sustainability

MDPI

Article

Energy Efficient Clustering Protocol for FANETS Using Moth Flame Optimization

Salil Bharany [1], Sandeep Sharma [1], Surbhi Bhatia [2], Mohammad Khalid Imam Rahmani [3,*], Mohammed Shuaib [4] and Saima Anwar Lashari [3,*]

[1] Department of Computer Engineering & Technology, Guru Nanak Dev University, Amritsar 143005, India; salil.bharany@gmail.com or salil.csersh@gndu.ac.in (S.B.); sandeep.cse@gndu.ac.in (S.S.)
[2] Department of Information Systems, College of Computer Sciences and Information Technology, King Faisal University, Al Hofuf 36362, Saudi Arabia; surbhibhatia1988@yahoo.com
[3] College of Computing and Informatics, Saudi Electronic University, Riyadh 11673, Saudi Arabia
[4] College of Computer Science & IT, Jazan University, Jazan 45142, Saudi Arabia; talkshuaib@gmail.com
* Correspondence: m.rahmani@seu.edu.sa (M.K.I.R.); s.lashari@seu.edu.sa (S.A.L.)

Abstract: FANET (flying ad-hoc networks) is currently a trending research topic. Unmanned aerial vehicles (UAVs) have two significant challenges: short flight times and inefficient routing due to low battery power and high mobility. Due to these topological restrictions, FANETS routing is considered more complicated than MANETs or VANETs. Clustering approaches based on artificial intelligence (AI) approaches can be used to solve complex routing issues when static and dynamic routings fail. Evolutionary algorithm-based clustering techniques, such as moth flame optimization, and ant colony optimization, can be used to solve these kinds of problems with routes. Moth flame optimization gives excellent coverage while consuming little energy and requiring a minimum number of cluster heads (CHs) for routing. This paper employs a moth flame optimization algorithm for network building and node deployment. Then, we employ a variation of the K-Means Density clustering approach to choosing the cluster head. Choosing the right cluster heads increases the cluster's lifespan and reduces routing traffic. Moreover, it lowers the number of routing overheads. This step is followed by MRCQ image-based compression techniques to reduce the amount of data that must be transmitted. Finally, the reference point group mobility model is used to send data by the most optimal path. Particle swarm optimization (PSO), ant colony optimization (ACO), and grey wolf optimization (GWO) were put to the test against our proposed EECP-MFO. Several metrics are used to gauge the efficiency of our proposed method, including the number of clusters, cluster construction time, cluster lifespan, consistency of cluster heads, and energy consumption. This paper demonstrates that our proposed algorithm performance is superior to the current state-of-the-art approaches using experimental results.

Keywords: FANETS; energy efficiency; clustering; routing; WSN; Cloud; transmission range; bio-inspired

check for updates

Citation: Bharany, S.; Sharma, S.; Bhatia, S.; Rahmani, M.K.I.; Shuaib, M.; Lashari, S.A. Energy Efficient Clustering Protocol for FANETS Using Moth Flame Optimization. *Sustainability* **2022**, *14*, 6159. https://doi.org/10.3390/su14106159

Academic Editors: Saqib Iqbal Hakak and Thippa Reddy Gadekallu

Received: 8 April 2022
Accepted: 12 May 2022
Published: 19 May 2022

1. Introduction

A FANET is a hybrid model of VANET (vehicular ad hoc networks) and a mobile ad-hoc network (MANET). In FANETs, UAVs are the network nodes, and the peer-to-peer communication model is used in communication. FANETs, MANETs, and VANETs have many similar properties. However, there are also some significant variances. These characteristics make FANETs incredibly fast-moving and capable of coping with various 3D environments. The FANET's changing structure and hostile surroundings make networking and communication difficult [1,2]. While in the air, the nodes have a clear line of sight and are pretty far apart. UAVs have limited resources (batteries, processing power, and bandwidth), which also hampers UAVs working efficiency. UAVs need to devise a communication system with minimal routing overhead, high throughput, and little processing complexity to make the most reliable working model [2]. We can see that communication in an ad-hoc network can be accomplished via various methods. There are three main types of routing: proactive, reactive, and cluster-based. Routes to other nodes in the network are preserved in routing tables in proactive routing. Each time a message is exchanged, a new routing table is created. We can see a FANETS network in Figure 1 used for monitoring and searching applications.

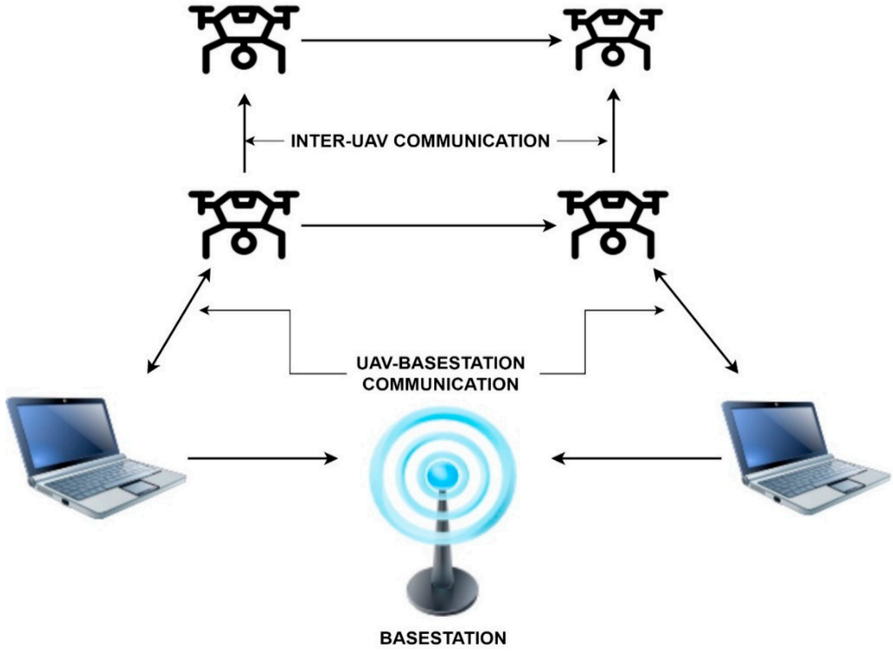

Figure 1. FANETS Network.

On the other hand, reactive routing protocols do not keep track of the routing table. Every time a node needs to communicate data, it searches for a new route. It helps to find a new path once the present one is disrupted [2,3]. When nodes constantly move around, maintaining the current routing table or finding a new way is a substantial communication overhead. Network throughput is cut down, extra delays are brought in, and the limited battery power of the UAV is wasted as a result of these communication overheads [4]. There is a solution to the problem of limited resources in FANETS, which is clustering. Clustering is a method for grouping nodes based on their geographic location. It contributes to the network's scalability, efficiency, and throughput [4,5] as the UAVs' energy consumption will be reduced due to the minimal routing overhead. There is a cost associated with clustering, although it is substantially lower than the cost of the other approaches. When a network is clustered, it is broken down into several distinct groups or clusters [6,7]. There is a leader or "cluster head" (CH) for each cluster, which is in charge of communication between clusters and inside clusters, as seen in Figure 1. CH serves as a first-hop node that takes the message to its final destination to facilitate routing [8].

Cluster creation and maintenance consume some of the computational network radio resources. In the early stages of cluster development, nodes communicate information about their immediate area, including node ID, location, residual energy, etc. are computed. Control messages are used to provide this information to other nodes. As a result, a portion of the network's radio resources is dedicated to clustering [8]. A CH is elected in each cluster during network clustering, where nodes can execute calculations to organize neighboring nodes into clusters. The development of clusters and the selection of CHs are critical to the stability of the cluster structure [6,7]. It is possible to alter the design of a cluster by repositioning cluster members (CMs). It is imperative that each CH regularly broadcasts its presence to its CMs and that each CM responds with its current status so that any changes to the cluster structure may be tracked. The radio resources of the network are also used for this cluster maintenance signaling. Overhead for cluster creation and maintenance is defined as transposing of control messages. In addition to using radio resources (such as channel bandwidth), this overhead is also a drain on the UAVs' energy. A clustering model's performance is evaluated using several parameters, including the cluster's lifespan and building time [4,9]. As the cluster's life span increases, the model's ability to save money and time increases. Clustering algorithms significantly impact a network's performance, and to ensure the cluster's long-term viability, the CHs must be carefully chosen. Networks may be divided into clusters using artificial intelligence algorithms [10]. The fundamental downside of these strategies is their high computing complexity, which prevents them from providing the best outcomes. They take too long to get to the best possible work [7]. Clustering in FANETs cannot be carried out using costly approaches due to the dynamic nature of the environment and the low processing capability of UAVs. UAV energy usage is also heavily influenced by transmission power options. Transmission power and energy use go hand in hand. SNs will use more energy if we choose a transmission power of either high or low [4]. Transmission power should not be as high as possible while not as low as possible to minimize energy wastage.

Evolutionary algorithms (EA) are based on the theory of biological evolution [7]. EA includes genetic algorithms/programming, evolutionary techniques, and learning classifier systems. When other strategies fail, evolutionary algorithms can provide a viable alternative. Evolutionary approaches are frequently accepted when faced with issues that seem impossible [8,11] to solve. Even if EA is computationally costly, a near-optimal solution to an unsolved case is acceptable. Node clustering using evolutionary algorithms may be carried out well in FANETS and VANETS systems. To identify a solution to the problem, an environment will be created that allows for the evolution of viable solutions [12]. It is feasible to find the best possible solution for the issues related to building environments through these biological algorithms. Nodes are aggregated, and their geographic locations are shared to tackle the scalability problem. Load balancing ensures that resources are effectively used in each cluster. One of the best clustering methods is the moth flame optimizer, which provides the ideal number of clusters. Insects that resemble butterflies are called moths and currently, there are about 16,000 moth species known to science. The larvae of moths, as with those of other insects, develop into cocoons as they mature. Moths fly at night and use the moonlight to find their way around [13]. An intelligent moth flame optimization-based clustering for FANET was developed to reduce energy consumption and expand the coverage area of SNs. This study proposes EECP-MFO, which employs a moth flame and a variation of the K-Means Density clustering method to choose CHs for usage in a distributed computing environment. K-Means Density is utilized in conjunction with the MFO algorithm for the first selection of centroids/CH. The original K-Means Density algorithm only considers one parameter, namely the degree of the neighborhood [13]. However, the EECP-MFO algorithm considers two additional factors, namely the energy level and the distance between the neighbors, to select the ideal CH. EECP-MFO extends the lifetime of a cluster while simultaneously reducing its energy usage [14–16]. It also saves energy by efficiently choosing the transmission power of nodes according to operational needs, reducing the network load [16–19]. In terms of cluster construction time, cluster longevity, the chance of success [20–24], and energy consumption, EECP-MFO beats the ant colony optimization [17,20], PSO-particle swarm optimization [18], and grey wolf optimization-based clustering models [14,25].

