Routing Overhead Aware Optimal Cluster based Routing Algorithm for IoT Network using Heuristic Technique

Srinivasulu M[1], Dr. Shiva Murthy G[2]
Dept. of Master of Computer Application, University BDT College of Engineering,
Davanagere, Karnataka, India[1]
Dept. of Computer Science and Engineering, VTU PG Center, Muddenahalli,
Chikkaballapura, Karnataka, India[2]

Abstract*—Globally, billion of devices in heterogeneous networks are interconnected by the Internet of Things (IoT). IoT applications require a centralized decision-making system due to the failure-prone connectivity and high latency of such a system. Low-latency communications are a primary characteristic of applications. Since IoT applications usually have small payload sizes, reducing communication overhead is crucial to improving energy efficiency. Researchers have proposed several methods to resolve the load balancing issue of IoT networks and reduce communication overhead. Although these techniques are not effective, in terms of high communication costs, end-to-end delay, packet loss ratio, throughput, and node lifetimes negatively impact network performance. In this paper, we propose a communication overhead aware optimal cluster-based (COOC) routing algorithm for IoT networks based on a hybrid heuristic technique. Using three benchmark algorithms, we form load-balanced clusters using k-means clustering, fuzzy logic, and genetic algorithm. In the next step, compute the rank of each node in a cluster using multiple design constraints, which are optimized by using the improved COOT bird optimum search algorithm (I-COOT). After that, we choose the cluster head (CH) according to the rank condition, thereby reducing the communication overhead in IoT networks. Additionally, we design chaotic golden search optimization algorithm (CGSO) for choosing the optimal best path between IoT nodes among multiple paths to ensure optimal data transfer from CHs. To conclude, we validate our proposed COOC routing algorithm against the different simulation scenarios and compare the results with existing state-of-the-art routing algorithms.

***Keywords*—Internet-of-things; communication overhead; cluster based routing; multipath routing; cluster head**

I. INTRODUCTION

The Internet of Things (IoT) in the modern world gives academics a platform to expand the communication paradigm to new and interesting heights [1]. IoT includes computing and sensor devices that offer services at anytime, anywhere. Computers, mobile phones, laptops, household appliances, consumer electronics, sensors, and actuators are just a few examples of the homogeneous and heterogeneous systems and components that make up this system [2]. Wireless sensor networks (WSN) are one of the IoT's components. The combination of a massive number of sensor nodes is what produces the data for the IoT network. Due of their extremely low power consumption, these sensor nodes have a limited communication range. The data is generated by the wireless sensor network and is transmitted to the sink node by way of an intermediary sensor node. The sink node, which can also be referred to as a gateway node or Base Station (BS), gathers and aggregates the data before sending it to the cloud for additional processing and storage [3][4]. A separate routing protocol is used to send the data to the gateway node in an effort to use less energy. There have been a lot of studies done on data transmission schemes that balance IoT energy usage with data compression techniques that lower the energy needed for data transmission [5]. Data fusion is a complex issue, and there are still a number of issues that need to be researched. As a result, unlike the internet, the data from any terminal is crucial for IoTs. The loss of total control over a piece of equipment may result from the death of nodes from one location [6][7].

Distributed data routing and adaptable networking would therefore be more appropriate for IoT operation. Clustering is a crucial step in the process of making the IoT network more durable. These protocols address a number of concerns, including network longevity, scalability, dependability, and energy efficiency [8]. The IoT network's sensor nodes are dispersed throughout, and the clusters meet the following requirements. Each cluster's cluster head (CH) is chosen depending on a number of factors, such as queue size, link quality, and residual energy. One major limitation of IoT projects of this scale is in the name, the requirement of internet access. To overcome this problem, our design focuses on peer-to-peer communication [9]. With each unit not depending on an available network, this system can be deployed more quickly, with less overhead, and for a lower cost, meaning more of the world’s cities can be supported. The network will have a single master device requires internet connectivity through an available access point, LTE connectivity, or direct connection to a local server to store and distribute the data produced. This device will handle storing the location and index of each new device on the network so that the data it receives can be easily analyzed and passed along through the proper channels [10].

Recently, several routing algorithms [11]-[20] have been proposed to solve security issues in IoT-WSN. A security-aware routing algorithm, or security-aware probability of success (SA-PoS), addresses proactive jamming attacks that target IoT-based multi-hop WSNs [11].

Confidential cluster-based routing (SNCR) protocol [12] in WSN uses a secure network coding-enabled method that resists eavesdropping attempts and reduces energy dissipation within a clustered network. For NDN-IoT networks, lightweight authentication and secured routing (LASeR) [13] was developed. That network allows for scalability with minimum computational or cryptographic overhead. SCOTRES is a trust-based solution for secure routing in adhoc networks that uses design metrics to increase the intelligence of network components [14]. In order to improve the performance of energy efficiency with the multi-hop data security against malicious assaults, an energy-aware and secure multi-hop routing (ESMR) protocol [15] is created. For mobile IoT devices with its connection to WSN, an intrusion prevention framework is employed to ensure data security with increased network delivery ratio [16]. A game theoretic approach is used in an energy-conscious trust derivation system [17] to reduce overhead while ensuring IoT-WSN security. For multi-gateway IoT-WSN, an effective authentication and key agreement mechanism [18] is employed to increase security. IoT is used in the context of several services, including business analytics, cancer care, emergency and operational services, in a smart Saskatchewan healthcare system [19]. To reduce the total energy consumption needed by sensor nodes to meet quality of service requirements, an energy-efficient secure routing algorithm [20] is implemented (QoS). In order to maximize dependability and decrease the path failure rate in IoT-WSN, communication overhead cognizant solutions are also implemented.

In order to further improve, a hybrid heuristic technique-based Communication Overhead aware Optimal Cluster based routing algorithm (COOC) is created for IoT networks algorithm is designed for IoT networks based on hybrid heuristic technique. The main contributions of our proposed COOC routing algorithm are given as follows:

- We utilize three benchmark algorithms for optimal cluster formation, k-means clustering, fuzzy logic, and genetic algorithm.
- To compute the rank of each node in a cluster using multiple design constraints, which are optimized by using the Improved COOT bird optimum search algorithm (I-COOT).
- To select CH according to trust degree condition, thereby reducing the communication overhead in IoT networks.
- A chaotic golden search optimization algorithm (CGSO) is used for choosing the optimal best path between IoT nodes among multiple paths to ensure optimal data transfer from CHs.
- We validate our COOC routing algorithm against the different simulation scenarios and compare the results with existing routing algorithms.

