applied sciences

MDPI

Article

Optimized Routing by Combining Grey Wolf and Dragonfly Optimization for Energy Efficiency in Wireless Sensor Networks

Hiba Apdalani Younus [1] and Cemal Koçak [2,*]

[1] Faculty of Technology, Institute of Science and Computer Engineering, Gazi University, Ankara 06560, Turkey
[2] Faculty of Technology Computer Engineering, Gazi University, Ankara 06560, Turkey
* Correspondence: ccckocak@gazi.edu.tr; Tel.: +90-312-202-85-69

Abstract: The rapid development of technology has resulted in numerous sensors and devices for performing measurements in an environment. Depending on the scale and application, the coverage and size of a wireless sensor network (WSN) is decided. During the implementation, the energy consumption and life of the nodes in the WSN are affected by the continuous usage. Hence, in this study, we aimed to improve the lifespan of the WSN and reduce energy consumption by the nodes during the data transfer using a hybrid approach. The hybrid approach combines Grey Wolf Optimization (GWO) and Dragonfly Optimization (DFO) for exploring a global solution and optimizing the local solution to find the optimum route for the data transfer between the target node and the control center. The results show that the proposed approach has effective energy consumption corresponding to the load applied. Our proposed system scored high in the average residual energy by the number of rounds compared to other methods such as k-means, LEACH-C, CHIRON, and Optimal-CBR. The first dead node was found after 500 rounds, showing that the proposed model has nodes with better reliability. It also showed a comparative analysis of the transmission rate of a packet concerning mobility speed among various methods. The proposed method has the highest ratio at all mobility speeds, i.e., 99.3, 99.1, 99, 98.8, and 98.6, and our proposed system has the lowest computational time of all the evaluated methods, 6 s.

Keywords: energy efficient; wireless sensor networks; grey wolf optimization; dragonfly optimization (DFO)

check for updates

Citation: Younus, H.A.; Koçak, C. Optimized Routing by Combining Grey Wolf and Dragonfly Optimization for Energy Efficiency in Wireless Sensor Networks. *Appl. Sci.* **2022**, *12*, 10948. https://doi.org/10.3390/app122110948

Academic Editor: Christos Bouras

Received: 26 September 2022
Accepted: 26 October 2022
Published: 28 October 2022

1. Introduction

A wireless sensor network (WSN) is one of the widely adopted network systems in industry health care and in many intrusion detection areas. It comprises a network of devices defined as nodes. The nodes have the ability to sense the environment from the area under monitoring, gather information, and communicate the collected data by means of wireless links. The final destination node is called the destination node or sink node. The data from the initial node are forwarded through multi-hop routing to the destination node and the collected data are utilized by other networks through the gateway. The structure of the nodes can be moving or stationary, and they must be aware of their identity location. Since a WSN does not require cables or wires, it is flexible and may add new nodes or devices to the network at any time.

The recent developments in WSN extended its application to sophisticated systems. The challenges in WSNs persist even with the modern developments [1]. A WSN is interconnected with various nodes and forms a network within a range. The expansion of coverage is determined by the type of application as well as the number of nodes and devices. The energy used by these nodes depends on the establishment and the remote areas that require an improved, energy efficient system [2]. The data collected by the sensor nodes must be transmitted consistently over a long period of time. The modern approaches and recent studies have shown that the data transfer happens through WSN and makes them consume more energy and reduces the network lifespan. The power resource constraint is critical, considered a lifetime factor for WSNs which cannot be replaced if they face certain technical issues in their application. For optimal routing mechanisms, the pattern of clustering and the decision of head selection for clusters are critical. The selection is based on the nodes having maximum energy and minimum traversing distance between the neighboring nodes with medium speed quality, followed by the patterns of clustering for the determination of routing paths. Algorithms in artificial intelligence should be employed for determining the shortest. It is the responsibility of the protocol to administer and update the routing table for effective routing and to minimize the energy consumption. Protocols used for routing mechanisms should focus mainly on the energy needed for the network operation.

Many researchers have developed new techniques to overcome the energy efficiency problem in order to improve the WSN lifespan. However, these techniques are not sufficient to balance everything. Researchers have adopted hybrid approaches to take advantage of different techniques to achieve a balance in WSN. This is because an overperforming WSN results in premature node failures or dead nodes, indicating that nodes die with burden.

If the nodes are used for a long time, they face premature failure [3]. To overcome this issue, recent studies have attempted to optimize routing to reduce the energy consumed and improve the network lifespan.

The optimization of path in WSNs is important for solving the issue of redundant data forwarded into the network and for reducing energy consumption. Cluster formation helps in the process of minimizing the energy level, where data aggregation is an important criterion that needs to be addressed both at the level of the cluster head and at the base station. The process of data aggregation is performed at each router while forwarding the data packets. The packet transmission follows the working principle of routing protocols that involve many algorithms for the determination of optimal network data transfer and the pathway for communication between the sensor nodes.

Various optimization techniques have been used [4] by researchers and developers in WSNs, such as Particle Swarm Optimization (PSO) [5,6], Grey Wolf Optimization (GWO) [7], Crow Search Optimization (CSO) [8,9], and Dragonfly Optimization (DFO) [10]. Alternative approaches are also used, such as adopting new network models and topologies [11,12] to make the WSN work effectively with available resources [13,14]. In the current study, we mainly intended to develop an energy efficient routing mechanism using artificial intelligence to select the optimal path routing in a WSN. Although the existing approaches have attempted to provide routing protocols through cluster formation, most of them fall short in terms of energy consumption and time delay in the transmission, along with security threats in the network. Hence, the use of algorithmic hybridization leads to improvements in the network and improves the overall performance of the routing model used to route data from a sender to a receiver in the network. The aim of the proposed work is to provide an optimal path considering the parameters of computational time, performance, energy consumption, and resource usage. The enhanced system addresses the problem of computational complexities through an artificial intelligence-based model of meta-heuristic and optimization algorithms. For the implementation of this work, we employed an optimization algorithm, GWO, for cluster head selection and a meta-heuristic algorithm, DFO, for path determination. Both algorithms provide efficient transmission of data packets through clustering-based techniques, thereby finding the optimal path for the densely populated network. The proposed hybridization of GWO and DFO focuses on increasing the efficiency in performance in the selection of the optimal path, thereby using minimal energy for transmission. Finally, the outcome of the combined approach can be tested with the QoS metrics for bringing out the advantages of the system. Hence, using the proposed work of artificial intelligence-based optimal path selection in WSNs, the network achieves an extended lifetime through energy consumption and transfers the data reliably with minimum time.

This article is organized as follows. Section 2 provides the literature review. Section 3 details the materials and methods of the proposed method. Section 4 provides a comparative analysis and the results of the proposed method with other studies. Finally, Section 5 concludes with conclusions and future research prospects.

2. Literature Review

Energy efficiency is a key task for routing in WSN. The network lifetime and energy efficiency of earlier studies showed less efficiency; hence, optimized routing systems are used for IoT networks in 5G environments [15]. The k-means algorithm used for the clustering phase and the nodes in the network perform multi-hop routing to create a chain so that the data will be transferred to the control center, which ensures the efficient use of energy. Studies have generated a newer set of protocols [16] to overcome the packet routing challenges specifically for the optimized routing. The energy would be used to make predictions and at the time of connection failures. Hence, the energy efficient protocol is used along with fuzzy rules, which prevent link failures. As a result, energy loss during link failures is prevented. Similar to earlier studies, Exponentially Ant Lion Whale Optimization (E-ALWO) was used to optimize the routing [17]. The optimization by this hybrid algorithm is executed through energy, delay, trust, and distance. The integration of the exponentially weighted moving average is used to optimize the above parameters based on the fitness function.

