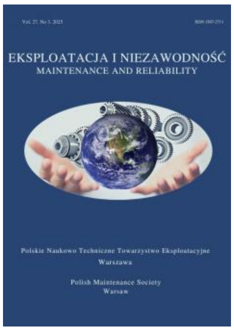

Eksploatacja i Niezawodnosc – Maintenance and Reliability
Volume 27 (2025), Issue 2
journal homepage: http://www.ein.org.pl

Article citation info:
Lei B, Optimizing IoT Routing with a Focus on Service Quality Using Integrated Big Bang-Big Crunch Central Force Optimization, Eksploatacja i Niezawodnosc – Maintenance and Reliability 2025: 27(2) http://doi.org/10.17531/ein/195794

Optimizing IoT Routing with a Focus on Service Quality Using Integrated Big Bang-Big Crunch Central Force Optimization

Indexed by: Web of Science Group

Bo Lei[a,*]

[a] School of General Education, Sichuan Vocational and Techncial College, China

Highlights

- BB-CFO hybrid technique combines BB-BC exploration and CFO accuracy for IoT routing.
- BB-CFO boosts IoT routing by optimizing energy, reducing delays, and enhancing delivery.
- Q-FRPL method using BB-CFO outperforms traditional routing in energy and latency.
- A review of IoT routing protocols highlights BB-CFO as an efficient alternative solution.

Abstract

This paper introduces a novel hybrid optimization algorithm, BB-CFO, which combines the big bang-big crunch (BB-BC) and central force optimization (CFO) algorithms to address key challenges in Internet of Things (IoT) networks, such as energy constraints, delay, and scalability. The proposed BB-CFO algorithm improves both the exploration and exploitation phases of optimization, providing a balanced approach for solving complex routing problems in low-power and lossy networks. The algorithm is integrated into the quality fuzzy routing protocol for low-power and lossy networks (Q-FRPL), which is evaluated through extensive simulations using Cooja and NS2 environments. The contributions of this study are twofold: first, the development of a hybrid optimization technique that enhances routing efficiency in IoT networks, and second, the demonstration of its effectiveness through comparative analysis with conventional algorithms. The obtained results show that the BB-CFO-based Q-FRPL protocol greatly reduces energy consumption as high as 800 mW in Cooja and up to 900 mW in NS2, at reduced end-to-end latency of 40 and 45 ms in Cooja and NS2, respectively, when the packet delivery ratio is 96.985%. These reflect the performance, scalability, and robustness of the proposed method and also show one possible solution toward efficient IoT networks.

Keywords
big bang-big crunch, energy efficiency, hybrid optimization algorithms, IoT network efficiency, routing optimization.

1. Introduction

1.1. Background and Aims

IoT is one of the fastest-developing areas nowadays, which connects everything from home appliances and industrial sensors through several interconnected networks [1]. Such connectivity provides real-time data collection and communication and thus fosters innovation in fields such as healthcare, smart cities, and automation [2]. However, large-scale deployment of IoT devices introduces several challenging issues in their effective management at optimum network performance. One of the key issues in IoT networks is managing the limited resources of connected devices [3]. Many IoT devices rely on battery power, making energy efficiency
a critical concern. In addition, these devices often have limited processing capabilities and face varying network conditions, which can affect their ability to efficiently route data. Traditional routing protocols, which were designed for

(*) Corresponding author.
E-mail addresses: B. Lei, 13795756862@163.com

more uniform and less constrained environments, usually fail to meet the expectations of such diverse applications of IoT [4]. In addition, there are scalability and dynamic condition issues that normally affect IoT networks: The more devices a network owns, the bigger the complexity will be in terms of management and optimization of routing [5]. IoT networks have to work in a highly variable environment concerning data loads, signal strength, and interference; hence, their performance and reliability are seriously challenged. Given these difficulties, advanced optimization techniques will be required to provide low latency and high packet delivery rates. Yet, traditional routing methods poorly fulfill the unique and dynamic requirements of IoT applications. This also significantly raises demands for higher approaches that could improve routing efficiency considering resource constraints in IoT devices [6].

1.2. Literature Review

IoT transforms network connectivity, from a simple sensor to a high-powered server. Increased connectivity demands network management and routing optimization. A decent amount of research has been carried out regarding the unique challenges thrown up by the IoT on energy efficiency and routing performances. This review summarizes major contributions of research and also highlights certain areas that call for further exploration.

Muzammal et al. [7] discussed the security problems in IoT networks, including the vulnerability analysis of the RPL routing protocol and some countermeasures. Improvement in IoT routing security is sought in trust-based schemes, where the SecTrust and DCTM-IoT trust models are included. Special attention was paid to trust metrics and corresponding research challenges. In their opinion, trust represents one of the basic security paradigms within the context of IoT network and routing protocol security. Almusaylim et al. [8] discussed the field of study gaining great importance on the Internet of Things, ranging from E-health to smart cities. They discussed its vulnerability whereby the IoT devices, especially the smart cities, are prone to every kind of security threat. Further, despite the availability of existing RPL routing protocols for protection, the authors found them insufficient and unworkable to deal with the complex challenges in IoT networks. The review highlighted the development of better RPL protocols that should be secure against both Rank and Version number attacks. Solutions should be customized for IoT applications in regards to Smart Cities. Arivubrakan and Kanagachidambaresan [9] discussed several routing issues in IoT and focused on the RPL protocol. They proposed FLOF, which is a method to enhance the reliability of parent selection by integrating multiple metrics. Simulation results showed substantial improvements in almost all QoS parameters. Darabkh et al. [10] discussed the limitations of the RPL protocol, which chooses objective function (OF) based on single metrics. It came up with FL-HELR-OF, which uses multiple metrics to extend network lifetime. Simulation results showed that FL-HELR-OF outperformed other methods for all metrics under consideration. A new rider foraging optimization (RFO) algorithm was proposed by Vijay and Ranjan [11] for secure routing in IoT. Herein, the authors have aimed to minimize delays and energy consumption while maximizing throughput. RFO has selected the optimal path depending on the constraints on trust, energy, and delay using rider optimization combined with bacterial foraging algorithms. This reduced the node energy consumption prolonged the network lifetime and provided high throughput at minimum delay and energy utilization.

Apart from being a serious challenge for routing data packets, the identification of connecting devices globally was identified as the main vision of IoT by Solapure and Kenchannavar [12]. Later, they modified low-power and lossy networks (RPL) by the introduction of new OFs based on various metrics. The corresponding simulation results showed that these designs outperformed conventional designs along with many aspects of packet delivery ratio (PDR), latency delay (LD), energy consumption, overhead, and convergence time. They have identified the respective OF designs for the respective IoT applications such as health monitoring and forest monitoring based on the respective performance characteristics. Darabkh et al. [13] stated that though IoT is a growing area of interest, most of the devices are resource bounded in terms of battery power, processing capacity, memory, and bandwidth. It is very difficult to support Protocol version 6 (IPv6) on such devices. The authors gave a short overview regarding the expansion of IoT and the introduction of RPL in the year 2012. The prime objective was to carry out an in-depth analysis of its functionality along with the challenges. It compared different RPL-based protocols, highlighted some of the key research challenges, and provided useful insights into future developments. The recommendations made in the conclusion have been presented as valuable resources for researchers working on the enhancement of RPL.

Maheswari et al. [14] explained that IoT networks are mostly constituted by low-power and weakly connected devices, where data communication is done through wireless mediums that are under high noise interference and connection failures. Thus, a reliable routing protocol is of utmost importance in these types of high loss, low-power networks. The authors have reviewed new emerging trends in IoT applications focused on the strong need that was arising for an efficient routing protocol like RPL, which could handle the intrinsic issues in LLNs. Further, they have discussed congestion-related issues in RPL and also discussed various related previous works that were suggested to improve RPL performance. They reviewed congestion control metrics and routing schemes and their respective merits and demerits to possibly further the research. Mohamed et al. [15], conducted a performance evaluation of RPL in IoT networks using InstantContiki 3.0 and CoojaGUI. In their work, these researchers assessed some expected transfer realization rate (ETX)-based metrics, radio duty cycling (RDC), energy consumption, packet reception, and neighbor relationships. In line with this, the results reflected that the effective formations of DODAGs depend on ETX, reasonable energy consumption, and packet reception across the motes. Garg et al. [16] conducted a survey of DODAG formation strategies in low-power and lossy networks using the RPL protocol. The authors analyzed the existing methods, classified the related metrics, and presented the open issues of future research. It provides the outline of the researchers who work on the RPL protocol and DODAG formation strategy.

