Alexandria Engineering Journal (2023) 63, 339–357

HOSTED BY

ELSEVIER

Alexandria University

Alexandria Engineering Journal

www.elsevier.com/locate/aej
www.sciencedirect.com

Optimal data transmission and pathfinding for WSN and decentralized IoT systems using I-GWO and Ex-GWO algorithms

Amir Seyyedabbasi [a], Farzad Kiani [a], Tofigh Allahviranloo [a], Unai Fernandez-Gamiz [b], Samad Noeiaghdam [c,d,*]

[a] *Software Engineering Department, Faculty of Engineering and Natural Science, Istinye University, Istanbul, Turkey*
[b] *Nuclear Engineering and Fluid Mechanics Department, University of the Basque Country UPV/EHU, Nieves Cano 12, 01006 Vitoria-Gasteiz, Spain*
[c] *Industrial Mathematics Laboratory, Baikal School of BRICS, Irkutsk National Research Technical University, Irkutsk 664074, Russia*
[d] *Department of Applied Mathematics and Programming, South Ural State University, Lenin prospect 76, Chelyabinsk 454080, Russia*

Received 4 May 2022; revised 13 July 2022; accepted 7 August 2022
Available online 19 August 2022

KEYWORDS

WSN;
DIoT;
Pathfinding;
Metaheuristic algorithm;
Swarm intelligence

Abstract Efficient resource use is a very important issue in wireless sensor networks and decentralized IoT-based systems. In this context, a smooth pathfinding mechanism can achieve this goal. However, since this problem is a Non-deterministic Polynomial-time (NP-hard) problem type, metaheuristic algorithms can be used. This article proposes two new energy-efficient routing methods based on Incremental Grey Wolf Optimization (I-GWO) and Expanded Grey Wolf Optimization (Ex-GWO) algorithms to find optimal paths. Moreover, in this study, a general architecture has been proposed, making it possible for many different metaheuristic algorithms to work in an adaptive manner as well as these algorithms. In the proposed methods, a new fitness function is defined to determine the next hop based on some parameters such as residual energy, traffic, distance, buffer size and hop size. These parameters are important measurements in subsequent node selections. The main purpose of these methods is to minimize traffic, improve fault tolerance in related systems, and increase reliability and lifetime. The two metaheuristic algorithms mentioned above are used to find the best values for these parameters. The suggested methods find the best path of any length for the path between any source and destination node. In this study, no ready dataset was used, and the established network and system were run in the simulation environment. As a result, the optimal path has been discovered in terms of the minimum cost of the best paths obtained by the proposed methods. These methods can be very useful in decentralized peer-to-

* Corresponding author at: Industrial Mathematics Laboratory, Baikal School of BRICS, Irkutsk National Research Technical University, Irkutsk 664074, Russia.
E-mail addresses: amir.seyyedabbasi@istinye.edu.tr (A. Seyyedabbasi), farzad.kiani@istinye.edu.tr (F. Kiani), tofigh.allahviranloo@istinye.edu.tr (T. Allahviranloo), unai.fernandez@ehu.eus (U. Fernandez-Gamiz), snoei@istu.edu, noiagdams@susu.ru (S. Noeiaghdam).
Peer review under responsibility of Faculty of Engineering, Alexandria University.

https://doi.org/10.1016/j.aej.2022.08.009

peer and distributed systems. The metrics for performance evaluation and comparisons are i) network lifetime, ii) the alive node ratio in the network, iii) the packet delivery ratio and lost data packets, iv) routing overhead, v) throughput, and vi) convergence behavior. According to the results, the proposed methods generally choose the most suitable and efficient ways with minimum cost. These methods are compared with Genetic Algorithm Based Routing (GAR), Artificial Bee Colony Based routing (ABCbased), Multi-Agent Protocol based on Ant Colony Optimization (MAP-ACO), and Wireless Sensor Networks based on Grey Wolf optimizer. (GWO-WSN) algorithms. The simulation results show that the proposed methods outperform the others.

1. Introduction

Wireless Sensor Networks (WSNs) are one of the subcategories of ad-hoc networks and consist of many distributed sensor nodes. These nodes can also be used in systems comprising of the Internet of Things (IoT). One of the advantages of these nodes is their ease of assembly in difficult environments. WSNs can be used in a vast variety of application areas, such as traffic monitoring [1,2], agriculture [3,4], automobiles [5], health monitoring [6], etc. There are also many application areas in IoT, such as the Internet of Drones [7,8], Internet of Food [9], Internet of Medical-Things [10], Industrial IoT (IIoT) [11], and autonomous vehicles [12]. Moreover, WSNs and IoT systems can also collaborate as a single system [13–17]. It can be used widely, especially in decentralized IoT architectures [18,19].

In scenarios, where there are internet availability issues, or when low cost is desirable, problems can occur in an IoT system with classical centralized architectures. Furthermore, in this architecture, a large part of the load falls on the server-side cloud system. Some methods are proposed for this, such as fog/edge compute nodes. Another recommendation to troubleshoot this architecture is blockchain technology. A decentralized architectural design can be more efficient and is used extensively in application areas. Therefore, decentralized distributed architectures can be employed as a solution. We define these systems as Decentralized IoT (DIoT) systems. WSNs are generally designed in a decentralized form. Therefore, DIoT and WSN are similar in architectural aspects. One important issue in these structures is finding a suitable, optimal, and efficient path for data communication between nodes. To achieve this goal, this study proposes two efficient methods that are inspired by metaheuristic algorithms. The proposed methods can, both, find the optimal paths with efficient resource usage, and provide features such as scalability and fault tolerance. A sample system with the hybrid architecture of DIoT and WSN technologies is shown in Fig. 1.

Each sensor node or IoT device can send its data packets to Base Station (BS) via single-hop or multi-hop model. In a decentralized distributed structure, multi-hop methods are frequently employed. The BS collects all data packets and transmits them to a server (possibly a cloud) for data analysis and end-user access [20]. As sensor nodes work in collaboration, it is necessary to have an efficient data transfer method. These nodes suffer from a limited power battery, bandwidth, computational capacity, and memory space. Therefore, performing the complex computations, in each sensor node, is a challenging task. Furthermore, recharging is mostly impossible due to physical constraints, such as the location of the nodes. At the same time, changing the batteries is not possible, as these sensors use one-time batteries. The main issue in WSN and IoT systems is increasing the network lifetime. It is worth noting that resource management and network topology has an important role to play in network lifetime and availability [21]. Since network lifetime deals directly with sensor nodes' remaining battery level, energy consumption is a vital factor in these systems, making energy-one of the most important resources. However, while focusing on this goal, it is also essential to efficiently consume the other necessary resources. Therefore, the methods proposed uses the resources in a balanced manner. Unsurprisingly, efficient resource consumption, such as that of energy, increases the life of the network and, as such, the system [22,23].

In WSN and DIoT systems, one of the most important challenges is efficient resource consumption such as energy [24,25]. Techniques to find the optimal paths, in an energy-efficient manner, are of vital importance. To tackle this problem, numerous multi-purpose routing solutions have been introduced in the literature but finding and proposing a general routing technique that preserves the integrity, connectivity, and inclusiveness of the network is a very costly and complicated process. In addition, finding the most efficient route among many possible paths, in a wide and complex network demands further processing. Moreover, it is not easy to find appropriate, effective coefficients for the relevant routing parameters. In addition, analytical solutions to such problems are difficult to find. In fact, these problems are categorized under Non-deterministic Polynomial-time (NP-hard) problems [26–28]. Therefore, it is fitting to use metaheuristic algorithms to solve it. However, when these algorithms are implemented in the entire routing process, they tend to cause additional overhead in the system and result in inefficient usage of some of the system's resources.

In this study, a generic system architecture is proposed, and this architecture can easily perform routing without incurring any additional cost, integrating with many different metaheuristic algorithms. Since the proposed model is comprehensive, it will be able to work well by including various algorithms for many purposes. In this study, we discussed our performance metrics as follows. i) network lifetime, ii) the alive node ratio in the network, iii) the packet delivery ratio and lost data packets, iv) routing overhead, v) throughput, and vi) convergence behavior.

This paper proposes two new energy-efficient methods based on the Incremental Grey Wolf Optimization (I-GWO) and Expanded Grey Wolf Optimization (Ex-GWO) algorithms

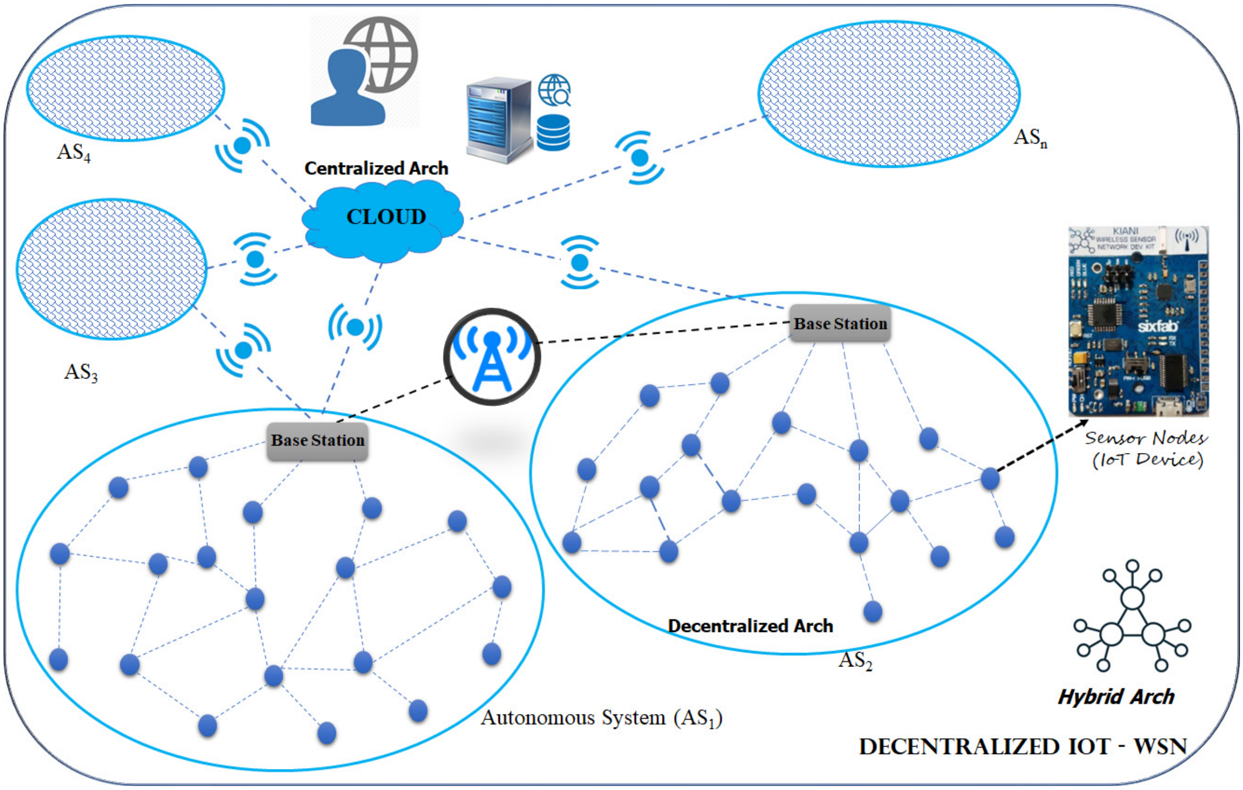

Fig. 1 Sample architecture for DIoT and WSN [19].

to help find optimal paths in DIoT and WSN systems. In the GWO algorithm [29], swarming is controlled by the leader of the group, which helps to get the optimum solution for a defined problem. It can outperform other metaheuristic algorithms thanks to its hierarchy group working mechanism and balanced transitions between exploration and exploitation phases. These wolves can exhibit a successful mechanism because they have an extremely dominant hierarchy. In addition, this algorithm does not require additional cost in finding the optimal solutions in line with the simple working mechanism and parameters. In other words, the GWO algorithm works simply with a small number of parameters, preserving the random principle. Thanks to these features, it has suitable behavior in the exploration and exploitation phases which are effective in finding the optimal solution. Additionally, it only requires one vector of position, which decreases the memory demand. On the other hand, other metaheuristic methods suffer from computational overload and time inefficiency in their approach to the optimum answer. Therefore, thanks to the characteristics of the GWO algorithm, it can be used to find solutions to different complex and real problems. In this regard, the GWO algorithm may be more likely to be successful than other metaheuristic methods in this type of problem on various parameters due to its working mechanism. Therefore, the GWO variants may be more likely to be successful than other metaheuristic methods in this type of problem on various parameters due to its working mechanism.