The hybrid approach in WSN networks has resulted in efficient outcomes, such as the hybridization of firefly and DFO. The static and dynamic properties of the DFO can be explored further and the firefly technique provides the optimal global solution though exploitation [18]. GWO is known for its search precision in IoT networks. Studies have enhanced it to provide an optimum solution [19]. Finding a global solution for the problem using GWO is enhanced but with the improved position update. In this work, it was used for the coverage problem and yielded effective results.

GWO in node deployment could reduce the computational cost for the WSN. The appropriate position of sensor nodes' deployment has been tested under various coverage scenarios and connectivity configurations [20]. The multi-objective fitness function makes it efficient for optimizing the cost. The coverage optimization by the GWO algorithm with simulated annealing is the coverage optimization of the WSN. By optimizing the coverage problem, energy consumption and e redundancy of the nodes during distribution can be reduced, and the network life can be improved. Annealing enables the GWO properties and improves the global optimization ability and is found to be better than the PSO [21]. Delayed response time, queueing, and increased packet loss reduce the transmission rate. It could be controlled by adjusting the traffic load in the downstream node. This load adjustment could be achieved with multiple classifications [22].

In summary, the above literature indicates that improving efficiency was achieved through various approaches. The optimization techniques mainly revolve around finding the optimum route in the WSN. The significant issue to alleviate energy consumption is the optimum path selection. Therefore, several authors have adopted different optimization techniques.

Based on the literature review, the following objectives are framed for this study:

1. To optimize the data transfer route in the WSN using the optimization techniques.
2. To develop a hybrid approach based on GWO and DFO to optimize the network lifetime and reduce energy consumption.

3. Materials and Methods

The proposed method is based on GWO and DFO, forming a hybrid approach to improve energy efficiency and network lifetime. We used GWO for the exploration and DFO for finding the optimized solution.

3.1. K-Means Clustering

K-means clustering is used to make clusters with similar data points to identify the pattern hidden in a dataset. It finds an association between two points, and groups are formed in multiple numbers depending on the data available. Here, in Algorithm 1, for clustering, the nodes in the networks are treated as data and the clusters are formed using the distance between the two nodes. With all the data, we carry out the process of finding the points which are related to each other. On that basis, related points are grouped in a process called clustering. For grouping, data are divided into a certain number of groups and the clusters are referred to as W, defined in the algorithm. Based on the similarity of data points to each other, clusters are formed on W groups. W can also be referred to as the number of centroids in the dataset, which is considered a location that determines the center of a cluster.

Algorithm 1: K-means clustering

```
1. While True do
2. for i = 1 to m ... do
3. $u^{(i)}$: = index (1 to k) cluster centroid closest to $p^i$
4. end for
5. for W = 1 to W ... do
6. $\mu_{(W)}$: = average (mean) of points assigned to cluster W.
7. end for
8. end While
```

Based on Algorithm 1, W defines the total number of centroids, and P^i is considered the current data point. $u^{\ (i)}$ gives the centroid's index measure, expressed by Equation (1):

$$u^i = min \left|\left| \ p^{(i)} - \mu_{(w)} \right|\right| \tag{1}$$

If $p^{(1)}$, $p^{(5)}$, and $p^{(7)}$ belong to one centroid, then with the new position, Equation (2) can be expressed as follows:

$$\mu_{(2)} = \frac{1}{3}\left(p^{(1)} + p^{(5)} + p^{(7)}\right) \tag{2}$$

The algorithm steps are repeated until the grouping of all nodes in the network is accomplished.

3.2. Dragonfly Algorithm (DFO)

The dragonfly algorithm (DFO) was used to find the optimal local solution for this study. The insect-inspired algorithm uses attraction and distraction methods to find the best solution. The static and dynamic behavior of the algorithm makes it effective in searching for an optimal solution in a specific area. In this study, the DFO is used to search for the target node within the selected cluster. The optimum path selection for data transfer is obtained by finding the optimal path from the available paths in the WSN.

Algorithm 2 shows the steps for the DFO, where the food source is the target node and the enemy is the node that is furthest from the target node. Based on this approach, the target node is easily accessed with optimized routing within the cluster. This technique would be effective for a system where each cluster has multiple sensor nodes.

Algorithm 2: Dragon-fly

```
Initialization of population $Qi$ ($i$ = 1,2,3, . . . , $n$)
Step vectors initialization $Qi$ ($i$ = 1,2,3, . . . , $n$)
**While** (condition)
    Calculation of the objective values of all the dragon flies
    Updating the position for food source and enemy
    Updating the values of $R$, $W$, $M$, $K$, $N$ and $L$ estimated in: Equations from (3) to (9)
    Neighbouring radius updating process
    If the dragonfly has only one neighboring fly
        Updating velocity vectors using the Equation (8)
        Updating the position vectors using Equation (9)
    **Else**
        Levy Flight is used to Update the position vector
    **End if**
    Checking and correcting the new - positions
**End while**
```

$$R_i = -\sum_{j=1}^{N} Q - Q_i \tag{3}$$

In Equation (3), Q indicates this individual position. Q_i is the ith neighboring dragonfly position. N is the total number of individual neighbors of the dragonfly swarm, and S indicates the separation motion for the ith individual.

$$M_i = \frac{\sum_{j=1}^{N} D_i}{N} \tag{4}$$

The alignment is calculated using Equation (4), where M_i is the alignment motion. D_i is the velocity of the ith neighboring dragonfly.

$$K_i = \frac{\sum_{i=1}^{N} Q_i}{N} - Q \tag{5}$$

Equation (5) represents the cohesion, where K_i is the ith individual cohesion, N is neighborhood size, Q_j is the ith neighboring dragonfly position, and X is the current dragonfly individual.

$$T_i = Q^{+} \; - \; Q \tag{6}$$

The attraction motion T_i for searching for food is calculated using Equation (6), where Q^{+} is the food position and Q is the position of the dragonfly.

$$L_i = Q^{-} + Q \tag{7}$$

The distraction or escape motion L_i for searching for food is calculated using Equation (7), where Q^{-} is the enemy's position and Q is the dragonfly position.

$$\Delta Q_{t+1} = (rR_i + mM_i + kK_i + nT_i + lL_i) + w\Delta Q_t \tag{8}$$

The step vector, denoted by the symbol Q_t, can be found by solving Equation (6). Similarly to how PSO's "position update" function operates, this PSO-inspired equation provides a solution. The step vector sums Equations (3)–(8) with inertia, where weight w and t are the iteration.

$$Q_{t+1} = Q_t + \Delta Q_{t+1} \tag{9}$$

Equation (9) gives the position of the dragonfly in the current iteration t.

In case of the absence of neighboring solutions, the swarm improves the randomness of the search by applying the random walk. This refers to a levy flight mechanism. The

DFO can choose a food source from the smaller population, and in a WSN, the number of nodes in a cluster could be less or more. This makes the DFO an ideal optimization technique for finding the optimal local solution. It is also quicker in terms of searching, and has fewer parameters while configuring. However, the exploitation behavior and large search steps of DFO make it difficult to optimize the global solution. Therefore, in this study, the DFO exploitation behavior is applied only for finding the local optimal solution [23].

3.3. Grey Wolf Optimization

The grey wolf method in cluster analysis is derived from the swarm intelligence technique. This algorithm follows the natural behavior of the grey wolf considering its efficient strategy of leadership and hunting. This particular animal has the quality of a strict leader, which can be used for the proposed system with the implementation of a cluster head. The four different types of wolves are alpha, delta, beta, and omega. The alpha wolf is the leader of the group and provides instructions to the others to follow, and the beta wolf helps the alpha in making better decisions and examines the instructions given by alpha. Delta wolves are deemed the subordinates, observing the instructions from the other wolves.

These wolf functionalities are manipulated using mathematical notations. The wolf naturally follows certain processes in the strategy of hunting, such as prey prediction, chasing, approaching, harassing, encircling, and finally attacking the prey. They identify prey using search space determination and then form a circle around the prey. After encircling it, they move closer to the prey slowly and reduce their fast movement. After confirming that the prey is close, they start attacking it. During the process of sending packets to the sink within the given network time, the node sends its own information along with the remaining energy it holds to the destination sink node. All this information is fed into the module of the GWO clustering phase for clustering the nodes optimally into a predefined count of cluster heads, checking for the maximum remaining energy that each cluster head holds [24]. The detailed implementation of GWO in this study is given in the next section.