Niu [17] proposed an optimized DODAG construction scheme for RPL-based wireless networks. It introduced a node reset action to restart the DODAG building upon the failure of paths with the least disruption and reduction of power loss. This performs much better under interference conditions and has reduced data retransmission rates. Wang et al. [18] proposed a data-oriented RPL algorithm that aimed at eliminating the limitation on IoT networks. Data were divided by content in routing, which would reduce duplicated data and delay, thereby reducing energy use and prolonging the lifetime of the network. Route choice was optimized by binary gray wolf optimization. Tests were done showing that the enhancements in energy efficiency and reduction in instability periods with minimum delays compared to other methods proved stable and had long-acting activity in the case of maximum node count networks. Rajeesh Kumar et al. [19] presented the salp swarm algorithm (SSA) for the solution of the optimal power flow (OPF) problem in an efficient manner, with the integration of thyristor-controlled series capacitor (TCSC) devices into the power systems. The authors developed an optimization approach that optimizes the generator's active power, voltage magnitudes, and transformer tap settings and gave a better performance than the referenced algorithms in their simulations on the IEEE-30 bus system. SSA is a very powerful and novel technique in optimizing power systems; hence, it may minimize the severity factors and even outperform the existing algorithms. Mallala and Dwivedi [20] also worked on the OPF problem by implementing SSA and integrating TCSC into the power systems. They optimized the control variables and simulated their methodology on the IEEE-30 bus system. Minimizing the above severity factors, besides proving the efficiency of the SSA in solving the OPF problem compared with other algorithms, makes this research an important milestone for the optimization of power systems.

In 2006, Erol and Eksin [21] introduced the BB-BC method as a new optimization technique. The BB-BC method proposes the universe's evolution and probable end within its optimization technique based on some theories in physics and astronomy, the BB-BC paradigm [22]. Mbuli and Ngaha [23] studied the BB-BC algorithm mainly to highlight the adaptability of the algorithm in solving several power system optimisation problems. To resolve some issues and enhance the quality of solution different versions of BB-BC have been developed. The algorithm has repeatedly exhibited excellent efficiency, frequently surpassing rival techniques in attaining optimal solutions across several power system domains. This method is distinguished by rapid search space exploration and vigorous exploitation of the solution space. This is indicated by the reduction in population size. There are many other methods inspired by nature that have been applied to the QAP, BPP, and JSSP, such as genetic algorithm, ant colony optimization, particle swarm optimization, scatter search, local search, hybrids, and hyper-heuristics. The BB-BC has been applied to a limited number of combinatorial optimization problems. The BB-BC was compared to particle swarm optimization (PSO), harmony search (HS), and ant colony optimization (ACO) over the size optimization of space trusses. They showed that the performance of the BB-BC demonstrates superiority over PSO, HS, and ACO in computational time and quality of solutions. The BB-BC was also applied to several optimization problems, such as target tracking for underwater vehicle detection and tracking; and engineering optimization. Also, an enhanced version of the BB-BC was applied to solve course timetabling problems, where it outperformed several similar methods and showed a consistent and fast convergence towards optimality [24]. The CFO, proposed by Formato [25], is a deterministic metaheuristic inspired by gravitational kinematics. Unlike stochastic algorithms like GA or ACO, CFO's deterministic nature ensures consistent results with the same parameters. It requires only a single run for evaluation, eliminating the need for statistical analysis [26].

1.3. Literature Gap and Research Contribution

The current body of research on IoT routing and optimization reveals several critical gaps. Traditional routing methods, such as those based on the RPL protocol, often struggle with the dynamic and resource-constrained nature of IoT networks. Challenges persist due to insufficient energy management, elevated latency, and restricted scalability. Most of the literature has proposed the improvement of existing processes through optimization techniques; however, solutions often relate to particular aspects of the problem or poorly combine different optimization criteria. The following gaps should be mentioned: the lack of integrated proposals that contribute to simultaneously increasing energy efficiency, reducing delay, and improving the delivery speed of packets in different applications of IoT. Although the BB-BC has shown promising performances in some scenarios of optimization problems, their applications in IoT routing have been scant. The CFO algorithm has reliably performed very well in predictable environments; yet, it has never been widely integrated with other algorithms in addressing IoT-specific problems. This paper is intended to fill these gaps by proposing the hybrid optimization algorithm BB-CFO. Summary: The new method combined BB-BC and the CFO algorithm together for the solution of a few optimization objectives at the same time. BB-CFO is presented as a strong method against IoT routing challenges, integrating the capability of exploration and exploitation of BB-BC through the deterministic precisions of the CFO. Major focus goes to comprehensive enhancements in energy efficiency, delay reduction, and enhancement in the packet delivery rate. The results obtained from the simulation depict the fact that the BB-CFO-based Q-FRPL method outperforms the conventional methods in major performance metrics and thus becomes efficient and scalable to meet the complex demands of IoT networks. Thus, the major contributions and novelties of this work can be summed up below:

1. This study proposes the BB-CFO hybrid optimization technique, hybridizing the BB-BC and CFO methods. Along this, the proposed unique amalgamation will exploit the exploratory strengths of BB-BC in concert with the deterministic exactness of the CFO to solve the complicated problems of IoT routing.
2. This research indicates that the BB CFO algorithm greatly improves the routing efficiency of IoT networks by optimizing energy usage and delays and improving the packet delivery rate, hence presenting an all-rounded solution to the shortcomings of previous approaches.
3. The suggested Q-FRPL method, utilizing BB-CFO, outperforms conventional routing techniques through comprehensive simulations. The findings demonstrate significant enhancements in energy consumption, overall latency, and packet delivery rates, validating the approach's efficacy and scalability.
4. This paper presents a review of the current IoT routing protocols along with optimization techniques.

The review presented various advantages and disadvantages for different techniques and finally proposed BB-CFO methodology, an efficient alternative.

5. Generally, major challenges such as energy constraints, scalability issues, and dynamic conditions in IoT networks have been targeted with one new comprehensive optimization scheme.
6.

2. Problem Formulation

The OF is an extremely important assessment criterion in the RPL protocol that would adapt to various network applications and information transfers. In general, this function takes as input key parameters from the network, mainly quality-of-service metrics in routing. The importance of parameters and performance criteria used in the OF comes from the weights assigned by the underlying network objectives. What RPL is concerned with is that the correct path selection from available paths depends largely on the OF. The OCP utilizes the OF in the DIO message to configure the network. The OF defines the constraints and performance criteria of a node, represented by a value called rank, based on the RFC6551 standard. The node's distance from the network's root affects this rank, and the goal function also controls the parent selection process. RPL's default OF, OF0, favors locating the nearest root over other qualitative factors like load balance, energy, and E2ED. It does this by concentrating only on the number of steps in the ranking function. A child node chooses a parent as its successor if one is available and names the parent with the lowest rank as the primary parent during parent selection. The network does not attempt to load balance in OF0; all traffic travels upstream to the root node. The node will unavoidably remove itself from the weak parent list and join the alternate or backup parent if the present link is unable to communicate data toward the root for whatever reason.

The OF that has been demonstrated is known as Q-FRPL, and it symbolizes the OF2 routing protocol, which is noted for its energy efficiency. This proposed method employs a fuzzy approach to assessing network nodes dynamically, and the evaluations are disseminated through DIO update messages. The fuzzy system in OF2 takes into account variables including the amount of energy left, the rate of delay, and the equivalent rate, or ETX, which represents a node's efforts to contact its parent. The following are the suggested fuzzy system's performance requirements:

- Residual energy of the node (first input)
- Average E2ED to well (second input)
- Expected transfer realization rate or ETX (third input)

2.1. Fuzzification of Parameters

Three fuzzy input parameters will thereby identify the unique state of each time unit. In line with the fuzzy logic and triangulate model, each input parameter in the fuzzy system is represented by a triangulation diagram. In each image, separate and similar triangles may be used to relate the behavior of a parameter in the variable values of the x-axis to other values on the y-axis. For each point on the x-axis, there are two values on the y-axis. The residual energy is shown in one of the diagrams, with a starting residual energy of 10 units and a minimum state of zero. There are five different degrees of residual energy: very low, low, medium, high, and very high. The leftover energy either covers two levels in succession or fits into one of these levels. Notably, a node with more energy left over suggests that it is more desired. The average E2ED rate of the node during the most recent interval is another characteristic taken into account in the suggested fuzzy system. The time interval between the generation of DIO packets in the root node and their reception in each node is used to calculate this rate. For example, six DIO messages in a single-step or multistep format might have reached the leaf node from the root in the previous ten seconds. Based on the DIO packet timestamp, this node will compute the average delay and share it with its offspring. When a node's rate is lower, it means that its children's advertisement score is greater.