The paper's remaining section is organized as follows: Section II describe the recent works related to secure aware routing protocols for IoT. In Section III, we deliberate the problem statement and network's model of suggested COOC routing algorithm. Section IV discusses the proposed methodology of COOC routing algorithm. Section V illustrates the simulation results and comparative analysis. Section VI concludes the paper.

II. Literature Review

The existing related works of routing protocols for IoT networks are discussed in this section. Table I summarizes the research gaps we gathered from the previous studies.

For WSN, an enhanced energy-efficient CH selection technique [21] is suggested, which is used to increase network throughput and lifetime while reducing energy usage. They took into account the LEACH method's cluster head selection, and they presented data fusion strategies based on the clustering of dual cluster heads. The two clusters were chosen to gather, consolidate, and send the data, and in this two CH selection approach, the cost of communication between the two clusters is reduced.

To calculate the spread of jamming assaults, experiments based on the IEEE 802.15.4 standard's MPH, AODV, and DSR protocols are utilized to generate an epidemic model [22]. The routing path in IoT networks as well as the impact of the jammer attack in terms of attack intensity and attack persistence is validated using the Susceptible-Infected-Recovered (SIR) model.

TABLE I. Summary of Research Gaps for Routing Protocols for IoT Network

Ref.	Protocol	Cluster, CH	Application	Enhancement	Research gap
[21]	IEECHS	☑	Smart city	Consumption of Energy	Suffer from excessive energy drain
[22]	SIR	☑	Industrial	Security	Lack of reliability
[23]	MQTT	☒	Smart city	Consumption of Energy	Not ensures real-time packet transmission
[24]	CMMA	☑	Healthcare	Consumption of Energy	Not suitable for high density nodes
[25]	SIoMT	☑	Healthcare	F-measure	High network latency
[26]	ATAR	☑	Healthcare	Consumption of Energy	Vulnerable to dictionary attacks

For quick and timely data communication between M2M, which improves QoS with the minimum degree of reliability standard, MQTT protocol [23] is employed. In order to provide efficient communication for IoMT-based applications, Clustering Model Medical Application (CMMA) [24] is utilized for CH selection. For edge-computing based IoT, the CMMA protocol outperforms the performance in terms of sustainability and energy efficiency. With the goal of network optimization, swarm intelligence optimization is applied in IoT. The bee colony optimization (BCO) technique, which aims to build distributed groups of nodes with common properties without any initialization knowledge pertinent to the output or utilizing complex parameters, is modified to change the key parameters in order to automatically organize the clusters [25]. IoMT-WSN uses the thermal aware routing protocol (ATAR) [26] to enhance latency and energy economy. Each node modifies its power level during transmission by observing its surrounding nodes. The received signal strength indicator value yields the value of the neighbor node. It is necessary to have a neighbor with high throughput values, which ultimately leads to energy efficiency and low heat generation.

III. Problem Statement and System Model

A. Problem Statement

A clustering technique has been proposed by Yarinezhad et al. [27] to balance the traffic strain placed on the CHs in IoT-WSNs. The 1.2-approximation approach is used in the clustering process. Data packets were sent from the CHs to their final destination using an energy-conscious routing mechanism. By properly segmenting the area, this routing technique spreads the communication load of the data packets among many nodes close to the destination. In order to prevent the creation of a hot spot close to the sink, the data from the cluster heads is transported to it along the best possible paths. The Fixed-Parameter Traceable Approximation Clustering (FPTAC) approach used grouping techniques to cut down on the number of individual sensor nodes, which also decreased the algorithm's temporal complexity. IoT devices with energy constraints use more energy since nodes are mobile, which reduces the network lifetime. Due to each node's finite energy supply, optimization of energy consumption is thought to be the main goal in the study of WSN system architecture. By using the energy more effectively and extending the lifespan of the network, clustering of nodes helps lower the energy consumption of the network in WSNs.

As the number of sensor-enabled physical devices connected to the internet has dramatically increased, it is crucial for data to be transferred from source to destination as quickly as possible. So, routing is important in the Internet of Things. However, IoT is mobile by its very nature. Mobility is a good contender for effectively addressing hand-off time concerns, data transmission delays, overhead, and low packet delivery rates. The requirements for IoT routing protocols change constantly depending on the application. IoT have risen to the forefront of medical media technologies due to their small size and capacity for wireless data transport. High energy efficiency, transmission reliability, and extended battery life of sensor devices are necessary for a dependable network. The effectiveness of healthcare delivery is increased by taking protocol layers, data routing, and energy optimization measures into account. The need for steady, dependable, and real-time transmission due to the sizeable volume of data makes it imperative to find a solution. Numerous heterogeneous devices, a high bit error rate, frequent network failures, and QoS assurance are the main problems with routing protocols. The quick and broad use of the Internet of Things (IoT) around the world has boosted the significant performance attained in terms of applications, technology, and security. Finally, we address the issues in IoT network, communication overhead, energy consumption and congestion issues for an optimal reliable solution. To overcome those problems in previous studies, COOC routing algorithm is proposed for IoT networks. The main objectives of COOC routing algorithm is describes as follows:

- Optimize clustering and multipath routing is used to formulate communication overhead aware optimal cluster-based (COOC) routing.
- By using COOC routing, we were able to increase network throughput and reduce computation and communication costs.
- Furthermore, COOC routing reduces the battery-power consumption of the network, increasing its overall lifetime.
- NS-2 simulator is used to evaluate our COOC routing.

B. Network Model of our COOC Routing Algorithm

Fig. 1 shows the typical structure of IoT network with our proposed COOC routing algorithm using optimal clustering and efficient multipath routing. The routing protocol is then used to send the data gathered from IoT sensors to the base station (BS). Almost all of the IoT network's gadgets run on limited, non-rechargeable energy sources like batteries. IoT applications typically operate in crowded, harsh locations, making it difficult to add or swap out the sensors' power sources.