The I-GWO finds solutions much more quickly, owing to its exploitation feature and fast convergence rate in non-complex environments, whereas Ex-GWO, due to its structure, is deemed successful in complex and large-scale systems. Hence, an appropriate choice can be made for different needs and systems. In this context, the routing methods proposed in this paper suggest the most appropriate model for various networks using these two algorithms. In these algorithms, swarming is controlled by the leader of the group, which helps to get the optimum solution for a defined problem. As a result, these algorithms are useful in decreasing network complexity and increasing the efficiency of resources used in pathfinding. Besides, these algorithms are used to present low-cost paths among the various probable paths. The proposed pathfinding methods, which use the metaheuristic algorithms, are named energy-efficient routing based on I-GWO ($EERI_{-GWO}$) and energy-efficient routing based on Ex-GWO ($EER_{Ex\text{-}GWO}$). The proposed methods try to find paths that are most suitable and most efficient with minimum costs. The other features and contributions of the proposed methods are:

1) A generic system architecture is proposed that combines the metaheuristic and network model. Due to this, the architecture is adaptable in various systems and for numerous purposes. Furthermore, many metaheuristic algorithms can be readily applied in these systems.
2) In order to increase the pathfinding efficiency, metaheuristic algorithms are used to discover the most appropriate coefficients for each parameter of the defined fitness function.
3) A novel and comprehensive fitness function is defined with an emphasis on balancing trade-offs between important parameters. This function concerns with five

parameters (distance between nodes, BS-hop, valid-traffic, energy consumption, and buffer capacity), and a tradeoff between related parameters.
4) Finding the best routes between the nodes means that less energy is consumed in the network. This results in increased network resilience and lifetime.
5) The global knowledge and processing of the network are performed at the BS, which have abundant resources.

The rest of this paper is organized as follows. In the next section, research that deals with finding a route using metaheuristic methods, present in literature, is described. In the third section, the proposed metaheuristic algorithms are briefly explained. In the fourth section, the proposed methods are detailed. In the fifth section, simulation results and analyses are provided. Conclusion and future works are given in the last section.

2. Related works

In general, metaheuristic algorithms can arrive at optimal solutions for real-world problems at a low cost. In the literature, there are several widely used classifications of metaheuristic-based algorithms [30]: nature-inspired vs non-nature-inspired algorithms, population-based vs single point search algorithms, dynamic vs static objective functions, single vs various neighborhood structures, and memory-less vs memory-independent algorithms [30]. One of the most popular discussions and classifications is the population-based and single-point search category. The improvement of a single-based solution is achieved by iterations, while the optimization of a population-based solution is achieved through a set of solutions. Another important area is nature-and non-nature-inspired classification. Recent research shows that nature-inspired algorithms are a trend and perform quite well at solving a wide variety of problems. Methods in this category are defined into four main categories [29]. They are evolution-based, Swarm-Intelligence (SI)-based, physics-based, and human-based [31,32] approaches.

This study focuses on the problem of finding the best route. Studies in lately years prefer SI methods because they generally outperform other methods in solving problems, particularly in pathfinding problems. SI methods are generally nature-inspired and are based on a herd or collective social behavior and community mindset. There are many studies in this category in the literature [33–36] showcasing that SI-based methods can solve complex problems more efficiently. These algorithms consist of a group of simple particles and homogeneous members that interact with each other as well as their environment. Their agents try to find the best solutions that cooperate in the local search area and benefit from the collective effort of all the agents involved. In this study, the use of SI methods in WSN and IoT to find optimal paths is discussed [37–39].

Ant Colony Optimization (ACO) is one of the most frequent metaheuristic algorithms that is used in the systems discussed. Authors in [40] proposed a routing method for a distributed multi hop-based system using the ACO algorithm for reliable data communication. The next hops on the path are based on sensor nodes with high energy levels. However, it is not considered very successful in energy efficiency because it does not work in a fair and balanced manner. The main reason for this is that the fitness function used does not use sufficient parameters. Researchers in [41] investigated a new pheromone update mechanism in the ACO algorithm and used it to achieve energy efficiency of WSNs. The authors discuss two energy measures. In next node selection for routing, sensor nodes closer to the target are more likely to be selected. They also use four control parameters in the probabilistic decision function. Since it does not use memory efficiently, it cannot be very successful in showing efficient performance in the general analysis. Authors in [42] proposed a new routing algorithm based on ACO algorithm to achieve balanced energy consumption on each network sensor node beside the choice of the path with minimal cost. In their work, called IEMACO, they make route discovery based on a number of factors: the convergence speed of the routing algorithm, the probability of transition, and the remaining life of the nodes. Position and speed information predicts the remaining lifetime of the link. The most obvious shortcoming of this study is the usage of the memory method. In [43], the authors have proposed a dynamic energy threshold strategy different from the multipath approaches, so-called ACOHCM. It has some advantages such as network topology, searching the optimal path, and network load balancing. In the ACOHCM, initially hop counting mechanism is applied. The hop count for the sink (BS) is 0. The number of hops of other nodes is incremented by one depending on their neighbors. When the topology of the network changes, the hop counting mechanism is run again. So, the hop counts should be updated at different time intervals. Finally, an energy threshold strategy is used that is applied to each node. The authors of [44] proposed a dynamic decision-making system based on ACO algorithm for connected cars in IoT systems. They used artificial ants to control the dynamics of connected vehicles in traffic flow and for autonomous calculations. An ant colony optimization-based routing protocol for multi-agents is presented in this paper that manages network resources effectively in real-time [19]. In addition to finding the next destination of ants, the proposed method is also used to manage pheromone updates and evaporation rates. Several key parameters are taken into account when determining the next destination under various conditions, including energy remaining, buffer size, traffic rate, and distance. In terms of network lifetime and energy consumption, simulation results of the proposed method have remarkable performance. An ant colony optimization-based routing protocol for multi-agents is presented in this paper that manages network resources effectively in real-time [45]. In addition to finding the next destination of ants, the proposed method is also used to manage pheromone updates and evaporation rates. Several key parameters are taken into account when determining the next destination under various conditions, including energy remaining, buffer size, traffic rate, and distance. In terms of network lifetime and energy consumption, simulation results of the proposed method have remarkable performance.

Apart from ACO, the Genetic Algorithm (GA) is another technique also recommended in such systems. In [46], it was proposed to combine simulated annealing with genetic algorithms in order to achieve optimal performance. There has been a comparison of the observed results in terms of the average residual energy, the network lifespan, and the packet transport between the BS and sink, with that of a GA-based approach. Gupta et al [47] proposed an energy-efficient algorithm to minimize the energy consumption in each round based on GA. The proposed method attempts to reduce the total distance traveled by data in the system. In this study, a Directed Acyclic Graph (DAG) model was used and the chromosome representation, as well as a crossover method, were proposed. In their strategy, they also emphasized the minimization of the total path length. This study, which is ambitious in terms of energy efficiency, is used in comparison with the proposed methods in our study. IoT has been added to Clustered-Based Routing (CBR) for Information-Centric WSNs (ICWSNs) in a protocol known as CBR-ICWSN, which enables CBR for these networks [48]. There are two phases to this paper, which include the choice of a Cluster Head (CH) and the determination of the optimal route. Thus, by employing a Black Widow Optimization (BWO) method in order to choose an optimal set of CHs, an optimal set of CHs is selected. It is interesting to note that the authors in this paper used a different algorithm to find the optimal route. CBR-ICWSN is a routing protocol that is based on Oppositional ABC (OABC) and can be used to select routing routes more efficiently.

The artificial Bee Colony (ABC) algorithm is yet another metaheuristic method used in such systems. Authors in [49] have proposed a new clustering routing method based on an ABC algorithm for cluster formation. Their main goal is to reduce energy consumption and exploit low-power clusters. They are concerned about the trade-off between energy consumption and the quality of the communication link within clusters. Authors in [50] propose a method based on the Grey Wolf Optimizer (GWO) algorithm to solve the energy problem in WSNs. They attempt to handle the problem of finding the correct position of unknown nodes in the network. Based on their results, their GWO-based method is better than Partial Swarm Optimization (PSO) and Modified Bat Algorithm (MBA) algorithms in the convergence and success rate. In [51] researchers have proposed a new routing algorithm in a hierarchical structure using the GWO algorithm. It avoids the energy hole by balancing the load on the nodes nearer to BS and cluster head nodes. The new fitness function, proposed in their work, takes into account the total distance and the total number of hops. This fitness function is solely used to help the wolves. One shortcoming of this study is that it does not focus sufficiently enough on the effective parameter. Without taking into account necessary and sufficient parameters, the results obtained from the fitness function can be, at best, of very limited use in real-world cases. In contrast, the fitness function proposed in our study is general and multi-purpose and can also be easily adapted to many metaheuristic algorithms. It is worth mentioning that the architecture proposed is the leading reason for this adaptability.

In another study, the authors proposed a *meta*-heuristic artificial intelligence approach based on grey wolf social behavior to minimize the energy consumption of WSNs from the livestock industry [52]. In order to determine an algorithm's performance, energy level, grid size, transmission range, and direction of transmission were used as factors. A metaheuristic-driven, energy-aware routing scheme (IMD-EACBR) is proposed in [53]. The IMD-EACBR model aims for maximum energy usage and lifetime. IMD-EACBR employs an improved Archimedes optimization algorithm-based clustering (IAOAC) technique to cluster head selection. Furthermore, the TLBO-MHR technique is applied for optimum route selection using teaching–learning-based optimization (TLBO). Simulated outcomes reveal improvements in dead node proportions, network lifespan, energy consumption, packet delivery ratio (PDR), and latency. A novel clustering and routing method is presented in this paper in an effort to enhance system efficiency [54]. In order to optimize it, it relies mostly on genetic algorithms as well as equilibrium optimization. Using genetic algorithms, a first phase is carried out that clusters the sensor nodes based on their features. As a result, the best cluster heads are selected to improve system stability. The purpose of this work is to reduce the energy consumption of WSN networks by improving the clustering algorithm and the equilibrium optimization algorithm used for selecting the optimal path between cluster heads and base stations. Consequently, the proposed method has been obtained to be the most energy-efficient, have a longer network lifespan, and deliver more packages than other methods. This study aims to develop an energy-efficient cluster routing protocol that can be applied to wireless sensor networks [55]. In the first step of the cluster head selection process, we used the Honey Badger Algorithm to select cluster heads. In order to find the optimal cluster head among all sensors, the Honey Badger Algorithm is used. This algorithm takes into account factors including distance to the base station, residual energy, distance to its neighbors, node degree, and centrality. It then selects the optimal cluster head. A fuzzy Firebug Swarm Optimization algorithm is used to perform the routing between the cluster heads and the base stations. This method offers a reduction in the amount of end-to-end delay, an increase in the number of packets that are delivered, a higher throughput, and a reduction in the number of packets lost, which are all factors that affect how much energy is consumed by the network.

In another study in the literature, a hybrid optimization algorithm is proposed to propose a new energy-aware CH selection framework in WSNs through hierarchical routing [56]. As well as energy and distance, delay, and Quality of Service (QoS) are considered when selecting the CH. It is proposed to develop a hybrid algorithm that combines the principles of Sea Lion Optimization (SLnO) and Particle Swarm Optimization (PSO) to select the optimal CH. The performance of the adopted method is compared with other traditional models using a variety of metrics. Compared to other conventional methods, the proposed algorithm has higher normalized energy. In this paper, the chaotic fuzzy grasshopper is applied to optimizing routing on the Internet of Things, focusing in particular on the sleep-wake schedules of nodes, which are an essential part of the routing [57]. During the evaluation of the efficiency of the proposed method, the following three criteria were utilized: the remaining energy, the network life, and the coverage rate of the network. It has been determined that the results are based on two different scenarios that have been analyzed. Consequently, the proposed method performs better than the base method in all scenarios and is more effective for all criteria of comparison than the base method.

The use of metaheuristic methods has become very popular in IoT and WSN systems, especially in recent years. In this paper, two methods to find optimal paths in DIoT and WSN applications are provided using two metaheuristic algorithms (I-GWO and Ex-GWO). These methods can be applied in both DIoT and WSNs.

3. I-GWO and Ex-GWO algorithms

This section briefly describes two metaheuristic algorithms used in the methods proposed in this paper. The Grey wolf optimizer (GWO) algorithm is inspired by grey wolves in their natural habitat [29]. Alpha (α), beta (β), delta (δ), and omega (ω) are the four types of wolves found in a pack. These wolves have different responsibilities in the pack. Alpha Wolf is the leader of the pack. Beta wolves are the co-leaders of the alpha wolf. The third level of hierarchy in the pack is that of delta wolves. The remaining wolves which are not part of the upper level of the hierarchy are omega wolves. Encircling, hunting, and attacking are the three main attributes of the wolves.

