Journal of Optimization in Soft Computing (JOSC)
Vol. 3, Issue 3, pp: (18-29), Autumn-2025
Journal homepage: https://sanad.iau.ir/journal/josc

Research paper

A QoS-Aware Multi-Criteria Routing Approach Based on Grey Wolf Optimization Algorithm in Internet of Things Networks

Niloufar Hashemi[1] and Pouya Khosravian Dehkordi[1,*]

1. Department of Computer Engineering, ShK.C., Islamic Azad University, Shahrekord, Iran.

Article Info

Article History:
Received: 2025/08/13
Revised: 2025/09/14
Accepted: 2025/09/26

DOI: 10.82553/josc.2025.202508131214764

Keywords:
Internet of Things, Routing, Quality of Service, Multi-Criteria Algorithm, Grey Wolf Optimization, Normalization

**Corresponding Author's Email Address: drkhosravian@iau.ac.ir*

Abstract

The Internet of Things (IoT) is an information architecture based on the Internet that fosters interaction between goods and services in a secure and reliable environment. Several critical challenges exist in the IoT ecosystem, including security, scalability, availability, interoperability, performance, and big data analytics. Among these, routing remains a fundamental challenge in IoT environments. Various routing approaches have been proposed by researchers, each with its own advantages and limitations. To address some of these limitations, the present study proposes a multi-criteria QoS-aware routing approach using Grey Wolf Optimization (GWO) in IoT networks. Evaluation metrics for the proposed method include waiting time, packet loss rate, CPU utilization rate, network usage time, resource utilization rate, and response time. The evaluation results indicate that the proposed approach outperforms comparable methods in terms of reducing waiting time and packet loss. Additionally, the network lifetime achieved by the proposed method is superior to that of the compared approaches. Furthermore, analysis of CPU and resource usage confirms the higher efficiency of the proposed method. The experimental results demonstrate that the proposed method is better than other methods.

1. Introduction

The Internet of Things (IoT) fundamentally refers to a network of interconnected devices, in which various objects can communicate with each other via computing resources and Internet-based connections. An object in such networks may include any entity equipped to collect data, control processes, or perform remote communication. Examples of such objects include automobiles, household appliances, medical equipment, power grids, industrial tools, and construction machinery. IoT consists of a vast number of interconnected entities and is considered a pivotal stage in the evolution of cyberspace [1]. Two core characteristics define objects that qualify as part of the IoT ecosystem.

In recent years, both the volume of IoT data and its utilization have increased exponentially. This surge creates a critical need for efficient algorithms capable of transmitting data optimally from source to destination. Routing algorithms serve this purpose by identifying the most efficient paths between nodes, providing a practical solution for optimal data transmission. Typically, these algorithms traverse a graph, beginning at a source node and exploring neighboring nodes until reaching the destination—often aiming to minimize hop count or cumulative path weight.

Routing protocols are classified based on their decision-making mechanism into proactive, reactive, and hybrid protocols [2]. Proactive protocols maintain up-to-date routing tables at all times; reactive protocols generate routes on demand; and hybrid protocols combine features of both strategies. Reactive protocols are better suited for dynamic networks due to optimized bandwidth usage, whereas proactive protocols are more appropriate for static environments. Furthermore, routing protocols in IoT can be categorized into data-centric, hierarchical, location-based, and QoS-aware protocols. Data-centric protocols rely on attribute-based naming rather than globally unique identifiers. Hierarchical protocols organize networks into clusters managed by cluster head nodes. Location-based algorithms utilize geographic positioning to achieve energy-efficient routing. QoS-aware protocols handle multi-band data requests with high accuracy [3]. Energy efficiency is paramount in IoT networks due to limited battery life, directly affecting network performance and lifespan. Optimized routing between nodes is one of the most effective strategies for energy conservation. Routers coordinate network operations, aggregate sensor data, filter redundant packets, reduce routing complexity, and improve scalability. Moreover, effective routing distributes workloads across member nodes, balancing energy and memory usage.

Fault tolerance is essential in IoT networks, particularly at the hardware and communication layers. Traditional methods rely on software-based fault detection, but these have limitations. Novel approaches aim to enhance the IoT framework by reducing the number of devices involved and improving power management [4].

IoT networks enable communication between devices, humans, and other entities. Many IoT applications require message relay through intermediate nodes, sometimes using unicast to minimize bandwidth usage. Routing enables the creation of multiple paths, allowing the source to forward packets along the most suitable route. Centralized hubs for traffic management are often complex and slow, which has led to the adoption of distributed routing protocols. Existing studies have proposed distributed protocols incorporating geographic information and service-based metrics to select optimal transmission paths. A variety of unicast and multicast routing protocols exist to facilitate packet delivery. However, multicast approaches may suffer from high overhead, latency, reduced network lifetime, redundancy, and increased traffic. Some methods employ greedy or face routing protocols to mitigate these challenges, though they may result in longer paths or routing loops.

To address these limitations, this study introduces a novel multi-criteria routing method based on the Multi-Objective Grey Wolf Optimization (MOGWO) algorithm. The proposed approach is designed to improve route classification and efficiency while reducing routing time through intelligent path selection. The MOGWO algorithm evaluates routes using a fitness function that incorporates multiple QoS parameters, offering a comprehensive solution to optimize IoT routing. This represents the main innovation of the current study. The remainder of the paper is organized as follows: Section 2 reviews relevant literature, focusing on IoT architecture, challenges, and the routing problem. Section 3 evaluates and compares previously proposed routing strategies. Section 4 concludes the study and outlines potential future research directions.