The k-means clustering, fuzzy logic and genetic algorithm is used for the cluster formation using the basic information of IoT nodes location and distance between nodes to BS. Then, the rank of the nodes is compute by the individual position of nodes with respect to other nodes using multiple design constraints. The rank decreases in the up direction and increases in the down direction. Next, we develop an I-COOT algorithm for design constraints optimization which used to select CH among multiple nodes. Finally, we find the optimal best path between IoT nodes among multiple paths by using CGSO algorithm.

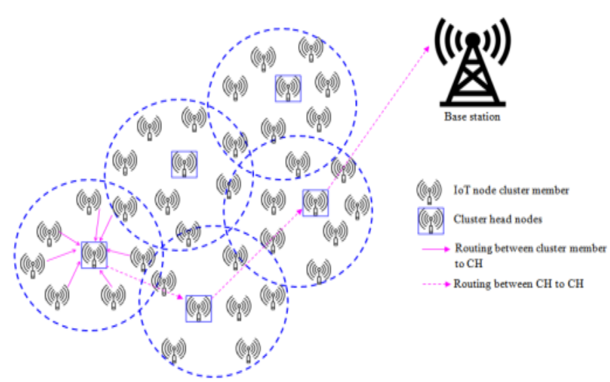

Fig. 1. Typical structure of IoT network with our COOC routing algorithm

IV. Proposed Methodology

We describe working function of our COOC routing algorithm which consists following set of process are clustering, design constraints for rank computation, optimization of design constraints and optimal path selection.

A. Clustering using Benchmark Algorithms

The creation of energy-efficient solutions becomes crucial since IoT nodes are energy-constrained and run on a small internal battery. In order to prepare for impending demand, energy conscious IoT networks must simultaneously forecast their energy use. A collection of sensor nodes that can sense, calculate, and transmit make up the network. Energy conservation in IoT becomes a major concern to increase network lifetime. Since clustering is regarded as an efficient and suitable way for transmitting the data without any issues, multiple efforts have been made to improve the routing protocols in the network to date. In this study, we used the k-means clustering, fuzzy logic, and genetic algorithms as three benchmark load-balanced clustering methods. It is necessary for this particular application to illustrate the chromosomal distribution to utilize the k-means clustering technique to partition unsatisfactory groupings. Our supposition is that the population is divided into clusters.

$$cq^{i,*} = \frac{1}{m_i} \sum_{cp_N \in c_i} cq_N, \quad i = 1,2,...,k \tag{1}$$

Where the number of cluster-related elements is m_i. The f symbol represents the altered feature space with a greater or even infinite dimension, and Y stands for the data space. The following objective function is minimized by KFCM.

$$I_{kfcm}(u,v) = \sum_{K=1}^{C} \sum_{j=1}^{N} \mu_{Kj}^{M} \mid \Phi(y_K) - \Phi(v_j) \mid^2 \tag{2}$$

The difference between the sizes of the largest cluster and the smallest cluster normalizes the size of the cluster and the size of the cluster.

$$\hat{H}_A = \frac{H_A - H_{\min}}{H_{\max} - H_{\min}} \tag{3}$$

$$\hat{H}_w = \frac{H_w - H_{\min}}{H_{\max} - H_{\min}} \tag{4}$$

Where H_{max} and H_{min} represent the normalized values, which range from zero to one, respectively. By explicitly detecting its presence in the algorithm, this circumstance can be avoided. Fuzzy logic provides the ability to make defensible conclusions in a world of uncertainty, imprecision, and missing data. It is therefore the optimal strategy to use in scenarios with real, continuous-valued elements because it uses data acquired in surroundings that include such qualities. The aforementioned context is appropriate for the information gathered about computer network traffic, which supports its use in anomaly detection. The membership degree of an element is obtained using a fuzzy membership function, which accepts a variety of arguments. The Gaussian membership function is an illustration of such a function.

$$\xi = E^{\frac{-(y-\hat{y})^2}{2\theta^2}} \tag{5}$$

Where θ is a parameter that defines the standard deviation, $\hat{y}$ is the center, y is the value to calculate its membership. In this paper, the function is derived from the Gaussian membership function as

$$\xi_k = 1 - E^{\frac{-(y_k-\hat{y}_k)^2}{2\Phi_k^2}} \tag{6}$$

Using fuzzy logic, it is possible to determine whether an abnormality is happening right now. A fuzzy method is used to reduce this issue without impairing the system's capacity to detect anomalies.

B. Genetic Algorithm

Genetic Algorithm (GA) is a worldwide, parallel, stochastic search approach that exhibits significant robustness in problem domains where formal, strict, classical analysis is not feasible. The roulette wheel and a competition are two of these selection techniques. The odds of winning in roulette are determined by the fitness values of the chromosomes, which dictate

$$q_j = \frac{F_j}{\sum_{i=1}^{N} F_i} \tag{7}$$

Based on the results of the trials, it can be concluded that fuzzy logic and k-means clustering are not as exact as the best option for categorization. The precision and recall are both greatly enhanced by the GA algorithm. The definitions of recall and precision

$$precision = \frac{tp}{tp + fp} \tag{8}$$

$$recall = \frac{tp}{tp + fn} \tag{9}$$

The superiority of the GA algorithm becomes more apparent and we can get the meaningful findings more quickly when it is applied to a few additional data sets.

$$precision = \frac{|\,relevant \cap retrieved\,|}{|\,retrieved\,|} \tag{10}$$

$$recall = \frac{|\,relevant \cap retrieved\,|}{|\,testing\,|} \tag{11}$$

When the user is aware of the nonlinearities in the problem, the GA method can perform pretty well. However, the GAKFCM is more accurate and can overcome problems for the GA algorithm. And take a different look at the GA algorithm. The workings of cluster construction employing k-means clustering, fuzzy logic, and genetic algorithm are described in algorithm 4.1.