In a nutshell, the originality and contribution of this article are summarized in the following section.

- A method for data clustering based on the MFO is presented;
- The quality of the solutions provided by the suggested technique is compared to three well-known algorithms to determine which is superior;
- A total of five statistical tests have been conducted using different grid sizes to evaluate the proposed approach's statistical quality;
- The use of k-means density and the MRCQ approach for data compression has been employed to improve the CH selection process;

- Experimental and statistical graphs demonstrate the effectiveness of the suggested technique.

The remainder of the paper is arranged as follows. In Section 2, the background information on FANETS and Moth flame optimization algorithms can be seen. In Section 3, a description and working details of our proposed EECP-MFO are present. While our result and discussion is presented in Sections 4 and 5, we end the paper with a conclusion in Section 6.

2. Background and Motivation

2.1. FANETS

An unmanned aerial system (UAS) comprises small unmanned aerial vehicles (UAVs) that are small in size, flexible, and quick to deploy, as seen in Figure 2.

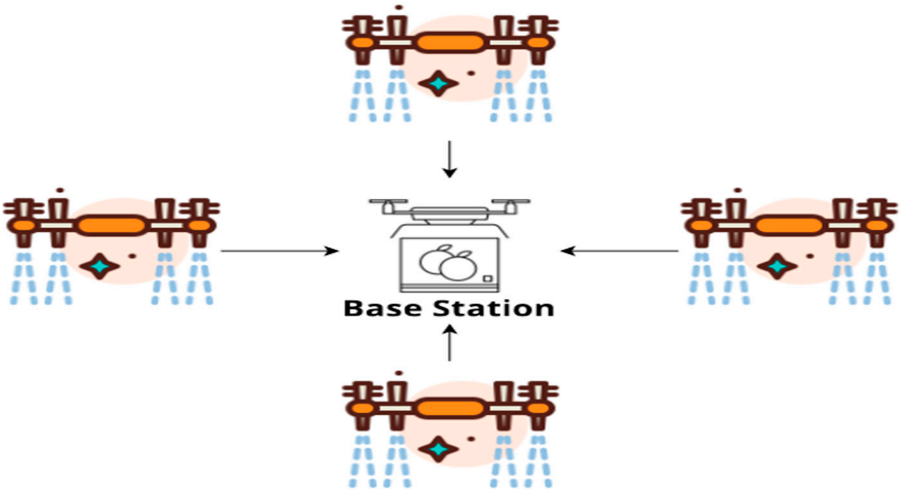

Figure 2. FANETS structure.

When a single UAV is employed, a star topology network is present where the UAV is at the center. A ground node can indirectly connect with other ground nodes via the UAV [19,20]. On the other hand, single UAV systems have specific complex challenges in peer-to-peer communication [26], such as decreasing interference, improving transmission range, and transmitting more data to multiple UAV systems [27].

The use of high gain directional antennas, rather than omnidirectional antennas, can be found as one of the possible solutions to these issues [21]. If the UAV or a sensor/hardware fails in a single UAV system, the UAV should return to its home base to be repaired or replaced [5]. The fault tolerance of a multi-UAV system is increased since other UAVs may share duties among themselves, which raises the overall fault tolerance of the system. It is feasible to take advantage of the capabilities of other UAVs while working with a heterogeneous UAV cluster [22]. FANET is also considered a component of the VANET network. Due to the excellent mobility of nodes in FANETs networks, conventional routing methods are also not practical in FANETs and do not offer sufficient throughput [19–21]. FANET topologies change significantly more often than MANETs or vehicle ad hoc networks, which is typical [15]. FANETs, which are not only applicable for multi-UAV systems but can also be formed by single-UAV systems, according to the specification seen in Figure 3.

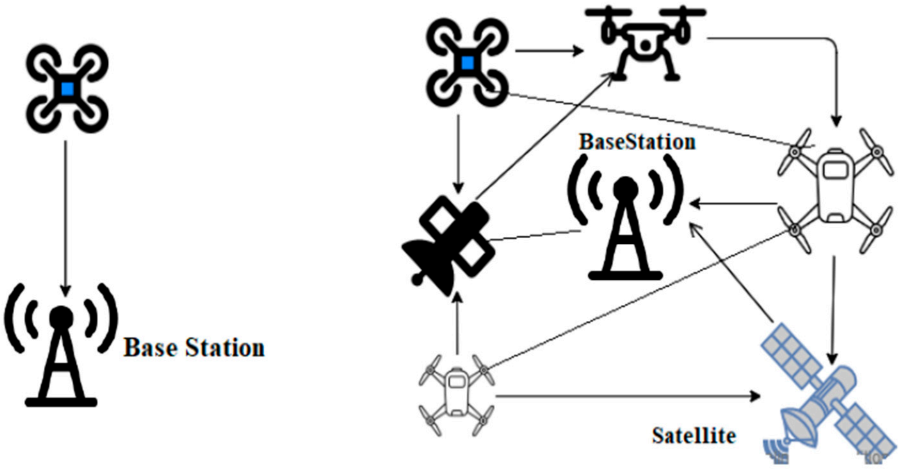

Figure 3. Types of FANETs network.

2.2. *What Links Make FANETs, Different from MANETs, and VANETs?*

The IEEE 802.11a network technology is today's essential network technology for constructing FANETs. Mesh ad hoc networks utilize the 802.11s wireless networking standard [14]. When two UAVs interact, this is called Single Hop Communication. Alternatively, a Multi-Hop Communication channel may be created across the other UAVs in UAV-to-UAV communication, as seen in Figure 3.

Fixed infrastructure, such as a satellite or a ground station, is used in the case of FANETS-to-FANETS Communication, in which UAVs connect with improved infrastructure. FANET may be a more advanced version of MANET and VANET [15]. However, there are several distinctions between FANET and the existing ad hoc networks, including the following: FANET nodes have far greater mobility than MANET or VANET nodes. FANET nodes fly in the air, in contrast to the traditional MANET and VANET nodes, representing walking men and vehicles.

1. Due to the high mobility of FANET nodes, the network's topology changes more frequently than the network itself [16,19,23]. A typical MANET or VANET network topology is shown in Figure 4.

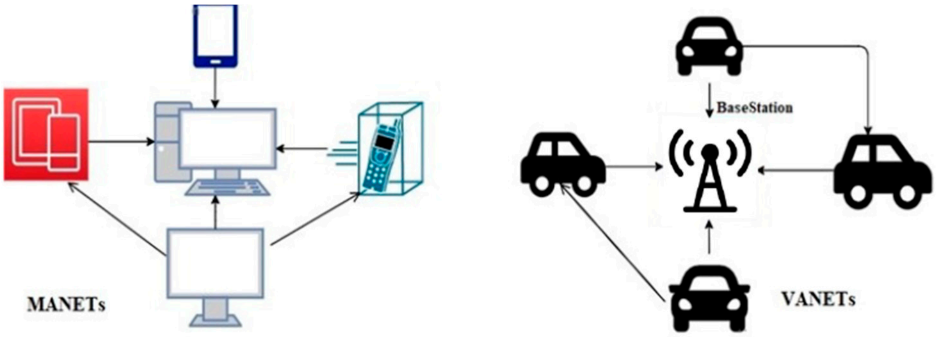

Figure 4. MANETS and VANETS network structure.

2. There are now many ad hoc networks striving to connect. For example, Wireless sensor networks make extensive use of these technologies for gathering and transmitting data about the surrounding environment [24]. Peer-to-peer and broadcast traffic must be allowed simultaneously for FANET to work effectively.
3. The Distances between FANET and FANETS nodes are much longer than the distances between the two networks [12,13,24]. Unmanned aerial vehicles (UAVs) need a more extended communication range than either MANETs or VANETs if they are to be linked together. Consequently, radio links, hardware circuits, and physical layer behaviour are all impacted.

4. Multi-UAV systems may contain various types of sensors, each of which may need a separate data transmission strategy [19].

2.3. What's the Roles of Bioinspired Algorithms in FANETs?

A clustering method comprises virtual sets, which are represented by clusters. Each cluster consists of cluster nodes (CN) or cluster members, each contributing to the selection or suggestion of a CH [15]. A CH's neighbors' nodes within the CH's transmission range are considered cluster members. In theory, any cluster node can be nominated as the cluster head. However, some characteristics may be more critical for a CH than others [20,21]. These characteristics are considered when selecting a CH. Using the example, an SN with an additional 5G connection is preferred to become a CH than all nodes that do not have this characteristic [21–24]. The transmission range of the nodes controls the size of a cluster and, as a result, the outcome varies from cluster to cluster [22].

The term "optimization" refers to improving performance by efficiently using specific limited resources that exist. During the previous several years, the difficulties associated with solving specific issues have increased, resulting in the need for novel optimization strategies to handle a problem efficiently. The methods were theoretically modelled before developing heuristic optimization algorithms to get the desired results. The most common drawback with mathematical optimization approaches is that local optima tend to become stuck in place [21,22,25,26]. This makes them exceedingly ineffective in resolving real-world problems due to their behaviors. Population-based algorithms, based on randomization, include two critical steps for obtaining improved results: solution development (finding the optimal answer) and exploration (finding alternate solutions). The MFO method is an innovative nature-inspired optimization technique that may be used to address challenging optimization issues. Transverse orientation mimics the movement of moths in nature [22]. Our proposed algorithm uses moths and flames techniques, by which it generates efficient solutions.

Moreover, it has been proven that this algorithm can outperform other meta-heuristic optimization techniques in terms of performance. It takes its inspiration from transverse orientation, which is the term used to describe the maritime strategy of moths [20]. Keeping a steady angle concerning the moon's movement allows a moth to hover, as seen in Figure 5. This method is quite effective in drifting over long distances with a straight track. Quite the opposite, artificial lighting can occasionally fool moths [27]. As a result, in comparison to the moon, a manufactured light is very close. While maintaining an equivalent viewpoint to the light, a moth may begin to travel in a spiral route with the light source, which can be disastrous for it [7,8,11]. This technique has many advantages, but two stand out.

First and foremost, MFO avoids the problem of local optima stagnation [12]. However, some other optimization methods, such as the genetic algorithm (GA) [7–10], continue to suffer from this problem. Second, MFO offers tremendous potential for exploitation and exploration, enabling it to outperform other processes in the long run.