Incremental Grey Wolf Optimizer (I-GWO) algorithm, used in the first pathfinding method of this paper, is an upgraded version of the GWO [36]. In the I-GWO algorithm, the leader encircles the prey (Eq. (1)), hunts it and finally (Eq. (2)), attacks the prey based on the $\overrightarrow{A}$ value. If $|A|$less than1, a wolf is attacking its prey, otherwise, it's busy finding other prey. The second wolf on the pack follows the leaders' position and updates its own position to attack the prey. Generally, the n^{th} wolf in the pack updates its own position based on the n-1 wolf before it (Eq. (3)). Eq. (4), 5, and 6 are the control mechanisms to avoid trapping in local optima and to balance movements between exploration and exploitation phases.

$$\overrightarrow{D_\alpha} = \left|\overrightarrow{C_\alpha}\cdot\overrightarrow{X_\alpha} - \overrightarrow{X}\right| \tag{1}$$

$$\overrightarrow{X_1} = \overrightarrow{X}_\alpha - \overrightarrow{A_1}\cdot\overrightarrow{D_\alpha} \tag{2}$$

$$\overrightarrow{X_n}(t+1) = \frac{1}{n-1}\sum_{i=1}^{n-1} X_i(t)\,;\ \mathrm{n} = 2,\,3,\cdots\,\mathrm{m} \tag{3}$$

$$\overrightarrow{A} = 2\overrightarrow{a}\cdot\overrightarrow{r_1} - \overrightarrow{a} \tag{4}$$

$$\overrightarrow{C} = 2\cdot\overrightarrow{r_2} \tag{5}$$

$$\overrightarrow{a} = 2\left(1 - \frac{t^2}{T^2}\right) \tag{6}$$

Additionally, in both I-GWO and Ex-GWO, $\overrightarrow{a}$ is linearly decreased from 2 to 0 over the course of iterations, and is obtained using Eq. (6) and (12), respectively. The effect of $\overrightarrow{a}$ is on the range of motion, directing the algorithm in finding the solution and is used to get closer to the solution range. Random vectors $\overrightarrow{r_1}$ and $\overrightarrow{r_2}$ lie in the range [0, 1]. $\overrightarrow{A}$, and $\overrightarrow{C}$ are coefficient vectors that lead to encircling the prey [29,33–35]. These parameters control the tradeoff between exploration and exploitation phases. Due to this, wolves do not always go in the same direction. In all variants, whenever $\overrightarrow{A}$ is less than 1, the wolves in the pack attack to hunt, otherwise, they try to find the prey. $\overrightarrow{X}$ is the position vector of the prey, whereas $\overrightarrow{X_i}$ is the position vector of the grey wolf, and $\overrightarrow{D_i}$ is a vector that depends on the location of the target. Where $i \in \{\alpha,\,\beta,\,\delta\}$. Moreover, t is current iteration and T is maximum iteration numbers.

The other metaheuristic algorithm used in this paper is the Expanded Grey Wolf Optimizer (Ex-GWO) [36] algorithm. The hunting mechanism of the Ex-GWO is uses a technique dissimilar to one used in I-GWO and GWO algorithms. Encircling of the prey is performed using the first wolf in the pack (Eq. (7)). The fourth wolf in the pack updates its position based on the first three wolves before it. Generally, the n^{th} wolf in the pack updates its own position based on the first wolf in the pack as well as the wolves before it (Eq. (8) and (9)). In Ex-GWO the attacking mechanism ensures that the prey does not escape. The coefficients a, A, and C are calculated using Eq. (10), 11, and 12.

$$\begin{gathered}\overrightarrow{D_\alpha} = \left|\overrightarrow{C_\alpha}\cdot\overrightarrow{X_\alpha} - \overrightarrow{X}\right| \\ \overrightarrow{D_\beta} = \left|\overrightarrow{C_\beta}\cdot\overrightarrow{X_\beta} - \overrightarrow{X}\right| \\ \overrightarrow{D_\delta} = \left|\overrightarrow{C_\delta}\cdot\overrightarrow{X_\delta} - \overrightarrow{X}\right|\end{gathered} \tag{7}$$

$$\begin{gathered}\overrightarrow{X_1} = \overrightarrow{X}_\alpha - \overrightarrow{A_1}\cdot\overrightarrow{D_\alpha} \\ \overrightarrow{X_2} = \overrightarrow{X}_\beta - \overrightarrow{A_2}\cdot\overrightarrow{D_\beta} \\ \overrightarrow{X_3} = \overrightarrow{X}_\delta - \overrightarrow{A_3}\cdot\overrightarrow{D_\delta}\end{gathered} \tag{8}$$

$$\overrightarrow{X_n}(t+1) = \frac{1}{n-1}\sum_{i=1}^{n-1} X_i(t)\,;\ \mathrm{n} = 4,\,5,\cdots\,\mathrm{m} \tag{9}$$

$$\overrightarrow{A} = 2\overrightarrow{a}\cdot\overrightarrow{r_1} - \overrightarrow{a} \tag{10}$$

$$\overrightarrow{C} = 2\cdot\overrightarrow{r_2} \tag{11}$$

$$\overrightarrow{a} = 2\left(1 - \frac{t}{T}\right) \tag{12}$$

I-GWO algorithm is based on the leader wolfs' behavior. Other wolves in the pack update their own position based on all the wolves selected afore themselves. In the Ex-GWO algorithm, the n^{th} wolf updates its own position relevant to the prey according to their immediate successor and the first three wolves. In [36], it is proved that the performance of I-GWO and Ex-GWO algorithms is better than GWO. On the other hand, I-GWO tries to find solutions much quickly due to its exploitation feature and its fast convergence rate, and Ex-GWO, owing to its structure, is likely to be successful in complex and large-scale systems.

4. Proposed pathfinding methods

The used metaheuristic algorithms are a natural match for the problem suit and exhibit a balanced behavior, as such, they have been used in this paper as the problem-solving methodology. As explained in the literature section, many metaheuristic-based algorithms have been used for similar systems. It is known that metaheuristic-based methods do not guarantee to find optimal solutions, but they try to find the solutions close to the optima, providing more efficient execution time and CPU power consumption in time and space complexities. Each of the proposed metaheuristic methods in the literature has its advantages along with its shortcomings. This study focuses on broader parameters in proposing comprehensive and accurate methods to be used in WSN and DIoT. Accordingly, a new fitness function has been defined. The defined fitness function is used to calculate the cost of each path in the network and includes residual energy, traffic status, buffer rate, BS-hop, and neighbor list of each node as formulated in Eq. (13). BS-hop indicates the hop counts of each node to BS. In this study, BS is assumed to be the destination node. The BS node does not look only at the distance or number of hops of each node relative to itself to find the most suitable path (between each node and itself) but also focuses on other effective parameters that are defined in the new fitness function. It takes into account the dynamic resources of the nodes in the system and the variable parameters of the network. For this purpose, a new fitness function is defined. The paths between the source and destination nodes are selected according to hop values and passed through the fitness function. The sum of the best fitness values for each hop will be the candidate for the best route (Eq.14.). Subsequently, the minimum value among candidates is chosen as the best path between the two relevant nodes for each hop count (Eq.15). At this stage, the lowest-cost path is chosen for each hop count (step 2). Step 3 selects the best path with the minimum cost among all hop sizes (Eq.16). At the same time, as mentioned before, one of the most important issues discussed in these systems is energy saving. In this section, the new energy-efficient routing methods based on the two metaheuristic algorithms, $EER_{I\text{-}GWO}$ and $EER_{Ex\text{-}GWO}$, are introduced. They can aid in modeling useful solution models in the pathfinding of wide and complex networks (especially in decentralized architecture). These methods focus on the critical features of sensor nodes in pathfinding. As aforementioned, pathfinding and routing are NP-hard problems in the complex distributed and Peer-to-Peer (P2P) structures such as WSN and DIoT. Therefore, these proposed methods can provide a good solution for finding optimal paths in the entire search space. In short, they find the optimal path from the sets of possible paths in multiple hops.

4.1. Proposed architecture

In this subsection, some definitions and design factors of the proposed methods have been summarized along with the description of the proposed methods. Sensor nodes (IoT devices) are deployed randomly in the network, and different paths are created between any pair of source and destination nodes. The metaheuristic algorithms used in this paper belong to the SI category and are population-based. A general architecture is suggested for the relevant metaheuristic algorithms to work harmoniously with the proposed methods. Thanks to this architecture, the algorithms used can be easily adapted to the relevant system. Furthermore, it should be noted that other metaheuristic methods are also able to easily use such systems. The conceptual schema of the proposed architecture is presented in Fig. 2.

In this architecture, the search space is considered as a matrix where the rows represent the number of search agents, and the columns signify the coefficient numbers. In the simulation of I-GWO and Ex-GWO, the number of search agents is assumed to be equal to the number of grey wolves in the pack. Moreover, the coefficient numbers, which are used as the dimension of the problem, are assumed to be four and their values are obtained using Eq. (13). The fitness function, defined earlier, is used to calculate the cost of each path. In addition, all the coefficients used in the proposed methods are updated at every round of the network based on metaheuristic algorithms. When the number of hops between two nodes is one, they are already single-hop and are direct neighbors, so there is no need to specify a route. The problem arises with multi-hop structures. In these cases, there may be paths of various lengths between the two nodes. There may be intermediate nodes between source and destination when the hop counts are more than one. In the proposed architecture, best route for any hops of paths is found considering Eq. (14) and (15), and the best among them is chosen using Eq. (16). Therefore, the best path between two intermediate nodes is obtained, along with the calculation of the final best path cost between source and destination. The fitness function is given by Eq. (13), which calculates the cost between two nodes i and j.

$$Cost_{i,j} = \left(c_1 d_{i,j}\right) + \left(c_2 H_j\right) + \left(c_2 \frac{\mathrm{ValidTraffic}}{\mathrm{T}_{i,j}}\right) + \left(c_3 \frac{E_{initial}}{E_j} . \frac{\mathrm{Buffer\,Capacity}}{\mathrm{B_j}}\right) \tag{13}$$

Where $d_{i,j}$ is distance between nodes i and j. E_j indicates the residual energy of the node j, and H_j is hops count of node j to BS. $T_{i,j}$ is the traffic status between nodes i and j. B_j indicates the buffer rate of the nodes j. $ValidTraffic$, $BufferCapacity$, and $E_{initial}$ are common variables that are used for each node. The values of these three variables are their maximum values at each node and they are related to node hardware properties. Furthermore, c_1, c_2, c_3, c_4 are three control parameters, with values between 0 and 1 where $c_1 + c_2 + c_3 + c_4 = 1$ and $c_1 < c_2 < c_3 < c_4$. These control parameters are calculated by metaheuristic algorithms (I-GWO and Ex-GWO). The coefficients updates are done at each round of the network. It is worth mentioning that the network rounds term is different with metaheuristic iterations. This difference is described in the subsequent subsection.

4.2. The network rounds and metaheuristic iterations

In the actual application of sensor networks and IoT, the network round and metaheuristic algorithms iteration work separately. In the network considerations, the rounds and iterations should be handled separately. Each round of the network occurs in certain time periods. In the proposed methods, a time interval between each round of the network is considered. If the round and iterations work together, it causes an overload on the network. Metaheuristic algorithms try to find the best solutions. At the same time, there is no specific time to reach the solution, as such, the pathfinding operation is done in the BS. After that, over a period of time, data packet transfer is completed between the target (destination) and the source. The concept of iteration is an expression used widely in metaheuristic methods. Each iteration tries to approach the solution based on the results obtained in the previous iteration. Both the number of iterations and the number of network rounds depend on the system design. In this paper, these parameters are defined and quantified. Furthermore, they are described in the simulation section.