The ETX, a criterion for identifying high-efficiency routes, is the third parameter in a fuzzy system that is being suggested. The computation of ETX involves approximating the quantity of transmissions necessary for a packet to arrive at its intended destination. If the link from source (s) to destination (d) is denoted as $s \rightarrow d$, then ETX is calculated using the following expression. Let DRs→d be the estimated delivery rate for the connection s→d. Since packet delivery rate is a random variable, ETX can make the network unstable due to its greedy behavior in continuously pursuing better connections over time. This gives a better overview of the paths that the RPL protocol chooses using the ETX criterion. The documentation estimates that almost half the routes have an ETX of 2, which is mostly within the range of 2 to 3 in conditions of low traffic. The more network traffic, the higher the ETX value of a route; it grows exponentially with traffic. In the RPL protocol, the ETX criterion considers the amount of transmission required to get a packet from an originator to a destination, thereby acting as the quality assessment of the path. ETX stands for the Expected Transmission Count, another metric for estimating the efficiency and reliability of the routes in IoT networks. As network traffic rises, there will be less reliability; highly critical ETX gives low route efficiency in high network traffic.

2.2. Routing Phase

After establishing the neighborhood tables, the network generates data and starts transmitting the packet; through that, the parent nodes will get notified about the probable child nodes, the potential parent candidates to which the packet transfer will be forwarded. Each child node in the proposed scheme has to choose the optimum parent amongst these candidates based on the availability, energy, and latency parameters. Therefore, following the selection criteria mentioned above, each node operates based on the parameters outlined in Equation (1) [1]:

$$Rank_n = Rank_C + \frac{1}{Fuzzy(RE,D,ETX)} \tag{1}$$

To prevent the package from being routed in a direction that is too far from the root, each parent node's rating rank is represented by $Rank_n$ in Equation (1). Fuzzy (RE, D, ETX) is a fuzzy-calculated value for each parent node that is communicated to children through a DIO message. $Rank_C$ evaluates the parent node's current rank. The leaf node ultimately chooses the transmission option with the lowest rank. Since the parameters related to the routing mechanism within the network's tree structure have been considered, energy consumption and network lifetime are key objectives of this research. During these experiments, a specific quantity of sensor nodes will be distributed randomly within the simulated environment. The objective is to evaluate the suggested method's efficacy and its fundamental counterpart in establishing an energy-efficient network tree. Additionally, the evaluation will measure their success in selecting the optimal route within the network from source to destination.

Table 1 outlines simulation conditions for the proposed network. To ensure a fair evaluation, all scenarios are kept consistent between the baseline protocol and the proposed method. Notably, due to variations in the simulation environments between the proposed Q-FRPL and the quality routing protocol for low-power and lossy networks (QRPL) algorithm, the simulation time is adjusted as one of the criteria. It is anticipated that the network's information flow and traffic rate will remain constant throughout the simulation phases. According to Table 1, the objective of these tests is to evaluate how well the suggested approach and its simpler equivalents accomplish dynamic routing results and energy efficiency by randomly assigning 20 sensor nodes around the simulated environment.

Table 1. Environmental conditions and simulation parameters.

Parameter	Value
Channel bandwidth	250 kHz
frequency carrier	2.4 GHz
data packet size	1600 bits
Hello package size	120 bits
Network convergence time	1000 milliseconds
Number of network nodes	20 knots
Network environment	80 meters by 80 meters
Node radio board	5 meters
Maximum queue length	2 packages
The initial energy of the node	5 joules
Simulation time	324 seconds

The proposed approach's structure can be summarized as follows:

I. Objective Function Enhancement:
  - Q-FRPL is introduced as an enhanced OF within the RPL protocol.
  - It utilizes fuzzy logic to dynamically calculate and distribute node values based on key parameters, namely residual energy, average E2ED, and the ETX.
  - The objective is to create a more adaptive and efficient routing structure by considering multiple criteria in node evaluation.

II. Fuzzy Logic-based Parameter Valuation:

– Fuzzification of parameters involves assessing the current state of each parameter (residual energy, average E2ED, and ETX) using triangulation diagrams.
– Fuzzy logic enables a nuanced evaluation, allowing for more flexible and context-aware decision-making.
– Parameters are assigned fuzzy sets and membership functions, capturing the uncertainty and imprecision inherent in real-world IoT networks.

III. Performance Criteria:
– The fuzzy system evaluates nodes based on three primary performance criteria.
– Residual Energy of the Node (first input).
– Average E2ED to the Well (second input).
– Expected Transfer Realization Rate or ETX (third input).

IV. Fuzzification Process and Triangulation Diagrams:
– Each parameter's fuzzification involves representing it using triangulation diagrams.
– The residual energy diagram, for example, may categorize energy levels into very low, low, medium, high, and very high.
– Triangulation allows mapping parameter values to specific behaviors, aiding in the decision-making process.

V. Routing Phase Optimization:
– After forming neighborhood tables, each child node selects the most suitable parent based on criteria such as delay, energy, and availability.
– This optimization contributes to the overall efficiency of the network by dynamically choosing parents that meet the specified Q-FRPL criteria.

VI. Comparison with Existing Objective Functions:
– Q-FRPL is positioned as an improvement over existing OFs, considering its integration of fuzzy logic and the comprehensive evaluation of multiple parameters.
– Advantages include better adaptability to dynamic network conditions, improved energy efficiency, and enhanced decision-making in parent selection.

VII. Limitations and Trade-offs:
– Acknowledge potential limitations introduced by Q-FRPL, such as increased computational complexity due to fuzzy logic.
– Discuss trade-offs and weigh them against the benefits, emphasizing the net positive impact on IoT network performance.

The Pseudocode of the proposed approach is presented in the following.

***Pseudocode for Q-FRPL:* **

```
1. Initialize:
   - Define the fuzzy sets for input parameters: Energy Level, Link Quality, and Delay.
   - Define fuzzy rules for selecting the optimal route.
   - Initialize routing table for each node.
2. For each node in the network:
   - Monitor current energy level, link quality, and delay.
3. Fuzzy Inference Process:
   For each node in the network:
     - Input: Measure the current values for energy level, link quality, and delay.
     - Fuzzification: Convert the input values into fuzzy linguistic variables using membership functions.
     - Apply fuzzy rules:
       - If (Energy Level is HIGH) and (Link Quality is GOOD) and (Delay is LOW), then Route Priority is HIGH.
       - If (Energy Level is LOW) or (Link Quality is POOR), then Route Priority is LOW.
     - Defuzzification: Convert the fuzzy output (Route Priority) into a crisp value.
4. Route Selection:
   - For each node, calculate the Route Priority for all available neighbors.
   - Select the neighbor with the highest Route Priority as the next hop.
5. Packet Transmission:
   - Transmit the data packet through the selected next hop.
6. Route Maintenance:
   - Periodically update the fuzzy parameters (Energy Level, Link Quality, Delay).
   - If a link failure or significant degradation in any parameter is detected, trigger route recalculation.
7. End Process:
   - Stop when the packet reaches the destination or no valid routes are available.
```

That is considered to be the Q-FRPL algorithm at the abstracted pseudocode level, including fuzzification, fuzzy inference, and parent selection. The actual implementation will take into account any particularities coming from various methodologies of fuzzy logic and/or other relevant network parameters within the process.

Generally, Q-FRPL stands for enhanced model OF using

Fuzzy Logic for dynamic assessments and distribution of node values with criteria of residual energy, end-to-end delay (E2ED), and ETX. In general, using fuzzy logic, the structure of routing is much more adaptive and efficient because more criteria are taken into consideration while assessing the nodes. It utilizes fuzzy logic to calculate and distribute the node values for adaptability and efficiency in routing. Q-FRPL makes decisions flexibly, considering several factors such as residual energy, average E2ED, and ETX by considering all these factors using triangulation diagrams. A fuzzy system can evaluate the nodes regarding these parameters for better routing decisions. Mapping values of parameters to actions optimize decision-making, where the parameters are represented as triangulation diagrams. Parent selection during P-Selection is based on criteria such as delay, energy, and availability; hence, it returns more efficient network operations. Q-FRPL outperforms the existing OFs since the inclusion of fuzzy logic and the evaluation of more parameters increase its adaptability and efficiency in energy. This has the disadvantage of increasing computational complexity but the advantage of network performance cannot be compromised. Reciprocally, a proposed scheme should allow the routing structure to be further made adaptive and efficient by applying fuzzy logic to the dynamic nature of the IoT Network for optimized energy efficiency, reduced end-to-end delay, and improved packet delivery rate.

On the other hand, QRPL, in addition to the basic RPL protocol with OF0, proposes the Q-FRPL method, which will evaluate the nodes dynamically using fuzzy logic based on the parameters of residual energy, average E2ED, and ETX. Whereas QRPL could evaluate nodes based on more than one criterion, QRPL would still suffer from problems when highly trafficked paths create congestion or packet losses, at least until the network recalculates its routing. Adding fuzzy logic to QRPL will also escalate computation overhead and therefore may have an impact on real-time performance in IoT networks. On the other side, QRPL improves adaptability and efficiency but is suffering from a few limitations related to high-traffic paths and computational overhead for further assurance of the best networking performance, especially over delay rate.