However, even though FANETs are part of ad-hoc networks, they cannot use the same MANET and VANET clustering methods due to their unique characteristics. Instead, new techniques need to be made, or existing practices need to be changed to consider the unique features of a UAV network. UAVs were grouped into clusters, each of which had a specified number of UAVs and one of which was designated as a CH as by Bilal et al. [7]. Neighboring UAVs start by exchanging information about their node's specifications. Node information messages include a "zone ID" field that categorizes each UAV in a particular cluster. Each node keeps a database of connection quality information in the cluster, including information about its neighbors' distance, SNR, and latency. LEACH was developed to ensure balanced energy use and improve WSN efficiency by dividing the network into numerous clusters and rotating the CHs. LEACH is a MAC protocol based on TDMA, the first hierarchical routing protocol to use clustering. LEACH is a routing system based on clustering, where nodes form different clusters. Every cluster has a CH node that gathers data from cluster members and transmits it to the sink. The LEACH protocol [28] is divided into turns, each of which has two primary stages: setup and data transmission. The setup stage comprises selecting CHs, clustering, and creating the TDMA schedule for data transmission, which is carried out in the data transmission stage. LEACH-C picks CHs based on node residual energy and generates clusters accordingly in the network. As the name implies, LEACH-C [29] employs a centralized technique to choose the best candidate for CH from nominated nodes. Initially, sensor nodes report their remaining energy and position to the sink. The sink computes the average energy of nodes using this data. Then it chooses which nodes will be CHs. Thus, nodes with more leftover energy than average will be chosen as CHs in this round. The rest, however, remains a simple node. Centralization improves CH distribution and selection and the number of communications between nodes and sinks, which improves the network life-cycle. Like LEACH, LEACH-C transfers data to the sink in a single hop, reducing CH lifespan. The number of CHs varies from turn to turn in the LEACH technique, but in the LEACH-C approach, it is fixed.

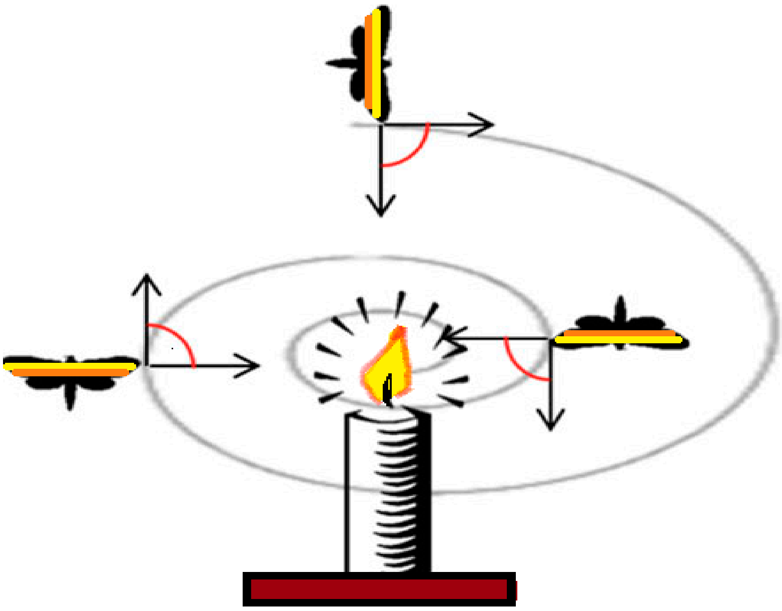

Figure 5. Spiral flying path.

CBLADSR (cluster-based location aided dynamic source routing) is a new routing technique suggested by Shi et al. [8]. Using three parameters, CBLADSR selects the CHs. A CH will be chosen among cluster members with a low-speed ratio, a high energy level, and several neighbor nodes. Each node in the cluster has a neighbor table, which keeps track of all of the other nodes in the group. To communicate between clusters, CBLADSR employs short and long-range transmissions. As part of their research, Zang and Zang [11] came up with an algorithm for mobility prediction. Each node keeps track of its one-hop neighbors in a neighbor table. The neighbor table also contains the probability of how long a node will remain in its table. A dictionary tree structure is used to calculate the probability. This neighbor node's probability and time determine when the link will end. Neighboring nodes' LET probabilities and degrees are used to create a weight for each node. A CH will be chosen from among the nodes based on the node's weight [10–13].

Clustering in ad-hoc networks has also been performed using artificial intelligence approaches such as ant colony optimization (ACO) [17,20], particle swarm optimization (PSO) [13,18], and grey wolf optimization (GWO) [14]. Numerous options are generated by using these methods. Nodes that potentially serve as CHs are included in each solution. The future orientation of ad-hoc networks was also examined by Bilal et al. [15] and Rizwan et al. [16]. In addition, these ideas are based on the nodes' mobility. Peer et al. [23] discussed the routing method using the fuzziness in wireless multi-hop networks. Nadeem et al. [19] came up with a way to choose a CH by a recommendation system that takes information from many different datasets and suggests an optimal CH node. As a result of this research, Adil et al. [20] developed a clustering technique based on the ACO (CACONET). ACO employs the social behavior of ants to find some food or a solution to a shared problem [17]. Another ACO-based algorithm called ACONET [20] has also been used as a clustering technique but has the same problem as [19,20,23]. GWOCNETs [25] simulate the grey wolf's leadership hierarchy and hunting strategy. Alpha, beta, delta, and omega are the four varieties of grey wolves. The alpha wolf is the most powerful in a pack and serves as the pack's leader. The rest of the group follows their lead to keep up with the alpha, beta, and delta wolves. Each wolf will be considered as a solution to the optimization issues. As an alpha solution, it is considered the best one. CAVDO [22] utilized the dragon fly algorithm's feature extraction to resolve the problem of clustering and selecting the best CH. The fundamental drawback of artificial intelligence approaches is that they require a large amount of computer power, even though they generate better outcomes. It takes a long time for them to get to an ideal solution. A vast number of random selections and random population sizes slow down the convergence of these methods. Due to the high mobility of UAVs, artificial intelligence approaches take too long to generate a correct result. Hence, they cannot be employed alone for a changing network architecture due to their low processing capacity. In Tables 1 and 2 various techniques have been compared using different metrics. Energy-constrained networks are not that suited for computationally intensive methods. So, our aim is to enhance the bio-inspired algorithm to perform efficiently and with less energy consumption in the FANET environment.

Table 1. Comparison of various algorithms.

Protocol	Year	Network Type	Cluster Method	Complexity	No of CH's	No of Nodes in Cluster	Mobility	Energy Efficiency
LEACH	2000	Homogenous	Distributed	Low	Uncertain	Unforeseeable	Inactive	Yes
LEACH-C	2002	heterogenous	Centralized	Low	Certain	Unforeseeable	Inactive	Yes
CBLADSR	2012	Heterogenous	Distributed	High	Uncertain	Unforeseeable	Inactive	Yes
CACONET	2016	Homogenous/heterogenous	Centralized	High	Uncertain	Unforeseeable	Inactive	Yes
PSONET	2011	Homogenous/heterogenous	Centralized	Very high	Uncertain	Unforeseeable	Inactive	Yes
GWOCNET	2014	Homogenous/heterogenous	Centralized	Very high	Uncertain	Unforeseeable	Inactive	Yes
CAVDO	2018	Heterogenous	Distributed	Medium	Uncertain	Unforeseeable	Inactive	Yes

Table 2. Comparability based on certain metrics.

Protocol	Energy model	Location Awareness	Connectivity to Bs	Link Quality Based	Connection Awareness	Collison Avoidance	Position of Base Station	Deployment Mode
LEACH	First order	No	Singe hop	Distance	No	No	Outside	Random
LEACH-C	First order	Yes	Singe hop	Distance	No	No	Outside	Random
CBLADSR	First order	No	Singe hop	Distance	Partially	No	Outside	Random and uniform
CACONET	First order	Yes	Singe hop	Distance	No	No	Outside	Random
PSONET	First order	Yes	Singe hop	Distance	No	No	Outside	Random
GWOCNET	First order	No	Singe hop	Distance	No	No	Outside	Random
CAVDO	First order	No	Singe hop	Distance	Partially	No	Outside	Random and non-uniform

3. Proposed Methodology

3.1. Network Building and Nodes Positioning

EEP-MFO is a FANET communication model that seeks to reduce the computational, energy consumption, and communicational cost to the absolute minimum. When the clustering mechanics are kept simple, the computational overhead may be lowered, and the communicational overhead can be reduced by lengthening the cluster's lifetime. Once the UAVs begin flying, network creation begins [11]. In addition to task-oriented sensors, unmanned aerial vehicles (UAVs) are equipped with GPS and height sensors. These two pieces of equipment are responsible for providing the 3D positioning information of the UAV node to the controller. As an additional assumption, we suppose that UAVs have three discrete transmission grid levels corresponding to communication ranges of 1000 to 3000 m for each of the four discrete power levels [12]. Nodes begin by selecting the highest power level available. Later on, nodes can choose the power level that is best appropriate for them based on their position and the position of their neighbors [22]. This strategy is used to conserve the energy of the node.

Figures 6 and 7 illustrate our proposed EECP-MFO algorithm's workflow. Given fitness values, random positions in solution space (m × n) are allocated to each moth and their respective moth arrays. Arrays and matrixes are produced in the same way as flames. The flame matrix contains the best value for the moth discovered [4]. Every time the best moth against its flame is found, the procedure iterates until it reaches the ideal number of search space flames. After each successful iteration, the moth position is updated. The quest for an optimal solution is similar to a moth's journey across the solution space. The fitness values are kept in the order of the m × n solution space, starting with the random location provided to each moth during the startup phase and continuing with the moth array. Similarly, the flame matrix and related array are created. The flame matrix contains the value of the moth that has been discovered to be the best so far [24]. Moths are moved around in a solution space to find the best solution or end the search procedure.

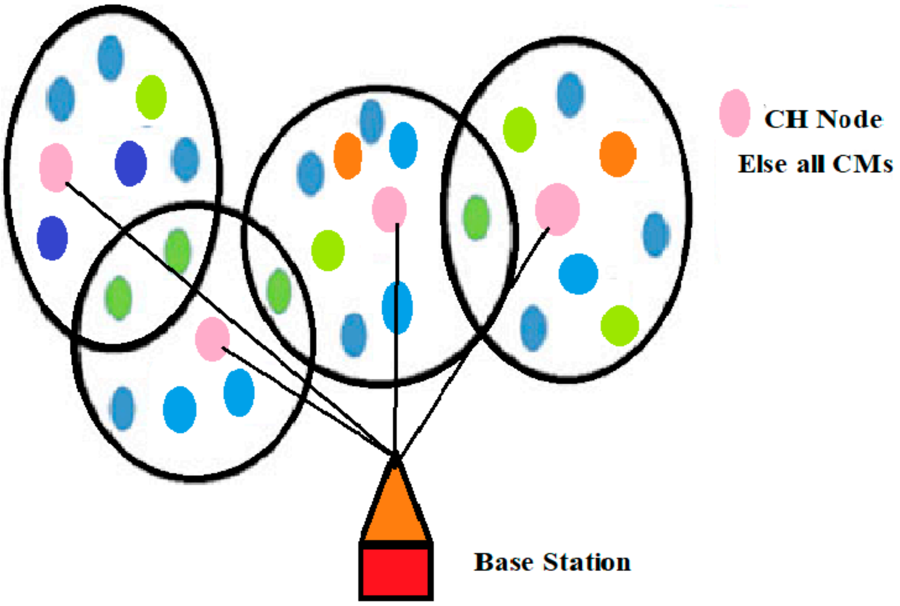

Figure 6. Clustering structure.