4.3. Pathfinding mechanism

In the proposed methods, the metaheuristic algorithms used help in finding the best path with a minimum cost between

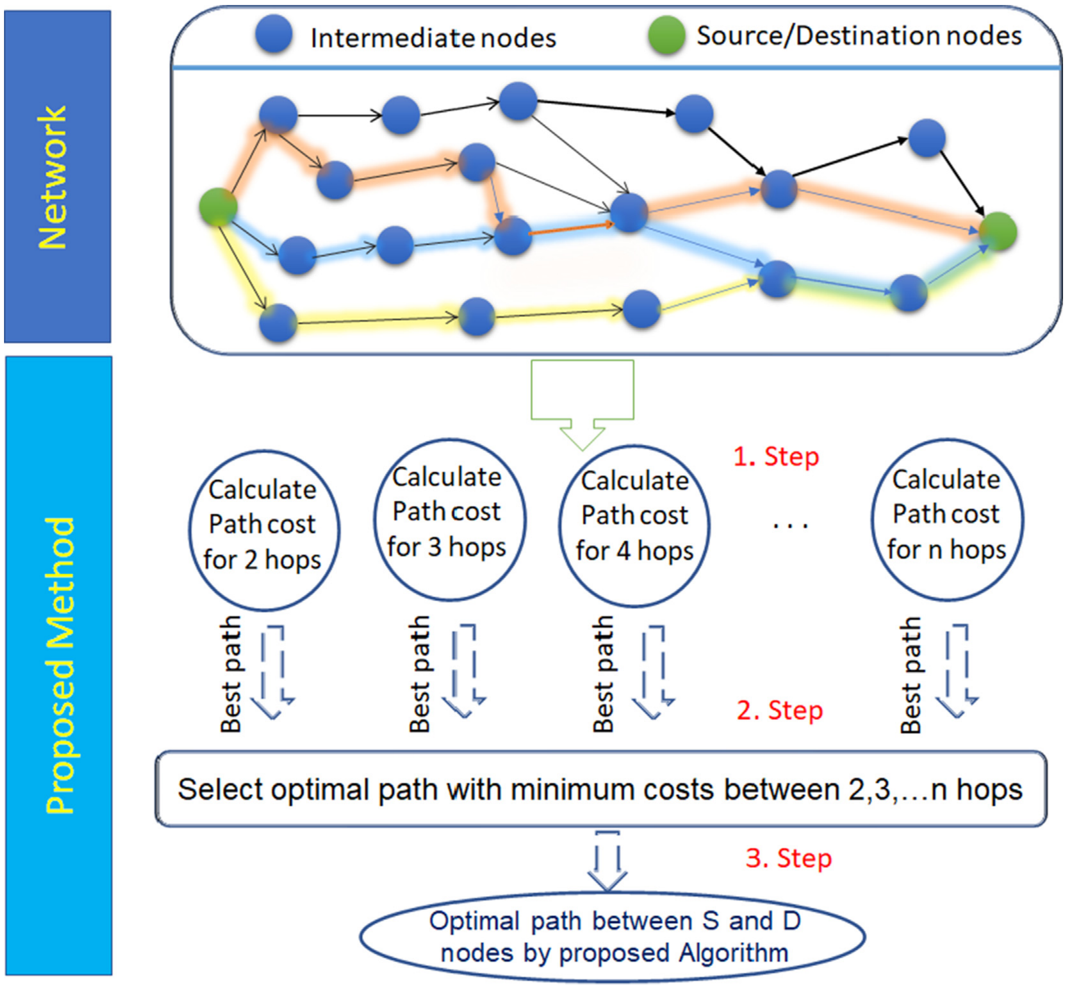

Fig. 2 Conceptual schema of the proposed architecture in finding optimal paths.

the source and destination (BS) node. There are paths with various numbers of hops as seen in the first step of Fig. 2. In this step, the cost of the path for all intermediate nodes between the source and destination nodes is calculated utilizing Eq. (13). The most optimized coefficient values of each parameter in this equation are obtained from metaheuristic algorithms, as described in the previous subsection. In the second step, the algorithm calculates costs for the candidate path applying Eq. (14) and then selects the path with the minimum cost as the best path for each hop count using Eq. (15). This procedure is applied for all hop counts. Indeed, this process is performed for all paths with different sizes. Naturally, the best candidate path is chosen among the paths of the same length. In the end, as indicated in step 3 of Fig. 2, the algorithm selects the minimum cost path from the obtained candidate paths, as an optimal solution between these nodes (source and destination) based on Eq. (16). In other words, the optimal path is chosen among the best paths of different lengths. The proposed methods attempt to select the paths that are the most convenient and efficient routes with the minimum costs.

$$Cost_{S,D}^{Condidate} = \sum_{i=1}^{j=n} cost_{i,j} \tag{14}$$

$$Cost_{S,D}^{h} = Min(Cost_{S,D}^{Condidate}) \quad \forall h \epsilon HopCount \tag{15}$$

$$Cost_{S,D} = Min\left(Cost_{S,D}^{h}\right) \tag{16}$$

Where $Cost_{S,D}^{Condidate}$ is shows the total cost between nodes i and j. This process is calculated separately for each hop count. The shortest path found for each hop is considered $Cost_{S,D}^{h}$. After finding the shortest path for all hops, one shortest path among all is accepted as the final answer and it is called $Cost_{S,D}$. This process is described in Fig. 3 with a schematic example.

For instance, the hop size in the first round of the network may be different from hop sizes in subsequent rounds. At the same time, the hop sizes in different rounds may vary from each other. In this study, the destination node is assumed to be the BS, and therefore, the costs between each sensor node and BS are calculated. Here, in the path, the packet is also passed just once from each sensor node. In the end, BS chooses the minimum cost path using Eq. (16). Finally, the BS broadcasts selected optimal paths to source nodes. For example, if the candidate path between node 4 and BS is N4, N61, N98, N43, and BS, then, first of all, the cost of tuples (N4, N61), (N61, N98), (N98, N43), and (N43, BS) is calculated from Eq.13. An example of candidate paths with sample costs are represented in Table 1. The sum of each tuple value is calculated through Eq.14, which is the cost of each individual path. After all candidate paths for each hop count have been calcu-

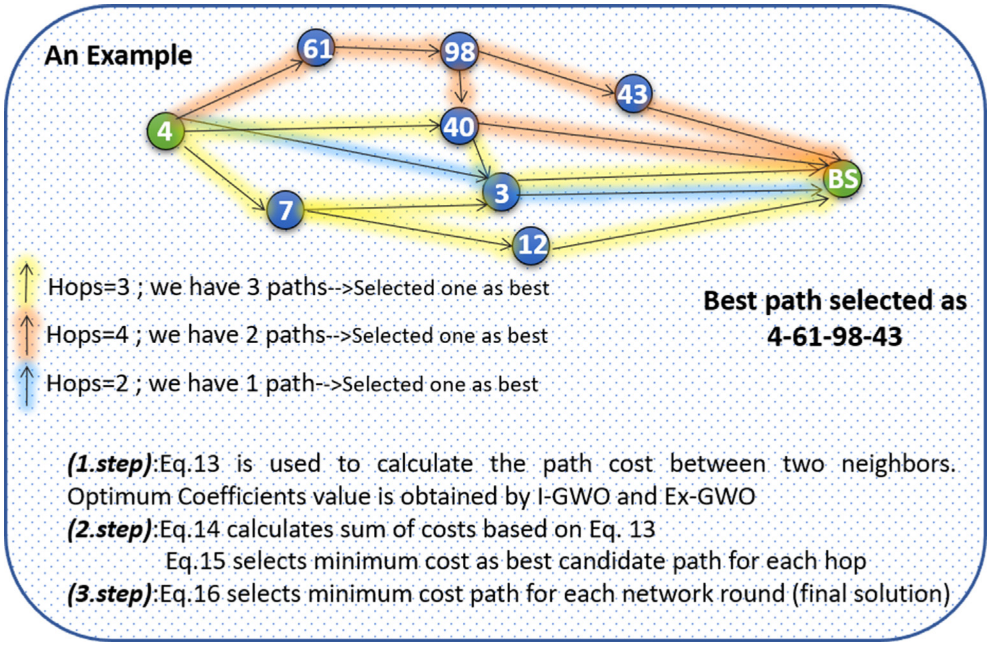

Fig. 3 A working mechanism of proposed method in pathfinding.

Table 1 Template candidate paths cost at the end of each iteration.

Candidate paths	Sample values
Path 1	0.78
Path 2	1.36
⋮	⋮
Path n	n

lated, their minimum is selected as best, by means of Eq.15. In the end, only one of the best paths for all hops is obtained according to Eq.16. This example is schematically represented in Fig. 3.

A hand-shaking method for checking the availability of next-hop is performed. On a default network, each node's information is considered to be recorded in the BS, as outlined in Table 2. This information is obtained by a request data packet that is sent to sensor nodes via BS in the initialization phase of the network. Residual energy, traffic status, buffer size, distance to the BS, and neighbor list are stored in the BS. The decisions in finding the optimum path are made using metaheuristic algorithms by BS, which have unlimited energy sources. Note that balanced behavior is required between these five parameters. The remaining energy level and the remaining buffer size are desired to be high, whereas the network traffic and the distances are desired to be low. The methods proposed and detailed earlier, handle the balancing requirement.

As mentioned, the optimal paths are obtained in the BS. For this, as previously emphasized, I-GWO and Ex-GWO methods are used to find the optimized coefficients of the defined parameters. The BS node has a table regarding nodes' information. In this table, some basic information such as residual energy, traffic status, buffer rate, BS-hop and neighbor list of each node is stored. Each sensor node also holds a table, which is called the routing table, that includes a neighbors list, distance to neighbors, distance to BS, and BS-hop. The relevant routing table is presented in Table 3.

4.4. Definition of data packet frames

Data packets are used for communication between system nodes. These packets have various suitable formats that are defined in these devices. However, they can be customized to optimize the use of resources. Efficient system resource utilization can be ensured with the definition of the appropriate packet template and dynamic structure, according to the system needs. The use of custom data packets is also helpful in finding paths. In this study, two general types of data packet frames are defined. As mentioned before, in the initialization phase of the network, BS broadcasts a message to request global information about the sensor devices. This information is obtained by a request data packet broadcasted to nodes via BS, as depicted in Fig. 4(a). In response, the sensor nodes transmit the relevant information (residual energy, traffic situation, buffer rate, BS-hop, and neighbor list) to BS. The formats of the sensor node's response packet are also shown in Fig. 4(b) along with different fields defined in these packets.

The *TTL* field is intended to prevent the occupation of network traffic. A deadline value is defined for each packet. Each node reduces the *TTL* value by one for each packet received. The initial value of this field varies depending on the type of application. Owing to the *source* and *destination* addresses filed, each node can be applied to multiple sources and destination scenarios at the same time as parallel and concurrent models because it knows which nodes are source and which is destination. Due to the nature of the proposed methods, parallel and concurrent models are naturally supported. This feature is very important in the proposed methods, and it offers the opportunity to work in many parallel and concurrent application areas.

Table 2 Stored information of each node in the sink/BS: as a sample.

Node ID	Energy	Traffic	Buffer	BS-hop	Neighbors list
i	0.5	1	1	4	j, k
j	0.4	0.8	1	3	i, k

Table 3 Routing table for each node (sample node: i).

Neighbor lists	Distance	Distance to BS	BS-hop
i	–	18	4
j	15	12	3

This global information helps BS in finding the optimal path from the source to the destination node. Related processes are performed by the BS node with an unlimited energy supply. The BS finds the minimum cost of the path, so the sensor node is also aware of the routing path. In this way, BS broadcasts a packet containing the source and destination ID, as well as all nodes on the selected path as illustrated in Fig. 5. The *seq* field is used to control and avoid duplicate packets. This field may be ignored, when needed, depending on the definition of the problem. Each sensor node first checks the *seq* frame. Using this field helps the node decide to discard a packet in its entirety if it has received this packet earlier. It guarantees efficient consumption of node resources such as its batteries. Consider the index of nodes (N54, N17, N87) written between Source ID and BS in the *Path* field of Fig. 5. In this example, the optimal path between Source ID and BS passes through N54, N17 and N87 respectively. *Options* field can be custom-defined to meet the needs that may arise depending on the situation in various applications and architectural areas.

In this study, the definition of the five parameters used in response packet frames is explained. They are also used in the relevant fitness function. The *residual energy* of each node is the remaining battery level of the sensor node. This parameter is defined by the Joule metric. *Traffic* status is obtained using the valid traffic divided by traffic between nodes *i* and *j*. The valid traffic is defined in network assumptions of the simulation section. *Buffer size* (ratio) is also calculated by the buffer capacity of each node divided by the node *j*'s current buffer size. *BS-hop* is calculated by each node when getting a broadcasting message based on the distance between the sensor node and BS. Besides, each node should find the *neighbors' list* using handshaking protocol. These parameters depend on the physical and electronic properties of the devices used in the real-world scenario.

The flowchart and the pseudocode of the proposed methods are presented in Algorithm 1 and Fig. 6, respectively. The next section discusses the network model and simulation parameters. Simulation and comparison results are also detailed.