2. Literature Review

One of the fundamental challenges in Internet of Things (IoT) networks lies in ensuring the efficient transmission of information from mobile nodes to base stations and identifying the most effective method for this transmission. The selection of an optimal path is influenced by multiple factors, including energy consumption, response time and latency, data transmission accuracy, and other critical network performance parameters. Broadly, the process of transmitting data between the observation point and the base station is referred to as routing. From one perspective, routing specifies the mechanism for delivering data among sensor nodes in an IoT network; from another, it serves as an optimization process for communication between sensors and base stations.

A simplistic approach would involve each sensor node transmitting its data directly to the base station. However, this single-hop strategy is highly energy-inefficient, particularly for nodes located at greater distances from the base station, as it accelerates battery depletion and consequently shortens the overall network lifetime [5]. The problem becomes even more critical in large-scale deployments or scenarios involving mobile sensors that may continuously move away from the base station.

To overcome these limitations, multi-hop communication with short-range transmissions is commonly adopted between mobile sensors and base stations. This method not only conserves energy but also minimizes wireless interference among sensor nodes competing for shared channels [6], [7].

In recent years, distributed computing in wireless networks has gained significant attention, especially with the rapid growth of IoT. Within this paradigm, objects are capable of independently processing, storing, and exchanging data. Instead of transmitting raw information over energy-intensive wireless links, only processed outcomes are shared, thereby significantly lowering both energy usage and communication latency. A recently introduced approach leverages local online processing to reduce data volume and communication costs, which in turn helps minimize bandwidth usage, energy expenditure, latency, and overall traffic in IoT networks [8],[25].

In general, routing in IoT refers to the logical determination of paths through which network traffic is directed toward its intended destination. This typically involves the forwarding of messages—sometimes via intermediate nodes—from a source to a target node. Numerous routing protocols have been developed for IoT, with core objectives that include:

- Maximizing data transmission efficiency,
- Minimizing network congestion,
- Extending node lifetime, and
- Reducing redundancy.

To address the shortcomings of conventional techniques, this study introduces a multi-objective path optimization approach based on the Multi-Objective Grey Wolf Optimizer (MOGWO). The proposed method integrates multiple decision-making parameters to determine the most effective route for packet forwarding, thus improving network performance and efficiency.

Routing essentially involves two major tasks: path estimation and packet forwarding. Residing within the network layer of communication software, routing algorithms determine the outgoing interface for each incoming packet. Different routing algorithms impose various constraints and trade-offs on network resources, meaning that path computation is inherently dependent on dynamic system parameters.

In IoT environments, long-range direct communication between nodes may be possible; however, it is restricted by factors such as radio transmission capacity and node bandwidth. In the absence of a direct path, nodes rely on multi-hop relaying through neighboring nodes—often without user awareness. In such cases, packets are dynamically forwarded across designated intermediate nodes until they successfully reach their destination [9],[26].

3. Related Work

This section reviews prior studies focusing on optimized routing techniques in IoT networks, evaluating the advantages and disadvantages of each, and providing a comparative overview. Due to the inherently dynamic topology of IoT environments, energy consumption remains high and network reliability low. Numerous routing protocols have been developed to tackle this issue.

Debroy et al. [10] proposed a multi-channel routing solution for IoT communication, assuming the availability of radio environment knowledge via spectrum sensors. Their approach generates a spectrum map used to determine optimal multi-hop routes and corresponding channels at each stage. By applying evolutionary game theory, their model supports parallel communications between devices, identifying optimal point-to-point paths and transmission power levels.

Priyanka et al. [11] utilized IoT for secure healthcare data sharing between remote hospitals. To ensure efficient delivery, they applied Ant Colony Optimization (ACO), considering factors such as path length, discovery time, and local performance thresholds. The enhanced ACO led to improvements in energy use, delay, and throughput.

Niu et al. [12] presented a routing solution aimed at increasing network lifetime. Their design includes all phases from node deployment and architecture definition to environmental awareness. Their energy-aware clustering protocol minimizes packet delivery time and processing load at the node level while optimizing energy consumption and boosting throughput.

Kang et al. [13] introduced a hybrid centralized-distributed Reinforcement Learning-based routing scheme incorporating multiple routing metrics. Their experiments revealed that centralized learning, due to its faster convergence, is more suited to dynamic networks, outperforming traditional strategies in flexibility and responsiveness.

Yang et al. [14] addressed limitations in geographic routing caused by outdated distribution patterns. Their proposed method uses distributed unicast routing with three stages: intermediary node selection, loop elimination and path correction, and parameter updates. Simulations showed that their method significantly reduces the number of transmissions and end-to-end delays.

Mahmoud Anam et al. [15] proposed a highly efficient multi-objective routing method that employs optimized transmission control to enhance energy-aware routing in IoT. While the method greatly improves network bandwidth utilization, it is less effective in heterogeneous networks.

Chemodanov et al. [16] developed an AI-driven geographic routing strategy using satellite imagery and deep learning to avoid local optima. Though the algorithm optimizes worst-case paths, it fails to account for symmetric paths in its design.

Wang et al. [17] presented a source-routing algorithm based on on-demand packet reduction in IoT. Unlike traditional schemes, each packet only carries a compact representation of the route. This reduces overhead while preserving full routing context, enabling effective transmission with low latency.

Agrawal et al. [18] proposed a fuzzy-based unequal clustering technique that improves network lifespan and load balancing. Unlike earlier approaches, their system accounts for residual energy, distance to the base station, node degree, and centroid position. Comparative evaluations confirm the superiority of their method.

Chia [19] conducted a comprehensive performance analysis of various routing protocols within 6LoWPAN networks, including RPL, AODV, and LOADng. Utilizing the Cooja simulator, the study assessed power consumption metrics such as CPU, listen, and total power across different network topologies.