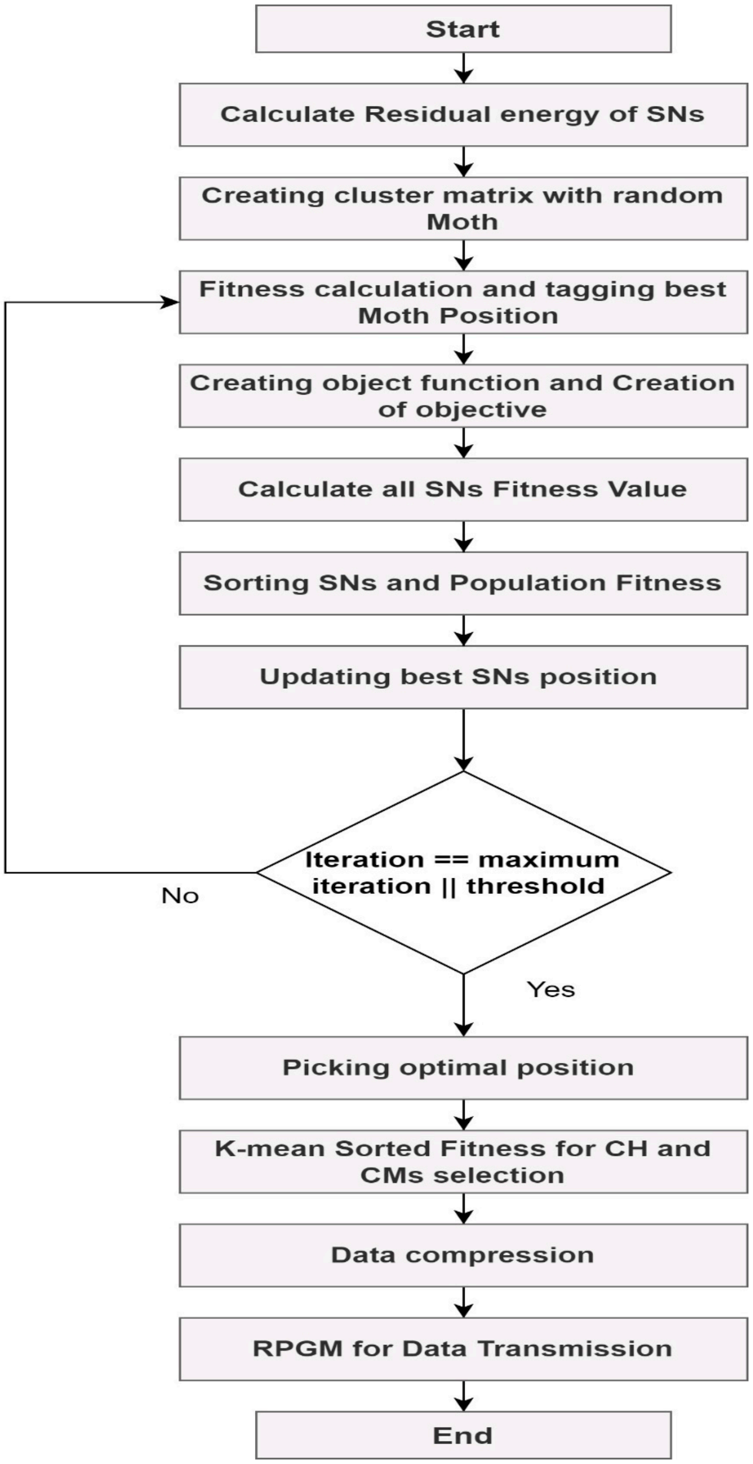
Start
Calculate Residual energy of SNs
Creating cluster matrix with random Moth
Fitness calculation and tagging best Moth Position
Creating object function and Creation of objective
Calculate all SNs Fitness Value
Sorting SNs and Population Fitness
Updating best SNs position
Iteration == maximum iteration || threshold
No
Yes
Picking optimal position
K-mean Sorted Fitness for CH and CMs selection
Data compression
RPGM for Data Transmission
End

Figure 7. Flowchart of proposed algorithm.

This procedure took advantage of the dimension of the search space's lower bound and upper bound boundaries. It is also used to determine the fitness value of each moth depending on their location in the search space [11,12,24]. Creating a fitness matrix is an iterative process; updated values are placed in the matrix in ascending order as the process progresses. The fitness matrix supplies the lower fitness value for each moth [13]. It is calculated by combining the moth's location in the search space with its fitness value, and this value is used to update the moth's position in the search space as the moth moves through the search space. Convergence was achieved using a linear decreasing factor "x" for the ideal solution. Selecting a Cluster Head (CH) follows the creation of a cluster, as seen in Figure 6. The selection of the CH can be made based on several factors, including the number of nodes, node density, residual energy, and load balancing factor [22,26]. Weights are assigned to these parameters before sending them to the fitness function. The selection process in EECP-MFO relies heavily on using a fitness function. Extending the cluster's lifespan by picking the optimal CH, reducing network energy, and avoiding excessive broadcast overhead [26].

The fitness value for the EECP-MFO algorithm is calculated using the following Equation (1):

$$\text{Fitness Function} = \frac{\text{W1} \times \text{Residual}_{\text{energy}}}{(\text{W2} \times \text{Avg}_{\text{Distance}})\text{W3} \times \text{Delta\_Difference}} \tag{1}$$

$$\text{Residual}_{\text{energy}} = \text{Initial energy} - \text{Consumed energy} \tag{2}$$

$\text{Residual}_{\text{energy}}$ represents the FANETS nodes' remaining energy calculated by using Equation (2), $\text{Avg}_{\text{Distance}}$ is the average distance between nearby nodes, and Delta_Difference represents the load balancing factor (LBF). Clusters with equal numbers of members are only possible if everything goes according to plan. However, it is challenging to achieve in a real-world environment since sensor nodes move around and modify their properties. The load balancing factor is calculated using the delta difference. Weights for energy, average distance, and delta difference are represented by the variables W1, W2 and W3. In an ideal world, all CHs would have an equal number of nodes. However, this is not realistic in real-world settings since nodes change their position regularly and change their neighborhood association degree. When a node's degree of the neighborhood varies from the ideal degree, this is called the delta difference. The exemplary nodes of SNs from their neighbors are calculated using the following formula:

$$\text{Delta\_Difference} = \text{ABS}(\text{Ideal\_degree} - \text{Node\_degree}) \tag{3}$$

Suppose the selection criteria for CH are static, and a single parameter has the potential to bias the fitness function. In that case, it is possible to make an incorrect choice of CH. EECP-MFO assigns weight to parameters dynamically depending on the circumstance to mitigate the skewing problem and reduce the negative influence on fitness values due to the skewing problem. First, each parameter value was normalized between 0 and 10.

According to Equation (3), the negative impact of each parameter variation is determined for each parameter deviation.

$$\text{dev(i)} = \text{ABS}(\text{mean} - \text{parameters(i)}) \tag{4}$$

To account for deviation from the mean, penalized outlier parameters are employed in Equation (3), and these are used to derive updated values for parameters. Another Equation (5) is used to penalize the outlier penalty, and it has the following formula:

$$\text{w(i)} = 1/\text{dev(i)} \tag{5}$$

The sum of all weights is equal to "1." Equation may be used to compute the fitness of each node for all Sensor nodes using Equation (1).

3.2. Cluster Formulation and CH selection with K-Means Sorted Fitness

When it comes to the actual clustering and the selection of CHs, K-means sorted fitness (a variation on K-means density) is used. K-means density is utilized in conjunction with K-means for the first selection of CH/centroids. Input parameter for K-means computes the "k" ideal centroids placement depending on the neighborhood of data points, as seen in Figure 8. The K-means algorithm [30,31] can swiftly arrive at an optimal solution due to the pre-computed centroids.

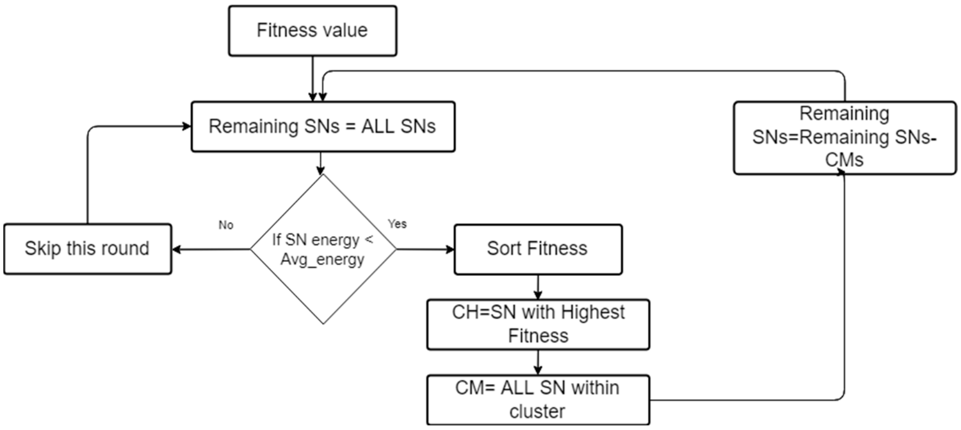
Fitness value
Remaining SNs = ALL SNs
Remaining SNs=Remaining SNs-CMs
Skip this round
No
If SN energy < Avg_energy
Yes
Sort Fitness
CH=SN with Highest Fitness
CM= ALL SN within cluster

Figure 8. Flowchart of K-means sorted fitness.

There is a set value for “k” in K-means density. However, with FANETs, the nodes move around often, and the network topology varies. Node positions and transmission range determine how many clusters are needed in a given network configuration. When choosing the size and number of clusters in K-means sorted fitness, we consider the transmission range of CHs [32,33]. The nodes’ fitness values are supplied into K-means sorted fitness, which returns CHs and the members connected with them as output, and the flow of this process can be seen in Figure 8. By keeping the clustering function as basic as feasible, EECP-MFO aims to minimize the computing effort required. Before the selection of CH occurs, we try to make it more efficient as here, and we already had residual energy computed for all nodes. So, we try to evaluate the average residual energy first. Suppose a candidate node for CH has less than the average residual energy. In that case, it is skipped for the current round of the CH selection process. The fitness value is used to make decisions in this function [34]. To choose the best CHs, a more precise fitness value must be calculated first in this scenario. In the beginning, the fitness value is ordered in decreasing order for more usefulness. The node will take the CH position with the fitness values. All additional nodes within the transmission range of the selected CH will be designated as cluster members (CMs). The elected node for CH will not be considered as the remaining nodes. After that, the Selection process of CH is repeated until there are no more nodes left to be chosen. All nodes have assumed their respective roles, whether as cluster leaders or cluster members [35].