Algorithm 4.1 Benchmark algorithm for cluster formation

Input: location, distance between nodes and BS
Output: Formation of cluster

1. Initiate the random population
2. Minimize the KFCM by objective function
$$I_{kfcm}(u,v)=\sum_{K=1}^{C}\sum_{j=1}^{N}\mu_{Kj}^{M}\,|\,\Phi(y_K)-\Phi(v_j)\,|^2$$
3. The condition is avoided by specifically checking for it within the algorithm
4. j=0 and i=1
5. Define Gaussian membership function $\xi = E^{\frac{-(y-\hat{y})^2}{2\theta^2}}$
6. Find fitness values of the chromo $q_j=\frac{F_j}{\sum_{i=1}^{N}F_i}$
7. Get the useful results faster
$$precision=\frac{|\,relevant \cap retrieved\,|}{|\,retrieved\,|},$$
$$recall=\frac{|\,relevant \cap retrieved\,|}{|\,testing\,|}$$
8. Update the final values
9. End

C. Design Constraints Optimization

The practice of minimizing the amount of input constraints when creating a predictive model is known as design constraint optimization. In some circumstances, less input constraints might increase model performance while also lowering the computing cost of simulations. Using the Improved COOT bird optimal search method (I-COOT), this work chooses the cluster head (CH) based on the rank of each node. Coots are little waterfowl that belong to the Rallidae family of rails. They belong to the Fulica genus, which is named after the Latin word for "coot." This bird's actions on the water's surface can be used as an optimization technique. Coots appear to be well within what is, for surf scoters, a zone of repulsion as they travel at an angle to their direction of motion. There is no assurance that a solution will be found in one run when using population-based optimization approaches to discover the ideal number of optimization issues. However, if there are enough random solutions and optimization processes, the likelihood of discovering the overall best improves. Using the formula, the population is produced at random in the small area.

$$cootpos(j)=rand(1,D).*(ua-la)+la \tag{12}$$

where *cootpos* (*j*) is the coot position, d the number of variables or problem dimensions, *la* is the lower bound of the search space and '*ua*' is the upper bound of the search space. Each variable may have a different lower bound and upper bound problem.

where *D* is the number of variables or problem dimensions, and lb and 'ua' represent the lower and upper bounds of the search space, respectively. Coot position is cootpos(j), there could be many lower bound and upper bound issues for each variable.

$$la=[la_1,la_2,...la_D],ua=[ua_1,ua_2,..,ua_D] \tag{13}$$

In order to carry out this movement, we take into account a random place within the search space and move the coot in that direction.

$$P=rand(1,D).*(ua-la)+la \tag{14}$$

The search space is explored by this coot movement in many areas. This movement will let the algorithm escape the local optimal if it becomes stuck in the local optimal. The new position of the coot is calculated as follows:

$$cootpos(j)=cootpos(j)+B\times r2\times(P-cootpos(j)) \tag{15}$$

we compute B using the random movement of the coot in various directions, where "r2" is a random value in the range [0, 1].

$$B=1-l\times\left(\frac{1}{Iter}\right) \tag{16}$$

Where, Iter is the maximum iteration and l is the current iteration. The typical alignments of two coots are used for implementing chain movement. We may also move the coot toward the other coot by roughly halving the distance between them after first calculating the distance vector between them. We employed the first technique, and a formula was applied to determine the coot's new position.

$$cootpos(j)=0.5\times(cootpos(j-1)+cootpos(j)) \tag{17}$$

The group is often led by a few coots in the front, and the remainder of the coots must move closer and alter their posture in accordance with the group's leaders. One possible query is if each coot will change its position according to which leader. The coots can adjust their position based on the average position of the leaders, which can be taken into consideration. Premature convergence results from taking the average position into account. We employ a system in accordance with the leader-selection process to carry out this movement.

$$k=1+(j \bmod nl) \tag{18}$$

where K is the leader's index number, NL is the total number of leaders, and I is the index number of the current coot. Based on the leader's k, the coot (j) must update its location to determine the coot's subsequent position based on the chosen leader.

$$\begin{aligned} cootpos(j) &= leaderpos(K)+2\times r1\times \\ &\cos(2r\pi)\times(leaderpos(K)-cootpos(j)) \end{aligned} \tag{19}$$

Where, the coot's current position is cootpos(j), leaderpos(K) has been chosen as the leader position. R1 is a random number between 0 and 1, π is the same as pi, or 3.14, and 'r' is a random number between 0 and 1. Around this current ideal location, this formula seeks out better positions. Leaders may need to shift away from the present best position in order to find better positions. This formula offers a useful method for moving away from and toward the ideal place.

$$leaderpos(j)=\begin{cases} A\times r3\times\cos(2r\pi)\times & r4<0.5 \\ (gbest-leaderpos(j))+gbest & \\ A\times r3\times\cos(2r\pi)\times & r4>=0.5 \\ (gbest-leaderpos(j))-gbest & \end{cases} \quad (20)$$

Where the best position ever discovered is $gbest$, R3 and R4 are random numbers in the range [0, 1], is equal to 3.14 pi, R is a random number in the range [1, 1], and B is determined using the formula.

$$A=2-l\times\left(\frac{1}{Iter}\right) \quad (21)$$

Where 'Iter' for the most iterations and 'l' stands for the current iteration. 2 x r3 greater random movements are made to prevent the algorithm from becoming stuck at the local optimum. This indicates that while we are in the exploitation phase, we are simultaneously undertaking exploration. Cos(2rπ) looks for a better position near the best search agent by searching in various radii about it. Algorithm 4.2 describes the working process of CH selection using I-COOT.