Findings indicated that RPL outperformed the other protocols in terms of energy efficiency, making it a suitable choice for low-power IoT applications. Alsodairi et al. [20] evaluated the performance of RPL and its variants, ORW and ORPL, in intermittent energy harvesting IoT scenarios. Their study highlighted that ORPL slightly outperformed RPL in terms of packet reception rates, suggesting its potential for applications where energy availability is inconsistent.

Tarif et al. [21] introduced an enhanced RPL routing mechanism by integrating the Tabu Search optimization algorithm. This approach aimed to improve link stability and reduce energy consumption by optimizing parent-child selection processes. Simulation results demonstrated significant improvements in link stability and packet delivery ratios, indicating the effectiveness of the proposed method in resource-constrained IoT networks.

Homaei [22] proposed the Learning Automata-based Load-Aware RPL (LALARPL), an enhancement to the standard RPL protocol that incorporates learning automata to optimize traffic distribution.

4. Proposed Methodology

In the proposed routing optimization framework, each IoT entity is modeled as a variable within the network topology. While these entities are inherently discrete or integer-based, they are represented as continuous variables to enable mathematical modeling and optimization. Each entity explores the solution space by evaluating its position with respect to key Quality of Service (QoS) parameters, including bit error rate (BER), throughput, bandwidth, and reliability. These parameters are expressed in the form of multidimensional vectors for systematic evaluation.

The framework initializes node positions through a uniform random distribution and assigns initial velocities to ensure diversity in the search space. QoS parameters are normalized to standardize their scales prior to optimization. To further enhance computational efficiency, a feature correlation filtering process is applied. Using Pearson's correlation coefficient, redundant or weakly correlated metrics are removed, retaining only the most influential features for route optimization.

4.1. Data Preprocessing

datasets must first be correlated to extract meaningful information into a unified model while eliminating irrelevant or redundant attributes. This correlation process can be carried out using various statistical approaches, all of which fundamentally rely on the computation of correlation coefficients among variables.

In this study, the Pearson correlation coefficient is employed to quantify the strength and direction of linear relationships between quantitative variables. The obtained coefficients are subsequently utilized as input parameters for the optimal path selection model. The Pearson correlation coefficient ρ for a population is mathematically defined in Equation (1):

$$\rho_{X,Y} = \frac{cov(X,Y)}{\sigma_X \sigma_Y} = \frac{E[(X-\mu_X)(Y-\mu_Y)]}{\sigma_X \sigma_Y} \quad (1)$$

Where:

- Cov(X,Y) is the covariance of variables X and Y,
- σ_X and σ_Y are the standard deviations of X and Y,
- μ_X and μ_Y represents the expected value operator.

For a sample of *n* paired values, the sample Pearson correlation coefficient is calculated using Equation (2):

$$r = \frac{\sum_{i=1}^{n}(X_i-\bar{X})(Y_i-\bar{Y})}{\sqrt{\sum_{i=1}^{n}(X_i-\bar{X})^2}\sqrt{\sum_{i=1}^{n}(Y_i-\bar{Y})^2}} \quad (2)$$

The above equation can be expressed in a more compact form using Equation (3) [23].

$$r = \frac{i}{n-1}\sum_{i=1}^{n}\left(\frac{X_i-\bar{X}}{s_X}\right)\left(\frac{Y_i-\bar{Y}}{s_Y}\right) \tag{3}$$

In this context, each of the values of $\bar{X}$, $\bar{Y}$, s_X and s_Y is defined respectively by Equations (4), (5), (6), and (7) [23].

$$\bar{X} = \frac{1}{n}\sum_{i=1}^{n} X_i \tag{4}$$

$$\bar{Y} = \frac{1}{n}\sum_{i=1}^{n} Y_i \tag{5}$$

$$s_X = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(X_i-\bar{X})^2} \tag{6}$$

$$s_Y = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(Y_i-\bar{Y})^2} \tag{7}$$

After establishing the significance and direction of relationships between variables, it is essential to assess the magnitude of the correlation. Depending on the specific application, several classification schemes have been proposed to interpret the strength of association between two variables. These classifications are applied to conduct data correlation analysis and to remove irrelevant or non-informative attributes from large datasets. Through this correlation process, a considerable portion of evaluations related to non-essential variables can be effectively filtered, thereby improving the efficiency and accuracy of subsequent analyses.

4.2. Multi-Objective Grey Wolf Optimization

Multi-objective optimization is carried out using the alpha, beta, and delta wolves. The alpha wolf is considered the primary leader of the algorithm, while the beta and delta wolves also contribute to the process. The remaining wolves are regarded as followers. In this study, the optimization process is modeled using equations (8), and (9):

$$\vec{D} = \left|\vec{C}.\vec{X}_p(t) - \vec{X}(t)\right| \tag{8}$$

$$\vec{X}(t+1) = \vec{X}_p(t) - \vec{A}.\vec{D} \tag{9}$$

where:

- $\vec{A}$ *and* $\vec{C}$ are coefficient vectors
- $\vec{X}_p$ is the position vector of the **prey**
- $\vec{X}$ is the position vector of each wolf, and
- t denotes the iteration number

The coefficient vectors $\vec{A}$ and $\vec{C}$ are computed using Equation (10) [24].

$$\vec{A} = 2\vec{a}.\vec{r}_1 - \vec{a} \quad ,\vec{C} = 2\vec{r}_2 \tag{10}$$

In this context, the components of $\vec{A}$ decrease linearly from 2 to 0 over successive iterations, and $\vec{r}_1$ and $\vec{r}_2$ are random vectors within the range [0, 1]. Grey wolves are capable of estimating the position of the prey. However, during the initial stages of the search, there is no prior knowledge of the prey's location. It is assumed that the alpha, beta, and delta wolves have better prior estimates of the prey's position. The positions of these three candidate solutions are determined using the equations presented below [24].