3.3. Data Compression and Network Commination

a. Data compression Our proposed protocol is used for captured data, primarily images and videos. So, to reduce the data transmission energy, we need to use a compression algorithm. Our proposed protocol uses MRCQ (multi-resolution compression and query) image-based compression approach [26]. Sensor nodes are organized in a hierarchy to establish multiresolution summaries of sensed data in the network [11]. Lower-resolution summaries are transmitted to the sink, whereas high-resolution outlines remain in the network and can be accessed for further analysis [32]. As a result, MRCQ has lower implementation costs and may be used with low-cost sensor systems [33,34].
b. Node Movement and Network Communications Communication and data transfer between nodes begin when clustering is complete. Whether a node inside a cluster [35], a node across the cluster, or the base station is the intended destination for the data, the CH is responsible for getting it there [36,37]. EECP-MFO adheres to the RPGM [38] reference point group mobility model. There is a point of reference for all nodes in RPGM that they all will follow. EECP-MFO considers a reference point for the CHs, and all CMs adjust their positions under how their respective CHs move.

4. Experimental Results and Analysis

Performance comparisons of our proposed clustering algorithms named EECP-MFO with GWO-based clustering, ACO-based clustering, and PSO-based clustering are presented in this part. For the assessment and evaluation process, we employ critical vital criteria, including the number of clusters, the probability of success, the time it takes to form a cluster, the longevity of the cluster, and the amount of energy consumed. The experiments are carried out using the MATLAB programming language. A variety of parameters are included to perform the simulation, as shown in Table 3. We used ten simulation runs in each set to develop the final results.

Table 3. Simulation parameters.

Parameters	Values
Grid Size	1000 × 1000 m^2, 2000 × 2000 m^2 and 3000 × 3000 m^2
Density of Connected Nodes	20, 30, 40, 50, 60
Minimum Distance Between Nodes	5 m
Mobility Model	Reference Point Mobility Model
Simulation Runs	10
Simulation Time	120 s
Position Exchange Interval	2 s
Node Energy Level at Start Time	80-Watt Hour
Transmission Range	Dynamic
Transmission Frequency	2.45 GHz
Constant Bit Rate	100 kbps
Receiver Sensitivity	−90 dBm
W1 +W2 + W3	1

4.1. Cluster Building Time

The algorithm's computational complexity is measured by its cluster construction time during the clustering process. Clustering algorithms often use nodes with accompanying fitness values as input to choose the CH and associate members. Cluster building time refers to the time it takes an algorithm to go from receiving inputs to creating outputs. The performance of an unmanned aerial vehicle (UAV) would be negatively impacted if the time it takes to assemble a cluster is too long. The longer it takes to create a cluster, the more energy is consumed, resulting in a shorter lifespan for UAVs in a network. Unlike ACO and GWO, which start with various solutions and converge to the best solution, our EECP-MFO quickly connects to the ideal solution. This makes it outperform ACO, PSO, and GWO. A shorter cluster building time has been achieved to shorten the energy and time it takes for the UAVs to pick a path. It can be noted from Figures 9–11 that the greater the grid size, the more UAVs in a network will take more time to execute the clustering algorithm, which increases cluster building time. In Figure 11, we do not see any massive difference in clustering time for the cluster building by our proposed algorithm, making it superior to the other algorithms.

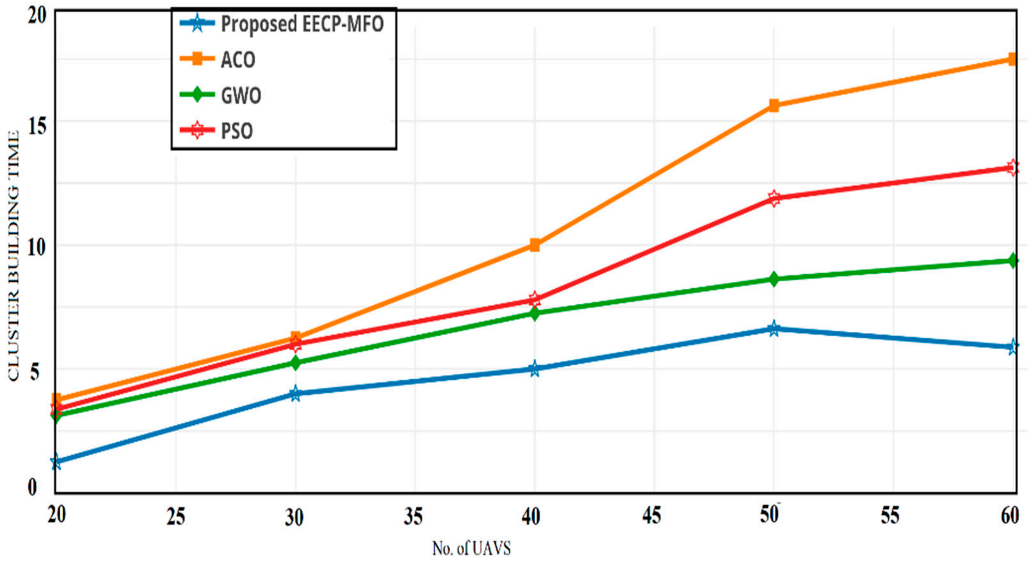

Figure 9. Time for clustering vs. number of UAVs (1000 × 1000 m^2).

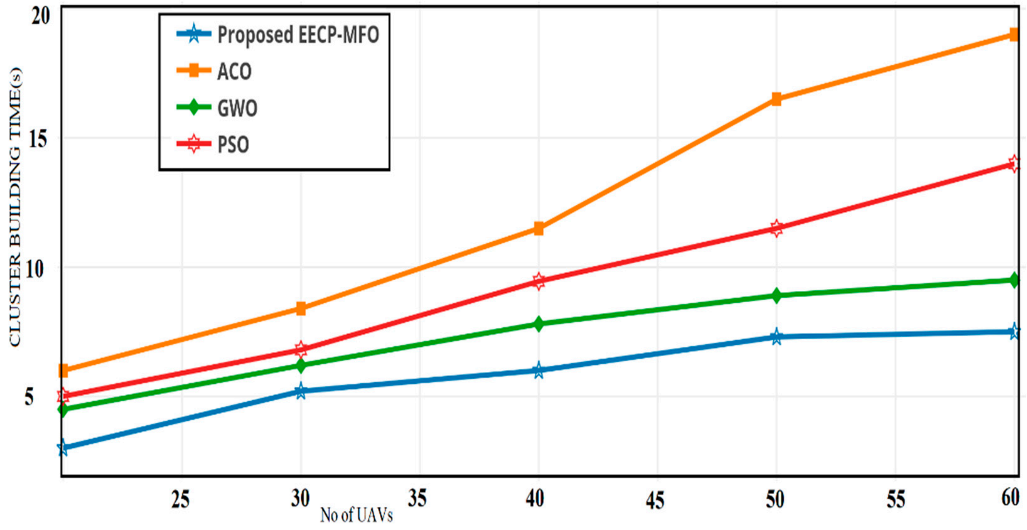

Figure 10. Time for clustering vs. number of UAVs (3000 ×3000 m^2).

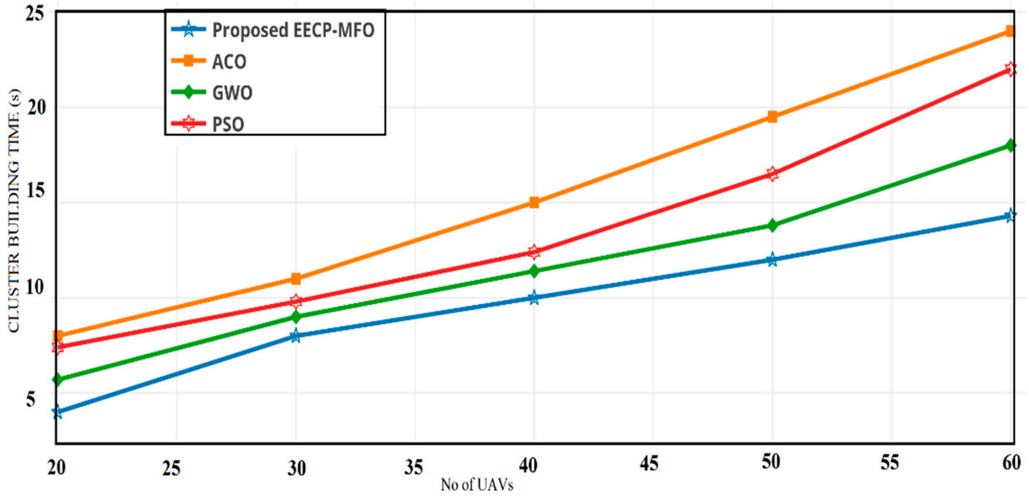

Figure 11. Time for clustering Vs number of UAVs (2000 ×2000 m^2).

4.2. Energy Consumption

In UAVs, energy is one of the most limited resources. Due to a small dry-cell battery, it is possible to fly micro-UAVs for just a few minutes, which is from 25 and 30 min. Due to their low battery power, these unmanned aerial vehicles (UAVs) have several challenges. To maximize the lifespan of a UAV and increase probability of success, it is essential to maximize energy efficiency. In UAVs, three processes are responsible for energy loss: actuating the motor control system, energy used by various sensors, and energy used for inter-UAV communication. UAVs uses most energy since they communicate with each other. The total amount of energy is used in communication, including transmission and reception, can be computed using Equations (6)–(9).

$$\text{Energy}_{\text{total}} = \text{Energy}_{\text{com}} + \text{Energy}_{\text{flying}} + \text{Energy}_{\text{sensors}} \tag{6}$$

$$\text{Energy}_{\text{com}} = \text{Energy}_{\text{Tx}} + \text{Energy}_{\text{Rx}} \tag{7}$$

$$\text{Energy}_{\text{Tx}} = \text{Energy}_{\text{TRC}} \times \text{k} + \text{Energy}_{\text{A}} \times \text{k} \times \text{D}^2 \tag{8}$$

$$\text{Energy}_{\text{Rx}} = \text{Energy}_{\text{TRC}} \times \text{k} \tag{9}$$

Here energy consumption refers to the sum of energy consumed during transmission and reception of data. While running transmitter and receiver circuits, $\text{Energy}_{\text{TRC}}$ is the energy dissipation. Energy_{A} is the amount of energy used by the transmit amplifier to send data to the other end. The number of bits sent and received is represented by “L,” whereas the distance between the sending and receiving nodes is represented by “d.” $\text{Energy}_{\text{flying}}$ is energy consumed for flying a node, in this scenario we are using homogenous nodes so there would be same energy consumed for flying by all nodes.

Here whole energy consumption refers to the sum of energy consumed during transmission and reception of data. By using Equation (6), we can calculate the total energy. In our case we are using homogenous nodes so there would be same energy for flying in all nodes.

We calculate the whole energy consumption by EECP-MFO, PSO, ACO, and GWO over the course of 120 s. As the number of UAVs and the grid size increase, the network’s energy usage also rises. Figures 12–14 clearly shows that EECP MFO beats the other three methods. The lower EECP MFO energy usage is since the right CHs were chosen, and the CH number is within the optimal limit.