Algorithm 4.2 CH selection using I-COOT

Input : Energy efficiency, Link quality, Path loss, Distance and Delay
Output : design constraints optimization and CH

1. Initialize the random population
2. The population is randomly generated $cootpos(j)=rand(1,D).*(ua-la)+la$
3. J=0, and i=1
4. Define coot towards this random position $P=rand(1,D).*(ua-la)+la$
5. Selection of leader $k=1+(j \bmod nl)$
6. Selection of coot based on selected leader $cootpos(j)=leaderpos(K)+2\times r1\times \cos(2r\pi)\times(leaderpos(K)-cootpos(j))$
7. Compute B $A=2-l\times\left(\frac{1}{Iter}\right)$
8. Update the final values
9. End

D. Optimal Path Selection

A well-known trade-off in the architecture of IoT is minimizing power consumption at the expense of the performance of the network. Traditional sensor network platforms were created with a focus on low power consumption at the expense of communication throughput. To collect auditory and visual data, which has a high need for transmission throughput, new apps are being used. According to the investigation, conventional ways fail to deliver improved security and quality of service (QoS), as well as to balance the temperature and load of WSN-IoT devices. They also fail to extend the network lifetime and reduce energy depletion. In order to ensure optimal data transfer from CHs, we have created the chaotic golden search optimization algorithm (CGSO), which selects the best path among numerous paths connecting IoT nodes. Kiefer introduced the golden section search (GSO) in 1953. (Kiefer, 1953). When an object function is uni-modal, this approach can be used. The approach performs admirably when solving object functions that are either impossible to discriminate or difficult to differentiate. 2-D GSS for object tracking is a newly introduced variation of the golden section search. It also appears in straightforward maps like the logistic map. Typically, a one-dimensional chaotic map looks like this:

$$y(N+1)=F(\mu_1,\mu_2,...\mu_M,y(N)),\ N=0,1,2,3... \quad (22)$$

A chaotic map is fused with GSO algorithm to optimize path selection constraints. The chosen control parameters $\mu_j,\ j=1,....,M$ are quite modest, yet even a little shift in the chaotic variable's starting value, x, will have a significant impact on subsequent values of the chaotic variable, y. To define one-dimensional chaotic maps as follows:

$$y(N+1)=by(N)(1-y(N))y(0)\in(0,1), \\ y(0)\notin\{0,0.25,0.5,0.75,1\} \quad (23)$$

$$y(N+1)=\cos(K\cos^{-1}(y(N)))\,y\in(-1,1) \quad (24)$$

Chaos with b=4 is generated by the logistic map. le=0.6932 is the Lyapunov exponent of the Chebyshev map with k=2. The scout bee uses abandoned food sources as new ones.

$$Y_{ji}=y_{\min,i}+f(y_{\max,i}-y_{\min,i}) \quad (25)$$

Depending on the scale factor f_1 & f_2 the creation of new food sources is viewed as a black box procedure. The primary concept is that, in order to ensure a high-quality solution that will play a crucial part in succeeding generations, the updating of the scale factor and, consequently, the creation of the food sources, are, with a given probability, controlled. The procedure introduces the Chaotic Golden Search Optimization Technique (CGSO), a traditional local search algorithm for non-differentiable fitness functions. In order to produce high-quality food sources, the scale factor golden section search uses the golden section search to scale factor. This procedure produces two intermediate points in the range [a = 1, b = -1]:

$$f_1=a-\frac{a-b}{\delta}, \quad (26)$$

$$f_2=b+\frac{a-b}{\delta} \quad (27)$$

The scaling factor's upper and lower bound values are calculated as follows. In the GSS algorithm, the first two points $y_1,y_2\in[l,u]$ are calculated as follows.

$$C=\frac{-1+\sqrt{5}}{2} \quad (28)$$

$$y_1=Cl+(1-C)u \quad (29)$$

$$y_2=(1-C)l+Cu \quad (30)$$

The search is carried out until the stop criteria are met. As a result, just one of these portions is chosen for the subsequent iteration. As a result, it is essential that the two parts are of equal width. In certain circumstances, the bigger portion is therefore taken more repeatedly, and the convergence speed is slowed.

$$Q + P = P + R \tag{31}$$

$$\frac{Q}{P} = \frac{Q}{P} = \frac{(P+Q)}{Q} = \frac{1}{C} = \varphi \tag{32}$$

From above equations, we follow that φ=161803398… and C=1.61803398... Thus, if n is the number of iterations then, φ^n is the convergence rate of CGSO algorithm. The search interval shrunk to less than 1.0% of the original interval for n=15. The algorithm 4.3 describes the working function of optimal path selection using CGSO. We follow from the aforementioned equations that φ=161803398… and C=1.61803398....As a result, if n is the number of iterations, then the CGSO algorithm's rate of convergence is φ^n. Less than 1% of the initial search interval for n=15 remained after the search interval shrank. The operation of the CGSO-based optimal path selection algorithm is described in algorithm 4.3.

Algorithm 4.3 Chaotic Golden Search Optimization algorithm (CGSO)

Input: CH, congestion rate, aggregation delay
Output: optimal paths

1 Initialize the random population
2 Define one-dimensional chaotic map $y(N+1) = F(\mu_1, \mu_2, ...\mu_M, y(N)),\ N = 0,1,2,3...$
3 The interval values are a = 1, b = -1
4 Define the initial food sources $Y_{ji} = y_{\min,i} + f(y_{\max,i} - y_{\min,i})$
5 Calculate two points in GSS algorithm: $y_1, y_2 \in [l,u] \qquad c = \frac{-1+\sqrt{5}}{2}$
6 The convergence speed is reduced $Q + P = P + R$
7 Update the final values
8 End

V. Results and Discussion

We describe simulation results and comparative analysis of suggested COOC and existing routing algorithms with the different simulation scenarios. COOC routing algorithm is simulate and analyze using Network simulator (NS-2). The simulation results are done for performance evaluation of the COOC routing algorithm against existing state-of-art algorithms' performance, based on clustering FPT (CFPT) [28], routing FPT-approximation (RFPT) [29], energy-efficient and EB-CRP (energy-balanced cluster-based routing protocol) [30] and fixed-parameter tractable approximation clustering (FPTAC) [27].

A. Simulation Setup

The sink is situated in the centre of a terrain measuring 1000 m by 1000 m, where the nodes are situated. A grid of 100 nodes is created, and the remaining 900 are distributed at random. The sensor node's transmission and reception power consumption are 24.92 and 19.72 mJ per byte, respectively, according to the simulation requirements. IoT sensors come in a variety of numbers, ranging from 200 to 1000. Each sensor node and each gateway are assumed to have a starting energy of 2J. Each node communicates with the others via the CSMA/CA MAC layer protocol. To send the data packets to the gateways, the sensor nodes employ the TDMA that the BSs choose. The IoT sensor nodes' transmission range is 100 metres, and the data packet size is 4000 bits. The simulations are run 30 times, and the accompanying graph shows the typical outcome of the runs. The precise configuration of our simulation is described in Table II.