$$\begin{aligned}
&\vec{D}_\alpha = \left|\vec{C}_1.\vec{X}_\alpha - \vec{X}\right| \\
&\vec{D}_\beta = \left|\vec{C}_2.\vec{X}_\beta - \vec{X}\right| \\
&\vec{D}_\delta = \left|\vec{C}_3.\vec{X}_\delta - \vec{X}\right| \\
&\vec{X}_1 = \vec{X}_\alpha - \vec{A}_1(\vec{D}_\alpha) \\
&\vec{X}_2 = \vec{X}_\alpha - \vec{A}_2(\vec{D}_\beta) \\
&\vec{X}_3 = \vec{X}_\alpha - \vec{A}_3(\vec{D}_\delta) \\
&\vec{X}(t+1) = \frac{(\vec{X}_1 + \vec{X}_2 + \vec{X}_3)}{3}
\end{aligned} \tag{11}$$

In fact, the alpha, beta, and delta wolves estimate the position of the prey, while the remaining wolves update their positions randomly around the estimated prey location. The positions of the top three candidate solutions are continuously preserved throughout the optimization process. The two-dimensional and three-dimensional representations of the wolves' position vectors and their possible future movements are illustrated in Figure 1. When the prey is surrounded by the wolves and becomes stationary, the attack phase is initiated under the leadership of the alpha wolf. The overall behavior of this process is modeled by the reduction of the coefficient vector $\vec{A}$. Since the vector $\vec{A}$is randomly generated within the range [−2a, 2a], a decrease in aa also leads to a proportional decrease in the magnitude of vector $\vec{A}$. If $|\vec{A}|$<1, the alpha wolf (along with the other wolves) moves closer to the prey, whereas if $|\vec{A}|$>1, the wolf diverges from the prey (and the others), enabling exploration of the search space.

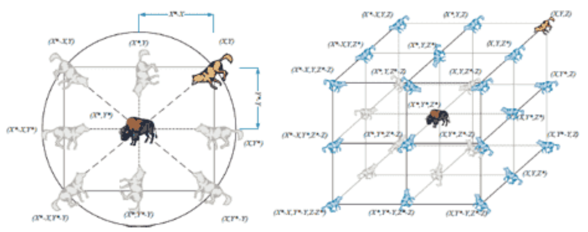

Figure 1. Grey Wolf Attack Mechanism [24]

The multi-objective Grey Wolf Optimizer requires that all wolves update their positions based on the positions of the alpha, beta, and delta wolves. The search process is exactly the inverse of the attack process: during the search phase, wolves disperse to track the prey ($|\vec{A}|>1$), while after detecting the prey, they converge during the attack phase ($|\vec{A}|<1$). This process is referred to as divergence in exploration – convergence in exploitation.

Role of vector $\vec{C}$:The vector $\vec{C}$ is considered to represent natural obstacles that slow down the wolves' approach toward the prey. It assigns weight to the prey, making it appear less reachable. Unlike $\vec{A}$, this vector does not decrease linearly from 2 to 0. Thus, the algorithm can be summarized in the following:

- The fitness of all candidate solutions is evaluated, and the top three are selected as the alpha, beta, and delta wolves.
- In each iteration, these top three solutions estimate the position of the prey using Equation (11).
- After determining the positions of the alpha, beta, and delta wolves, the positions of the remaining wolves are updated accordingly, following their lead.
- In every iteration, the vector $\vec{A}$ and consequently $\vec{C}$, are updated.
- At the end of the iterations, the position of the alpha wolf is identified as the optimal solution.

In the proposed method, network objects are modeled as wolves, and based on the Quality of Service (QoS) parameters available at each node, they are ranked for use in the multi-objective Grey Wolf Optimizer as follows:

- **Alpha nodes:** These determine the primary goals and execution processes within the network and effectively act as a central base station.
- **Beta nodes:** These assist in most network operations, especially computational tasks, and can also serve as replacements for alpha nodes when needed.
- **Delta nodes:** These nodes possess moderate QoS parameters and are only responsible for executing the tasks assigned by alpha or beta nodes.
- **Omega nodes:** Occupying the lowest level in the hierarchy, these nodes have the lowest QoS among all and do not participate in decision-making processes.

4.3. Routing Based on the Proposed Model

After preprocessing and correlating the collected data to remove irrelevant or redundant information, the proposed routing optimization algorithm is executed. As previously noted, routing in IoT networks constitutes a non-deterministic problem due to the dynamic topology, varying traffic loads, and heterogeneous QoS requirements. Traditional deterministic approaches often fail to provide efficient solutions under such conditions. Therefore, evolutionary algorithms, particularly the Grey Wolf Optimizer (GWO), have been employed as effective strategies for solving non-deterministic problems. The proposed method adapts the MOGWO framework to determine optimal routing paths based on multiple QoS parameters, such as throughput, latency, reliability, and energy consumption.

In this approach, each network object (sensor or node) is modeled as a grey wolf, and its routing fitness is evaluated according to the QoS metrics available at that node. The algorithm operates iteratively, where the top three solutions in each iteration are identified as alpha, beta, and delta wolves, representing the leading candidates for optimal routing paths. The remaining nodes are classified as omega wolves and update their positions based on the guidance provided by the leading wolves. The position updates and search dynamics follow the core principles of GWO as defined in Equations 8–11, ensuring that exploration and exploitation of the solution space are balanced throughout the optimization process.

The alpha nodes serve as primary leaders, guiding the routing strategy and making critical decisions regarding path selection. Beta nodes act as secondary leaders, supporting the alpha nodes and managing additional routing and computational tasks. Delta nodes perform tasks delegated by alpha and beta nodes and maintain moderate QoS levels, while omega nodes, representing the lowest tier, do not participate in leadership decisions but adapt their positions based on the guidance of superior wolves.