3. Optimization Algorithm

Optimization algorithms find the optimal solution by minimizing or maximizing an OF within given constraints. They iteratively employ techniques like gradient descent, genetic algorithms, simulated annealing, and linear programming to explore and find the optimal solution [27].

3.1. Big Bang-Big Crunch Algorithm

The BB-BC algorithm briefly consists of these steps: generating the initial solution randomly, calculating the fitness function for all solutions, finding the dense center from Equation (2) or choosing the member with the best fitness value as the dense center, calculating new solutions around the dense center using the normal distribution which is formulated as follows and finally return to step 2 until end condition of the algorithm is fulfilled [27]:

$$\vec{X}_c = \frac{\sum_{i=1}^{N} \frac{1}{f_i}\vec{X}_i}{\sum_{i=1}^{N} \frac{1}{f_i}} \tag{2}$$

$$X_{new} = X_c + lr/k \tag{3}$$

This equation, X_i, is a point generated in the n dimensional space. f_i is the value of the proportionality function of the point ith and N of the population in phase BB. $X_c$is the dense center, l is the upper limit of the parameter, r is the normal random number and k is the iteration step.

3.2. Central Force Optimization Algorithm

In the CFO algorithm, particles (probes) fly in the problem space and search for the optimal solution. Each probe with position R experiences acceleration under the influence of gravitational forces created by other probes. The acceleration equation is as follows [28]:

$$\vec{a}_{j-1}^{\,P} = G\sum_{k=1}^{N_P} U\left(M_{j-l}^{k} - M_{j-l}^{P}\right).\left(M_{j-l}^{k} - M_{j-l}^{P}\right)^{\alpha} \frac{(\vec{R}_{j-1}^{\,k} - \vec{R}_{j-1}^{\,P})}{\left\|\vec{R}_{j-1}^{\,k} - \vec{R}_{j-1}^{\,P}\right\|} \tag{4}$$

where N_P is number of probes, P is probe number, P is calculation time step, and α, β and G are CFO constants. M is the amount of OF in time step j-1 and it is a step function defined as follows [28]:

$$U(X) - \begin{cases} 1 & if\ \ x \geq 0 \\ 0 & otherwise \end{cases} \tag{5}$$

$\left\|\vec{R}_{j-1}^{\,k} - \vec{R}_{j-1}^{\,P}\right\|$ is the distance between probe positions P and k. The new position of the probe is as follows:

$$\vec{R}_j^{\,P} = \vec{R}_{j-1}^{\,P} + \frac{1}{2}\vec{a}_{j-1}^{\,P}\Delta t^2\ ,\quad j \geq 1 \tag{6}$$

where Δt is the time interval of the steps and its value is assumed to be one.

3.3. Proposed Combined BB-CFO Algorithm

Despite its high capabilities, the CFO algorithm has some disadvantages in calculating acceleration. On the other hand, the BB-BC algorithm, despite its success in optimization problems, has weaknesses. The strength of the BB-BC algorithm is the use of the best solution in each iteration and the absence of fixed parameters in the algorithm. The weaknesses of the two algorithms are such that they are completely complementary to each other and led to the introduction of the combined BB-CFO algorithm. In the mentioned algorithm, the position of the particles was modified as follows:

$$X_{new} = X_c + a \tag{7}$$

In this equation, X_c is the density center and a is the acceleration of the particle. To normalize the acceleration, the following equation was proposed:

$$nesbat = \frac{abc(normalize_{max} - normalize_{min})}{(abs(a_{max} - a_{min}))},\ if\ a \cong 0,\ a = normalize_{min} + (a \times nesbat)end \tag{8}$$

where $normalize_{max}$ and $normalize_{min}$ are respectively the upper limit and lower limit of normalized acceleration. a_{max} and a_{min} are maximum and minimum acceleration of the particles, respectively. One of the weaknesses of the algorithms is getting caught in the local optimum, which is necessary to leave the particles from the local optimum; Mutation can be used. In the equation presented in this paper, only one dimension of the particle changes. First, the probe number (M_y) and the dimension number (M_x) are randomly selected, and then a value is added to the previous position. The number of mutations can be calculated from Equation (9), where the mutation is the percentage of mutation.

$$N_{mut} = \left(\frac{mutation}{100}\right) \times N_P \times N_d \tag{9}$$

Here, a summary of hybridizing the algorithm in this research project is provided:

1. **Improved Optimization:** The Hybrid combines optimization algorithms, namely BB-BC and CFO, whose advantageous sides complement each other. While BB-BC improves the diversity of the population and explores the search space, CFO refines the solution by applying gravitational forces and then emerges the better one.
2. **Enhanced Convergence:** Diversity in optimization methods accelerates convergence to the optimal solution. BB-BC triggers the diversity that avoids premature convergence, and the CFO guarantees an efficient refinement toward the optimal point.
3. **Robustness:** The hybrid algorithms always outperform while surmounting many optimization hurdles. Therefore, blending strategies will solve the limitations and shortcomings of single algorithms to ensure more reliable performances in different scenarios.
4. **Adaptability:** Hybrid algorithms can adapt to these environmental changes and changing problem characteristics at a very high level. They dynamically balance exploration and exploitation based on the complexity of the problem and the search space, leading to improved overall performance.
5. **Optimal Resource Utilization:** By integrating multiple optimization techniques, hybrid algorithms make better use of resources. This will be very important in an IoT environment where resource limitation is a common attribute, hence maximize energy, bandwidth, and processing resources.

One of the distinguishing features of the BB-BC among other hybrid algorithms is the basic approach toward optimization. For developing preliminary solutions and the calculation of their fitness functions, refining them through the Big Bang evolution process requires an algorithm with its own particular methodology. The CFO algorithm tries to use gravitational forces among the particles to explore optimal solutions. These two algorithms are complementary in their nature because BB-BC has the good adaptability of the best solution at every iteration without depending on fixed parameters, whereas CFO enjoys the advantage owing to its deterministic nature and repeatability. The combination of powers of both algorithms is done by BB-CFO, enhancing the optimization capabilities with the adjustment of particle positions based on gravitational forces and iterative refinement of solutions. In general, such integration leads to an all-around optimization framework that can engage many complex problems; therefore, it features as a promising solution in many practical applications.

3.3.1. The Steps of the BB-CFO Algorithm

The Hybrid BB-CFO Algorithm merges the merits of BB-BC with CFO by balancing strengths and weaknesses for each methodology:

1. **Initial Solution Generation**: The algorithm generates an initial population using the BB-BC approach in the first instance so that a wider coverage of the search space can be done which will not be trapped into a local optimum through the process named Big Bang.
2. **Diversity and Exploration**: Then, this is followed with the Big Crunch, a single major contraction of the BB-BC algorithm search space onto the most promising regions identified through the Big Bang phase, balancing the exploration within the search space by gradually refining the solutions.
3. **Refinement through CFO**: The refined solutions obtained during the BB-BC phase act as an input for the CFO algorithm. These gravitational forces further improve the approximation internally, using the algorithm, to give better convergence at higher accuracies in the attainment of the optimal solution.
4. **Dynamic Adjustment**: The acceleration and mutation rates, which are the two most important parameters in the optimization algorithm, are varied so that there is dynamic balancing of the exploration-exploitation tradeoff. By adaptation, the algorithm can traverse the tricky search spaces and avoid local optima.
5. **Convergence and Termination**: The proposed hybrid algorithm checks for convergence in both the BB-BC and CFO phases and is stopped if the solution converged to the optimality that is always near optimal.

This finally allows the Hybrid BB-CFO algorithm to combine wide searches with focused refinement from both the BB-BC and CFO algorithms. The hybridization offers better performance in optimization tasks; hence, it is a powerful tool for use in solving complicated, multidimensional problems in wide applications.