TABLE II. SIMULATION SETUP

Simulation Area	1000m×1000m
Number of IoT sensors	200-1000
Data size	4000 bits
Control packet size	200 bits
Senor sensing range	80 m
Initial energy of sensor nodes	2J
MAC protocol	CSMA/CA
Bandwidth	250 Kb/s
Payload size	30 bytes
Transmission range	100 m
Avg. energy consumption of transmitting node	24.92mJ per byte
Avg. energy consumption of receiving node	19.72mJ per 1 byte
Simulation time	30 times

B. Comparative Analysis on Routing Algorithms

1) Impact of node density: Using 200, 400, 600, 800, and 1000 nodes as well as a fixed network size of 1000m×1000m, we analyze the performance of our proposed and existing routing algorithms. A simulation analysis of existing and proposed routing algorithms, energy consumption, throughput, network lifetime, routing overhead, reception ratio, and average link lifetime is presented. Energy consumption of our proposed and existing routing algorithms is compared in Table III and Fig. 2. By utilizing node density as an indicator of energy consumption performance, the proposed COOC routing algorithm ensures better solution. The energy consumption of our proposed COOC routing algorithm is 21.685%, 17.196%, 12.161% and 6.474% efficient than the existing CFPT [28], RFPT [29], EB-CRP [30], and FPTAC [27] routing algorithms respectively. The throughput of our proposed and existing routing algorithms is compared in Table IV and Fig. 3. It appears that the COOC scheme has the most throughput compared to the other routing schemes, with a value of 80300Mbps for 200 nodes and 66000Mbps for 1000 nodes. The CFPT scheme has the lowest throughput, with a value of 25000Mbps for 200 nodes and 8000Mbps for 1000 nodes. Overall, the results suggest that the COOC scheme provides the best performance in terms of throughput, followed by FPTAC, EB-CRP, RFPT and CFPT.

TABLE III. ENERGY COMPARISON (J) WITH NODE DENSITY

Routing scheme	Number of nodes				
	200	**400**	**600**	**800**	**1000**
CFPT	173.48	274.4	558.76	958.34	1229.3
RFPT	137.56	236.41	522.22	919.08	1195.3
EB-CRP	97.28	197.20	485.73	887.08	1139.3
FPTAC	51.79	159.52	449.71	850.52	1106.5
COOC	12.53	121..51	412.5	812.54	1070.5

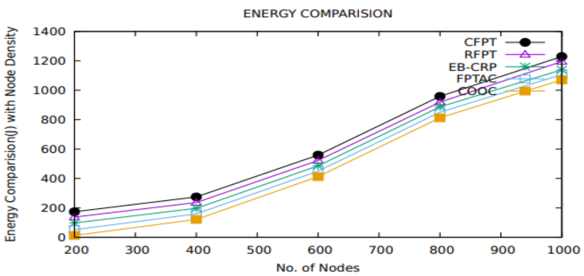

Fig. 2. Energy consumption with node density

TABLE IV. THROUGHPUT WITH NODE DENSITY

Routing scheme	Number of nodes				
	200	**400**	**600**	**800**	**1000**
CFPT	25000	20000	18080	11000	8000
RFPT	40000	35008	30000	27500	22391
EB-CRP	54000	50000	45000	40100	37000
FPTAC	69000	65000	60000	57000	51000
COOC	80300	79900	73123	68500	66000

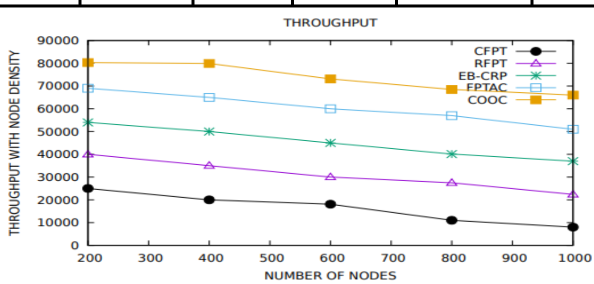

Fig. 3. Throughput with node density

The network lifetime of our proposed and existing routing algorithms is compared in Table V and Fig. 4. By utilizing node density as an indicator of network lifetime performance, the proposed COOC routing algorithm ensures better solution. The network lifetime of our proposed COOC routing algorithm is 26.147%, 19.09%, 13.073% and 6.537% efficient than the existing CFPT [28], RFPT [29], EB-CRP [30], and FPTAC [27] routing algorithms respectively. The routing overhead of our proposed and existing routing algorithms is compared in Table VI and Fig. 5. By utilizing node density as an indicator of routing overhead performance, the proposed COOC routing algorithm ensures better solution.

TABLE V. NETWORK LIFETIME WITH NODE DENSITY

Routing scheme	Number of nodes				
	200	**400**	**600**	**800**	**1000**
CFPT	28.02	27.05	26.16	24.27	23.89
RFPT	33.07	30.12	28.46	27.49	26.04
EB-CRP	34.02	33.25	30.30	28.16	27.01
FPTAC	35.05	34.34	33.32	32.43	30.35
COOC	38.70	36.14	35.05	34.41	33.17

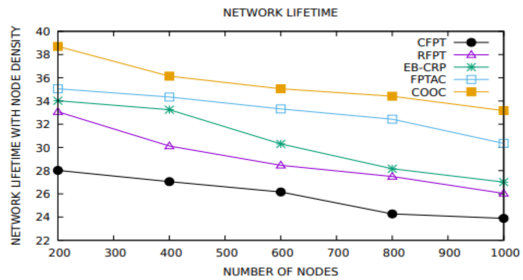

Fig. 4. Network lifetime with node density

TABLE VI. ROUTING OVERHEAD WITH NODE DENSITY

Routing scheme	Number of nodes				
	200	**400**	**600**	**800**	**1000**
CFPT	65.01	69.15	73.21	77.21	80.12
RFPT	63.24	67.61	70.16	74.92	78.73
EB-CRP	60.12	63.97	67.24	72.84	75.12
FPTAC	57.86	62.62	65.13	68.71	73.34
COOC	55.12	59.21	63.89	67.29	70.10