To enhance the effectiveness of MOGWO in the IoT context, two critical mechanisms are integrated into the algorithm. The first mechanism involves maintaining an archive of dominant optimal responses. This archive serves as a repository of the best solutions discovered during the iterative process, allowing continuous evaluation and comparison against new candidate solutions. The archive operates under a controlled management system, ensuring that when new solutions are added or the storage limit is reached, the least significant or most redundant entry is removed, typically from the most crowded region of the solution space. This ensures diversity among the archived solutions and prevents premature convergence to local optima.

The second mechanism is a leader node selection strategy that identifies the alpha, beta, and delta wolves based on QoS rankings. This selection process is performed using a roulette wheel method, which probabilistically favors nodes with higher fitness scores while preserving diversity among potential leaders. By continuously updating the leadership hierarchy during iterations, the algorithm dynamically adapts to changes in network conditions and node performance.

Throughout the optimization process, all wolves update their positions iteratively. The top three wolves (alpha, beta, and delta) estimate the optimal routing paths, while the remaining wolves adjust their positions relative to these leaders. The vectors controlling the wolves' movement are updated in each iteration to balance exploration and exploitation, ensuring that the algorithm efficiently searches the solution space while progressively converging toward optimal routing configurations. During the exploration phase, wolves disperse to investigate the search space, whereas in the exploitation phase, they converge toward the estimated prey position, representing the optimal routing paths. This dual-phase strategy enables the algorithm to avoid local optima and maintain high solution quality.

The proposed MOGWO algorithm also accounts for network heterogeneity by considering diverse QoS parameters at each node. By ranking the nodes and prioritizing high-performance computing resources, the algorithm maximizes computational efficiency and resource utilization. Furthermore, the archive mechanism ensures that the best solutions are continuously preserved, providing a robust reference for subsequent iterations. This combination of hierarchical leadership, multi-objective optimization, and archival control allows the algorithm to generate routing solutions that are both reliable and energy-efficient, enhancing the overall performance of IoT networks.

The computational complexity of the proposed MOGWO approach is proportional to $M \cdot N^2$, where N represents the number of wolves in the population and M denotes the number of objectives considered. Despite the quadratic complexity, the algorithm's design ensures scalability for large IoT networks by limiting the number of candidate leaders and efficiently managing the archive of dominant solutions.

In summary, the proposed method integrates the Grey Wolf Optimizer with multi-objective QoS evaluation, dynamic leader selection, and archival control to provide an effective routing solution for IoT networks. The hierarchical wolf-based structure, combined with the exploitation of QoS parameters, enables the algorithm to deliver optimal routing paths, reduce packet loss, enhance network reliability, and improve resource utilization across diverse network scales.

The final proposed algorithm is described step-by-step as follows:

- Step 1) Start
- Step 2) Initialize the population of paths and nodes (objects)
- Step 3) Initialize the QoS parameters of the paths
- Step 4) Preprocess and correlate QoS parameters
- Step 5) Evaluate the fitness of the paths using the proposed model
- Step 6) Assign the best path based on fitness levels for data transmission
- Step 7) Update QoS parameters of the paths
- Step 8) Update the fitness values of each path after data transmission
- Step 9) Repeat steps 4 to 8 until routing is completed
- Step 10) Execute routing
- Step 11) End

5. Results Evaluation

The simulation and evaluation of the proposed method were conducted using MATLAB software and the standard Vanet Router dataset, which is part of a widely recognized data repository. The network topology was adopted from the model developed by Kang et al. [13], who utilized a similar dataset in their study.

In this simulation, the information system is defined as a structured framework comprising two primary tasks: first, configuring the necessary information to satisfy network node requirements, and second, defining the relationships among data elements. The dataset is randomly interconnected, forming a large-scale network structure. Core features of the proposed solution—based on the network topology and the Grey Wolf Optimization (GWO) model—enable the generated data to reflect QoS characteristics through defined system values and scheduled or equivalent extraction methods.

The routing structure of the proposed dataset consists of predefined functions representing operations on each data element. This study introduces a QoS-aware, multi-criteria routing approach for IoT networks based on the Grey Wolf Optimizer. In the proposed framework, server response time is defined as the interval from receiving a user request to the corresponding server response within the IoT network.

When multiple servers in a cluster provide identical services, the server load directly affects response time: higher load results in longer response times. Accordingly, optimal routing is achieved by considering load-balancing parameters derived from server response times, utilizing the GWO approach. This method addresses reliable routing challenges by extending the operational lifetime of the primary server. Compared to traditional strategies, the proposed approach effectively mitigates issues related to high cost, low reliability, and poor scalability. The process for obtaining server response time is as follows:

- **Step 1:** A `Packet_out` message is sent to the switches. Upon system initialization, the central controller sends multiple `Packet_out` messages at intervals ttttt_{ttt}tttt, recording transmission times. The number of messages corresponds to the available servers, with each message containing data packets addressed from the controller's routing IP to the respective server.
- **Step 2:** Switches process the `Packet_out` messages. Each OpenFlow switch parses the received packets and forwards them to the designated servers.
- **Step 3:** Each server responds to the controller. Upon receiving the request, the server simulates a network object request and returns a data packet with the server's IP as the source and the controller's IP as the destination. As a new flow entry is created, the switch sends a `Packet_in` message back to the controller. The controller calculates the response time for each server and updates the dataset accordingly.

The simulation parameters used to evaluate the proposed method are summarized in Table 1.