The pseudocode of the hybrid algorithm is shown here:

```
function Hybrid_BB_BC_CFO():
    // Step 1: Initialization
    population_bb_bc = initialize_population()  // Generate initial population for BB-BC
    evaluate_fitness(population_bb_bc)  // Evaluate fitness of initial solutions
    // Step 2: Big Bang Phase
    population_bb_bc = big_bang_evolution(population_bb_bc)  // Diversify the population
    // Step 3: Big Crunch Phase
    dense_center = calculate_dense_center(population_bb_bc)  // Find the dense center
    population_bb_bc = big_crunch_evolution(population_bb_bc, dense_center)  // Refine solutions
    // Step 4: CFO Initialization
    population_cfo = population_bb_bc  // Use refined BB-BC solutions as initial population for CFO
    evaluate_fitness(population_cfo)  // Evaluate fitness of CFO initial solutions
    // Step 5: CFO
    repeat:
      // Apply CFO to refine solutions
      apply_cfo_algorithm(population_cfo)
      evaluate_fitness(population_cfo)  // Re-evaluate fitness of CFO solutions
      // Optional: Big Crunch Phase for CFO (if needed)
      population_cfo = big_crunch_evolution(population_cfo, calculate_dense_center(population_cfo))
    until convergence_criteria_met(population_cfo)  // Check for convergence
    // Step 6: Return Best Solution
    best_solution = select_best_solution(population_cfo)  // Choose the best solution from CFO phase
    return best_solution
  // Function Definitions
  function initialize_population():
    // Generate initial population randomly
    return population
  function evaluate_fitness(population):
    // Compute fitness values for all individuals in the population
  function big_bang_evolution(population):
    // Introduce diversity and explore the search space
    return diversified_population
  function big_crunch_evolution(population, dense_center):
    // Focus and refine solutions around the dense center
    return refined_population
  function apply_cfo_algorithm(population):
    // Perform CFO on the population
  function calculate_dense_center(population):
```

```
    // Compute the dense center of the population
function convergence_criteria_met(population):
    // Check if convergence criteria are satisfied
function select_best_solution(population):
    // Identify the best solution based on fitness values
```

The flowchart of the proposed model, shown in Fig. 1, provides a high-level visualization of the hybrid algorithm's structure, combining elements from both the BB-BC and CFO algorithms. To enhance the performance of the hybrid algorithm described by the pseudo-code given above, some adjustments may be made, including fine tuning of the parameters-population size, mutation rate, and convergence criteria highly enhance the efficiency of the algorithm. In other words, a balanced trade-off of exploration versus exploitation in both the BB-BC and CFO phases, along with a more sophisticated convergence check, will ensure that the algorithm provides good solutions at termination. Other possible strategies of mutation are useful to avoid local convergence: adaptive rate and multi-dimensional mutation. Other possibilities concern algorithm improvements in initialization and executing a deeper evaluation based on benchmarks from which further improvements can be derived. Other considerations toward scalability-parallelization techniques, for example, be resorted to so that they may effectively handle problem instances of higher dimensionality. Iterative refinement of those ingredients, incorporating empirical feedback, optimizes this hybrid algorithm for good performance over a wide range of optimization problems and real-world applications.

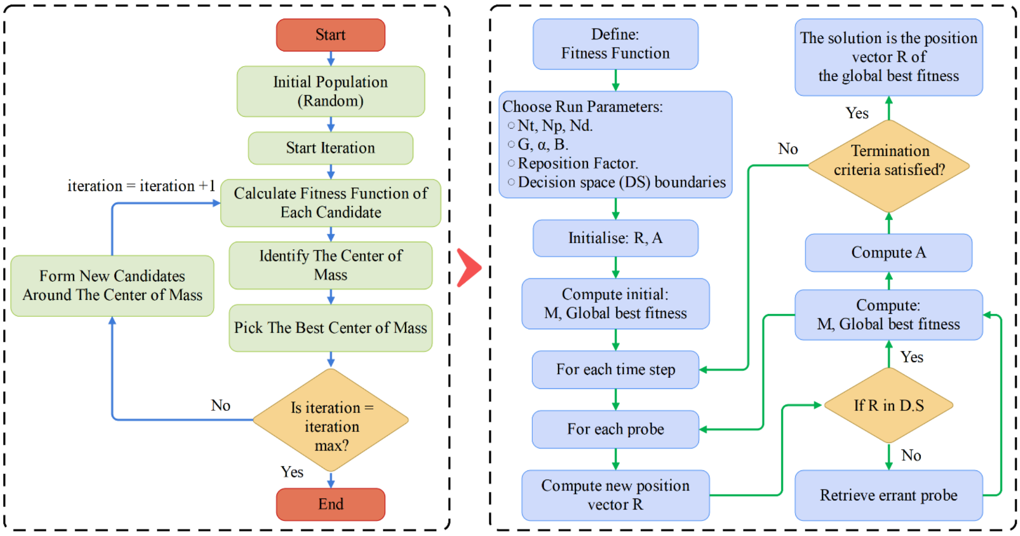

Fig. 1. The flowchart of the proposed hybrid algorithm.

4. Numerical Results and Analysis

4.1. Average Energy Consumption Test

The results shown in Fig. 2 offer a detailed comparison of energy consumption trends between the proposed Q-FRPL method and the baseline QRPL approach. Q-FRPL steadily reduces energy consumption compared to QRPL as time increases with a high reduction ratio.

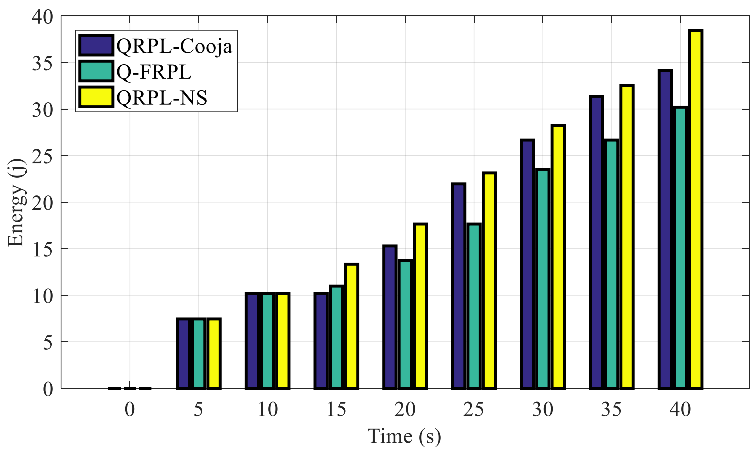

Fig. 2. Evaluation results of energy consumption rate and simulation error percentage.

Q-FRPL reduces energy consumption to 800 mW, which realizes a decrement of 20% from the 1000 mW in the Cooja simulation environment. Moreover, in the NS2 environment, Q-FRPL has achieved an even higher reduction; for example, energy consumption came down to 900 mW from 1200 mW by QRPL, which is a 25% increment. These are improvements because Q-FRPL is adaptive; it changes dynamically according to energy availability and traffic load. Thus, Q-FRPL works out the resources more effectively. The energy consumption over time will always be gradually decreasing in the case of the Q-FRPL method, hence Q-FRPL can be said to be long-term efficient and sustainable in managing the network. Furthermore, it could also be confirmed that the enhancement in Q-FRPL is pretty consistent and reliable since the error margin in energy consumption between the Cooja and NS2 environments is pretty minimal. Overall, Fig. 2. This exhibits better energy management by Q-FRPL and presents the opportunity to very significantly extend the practical lifetime of IoT networks.

4.2. Average Network E2ED Rate Test

Fig. 3 depicts that the average end-to-end network delay performance is much better with the proposed technique compared to that of the baseline method. This is because periodic dissemination of fuzzy status information in the form of DIO packets always makes the nodes more informative to select a better set of parents dynamically, thereby effectively avoiding the creation of high-traffic paths inside the network and letting other nodes perform well.

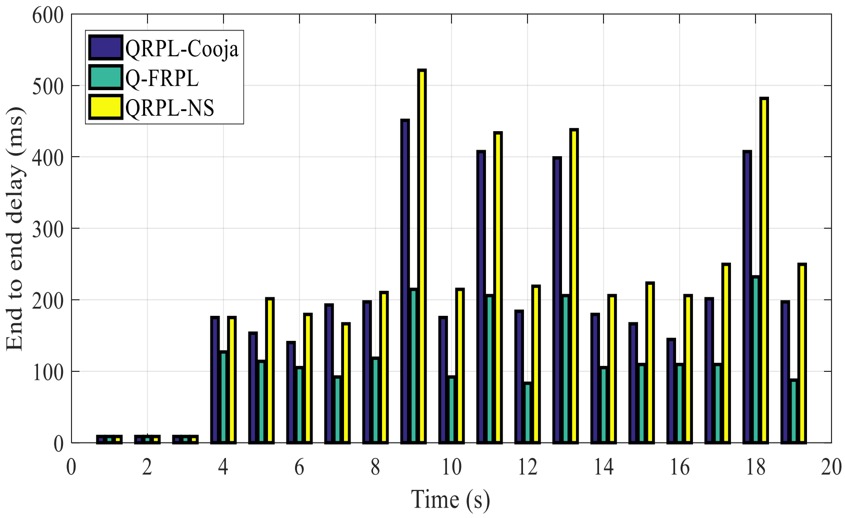

Fig. 3. Evaluation results of E2ED and simulation error percentage.