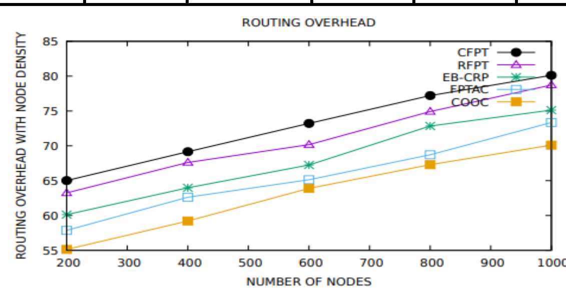

Fig. 5. Routing overhead with node density

TABLE VII. RECEPTION RATIO WITH NODE DENSITY

Routing scheme	Number of nodes				
	200	**400**	**600**	**800**	**1000**
CFPT	64.02	63.12	61.42	60.12	59.29
RFPT	72.12	70.44	68.12	65.69	64.90
EB-CRP	82.24	80.15	78.60	76.75	75.50
FPTAC	90.21	89.21	88.12	87.30	86.21
COOC	98.13	97.41	96.3	95.20	94.12

The routing overhead of COOC routing algorithm is 13.465%, 10.451%, 7.219%, and 3.744% efficient than the existing CFPT [28], RFPT [29], EB-CRP [30], and FPTAC [27] routing algorithms respectively.

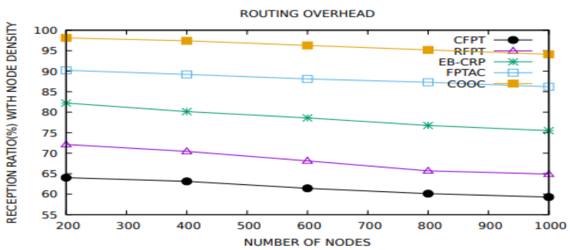

Fig. 6. Reception ratio with node density

TABLE VIII. AVERAGE LINK LIFETIME WITH NODE DENSITY

Routing scheme	Number of nodes				
	200	400	600	800	1000
CFPT	12.51	11.25	10.14	8.75	7.51
RFPT	15.21	13.75	12.52	11.25	10.54
EB-CRP	17.25	16.16	14.57	13.48	11.75
FPTAC	19.53	18.25	17.54	15.75	14.51
COOC	22.01	20.75	19.51	18.25	17.48

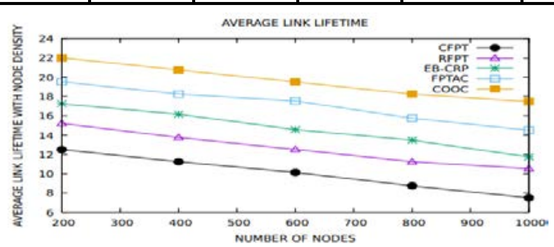

Fig. 7. Average link lifetime with node density

The reception ratio of proposed and existing routing algorithms is compared in Table VII and Fig. 6. By utilizing node density as an indicator of reception ratio performance, the proposed COOC routing algorithm ensures better solution. The reception ratio of COOC routing algorithm is 32.119%, 21.94%, 16.059% and 8.03% efficient than the existing CFPT [28], RFPT [29], EB-CRP [30], and FPTAC [27] routing algorithms respectively. The average link lifetime of proposed and existing routing algorithms is compared in Table VIII and Fig. 7. It appears that the COOC scheme has the most average link lifetime compared to the other routing schemes, with a value of 22.01 seconds for 200 nodes and 17.48 seconds for 1000 nodes. The CFPT scheme has the lowest average link lifetime, with a value of 12.51 seconds for 200 nodes and 7.51 seconds for 1000 nodes. Overall, the results suggest that the COOC scheme provides the best performance in terms of average link lifetime, followed by FPTAC, EB-CRP, RFPT and CFPT.

VI. CONCLUSION

In this work, based on hybrid heuristic, we propose communication overhead aware optimal cluster-based (COOC) routing algorithm for IoT networks. With the use of k-means clustering, fuzzy logic, and genetic algorithms, we form load-balanced clusters. I-COOT is used to optimize multiple design constraints to compute the rank of each node in a cluster. In IoT networks, we reduce communication overhead by select CH according to the rank condition. A chaotic golden search optimization algorithm (CGSO) is designed for optimizing data transfer from the CHs by identifying the best path among multiple paths among IoT nodes. In conclusion, we validate our proposed COOC routing algorithm against different simulation scenarios. From the simulation results, we observed that the effectiveness of our proposed COOC routing algorithm perform very effective manner in terms of consumption, throughput, network lifetime, routing overhead, reception ratio, and average link lifetime compared to existing routing algorithms.

REFERENCES

[1] Li, X., Qi, H. and Wu, J., 2022. Node social nature detection OSN routing scheme based on IoT system. IEEE Internet of Things Journal.

[2] Ghosh, S., Dagiuklas, T., Iqbal, M. and Wang, X., 2022. A cognitive routing framework for reliable communication in iot for industry 5.0. IEEE Transactions on Industrial Informatics, 18(8), pp.5446-5457.

[3] Deng, S., Zhao, H., Xiang, Z., Zhang, C., Jiang, R., Li, Y., Yin, J., Dustdar, S. and Zomaya, A.Y., 2021. Dependent Function Embedding for Distributed Serverless Edge Computing. IEEE Transactions on Parallel and Distributed Systems, 33(10), pp.2346-2357.

[4] Yu, C., Shen, S., Yang, H., Zhang, K. and Zhao, H., 2021. Leveraging Energy, Latency and Robustness for Routing Path Selection in Internet of Battlefield Things. IEEE Internet of Things Journal.

[5] Ghahramani, M., Zhou, M., Molter, A. and Pilla, F., 2021. IoT-based Route Recommendation for an Intelligent Waste Management System. IEEE Internet of Things Journal.

[6] Wang, X., Hu, J., Lin, H., Garg, S., Kaddoum, G., Jalilpiran, M. and Hossain, M.S., 2022. QoS and privacy-aware routing for 5G enabled industrial internet of things: A federated reinforcement learning approach. IEEE Transactions on Industrial Informatics.