3.4. Overall Flow of the Architecture

In the meta-heuristic ideology, the term hybridization is generally defined as the process of merging the powerful characteristics of two or more algorithms to generate an algorithm based on the features of the merged concepts. The proposed hybridization of algorithms follows a two-stage process. In the initial stage, it starts with generating the wolf population. After the generation of the population, the fitness of the agents must be determined, followed by updating the wolf position. In the next stage of the process, it calculates the fitness again and updates the position using the dragonfly algorithm. During the process of updating, it calculates the position and velocity vectors for which it calculates the cosine distance with other flies. Through the continuous process of calculating the fitness and updating the positions, it can yield the optimal solution.

Generally, WSNs are constrained in the factors of energy efficiency and communication ability. The proposed work mainly focuses on the design of energy and computationally efficient algorithms. It may provide continuously controllable energy to the dead sensors to resume the operation, as mentioned in Figure 1. The other significant aspect of this work is improving the operating performance through the process of the base station forming unequal size clusters and selecting cluster heads, where the clusters much closer to the base station have a smaller size. The system builds an efficient path among the cluster heads. The geographic-based routing enables adaptive load balancing and it has the responsibility of selecting the relay nodes. Through clustering in the present work, scalability is greatly enhanced and reduces the network traffic. It also helps in preventing redundant messages from propagating over the network. In a clustering network, the cluster heads play a significant role, and hence, CH selection is an important task handled by the GWO algorithm. Traditional optimization algorithms lag in providing the appropriate solution within a specified time. Hence, the proposed work uses a meta-heuristic algorithm for optimization. The proposed hybridization of the GWO and dragonfly algorithm mainly intends to provide the best solution based on the fitness function calculated, which gives out the minimum number of clusters with extensible link quality, and the CH, selected dynamically, provides high remaining. This process is described using the equations explained in the next section.

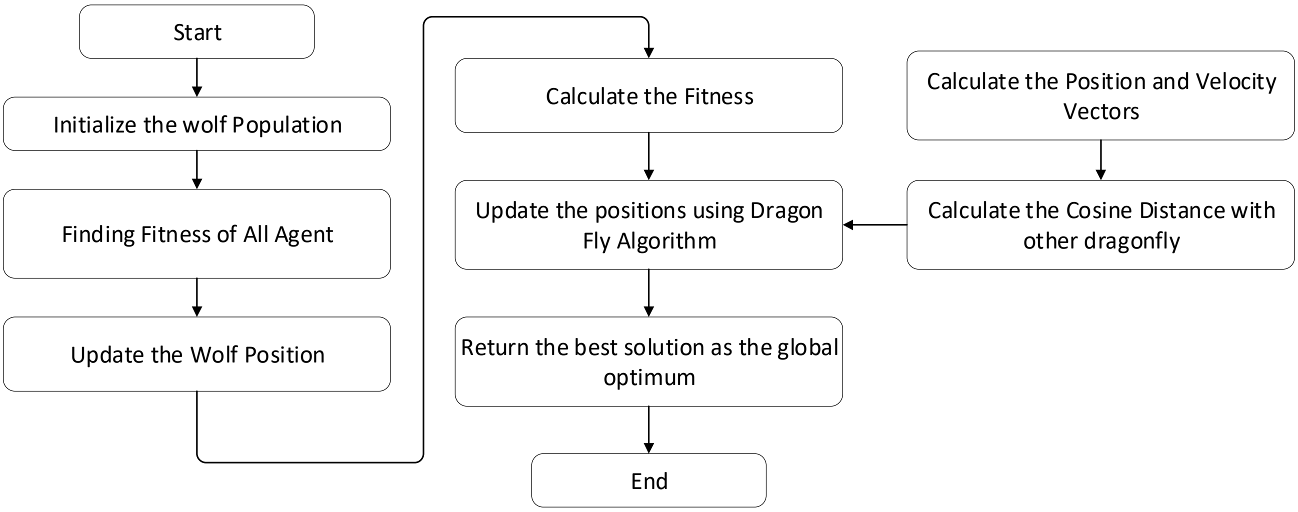
Start
Initialize the wolf Population
Finding Fitness of All Agent
Update the Wolf Position
Calculate the Fitness
Update the positions using Dragon Fly Algorithm
Return the best solution as the global optimum
End
Calculate the Position and Velocity Vectors
Calculate the Cosine Distance with other dragonfly

Figure 1. Overall flow of the architecture.

The GWO algorithm performs the selection of CH by calculating the fitness function. It categorizes the wolves into four types, namely alpha, beta, delta, and omega. The leader position is nominated to the alpha category of wolves. All these estimations are calculated using these equations. By considering this optimization algorithm, the alpha solution is the highly optimal solution. The next two best solutions are the beta and delta solutions. The given equation uses the GWO algorithm and represents the encircling phase of the model.

$$\overrightarrow{X}(i+1) = \overrightarrow{x_p}(i) + \overrightarrow{A} \times \overrightarrow{D} \tag{10}$$

Equation (10) gives the encircling phase, where $\overrightarrow{x_p}$ is the location of the prey, $\overrightarrow{X}$ is the location of the wolf, and $\overrightarrow{A}$ and $\overrightarrow{D}$ are the coefficient vectors. Here, the prey is the target node and the wolf is the control center, and the data must be transferred between the node and the control center. The consecutive nodes will be further selected to facilitate the routing between the control center and the target node.

$$\overrightarrow{D} = \left| \overrightarrow{C} \times \overrightarrow{X}_p(i) - \overrightarrow{X}(i) \right| \tag{11}$$

$$\overrightarrow{A} = 2\overrightarrow{A} \times \overrightarrow{r_1} - \overrightarrow{b} \tag{12}$$

The variables $\overrightarrow{A}$ and $\overrightarrow{D}$, used in Equation (10), may be written as Equations (11) and (12), respectively. The variables $\overrightarrow{C}$ in Equation (11) can be found in Equation (13).

$$\overrightarrow{C} = 2\overrightarrow{r_2} \tag{13}$$

$\vec{r_1}$ and $\vec{r_2}$ are the random vectors for the Equations (12) and (13). The $\vec{b}$ reduces from (2 to 0) with respect to the number of iterations. Equation (14) updates the position of the wolf.

$$\vec{Q}\ (i+1) = \frac{\vec{Q_1} + \vec{Q_2} + \vec{Q_3}}{3} \tag{14}$$

After calculating fitness, Equation (15) updates the position by the DA position function:

$$Q_{t+1} = Q_t + \Delta D_{t+1} \tag{15}$$

The parameter in $\vec{b}$ GWO is controlled to maintain the exploration process for the proposed model.

$$\vec{b} = 2 - i \times \frac{2}{Max_i ter} \tag{16}$$

In Equation (16), the maximum number of iterations is given by $Max_i ter$.

$$N = \sum_{j=1}^{x} \sum_{i=1}^{k} \parallel Q_i^j - c_j \parallel^2 \tag{17}$$

Equation (17) gives the cluster formation for the nodes. One node acts as a cluster head. To select the cluster head,

$$CH_i = max_{nodes}\ \in\ \ C_i\ RemainingEnergy \tag{18}$$

Cluster head CH_i selection is based on Equation (18). The program is executed and the results are obtained. The GWO can find search for path that are faster, more precise and security to transfer data, which makes it easier to search for the global solution in a WSN. The hierarchy-based decision models could easily identify the cluster. Additionally, GWO has the slowest convergence, which makes it inefficient in finding the local optimal solution. Thus, it would be effective to apply GWO in a global search [25].