Algorithm 1: The proposed routing algorithms

Input: sensor nodes local information
Output: minimum cost path
while (r < network round)
//Base station broadcasts a message to get local information of sensor nodes (Fig. 4.a)
//Sensor nodes send their local information to base station (Fig. 4.b)
while (h <= max hop count)
// Metaheuristic algorithms (I-GWO and Ex-GWO) initialize the search space for hop count h
while (i < max iteration) *//initialize a, A, C*
calculate x(t + 1) according to Eq.3 for I-GWO and Eq.9 for Ex-GWO *// finding coefficient values*
calculate cost of each path according to Eq.13
i++
end
calculate the path cost for each hop count Eq.14
select the candidate path with minimum cost for each hop count Eq.15
h++
end
select the best path with minimum cost for current round among candidate paths Eq.16
r++
end

4.5. Robustness and fault tolerance feature

The main goal of this study is to increase the availability and lifetime of the network while reducing the traffic rate and increasing fault tolerance as well as packet delivery rate. In general, robustness and fault tolerance are important issues

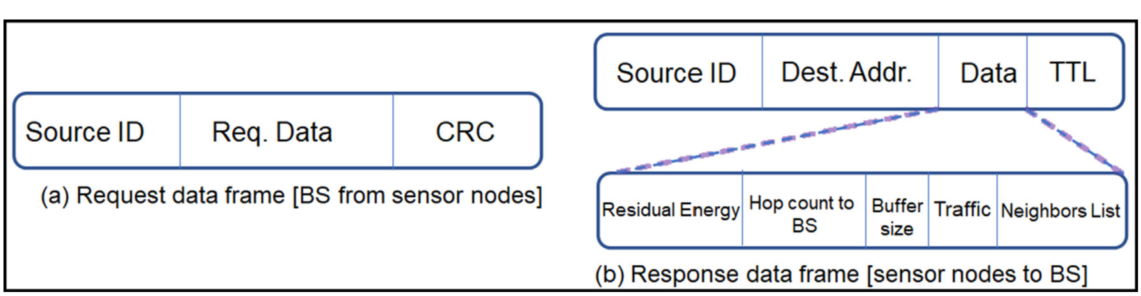

Fig. 4 Data packet frames used to find paths.

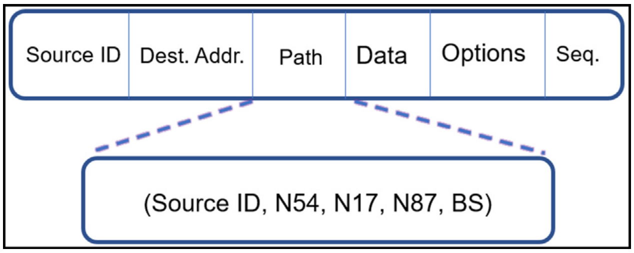

Fig. 5 Data packet frames after pathfinding that is transmitted by BS to source nodes.

in the WSN and IoT systems. Using the proposed method, all the aforementioned objectives have been achieved. Furthermore, an additional measure is required to ensure fault tolerance. This measure is provided with the proposed Eq. (17) and is included in the relevant methods. One of the difficulties encountered in routing algorithms is to withstand the failures that may arise in sensor nodes or networks. For example, when any node on the selected path becomes unavailable (such as battery depletion), data communication in the system should not be interrupted, traffic should slow seamlessly, and similar faults should be tolerated by the method. In the proposed methods, in such cases, an alternative neighbor node comes into play, taking the load of the disabled node and finding a new substitute path. This feature is provided to ensure that the system finds paths with minimum cost. An example of this mechanism is shown in Fig. 7. A node, *i*, requests to send a data packet to another node *j*. When node *j* becomes unavailable, node *i* looks up the routing table it holds and chooses a neighbor (node *k*) with minimum hop size to the BS. Hence, node *i* calculates the cost between itself and *k* and adds the cost in the *Options* frame of the data packet using Eq. (17). This will be used in the round currently in the network. Because, as the network moves to a new tour, a path will be rediscovered from the beginning. Besides when a small number of nodes fail, the average number of nodes and the proportion of nodes that may be connected to the BS remains relatively stable, which implies the network's stability regardless of node failures.

$$Cost_{S,D}^{Condidate} = \sum_{i=1}^{j=n} Cost_{i,j} + \sum_{i=1}^{k=n-j} Cost_{i,k} \tag{17}$$

4.6. Other features

The proposed methods are suitable to be used in parallel and concurrent applications. In these methods, the most appropriate and efficient paths can be found between multiple sources and destination nodes concurrently or in parallel (depending on the needs of the problem and application area). Achieving

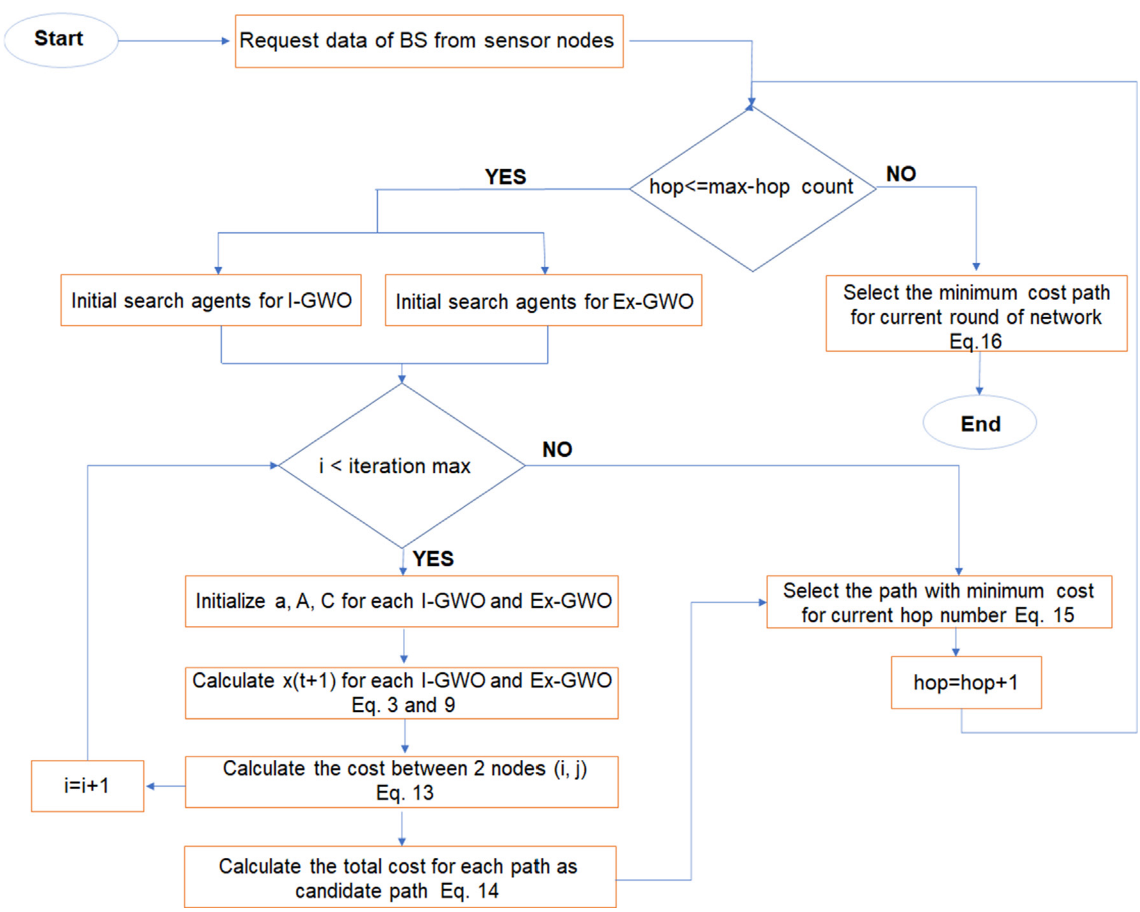

Fig. 6 Flowchart of the proposed methods.

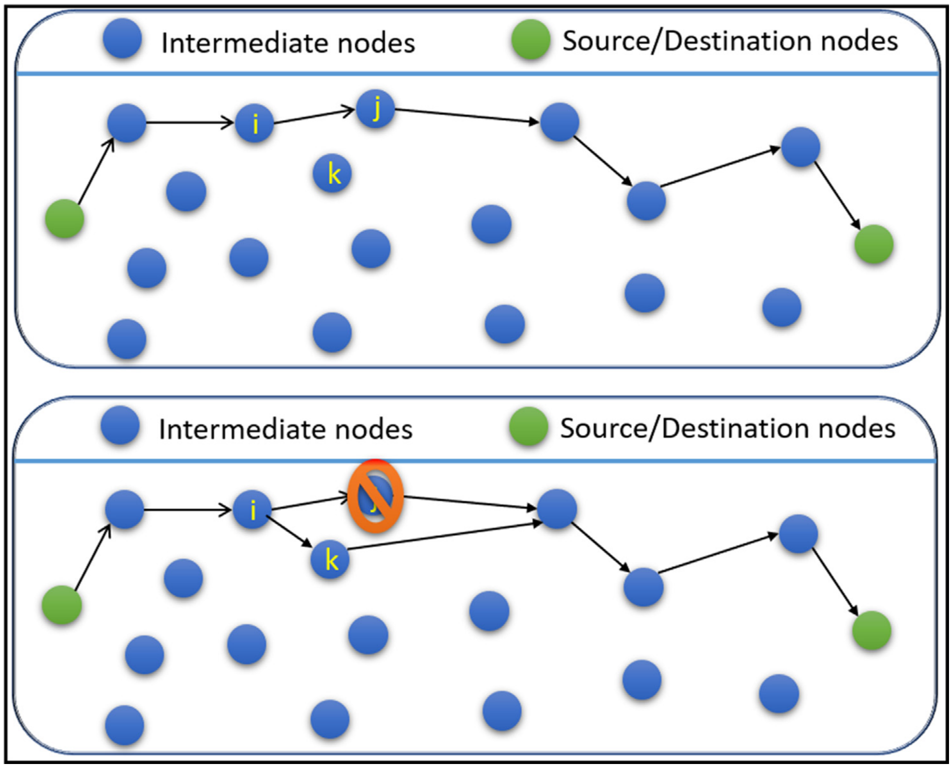

Fig. 7 Fault tolerance in proposed methods.

this goal can become easier with the proposed abstractly comprehensive architecture. It is possible to achieve results rapidly, particularly when the intermediate nodes are not common in different source and destination nodes. Due to this feature, the proposed approach can perform surpassingly in multi-objective and Pareto-based problems. In multi-objective problems, the concept of optimum, which has several objective functions, changes because these problems try to find good compromises (or exchanges) instead of a single solution, as in global optimization. It should be noted that the methods proposed in this paper is conceivably an apt solution for large-scale networks. As the system size expands, it becomes even more compatible and provides good performance when compared to other methods. Due to this feature, systems such as IoT of big data (e.g., smart cities) can offer useful solutions. The computational complexity of both metaheuristic methods proposed is $O(n^2)$. In the hunting mechanism of these algorithms, the I-GWO is faster than the Ex-GWO. I-GWOs' time complexity success in the result is due to its choice of exactly-one wolf as an alpha wolf (best solution). Ex-GWO uses swarm intelligence in the pack to update the wolves' position, and as such is relatively slow.

5. Simulation, comparison and results

This section describes the simulation configuration of the proposed methods implemented using MATLAB on a computer with a Core i7-5500 U 2.4 processor and 8 GB of RAM. In the proposed methods, energy-efficient routing methods have been implemented using two metaheuristic-based algorithms ($EER_{I\text{-}GWO}$ and $EER_{Ex\text{-}GWO}$).

5.1. Simulation settings

The network model used in this method is flat and the nodes are randomly deployed in a $100x100m^2$ area. The proposed methods are compared with MAP-ACO [19], GAR [47], ABC-based [49], and GWO-WSN [50]. The network input configuration parameters are the same for all algorithms used. In addition, BS is at the center of the network and all sensor nodes have at least one neighbor. Each sensor can transfer the data packet to the BS in one or multi-hops. The maximum hop sizes to transfer a data packet from a node to the BS are assumed to be four. All sensor nodes are homogeneous and have the same initial energy level and communication range. The network has 500 rounds where the duration time for each round is 2 s. Additional details about the system are presented in Table 4.

Table 4 Input parameters for routing algorithm.