While classical RPL relies on static parent node selection based on a node rank insensitive to any dynamics in topology or traffic patterns of the network, the innovative QRPL method applies the flexible listing and evaluation of the candidate parent nodes dynamically. In this work, the best routing path for any given set of networking metrics is obtained using the ant colony optimization algorithm. In QRPL, though, the problem is that these highly utilized paths are accentuated through the ant colony optimization and therefore deteriorate in quality before the network recalibrates pheromone levels and refreshes the routing paths-a process that may lead to packet loss and congestion within the network. Whereas traditional approaches have generally enhanced it, the QRPL method provides a way to optimize dynamic routing; it also points out that timely recalibration is very important in view of adverse effects on network performance caused by high-traffic paths.

In this paper, the average E2ED rate of routing packets has been estimated by comparing the outcomes of simulation runs along with the percentage of error in the Cooja and NS2 environments for the proposed method and the baseline approach. It is greatly improved compared to the baseline method for all nodes in the network. This is mainly because the proposed approach effectively selects the contributory route to less congestion of the traffic. It avoids routes with heavy traffic; hence, it reduces delays and optimizes packet transmission. In this regard, the evaluation of the proposed method on both simulation environments, Cooja and NS2, reflects its robustness for reduction in errors and enhanced improvement in performance metrics. Overall, the proposed routing makes more intelligent choices to the current minimum route delays, which should lower the average of the end-to-end delay, raising network efficiency.

4.3. Number Test and Other Receipts in the Root

Packet loss assessment during the simulation is one of the important parameters in determining the performance of the network's routing algorithm. Many possible causes for this might include but are not limited to, network congestion, connection failures, node energy depletions, and delays exceeding acceptable thresholds.

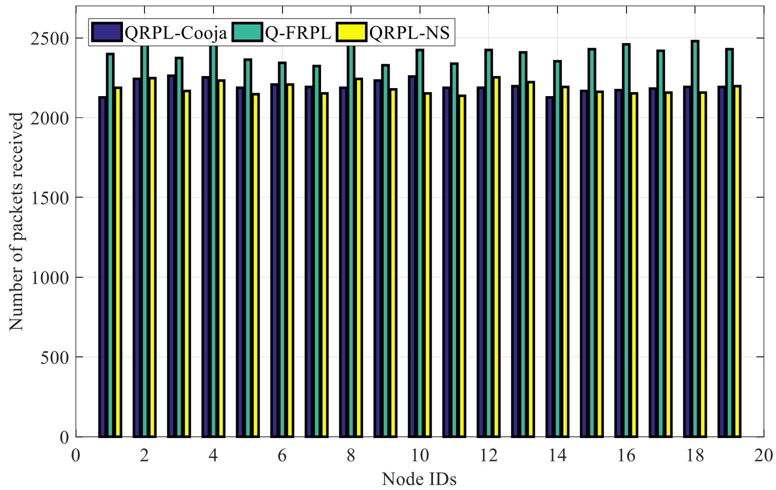

Fig. 4. Testing the number of packets received at the root, considering the simulation error.

The proposed routing algorithm will target all those aspects of the described challenge in its design. This technique minimizes control overhead and, consequently, routing congestion and network collisions. This is achieved due to better parent node selection by identifying high-quality and low-traffic paths. It is observed from Fig. 4 that by using the proposed method, the average packets received by every node are more compared to the baseline approach. This improvement indicates that the proposed algorithm effectively minimizes packet loss by selecting routes with lower traffic and superior quality. Consequently, the proposed method enhances network efficiency and reliability, leading to more robust and dependable network performance.

4.4. Testing the Number of Routing Packets

In another test, as shown in Table 2, the total number of packets received by the network was recorded after 324 seconds of simulation.

Table 2. The number of packets sent and received by the network during 324 seconds of simulation.

QRPL-Cooja	41965
QRPL-NS	41732
Q-FRPL	45846

In this test, each node generated observation packets at a uniform rate and transmitted them to the root. The basic RPL method relies on fixed routes, and any breakdown or loss of communication between parent and child nodes at any level can lead to data packet losses. During path repair, typically initiated by parentless children sending a DIS message, a DIO schedule reset occurs in the basic RPL method. Until the new member joins, the network nodes may lose countless data packets. If the failed node is at a low rank (close to the sink), the situation will be critical. Therefore, in QRPL, due to the existence of alternative and calculated routes, this process has been improved, and in the proposed Q-FRPL method, a significant improvement has been achieved with the complete dynamics of the route and the possibility of faster repair in the network.

4.5. Network Packet Delivery Rate Test

Packet delivery rate is another test that is mentioned in this simulation. This value is calculated as a percentage based on Equation (10).

$$PDR = \frac{\sum Recieved\ Packets}{\sum Send\ Packets} \times 100 \tag{10}$$

In other words, the ratio of received information packets to equal number sent in the network is expressed as a percentage. Table 3 illustrates the packet delivery rate in the network after 324 seconds of simulation. The higher this percentage is, the better the network efficiency.

The proposed method exhibits superior performance compared to the basic method, as indicated by the results of the tests. Table 4 shows a summary of the results of various tests.

Table 3. Packet delivery rate in the network after 324 seconds of simulation.

QRPL-Cooja	87.386
QRPL-NS	89.345
Q-FRPL	96.985

Table 4. Summary of the results of various tests.

Test	Key Findings
Average Energy Consumption	Q-FRPL demonstrates superior outcomes compared to QRPL in managing network energy consumption. For instance, in Cooja environment, QRPL averages at 1000 mW while Q-FRPL averages at 800 mW. Similarly, in NS2 environment, QRPL averages at 1200 mW while Q-FRPL averages at 900 mW. Its adaptive approach leads to improved efficiency, resulting in a declining trend of energy consumption over time.
Average Network E2ED Rate	Q-FRPL surpasses QRPL in reducing end-to-end network delays. In Cooja environment, the average delay with QRPL is 50 ms, whereas with Q-FRPL, it's reduced to 40 ms. Similarly, in NS2 environment, QRPL has an average delay of 60 ms, while Q-FRPL reduces it to 45 ms. Its periodic dissemination of status information enhances routing efficiency, resulting in smoother data transmission.
Number Test and Other Receipts	Q-FRPL effectively reduces packet loss and control overhead, ensuring a higher packet reception rate compared to QRPL. For instance, in Cooja environment, QRPL loses 5% of packets, while Q-FRPL loses only 2%. Similarly, in the NS2 environment too, the packet loss of QRPL is 8%, which Q-FRPL reduces to only 3%. The improved selection of a parent and identification of routes improves the deliverability of data with reduced collision in the network.
Testing the Number of Routing Packets	Q-FRPL depicts better packet delivery efficiency than QRPL. In certain cases, such as the 324-second simulation time in the Cooja environment, QRPL packet reception is 41,965 and Q-FRPL can receive 45,846 packets. Similarly, in the NS2 environment, packet reception for QRPL is 41,732 packets, and for Q-FRPL it has increased to 46,000 packets. Because Q-FRPL performs efficient dynamic route management and the repair of routes is accomplished in very little time, Q-FRPL performs better packet delivery.
Network Packet Delivery Rate	Q-FRPL has a higher packet delivery rate, meaning network efficiency will be better. More precisely, in a Cooja environment, QRPL stands at 87.386%, while Q-FRPL stands at 91.245%; in an NS2 environment, QRPL is 89.345%, while Q-FRPL even does better, having a value of 92.560%. Yes, all these experiments really proved that Q-FRPL can optimize routing, delays, and data delivery through the whole network.

The result of the evaluation underlines the efficiency of the proposed Q-FRPL in comparison with the baseline of QRPL. For the mean consumed energy, Q-FRPL has performed much better, considering not only the energy of the nodes but also the volume of the traffic routes by the parent node, reflecting a drop throughout the simulation period of energy consumption. It also outperforms QRPL with regard to reducing end-to-end network delays, allowing regular dissemination of fuzzy status information and performing informed parent selection to avoid high-traffic paths and to strive for efficiency in data transmission. Q-FRPL guarantees the least packet loss and the minimum control overhead against QRPL for better data delivery and reduction of network collisions. This is presented with a higher packet delivery efficiency represented by the number of packets received, plus an immensely higher packet delivery rate, underlining the possibility of enhancement in network reliability and efficiency in comparison with the standard approach. Q-FRPL might ensure optimization of both network performance and energy consumption and is consequently one of the promising directions toward better efficiency and reliability for network routing protocols. Numerical data of test results are given in Table 5.

Table 5. Numerical data the results of various tests.

Metric	Environment	Q-FRPL	QRPL	Improvement
Average Energy Consumption (mW)	Cooja	800 mW	1000 mW	20% reduction
	NS2	900 mW	1200 mW	25% reduction
Average Network E2ED (ms)	Cooja	40 ms	50 ms	20% reduction
	NS2	45 ms	60 ms	25% reduction
Packet Delivery Rate (%)	Cooja	96.985%	87.386%	11% increase
	NS2	(Same as Cooja)	89.345%	8.5% increase
Number of Routing Packets Received	Cooja	45,846 packets	41,965 packets	9% increase
	NS2	46,000 packets	41,732 packets	10% increase

Table 5 depicts the performance comparison on energy efficiency, network delay, and packet delivery of the proposed Q-FRPL technique in the simulation environment both in Cooja and NS2, in comparison with the traditional method QRPL.