In summary, GWO takes an exploration approach to identify the optimized route for the data transfer and DFO assists in finding the local optimal solution to select the appropriate node. With help of the cluster head, the nodes that require data transfer could be easily detected for subsequent processing. The average residual energy with the number of rounds was measured for further comparison.

4. Comparative Analysis and Results

In this study, the efficiency and execution of the proposed work are correlated with the methods based on the algorithms of machine learning (ML) and routing protocols. The simulation parameters used for the analysis of wireless sensor networks are depicted in Table 1.

Table 1. Simulation parameters used for the analysis of wireless sensor networks.

Parameter	Value
3D area	100 × 100 × 50 m
Simulation time	3600 s
No. of nodes	100
No. of layers	varied from 1 to 9
Transmission range	30 m
Routing protocol	Leach
Type of antenna	Omnidirectional
Message rate	1 packet/s
Size of message	38 bytes
Energy model	Generic
Transmit circuitry power consumption	24.75 mW
Receive circuitry power consumption	13.5 mW
Idle circuitry power consumption	13.5 mW
Sleep circuitry power consumption	0.05 mW

The results obtained for the proposed methodology and the performance evaluation and comparison with other studies are discussed in this section. The energy consumed for data transfer, the data transfer rate, the computation time, and the criteria of the first dead node end-to-end delay and throughput are compared. Figure 2 shows the comparative analysis between the proposed method and other studies for the average residual energy spent on the number of rounds. Residual energy is the total amount of energy depleted in each stage of the nodes. The outcomes of the simulation graphs signify that the proposed method has better recorded residual energy than Optimal Cluster-Based Routing (CBR). Compared with k-means, Chain-Based Hierarchical Routing Protocol (CHIRON), and Low-Energy Adaptive Clustering Hierarchy Centralized (LEACH-C) protocols, the proposed method has an increased lifespan. This is mainly due to the quicker optimum solution techniques of the combined GWO-DFO technique.

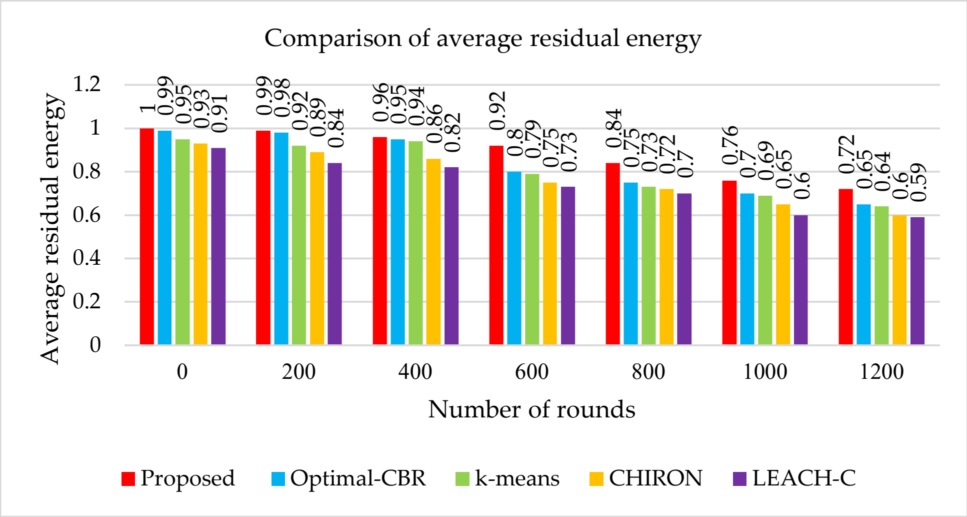

Figure 2. Comparative analysis between the proposed methods and several other methods.

An improved k-means algorithm is used to choose a cluster head and transform it into a chain route in a network just after the threshold value is larger than the node energy.

Another method is a basic clustering algorithm termed clustering by k-means that belongs to an unreasoned approach of clustering. The estimation of the mean value (1, 10) as a centroid is estimated by partitioning the provided node set into k-clusters. The steps involved in the k-means algorithm are as follows:

- The selection of k points non-specifically as a centroid in which k is considered to be a number of clusters.
- Allocate every node to the nearest median (centroid) by using Euclidean distance to produce clusters.
- Calculate the latent centroid for each cluster.
- The repetition of the last two steps is performed until no fluctuation occurs in the centroid of each cluster.

By eliminating some sensors in a network, this method extends its useful life while also improving performance metrics.

Another protocol implemented here is a chain-based routing approach called the Chain-Based Hierarchical Routing Protocol (CHIRON) used to reduce several defects related to the data dissemination delay. In the initial stage, the segregation of the network causes various fan-shaped fragments to occur. The base station supplies the control message to each individual node in a group and helps the node determine the group it belongs to. The second step involves far-away nodes available from a base station activated to produce group chains inside a single group. A greedy algorithm is implemented to find the nearest node associated with a node that converts to a new node, which starts the consecutive association levels. The leader/head node is elected in the third level or step and the distant node away from the base station is the head in a community chain. Then, the node with higher residual energy is elected as the chain head of a group. The transmission of data to a group chain head in every group over a chain is implemented in the fourth step. Finally, the obtained data are dispatched to a base station using a leader-to-leader transmission method. This method provides low energy dispersion and clustering overhead can occur.

The other method used is LEACH-C (Low-Energy Adaptive Clustering Hierarchy Centralized), an individual-hop data transmission structure where cluster heads give the combined information to a base station, along with nodes' endurance level. The head node is the one whose endurance is above the threshold for the ongoing round, and the data are circulated throughout the entire arrangement. The matching of circulated identity to the core identity is interpreted for a lifetime of rounds. This protocol is mainly useful for short distances. The dissipation of endurance usage among available hubs is provided by LEACH-C, but this method is not easily upgraded or updated.

Therefore, all procedures executing WSNs are compared with the proposed work to evaluate the average of the residual energy determined by the score of rounds. The average of the residual energy levels determined by the score rounds is calculated by various methods and classified in Table 2.

Table 2. Differentiation of the average of residual energy along with the score of rounds in various methods.

Various Methods and Number of Rounds	0	200	400	600	800	1000	1200
Proposed	1	0.99	0.96	0.92	0.84	0.76	0.72
LEACH-C	0.91	0.84	0.82	0.73	0.7	0.6	0.59
Optimal-CBR	0.99	0.98	0.95	0.8	0.75	0.7	0.65
CHIRON	0.93	0.89	0.86	0.75	0.72	0.65	0.6
k-means	0.95	0.92	0.94	0.79	0.73	0.69	0.64

The various methods for routing protocols and machine learning algorithms are analyzed and the outcomes are tabulated. From Table 1, the average residual energy by the number of rounds by Optimal-CBR is 0.99 per 0 rounds, 0.98 per 200 rounds, 0.95

per 400 rounds, 0.8 per 600 round, 0.75 per 800 rounds, 0.7 per 1000 rounds, and 0.65 per 1200 rounds. K-means produced 0.95 per 1, 0.92 per 200, 0.94 per 400, 0.79 per 600, 0.73 per 800, 0.69 per 1000, and 0.64 per 1200. The CHIRON method yields 0.93 per 0, 0.89 per 200, 0.86 per 400, 0.75 per 600, 0.72 per 800, 0.65 per 1000, and 0.6 per 1200. LEACH-C gives 0.91 per 0, 0.84 per 200, 0.82 per 400, 0.73 per 600, 0.7 per 800, 0.6 per 1000, and 0.59 per 1200. Among these methods, our proposed method has higher average residual energy by the number of rounds than the other methods, and the values of the proposed method are 1 per 0, 0.99 per 200, 0.96 per 400, 0.92 per 600, 0.84 per 800, 0.76 per 1000, and 0.72 per 1200. Therefore, our proposed system scored high compared to the other methods.