Parameters	Values
Network Size	100*100 m^2
Number of nodes	100
Base station location	50*50 *m*
Data packet size	4000 bits
e_0	0.5 J
e_{fs}	10 pj/bit/m2
e_{mp}	0.0013 pj/bit/m4
e_{elec}(TX, RX)	50n j/bit
Data Aggregation Energy cost	50n j/bit

Here the parameters of metaheuristic algorithms are defined. The I-GWO and Ex-GWO have 30 search agents (population size). a linearly decreases from 2 to 0. The A is a value between [-2, 2], and the range for C is [2, 0]. Furthermore, the maximum iterations are 100. In GAR algorithm, initial population is considered to have 30 chromosomes. In the crossover operation, 5 % of the best chromosomes are selected by the tournament selection procedure. Likewise, maximum number of iterations for the GAR algorithm is 100. In routing algorithm, namely ABCbased routing algorithm, the population size is fixed to be 30. The limit for neighborhood search is 20. In addition, the iteration maximum size is 100. In the MAP-ACO methods, the population size is 30 in 100 maximum iterations. $\alpha = 0.5$, $\beta = 0.5$, $c_1 = 0.15$, $c_2 = 0.20$, $c_3 = 0.25$, $c_4 = 0.4$, $\rho = 0.5$ and $\tau_{i,j}(0) = 0.008$. Finally, in the GWO-WSN, number of search agents is assumed to be 30 with 100 iterations.

The metrics for performance evaluation are i) network lifetime, ii) the alive node ratio in the network, iii) the packet delivery ratio and lost data packets, iv) routing overhead, v) throughput, and vi) convergence behavior. They clarify the performance of the routing methods and are simulated to evaluate the network, based on defined input parameters. These metrics are also used for evaluation and comparison purposes. Finally, the performance sequences of all algorithms are presented in Table 5. The following section discusses the results obtained.

5.2. Network lifetime

Fig. 8 presents the residual energy of the network during simulation rounds. This metric analyzes the remaining energy of the network. The results obtained show that both algorithms have nearly similar performance in energy consumption. But the energy consumption of the routing method based on the Ex-GWO algorithm is better compared to other methods. The routing and data transfers in both proposed methods are the same but, the metaheuristic structures are different. In Fig. 8, the performance of the network lifetime of proposed methods compared to other methods is presented in 500 rounds. As stated in Table 4, it is assumed that the energy of each node is 0.5 J and the number of sensor nodes is 100 in the network. Therefore, the total energy is evaluated over 50 J. Network connectivity and data are strongly based on the residual energy of the sensor nodes. In addition, in flat-based networks, energy level of each sensor affects the other. In this type of network, there is no cluster schema to transfer collected data from cluster heads to the BS. So, in a flat-based network, any sensor nodes' residual energy is important in network performance. Using this parameter, lifetime of the entire network is obtained. Related simulation results are presented in Fig. 8. Evidently, from the results, all the energy of the nodes based on the ABC algorithm is completely finished before the 300th round, before the 350th round in the GAR method, and before the 450th round in the GWO-WSN. In

Table 5 the obtained result for tests of nonparametric significance.

Algorithm	$EER_{I\text{-}GWO}$		$EER_{Ex\text{-}GWO}$	
	Friedman	Wilcoxon	Friedman	Wilcoxon
MAP-ACO	1.5610E-02	1.5113E-03	2.0145E-01	1.0196E-03
ABCbased	1.1556E-01	3.0151E-01	2.1513E-01	2.7851E-01
GAR	4.0016E-01	1.9581E-03	3.0191E-01	3.1541E-02
GWO-WSN	3.0110E-01	2.1501E-02	2.2158E-01	3.9612E-02

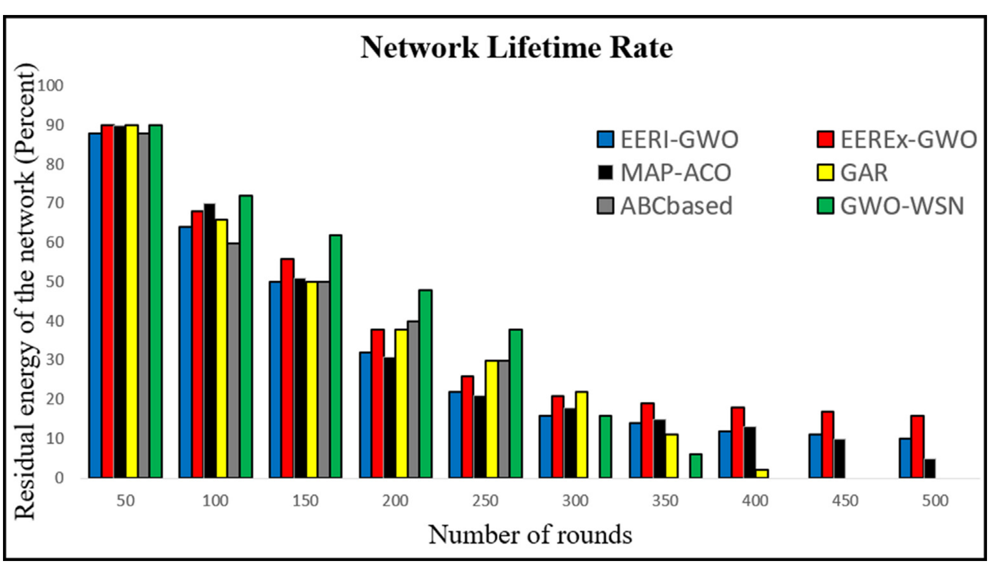

Fig. 8 Network lifetime ratio.

addition, MAP-ACO continues until the 500th round but is less than I-GWO and Ex-GWO based methods. In general, it is found to be more successful in the network lifetime parameter using I-GWO and Ex-GWO based methods. In addition, Ex-GWO based proposed method appears to provide the best performance.

In order to analyze the accuracy and reliability of the results obtained in this study, Friedman and Wilcoxon nonparametric statistical significance nonparametric tests were employed as a method of analyzing the significance of the results. This is accomplished by running each algorithm 30 times with 100 iterations so that we can achieve the desired result. As shown in Table 5, the p-values calculated with the $EER_{I\text{-}GWO}$ and the $EER_{Ex\text{-}GWO}$ can be summarized. Considering the results of the study, it seems that the I-GWO and Ex-GWO algorithms perform statistically better than the other algorithms used for this experimental design for a level of 0.05 than other algorithms with similar experimental conditions.

5.3. Alive nodes number

Fig. 9 shows the number of alive nodes in each round of the network for 100 nodes. The number of live nodes indicates the number of nodes energized in the round. The nodes with drained energy are so-called death nodes. The $EER_{I\text{-}GWO}$ and $EER_{Ex\text{-}GWO}$ methods also exhibit good performance in improving energy consumption on the number of nodes alive. This signifies that energy consumption in nodes is more balanced than in the other four algorithms. It should be noted that the network lifetime does not depend on the number of nodes surviving. The main point is that there is a path from the surviving nodes to the BS node.

5.4. Packet delivery

Fig. 10 represents the successful delivery packet ratio to evaluate the throughput between all algorithms. The throughput of the network is analyzed by the number of successfully delivered packets to the BS. So, the packets received by all the nodes can be summed to calculate the value. Indeed, the sum of the successfully received data packets by the BS, divided by the sum of all data packets gives this ratio. Packets that have not been received successfully are considered to be lost. As mentioned above, the total ratio of the two values must be 1. In other words, the packet loss performance of the algorithms is inversely proportional to the packet success rate. The proposed methods have a considerably good packet delivery rate compared to others. Based on the results obtained, it can be stated, with confidence, that for the network lifetime parameter and alive node ratio, $EER_{Ex\text{-}GWO}$ is in the first rank and $EERI_{\text{-}GWO}$ is in the second.

5.5. Routing overhead

Overhead is the number of resources used by every sensor node in the network. It is considered as the amount of request and reply packets for pathfinding and influences energy efficiency. With the increase in the number of nodes, the overhead and the number of transmissions increase. This parameter is calculated as a percentage. Based on the Fig. 11, results, $EER_{I\text{-}GWO}$ and $EER_{Ex\text{-}GWO}$ offered the best result, ranking first and second, respectively. The main reason for this is the proposed architecture. It has the least overhead rate, as the relevant metaheuristic algorithms are used in our pathfinding mechanisms efficiently and at a lesser cost. The reason why the $EER_{\text{-}I\text{-}GWO}$ method is better than the other methods compared is that the I-GWO depends solely on the alpha wolf. In the 500-node network, $EER_{I\text{-}GWO}$ has an overhead of 6.03 % and $EER_{Ex\text{-}GWO}$ is in second place with 7.18 %. In third place, the GWO-WSN method has a rate of 11.14 %. Other methods that follow are MAP-ACO, GAR, and ABCbased with 11.71 %, 12.42 %, and 13.53 % ratios, respectively.

5.6. Throughput

Throughput is defined as the number of packets delivered by the BS node per unit of time. It is an important factor in measuring the computational and time efficiency of a protocol. The results, shown in Fig. 12, depict that the proposed routing algorithms have better performance in the throughput parameter and their throughput values are higher than the other algorithms. In the scenario used, the BS is at 50*50 of the network. Furthermore, due to the spreading of sensors in the net-

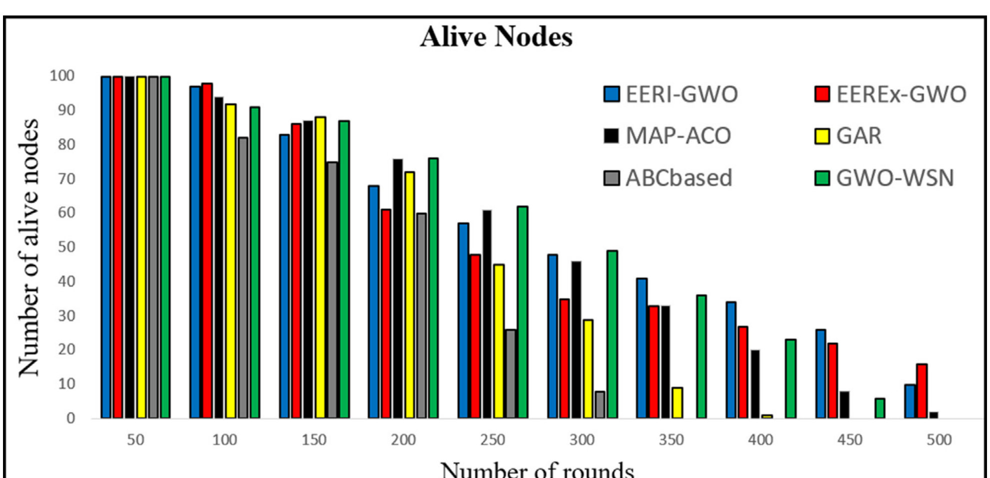

Fig. 9 Alive nodes number.

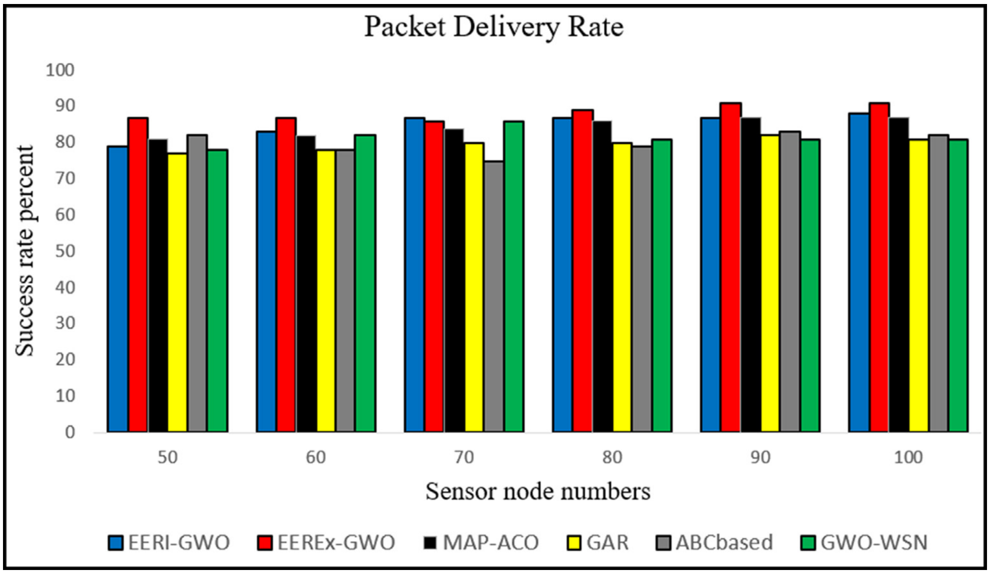

Fig. 10 Packet delivery ratio.