[7] Zhang, Y., Ren, Q., Song, K., Liu, Y., Zhang, T. and Qian, Y., 2021. An Energy Efficient Multi-Level Secure Routing Protocol in IoT Networks. IEEE Internet of Things Journal.

[8] Ma, N., Zhang, H., Hu, H. and Qin, Y., 2021. ESCVAD: An Energy-Saving Routing Protocol Based on Voronoi Adaptive Clustering for Wireless Sensor Networks. IEEE Internet of Things Journal, 9(11), pp.9071-9085.

[9] He, Y., Han, G., Xu, M. and Martínez-García, M., 2021. A Pseudo-Packet Scheduling Algorithm for Protecting Source Location Privacy in the Internet of Things. IEEE Internet of Things Journal.

[10] Liang, J., Liu, W., Xiong, N.N., Liu, A. and Zhang, S., 2021. An intelligent and trust UAV-assisted code dissemination 5G system for industrial Internet-of-Things. IEEE Transactions on Industrial Informatics, 18(4), pp.2877-2889.

[11] H. Bany Salameh, R. Derbas, M. Aloqaily and A. Boukerche, "Secure Routing in Multi-hop IoT-based Cognitive Radio Networks under Jamming Attacks", Proceedings of the 22nd International ACM Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems - MSWIM '19, 2019. Available: 10.1145/3345768.3355944.

[12] H. Rhim, R. Abassi, K. Tamine, D. Sauveron and S. Guemara, "A secure network coding-enabled approach for a confidential cluster-based routing in wireless sensor networks", Proceedings of the 35th Annual ACM Symposium on Applied Computing, 2020. Available: 10.1145/3341105.3373969.

[13] T. Mick, R. Tourani and S. Misra, "LASeR: Lightweight Authentication and Secured Routing for NDN IoT in Smart Cities", IEEE Internet of Things Journal, vol. 5, no. 2, pp. 755-764, 2018. Available: 10.1109/jiot.2017.2725238.

[14] G. Hatzivasilis, I. Papaefstathiou and C. Manifavas, "SCOTRES: Secure Routing for IoT and CPS", IEEE Internet of Things Journal, vol. 4, no. 6, pp. 2129-2141, 2017. Available: 10.1109/jiot.2017.2752801.

[15] K. Haseeb, N. Islam, A. Almogren, I. Ud Din, H. Almajed and N. Guizani, "Secret Sharing-Based Energy-Aware and Multi-Hop Routing Protocol for IoT Based WSNs", IEEE Access, vol. 7, pp. 79980-79988, 2019. Available:10.1109/access.2019.2922971.

[16] K. Haseeb, N. Islam, A. Almogren, and I. Ud Din, “Intrusion Prevention Framework for secure routing in WSN-based mobile internet of things,” IEEE Access, vol. 7, pp. 185496–185505, 2019.

[17] J. Duan, D. Gao, D. Yang, C. H. Foh, and H.-H. Chen, “An energy-aware trust derivation scheme with game theoretic approach in wireless sensor networks for IoT applications,” IEEE Internet Things J., vol. 1, no. 1, pp. 58–69, 2014.

[18] F. Wu et al., “An efficient authentication and key agreement scheme for multi-gateway wireless sensor networks in IoT deployment,” J. Netw. Comput. Appl., vol. 89, pp. 72–85, 2017.

[19] A. Onasanya, S. Lakkis, and M. Elshakankiri, “Implementing IoT/WSN based smart Saskatchewan Healthcare System,” Wirel. netw., vol. 25, no. 7, pp. 3999–4020, 2019.

[20] J. K. Jain, “Secure and energy-efficient route adjustment model for internet of things,” Wirel. Pers. Commun., vol. 108, no. 1, pp. 633–657, 2019.

[21] S. A. Jesudurai and A. Senthilkumar, “An improved energy efficient cluster head selection protocol using the double cluster heads and data fusion methods for IoT applications,” Cogn. Syst. Res., vol. 57, pp. 101–106, 2019.

[22] M. López, A. Peinado, and A. Ortiz, “An extensive validation of a SIR epidemic model to study the propagation of jamming attacks against IoT wireless networks,” Comput. netw., vol. 165, no. 106945, p. 106945, 2019.

[23] V. Kumar, G. Sakya, and C. Shankar, “WSN and IoT based smart city model using the MQTT protocol,” J. Discrete Math. Sci. Cryptogr., vol. 22, no. 8, pp. 1423–1434, 2019.

[24] Han, T., Zhang, L., Pirbhulal, S., Wu, W. and de Albuquerque, V.H.C., 2019. A novel cluster head selection technique for edge-computing based IoMT systems. Computer Networks, 158, pp.114-122.

[25] El-shafeiy, E., Sallam, K.M., Chakrabortty, R.K. and Abohany, A.A., 2021. A clustering based Swarm Intelligence optimization technique for the Internet of Medical Things. Expert Systems with Applications, 173, p.114648.

[26] Ahmed, G., Mehmood, D., Shahzad, K. and Malick, R.A.S., 2021. An efficient routing protocol for internet of medical things focusing hot spot node problem. International Journal of Distributed Sensor Networks, 17(2), p.1550147721991706.

[27] Yarinezhad, R. and Sabaei, M., 2021. An optimal cluster-based routing algorithm for lifetime maximization of Internet of Things. Journal of Parallel and Distributed Computing, 156, pp.7-24.

[28] R. Yarinezhad, S.N. Hashemi, A routing algorithm for wireless sensor networks based on clustering and an fpt-approximation algorithm, J. Syst. Softw. 155 (2019) 145–161.

[29] R. Yarinezhad, S.N. Hashemi, Solving the load balanced clustering and routing problems in WSNs with an fpt-approximation algorithm and a grid structure, Pervasive Mob. Comput. 58 (2019) 101033.

[30] R. Yarinezhad, S.N. Hashemi, Increasing the lifetime of sensor networks by a data dissemination model based on a new approximation algorithm, Ad Hoc Netw. 100 (2020) 102084.