The node with a transmission latency higher than the threshold value in the WSN is called a dead node. The presence of a dead node indicates the reliability of the WSN. The comparative analysis of the first dead node for validation is shown in Figure 3. The measured value is greater for the proposed model than for the other values. This is due to the threshold configured in the models. It also depends on the user-defined data presented in the node. The first dead node was found after 500 rounds, showing that the proposed model has nodes with better reliability.

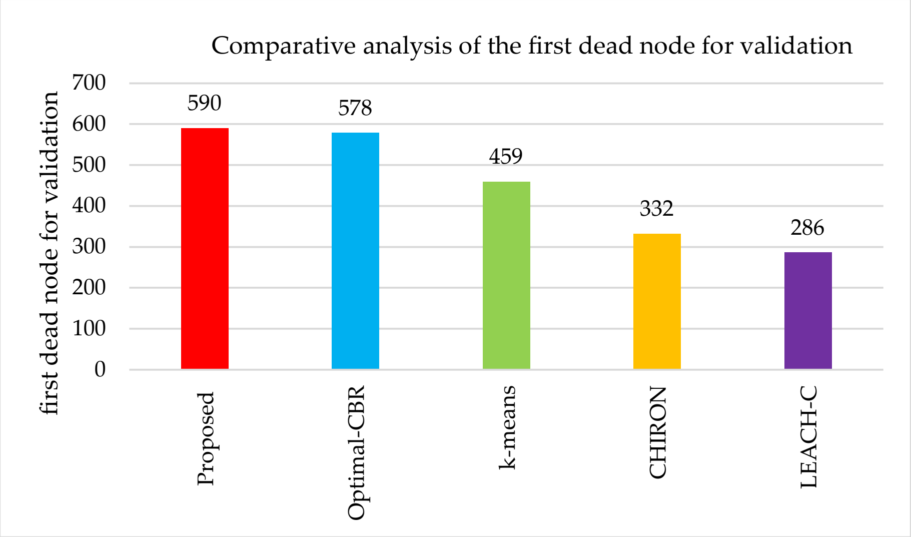

Figure 3. Comparative analysis of the first dead node for validation by the number of rounds in various methods.

The efficiency of data transmission is determined by examining the alive nodes since the looping proceeds because of the network lifetime. In Optimal-CBR, the first node dead by the rounds of communication is higher due to the threshold of user-defined data available in every node, which in turn decreases the energy used by the cluster head. Hence, several methods for routing protocols are employed to compare with the proposed method for the first dead node for validation by the rounds of communication, and the values are categorized in Table 3.

Table 3. Differentiation of first dead node for validation in several methods.

Various Protocol Methods	First Dead Node for Validation
Proposed	590
LEACH-C	286
Optimal-CBR	578
CHIRON	332
k-means	459

In Table 3, the first dead node for validation by the rounds of LEACH-C, Optimal-CBR, CHIRON, and k-means are 286, 578, 332, and 459, respectively. However, the proposed method leads with 590. The end-to-end latency analysis used for performance analysis is shown in Figure 4. It is seen that less delay is obtained with the proposed algorithm compared to other algorithms.

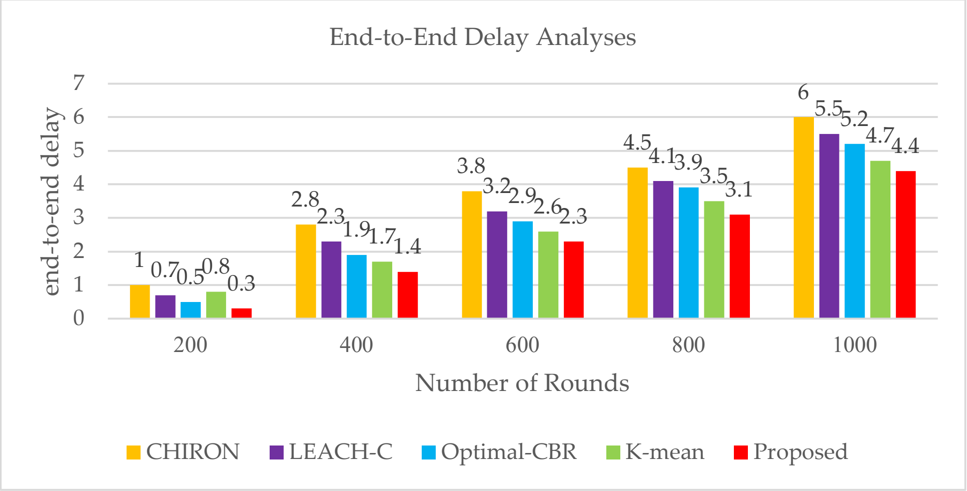

Figure 4. Comparison end-to-end delay analyses.

The occurrence of a delay in the proposed method is compared to Optimal Cluster-Based Routing (CBR), k-means, Chain-Based Hierarchical Routing Protocol (CHIRON), and Low-Energy Adaptive Clustering Hierarchy Centralized (LEACH-C) in Table 4.

Table 4. Differentiation of end-to-end delay by several methods.

End-to-End Delay and Number of Rounds	200	400	600	800	1000
CHIRON	1	2.8	3.8	4.5	6
LEACH-C	0.7	2.3	3.2	4.1	5.5
Optimal-CBR	0.5	1.9	2.9	3.9	5.2
K-mean	0.8	1.7	2.6	3.5	4.7
Proposed	0.3	1.4	2.3	3.1	4.4

Table 4 shows that the end-to-end delay by the number of rounds by Optimal-CBR is 0.5 per 200 rounds, 1.9 per 400 rounds, 2.9 per 600 rounds, 3.9 per 800 rounds, and 5.2 per 1000 rounds. The method k-means produced 0.8 per 200, 1.7 per 400, 2.6 per 600, 3.5 per 800, and 4.7 per 1000. The CHIRON method yields 1 per 200, 2.8 per 400, 3.8 per 600, 4.5 per 800, and 6 per 1000. LEACH-C gives 0.7 per 200, 2.3 per 400, 3.2 per 600, 4.1 per 800, and 5.5 per 1000. Among all these methods, the proposed method has lower end-to-end delay by the number of rounds than the other methods, and the values in the proposed method are 1 per 0.3 per 200, 1.4 per 400, 2.3 per 600, 3.1 per 800, and 4.4 per 1000. Therefore, our proposed system scored low in the end-to-end delay by the number of rounds compared to k-means, LEACH-C, CHIRON, and Optimal-CBR.

The simulation graphs in Figure 5 show that the proposed method has better recorded throughput than Optimal Cluster-Based Routing (CBR). Compared with the k-means, Chain-Based Hierarchical Routing Protocol (CHIRON), and Low-Energy Adaptive Clustering Hierarchy Centralized (LEACH-C) protocols, the proposed method has an increased lifespan. This is mainly due to the quicker optimum solution techniques of the combined GWO-DFO technique.

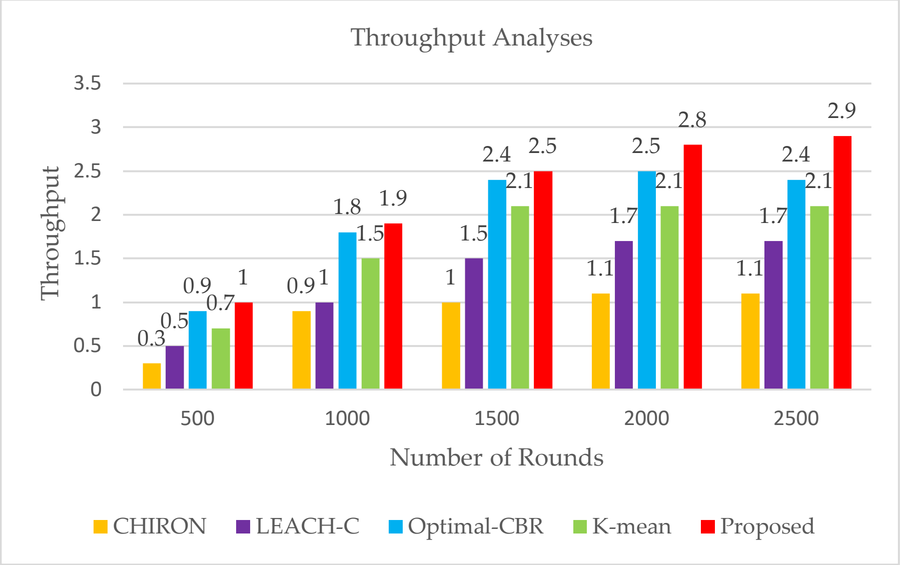

Figure 5. Comparison throughput analyses.