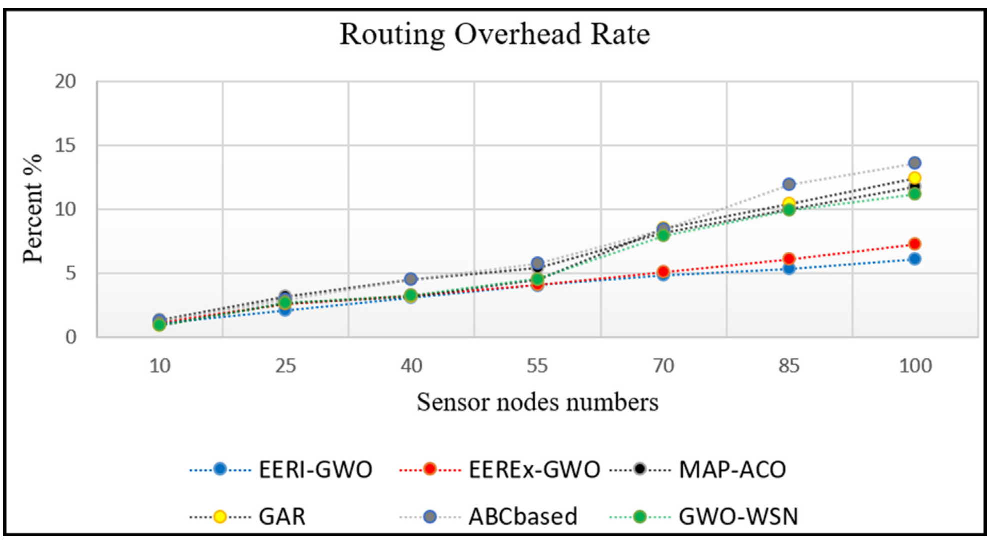

Fig. 11 Routing overhead ratio.

work, the efficiency of the proposed algorithms is significant. The throughput value is calculated using various sensor numbers to evaluate the performance of the proposed algorithms. Due to the decentralized structure of the sensor node in the network, the number of packets delivered to the BS is of significant importance. In the comparison of throughput results, $EER_{Ex\text{-}GWO}$ showcases the best performance among the six algorithms.

5.7. Convergence speed of the algorithm

The proposed routing algorithms in this paper benefit from two metaheuristic algorithms as mentioned before. Metaheuristic algorithms solve the problem in stochastic search space, as such the convergence analysis is of high significance. It is also worth mentioning that in WSN and DIoT, while designing routing protocols, it is important to find the appropriate path with low cost as the resources are limited. Therefore, these algorithms must have a high convergence speed and success rates to approach the best solution. The convergence curve of the proposed algorithms and the others are shown in Fig. 13. As presented in this figure, the I-GWO and Ex-GWO-based routing algorithms have a greater convergence curve, which causes a reduction in the computation time. This figure shows the instantaneous remaining energy in the network while each routing algorithm continues to run. According to the data in the Table 4, the network has an energy of 50 J in the first and the remaining energy will naturally decrease as the iterations progress. In the proposed routing algorithms, the main process tries to find the optimal path, in this way the high convergence speed caused a low computation cost. According to the simulation results obtained, the convergence speed of the proposed algorithms is higher than others. The best result is achieved using $EER_{Ex\text{-}GWO}$ method.

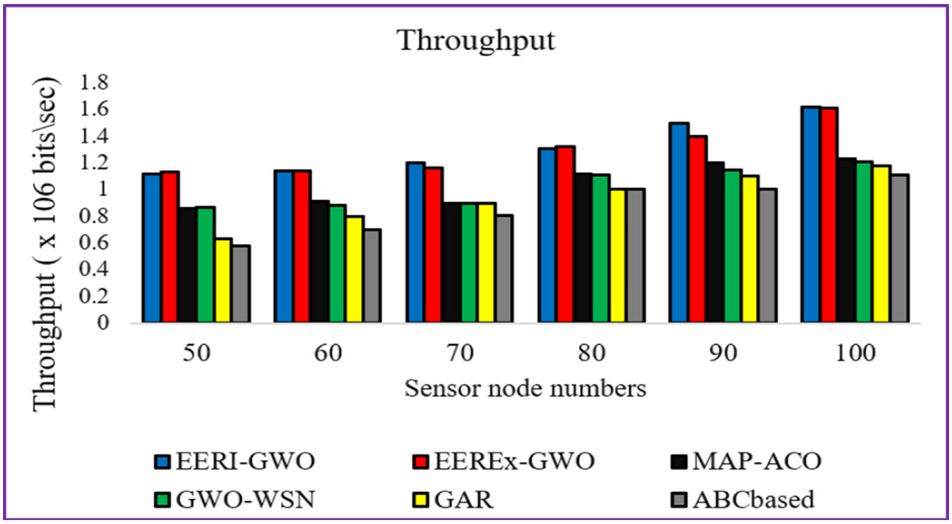

Fig. 12 Throughput analysis.

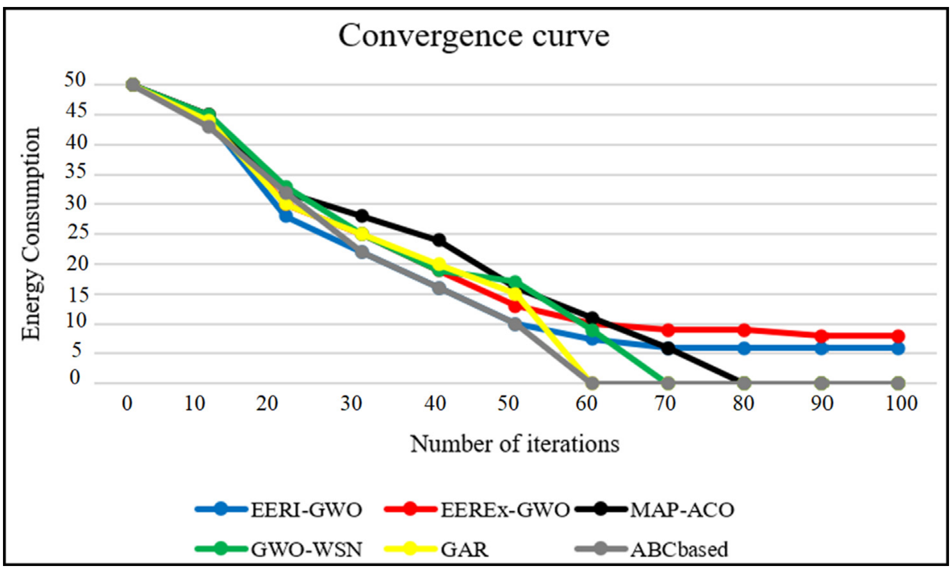

Fig. 13 Convergence speed analysis.

5.8. General comparison and discussion

In this study, simulations were investigated on 6 different parameters and the results of each were shown. In addition, in line with the results obtained, the methods used in each parameter have their performances ranked from the best to the worst (Table 6). As can be seen from the results obtained, both of the proposed methods perform better. I-GWO tries to find solutions more quickly thanks to its exploitation feature and fast convergence rate, and Ex-GWO, due to its structure, is likely to be successful in complex and large-scale systems. In these algorithms, swarming is controlled by the leader of the group, which helps to get the optimum solution for a defined problem. Besides, benefiting from the defined fitness functions and comprehensive architecture, these algorithms were made easier to adapt to the proposed pathfinding methods and exhibit efficient behavior. The third place, MAP-ACO, has performed well, but because of the use of a metaheuristic algorithm in all operations in its method, the performance in the network, naturally, was limited. The main reason for this is that the devices used have limited resources. However, in the architecture of this study, metaheuristic algorithms used were run only in the first part of the method and this causes increased efficiency. In fourth place, GWO-WSN is listed. GWO-WSN has not been very successful due to its non-comprehensive fitness function. However, this method could have had a more stable working mechanism due to its GWO structure. When the performance analysis of the other two methods is done, it is seen that they are not very successful.

6. Conclusion and future works

This work solved one of the main challenges in wireless sensor networks and decentralized IoT systems by improving the

Table 6 Rank of algorithms performance (Summary).

Parameters	$EER_{I\text{-}GWO}$	$EER_{Ex\text{-}GWO}$	MAP-ACO	GAR	ABCbased	GWO-WSN
Packet Delivery Rate	2	**1**	3	6	4	5
Alive Nodes Number	2	**1**	3	5	6	4
Network Lifetime Rate	2	**1**	3	5	6	4
Overhead	**1**	2	4	5	6	3
Throughput	2	**1**	3	5	6	4
Convergence	2	**1**	3	5	6	4

energy consumption of the network. It finds the best route by examining all available paths between any two nodes with a proposed general architecture. Finding the best routes between nodes results in less energy being consumed in the network, thus efficient use of resources and increasing the overall lifetime of the system. Thanks to this architecture, many metaheuristic algorithms can work in an adaptive way, so it takes the role of a multi-purpose general model and will provide convenience to researchers working in this field. In this study, $EERI_{-GWO}$ and $EER_{Ex\text{-}GWO}$ routing methods are proposed using I-GWO and Ex-GWO algorithms as metaheuristic algorithms. These two methods are energy efficient routing methods that try to find optimum paths. These methods provide more efficient execution time and CPU power in time and space complexities. The search space is considered as a matrix, where the rows represent the number of search agents, and the column signifies the coefficient numbers. These coefficients are updated by the metaheuristics used.

This study focuses on broader parameters in proposing more comprehensive and accurate methods in WSN and DIoT. Accordingly, a new fitness function has been defined. The defined fitness function is used to calculate the cost of each path in the network and includes residual energy, traffic status, buffer rate, BS-hop, and neighbor list of each node. The paths between the two source and destination nodes are selected according to hop values and passed through the fitness function. The sum of the best fitness values for each hop will be the candidate for the best route. Subsequently, the minimum value among candidates is chosen as the best path between the two competing nodes. Each node acquires its best neighbor from its routing table. Related network equations were mapped in accordance with metaheuristic algorithms. The performances of two metaheuristic algorithms used in the proposed routing methods were evaluated on various parameters. After iterations of metaheuristic algorithms, the best solution is found as an optimal path for the network in the current rounds. The routing operations are performed in the BS. The results have displayed those proposed methods have better performance than ABCbased, GAR, GWO-WSN, and MAP-ACO methods. Furthermore, results show that these two methods are more successful in finding the most appropriate paths in these systems. According to the results of this study, and other studies in the literature, it can be said with confidence that swarm intelligence is stronger than particle intelligence in similar systems. The proposed methods in this paper may be more suitable for a network of any scale. In these methods, the most appropriate and efficient path can be found between multiple sources and destination nodes concurrently or in parallel (depending on the needs of the problem and application area). In addition, the proposed methods have better performance in terms of robustness and fault tolerance factors. Apart from the pros and strengths of the study, the shortcomings can be summarized as follows.

The shortcomings of this study are planned to be continued and completed in future studies. In this study, no tests were performed on a real system covering big data. In this study, simulations were made using homogeneous sensor nodes. However, heterogeneous sensor nodes were not used. This study did not focus on the multi-objective and Pareto-based problem. In future work, the proposed methods will be tested on real testbeds with a large density of various devices for the generation and analysis of big data. Similarly, the proposed approach can perform more efficiently in multi-objective and Pareto-based problems. Especially in parameters that have trade-offs with each other (e.g., network connectivity and energy consumption) can be applied. The proposed methods will be used for solving many complex problems such as feature selection, complex electrical circuits, 3D path planning in mobile robotics or connected vehicle networks, and optimized node localization in the systems. It should be noted that with the growth in IoT technology, most of the proposed path planning methods focus on homogeneous sensor networks, but IoT devices can greatly benefit from heterogeneous sensor nodes. In this way, this work can help use heterogeneous sensor nodes to support different IoT devices. Accordingly, the proposed method can be easily applied to the wearable sensor network, which has become extremely popular in the last decade.

Funding

The work of U.F.-G. was supported by the government of the Basque Country for the ELKARTEK21/10 KK-2021/00014 and ELKARTEK22/85 research programs, respectively.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] M. Bottero, B.D. Chiara, F.P. Deflorio, Wireless sensor networks for traffic monitoring in a logistic centre, Transportation Research Part C: Emerging Technologies 26 (2013) 99–124.