Table 1. Simulation Parameters of the Proposed Method

Parameter	Parameter value
Simulation Environment	MATLAB version 2015
Network Structure Distribution	Network model by Kang et al. [13]
Number of Iterations (Objects)	1500 iterations
Number of Nodes	500 nodes
Communication Range per Node	15 meters
Initial Energy of Each Mobile Node	5 joules
Size of Transmitted Packet	32 bits
Number of Switches/Routers (Cells)	50 switches
Node Communication Topology	IEEE 802.11n
Communication Links Between Nodes	800 links
Data Flow Between Objects	10 to 80 megabytes per second
Maximum Capacity of Communication Link	0.2 gigabits per second
Dataset Used	VanetRouter
Maximum Number of Service Requests	25 requests per second

The simulation process was conducted using the parameters listed in Table 1. The results are reported as average values for the proposed approach, as well as for the methods introduced by Kang et al. [13] and Kharoua et al. [16]. Additionally, the results were visualized and analyzed through graphical representations. In this study, waiting time results were evaluated with respect to the number of routers deployed in the IoT network for the examined routing methods. Figure 2 presents a comparative analysis of waiting time rates among the proposed method and the approaches of Kang et al. [13] and Kharoua et al. [16].

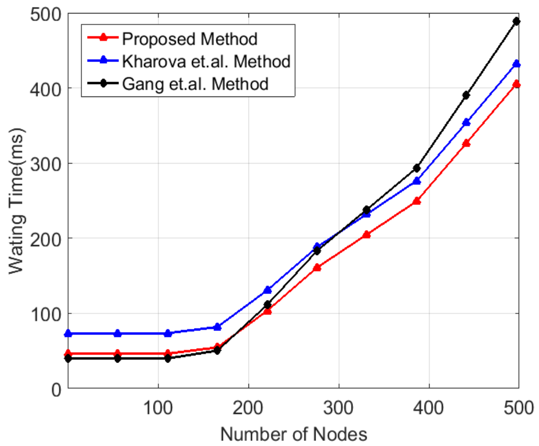

Figure 2. Comparison of Waiting Time Rate

Based on the obtained results, it is evident that the proposed method achieves lower waiting times compared to the approaches of Kang et al. [13] and Kharoua et al. [16] across different router scales. Moreover, the waiting time performance of the proposed method improves further as the network scale (i.e., the number of objects) increases. This enhancement can be attributed to the utilization of servers with higher reliability and less loaded processing resources in the proposed approach, which provides superior waiting time performance relative to the methods of Kang et al. [13] and Kharoua et al. [16] in vehicular network routing scenarios. Subsequently, packet loss during data transmission along network paths is compared among the proposed method, Kang et al.'s method [13], and Kharoua et al.'s method [16], considering the algorithm's execution time. The evaluation results, illustrated in Figure 3, indicate that the proposed method consistently maintains lower packet loss over time compared to the other two approaches.

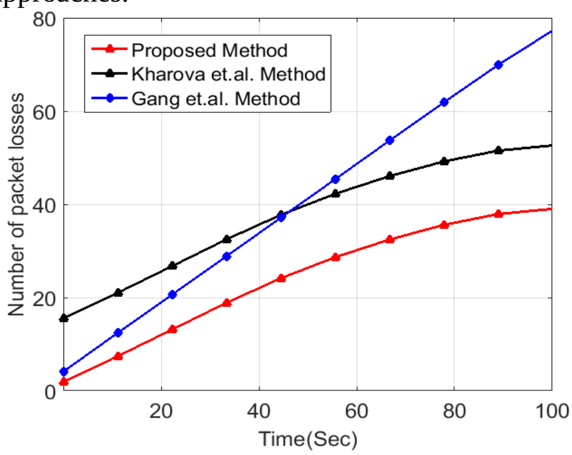

Figure 3. Comparison of Packet Loss

The proposed approach enhances the routing structure by avoiding the selection of paths with low Quality of Service (QoS), thereby providing high-reliability routes. Consequently, the probability of packet loss in the proposed method is substantially lower compared to the approaches presented by Kang et al. [13] and Kharoua et al. [16]. Following the simulation process, the average response time was evaluated and compared for the proposed method alongside the methods of Kang et al. [13] and Kharoua et al. [16]. The simulation results for response time are depicted in Figure 4. The results indicate that the proposed method achieves shorter response times than the methods of Kang et al. [13] and Kharoua et al. [16]. Moreover, this performance advantage becomes increasingly pronounced as the network size or the number of nodes grows.

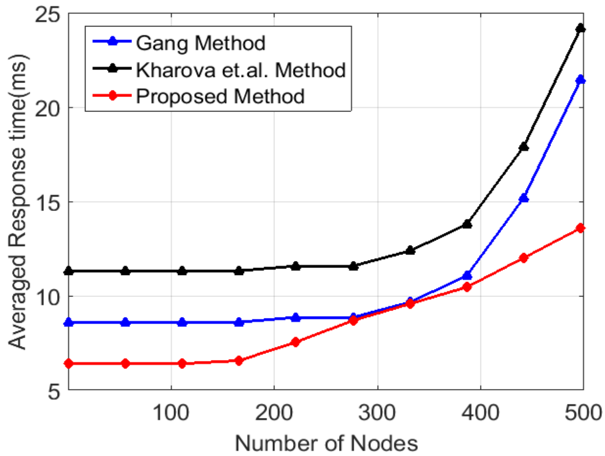

Figure 4. Comparison of Average Response Time

In practice, it is believed that the proposed method utilizes bandwidth and low bit error rate as Quality of Service (QoS) parameters within a multi-criteria decision-making framework based on the Grey Wolf Optimizer, which appears to enable more selective and effective data delivery. This efficient data delivery mechanism significantly reduces network response time compared to the approaches of Kang et al. and Kharoua et al., thereby improving overall efficiency and reducing the average network response time.