The calculation of the throughput value of the proposed simulator is given in Equation (19). Here $D_{successful\ delivered}$ is refers to the successful delivered packet in each communication, $D_{Average\ Size}$ is refers to the average packet size, and $D_{Sent\ Time}$ is refers to the total time taken for the sent packets. The list of throughput values measured is given in Table 5.

$$Throughput = \frac{\sum_{i=1}^{node}\left(D_{Successful\ delivered}\right) \times \left(D_{Average\ Size}\right)}{D_{Sent\ Time}} \tag{19}$$

Table 5. Differentiation of throughput in several methods.

Various Methods and Number of Rounds	500	1000	1500	2000	2500
CHIRON	0.3	0.9	1	1.1	1.1
LEACH-C	0.5	1	1.5	1.7	1.7
Optimal-CBR	0.9	1.8	2.4	2.5	2.4
K-mean	0.7	1.5	2.1	2.1	2.1
Proposed	1	1.9	2.5	2.8	2.9

Table 5 illustrates that throughput by the number of rounds by Optimal-CBR is 0.9 per 500 rounds, 1.8 per 1000 rounds, 2.4 per 1500 rounds, 2.5 per 2000 round, and 2.4 per 25,000 rounds. K-means produced 0.7 per 500, 1.5 per 1000, 2.1 per 1500, 2.1 per 2000, and 2.1 per 2500. The CHIRON method yields 0.3 per 500, 0.9 per 1000, 1 per 1500, 1.1 per 2000, and 1.1 per 2500. LEACH-C gives 0.5 per 500, 1 per 1000, 1.5 per 1500, 1.7 per 2000, and 1.7 per 2500. The proposed method has higher throughput by the number of rounds than the other methods. The values in the proposed method are 1 per 500, 1.9 per 1000, 2.5 per 1500, 2.8 per 2000, and 2.9 per 25,000. Therefore, from the above interpretation, our proposed system scored high in the throughput by the number of rounds compared to k-means, LEACH-C, CHI-RON, and Optimal-CBR.

The packet transmission rate comparison is shown in Figure 6. TC-SSA [26], EAT-SRA [27], ETGSA [28], ECATS [29], and SEORMP [16] are used for the comparison. It can be observed that the transmission rate reduces with the mobility. The earlier research models are lower than the proposed model and SEORMP model. The amount of data transfer that occurs shows the ability of the model to provide stability for a longer lifetime. For the proposed model, the transmission rate was higher at the initial stage, where the mobility was 1 m/s, and it was measured to be around 99.30, which is similar to the model EATSRA [27].

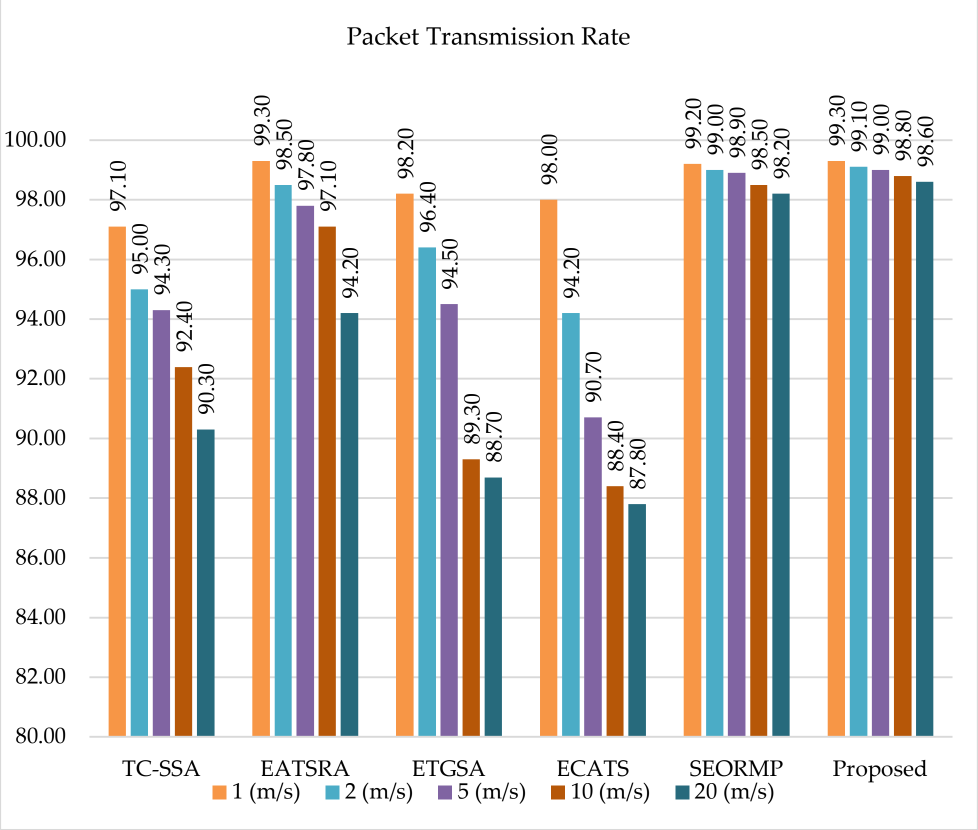

Figure 6. Comparison of packet transmission rate depending on mobility speed.

During the packet transmission, the status of nodes is mobile and there is a decrease in the energy level. According to many researchers, every node will maintain a routing table to recognize the transmitted node or nearby node. The maintenance of the routing table to create an effective protocol is the greatest challenge in WSN. Therefore, this comparative analysis is carried out to find an optimal routing protocol.

The optimization of the routing protocol is carried out with the aid of the gravitational search technique, considering indirect and direct trust in nodes and concern over energy saving. This routing approach is termed ETGSA (Energy-Aware Trust-Based Gravitational Search Approach). It is proved that the ETGSA algorithm maximizes the reliability to resist certain attacks in a network and decreases the computing overhead, and NS-2 is used for the simulation process [28].

With the help of the DT algorithm and the spatio-temporal constraint, wireless sensor networks can identify malicious users and choose safe paths. This algorithm develops security and a suitable delivery ratio, and to choose efficient decisions, certain kinds of constraints called spatio-temporal constraints are introduced in this model [27].

After the experimental analysis, the overall production of all systems is presented graphically in Figure 6. The ratio of transmission rate in a packet along with motility speed is calculated, and this criterion is analyzed with many systems and the proposed approach. Among existing approaches, the mobility of a node in the path construction is available in the Secured Energy Optimal Routing with Mobility Prediction (SEORMP) approach. By analyzing the total trust of a node and its instant trust, SEORMP builds an efficient and secure route. The energy consumption in a node is modified after the packet transmission maximizes throughput and reduces the time delay in the packet transmission. However, the proposed method, in terms of packet transmission rate, performs better than the SEORMP technique.

Many available studies, while suggesting an optimal path, do not take the mobility speed into account. On the basis of security concerns, a path is created given the path and energy level in a node. If mobility is considered, then automatically, time concern is dispersed. The comparative analysis of several systems along with the proposed system in terms of the transmission rate of the packet depending on the speed of mobility is categorized in Table 6.

Table 6. Comparative analysis of the transmission rate of a packet concerning speed among various methods.