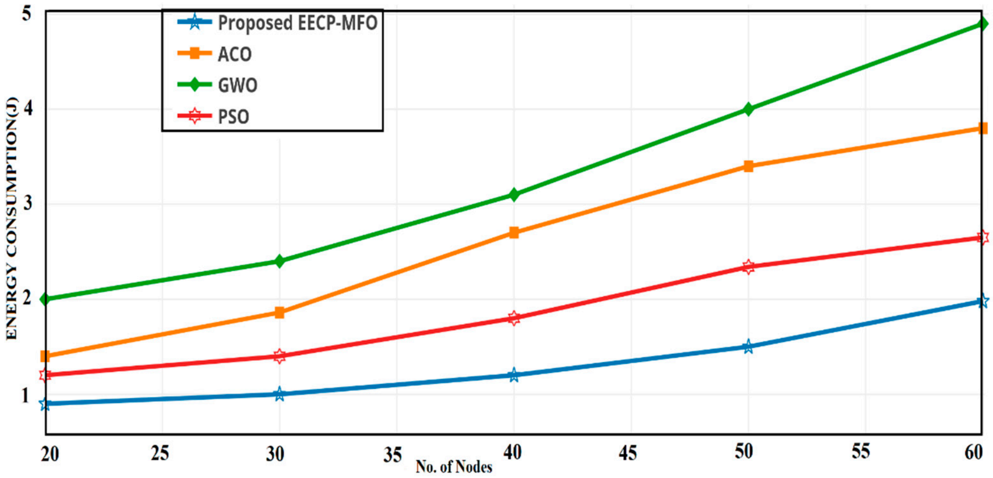

Figure 12. Energy consumption vs. number of nodes (1000 × 1000 m^2).

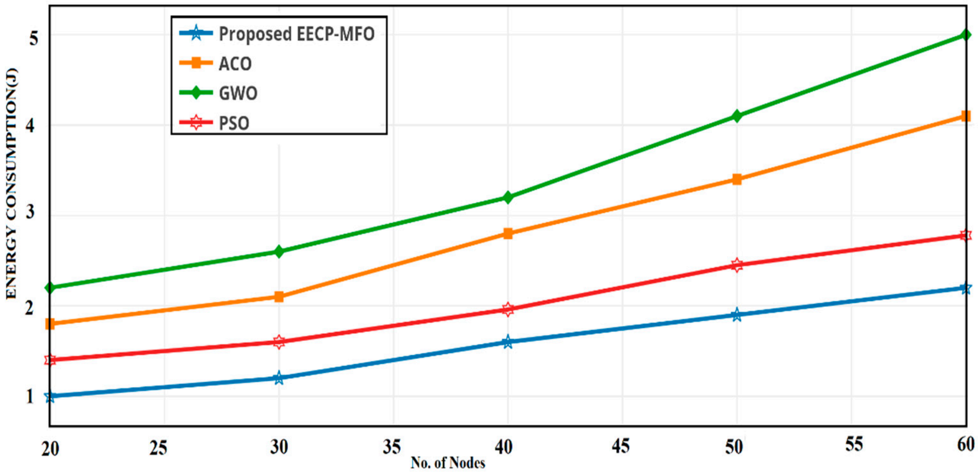

Figure 13. Energy consumption vs. number of nodes (2000 × 2000 m^2).

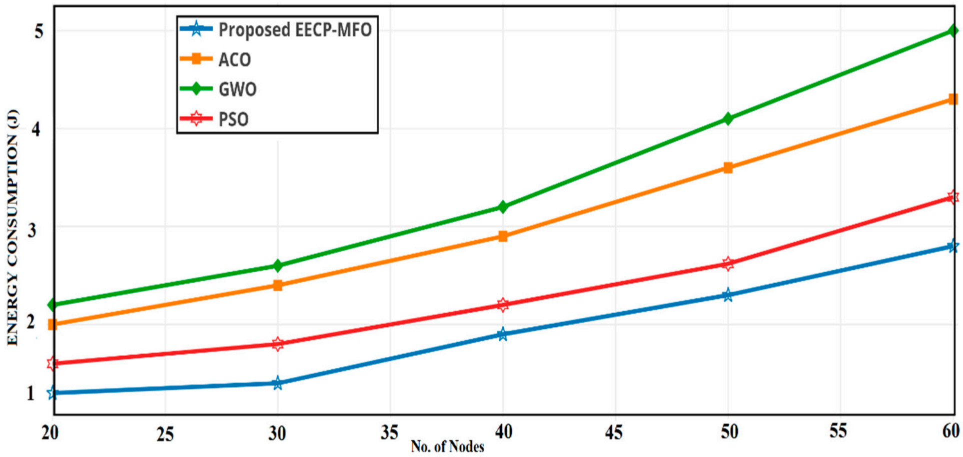

Figure 14. Energy consumption vs. number of nodes (3000 × 3000 m^2).

4.3. Probability of Success

The probability of delivery success is another significant metric to consider when evaluating the performance efficiency of the clustering process.

Based on the average number of hops per packet, this metric indicates how well a packet is delivered from intermediate nodes to base station in the network. Figure 15 shows that by increasing the number of UAVs in the network, the network density grows, increasing the chance of successful delivery. We can also say that packet loss ratio drops as the number of unmanned aerial vehicles (UAVs) in the network grows.

4.4. Cluster Lifetime

The cluster lifetime is when the cluster exists, from its formation to its dissolution. The UAV with the best fitness value is selected as the cluster leader when the clustering method is completed. When different operations are performed on the UAV, the fitness value of the UAV steadily declines over time. The CH election process is triggered again when the value falls below a certain threshold. A shorter cluster lifetime indicates that the clustering method must be conducted more frequently, increasing the amount of communication and computing that must take place in the network. As seen in Figures 16–18, our EECP-MFO algorithm outperforms the GWO, PSO, and ACO algorithms in terms of performance. As a result of these findings, it can be concluded that increasing the number of UAVs in the network reduces the longevity of the cluster. As the number of SNs in the network grows, the network's topology changes more often, making the grid less reliable.

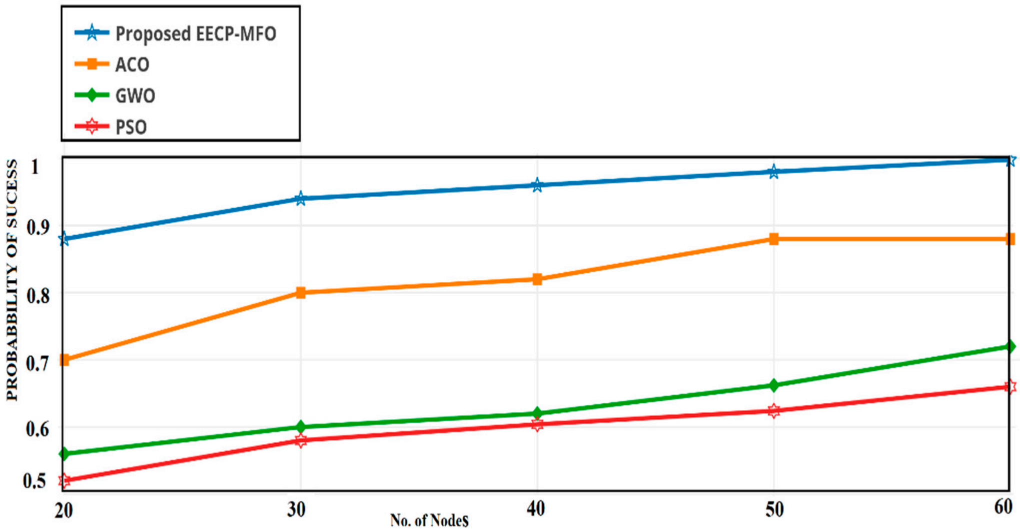

Figure 15. Probability of success for packet.

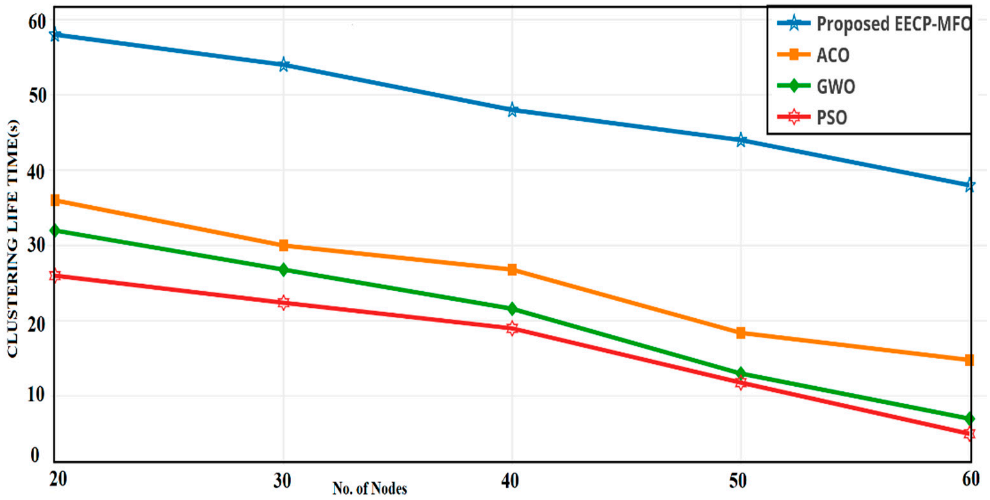

Figure 16. Cluttering lifetime vs. number of nodes (1000 × 1000 m^2).

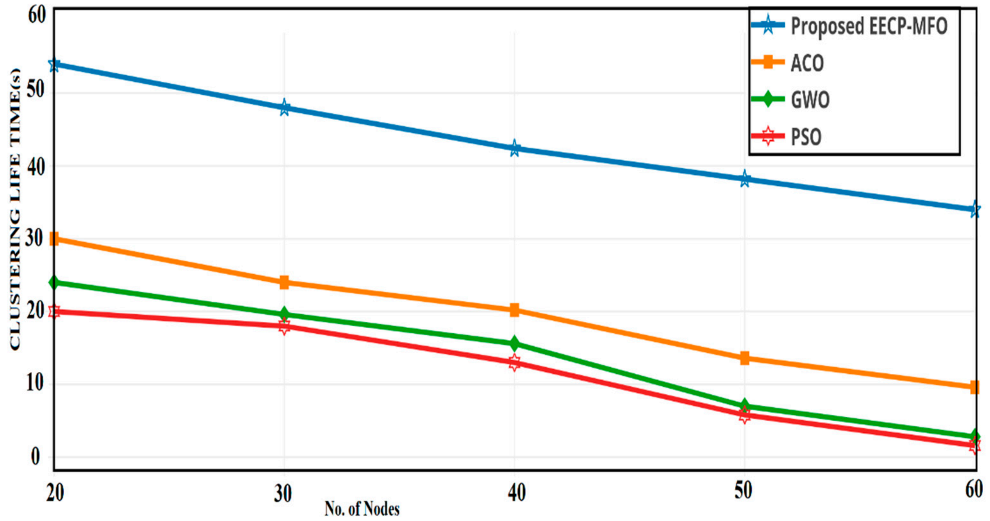

Figure 17. Cluttering lifetime vs. number of nodes (2000 × 2000 m^2).