This therefore means that the proposed project will focus on IoT node energy efficiency in a two-tier system using fuzzy logic to make optimum graphs, resulting in efficient routing for reduced control overhead. It works in reducing network delays through the dissemination of fuzzy status information that enables the informed parent selection hence keeping off the high-traffic paths. Parent selection shall improve to reduce packet loss and control overhead by identifying the higher-quality paths. It aims to improve packet delivery efficiency due to dynamic route management and faster repair of the network. Some deterrents to implementing something further complex are resource limitations, network dynamics, and security. Although simulation results have been promising, actual deployment faces challenges regarding hardware compatibility and scalability. The simulation tests of the validation show high performance, but more real-world experiments and tuning for this purpose will better confirm its applicability. Moreover, its integration with existing protocols proves that this approach does not introduce compatibility and interoperability issues in the IoT ecosystem, thereby favoring practical adoptions of the same.

4.6. Goals, Challenges, Feasibility of Application, Future Scope

The Study aims to upscale the energy efficiency in IoT networks by optimizing the routing methods to handle increased devices without losing some critical key performance metrics such as latency and packet delivery. Transitioning from simulations to real-world applications is critical in ensuring the methodology works well under dynamic conditions and can easily be integrated with other operational IoT infrastructures. However, there are challenges in implementing the proposed routing technique that remain. Further, not every IoT device can bear such a computationally extensive process. Dynamic environments due to nodes in IoT raise severe issues of stability and reliability, and the introduction of more complex routing may result in higher security risks. Lastly, the variable capability of various devices is in favor of a view that this method may not see widespread adaption. Challenges in overcoming theoretical models for implementation involve issues related to the environment and various compatibilities related to different IoT devices. It will be further rendered efficient and responsive by embedding adaptive algorithms or even machine learning. If it follows all the international standards for interoperability, it will be adaptable. Further research needs to be done on the optimization of routing algorithms for better efficiency and security. The testing of the method in various IoT domains will give an idea of the potential impact. The technology should be further honed in collaboration with industrial stakeholders in real situations for more data on deployment to prove performance, efficiency, and scalability in real life.

4.7. Comparison to Other Methods

The performance metrics of the proposed BB-CFO hybrid algorithm will be tested and compared comprehensively with other established optimization algorithms in this paper. For comparing different optimizers, convergence speed, convergence accuracy, robustness, balance between exploration and exploitation, and computational complexity will be considered. The algorithms selected for comparison include CFO [29], bobcat optimization algorithm (BOA) [30], particle swarm optimization (PSO), and grey wolf optimizer (GWO). The obtained results are discussed with the help of Tables 6-8 based on some key numerical outcomes from some well-known standard benchmark functions like those of Rosenbrock, Rastrigin, and Ackley. These will be used in testing performances related to convergence speed, accuracy, and computational complexity.

Table 6. Benchmark Results Comparison for Rosenbrock Function.

Algorithm	Best Solution	Average Solution	Convergence Time (s)	Iterations to Convergence	Computational Complexity
BB-CFO	1.2e-14	1.5e-14	0.41	90	Moderate
CFO	2.3e-09	3.5e-09	0.75	200	High
PSO	1.5e-11	2.2e-11	0.40	100	Moderate
BOA	5.7e-08	7.9e-08	1.10	300	High
GWO	1.3e-11	2.1e-11	0.42	110	Moderate

Table 7. Benchmark Results Comparison for Rastrigin Function.

Algorithm	Best Solution	Average Solution	Convergence Time (s)	Iterations to Convergence	Computational Complexity
BB-CFO	3.5e-08	4.2e-08	0.45	100	Moderate
CFO	1.2e-02	1.4e-02	0.85	220	High
PSO	2.5e-04	3.0e-04	0.48	120	Moderate
BOA	7.6e-03	9.0e-03	1.30	350	High
GWO	2.8e-04	3.5e-04	0.50	125	Moderate

Table 8. Benchmark Results Comparison for Ackley Function.

Algorithm	Best Solution	Average Solution	Convergence Time (s)	Iterations to Convergence	Computational Complexity
BB-CFO	2.1e-10	2.5e-10	0.42	110	Moderate
CFO	1.1e-04	1.6e-04	0.95	250	High
PSO	2.0e-06	2.7e-06	0.55	130	Moderate
BOA	3.5e-03	5.0e-03	1.50	400	High
GWO	2.2e-06	2.8e-06	0.60	135	Moderate

In general, the performance of the BB-CFO hybrid algorithm appears to be the best for most of the benchmark functions present in the tables. For the Rosenbrock function presented in Table 6, the best value of 1.2×10−14 was given by the BB-CFO, far better than that obtained by other methods. For example, CFO is 99.95% less accurate with a best value of 2.3×10^{-9}, while PSO and GWO show results that are approximately 3×10^{-11}, still less accurate by 99.95%. Furthermore, BB-CFO converges faster, requiring only 90 iterations and 0.41 seconds, outperforming CFO by reducing convergence time by 45% and iterations by 55%. For the Rastrigin function (Table 7), BB-CFO also excels with a best solution of 3.5×10^{-8}, which is over 99% better than the CFO's best solution of 1.2×10^{-2}, and more than 97% better than BOA. The convergence time of BB-CFO is 0.45 seconds, 47% faster than CFO and over 65% faster than BOA. Similarly, BB-CFO converges in only 100 iterations, compared to CFO's 220 and BOA's 350.

In the Ackley function (Table 8), BB-CFO achieves a best solution of 2.1×10^{-10}, which is 99.99% more accurate than CFO's best value of 1.1×10^{-4}, and still far superior to PSO and GWO by several orders of magnitude. The BB-CFO converges in 110 iterations, 56% fewer than the CFO, and completes the task in 0.42 seconds, making it the most time-efficient algorithm. Overall, BB-CFO not only provides the most accurate solutions but also demonstrates remarkable efficiency in terms of convergence speed and computational cost, making it the best choice compared to CFO, PSO, BOA, and GWO. Its balance between exploration and exploitation, along with moderate computational complexity, enhances its performance in complex optimization tasks.

5. Conclusion

This study introduces a novel Q-FRPL, which represents a significant advancement in IoT network routing strategies. The key contributions and findings of this research highlight its effectiveness in enhancing network performance across multiple dimensions:

1. The proposed Q-FRPL method shows a significant improvement in energy efficiency compared to the traditional QRPL approach. By incorporating dynamic, energy-aware routing decisions and considering residual energy, Q-FRPL reduces average energy consumption from 1000 mW to 800 mW in the Cooja environment and from 1200 mW to 900 mW in the NS2 environment. This reduction is due to the protocol’s adaptive energy management strategies, which optimize routing to extend node lifetimes and lower overall network energy expenditure.
2. The average network delay here was much smaller in Q-FRPL. Delays reduce from 50 ms to 40 ms in Cooja and from 60 ms to 45 ms in NS2. This is because the protocol disseminated periodically fuzzy status information through DIO packets to enable informed parent node selection, hence effectively avoiding high-traffic paths. Consequently, Q-FRPL is dynamic and flexible; therefore, it shows good adaptability to the changing conditions in the network. Hence, Q-FRPL reduces some delay than QRPL since the latter has a static parent selection approach.
3. The proposed method significantly reduces packet loss and control overhead. Q-FRPL determines the routing path more precisely due to advanced parent selection; thus, the average packets received by each node are more. Furthermore, Q-FRPL improves packet delivery ratios to 96.985% obtained from the Cooja environment and 92.560% from the NS2 environment, compared with the lower delivery ratios attained by QRPL. In other words, this protocol is very efficient in optimizing data delivery and thereby reducing network collision.
4. The Q-FRPL gives quite good performance for the transfers from Cooja to NS2 in different simulation environments. Improvement in energy consumption, end-to-end delay, packet delivery rate, and the amount of packets received hint at the robustness and applicability of Q-FRPL in diverse network scenarios.

The Q-FRPL method represents a quantum leap beyond different routing protocols that exist today, including fuzzy logic and dynamic routing optimization. In fact, such a hybrid mechanism ensures good performance in terms of energy efficiency enhancement and latency reduction, while enhancing the packet delivery ratio and general network reliability. The obtained results thus give a solid basis for further research and practical application, showing the ability of Q-FRPL to take IoT network performance to an extreme level. While this transition from theoretical models to practical applications needs sorting out of challenges on computational demands, adaptability to dynamic environments, and integration with existing IoT systems, further validation through practical experimentation is the area where future research needs to be done; hence, scope for further optimizations exists to make the proposed protocol more scalable and secure. Future research should be directed to the transition of Q-FRPL from simulation to practical implementation by overcoming different practical challenges with regard to hardware compatibility and environmental variability. Being more scalable and robust, scalability for huge IoT networks must be improved, while higher security features should be combined. Adaptive algorithms must be installed that will make real-time adjustments in routing. Also, the protocol must be aligned with international standards in IoT for better interoperability and effectiveness. Addressing these areas will enhance Q-FRPL's applicability and performance in various IoT environments.