Mobility Speed	ETGSA	SEORMP	TC-SSA	Proposed	ECATS	EATSRA
1	98.2	99.2	97.1	99.3	98	99.3
2	96.4	99	95	99.1	94.2	98.5
5	94.5	98.9	94.3	99	90.7	97.8
10	89.3	98.5	92.4	98.8	88.4	97.1
20	88.7	98.2	90.3	98.6	87.8	94.2

Table 6 shows that the packet transmission rate of the proposed system is higher than other methods such as Energy-Aware Trust-Based Gravitational Search Approach (ETGSA), Taylor-Based Cat Salp Swarm Optimization Algorithm (TC-SSA), Enhanced Fuzzy C-Means and Adaptive TDMA Scheduling (ECATS), and Energy-Aware Trust-Based Secure Routing Algorithm (EATSRA). When the mobility speed is 1, the packet transmission rates of ETGSA, SEROMP, TC-SSA, the proposed system, ECATS, and EATSRA are 98.2, 99.2, 97.1, 99.3, 98, and 99.3, respectively. When the motility speed changed to 2, the corresponding packet transmission rates of ETGSA, SEORMP, TC-SSA, the proposed system, ECATS, and EATSRA were 96.4, 99, 95, 99.1, 94.2, and 98.5, respectively. If the mobility speed increases to 5, the transmission rates of packets in ETGSA, SEORMP, TC-SSA, the proposed system, ECATS, and EATSRA are 94.5, 98.9, 94.3, 99, 90.7, and 97.8, respectively. The packet transmission rates of ETGSA, SEORMP, TC-SSA, the proposed system, ECATS, and EATSRA when the mobility speed is 10 are 89.3, 98.5, 92.4.98.8, 88.4, and 97.1, respectively. If the mobility speed increases to 20, then the packet transmission rates of ETGSA, SEORMP, TC-SSA, the proposed system, ECATS, and EATSRA are 88.7, 98.2, 90.3, 98.6, 87.8, and 94.2, respectively. Table 6 explains that the proposed system achieves a higher packet transmission rate based on the mobility speed compared to the other methods.

A graphical representation of the comparative analysis of the proposed method with other techniques such as RPL, EESRA, E-ALWO, MDSMAP, and GWO based on the computational time is illustrated in Figure 7. With the proposed technique, less computation time for data transfer is achieved compared to GWO [30], EESRA [31], MDS-MAP [32], RPL and E-ALWO [17]. This shows that the proposed method is energy efficient, as the more the computational power used, the more the model consumes energy for the process. Here, GWO implemented alone suffered from slow convergence and failed to find the local optimum effectively, having to search repeatedly and thus increasing the time to find the optimum solution.

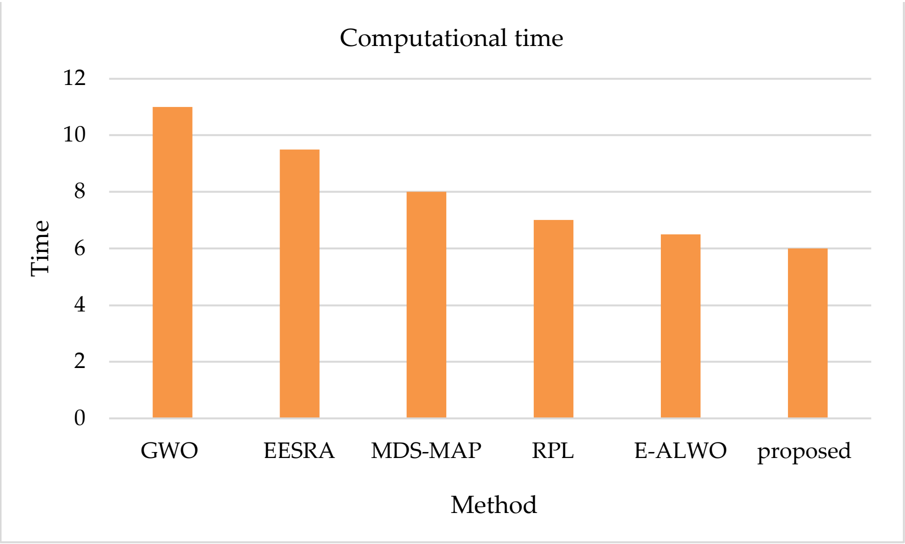

Figure 7. Computational time comparison with other studies.

The most efficient algorithms are those with low computational time and minimum steps. The computational time is defined as the time taken by an algorithm to compute a given task. the algorithm with minimum steps accomplishes the task with minimal computation time.

The same issue can be solved by many algorithms with different computation times, but the steps implemented by different routing protocols differ, which leads to high computation time in some cases. The algorithm with the minimum number of steps to resolve an issue has the lowest computational time. The computational times of the proposed technique and various techniques such as RPL, MSD-MAP, EESRA, E-ALWO, and GWO are calculated and categorized in Table 7.

Table 7. A comparison of the computational times of various protocols.

Method	PROPOSED	GWO	RPL	MDS-MAP	EESRA	E-ALWO
Time	6	11	7	8	9.5	6.5

Table 7 states that the computational time required for the proposed system is the lowest, 6 s, while the computational time taken by Grey Wolf Optimizer is 11 s. The computational time needed by the Routing Protocol for Low-Power and Lossy Networks is 7 s and the computational time of the Multidimensional-Scaling Map is 8 s. The computational time needed by an Efficient Energy Scaling Routing Algorithm is 9.5 s, and finally, the computational time required for the Exponentially Ant Lion Whale Optimization algorithm is 6.5 s. Therefore, our proposed system is efficient in terms of computational time since it provides the lowest computational time of the methods. The algorithm E-ALWO consumes less computational time than other existing methods such as GWO, MDS-MAP, RPL, and EESRA, but our proposed method beats the E-ALWO algorithm and secures the first position in the least computational time consumed by an algorithm.

By analyzing and comparing our proposed method with several other methods, it is confirmed that our proposed method is efficient in many ways and is suitable for resolving the energy concerns in wireless sensor networks (WSNs).

5. Conclusions

WSNs are widely applied in various fields. The need to improve WSNs towards greater sustainability has become an active research area. Its rapid development could result in the generation of various sensors around the world. In the present study on improving energy efficiency and network lifetime by facilitating an optimized data transfer route between the nodes and the control center, we took advantage of exploration methods in GWO and identified an appropriate local solution by the DFO.

The hierarchical method for identifying clusters and the selection of target nodes by attraction and distraction methods produced an optimum path for data transfer between the control center and the target nodes. As we successfully generated an optimal path, the benefits of optimal routing resulted in energy efficiency and a longer network lifetime. The proposed method had the best average residual energy spent on the number of rounds compared to other models, and the first dead node was found after 500 rounds, showing that the proposed model has nodes with better reliability. The proposed system also had lower end-to-end delay by the number of rounds compared to the other methods. Although the simulated results showed the effectiveness of the routing, the dynamic behavior with a real-time network would have shown the real-time ability of the proposed model.

Eventually, many existing models and techniques can be hybridized with other routing protocols in wireless sensor networks and can collaborate with many advanced protocols to develop the optimal ratio of packet delivery. Hybridization of connectivity and range measurement should be employed in studies on wireless sensor networks in the coming years, leading to new algorithms that will enhance WSNs in the future.