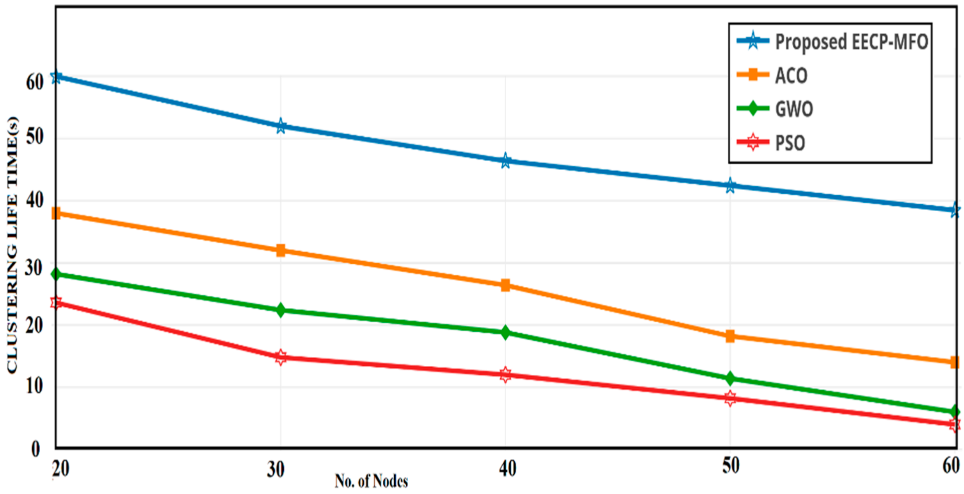

Figure 18. Cluttering lifetime vs. number of nodes (3000 × 3000 m^2).

4.5. Consistency of Cluster Heads

The energy efficiency of the protocol is greatly influenced by the number of cluster heads in use. If the number of cluster heads is too less, the data transmission duration between sensor nodes and the cluster head will be too long, resulting in increased energy usage. Excessive data transmission by the cluster will also increase energy usage. As the cluster heads rose, the overall network energy demand increased.

In Figure 19, we can see that ACO has many variances in the cluster head number since the cluster headcount is randomly reliant on the threshold function model, which is also random, which causes the cluster headcount to fluctuate a lot. The figure shows that the cluster headcount swings in the range of $4 \leq k \leq 19$ in the ACO and $2 \leq k \leq 18$ in the GWO, $3 \leq k \leq 17$ in PSO, but the suggested protocol has a headcount of $3 \leq k \leq 12$, which is the most appropriate as compared to others.

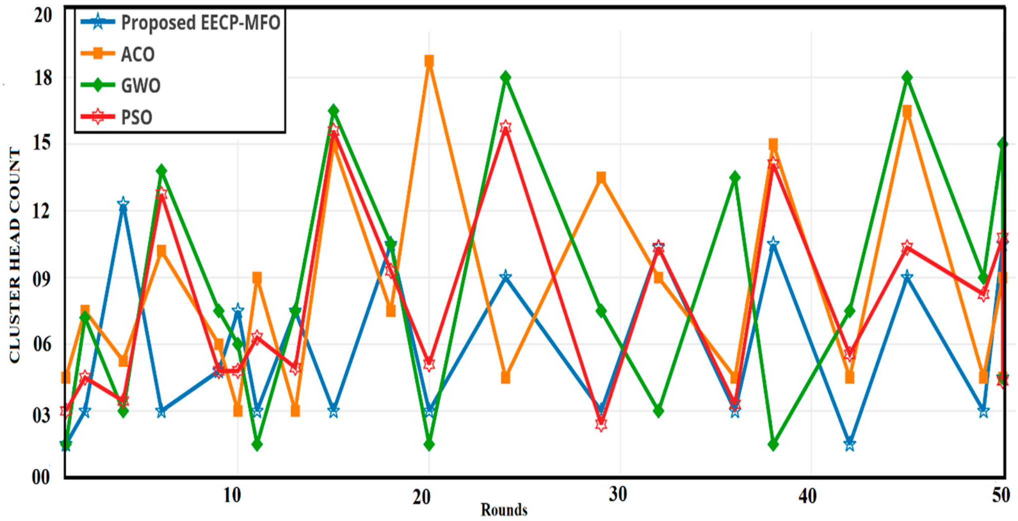

Figure 19. Consistency in number of cluster heads.

5. Discussion

The EECP-MFO model was presented in this research to overcome the routing problem in FANETS. Highly energy efficient and optimized solutions in search space can be found using the MFO method. In FANET, the EECP-MFO method improves cluster building time, reduces energy consumption, and has a high probability of packet delivery success with a balanced number of cluster heads. In addition, it decreased broadcasting efforts that were not essential and helped to minimize the cost of routing and preserve the energy of UAVs by regulating their transmission range. Figure 20 depicts a side-by-side comparison of the three existing methods with our own proposed protocol. Every assessing parameter shows that EECP-MFO outperforms other approaches since it can give an optimum node for CH selection and the best neighbor nodes selection for a cluster in an efficient and optimal way. However, if we talk about the second most superior algorithm, we cannot reach a single algorithm as GWO [14,25] performs better than the PSO [18] and ACO [17,20] in the case of clustering time. However, in the case of energy efficiency and probability of success, ACO performs better than GWO and PSO [36].

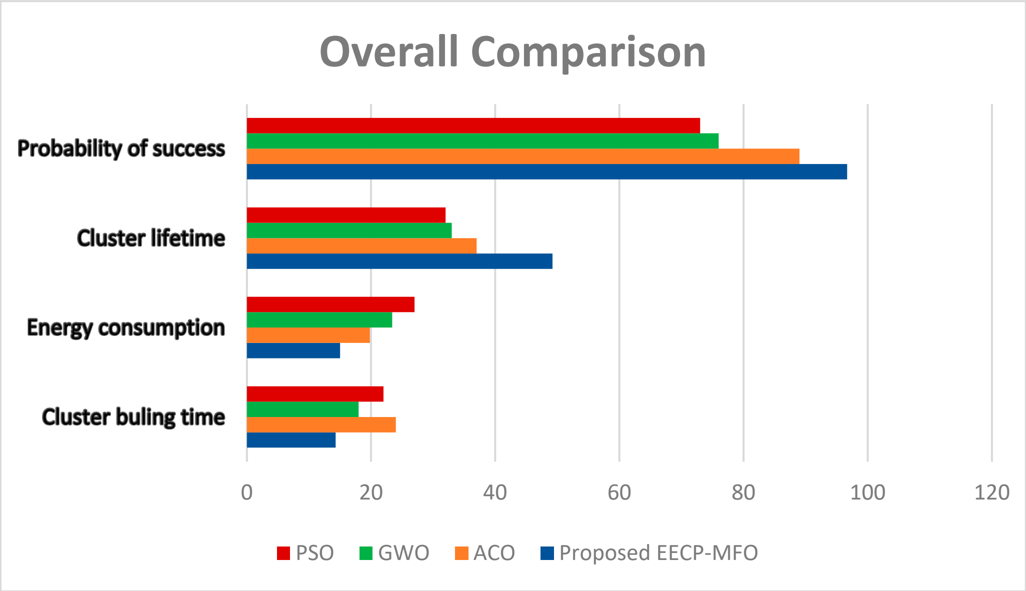

Figure 20. Comparison using various metrics.

One of the critical limitations of the proposed algorithm is tuning parameters, as it can bias the fitness function for selecting a CH and also leads to the skewing problem. Improper selection of those parameters will result in a poor clustering mechanism, leading to high energy consumption and a weak node lifetime [37–40]. However, secondly, the proposed algorithm does not provide any aggregation technique by which it can ignore the identical packet for processing as they consume substantial computation efforts and UAV's energy. A packet scheduling system is required to overcome the issues mentioned above, also tackling the congestion control concern. These limitations can be addressed in future studies [41]. Moreover, these limitations will encourage the researchers to take up this research problem. As part of our future work, we will also attempt to construct a dynamic model in a heterogeneous environment using the Moth flame optimization technique to make our proposed model more widely applicable for improving the lifetime of WSNs and providing an energy-efficient for the application and development of WSNs and IoT [42].

6. Conclusions

The EECP-MFO model as an efficient and optimized clustering algorithm for FANETS was proposed in this study. Fast-moving nodes have two significant drawbacks: limited energy and inefficient routing. By adjusting the transmission range and correctly clustering the network, we can improve routing and conserve energy on UAVs. The EECP-MFO approach proposed an evolutionary technique for cluster optimization to address the FANET routing problem. Using the EECP-MFO method for FANETs is a cost-effective way to reduce the number of clusters as it cuts down on the amount of unneeded broadcasting and decreases the cost of routing and speed of routing. The efficiency of the proposed EECP-MFO algorithm was assessed and validated as the experiments on the simulations were performed and monitored with variable transmission ranges of SNs. EECP-MFO gave near-optimal solutions in the FANETS topological restrictions and created minimum clusters in the search space. GWO, PSO, and ACO are all well-known evolutionary algorithms. However, EECP-MFO is the best solution for the challenges under investigation. It is the clear winner in cluster longevity and the number of clusters; however, EECP-MFO also succeeds in time and energy consumption in establishing clusters. In our proposed algorithm, the UAVs can process identical data packets, which exhausts the UAV's limited resources and degrades network performance. A packet scheduling system is required to solve these difficulties and improve packet management in FANETs. In the future, we will focus on this research area for congestion control in resource constraint UAV networks.

Author Contributions: Conceptualization: S.B. (Salil Bharany), S.S.; methodology: S.B. (Salil Bharany), S.B. (Surbhi Bhatia), M.S.; validation: S.B. (Salil Bharany), S.B. (Surbhi Bhatia), S.A.L., M.K.I.R.; formal analysis: M.K.I.R., M.S.; investigation: S.B. (Salil Bharany), S.S., M.S.; resources: S.B. (Salil Bharany), S.A.L., M.K.I.R.; data curation: S.S., S.B. (Surbhi Bhatia); writing original draft preparation: S.B. (Salil Bharany); writing review and editing: S.B. (Salil Bharany), S.S., M.K.I.R., S.B. (Surbhi Bhatia); visualization: S.B. (Salil Bharany); supervision: S.S., S.A.L.; project administration: S.B. (Salil Bharany), S.S.; funding acquisition: S.A.L., M.K.I.R. All authors have read and agreed to the published version of the manuscript.

Funding: This research work received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The study did not report any data.