[2] A.S. Ibrahim, K.Y. Youssef, A.H. Eldeeb, M. Abouelatta, H. Kamel, Adaptive aggregation based IoT traffic patterns for optimizing smart city network performance, Alexandria Engineering Journal 61 (12) (2022) 9553–9568.
[3] F. Kiani, A. Seyyedabbasi, Wireless Sensor Network and Internet of Things in Precision Agriculture, International Journal of Advanced Computer Science and Applications (IJACSA) 9 (6) (2018) 99–103.
[4] F. Kiani, G. Randazzo, I. Yelmen, A. Seyyedabbasi, S. Nematzadeh, F.A. Anka, A smart and mechanized agricultural application: From cultivation to harvest, Applied Sciences-Basel 12 (12) (2022) 1–22.
[5] J. Tavares, F. Velez, J. Ferro, Application of wireless sensor networks to automobiles, Measurement Science Review 8 (3) (2008) 65–70.
[6] S. Li, L. Meng, J. Liu, R. Wang, Design of a dynamic monitoring system for patient health indexes based on mobile terminal, Alexandria Engineering Journal 60 (5) (2021) 4573–4582.
[7] Nayyar A., Nguyen BL. & Nguyen N.G. (2020). The Internet of Drone Things (IoDT): Future Envision of Smart Drones, First International Conference on Sustainable Technologies for Computational Intelligence. Advances in Intelligent Systems and Computing, Springer, 1045, 563-580.
[8] N.S. Labib, M.R. Brust, G. Danoy, P. Bouvry, The Rise of Drones in Internet of Things: A Survey on the Evolution, Prospects and Challenges of Unmanned Aerial Vehicles, IEEE Access 9 (2021) 115466–115487.
[9] Y. Bouzembrak, M. Klüche, A.G. Hans, J.P. Marvin, Internet of Things in food safety: Literature review and a bibliometric analysis, Trends Food Sci. Technol. 94 (2019) 54–64.
[10] A. Gatouillat, Y. Badr, B. Massot, E. Sejdic, Internet of Medical Things: A Review of Recent Contributions Dealing with Cyber-Physical Systems in Medicine, IEEE Internet Things J. 5 (5) (2018) 3810–3822.
[11] E. Sisinni, A. Saifullah, S. Han, U. Jennehag, M. Gidlund, Industrial Internet of Things: Challenges, Opportunities, and Directions, IEEE Trans. Ind. Inf. 14 (11) (2018) 4724–4734.
[12] L. Kong, M.K. Khan, F. Wu, G. Chen, P. Zeng, Millimeter-Wave Wireless Communications for IoT-Cloud Supported Autonomous Vehicles: Overview, Design, and Challenges, IEEE Commun. Mag. 55 (1) (2017) 62–68.
[13] L. Sumi, V. Ranga, An IoT-VANET-Based Traffic Management System for Emergency Vehicles in a Smart City, Advances in Intelligent Systems and Computing, Springer 708 (2018) 23–31.
[14] F. Kiani, Animal behavior management by energy-efficient wireless sensor networks, Computer and Electronic in Agriculture 151 (2018) 478–484.
[15] F. Losilla, A.J. Garcia-Sanchez, F. Garcia-Sanchez, J. Garcia-Haro, Z.J. Haas, A Comprehensive approach to WSN-based ITS applications: a survey, Sensors (Basel, Switzerland) 11 (11) (2011) 10220–10265.
[16] F. Kiani, S. Nematzadehmiandoab, A. Seyyedabbasi, Designing a dynamic protocol for real-time Industrial Internet of Things-based applications by efficient management of system resources, Advances in Mechanical Engineering 11 (10) (2019) 1–23.
[17] A. Seyyedabbasi, R. Aliyev, F. Kiani, M. Gulle, H. Basyildiz, M. Shah, Hybrid algorithms based on combining reinforcement learning and metaheuristic methods to solve global optimization problems, Knowl Based Syst 223 (2021) 1–22.
[18] J. Chen, Z. Wei, S. Li, B. Cao, Artificial Intelligence Aided Joint Bit Rate Selection and Radio Resource Allocation for Adaptive Video Streaming over F-RANs, IEEE Wirel. Commun. 27 (2) (2020) 36–43.
[19] A. Seyyedabbasi, F. Kiani, MAP-ACO: An efficient protocol for multi-agent pathfinding in real-time WSN and decentralized IoT systems, Microprocess. Microsyst. 79 (2020) 1–9.
[20] F. Kiani, AR-RBFS: Aware-Routing Protocol Based on Recursive Best-First Search Algorithm for Wireless Sensor Networks, Journal of Sensors 2016 (2016) 1–10.
[21] A. Seyyedabbasi, G. Dogan, F. Kiani, HEEL: A new clustering method to improve wireless sensor network lifetime, IET Wireless Sens. Syst. 10 (3) (2020) 130–136.
[22] A. Mazinani, S.M. Mazinani, M. Mirzaie, FMCR-CT: An energy-efficient fuzzy multi cluster-based routing with a constant threshold in wireless sensor network, Alexandria Engineering Journal 58 (1) (2019) 127–141.
[23] Y. Han, G. Li, R. Xu, J. Su, J. Li, G. Wen, Clustering the Wireless Sensor Networks: A Meta-Heuristic Approach, IEEE Access 8 (2020) 214551–214564.
[24] G. Anastasi, M. Conti, M. Francesco, A. Passarella, Energy conservation in wireless sensor networks: A survey, Ad Hoc Netw. 7 (3) (2009) 537–568.
[25] F. Kiani, A. Seyyedabbasi, S. Nematzadeh, Improving the performance of hierarchical wireless sensor networks using the metaheuristic algorithms: efficient cluster head selection, Sensor Review (2021), https://doi.org/10.1108/SR-03-2021-0094.
[26] R. Chaudhry, N. Kumar, A multi-objective meta-heuristic solution for green computing in software-defined wireless sensor networks, IEEE Transactions on Green Communications and Networking 6 (2) (2022) 1231–1241.
[27] R. Kumar, R.P. Mahapatra, Hybrid metaheuristic algorithm for optimal cluster head selection in wireless sensor network, Pervasive Mob. Comput. 79 (2022) 1–14.
[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] S. Mirjalili, S.M. Mirjalili, A. Lewis, Grey wolf optimizer, Adv. Eng. Softw. 69 (2014) 46–61.
[30] C. Blum, A. Roli, Metaheuristics in combinatorial optimization: Overview and conceptual comparison, ACM computing surveys (CSUR) 35 (3) (2003) 268–308.
[31] S. Binitha, S.S. Sathya, A survey of bio inspired optimization algorithms, International journal of soft computing and engineering 2 (2) (2012) 137–151.
[32] S. Mirjalili, A. Lewis, The whale optimization algorithm, Adv. Eng. Softw. 95 (2016) 51–67.
[33] H. Zamani, M.H. Nadimi-Shahraki, A.G. Gandomi, Starling murmuration optimizer: A novel bio-inspired algorithm for global and engineering optimization, Comput. Methods Appl. Mech. Eng. 392 (2022) 114616.
[34] D. Karaboga, B. Basturk, On the performance of artificial bee colony (ABC) algorithm, Appl. Soft Comput. 8 (1) (2008) 687–697.
[35] A. Seyyedabbasi, F. Kiani, Sand Cat swarm optimization: a nature-inspired algorithm to solve global optimization problems, Engineering with Computers (2022), https://doi.org/10.1007/s00366-022-01604-x.
[36] A. Seyyedabbasi, F. Kiani, I-GWO and Ex-GWO: improved algorithms of the Grey Wolf Optimizer to solve global optimization problems, Engineering with Computers 37 (2021) 509–532.
[37] Cuevas, E., Rodríguez, A., Alejo-Reyes, A., Del-Valle-Soto, C. (2021). Metaheuristic Algorithms for Wireless Sensor Networks. In: Recent Metaheuristic Computation Schemes in Engineering. Studies in Computational Intelligence, Springer, 948, 31-52.
[38] B.M. Sahoo, H.M. Pandey, T. Amgoth, GAPSO-H: A hybrid approach towards optimizing the cluster based routing in wireless sensor network, Swarm Evol. Comput. 60 (100772) (2021) 1–19.
[39] B.M. Sahoo, T. Amgoth, H.M. Pandey, Particle swarm optimization based energy efficient clustering and sink mobility in heterogeneous wireless sensor network, Ad Hoc Netw. 106 (2020) 102237.

[40] S. Okdem, D. Karaboga, Routing in wireless sensor networks using an ant colony optimization (ACO) router chip, Sensors 9 (2) (2009) 909–921.
[41] A. Mohajerani, D. Gharavian, An ant colony optimization based routing algorithm for extending network lifetime in wireless sensor networks, Wireless Netw. 22 (8) (2016) 2637–2647.
[42] D. Yang, H. Xia, E. Xu, D. Jing, H. Zhang, Energy-Balanced Routing Algorithm Based on Ant Colony Optimization for Mobile Ad Hoc Networks, Sensors (Basel, Switzerland) 18(11) (3657) (2018) 1–19.
[43] A. Jiang, L. Zheng, An effective hybrid routing algorithm in WSN: Ant colony optimization in combination with hop count minimization, Sensors 18 (4) (2018) 1020.
[44] K.N. Bui, J. Jung, ACO-Based Dynamic Decision Making for Connected Vehicles in IoT System, IEEE Trans. Ind. Inf. 15 (10) (2019) 5648–5655.
[45] T. Vaiyapuri, V.S. Parvathy, V. Manikandan, N. Krishnaraj, D. Gupta, K. Shankar, A novel hybrid optimization for cluster-based routing protocol in information-centric wireless sensor networks for IoT based mobile edge computing, Wireless Pers. Commun. (2021) 1–24.
[46] A. Kavitha, R.L. Velusamy, Simulated annealing and genetic algorithm-based hybrid approach for energy-aware clustered routing in large-range multi-sink wireless sensor networks, Int. J. Ad Hoc Ubiquitous Comput. 35 (2) (2020) 96–116.
[47] S.K. Gupta, P. Kuila, P.K. Jana, GAR: an energy efficient GA-based routing for wireless sensor networks. Distributed Computing and Internet Technology. ICDCIT 2013, Springer 7753 (2014) 267–277.
[48] Vaiyapuri, T., Parvathy, V. S., Manikandan, V., Krishnaraj, N., Gupta, D., & Shankar, K. (2021). Anovel hybrid optimization for cluster-based routing protocol in information-centric wireless sensornetworks for IoT based mobile edge computing. $Wireless Personal Communications, 1-24.
[49] A. Ari, B.O. Yenke, N. Labraoui, I. Damakoa, A. Gueroui, A power efficient cluster-based routing algorithm for wireless sensor networks: Honeybees swarm intelligence based approach, Journal of Network and Computer Applications 69 (2017) 77–97.
[50] Al-Aboody N. A. & Al-Raweshidy H. S. (2017). Grey wolf optimization-based energy-efficient routing protocol for heterogeneous wireless sensor networks, 4th International Symposium on Computational and Business Intelligence (ISCBI), Olten, Switzerland, 2016, 101-107.
[51] A. Lipare, D.R. Edla, V. Kuppili, Energy efficient load balancing approach for avoiding energy hole problem in WSN using Grey Wolf Optimizer with novel fitness function, Appl. Soft Comput. 84 (2020) 1–11.
[52] K.M. Awan, H.H.R. Sherazi, A. Ali, R. Iqbal, Z.A. Khan, M. Mukherjee, Energy-aware cluster-based routing optimization for WSNs in the livestock industry, Transactions on Emerging Telecommunications Technologies 33 (3) (2022) e3816.
[53] K. Lakshmanna, N. Subramani, Y. Alotaibi, S. Alghamdi, O.I. Khalafand, A.K. Nanda, Improved Metaheuristic-Driven Energy-Aware Cluster-Based Routing Scheme for IoT-Assisted Wireless Sensor Networks, Sustainability 14 (13) (2022) 7712.
[54] E. Heidari, A. Movaghar, H. Motameni, B. Barzegar, A novel approach for clustering and routing in WSN using genetic algorithm and equilibrium optimizer, Int. J. Commun Syst e5148 (2022).
[55] K. Suresh, S.S. Sreeja Mole, J.S.A. Kumar, F2SO: an energy efficient cluster based routing protocol using fuzzy firebug swarm optimization algorithm in WSN, The Computer Journal. (2022).
[56] R.K. Yadav, R.P. Mahapatra, Hybrid metaheuristic algorithm for optimal cluster head selection in wireless sensor network, Pervasive Mob. Comput. 79 (2022) 101504.
[57] M. Mir, M. Yaghoobi, M. Khairabadi, A new approach to energy-aware routing in the Internet of Things using improved Grasshopper Metaheuristic Algorithm with Chaos theory and Fuzzy Logic, Multimedia Tools and Applications (2022) 1–27.