Author Contributions: The first author (H.A.Y.) conducted experiments and wrote the manuscript. The second author (C.K.) conducted the research and contributed to editing, analyzing the work, and correcting the direction of the work, as well as structuring the research. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Gulati, K.; Boddu, R.S.K.; Kapila, D.; Bangare, S.L.; Chandnani, N.; Saravanan, G. A review paper on wireless sensor network techniques in Internet of Things (IoT). *Mater. Today Proc.* **2021**, *51*, 161–165. [CrossRef]
2. John, J.; Rodrigues, P. A survey of energy-aware cluster head selection techniques in wireless sensor network. *Evol. Intell.* **2019**, *15*, 1109–1121. [CrossRef]
3. Singh, O.; Rishiwal, V.; Chaudhry, R.; Yadav, M. Multi-Objective Optimization in WSN: Opportunities and Challenges. *Wirel. Pers. Commun.* **2021**, *121*, 127–152. [CrossRef]
4. Sun, W.; Tang, M.; Zhang, L.; Huo, Z.; Shu, L. A Survey of Using Swarm Intelligence Algorithms in IoT. *Sensors* **2020**, *20*, 1420. [CrossRef] [PubMed]
5. Natarajan, Y.; Raja, R.A.; Kousik, D.N.; Johri, P. Improved energy efficient wireless sensor networks using multicast particle swarm optimization. In Proceedings of the International Conference on Innovative Advancement in Engineering & Technology (IAET), Jaipur, India, 21 February 2020; pp. 1–6.
6. Li, Y.; Cao, J. WSN Node Optimal Deployment Algorithm Based on Adaptive Binary Particle Swarm Optimization. *ASP Trans. Internet Things* **2021**, *1*, 1–8. [CrossRef]
7. Chouhan, N.; Jain, S. Tunicate swarm Grey Wolf optimization for multi-path routing protocol in IoT assisted WSN networks. *J. Ambient. Intell. Humaniz. Comput.* **2020**, 1–17. [CrossRef]
8. Subramanian, P.; Sahayaraj, J.M.; Senthilkumar, S.; Alex, D.S. A hybrid grey wolf and crow search optimization algorithm-based optimal cluster head selection scheme for wireless sensor networks. *Wirel. Pers. Commun.* **2020**, *113*, 905–925. [CrossRef]
9. Kaushik, A.; Indu, S.; Gupta, D. A Grey Wolf Optimization Approach for Improving the Performance of Wireless Sensor Networks. *Wirel. Pers. Commun.* **2019**, *106*, 1429–1449. [CrossRef]
10. Alshinwan, M.; Abualigah, L.; Shehab, M.; Elaziz, M.A.; Khasawneh, A.M.; Alabool, H.; Hamad, H.A. Dragonfly algorithm: A comprehensive survey of its results, variants, and applications. *Multimed. Tools Appl.* **2021**, *80*, 14979–15016. [CrossRef]
11. Prasanth, A.; Jayachitra, S. A novel multi-objective optimization strategy for enhancing quality of service in IoT-enabled WSN applications. *Peer-to-Peer Netw. Appl.* **2020**, *13*, 1905–1920. [CrossRef]
12. Yan, P.; Choudhury, S.; Al-Turjman, F.; Al-Oqily, I. An energy-efficient topology control algorithm for optimizing the lifetime of wireless ad-hoc IoT networks in 5G and B5G. *Comput. Commun.* **2020**, *159*, 83–96. [CrossRef]
13. Cotrim, R.S.; Caldeira, J.M.L.P.; Soares, V.N.; Azzoug, Y. Power saving MAC protocols in wireless sensor networks: A survey. *Telecommun. Comput. Electron. Control.* **2021**, *19*, 1778–1786. [CrossRef]
14. Ezhilarasi, M.; Krishnaveni, V. A survey on wireless sensor network: Energy and lifetime perspective. *Taga J.* **2018**, *14*, 41–48.

15. Jothikumar, C.; Ramana, K.; Chakravarthy, V.D.; Singh, S.; Ra, I.-H. An efficient routing approach to maximize the lifetime of IoT-based wireless sensor networks in 5G and beyond. *Mob. Inf. Syst.* **2021**, *2021*, 9160516. [CrossRef]
16. Dhanalakshmi, B.; SaiRamesh, L.; Selvakumar, K. Intelligent energy-aware and secured QoS routing protocol with dynamic mobility estimation for wireless sensor networks. *Wirel. Netw.* **2021**, *27*, 1503–1514. [CrossRef]
17. SureshKumar, K.; Vimala, P. Energy efficient routing protocol using exponentially-ant lion whale optimization algorithm in wireless sensor networks. *Comput. Netw.* **2021**, *197*, 108250. [CrossRef]
18. Singh, P.; Mittal, N. Efficient localisation approach for WSNs using hybrid DA–FA algorithm. *IET Commun.* **2020**, *14*, 1975–1991. [CrossRef]
19. Miao, Z.; Yuan, X.; Zhou, F.; Qiu, X.; Song, Y.; Chen, K. Grey wolf optimizer with an enhanced hierarchy and its application to the wireless sensor network coverage optimization problem. *Appl. Soft Comput.* **2020**, *96*, 106602. [CrossRef]
20. Jaiswal, K.; Anand, V. A QoS aware optimal node deployment in wireless sensor network using Grey wolf optimization approach for IoT applications. *Telecommun. Syst.* **2021**, *78*, 559–576. [CrossRef]
21. Zhang, Y.; Cao, L.; Yue, Y.; Cai, Y.; Hang, B. A Novel Coverage Optimization Strategy Based on Grey Wolf Algorithm Optimized by Simulated Annealing for Wireless Sensor Networks. *Comput. Intell. Neurosci.* **2021**, *2021*, 6688408. [CrossRef]
22. Kazmi, H.S.Z.; Javaid, N.; Imran, M.; Outay, F. Congestion control in wireless sensor networks based on support vector machine, Grey Wolf optimization and differential evolution. In Proceedings of the 2019 Wireless Days (WD), Manchester, UK, 24–26 April 2019; pp. 1–8.
23. Rahman, C.M.; Rashid, T.A. Dragonfly algorithm and its applications in applied science survey. *Comput. Intell. Neurosci.* **2019**, *2019*, 9293617. [CrossRef]
24. Rajakumar, R.; Dinesh, K.; Vengattaraman, T. An energy-efficient cluster formation in wireless sensor network using grey wolf optimization. *Int. J. Appl. Manag. Sci.* **2021**, *13*, 124–140. [CrossRef]
25. Hou, Y.; Gao, H.; Wang, Z.; Du, C. Improved Grey Wolf Optimization Algorithm and Application. *Sensors* **2022**, *22*, 3810. [CrossRef]
26. Vinitha, A.; Rukmini, M.; Dhirajsunehra. Secure and energy aware multi-hop routing protocol in WSN using Taylor-based hybrid optimization algorithm. *J. King Saud Univ.—Comput. Inf. Sci.* **2019**, *34*, 1857–1868.
27. Selvi, M.; Thangaramya, K.; Ganapathy, S.; Kulothungan, K.; Nehemiah, H.K.; Kannan, A. An Energy Aware Trust Based Secure Routing Algorithm for Effective Communication in Wireless Sensor Networks. *Wirel. Pers. Commun.* **2019**, *105*, 1475–1490. [CrossRef]
28. Zahedi, A.; Parma, F. An energy-aware trust-based routing algorithm using gravitational search approach in wireless sensor networks. *Peer-to-Peer Netw. Appl.* **2018**, *12*, 167–176. [CrossRef]
29. Kavidha, V.; Ananthakumaran, S. Novel energy-efficient secure routing protocol for wireless sensor networks with Mobile sink. *Peer-to-Peer Netw. Appl.* **2018**, *12*, 881–892. [CrossRef]
30. Daneshvar, S.M.M.H.; Mohajer, P.A.A.; Mazinani, S.M. Energy-Efficient Routing in WSN: A Centralized Cluster-Based Approach via Grey Wolf Optimizer. *IEEE Access* **2019**, *7*, 170019–170031. [CrossRef]
31. Elsmany, E.F.A.; Omar, M.A.; Wan, T.-C.; Altahir, A.A. EESRA: Energy Efficient Scalable Routing Algorithm for Wireless Sensor Networks. *IEEE Access* **2019**, *7*, 96974–96983. [CrossRef]
32. Beheshtiasl, A.; Ghaffari, A. Secure and Trust-Aware Routing Scheme in Wireless Sensor Networks. *Wirel. Pers. Commun.* **2019**, *107*, 1799–1814. [CrossRef]