments in various applications. In the future, the potential for applying e-SOA in real-world scenarios promises substantial benefits for IoT deployments, particularly in smart cities and industrial IoT applications, where managing energy effectively is crucial to operational success and sustainability.

References

Abdul-Qawy, A., Hamed, A. S., Alduais, N. A. M., Saad, A.-M. H. Y., Taher, M. A. A., Nasser, A. B., Saleh, S. A. M., & Khatri, N. (2023). An enhanced energy efficient protocol for large-scale IoT-based heterogeneous WSNs. Scientific African, 21, e01807.

Ahmad, R., Alhasan, W., Wazirali, R., & Aleisa, N. (2024). Optimization algorithms for wireless sensor networks node localization: An overview. IEEE Access.

Batra, P. K., & Vamsi, P. R. (2024). Fuzzy logic-based cluster head selection method for enhancing wireless sensor network lifetime. International Journal of Performability Engineering, 20(2).

CHANPA, R., JAMALI, M. A. J., HATAMLOU, A., & ANARI, B. (2023). Cluster head selection algorithm on the basis of mass defense of bees in IoT. Economic Computation & Economic Cybernetics Studies & Research, 57(3).

Chataut, R., Phoummalayvane, A., & Akl, R. (2023). Unleashing the power of IoT: A comprehensive review of IoT applications and future prospects in healthcare, agriculture, smart homes, smart cities, and industry 4.0. Sensors, 23(16), 7194.

Dhar Dwivedi, A., Singh, R., Kaushik, K., Mukkamala, R. R., & Alnumay, W. S. (2024). Blockchain and artificial intelligence for 5G-enabled Internet of Things: Challenges, opportunities, and solutions. Transactions on Emerging Telecommunications Technologies, 35(4), e4329.

Dogra, R., Rani, S., Kavita, Shafi, J., Kim, S., & Ijaz, M. F. (2022). ESEERP: Enhanced smart energy efficient routing protocol for internet of things in wireless sensor nodes. Sensors, 22(16), 6109.

Gomez, D. L., Montoya, G. A., Lozano-Garzon, C., & Donoso, Y. (2023). Strategies for assuring low latency, scalability and interoperability in edge computing and TSN networks for critical IIoT services. IEEE Access, 11, 42546-42577.

Hadded, M., Muhlethaler, P., Laouiti, A., & Saidane, L. A. (2017). A novel angle-based clustering algorithm for vehicular ad hoc networks. In Vehicular Ad-Hoc Networks for Smart Cities: Second International Workshop (pp. 27-38). Springer Singapore.

Jesi, P. M., Daniel, V. A. A., Rajagopal, R., & Femila, L. (2024). Cluster head selection using multi-dilation convolutional neural network optimized with BCMO for IoT networks. IETE Journal of Research, 1-9.

Kataria, A., & Rani, S. (2024). Advancements in energy harvesting: Design and techniques utilizing IoT for sustainable power generation. In Emerging Materials, Technologies, and Solutions for Energy Harvesting (pp. 213-233). IGI Global.

Kumari, P., & Bargavi, S. M. (2024). Sustainability by efficiency:

Energy considerations in IoT device development. Asian Journal of Basic Science & Research, 6(1), 35-43.

Sahoo, L., Sen, S. S., Tiwary, K., Moslem, S., & Senapati, T. (2024). Improvement of wireless sensor network lifetime via intelligent clustering under uncertainty. IEEE Access, 12, 25018-25033. https://doi.org/10.1109/ACCESS.2024.3365490

Mohd Aman, A. H., Shaari, N., & Ibrahim, R. (2021). Internet of things energy system: Smart applications, technology advancement, and open issues. International Journal of Energy Research, 45(6), 8389-8419.

Muthukkumar, M. S., & Diwakaran, S. (2023). Efficient load balancing in WSN using quasi–oppositional based Jaya optimization with cluster head selection. International Journal of Computer Network and Information Security (IJCNIS), 15(2), 85-96.

Paolone, G., Iachetti, D., Paesani, R., Pilotti, F., Marinelli, M., & Di Felice, P. (2022). A holistic overview of the internet of things ecosystem. IoT, 3(4), 398-434.

Pramodhini, R., Jagadheeswaran, R., Dhuriata, P., & Vivek, S. (2024). Optimizing environmental sustainability: Analyzing IoT-enabled energy harvesting systems. In 2024 Third International Conference on Distributed Computing and Electrical Circuits and Electronics (ICDCECE) (pp. 1-7). IEEE.

Sankar, S., Ramasubbareddy, S., Dhanaraj, R. K., Balusamy, B., Gupta, P., Ibrahim, W., & Verma, R. (2023). Cluster head selection for the internet of things using a sandpiper optimization algorithm (SOA). Journal of Sensors, 2023(1), 3507600.

Somula, R., Cho, Y., & Mohanta, B. K. (2024). SWARAM: Osprey optimization algorithm-based energy-efficient cluster head selection for wireless sensor network-based Internet of Things. Sensors, 24(2), 521.

Sundaravadivazhagan, B., Subashini, B., & Ashik, M. (2021). A review on Internet of Things (IoT): Security challenges, issues and the countermeasures approaches. Psychology And Education, 58(2), 6544-6560.

Tang, X., Wang, X., Cattley, R., Gu, F., & Ball, A. D. (2018). Energy harvesting technologies for achieving self-powered wireless sensor networks in machine condition monitoring: A review. Sensors, 18(12), 4113.