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

References

1. Tariq, R.; Iqbal, Z.; Aadil, F. IMOC: Optimization technique for drone-assisted VANET (DAV) based on moth flame optimization. *Wirel. Commun. Mob. Comput.* **2020**, 1–29. [CrossRef]
2. Aadil, F.; Raza, A.; Khan, M.F.; Maqsood, M.; Mehmood, I.; Rho, S. Energy aware cluster-based routing in flying ad-hoc networks. *Sensors* **2018**, *18*, 1413. [CrossRef] [PubMed]
3. Bharany, S.; Sharma, S.; Badotra, S.; Khalaf, O.I.; Alotaibi, Y.; Alghamdi, S.; Alassery, F. Energy-efficient clustering scheme for flying ad-hoc networks using an optimized LEACH protocol. *Energies* **2021**, *14*, 6016. [CrossRef]
4. Han, X.; Quan, L.; Xiong, X.; Almeter, M.; Xiang, J.; Lan, Y. A novel data clustering algorithm based on modified gravitational search algorithm. *Eng. Appl. Artif. Intell.* **2017**, *61*, 1–7. [CrossRef]
5. Bhatia, S.; Alam, S.; Shuaib, M.; Alhameed, M.H.; Jeribi, F.; Alsuwailem, R.I. Retinal Vessel Extraction via Assisted Multi-Channel Feature Map and U-Net. *Front. Public Health* **2022**, *10*, 355. [CrossRef] [PubMed]
6. Tavazoei, M.S.; Haeri, M. Comparison of different one-dimensional maps as chaotic search pattern in chaos optimization algorithms. *Appl. Math. Comput.* **2007**, *187*, 1076–1085. [CrossRef]
7. Alazab, M.; Lakshmanna, K.; Reddy, T.; Pham, Q.V.; Maddikunta, P.K.R. Multi-objective cluster head selection using fitness averaged rider optimization algorithm for IoT networks in smart cities. *Sustain. Energy Technol. Assess.* **2021**, *43*, 100973. [CrossRef]
8. Shi, N.; Luo, X. A novel cluster-based location-aided routing protocol for UAV fleet networks. *Int. J. Digit. Content Technol. Appl.* **2012**, *6*, 376.
9. Esmin, A.A.A.; Coelho, R.A.; Matwin, S. A review on particle swarm optimization algorithm and its variants to clustering high-dimensional data. *Artif. Intell. Rev.* **2015**, *44*, 23–45. [CrossRef]
10. Yang, D.; Li, G.; Cheng, G. On the efficiency of chaos optimization algorithms for global optimization. *Chaos Solitons Fractals* **2007**, *34*, 1366–1375. [CrossRef]
11. Zang, C.; Zang, S. Mobility prediction clustering algorithm for UAV networking. In *2011 IEEE Globecom Workshops (GC Wkshps)*; IEEE: Piscataway, NJ, USA, 2011; pp. 1158–1161.
12. Bhushan, B.; Sahoo, G. Recent Advances in Attacks, Technical Challenges, Vulnerabilities and Their Countermeasures in Wireless Sensor Networks. *Wirel. Pers. Commun.* **2018**, *98*, 2037–2077. [CrossRef]
13. Haque, A.K.M.B.; Bhushan, B.; Dhiman, G. Conceptualizing smart city applications: Requirements, architecture, security issues, and emerging trends. *Expert Syst.* **2021**. [CrossRef]
14. Mirjalili, S. Dragonfly algorithm: A new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems. *Neural Comput. Appl.* **2016**, *27*, 1053–1073. [CrossRef]

15. Haider, B.; Henna, S.; Gul, A.; Aadil, F. A survey on mobility management techniques in VANETs. In *2016 IEEE International Conference on Computer and Information Technology (CIT), Nadi, Fiji, 8–10 December 2016*; IEEE: Piscataway, NJ, USA, 2016; pp. 125–133.
16. Aadil, F.; Rizwan, S.; Akram, A. *Vehicular Ad Hoc Networks (VANETs), Past Present and Future: A Survey*; Springer: Berlin/Heidelberg, Germany, 2011.
17. Arif, M.; Rahmani, K.I. Adaptive ARA (AARA) for MANETs. In *2012 Nirma University International Conference on Engineering (NUiCONE), Ahmedabad, India, 6–8 December 2012*; IEEE: Piscataway, NJ, USA, 2012; pp. 1–6.
18. Sharma, R.; Kaushik, B.; Gondhi, N.K.; Tahir, M.; Rahmani, M.K.I. Quantum Particle Swarm Optimization Based Convolutional Neural Network for Handwritten Script Recognition. *Comput. Mater. Contin.* **2022**, *71*. [CrossRef]
19. Ishtiaq, S.; Majeed, N.; Maqsood, M.; Javed, A. Improved scalable recommender system. *Nucleus* **2016**, *53*, 200–207.
20. Aadil, F.; Bajwa, K.B.; Khan, S.; Chaudary, N.M.; Akram, A. CACONET: Ant colony optimization (ACO) based clustering algorithm for VANET. *PLoS ONE* **2016**, *11*, e0154080. [CrossRef]
21. Aadil, F.; Khan, S.; Bajwa, K.B.; Khan, M.F.; Ali, A. Intelligent clustering in vehicular ad hoc networks. *KSII Trans. Internet Inf. Syst. (TIIS)* **2016**, *10*, 3512–3528.
22. Aadil, F.; Ahsan, W.; Rehman, Z.U.; Shah, P.A.; Rho, S.; Mehmood, I. Clustering algorithm for internet of vehicles (IoV) based on dragonfly optimizer (CAVDO). *J. Supercomput.* **2018**, *74*, 4542–4567. [CrossRef]
23. Aldabbagh, G.; Shah, P.A.; Hasbullah, H.B.; Aadil, F.; Awan, K.M.; Marwat, F. Fuzzy logic based enhanced AOMDV with link status classification for efficient multi-path routing in multi-hop wireless networks. *J. Comput. Theor. Nanosci.* **2017**, *14*, 620–630. [CrossRef]
24. Bhushan, B.; Sahoo, G. E^2SR^2: An acknowledgement-based mobile sink routing protocol with rechargeable sensors for wireless sensor networks. *Wirel. Netw.* **2019**, *25*, 2697–2721. [CrossRef]
25. Mirjalili, S.; Mirjalili, S.M.; Lewis, A. Grey wolf optimizer. *Adv. Eng. Softw.* **2014**, *69*, 46–61. [CrossRef]
26. Wang, Y.-C.; Hsieh, Y.-Y.; Tseng, Y.-C. Multiresolution spatial and temporal coding in a wireless sensor network for long-term monitoring applications. *IEEE Trans. Comput.* **2009**, *58*, 827–838. [CrossRef]
27. Shuaib, M.; Hassan, N.H.; Usman, S.; Alam, S.; Bhatia, S.; Mashat, A.; Kumar, A.; Kumar, M. Self-Sovereign Identity Solution for Blockchain-Based Land Registry System: A Comparison. *Mob. Inf. Syst.* **2022**, *2022*, 8930472. [CrossRef]
28. Heinzelman, W.R.; Chandrakasan, A.; Balakrishnan, H. Energy-efficient communication protocol for wireless microsensor networks. In Proceedings of the 33rd Annual Hawaii International Conference on System Sciences, Maui, HI, USA, 4–7 January 2000.
29. Heinzelman, W.B.; Chandrakasan, A.P.; Balakrishnan, H. An application-specific protocol architecture for wireless microsensor networks. *IEEE Trans. Wirel. Commun.* **2002**, *1*, 660–670. [CrossRef]
30. Rahmani, M.K.I.; Ansari, M.A. A color based fuzzy algorithm for CBIR. *IET Conf. Publ.* **2013**. [CrossRef]
31. Rahmani, M.K.I.; Pal, N.; Arora, K. Clustering of Image Data Using K-Means and Fuzzy K-Means. *Int. J. Adv. Comput. Sci. Appl.* **2014**, *5*. [CrossRef]
32. Shuaib, M.; Hassan, N.H.; Usman, S.; Alam, S.; Bhatia, S.; Koundal, D.; Mashat, A.; Belay, A. Identity Model for Blockchain-Based Land Registry System: A Comparison. *Wirel. Commun. Mob. Comput.* **2022**, *2022*, 5670714. [CrossRef]
33. Naeem, A.; Javed, A.R.; Rizwan, M.; Abbas, S.; Lin, J.C.-W.; Gadekallu, T.R. DARE-SEP: A Hybrid Approach of Distance Aware Residual Energy-Efficient SEP for WSN. *IEEE Trans. Green Commun. Netw.* **2021**, *5*, 611–621. [CrossRef]
34. Reddy, T.; Swarna Priya, R.M.; Parimala, M.; Chowdhary, L.C.; Hakak, S.; Khan, Z.W. A deep neural networks based model for uninterrupted marine environment monitoring. *Comput. Commun.* **2020**, *157*, 64–75.
35. Guerrero-Sánchez, M.E.; Abaunza, H.; Castillo, P.; Lozano, R.; García-Beltrán, C.D. Quadrotor Energy-Based Control Laws: A Unit-Quaternion Approach. *J. Intell. Robot. Syst.* **2017**, *88*, 347–377. [CrossRef]
36. Shuaib, M.; Hassan, N.H.; Usman, S.; Alam, S.; Bhatia, S.; Agarwal, P.; Idrees, S.M. Land Registry Framework Based on Self-Sovereign Identity (SSI) for Environmental Sustainability. *Sustainability* **2022**, *14*, 5400. [CrossRef]
37. Rahmani, M.K.I. A graphical approach for image retrieval based on five layered CNNs model. In *2021 5th International Joint Conference on Advances in Computational Intelligence (IJCACI 2021), Online, 23–24 October 2021*; Jahangirnagar University: Dhaka, Bangladesh, 2021; p. 12.
38. Hong, X.; Gerla, M.; Pei, G.; Chiang, C.-C. A group mobility model for ad hoc wireless networks. In Proceedings of the 2nd ACM International Workshop on Modeling, Analysis and Simulation of Wireless and Mobile Systems, New York, NY, USA, 20 August 1999; pp. 53–60.
39. Xiao, X.; Jin, L.; Yang, Y.; Yang, W.; Sun, J.; Chang, T. Building fast and compact convolutional neural networks for offline handwritten Chinese character recognition. *Pattern Recognit.* **2017**, *72*, 72–81. [CrossRef]
40. Alam, S.; Shuaib, M.; Khan, Z.W.; Garg, S.; Kaddoum, G.; Hossain, S.M.; Zikria, B.Y. Blockchain-based Initiatives: Current state and challenges. *Comput. Netw.* **2021**, *198*, 108395. [CrossRef]
41. Iwendi, C.; Jalil, Z.; Javed, A.R.; Reddy, T.; Kaluri, R.; Srivastava, G.; Jo, O. Keysplitwatermark: Zero watermarking algorithm for software protection against cyber-attacks. *IEEE Access* **2020**, *8*, 72650–72660. [CrossRef]
42. Gadamsetty, S.; Ch, R.; Ch, A.; Iwendi, C.; Gadekallu, T.R. Hash-based deep learning approach for remote sensing satellite imagery detection. *Water* **2022**, *14*, 707. [CrossRef]