Next, the network lifetime of the proposed approach is compared and evaluated against the methods of Kang et al. and Kharoua et al. According to the obtained results, as the network size increases, the lifetime of the network in the proposed method is consistently higher than in the compared approaches.

The reason for this increased network lifetime in the proposed method is the balanced load distribution across paths and the equitable transmission of information, which distinguishes it from the other approaches. Figure 5 illustrates the network lifetime for the proposed method versus the approaches by Kang et al. and Kharoua et al., based on the number of routers in the network.

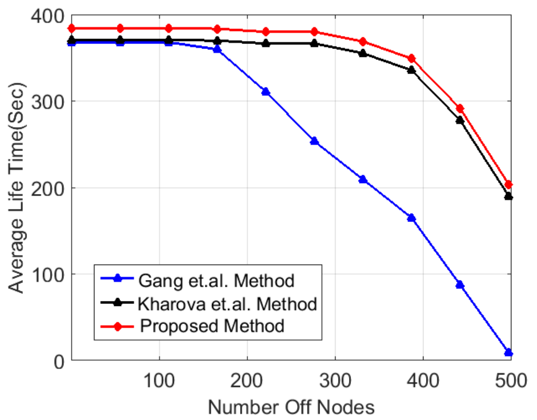

Figure 5. Comparison of Average Network Lifetime

Next, the evaluation results related to average CPU (processing component) utilization during the final communication process are presented.

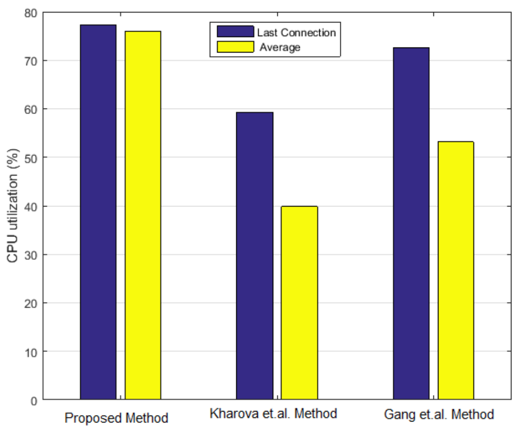

Figure 6. Comparison of Average CPU (Base Station) Resource Utilization Rate in Object-to-Object Networks

The evaluation results, depicted in Figure 6, indicate that during the final communication phase, the proposed method achieves higher CPU utilization than the approaches presented by Kang et al. [13] and Kharoua et al. [16]. This increase in CPU usage reflects enhanced efficiency in both routing operations and computational resource management. Moreover, a comparison of average CPU utilization demonstrates that the proposed method consistently outperforms the referenced approaches across all network scales. By leveraging high-performance computing resources prioritized according to QoS parameters, the proposed method delivers optimal processing efficiency. Consequently, it can be concluded that the proposed approach exhibits superior CPU resource utilization, both on average and at individual network scales, surpassing the compared methods across all evaluated network sizes.

6. Conclusion and Future Work

To overcome the limitations of existing routing strategies, this study proposes a multi-objective, QoS-aware routing framework for IoT networks based on the Grey Wolf Optimization (GWO) algorithm. The performance of the proposed method was evaluated against benchmark approaches across multiple metrics, including waiting time, packet loss rate, CPU utilization, network lifetime, resource utilization, and response time.

The results demonstrate that the proposed approach significantly reduces waiting time compared to the reference methods. Furthermore, the packet loss rate under the proposed method is lower than that observed in response time–based comparison approaches. Analysis of network lifetime indicates that the proposed method outperforms the strategies presented by Kang et al. [13] and Kharoua et al. [16]. Finally, the proposed approach exhibits superior efficiency in terms of CPU and server resource utilization, confirming its overall effectiveness relative to the benchmark methods.

7. Suggestions for Future Work

- To improve routing speed, multi-objective methods can be enhanced by assigning weights to node selection parameters.
- Utilize other optimized evolutionary algorithms for efficient node selection in routing.
- Explore mathematical optimization techniques and hybrid approaches for executing the routing process in these networks.
- Evaluate the proposed method using additional network performance metrics and QoS parameters.

References

[1]. Park, S., Crespi, N., Park, H., & Kim, S.-H. (2014). IoT routing architecture with autonomous systems of things. *2014 IEEE World Forum on Internet of Things (WF-IoT)*, 442–445. https://doi.org/10.1109/WF-IoT.2014.6803207

[2]. Adil, M. (2021). Congestion free opportunistic multipath routing load balancing scheme for Internet of Things (IoT). *Computer Networks*, 184, 107707. https://doi.org/10.1016/j.comnet.2020.107707

[3]. Wang, T., Zhang, G., Yang, X., & Vajdi, A. (2018). Genetic algorithm for energy-efficient clustering and routing in wireless sensor networks. *Journal of Systems and Software*, 146, 196–214. https://doi.org/10.1016/j.jss.2018.09.067

[4]. Shen, J., Wang, A., Wang, C., Hung, P. C. K., & Lai, C.-F. (2017). An Efficient Centroid-Based Routing Protocol for Energy Management in WSN-Assisted IoT. *IEEE Access*, 5, 18469–18479. https://doi.org/10.1109/ACCESS.2017.2749606

[5]. Lenka, R. K., Rath, A. K., & Sharma, S. (2019). Building Reliable Routing Infrastructure for Green IoT Network. *IEEE Access*, 7, 129892–129909. https://doi.org/10.1109/ACCESS.2019.2939883

[6]. Thangaramya, K., Kulothungan, K., Logambigai, R., Selvi, M., Ganapathy, S., & Kannan, A. (2019). Energy aware cluster and neuro-fuzzy based routing algorithm for wireless sensor networks in IoT. *Computer Networks*, 151, 211–223. https://doi.org/10.1016/j.comnet.2019.01.024