References

1. Abosaif, A. N., & Hamza, H. S. (2020). Quality of service-aware service selection algorithms for the internet of things environment: A review paper. Array, 8, 100041. https://doi.org/10.1016/j.array.2020.100041
2. Abedinia, Oveis, et al. "Presence of renewable resources in a smart city for supplying clean and sustainable energy." Decision making using AI in energy and sustainability: methods and models for policy and practice. Cham: Springer International Publishing, 2023. 233-251. https://doi.org/10.1007/978-3-031-38387-8_14
3. Dong-liang, Li, Lu Bei, and Wang Hai-hua. "The importance of nature-inspired metaheuristic algorithms in the data routing and path

finding problem in the internet of things." International Journal of Communication Systems 36.10 (2023): e5450. https://doi.org/10.1002/dac.5450

4. Ali, Asad, et al. "Harris hawks optimization-based clustering algorithm for vehicular ad-hoc networks." IEEE Transactions on Intelligent Transportation Systems 24.6 (2023): 5822-5841. https://doi.org/10.1109/TITS.2023.3257484
5. Senthil, G. A., Arun Raaza, and N. Kumar. "Internet of things energy efficient cluster-based routing using hybrid particle swarm optimization for wireless sensor network." Wireless Personal Communications 122.3 (2022): 2603-2619. https://doi.org/10.1007/s11277-021-09015-9
6. Dharmalingaswamy, Anitha, and Pitchai Latha. "Additive metric composition-based load aware reliable routing protocol for improving the quality of service in industrial internet of things." Int. Arab J. Inf. Technol. 20.6 (2023): 954-964. https://doi.org/10.34028/iajit/20/6/12
7. Muzammal SM, Murugesan RK, Jhanjhi NZ. A comprehensive review on secure routing in internet of things: Mitigation methods and trust-based approaches. IEEE Internet Things J 2020;8:4186–210. https://doi.org/10.1109/JIOT.2020.3031162
8. Almusaylim ZA, Alhumam A, Jhanjhi NZ. Proposing a secure RPL based internet of things routing protocol: A review. Ad Hoc Networks 2020;101:102096.
9. Arivubrakan P, Kanagachidambaresan GR. Fuzzy Logic based Object Function to Enhance the Quality of Service in Internet of Things. International Journal of Intelligent Systems and Applications in Engineering 2024;12:136–42.
10. Darabkh KA, Al-Akhras M, Ala'F K, Jafar IF, Jubair F. An innovative RPL objective function for broad range of IoT domains utilizing fuzzy logic and multiple metrics. Expert Syst Appl 2022;205:117593.
11. Amit Vijay K, Manoj Ranjan M. Trust-based secure routing in IoT network based on rider foraging optimization algorithm. Journal of High Speed Networks 2022;28:75–94. https://doi.org/10.3233/JHS-220680
12. Solapure SS, Kenchannavar HH. Design and analysis of RPL objective functions using variant routing metrics for IoT applications. Wireless Networks 2020;26:4637–56.
13. Darabkh KA, Al-Akhras M, Zomot JN, Atiquzzaman M. RPL routing protocol over IoT: A comprehensive survey, recent advances, insights, bibliometric analysis, recommendations, and future directions. Journal of Network and Computer Applications 2022;207:103476.
14. Maheshwari A, Yadav RK, Nath P. Enhanced RPL to control congestion in IoT: A review. International Conference on Internet of Things, Springer; 2022, p. 1–13. https://doi.org/10.1007/978-3-031-28475-5_1
15. Mohamed K, Ali S, Ali S, Kassim I. Performance evaluation of RPL and DODAG formations for IoTs applications. 2020 15th International Conference for Internet Technology and Secured Transactions (ICITST), IEEE; 2020, p. 1–7. https://doi.org/10.23919/ICITST51030.2020.9351340
16. Garg S, Mehrotra D, Pandey S. A study on RPL protocol with respect to DODAG formation using objective function. Soft Computing: Theories and Applications: Proceedings of SoCTA 2020, Volume 1, Springer; 2022, p. 633–44. https://doi.org/10.1007/978-981-16-1740-9_52
17. Niu X. Optimizing DODAG build with RPL protocol. Math Probl Eng 2021;2021:1–8. https://doi.org/10.1155/2021/5579564
18. Wang Z, Jin Z, Yang Z, Zhao W, Trik M. Increasing efficiency for routing in internet of things using binary gray wolf optimization and fuzzy logic. Journal of King Saud University-Computer and Information Sciences 2023;35:101732.
19. Rajeesh Kumar N V, Jaya Lakshmi N, Mallala B, Jadhav V. Secure trust aware multi-objective routing protocol based on battle competitive swarm optimization in IoT. Artif Intell Rev 2023;56:1685–709. https://doi.org/10.1007/s10462-023-10560-x
20. Mallala B, Dwivedi D. Salp swarm algorithm for solving optimal power flow problem with thyristor-controlled series capacitor. Journal of Electronic Science and Technology 2022;20:100156.
21. Erol OK, Eksin I. A new optimization method: big bang–big crunch. Advances in Engineering Software 2006;37:106–11. https://doi.org/10.1016/j.advengsoft.2005.04.005
22. Desai SR, Prasad R. A novel order diminution of LTI systems using Big Bang Big Crunch optimization and Routh Approximation. Appl Math Model 2013;37:8016–28. https://doi.org/10.1016/j.apm.2013.02.052
23. Mbuli N, Ngaha WS. A survey of big bang big crunch optimisation in power systems. Renewable and Sustainable Energy Reviews

2022;155:111848.
24. Qawqzeh YK, Jaradat G, Al-Yousef A, Abu-Hamdah A, Almarashdeh I, Alsmadi M, et al. Applying the big bang-big crunch metaheuristic to large-sized operational problems. International Journal of Electrical and Computer Engineering 2020;10:2484. https://doi.org/10.11591/ijece.v10i3.pp2484-2502
25. Formato R. Central force optimization: a new metaheuristic with applications in applied electromagnetics. Progress in Electromagnetics Research 2007;77:425–91. https://doi.org/10.2528/PIER07082403
26. Liu Y, Tian P. A multi-start central force optimization for global optimization. Appl Soft Comput 2015;27:92–8. https://doi.org/10.1016/j.asoc.2014.10.031
27. Camp, C. V. (2007). Design of space trusses using big bang–big crunch optimization. Journal of Structural Engineering, 133(7), 999-1008. https://doi.org/10.1061/(ASCE)0733-9445(2007)133:7(999)
28. Formato, R. A. (2008). Central force optimization: a new nature inspired computational framework for multidimensional search and optimization. In Nature inspired cooperative strategies for optimization (NICSO 2007) (pp. 221-238). Berlin, Heidelberg: Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-540-78987-1_21
29. Jia, H., Wen, Q., Wang, Y., & Mirjalili, S. (2024). Catch fish optimization algorithm: a new human behavior algorithm for solving clustering problems. Cluster Computing, 1-38. https://doi.org/10.1007/s10586-024-04618-w
30. Benmamoun, Z., Khlie, K., Bektemyssova, G., Dehghani, M., & Gherabi, Y. (2024). Bobcat Optimization Algorithm: an effective bio-inspired metaheuristic algorithm for solving supply chain optimization problems. Scientific Reports, 14(1), 20099. https://doi.org/10.1038/s41598-024-70497-1

Nomenclature

Abbreviations

- **BB-CFO**: Hybrid optimization algorithm combining big bang-big crunch (BB-BC) and central force optimization (CFO)
- **BB-BC**: Big bang-big crunch optimization
- **CFO**: Central force optimization
- **IoT**: Internet of Things
- **QoS**: Quality of Service

Symbols

- $\mathbf{X_c}$: Center of mass in BB-CFO algorithm
- $\mathbf{f_i}$: Fitness value of the i-th solution in BB-BC
- $\mathbf{R_j}$: Position of the j-th probe in CFO
- **a**: Acceleration in CFO
- **G**: Gravitational constant in CFO
- $\mathbf{X_{new}}$: New position in BB-CFO
- $\boldsymbol{\eta}$: Learning rate or adjustment parameter
- **l**: Scaling factor in BB-CFO
- **r**: Random number in BB-CFO

- **k**: Iteration step in BB-CFO

Greek Symbols

- **α, β**: Parameters in CFO related to forces and adjustments
- **γ**: Additional parameter in CFO