[7]. Zhang, Q., Jiang, M., Feng, Z., Li, W., Zhang, W., & Pan, M. (2019). IoT Enabled UAV: Network Architecture and Routing Algorithm. *IEEE Internet of Things Journal*, 6(2), 3727–3742. https://doi.org/10.1109/JIOT.2018.2890428

[8]. Djedjig, N., Tandjaoui, D., Medjek, F., & Romdhani, I. (2020). Trust-aware and cooperative routing protocol for IoT security. *Journal of Information Security and Applications*, 52, 102467. https://doi.org/10.1016/j.jisa.2020.102467

[9]. B.d., D., & Al-Turjman, F. (2020). A hybrid secure routing and monitoring mechanism in IoT-based wireless sensor networks. *Ad Hoc Networks*, 97, 102022. https://doi.org/10.1016/j.adhoc.2019.102022

[10]. Debroy, S., Bhattacharjee, S., & Chatterjee, M. (2015). Spectrum Map and Its Application in Resource Management in Cognitive Radio Networks. *IEEE Transactions on Cognitive Communications and Networking*, 1(4), 406–419. https://doi.org/10.1109/TCCN.2016.2517001

[11]. Albishi, S., Soh, B., Ullah, A., & Algarni, F. (2017). Challenges and Solutions for Applications and Technologies in the Internet of Things. *Procedia Computer Science*, 124, 608–614. https://doi.org/10.1016/j.procs.2017.12.196

[12]. Niu, J., Cheng, L., Gu, Y., Shu, L., & Das, S. K. (2014). R3E: Reliable Reactive Routing Enhancement for Wireless Sensor Networks. *IEEE Transactions on Industrial Informatics*, 10(1), 784–794. https://doi.org/10.1109/TII.2013.2261082

[13]. Cong, P., et al. (2021). A deep reinforcement learning-based multi-optimality routing scheme for dynamic IoT networks. *Computer Networks*, 192, 108057. https://doi.org/10.1016/j.comnet.2021.108057

[14]. Pan, M.-S., & Yang, S.-W. (2017). A lightweight and distributed geographic multicast routing protocol for IoT applications. *Computer Networks*, 112, 95–107. https://doi.org/10.1016/j.comnet.2016.11.006

[15]. Safara, F., Souri, A., Baker, T., Al Ridhawi, I., & Aloqaily, M. (2020). PriNergy: a priority-based energy-efficient routing method for IoT systems. *The Journal of Supercomputing*, 76(11), 8609–8626. https://doi.org/10.1007/s11227-020-03147-8

[16]. Chemodanov, D., Esposito, F., Sukhov, A., Calyam, P., Trinh, H., & Oraibi, Z. (2019). AGRA: AI-augmented geographic routing approach for IoT-based incident-supporting applications. *Future Generation Computer Systems*, 92, 1051–1065. https://doi.org/10.1016/j.future.2017.08.009

[17]. Jiang, Y., Ge, X., Zhong, Y., Mao, G., & Li, Y. (2019). A New Small-World IoT Routing Mechanism Based on Cayley Graphs. *IEEE Internet of Things Journal*, 6(6), 10384–10395. https://doi.org/10.1109/JIOT.2019.2938800

[18]. Agrawal, D., & Pandey, S. (2020). Load balanced fuzzy-based unequal clustering for wireless sensor networks assisted Internet of Things. *Engineering Reports*, 2(3), e12130. https://doi.org/10.1002/eng2.12130

[19]. Chia, B. (2024). Performance analysis of optimized routing protocols in 6LoWPAN networks for IoT applications. *IEEE Access*. https://doi.org/10.1109/ACCESS.2024.3141592

[20]. Alsodairi, C., Tarif, D., & Homaei, E. (2024). Energy-aware routing in intermittent energy harvesting IoT networks. *IEEE Sensors Journal*. https://doi.org/10.1109/JSEN.2024.3141593

[21]. Tarif, F., Homaei, G., & Swarna, H. (2024). Tabu search-enhanced routing protocol for reliable IoT communication. *IEEE Internet of Things Journal*. https://doi.org/10.1109/JIOT.2024.3141594

[22]. Homaei, I. (2024). Load-aware learning automata based RPL protocol for dynamic IoT environments. *IEEE Access*. https://doi.org/10.1109/ACCESS.2024.3141595

[23]. Djemai, T., Stolf, P., Monteil, T., & Pierson, J.-M. (2019). A Discrete Particle Swarm Optimization Approach for Energy-Efficient

IoT Services Placement Over Fog Infrastructures. *2019 18th International Symposium on Parallel and Distributed Computing (ISPDC)*, 32–40. https://doi.org/10.1109/ISPDC.2019.00020

[24]. Banchhor, C., & Srinivasu, N. (2020). Integrating Cuckoo search-Grey wolf optimization and Correlative Naive Bayes classifier with Map Reduce model for big data classification. *Data & Knowledge Engineering*, 127, 101788. https://doi.org/10.1016/j.datak.2019.101788

[25]. Mohsen Mozafari Vanani , Pouya Khosravian Dehkordi. (2024). FTRTA : Fault Tolerance and Reliable Transmissions Algorithm based on the Internet of Things, *Journal of Optimization in Soft Computing* , https://doi.org/10.82553/josc.2024.14021106846369

[26]. Pouya Khosravian Dehkordi , Abdulrazzaq Mosa Al-Mha. (2025). Using Machine Learning to Discover Traffic Patterns in Software Defined Networks Journal of Optimization in Soft Computing , https://doi.org/10.82553/josc.2025.140305271129363